Durham University

Durham E-Theses

Bayesian Uncertainty Analysis and Decision Support for Complex Models of Physical Systems with Application to Production Optimisation of Subsurface Energy Resources

OWEN, JONATHAN,DANIEL

How to cite:

OWEN, JONATHAN,DANIEL (2022) *Bayesian Uncertainty Analysis and Decision Support for Complex Models of Physical Systems with Application to Production Optimisation of Subsurface Energy Resources*, Durham theses, Durham University. Available at Durham E-Theses Online: http://etheses.dur.ac.uk/14751/

Use policy

The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a link is made to the metadata record in Durham E-Theses
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Please consult the full Durham E-Theses policy for further details.

Academic Support Office, The Palatine Centre, Durham University, Stockton Road, Durham, DH1 3LE
e-mail: e-theses.admin@durham.ac.uk Tel: +44 0191 334 6107
http://etheses.dur.ac.uk

Bayesian Uncertainty Analysis and Decision Support for Complex Models of Physical Systems with Application to Production Optimisation of Subsurface Energy Resources

Jonathan Daniel Owen

A Thesis presented for the degree of
Doctor of Philosophy

Statistics and Probability Group
Department of Mathematical Sciences
Durham University
United Kingdom

December 2022

Dedicated to

Melanie and Peter Owen,
my parents.

Bayesian Uncertainty Analysis and Decision Support for Complex Models of Physical Systems with Application to Production Optimisation of Subsurface Energy Resources

Jonathan Daniel Owen

Submitted for the degree of Doctor of Philosophy

December 2022

Abstract

Important decision making problems are increasingly addressed using computer models for complex real world systems. However, there are major limitations to their direct use including: their complex structure; large numbers of inputs and outputs; the presence of many sources of uncertainty; which is further compounded by their long evaluation times. Bayesian methodology for the analysis of computer models has been extensively developed to perform inference for the physical systems. In this thesis, the Bayesian uncertainty analysis methodology is extended to provide robust decision support under uncertainty.

Bayesian emulators are employed as a fast and efficient statistical approximation for computer models. We establish a hierarchical Bayesian emulation framework that exploits known constrained simulator behaviour in constituents of the decision

support utility function. In addition, novel Bayesian emulation methodology is developed for computer models with structured partial discontinuities. We advance the crucial uncertainty quantification methodology to perform a robust decision analysis developing a technique to assess and remove linear transformations of the utility function induced by sources of uncertainty to which conclusions are invariant, as well as incorporating structural model discrepancy and decision implementation error. These are encompassed within a novel iterative decision support procedure which acknowledges utility function uncertainty resulting from the separation of the analysts and final decision makers to deliver a robust class of decisions, along with any additional information, for further consideration. The complete toolkit is successfully demonstrated via an application to the problem of optimal petroleum field development, including an international and commercially important benchmark challenge.

Declaration

The work in this thesis is based on research carried out in the Department of Mathematical Sciences, Durham University, United Kingdom. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Copyright © 2022 Jonathan Daniel Owen.

"The copyright of this thesis rests with the author. No quotation from it should be published without the author's prior written consent and information derived from it should be acknowledged."

Acknowledgements

There are many people who I must thank for their part in my PhD journey. First and foremost, I would like to thank my supervisor, Ian Vernon. This research would not have been possible without his time, patience, guidance and encouragement throughout my PhD. I am incredibly grateful for the introduction to the exciting and dynamic research field of the Bayesian analysis of computer models, as well as his willingness to impart his deep knowledge. I would also like to thank my second supervisor, Michael Goldstein, for sharing his vast experience and his valuable contribution of the research ideas developed in this thesis. A special thanks must also go to the late Allan Seheult (1942–2019) for his advice, encouragement, and sincere interest in my research and personal development, whilst beyond statistics, for our many discussions about the world of cycling. You will be missed.

Thank you to Jonathan Carter, University of Coventry, for your collaboration on the TNO OLYMPUS Well Placement Optimisation Challenge and facilitating the simulations, as well as for sharing your extensive knowledge of oil reservoir engineering and the petroleum industry. Similarly, I would like to thank Bernard Piette and Peter Craig, Durham University, for their time and technical help in setting up and running the OPM flow simulator, without which, my analysis of the TNO OLYMPUS Well Control Optimisation Challenge would not have been possible.

I would like to extend my sincere thanks to Helena Nandi Formentin for your encouragement, eternal optimism, and our countless invaluable discussions that have helped me to better understand oil reservoir engineering and spawned further

research ideas. I am grateful to Sam Jackson for the in-depth discussions about our research, and for advice on navigating the PhD to the academic career path. Thank you to Eleanor Loughlin; it has been a pleasure to work with you as a manager and tutor for Maths and Stats Lab. I would like to thank Margaret Ray for the encouragement, advice and insight into the world of academia. In addition, I would to thank the many members of staff and postgraduate students in the Department of Mathematical Sciences who have supported my PhD and made my time in Durham so enjoyable.

I must also thank EPSRC and Emerson for funding my PhD, and the Smith Institute for facilitating the arrangement. I would like to thank the Emerson team at Oxford, and in particular: Richard Hammersley, Rafel Bordas, Paul Ward, Mohammad Ali Javed, Robert Frost and Taha Taha, for their support, advice and direction throughout my PhD. Moreover, thank you for the collaboration on the commercial client placement analysis.

I conclude by thanking my Mum and Dad. I will always be grateful for their time, patience and continual support throughout my long journey through eduction; their constant love and encouragement have been essential. In addition, I would like to thank my Mum for proofreading this entire thesis, as well as numerous other presentations, reports and a paper throughout my PhD, despite a limited understanding of the actual contents.

Contents

Abstract **iii**

Declaration **v**

Acknowledgements **vii**

List of Figures **xvii**

List of Tables **xxv**

1 Introduction **1**

2 Bayesian Uncertainty Analysis for Computer Models **7**

2.1 Computer Models of Physical Systems 8

2.1.1 Purpose of Computer Models 8

2.1.2 Challenges in the Analysis of Computer Models 9

2.1.3 Computer Model Input Parameters 10

2.2 Bayes Linear and Full Bayesian Analysis 11

2.2.1 Full Bayesian Analysis 12

2.2.2 Bayes Linear Analysis 12

2.2.3 Comparison of the Bayes Linear and Full Bayesian Approaches 14

2.3 Design of Computer Experiments 17

2.3.1 Types of Designs 18

2.3.2 Factorial Designs 18

2.3.3 Uniform Designs 19
2.3.4 Latin Hypercube Designs 20
2.3.5 Design Selection Criteria 20
2.4 Bayesian Emulation 23
2.4.1 Emulator Structure 23
2.4.2 Covariance Functions 27
2.4.3 Emulator Prior Belief Specification 32
2.4.4 Gaussian Process Emulation 34
2.4.5 Bayes Linear Emulation 38
2.4.6 Emulator Diagnostics 48
2.5 Multivariate Emulation 53
2.6 Multilevel Emulation 56
2.7 Uncertainty Quantification 57
2.7.1 Linking the Computer Model to the Real Physical System . 58
2.7.2 Types of Uncertainties 59
2.7.3 Observational Error 61
2.7.4 Structural Model Discrepancy 62
2.8 History Matching 66
2.8.1 Best Input Assumption 68
2.8.2 Implausibility Measures 69
2.8.3 Pukelsheim's 3-sigma Rule 72
2.8.4 History Matching Algorithm 73
2.8.5 Resampling from the Current Non-Implausible Region . . . 76
2.8.6 Stopping Rule 78
2.8.7 Comparison with Calibration 79

3 Iterative Decision Support for Well Control Optimisation 83

3.1 Introduction to the TNO OLYMPUS Field Development Optimisation Challenge 84

3.1.1 Overview of the TNO OLYMPUS Field Development Optimisation Challenge . . . 85
3.1.2 Critique of the TNO OLYMPUS Field Development Optimisation Challenge . . . 88
3.1.3 Petroleum Industry Approaches to Field Development Optimisation . . . 93
3.2 Efficient Geological Ensemble Subsampling . . . 99
3.2.1 Efficient Geological Ensemble Subsampling Graphical Investigations . . . 100
3.2.2 Efficient Geological Ensemble Subsampling Linear Modelling 101
3.3 Hierarchical Emulators Exploiting Known Simulator Behaviour . . 103
3.3.1 Motivation . . . 104
3.3.2 Emulators Exploiting Known Simulator Behaviour for NPV Constituents . . . 106
3.3.3 Emulation of the Approximate NPV . . . 117
3.3.4 Linking the Exact and Approximate NPV . . . 121
3.3.5 Emulation of the Ensemble Mean NPV . . . 122
3.3.6 Summary of the Hierarchical Emulator Construction . . . 125
3.4 Optimisation and Decision Theoretic Approaches . . . 125
3.4.1 Optimisation . . . 126
3.4.2 Bayesian Optimisation . . . 127
3.4.3 Decision Theoretic Approaches . . . 130
3.4.4 Decision Support . . . 134
3.5 Uncertainty Quantification within Decision Support . . . 136
3.5.1 Effect of Uncertainties on the Utility Surface . . . 137
3.5.2 Utility Function Cost Parameter Uncertainty . . . 138
3.5.3 Ensemble Uncertainty . . . 140
3.5.4 Decision Implementation Error . . . 141
3.5.5 Utility Function Uncertainty . . . 143

3.6 Iterative Decision Support Procedure 143
3.6.1 Link to Real World Decision Making 144
3.6.2 Decision Support Implausibility Measure 145
3.6.3 Iterative Decision Support Algorithm 146
3.6.4 Comparison with Optimisation and Decision Theoretic Approaches 147

4 TNO OLYMPUS Well Control Optimisation 151

4.1 Introduction to the TNO OLYMPUS Well Control Optimisation Challenge 152
4.1.1 TNO OLYMPUS Well Control Optimisation Challenge . . 152
4.1.2 OLYMPUS Exploratory Analysis 156
4.2 Application of the EGES Techniques 162
4.3 Targeted Bayesian Design 164
4.3.1 Prior Constraints and Targeted Bayesian Design Methodology 166
4.3.2 Wave 1 Design for Simulations 168
4.4 Bayes Linear Emulation of the Expected NPV 173
4.5 Hierarchical Emulation Exploiting Known Simulator Behaviour . . 177
4.5.1 Structured Emulators Exploiting Known Simulator Behaviour for NPV Constituents 177
4.5.2 Emulation of the Approximate NPV 182
4.5.3 Linking the Exact and Approximate NPV 187
4.5.4 Emulation of the Ensemble Mean NPV 188
4.5.5 Comparison of Bayes Linear and Hierarchical Emulation for the Ensemble Mean NPV 189
4.6 Uncertainty Analysis for Well Control Optimisation 191
4.6.1 Linking the OLYMPUS Model to the Real World System . 192
4.6.2 Geological Uncertainty 192
4.6.3 Efficient Geological Ensemble Subsampling Uncertainty . . 193
4.6.4 NPV Cost Parameter Uncertainty 194

4.6.5 Structural Model Discrepancy 200
4.6.6 Decision Implementation Error 201
4.6.7 Uncertainty Analysis Summary 201
4.6.8 Comparison of Uncertainty Quantification with the Utility Function Emulator Uncertainty 201
4.7 Iterative Decision Support for TNO OLYMPUS Well Control Optimisation 204
4.7.1 Decision Support Using a Bayes Linear Emulator 204
4.7.2 Decision Support Using a Hierarchical Emulator Exploiting Known Simulator Behaviour 207
4.7.3 Decision Support Results for Alternative Uncertainty Quantifications 212
4.8 Decision Support Conclusion 215

5 Commercial Client Placement Analysis: Emerson Jade Model Well Control Optimisation Application 219

5.1 Introduction to the Emerson Jade Model 221
5.1.1 Overview of the Emerson Jade Model 221
5.1.2 Emerson Jade Model Exploratory Analysis 224
5.2 Application of the EGES Techniques 226
5.3 Targeted Bayesian Design Wave 1 230
5.4 Hierarchical Emulation Exploiting Known Simulator Behaviour Wave 1 235
5.4.1 Structured Emulators Exploiting Known Simulator Behaviour for NPV Constituents 235
5.4.2 Emulation of the Approximate NPV 242
5.4.3 Linking the Exact and Approximate NPV 244
5.4.4 Emulation of the Ensemble Mean NPV 244
5.5 Wave 1 Uncertainty Analysis for Well Control Optimisation . . . 246
5.5.1 Linking the Jade Model to the Real World System 247

5.5.2 Geological Uncertainty 247
5.5.3 Efficient Geological Ensemble Subsampling Uncertainty . . 248
5.5.4 NPV Cost Parameter Uncertainty 249
5.5.5 Structural Model Discrepancy 255
5.5.6 Decision Implementation Error 257
5.5.7 Uncertainty Analysis Summary 259
5.6 Decision Support Wave 1 for Well Control Optimisation 261
5.6.1 Results 261
5.6.2 Decision Support Wave 1 Results for Alternative Uncertainty Quantifications 264
5.7 Wave 2 Design of Simulations 268
5.7.1 Wave 2 Design Methodology 268
5.7.2 Wave 2 Design 270
5.8 Hierarchical Emulation Exploiting Known Simulator Behaviour Wave 2 274
5.8.1 Structured Emulators Exploiting Known Simulator Behaviour for NPV Constituents 274
5.8.2 Emulation of the Approximate NPV 278
5.8.3 Linking the Exact and Approximate NPV 279
5.8.4 Emulation of the Ensemble Mean NPV 280
5.9 Wave 2 Uncertainty Analysis for Well Control Optimisation . . . 284
5.9.1 Geological Uncertainty 284
5.9.2 Efficient Geological Ensemble Subsampling Uncertainty . . 284
5.9.3 NPV Cost Parameter Uncertainty 285
5.9.4 Structural Model Discrepancy 288
5.9.5 Decision Implementation Error 289
5.9.6 Uncertainty Analysis Summary 290
5.10 Decision Support Wave 2 for Well Control Optimisation 292
5.10.1 Results 292

5.10.2 Decision Support Wave 2 Results for Alternative Uncertainty Quantifications 297
5.11 Validation of the Decision Support Non-Implausible Region . . . 300
5.11.1 Design of Validation Simulations 300
5.11.2 Validation of EGES Linear Model for the Ensemble Mean NPV 303
5.11.3 Validation of the Emulator for the Ensemble Mean NPV . . 306
5.11.4 Validation of the Emulator for the Expected NPV 309
5.12 Comparison of the Waves of Simulations 312
5.12.1 Progression of the Ensemble Mean NPV 313
5.12.2 Evolution of Simulator Outputs 314
5.13 Decision Support Conclusion 316

6 Bayesian Emulation of Computer Models with Known Structured Partial Discontinuities 323

6.1 Motivation of the TNO OLYMPUS Well Placement Optimisation Challenge 324
6.2 Current Approaches to Bayesian Emulation of Computer Models with Discontinuities 327
6.3 Methodology for Bayesian Emulation of Computer Models with Structured Partial Discontinuities 333
6.3.1 Torn Embedding Surfaces in Higher Dimensions 334
6.3.2 Controlling for the Warping Effect Induced by the Embedding – Local Effect 338
6.3.3 Controlling for the Global Impact of the Embedding Using Non-Stationary Emulation 352
6.4 2-Dimensional Example with Non-Linear Partial Discontinuities . . 355

7 TNO OLYMPUS Well Placement Optimisation 359

7.1 Introduction to the TNO OLYMPUS Well Placement Optimisation Challenge . . . 360
7.2 Choice of the Embedding Surface . . . 363
7.3 Prior Covariance Structure . . . 367
7.4 Well 1 Wave 1 Analysis . . . 369
7.4.1 Well 1 Wave 1 Design for Simulations . . . 369
7.4.2 Well 1 Wave 1 Emulation Results . . . 374
7.5 Well 1 Wave 2 Analysis . . . 378
7.5.1 Well 1 Wave 2 Design for Simulations . . . 379
7.5.2 Well 1 Wave 2 Emulation Results . . . 380
7.5.3 Well 1 Wave 2 Quantile Emulation Results . . . 383
7.6 Well 2 Analysis . . . 383
7.6.1 Well 2 Design for Simulations . . . 386
7.6.2 Well 2 Emulation Results . . . 386
7.7 Well 3 Analysis . . . 388
7.7.1 Well 3 Design for Simulations . . . 388
7.7.2 Well 3 Emulation Results . . . 388
7.8 Well Placement Optimisation Conclusion . . . 390

8 Conclusion **393**

Nomenclature **403**

A Linear Regression Stepwise Selection **431**

B Matrix Identities **435**

Bibliography **439**

List of Figures

3.1 Three-dimensional map of the TNO OLYMPUS oil reservoir model depicting the initial oil saturation for the OLYMPUS 28 realisation 89

3.2 OLYMPUS 25 WOPTPROD2_20180101 versus prod_2_2016_01 . 108

3.3 OLYMPUS 25 WOPTPROD2_20180101 versus prod_2_2016_01 focused around $c^{l}_{P2,2016}$ 109

3.4 OLYMPUS 25 WOPTPROD2_20180101 wave 1 preliminary Bayes linear emulator and structured emulator predictive credible intervals versus prod_2_2016_01 114

4.1 OLYMPUS cumulative field outputs versus time for exploratory simulations 158

4.2 OLYMPUS WOPTPROD10 versus time for the exploratory simulations 159

4.3 OLYMPUS 25 approximate absolute annual contributions to the NPV for the exploratory simulations 160

4.4 Comparison of the target and achieved rate controls for the OLYMPUS exploratory simulations 161

4.5 OLYMPUS model preliminary graphical investigations for EGES . 163

4.6 OLYMPUS EGES linear model Adjusted R^2 values 165

4.7 OLYMPUS wave 1 design sum of normalised parameter histograms 170

4.8 OLYMPUS wave 1 design pairs plot for producer well 2 coloured by NPV 172

4.9 OLYMPUS wave 1 design marginal density histograms 173
4.10 OLYMPUS 33 CWGOPT and CWGWIT versus date for the first 52 wave 1 simulations 174
4.11 Bayes linear emulator for the expected NPV leave-one-out diagnostic plot 176
4.12 OLYMPUS 25 WOPTPROD2_20180101 versus prod_2_2016_01 showing the change point upper bound 178
4.13 OLYMPUS wave 1 NPV constituents change point upper bound and extrapolation cut-off intervals 180
4.14 Leave-one-out diagnostic plots for the structured emulation of OLYMPUS 25 WOPT for producer well 2 183
4.15 Leave-one-out diagnostic plots for the structured emulation of OLYMPUS 25 WOPTPROD10_2020_01 and WWITINJ2_2022_01 . . 184
4.16 Leave-one-out diagnostic plots for the structured emulation of OLYMPUS 25 approximate and exact NPV 186
4.17 OLYMPUS wave 1 hierarchical emulation diagnostic plots for the ensemble mean NPV 189
4.18 Histograms of the wave 1 Bayes linear and hierarchical emulators adjusted variances for the OLYMPUS ensemble mean NPV . . . 190
4.19 Trace plot of simulated oil prices for TNO OLYMPUS Well Control Optimisation Challenge 196
4.20 OLYMPUS wave 1 NPV cost parameter uncertainty standard deviation projection plots 197
4.21 OLYMPUS wave 1 ensemble mean NPV standard deviation induced by NPV cost parameter variability versus the ensemble mean NPV . 198
4.22 Histogram of the OLYMPUS wave 1 NPV cost parameter uncertainty compared with geological and EGES uncertainty 199

4.23 Histograms of the OLYMPUS wave 1 ensemble mean NPV hierarchical emulator adjusted variances compared with the uncertainty quantification variances 203
4.24 OLYMPUS wave 1 decision support implausibility measure and optical density pairs plot with marginal density histograms for the active decision parameters using a Bayes linear emulator 206
4.25 OLYMPUS wave 1 decision support implausibility measure and optical density pairs plot with marginal density histograms for the producer well 2 decision parameters using a hierarchical emulator 210
4.26 OLYMPUS wave 1 decision support implausibility measure and optical density pairs plot with marginal density histograms for all 2016 and 2018 decision parameters using a hierarchical emulator 211

5.1 Emerson Jade model map with key of decision parameter well IDs and types 223
5.2 Jade exploratory analysis NPV boxplots 225
5.3 Jade model preliminary graphical investigations for EGES 227
5.4 Jade ensemble mean NPV linear models adjusted R^2 histograms . . 228
5.5 Jade ensemble mean EGES linear model leave-one-out diagnostics plots for $N_{\text{EGES}} = 3$ and $N_{\text{EGES}} = 4$ 229
5.6 Visualisation tools for selecting the truncation bounds for the normalised sum of parameters for the Jade wave 1 design 231
5.7 Jade wave 1 design sum of normalised parameter histograms. . . . 233
5.8 Jade wave 1 design pairs plot for producer well B4 coloured by NPV 234
5.9 Jade wave 1 design marginal density histograms 235
5.10 Jade 30 wave 1 WOPT for producer well B4 versus target production rates showing the change point upper bounds 236
5.11 Jade wave 1 NPV constituents change point upper bound and extrapolation cut-off intervals 238

5.12 Leave-one-out diagnostic plots for the wave 1 structured emulation of Jade 30 WOPT for producer wells C5 and C6 241
5.13 Leave-one-out diagnostic plots for the wave 1 structured emulation of Jade 30 approximate and exact NPV 243
5.14 Jade ensemble mean NPV wave 1 hierarchical emulator output plots 245
5.15 Trace plot of simulated oil prices for Emerson Jade Model application using multiple fixed means 251
5.16 Jade wave 1 NPV cost parameter uncertainty standard deviation projection plots prior to the removal of linear effects 252
5.17 Jade wave 1 NPV cost parameter uncertainty standard deviation projection plots after the removal of linear effects 254
5.18 Jade wave 1 ensemble mean NPV standard deviation induced by NPV cost parameter variability versus the ensemble mean NPV 255
5.19 Histogram of the Jade wave 1 NPV cost parameter uncertainty compared with geological and EGES uncertainty 256
5.20 Histogram of the Jade wave 1 NPV variance due to decision implementation error 259
5.21 Jade wave 1 decision support implausibility measure and optical density pairs plot with marginal density histograms for the producer well B4 decision parameters 263
5.22 Jade wave 1 decision support using the simulated ensemble mean NPV versus decision parameters 264
5.23 Comparison of the Jade wave 1 and 2 designs pairs plot for producer well B4 272
5.24 Comparison of the Jade wave 1 and 2 designs pairs plot for injector well A16 273
5.25 Jade wave 2 NPV constituents change point upper bound and extrapolation cut-off intervals 276

5.26 Leave-one-out diagnostic plots for the wave 2 structured emulation of Jade 30 WOPT for producer wells C5 and C6 278
5.27 Leave-one-out diagnostic plots for the wave 2 structured emulation of Jade 30 approximate and exact NPV 280
5.28 Jade ensemble mean NPV wave 2 hierarchical emulator output plots 282
5.29 Jade ensemble mean NPV wave 2 hierarchical emulator pairs plot of the decision parameters for producer well C5 283
5.30 Jade wave 2 NPV cost parameter uncertainty standard deviation projection plots after the removal of linear effects 287
5.31 Jade wave 2 ensemble mean NPV standard deviation induced by NPV cost parameter variability versus the ensemble mean NPV 288
5.32 Histogram of the Jade wave 2 NPV cost parameter uncertainty compared with geological and EGES uncertainty 289
5.33 Histogram of the Jade wave 2 NPV variance due to decision implementation error 290
5.34 Jade wave 2 decision support implausibility measure and optical density pairs plot with marginal density histograms for the producer well B4 decision parameters 294
5.35 Jade wave 2 decision support implausibility measure and optical density pairs plot with marginal density histograms for all 2022 and 2023 decision parameters 295
5.36 Jade wave 2 decision support using the simulated ensemble mean NPV versus decision parameters 296
5.37 Comparison of the Jade wave 2 and validation designs pairs plot for producer well B4 302
5.38 Validation diagnostic plots for Jade EGES linear model for the ensemble mean NPV over the decision support non-implausible space . 305
5.39 Decision support for the Jade initial ensemble mean NPV validation simulations 307

5.40 Validation diagnostic plots for the wave 2 hierarchical emulator for the Jade ensemble mean NPV over the decision support non-implausible space . . . 308
5.41 Validation diagnostic plots for the wave 2 hierarchical emulator for the Jade expected NPV over the decision support non-implausible space . . . 310
5.42 Decision support for the Jade full ensemble mean NPV validation simulations . . . 311
5.43 Progression of the EGES linear model predictions for the Jade ensemble mean NPV over the three waves . . . 314
5.44 Evolution of Jade 30 simulation outputs for a subset of wave 1, 2 and 3 simulations . . . 315

6.1 Physical map of the TNO OLYMPUS oil reservoir model . . . 326
6.2 Transformed map of the TNO OLYMPUS oil reservoir model . . . 327
6.3 Emulation of a 2D function with a single partial discontinuity . . . 329
6.4 Emulation of a 2D function with two partial discontinuities . . . 339
6.5 Emulation of a 2D function with four non-linear partial discontinuities . . . 358

7.1 Embedding surface used in the TENSE emulator for the TNO OLYMPUS Well Placement Optimisation Challenge. . . . 366
7.2 Prior correlation structure along vertical strips for the TNO OLYMPUS Well Placement Optimisation Challenge . . . 368
7.3 OLYMPUS well 1 location TENSE emulator expectation for the expected NPV . . . 371
7.4 OLYMPUS well 1 location TENSE emulator standard deviation for the expected NPV . . . 372
7.5 OLYMPUS well 1 location TENSE emulator credible interval upper bound for the expected NPV . . . 377

7.6 OLYMPUS well 1 location TENSE emulator for the expected NPV without training on the ghost runs 378
7.7 Boxplots of the NPV over the 50 OLYMPUS geological realisations for the well 1 location analysis. 384
7.8 Quantile emulation using the TENSE framework applied to the NPV over the 50 OLYMPUS geological realisations for the well 1 location 385
7.9 OLYMPUS well 2 location TENSE emulator expectation for the expected NPV comparing candidate producer and injector wells . . . 387
7.10 OLYMPUS well 3 location TENSE emulator expectation for the expected NPV 389

List of Tables

2.1 Valid stationary correlation functions 29

4.1 TNO OLYMPUS Well Control Optimisation Challenge operational rate constraints 154

4.2 Acronyms used in the analysis of the TNO OLYMPUS Field Development Optimisation Challenge 155

4.3 Expected NPV fixed cost parameters within the TNO OLYMPUS Well Control Optimisation Challenge 195

4.4 OLYMPUS wave 1 uncertainty analysis results 202

4.5 OLYMPUS wave 1 decision support non-implausible proportion of the decision space under various uncertainty analyses 213

5.1 Emerson Jade model application operational rate constraints . . . 223

5.2 Expected NPV fixed cost parameters within the Emerson Jade model application 249

5.3 Emerson Jade model application wave 1 uncertainty analysis results 260

5.4 Jade wave 1 decision support non-implausible proportion of the decision space under various uncertainty analyses 265

5.5 Emerson Jade model application wave 1 and 2 uncertainty analysis results 291

5.6 Jade wave 1 and 2 decision support non-implausible proportion of the decision space under various uncertainty analyses 298

7.1 OLYMPUS geological fault boundary locations 360

7.2 Expected NPV fixed contribution costs within the TNO OLYMPUS Well Placement Optimisation Challenge 361
7.3 TNO OLYMPUS Well Placement Optimisation Challenge well 1, wave 1, 47 point design locations 374
7.4 TNO OLYMPUS Well Placement Optimisation Challenge well 1, wave 2, 47 point design locations 381
7.5 TNO OLYMPUS Well Placement Optimisation Challenge field development strategy for the first three wells 390

Chapter 1

Introduction

Computer models are increasingly prevalent across many scientific disciplines, industry and government, to investigate complex real world phenomena with two primary objectives being: to improve the understanding of the behaviour of these systems; and to guide decision making processes. Over the last 30 years, a suite of Bayesian uncertainty analysis methodology has been developed for the study of computer models and to make inferences about the corresponding real world system. These have been successfully implemented across a broad spectrum of applications with examples including: climate science [40, 66, 151, 187]; cosmology [82, 104, 170, 171]; and epidemiology [2, 3, 5, 177]. The focus of this thesis is to develop a Bayesian framework for providing robust decision support under uncertainty using computer models; a problem which has yet to be adequately addressed in the literature, and to further develop the Bayesian emulation methodology.

An early adopter of computer models was the petroleum industry in the form of ensembles of oil reservoir models where Bayesian methodology has been effectively applied to learn about the underlying geological properties of the reservoir, for example in [26, 27, 28, 29, 30, 128, 132]. More recently, computer models are employed in decision making under uncertainty problems to address the challenges of optimal field development. The aim is to devise well control and placement strategies to maximise an objective function, often the expected Net Present Value (NPV) over

the field lifetime, whilst accounting for the uncertainty induced by the underlying geology. This represents a highly complex problem with the commercial significance evidenced by the advent of the TNO OLYMPUS Field Development Optimisation Challenge [4, 39, 93], as well as being of great interest to my iCASE PhD industry partner, Emerson. The field development optimisation problem therefore serves as recurring motivation for the methodological development in this thesis.

Bayesian emulators are a vital tool in the analysis of computer models [29, 129, 132, 171]. These are a type of surrogate for computer models which are constructed given a carefully chosen set of simulator evaluations to provide a fast and efficient statistical approximation, yielding predictions for as yet unevaluated parameter settings, along with a corresponding statement of the uncertainty. Emulators are typically several orders of magnitude faster to evaluate than the computer model enabling large numbers of simulations to be performed, whilst the quantification of the uncertainty facilitates inferences about the corresponding computer model. Further challenges arise in the accurate emulation of computer models exhibiting structured partial discontinuities.

Another essential component is an uncertainty analysis. This involves formulating a structured link between the computer model and the physical system it represents, before identifying, assessing and quantifying all major sources of uncertainty [59, 67, 107, 171]. The complete process is essential for a robust analysis and obtaining meaningful inferences about the real world physical system. However, such a detailed uncertainty analysis is frequently omitted from many applications, with a coherent uncertainty analysis framework for decision making yet to be established.

Within industry and government settings, analysts or statisticians perform decision analyses, exploiting computer models, and deliver results to a committee of decision makers, who may possess further unknown judgments and risk preferences that are inaccessible to the analysts. The actual utility function is therefore unknown. This can only be addressed via decision support incorporating a comprehensive uncertainty analysis along with Bayesian emulators to facilitate a full exploration of

often high-dimensional decision parameter spaces. Decision support returns a class of robust decisions, along with additional insights or information, for consideration by a committee of decision makers.

The importance of providing robust decision support for real world systems via an analysis of their representative computer models motivates the research presented in this thesis. The major contributions are as follows:

1. Establishment of a hierarchical Bayesian emulation framework that exploits known constrained simulator behaviour and that is adaptable to other (partially) known forms of computer model output and utility functions (Chapters 3 to 5).
2. Advancement of the uncertainty quantification methodology for performing decision support (Chapters 3 to 5).
3. Formulation of an iterative decision support procedure that addresses the separation of analysts from the final decision makers (Chapters 3 to 5 and 7).
4. Development of Bayesian emulators for computer models with structured partial discontinuities (Chapters 6 and 7).

This thesis is arranged into 8 chapters. We start in Chapter 2 with a literature review introducing computer models and presenting the Bayes linear versus a full Bayesian approach to belief specification. It is noted that this thesis follows the subjective Bayesian approach to uncertainty quantification. The review continues with a detailed discussion of Bayesian emulators, including a derivation of the Bayes linear update formulae for an emulator. Uncertainty quantification methodology is then described for linking the computer model to the real system with these techniques combined in the history matching process for identifying a set of model parameters that yield an acceptable match between the simulator output and observed data with respect to an uncertainty analysis.

The aim of Chapter 3 is to develop methodology to perform decision making under uncertainty using computer models. We begin by introducing the TNO OLYMPUS Field Development Optimisation Challenge and critiquing its limitations. This

motivates the development of a hierarchical Bayesian emulation framework exploiting known constrained simulator behaviour. The uncertainty quantification literature is then extended to performing decision support where conclusions are invariant to linear transformations of the utility function with a novel technique devised to assess, quantify and remove such effects. These are incorporated within a novel iterative decision support framework that addresses the problem of the separation of analysts from the final decision makers. This framework is flexible and adaptable to a wide variety of applications.

The methodology is then applied in Chapter 4 to the TNO OLYMPUS Well Control Optimisation Challenge reformulated as a decision support problem. Firstly, methodology is devised for constructing a targetted Bayesian design based on prior insight. It is demonstrated how exploiting known constrained simulator behaviour within a hierarchical emulator construction yields substantially greater accuracy compared with a Bayes linear emulator, as well as emphasising the need to remove uncertainty linear transformation effects. The first wave of iterative decision support is performed with an efficient assessment of alternative uncertainty specifications highlighting the importance of a comprehensive analysis.

Chapter 5 forms the commercial client placement part of my iCASE PhD studentship with Emerson where the decision support methodology is applied to their Jade model. Multiple waves of iterative decision support are performed to demonstrate the sequential refocusing of the decision space and how the uncertainty quantification may be revised at each iteration. The results of the hierarchical emulation and decision support frameworks are subsequently validated. Moreover, it is concluded that geological uncertainty constitutes the single largest source of uncertainty and thus requires a more careful treatment across the petroleum industry to yield robust and meaningful decision support results.

In Chapter 6 we develop novel Bayesian emulation methodology for computer models with structured partial discontinuities. This is motivated by the TNO OLYMPUS Well Placement Optimisation Challenge where the expected NPV as a function

of well location possesses partial discontinuities with respect to geological fault boundaries. Two example scalar functions are used to explicate the methodological development where emulators are constructed for an embedding of the input parameter space in higher dimensions with the embedding surface chosen to characterise the discontinuities. The general form of the embedded parameters variance matrix is derived and depends on the embedding surface and input settings, thus necessitating the use of non-stationary emulators. These steps are to correct for an induced warping effect. The complete framework is referred to as the Torn Embedding Non-Stationary Emulation (TENSE) approach.

The TENSE methodology is utilised in Chapter 7 to perform a sequential well location analysis for the TNO OLYMPUS Well Placement Optimisation Challenge. First, an appropriate form of embedding surface is devised and justified before illustrating the induced prior covariance structure. Application of the TENSE framework yields an emulator for the ensemble mean NPV which successfully encompasses the partial discontinuities induced by the geological faults. This is employed within the iterative decision support strategy of Chapter 3 and combined with knowledge from an experienced oil reservoir engineer to obtain an appropriate field development strategy.

Chapter 8 summarises this research and presents the conclusions before suggesting a number of directions for further research. An overview of the notation and acronyms used in this thesis is provided in the nomenclature.

Chapter 2

Bayesian Uncertainty Analysis for Computer Models

This chapter presents a detailed review of the theory and methodology for the Bayesian analysis of complex computer models used in the study of real world physical systems including the modelling and uncertainty quantification. In Section 2.1 we introduce computer models for physical systems describing their construction and purpose in Section 2.1.1, associated challenges in their use in Section 2.1.2, and the types of inputs in Section 2.1.3. A comparison of the Bayes linear and full Bayesian framework for analyses is detailed in Section 2.2. Computer experiments require a design of simulations over the input parameter space with a selection of methods for construction and design optimisation discussed in Section 2.3. An integral component in the analysis of computer models is Bayesian emulators; these are described in Section 2.4. This includes: their structure; choice and properties of covariance functions; and prior belief specification in Sections 2.4.1 to 2.4.3 respectively, Gaussian process and Bayes linear emulation in Sections 2.4.4 and 2.4.5 respectively, and diagnostics in Section 2.4.6. The extension of emulators to multivariate computer model outputs is outlined in Section 2.5, as well as an introduction to multilevel emulation for handling multiple computer model versions of differing levels of complexity for the same physical system in Section 2.6. Methods of linking the computer model

to the actual physical system and observations via an uncertainty quantification are described in Section 2.7. Bayesian emulators and uncertainty quantification techniques are combined in the history matching process presented in Section 2.8, including a comparison with calibration procedures.

2.1 Computer Models of Physical Systems

In this section we introduce computer models of physical systems discussing their formulation and purposes in Section 2.1.1, challenges associated with their use in Section 2.1.2, and the different types of input variables in Section 2.1.3.

2.1.1 Purpose of Computer Models

The advent of increasingly powerful computers with larger memory capacity and faster processing speeds has transformed the way in which we study complex physical systems with mathematical models in the form of computer codes used as surrogates [152, 153]. These are commonly referred to as computer models or simulators, which are used interchangeably throughout this thesis. They consist of a collection of coupled mathematical equations which describe the interacting behaviour of the various components of the system which results in complex and non-linear dynamics for the system output(s). Examples include coupled systems of (stochastic) ordinary and partial differential equations. These are often solved numerically. Moreover, simulators may be either deterministic or stochastic. Throughout this thesis we address the analysis of deterministic computer models representing the output as the vector-valued function $f(\cdot)$, whilst univariate outputs from a single simulator may be represented as a scalar function $f_i(\cdot)$. Different types of inputs are described in Section 2.1.3.

It is often the case that traditional physical experiments are impossible to conduct. This is due to them being prohibitively expensive, especially for large scale systems; difficulties in performing measurements, potentially due to safety concerns; and

their repeatability for one-time events. Computer models negate many of these issues by providing a cheaper means of experimentation for which measurements can be easily obtained, whilst also being repeatable. The increasing importance of simulators to study real world physical systems spawned the subject of the "Design and Analysis of Computer Experiments" (DACE) [152, 153], for which there are three main aims. These are: to advance knowledge of the system being studied; to perform optimisation over a set of design variables; or to provide decision support to achieve some objective in the system [185]. The principal focus of this thesis is decision support.

The analysis of computer models has been successfully employed in a number of scientific areas including: climate science [40, 66, 151, 167, 168, 169, 185, 187]; petroleum reservoir engineering [26, 27, 28, 29, 30, 135]; cosmology [82, 104, 170, 171]; epidemiology [2, 3, 5, 177]; cardiac electrophysiology [20, 25, 100, 122]; systems biology [173, 178]; volcanology [11, 78]; hydrology [65]; traffic flow modelling [10, 124]; and preparation for accidents involving the release of nuclear radiation [107].

2.1.2 Challenges in the Analysis of Computer Models

Within the subject of DACE, the systems studied have become increasingly complex over time, both due to a desire to learn more about new systems and as a consequence of an increased state of knowledge about those previously studied. To keep pace with this demand, the simulator code has become increasingly long and complex. Computer models therefore often behave like a "black-box"; for a particular input setting, the simulator output is unknown until it has been evaluated, whilst also possessing a high-dimensional input parameter space, and a large number of outputs. The long evaluation times, typically ranging from several minutes to months, along with constraints on computational resources renders the direct use of simulators for an analysis infeasible, thus necessitating the development of techniques such as Bayesian emulation (described in Section 2.4). Moreover, the presence of many sources of uncertainty demands a thorough uncertainty analysis (see Section 2.7).

Another challenge is the analysis of systems for which there exists a collection of simulators of differing levels of complexity, for example, in petroleum reservoir modelling [29, 30, 32, 108, 132], and in climate science [168, 169]. This is commonly addressed using multilevel or multiscale emulation which is briefly discussed in Section 2.6. Although not the subject of this thesis, additional challenges arise when the computer model is stochastic. See [7] for a review of current methodology.

2.1.3 Computer Model Input Parameters

Computer models map a vector of inputs to a vector-valued output, $f(\cdot)$. There are four distinct input types: model parameters; decision or control parameters; environmental variables [153]; and tuning parameters [67].

Model parameters are the inputs which determine the behaviour of the system due to the underlying physics and boundary conditions, and possess a physical interpretation for the real system. A common assumption is that there exists a setting for the model parameters which results in the model accurately portraying the physical system, although this value may differ from the true underlying physical constant. These are commonly denoted by the vector $\mathbf{x}$ with model parameter space $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^{\tilde{D}},\ \tilde{D} \in \mathbb{N}$, with an individual component denoted by x_i, where the subscript specifies the context.

Decision or control parameters represent inputs to the system which the analyst can modify in order to affect the system, potentially with the aim of achieving a certain objective. These are of primary interest within decision support and optimisation problems where the analysis of a computer model is used to guide the choice of decision parameter settings. Decision parameters are denoted by the vector $\mathbf{d}$, with individual input typified by the form d_i, where the subscript is context specific. The decision parameter space is indicated by $\Omega \subset \mathbb{R}^{D},\ D \in \mathbb{N}$.

Environmental variables are inputs which are specific to an individual user within a system for which an example in [153] is a model for a hip prosthesis in which the stress on the implant depends on the individual patient and their activity level.

Tuning parameters are introduced in [67] and are inputs included for several reasons: to ensure a good fit between the model and observed data (with error); account for poorly understood physics; and solver deficiencies. These are not always included with alternative approaches including: model calibration or history matching (Section 2.8) to train a computer model using observed data; whilst the other two considerations can be incorporated within model discrepancy in an uncertainty quantification (Section 2.7). Neither of these input types are considered further within this thesis.

Throughout this thesis computer models will be viewed as a function of model and decision parameters, $f(\mathbf{x}, \mathbf{d})$. For clarity, this will often be abbreviated to $f(\mathbf{x})$ or $f(\mathbf{d})$ in contexts where the other parameter type is implicit or unimportant. Specifically, $\mathbf{x}$ and $f(\mathbf{x})$ will be used when describing general methodology for the analysis of computer models in Chapters 2, 6 and 7, whilst $\mathbf{d}$ and $f(\mathbf{d})$ will be used when discussing decision support in Chapters 3 to 5.

2.2 Bayes Linear and Full Bayesian Analysis

Throughout this thesis we follow a Bayesian approach to the analysis of computer models for which there are two schools of thought: objective [13] and subjective Bayesianism [63], which differ in their means of specifying prior beliefs about uncertain quantities in a system of interest. We subscribe to the latter which provides a coherent framework to structure and combine priors which are regarded as statements of one's personal and subjective beliefs about these uncertainties. Uncertainties about these quantities may be expressed via either probability in a full Bayesian analysis, or expectations in a Bayes linear analysis which are described in Sections 2.2.1 and 2.2.2 respectively, and compared in Section 2.2.3.

2.2.1 Full Bayesian Analysis

A full Bayesian analysis uses probability as a primitive to convey beliefs about uncertain quantities. Consider a collection of random quantities or parameters, $\boldsymbol{\theta} \in \Theta$, for which the aim is to perform statistical inference, given a vector of observed data, $\mathbf{D}$.[1] In the full Bayesian paradigm, the current knowledge of $\boldsymbol{\theta}$ is used to formulate a prior probability distribution, $\pi(\boldsymbol{\theta})$, whilst the observed data is treated as a random quantity sampled from a probability distribution, $\pi(\mathbf{D} \mid \boldsymbol{\theta})$, known as the likelihood, which contains conditional probabilistic beliefs of observing $\mathbf{D}$ given $\boldsymbol{\theta}$. Bayes' theorem is used to update the prior beliefs about $\boldsymbol{\theta}$ to obtain a posterior probability distribution with the general expression in Equation (2.2.1).

$$\pi(\boldsymbol{\theta} \mid \mathbf{D}) = \frac{\pi(\mathbf{D} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\mathbf{D})} \tag{2.2.1}$$

$$= \frac{\pi(\mathbf{D} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\int_{\Theta} \pi(\mathbf{D} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})\, d\boldsymbol{\theta}} \tag{2.2.2}$$

Assuming that all elements of $\boldsymbol{\theta}$ are continuous random variables, an explicit version of Bayes' theorem is shown in Equation (2.2.2). In general, the partition theorem is used to compute $\pi(\mathbf{D})$. For discrete random variables, $\pi(\mathbf{D}) = \sum_{\boldsymbol{\theta} \in \Theta} \pi(\mathbf{D} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})$, whilst for a combination of parameter types, the appropriate mix of integrals and summations is employed.

Bayes' theorem leads to a coherent calculation of the posterior distribution which can be used to perform statistical inferences for $\boldsymbol{\theta}$, as well as to guide the choice of decisions, given both the prior knowledge and information gained through the observed data. There exist many bibliographic resources which provide comprehensive coverage to the full Bayesian approach, for example [14, 112].

2.2.2 Bayes Linear Analysis

The Bayes linear approach [57, 70, 171] to statistical inference uses expectation as a primitive following de Finetti [48, 49]. Consider the collections of random

[1] In the context of computer experiments, we must distinguish between observed data of the physical system and simulator evaluation data.

quantities $\mathbf{B} = (B_1, \ldots, \ldots, B_q)$, $\mathbf{C} = (C_1, \ldots, \ldots, C_r)$ and $\mathbf{D} = (D_1, \ldots, \ldots, D_s)$, with observed data $\mathbf{D}$. This is used to update prior beliefs about quantities $\mathbf{B}$ and $\mathbf{C}$. Within a Bayes linear analysis, a second order belief specification is based on the expectation, variance and covariance. The Bayes linear update formulae for the expectation and variance of $\mathbf{B}$, given observations $\mathbf{D}$, requires the specification of: prior vectors of expectations, $\mathbb{E}[\mathbf{B}]$, and $\mathbb{E}[\mathbf{D}]$; prior variance matrices, $\mathrm{Var}[\mathbf{B}]$ and $\mathrm{Var}[\mathbf{D}]$; and the prior covariance matrix between $\mathbf{B}$ and $\mathbf{D}$, $\mathrm{Cov}[\mathbf{B}, \mathbf{D}]$. These are termed the Bayes linear or adjusted expectation and variance of $\mathbf{B}$ given $\mathbf{D}$, and are presented in Equations (2.2.3) and (2.2.4) respectively. Similarly, the Bayes linear update for the covariance, termed the Bayes linear or adjusted covariance, between $\mathbf{B}$ and $\mathbf{C}$ given observations $\mathbf{D}$, is presented in Equation (2.2.5). This requires the additional specification of the prior covariance matrices between $\mathbf{B}$ and $\mathbf{C}$, $\mathrm{Cov}[\mathbf{B}, \mathbf{C}]$, and between $\mathbf{C}$ and $\mathbf{D}$, $\mathrm{Cov}[\mathbf{C}, \mathbf{D}]$. Note that the derivation follows from considering the corresponding off-diagonal sub-matrix of $\mathrm{Var}_D\left[\binom{\mathbf{B}}{\mathbf{C}}\right]$.

$$\mathbb{E}_D[\mathbf{B}] = \mathbb{E}[\mathbf{B}] + \mathrm{Cov}[\mathbf{B}, \mathbf{D}]\,\mathrm{Var}[\mathbf{D}]^{-1}(\mathbf{D} - \mathbb{E}[\mathbf{D}]) \tag{2.2.3}$$

$$\mathrm{Var}_D[\mathbf{B}] = \mathrm{Var}[\mathbf{B}] - \mathrm{Cov}[\mathbf{B}, \mathbf{D}]\,\mathrm{Var}[\mathbf{D}]^{-1}\,\mathrm{Cov}[\mathbf{D}, \mathbf{B}] \tag{2.2.4}$$

$$\mathrm{Cov}_D[\mathbf{B}, \mathbf{C}] = \mathrm{Cov}[\mathbf{B}, \mathbf{C}] - \mathrm{Cov}[\mathbf{B}, \mathbf{D}]\,\mathrm{Var}[\mathbf{D}]^{-1}\,\mathrm{Cov}[\mathbf{D}, \mathbf{C}] \tag{2.2.5}$$

In situations where $\mathrm{Var}[\mathbf{D}]$ is not invertible, the Moore-Penrose generalised inverse is used instead [137]. Moreover, probabilities can be expressed as the expectation of an indicator function, hence inference for probabilities of events of interest can also be made following the Bayes linear approach.

Several observations are made for the Bayes linear update formulae. The adjusted expectation may be viewed as an estimator for $\mathbf{B}$ where for each B_i it is the best linear fit over the observed data $\mathbf{D}$. Formally, this is the linear combination $\mathbf{a}_i^{\mathrm{T}}\mathbf{D}$ which minimises $\mathbb{E}[(B_i - \mathbf{a}_i^{\mathrm{T}}\mathbf{D})^2]$ over the possible choices of $\mathbf{a}_i \in \mathbb{R}^s$ for $i = 1, \ldots, q$, as evaluated in Equation (2.2.3), and hence may be interpreted as the linear combination of $\mathbf{D}$ which is most informative for $\mathbf{B}$ with respect to a squared error loss function. $\mathrm{Var}_D[\mathbf{B}]$ gives the expected mean square error of $\mathbb{E}_D[\mathbf{B}]$. In the case where $\mathbf{D}$

consists of a collection of indicator functions for events which form a partition,[2] the adjusted expectation is numerically equivalent to the conditional expectation. There is a strong formal relationship between Bayes linear and a full Bayesian analysis in which there is an assumed Gaussian structure, which is characterised by its mean and variance. Suppose that $(\mathbf{B}, \mathbf{D})$ follows a multivariate Gaussian distribution, the Bayesian update yields the normal conditional formulae for the mean and variance which precisely coincide with the adjusted expectation and variance of Equations (2.2.3) and (2.2.4) respectively. For an overview of Bayes linear methods, see [57], whilst for a comprehensive account, see [70].

2.2.3 Comparison of the Bayes Linear and Full Bayesian Approaches

Both the full Bayesian and Bayes linear approach provide a coherent framework in which to combine prior judgements with observed data in order to make posterior inferences. This section discusses their interrelationship as well as the merits and weaknesses of each approach.

A full Bayesian analysis of a problem requires the subjective specification of a joint prior probability distribution over all uncertainties as well as a likelihood, both of which may be of a complex form; an issue which is further exacerbated in high-dimensions. Constraints on time, resources and available knowledge to assess the uncertainties within the problem, potentially requiring the elicitation of expert judgements, makes accurate prior specification extremely challenging. Stipulating a specific prior distribution may accurately reflect certain aspects of the current knowledge, however the full distributional form also implicitly results in additional prior belief statements which may be unfounded, whilst not necessarily being in agreement with the current subjective judgements. Unless prior beliefs can be specified with confidence, the resulting analysis may be non-robust since the posterior

[2]Each D_i takes value zero or one, and exactly one element of $\mathbf{D}$ will be equal to one.

distribution inherits any inaccuracies within the prior distribution, thus it is unclear as to the interpretation of the posterior with regards to whose subjective views it represents.

A related issue pertains to the introduction of likelihood hyperparameters which in turn require a prior specification. A common approach is to fix them at their maximum likelihood estimates given the observed data. An alternative is the use of an objective prior distribution for any hyperparameters; a common practice in many applications [13]. Such objective priors are designed to be minimally informative, however this does not resolve the issue since there exist multiple forms of objective priors (Jeffrey's prior, reference priors, maximum entry priors, to name a few), hence conclusions of the analysis implicitly depend on this choice. Moreover, in some scenarios, particularly in high-dimensions, objective priors are also used for other uncertainties within the problem and thus do not accurately represent the current knowledge and are incompatible within a subjective analysis.

A further challenge within a full Bayesian analysis is computational tractability. Complex posterior probability distributions require the use of computational methods such as Markov Chain Monte Carlo (MCMC) [16] in order to make inferences. Such methods are computationally expensive and take a long time to perform, with the problem exacerbated with increasing dimensionality. To alleviate these issues, prior probability and likelihood distributions may be chosen for computational convenience such as through the use of conjugate prior families, rather than to reflect the true prior beliefs surrounding the uncertainties.

The Bayes linear approach alleviates many of these challenges. There is no need to specify a full probabilistic prior distribution, instead a second-order analysis requires only prior expectations, variances and covariances. Although this appears to require the specification of a larger number of quantities, it is in fact simpler than a full probability distribution since these are its moments, each of which is simpler to comprehend and elicit as belief statements about physical quantities. Importantly these prior beliefs can be verified to ensure an accurate reflection of current know-

ledge, with second-order exchangeability arguments used to transfer these beliefs to populations of models [60, 62, 71]. There is also no requirement for the specification of a likelihood since the Bayes linear update formulae in Equations (2.2.3) to (2.2.5) explain how to update prior beliefs given the observed data which is computationally tractable and is usually much simpler than full conditioning. This is particularly useful for analyses requiring iterative belief updates, for example, history matching [27, 29, 171] or decision support, as well as applications where answers are required quickly. Moreover, the Bayes linear approach provides a coherent means of handling partial prior belief specifications via the Temporal Sure Preference (TSP) principle which is used to establish a link between judgements at different time points [57, 59, 61]. This is connected with the concepts of partial adjusted expectation and partial resolved variance discussed in [27].

Further rationale for implementing a Bayes linear analysis stems from whether the objectives of a study can be sufficiently achieved without the need for a full probabilistic specification which may only lead to small improvements in terms of the information gained, yet comes with the risk of being non-robust to potential misspecification. Such considerations can be seen in history matching [171] compared to a full Bayesian calibration [107]. Moreover, a Bayes linear analysis may be used as a precursor to a full Bayesian analysis, both to ensure that prior beliefs are accurate, but also to initially restrict to regions of the parameter spaces which are not in conflict with the observed data. Note that if a full Bayes analysis is feasible, more detailed inferences may be drawn because of the additional information about all relationships contained within the posterior probability distribution versus the relevant Bayes linear adjusted quantities.

There are several perspectives on the interpretation of a Bayes linear analysis, as outlined in [57]. It may be viewed as a pragmatic compromise to a full Bayesian analysis replacing a costly full probabilistic prior specification with moments of the uncertain quantities of interest [57, 171]. Alternatively, the Bayes linear updates for uncertain quantities can be seen as an estimator that linearly combines prior

knowledge with observations [57, 185]. A further interpretation is a generalisation to conditional expectation which does not require the strict condition on indicator functions for a partition [30, 57, 70]. On a foundational level, the TSP provides a temporal coherence condition relating actual belief revisions with a formal analysis based on partial prior specifications which is derived through a stochastic model. This is described in more detail in [61]. Each of these perspectives of a Bayes linear analysis leads to interpretational differences between the updated quantities, in particular credible intervals, and their related notion within a full Bayesian analysis. Although there are similarities when a joint Gaussian distribution is assumed. Unsurprisingly each approach possesses its own set of interpretive and diagnostics tools.

The full Bayesian approach is of most value when the prior probability and likelihood distributions can be accurately specified, and when the resultant posterior distribution provides additional information to the analysis. The author believes that the Bayes linear approach should be adhered to whenever these conditions are not met for the reasons outlined above, and will be used throughout this thesis.

2.3 Design of Computer Experiments

Statistical inference using computer models requires a collection of simulations from which to learn presenting the question of how to construct designs to best achieve the aims of a study. It must be emphasised that prior to evaluating the simulator, the relationship between inputs and outputs is unknown. Constraints on computational resources limits the number of simulations which may be performed for which the design size must first be determined, whilst for sequential analyses consideration must also be given to the allocation of this computational budget at each stage. It is assumed that all computer model inputs are continuous and possess a fixed and finite range, whilst it is also common to assume the input ranges are independent leading, to a hypercube parameter space. Note that for deterministic computer models there is no requirement for replication.

Types of designs are discussed in Section 2.3.1 with several design generation methods described including: factorial designs in Section 2.3.2; uniform designs in Section 2.3.3; and Latin hypercube designs in Section 2.3.4. Other design methods such as Sobol sequences exist. The choice of design may be optimised with respect to design selection criteria which are presented in Section 2.3.5. A comprehensive treatment of the design for computer experiments can be found in [153].

2.3.1 Types of Designs

There are three types of design algorithms [152]. Single-stage methods, such as uniform and Latin hypercube designs, involve fixing the number of design points which are simultaneously chosen before any optimisation with respect to a design selection criterion. These are often preferred for generating an initial exploratory set of simulations and are most useful when the computational budget is known. Note that single-stage designs can be very computationally expensive to optimise for large designs.

Sequential methods with or without adaptation involve successively choosing design points. These are advantageous when the computational budget is unknown, whilst optimisation with respect to a criterion is generally less computationally expensive requiring a sequence of D-dimensional optimisations. The distinction between the two forms is that adaptation permits the modification of earlier design points as later ones are chosen. Note that without adaptation, the next simulation may be launched whilst still choosing subsequent design points. This also permits selecting design locations in light of the results of simulations in a myopic or one-step-ahead design, although this does not make best use of any parallel computing resources. A generalisation involves simultaneously selecting small batches of points.

2.3.2 Factorial Designs

Factorial designs, also known as grid designs, are a simple mechanism for sampling over a hypercube [34]. For a D-dimensional parameter space, sample n_j values for

the j^{th} parameter taking all possible combinations to construct an $n_1 \times n_2 \times \cdots \times n_D$ grid. Critically the number of simulations required grows exponentially making factorial designs computationally infeasible, even for a modest number of dimensions and relatively inexpensive simulators. Further disadvantages include: poor coverage both for the full parameter space and in projection onto sub-spaces; and wasting computational resources through unnecessary repetition when a parameter has little effect on the simulator output, termed an inactive parameter. Consequently, factorial designs are rarely used in practice.

2.3.3 Uniform Designs

A desirable property of a design is that points are uniformly spread over the parameter space. This is achieved via uniform sampling, also known as simple random sampling, in which points are sampled from a uniform distribution over the relevant space [121, 153]. A limitation is that points may be close to each other by chance, thus inhibiting the potential information gain from simulations, whilst projections onto lower-dimensional sub-spaces may also possess this undesirable property. This issue can be addressed using a discrepancy criterion to compare the empirical distribution with the cumulative distribution function for a uniform distribution over the parameter space, selecting a candidate design which minimises this discrepancy [153]. An alternative is stratified random sampling which attains a smaller estimator variance versus random sampling [121].

For situations where the shape of this space is complex, rejection sampling may be implemented, although this method does not scale efficiently as the dimensionality increases. This scenario arises within iterative procedures such as history matching where methods of efficiently constructing uniform designs over spaces of unknown shape constituting a very small fraction of their original volume are presented in [186]. Uniform design techniques, including the use of a discrepancy criterion, can also be extended to sampling from other distributions, although these often require Monte Carlo methods and hence may be intractable in higher dimensions.

2.3.4 Latin Hypercube Designs

Latin hypercube designs are a form of space filling design meaning they have no large gaps between design points [121, 153]. An n-point Latin hypercube design is constructed by dividing the range of each parameter into n-intervals of equal length, uniformly sampling a single value from each. The full design is constructed by joining random permutations of the samples for each parameter, potentially to satisfy a design criterion, see Section 2.3.5. Consequently there are no overlaps between points when projected onto the hypercube formed by any subset of the parameters. This is referred to as the Latin hypercube property and leads to an approximately orthogonal design; a desirable attribute. Note that any sub-design also satisfies this property. Latin hypercube designs are simple to generate irrespective of dimension and thus are a popular choice across large parts of the computer experimentation literature, for example [2, 28, 171, 178]. Unlike uniform designs, these rely on the parameter space being a hypercube, whilst implementation of rejection sampling will break the Latin hypercube property.

2.3.5 Design Selection Criteria

Design selection criteria are used to select the most suitable over a potentially infinite collection by optimising with respect to a rule [153]. Let $\mathcal{D}$ denote a design.

Space filling designs are often desired with several common criteria specified with respect to a distance metric such as the p-norm or L^p-norm between two D-dimensional inputs $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ defined in Equation (2.3.1) for $p \geq 1$. The case $p = 2$ corresponds to the Euclidean distance metric. To enable comparison of parameters with different ranges, it is good practice to scale each parameter onto the range $[0, 1]$.

$$\rho_p(\mathbf{x}, \mathbf{x}') \coloneqq \left[\sum_{j=1}^{D} |x_j - x'_j|^p \right]^{1/p} \tag{2.3.1}$$

A maximin design, $\mathcal{D}_{Mm}$, defined in equation (2.3.2), maximises the minimum distance between any pair of parameter vectors within the design, thus resulting in

points which are spread out over $\mathcal{X}$. This can lead to "holes" in the parameter space as design points tend to be pushed towards the boundary. Maximin Latin hypercube designs are frequently used across the computer experimentation literature, for example in [2, 40, 107, 171, 173].

$$\mathcal{D}_{Mm} \coloneqq \arg\max_{\mathcal{D}\subset\mathcal{X}} \min_{\mathbf{x}_1,\mathbf{x}_2\in\mathcal{D}} \rho_p(\mathbf{x}_1,\mathbf{x}_2) \tag{2.3.2}$$

An alternative approach uses the minimum distance between any $\mathbf{x} \in \mathcal{X}$ and its nearest neighbour in $\mathcal{D}$ defined in Equation (2.3.3).

$$\rho_p(\mathbf{x},\mathcal{D}) \coloneqq \min_{\mathbf{x}_i\in\mathcal{D}} \rho_p(\mathbf{x},\mathbf{x}_i) \tag{2.3.3}$$

The minimax design, $\mathcal{D}_{mM}$, defined in Equation (2.3.4), minimises the nearest neighbour distance and hence reduces the possibility of "holes" in the parameter space. This is desirable with a view to emulation. For even moderately high-dimensional parameter spaces it is not tractable to analytically evaluate the nearest neighbour distance for all $\mathbf{x} \in \mathcal{X}$. In practical implementation $\mathcal{X}$ is substituted for a large uniform sample or grid of points $\mathbf{S}$, which is sufficiently large to achieve adequate coverage.

$$\mathcal{D}_{mM} \coloneqq \arg\min_{\mathcal{D}\subset\mathcal{X}} \max_{\mathbf{x}\in\mathcal{X}} \rho_p(\mathbf{x},\mathcal{D}) \tag{2.3.4}$$

A generalisation is to consider the average distance between design points defined in Equation (2.3.5), where n is the size of the design and $d^{1/p}$ is a normalisation constant chosen such that $d^{1/p} \geq \max_{\mathbf{x}_i,\mathbf{x}_j\in\mathcal{D}} \rho_p(\mathbf{x}_i,\mathbf{x}_j)$ for all $\mathbf{x}_i \neq \mathbf{x}_j$, making distances comparable for pairs of points in different dimensions.

$$m_{(p,\lambda)}(\mathcal{D}) \coloneqq \left(\frac{1}{\binom{n}{2}} \sum_{\mathbf{x}_i,\mathbf{x}_j\in\mathcal{D}} \left[\frac{d^{1/p}}{\rho_p(\mathbf{x}_i,\mathbf{x}_j)} \right]^{\lambda} \right)^{1/\lambda}, \quad \text{for } \lambda \geq 1 \tag{2.3.5}$$

The minimum average distance design, $\mathcal{D}_{av}$, defined in Equation (2.3.6), minimises the average distance between design points favouring those which possess non-

redundancy between design point locations.

$$\mathcal{D}_{av} \coloneqq \min_{\mathcal{D} \subset \mathcal{X}} m_{(p,\lambda)}(\mathcal{D}) \tag{2.3.6}$$

When $\lambda = 1$, this criterion selects the design which maximises the harmonic mean distance between all pairs of design points which avoids clustering of design points. In the limit as $\lambda \to \infty$, this is equivalent to selecting the maximin design.

Beyond space filling designs, there exist many statistical criteria based on the form of the model to be fitted [153]. Design criteria based on a function of the covariance matrix defined over a set of outputs include: D-optimality, which minimises its determinant; A-optimality, which minimises its trace; and I-optimality, which minimises the average or Integrated Mean Squared Prediction Error (MSPE) over the input space, also known as the IMSE criterion. A variant on I-optimality is V-optimality which minimises the MSPE over a subset of input points. Designs may also be considered which minimise the Maximum Mean Squared Prediction error (MMSPE). Maximum entropy designs aim to reduce unpredictability [34, 152, 153]. For sequential designs it is common to use Bayesian optimal design criteria, for example: Probability Of Improvement (POI or PI); Expected Improvement (EI) [102]; multi-points EI [56]; Bayesian expected loss; and Lower or Upper Confidence Bounds (LCB or UCB) [15].

For all of the presented design selection criteria, as the input dimension and the number of design points increase, the computational expense of optimisation also increases rapidly whilst becoming unreliable. Consequently, practical design selection is usually performed by comparing a large collection of candidates with respect to a design selection criterion. Moreover, designs optimised with respect to specific criteria will possess certain desirable properties, but also have weaknesses. A means of averting this is to use multiple design criteria. For example, a first design criterion is used to obtain a restricted class of candidate designs on which a second criterion is applied to select a final design [153].

2.4 Bayesian Emulation

An emulator is a stochastic belief specification for a deterministic function that provides a fast and efficient statistical approximation, yielding predictions for as yet unevaluated parameter settings, along with a corresponding statement of the uncertainty [26, 29, 129, 132, 171]. These are also known as surrogate or meta-models. In Bayesian emulation, this involves combining prior knowledge with information gained from a (small) batch of simulations. Emulators are frequently employed as an approximation to computer models for performing tasks including: calibration; history matching; uncertainty quantifications; sensitivity analyses; and decision support. This is due to the limitations of computer models discussed in Section 2.1.2 which makes their direct use infeasible.

The structure of an emulator is presented in Section 2.4.1, whilst the choice of covariance function is discussed in Section 2.4.2. Before fitting a Bayesian emulator, it is necessary to construct a prior belief specification which is explored in Section 2.4.3. A fully Bayesian technique in the form of Gaussian process emulation and Bayes linear emulators are described in Sections 2.4.4 and 2.4.5 respectively. Emulator diagnostics techniques are reviewed in Section 2.4.6.

2.4.1 Emulator Structure

A Bayesian emulator for the i^{th} output of a simulator, $f(\mathbf{x})$, denoted by the function, $f_i(\mathbf{x})$, takes the general form shown in Equation (2.4.1), as seen in [29, 30, 171, 178], where $\mathbf{x}$ denotes a vector of input parameters. This may be either model or decision parameters, or a combination of the two.

$$\begin{aligned} f_i(\mathbf{x}) &= \mathbf{g}_i(\mathbf{x}_{A_i})^{\mathrm{T}}\boldsymbol{\beta}_i + u_i(\mathbf{x}_{A_i}) + w_i(\mathbf{x}) \\ &= \sum_{j=1}^{p} \beta_{ij} g_{ij}(\mathbf{x}_{A_i}) + u_i(\mathbf{x}_{A_i}) + w_i(\mathbf{x}) \end{aligned} \tag{2.4.1}$$

The subscript A_i denotes a subset of active inputs which are the parameters deemed to be most influential for $f_i(\mathbf{x})$. Within the emulator equation, the first term models

the global behaviour of $f_i(\mathbf{x})$ where the $g_{ij}(\cdot)$ are deterministic functions of the active inputs with unknown scalar regression coefficients, β_{ij} for $j = 1, \ldots, p$, where $p \in \mathbb{N}$. Collectively, these are denoted by the vector function $\mathbf{g}_i(\cdot) = \begin{pmatrix} g_{i1}(\cdot) \\ \vdots \\ g_{ip}(\cdot) \end{pmatrix}$, and the vector $\boldsymbol{\beta}_i = \begin{pmatrix} \beta_{i1} \\ \vdots \\ \beta_{ip} \end{pmatrix} \in \mathbb{R}^p$ respectively. The second term, $u_i(\cdot)$, models the local behaviour of $f_i(\mathbf{x})$ with respect to the active parameters. It is a weakly stationary stochastic process, see Definition 2.4.1, with zero mean and a pre-specified covariance structure.

Definition 2.4.1. Let $\zeta(\cdot)$ be a stochastic process with input vector $\mathbf{x} \in \mathcal{X}$, a parameter space. $\zeta(\mathbf{x})$ is described as **weakly stationary** if $\mathbb{E}[\zeta(\mathbf{x})] = a$ for all $\mathbf{x} \in \mathcal{X}$ where $a \in \mathbb{R}$ is a constant, and $\text{Cov}[\mathbf{x}, \mathbf{x}'] = c(\mathbf{x} - \mathbf{x}')$ for some function $c(\cdot)$ and all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$. That is $\zeta(\cdot)$ possesses a constant mean and a covariance function which is invariant to translations.
Note that a process is **strictly stationary** if it has constant mean, with all finite dimensional distributions also being invariant to translations of the parameters.

There exist many forms of covariance functions with a common choice being the squared exponential covariance function, also known as the Gaussian form, for which a general formula is shown in Equation (2.4.2), where $\sigma^2_{u_i}$ is the variance hyperparameter [146, 171] and the dependency between the active inputs is encapsulated by the variance matrix M. An alternative parameterisation involves the often reasonable simplifying assumption that the inputs are uncorrelated, each with its own distinct correlation length, also known as (characteristic) length scale hyperparameters, contained in the D-vector $\boldsymbol{\theta}_i = (\theta_{i1}, \ldots, \theta_{iD})$. This is equivalent to specifying that M is diagonal such that $M = \text{diag}\{\theta^2_{i1}, \ldots, \theta^2_{iD}\}$. A further assumption is a common correlation length such that $\theta_{i1} = \cdots = \theta_{iD}$. Alternative covariance functions are explored in Section 2.4.2.

$$\text{Cov}[u_i(\mathbf{x}_{A_i}), u_i(\mathbf{x}'_{A_i})] = \sigma^2_{u_i} \exp\left\{ -\left(\mathbf{x}_{A_i} - \mathbf{x}'_{A_i}\right)^{\mathrm{T}} M^{-1} \left(\mathbf{x}_{A_i} - \mathbf{x}'_{A_i}\right) \right\} \tag{2.4.2}$$

$$\text{Cov}[u_i(\mathbf{x}_{A_i}), u_i(\mathbf{x}'_{A_i})] = \sigma^2_{u_i} \exp\left\{ -\sum_{k=1}^{D} \left(\frac{x_{A_i,k} - x'_{A_i,k}}{\theta_{ik}} \right)^2 \right\} \tag{2.4.3}$$

The third term in Equation (2.4.1), $w_i(\mathbf{x})$, is an uncorrelated, zero-mean nugget term with covariance specified as:

$$\text{Cov}[w_i(\mathbf{x}, \mathbf{x}')] = \sigma^2_{w_i} \mathbb{1}_{\{\mathbf{x}=\mathbf{x}'\}} \tag{2.4.4}$$

This may also be thought of as a white noise process. Emulators may be constructed which do not contain a nugget term, although there are strong arguments for its inclusion discussed in [1, 75]. Two common arguments are for computational numerical stability, as favoured in [107], as well as to account for the remaining variability in $f_i(\mathbf{x})$ due to the inactive parameters, as discussed in [30, 171]. Further arguments pertain to failures in the underlying modelling assumptions of: simulator bias; stationarity; correct choice of covariance function; and that the simulator is deterministic in both the non-modelable and theoretical but not numerical sense. Inclusion of a nugget term protects against poor emulator fit compared with the real physical process and ensures that the emulator does not interpolate any fitting data [1, 75]. Note the similarities with model discrepancy within an uncertainty analysis discussed in Section 2.7. Collectively the weakly stationary stochastic process and nugget term are thought of as the residual process.

In order to fairly evaluate the effect of each parameter for each simulator output, where possible, such as when parameters have a fixed physical or operational range, it is good practice to linearly transform all parameters onto the range $[-1, 1]$, hence obtaining a parameter space $\mathcal{X} = [-1, 1]^D$ in D-dimensions. Prior knowledge from domain experts may be combined with information gained from an exploratory collection of simulations through graphical investigations plotting output versus each input along with linear regression modelling to guide the choice of active parameters, as well as appropriate forms for the deterministic functions, $g_{ij}(\cdot)$. This may involve the use of the stepwise selection method, with direction either forward, backward or both (also known as mixed), along with a model selection criterion such as Mallows' C_p, AIC, BIC or adjusted R^2 [98, 183]. Experience suggests both-direction stepwise selection using the AIC criteria is a good choice for constructing emulators

because this permits a more extensive exploration of possible models, whilst also not penalising higher numbers of model terms compared with BIC. Further discussion can be found in Appendix A.

A Bayesian emulator first requires a prior belief specification which is discussed in Section 2.4.3 including a distinction between a full Bayesian and a Bayes linear emulator. Simulator evaluations, also known as runs or data, are then necessary to perform a Bayesian update. Consider a design of n points where $\mathbf{x}^{(k)},\ k = 1, \ldots, n$, represents a vector of the input settings and for a univariate output, $f_i(\mathbf{x})$, the simulator evaluations are denoted by a vector $\mathbf{F}_i = (f_i(\mathbf{x}^{(1)}), \ldots, f_i(\mathbf{x}^{(n)}))^{\mathrm{T}}$. The update process for a Bayesian emulator will be presented in Sections 2.4.4 and 2.4.5 for a Gaussian process and Bayes linear emulator.

There exist two approaches to fitting an emulator. The simultaneous or single-step process involves firstly determining a suitable emulator structure, constructing a prior belief specification and then using the simulation data to perform a Bayesian update. This has the benefit of being quick to perform and can be completely automated. An alternative approach is sequential fitting. First a linear regression model is fitted using the simulator runs to estimate the regression coefficients, β_{ij}, whilst accounting for the uncertainty. In a second order belief specification, this includes: $\mathbb{E}[\boldsymbol{\beta}_i]$ and $\mathrm{Var}[\boldsymbol{\beta}_i]$. Secondly, a weakly stationary stochastic process is fitted to the residuals of the regression model using an appropriate estimate for $\sigma^2_{u_i}$ such as the fitted linear model residual variance, whilst input covariance structure may be specified a priori. The final part is to fit the nugget term (if in use), with the parameter $\sigma^2_{w_i}$ also either specified a priori, or obtained via suitable estimates from the residuals of the joint linear model and weakly stationary stochastic process. Compared with the simultaneous approach, the sequential method has the main advantage of the interpretability of each part in accounting for the behaviour of the function. Moreover, it avoids confounding issues between the regression terms and the weakly stationary stochastic process that can lead to unrealistic estimates for the regression coefficients. An example of this is seen in [153] for an emulator using

a constant regression term only, which is estimated to be too far above or below the data. In this thesis the simultaneous approach is used.

2.4.2 Covariance Functions

The emulator formulation in Equation (2.4.1) requires the prior specification of a covariance function. This measures the similarity in the output of a function between pairs of input parameter settings, $\mathbf{x}$ and $\mathbf{x}'$. A detailed discussion of covariance functions can be found in [146, ch. 4].

Covariance functions, also referred to as covariance kernels, commonly use the notation $\kappa(\mathbf{x}, \mathbf{x}')$ with the corresponding correlation function (or kernel) written as $r(\mathbf{x}, \mathbf{x}')$, with the relationship for a univariate output being that $\kappa(\mathbf{x}, \mathbf{x}') = \sigma^2 r(\mathbf{x}, \mathbf{x}')$, where $\sigma^2 > 0$ is a variance hyperparameter which controls the magnitude of the computer model output variance. For the weakly stationary stochastic process, this is $\sigma^2 = \sigma^2_{u_i}$. Given a collection of n parameter vectors such as the design for an emulator, $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\}$, the covariance function is evaluated for each pair of inputs to construct a covariance matrix, Σ, where $\Sigma_{ij} = \kappa(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ for the computer model output. Consequently, not all functions which map two vector arguments to a real number are valid covariance functions because the resultant covariance matrix must be positive semi-definite. There exists many classes of covariance functions, with the choice depending on beliefs about the function behaviour and thus what characteristics must be reflected in the covariance function. For the purposes of emulation a common stipulation is that both the simulator output as a function, and the covariance function are stationary which is explained in Definition 2.4.2.

Definition 2.4.2. A (covariance or correlation) function, $\kappa(\mathbf{x}, \mathbf{x}')$, is **weakly stationary** if is invariant to translations in $\mathcal{X}$, thus for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, it satisfies $\kappa(\mathbf{x}, \mathbf{x}') = \kappa(\mathbf{x} + \mathbf{h}, \mathbf{x}' + \mathbf{h})$ for all $\mathbf{h}$ such that $\mathbf{x} + \mathbf{h}, \mathbf{x}' + \mathbf{h} \in \mathcal{X}$. Consequently $\kappa(\mathbf{x}, \mathbf{x}') = c(\mathbf{x} - \mathbf{x}')$ for some function $c(\cdot)$.

Two further properties are that covariance function is either isotropic or aniso-

tropic which are defined in Definitions 2.4.3 and 2.4.4 respectively. Note that any isotropic or anisotropic covariance function is also stationary. The squared exponential covariance function is a common example of a stationary covariance function with the forms given in Equations (2.4.2) and (2.4.3) being anisotropic and isotropic respectively. Starting from an anisotropic covariance function, it is simple to obtain an isotropic covariance function by assuming M is diagonal with elements equal to $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_D)$ respectively, and defining $\|\cdot\|$ such that $\|\mathbf{x} - \mathbf{x}'\| = \sum_{k=1}^{D} \left(\frac{x_k - x'_k}{\theta_k}\right)$. In addition, it can be challenging to construct a positive semi-definite M which is non-diagonal, particularly such that it accurately reflects prior beliefs about the computer model inputs. Combined with the "black-box" nature of computer models, it is common practice to assume that M is diagonal.

Definition 2.4.3. A (covariance or correlation) function, $\kappa(\mathbf{x}, \mathbf{x}')$, is **isotropic** if it invariant to both translations and rotations of $\mathcal{X}$. Consequently, for a specified norm map $\|\cdot\|$, the function can be expressed in terms of $\|\mathbf{x} - \mathbf{x}'\|$ such that $\kappa(\mathbf{x}, \mathbf{x}') = c(\|\mathbf{x} - \mathbf{x}'\|)$ for some function $c(\cdot)$.

Definition 2.4.4. A (covariance or correlation) function, $\kappa(\mathbf{x}, \mathbf{x}')$, is **anisotropic** if it can be expressed as $\kappa(\mathbf{x}, \mathbf{x}') = c((\mathbf{x} - \mathbf{x}')^{\mathrm{T}} M^{-1} (\mathbf{x} - \mathbf{x}'))$ for some function $c(\cdot)$, where M is a positive semi-definite matrix, such that $M \neq \lambda^2 I_D$ for some $\lambda > 0$, which encapsulates the covariance between the individual inputs.

Examples of stationary correlation functions stated in their anisotropic form are listed in Table 2.1 for a pair of D-dimensional parameter vectors, $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ where M is a positive semi-definite covariance matrix for this inputs. A stationary covariance function is obtained by the scaling $\kappa(\mathbf{x}, \mathbf{x}') = \sigma^2 r(\mathbf{x}, \mathbf{x}')$ as above.

The choice of covariance function should depend on prior beliefs about the mean square continuity and differentiability of the computer model output in the stochastic emulator representation, viewed as a stochastic function. These are defined as:

Definition 2.4.5. A univariate function, $f(\mathbf{x})$, is **mean square continuous** if for all $\mathbf{x}^* \in \mathcal{X}$, there exists a sequence of points $\mathbf{x}_1, \mathbf{x}_2, \ldots$ with $\|\mathbf{x}_i - \mathbf{x}^*\| \to 0$ as $i \to \infty$,

Correlation Function	$r(\mathbf{x}, \mathbf{x}')$
Constant	1
Exponential	$\exp\{-\psi(\mathbf{x}, \mathbf{x}')\}$
Squared Exponential	$\exp\{-\psi^2(\mathbf{x}, \mathbf{x}')\}$
γ-Exponential	$\exp\{-\psi^\gamma(\mathbf{x}, \mathbf{x}')\}$ for $0 < \gamma \leq 2$
Matérn Class	$\frac{1}{2^{\nu-1}\Gamma(\nu)} \left(\sqrt{2\nu}\ \psi(\mathbf{x}, \mathbf{x}')\right)^\nu K_\nu \left(\sqrt{2\nu}\ \psi(\mathbf{x}, \mathbf{x}')\right)$ for $\nu > 0$
Rational Quadratic	$\left(1 + \frac{\psi^2(\mathbf{x}, \mathbf{x}')}{2\alpha}\right)^{-\alpha}$ for $\alpha > 0$

Table 2.1: Table of valid stationary correlation functions in the anisotropic form for pairs of D-dimensional input parameters, $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, where M is a positive semi-definite input covariance matrix. In each expression, $\psi^2(\mathbf{x}, \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^{\mathrm{T}} M^{-1} (\mathbf{x} - \mathbf{x}')$. A valid stationary covariance function is obtained via the scaling, $\kappa(\mathbf{x}, \mathbf{x}') = \sigma^2 r(\mathbf{x}, \mathbf{x}')$, where $\sigma^2 > 0$ is a variance hyperparameter which controls the magnitude of the computer model output variance. For the Matérn class of covariance functions: $\nu > 0$ is an additional hyperparameter which controls the number of times which simulator output (as a function) is mean-square differentiable; $\Gamma(\cdot)$ is the Gamma-function; and $K_\nu(\cdot)$ denotes the modified Bessel function of the second kind.

then $\mathbb{E}[\|f(\mathbf{x}_i) - f(\mathbf{x}^*)\|^2] \to 0$ as $i \to \infty$.

Definition 2.4.6. A univariate function, $f(\mathbf{x})$, is **mean square differentiable** if for all $\mathbf{x} \in \mathcal{X}$, the mean square derivative:

$$\frac{\partial f(\mathbf{x})}{\partial x_i} = \lim_{h \to 0} \mathbb{E}\left[\left\| \frac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h} \right\|^2\right] \tag{2.4.5}$$

exists and is finite for all $i = 1, \ldots, D$, where $\mathbf{e}_i$ is the canonical basis vector in the i^{th} direction. This definition naturally extends to higher order derivatives if $\frac{\partial^k f(\mathbf{x})}{\partial x_{i_1} \ldots \partial x_{i_k}}$ exist and are finite for all $\mathbf{x} \in \mathcal{X}$ and combinations of $x_{i_1}, \ldots, x_{i_k}$, possibly with repeated parameters.

Mean square continuity is satisfied if and only if the covariance function $\kappa(\mathbf{x}, \mathbf{x}')$ is continuous for all $\mathbf{x} = \mathbf{x}'$ with $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$. The covariance of $\frac{\partial f(\mathbf{x})}{\partial x_i}$ is determined by

$\frac{\partial^2\kappa(\mathbf{x},\mathbf{x}')}{\partial x_i \partial x_i'}$. It is therefore required that all $2k^{\text{th}}$ partial derivatives of $\kappa(\mathbf{x},\mathbf{x}')$ exist and are finite for all $\mathbf{x}=\mathbf{x}'$ in order for the $f(\mathbf{x})$ to be k-times mean square differentiable. For stationary covariance functions this reduces to an assessment at $\mathbf{x}=\mathbf{0}$.[3]

In Table 2.1, the constant covariance function, $\kappa(\mathbf{x},\mathbf{x}')=\sigma^2$, assumes that there is equal correlation between all pairs of inputs, irrespective of their separation. This leads to $\Sigma_{ij}=\sigma^2$ for all $i,j=1,\ldots,n$, hence Σ is of finite rank and so is non-invertible. This seems to be an unrealistic choice of covariance function for almost all computer models. Note that all other correlation functions specified in Table 2.1 are non-degenerate and hence lead to an invertible Σ.

The exponential covariance function is mean square continuous, but not mean square differentiable. Note that a stochastic process using an exponential covariance function corresponds to a model of the velocity of a particle under Brownian motion and is also known as an Ornstein-Uhlenbeck process. The squared exponential covariance function, as presented in Equation (2.4.2), is a popular choice of covariance function which possesses the following properties: it is positive for all $\mathbf{x},\mathbf{x}'\in\mathcal{X}$; it decays monotonically at an exponential rate in $(\mathbf{x}-\mathbf{x}')^{\mathrm{T}}M^{-1}(\mathbf{x}-\mathbf{x}')$; and it is infinitely mean square differentiable. It is therefore well suited to capturing the correlation between pairs of inputs for functions with derivatives of all orders. These are generalised by the γ-exponential family covariance functions which are mean square continuous, but only mean square differentiable for $\gamma=2$. Note that where M is assumed to be diagonal, these are also separable, as stated in Definition 2.4.7.

Definition 2.4.7. A covariance function, $\kappa(\mathbf{x},\mathbf{x}')$, for $\mathbf{x},\mathbf{x}'\in\mathcal{X}\subset\mathbb{R}^D$ is described as **separable** if it can be factorised into the form shown in Equation (2.4.6), where each $\kappa_i(\cdot),\ i=1,\ldots,D$ are valid covariance functions [132].

$$\kappa(\mathbf{x},\mathbf{x}')=\prod_{i=1}^{D}\kappa_i(x_i,x_i') \tag{2.4.6}$$

The Matérn class of covariance functions [119] introduces greater flexibility in

[3]Mean square continuity or differentiability do not necessarily imply sample continuity or differentiability of a function [146].

the modelling assumptions through the additional hyperparameter, $\nu > 0$. The function $f(\mathbf{x})$ is k-times mean square differentiable if and only if $k < \nu$, hence the roughness is controlled by ν with higher values corresponding to smoother output. For $\nu = p + \frac{1}{2}$ where $p \in \mathbb{Z}_{\geq 0}$, the Matérn covariance function expression simplifies to a product of an exponential covariance function and a polynomial of order p, where $p = 0$ $\left(\nu = \frac{1}{2}\right)$ returns an exponential covariance function. Note that a squared exponential covariance function is recovered by taking the limit as $\nu \to \infty$. The rational quadratic covariance function in Table 2.1 can be viewed as a scale mixture of squared exponential covariance functions with different prior input covariance matrix, M. Unlike the Matérn class, these are infinitely mean square differentiable everywhere for all $\alpha > 0$.

Beyond stationary covariance functions there exist various other forms. An interesting class is dot product covariance functions which depend on $\mathbf{x}$ and $\mathbf{x}'$ via an inner product, $\mathbf{x} \cdot \mathbf{x}'$, and hence are rotationally invariant, but not translational invariant. Examples of the simple and inhomogeneous polynomial dot product covariance functions are given as:

$$\kappa(\mathbf{x}, \mathbf{x}') = (\sigma^2 + \mathbf{x} \cdot \mathbf{x}') \tag{2.4.7}$$

$$\kappa(\mathbf{x}, \mathbf{x}') = (\sigma^2 + \mathbf{x} \cdot \mathbf{x}')^p, \quad \text{for} \quad p > 0 \tag{2.4.8}$$

The neural network covariance function is defined in Equation (2.4.9) where $\tilde{\mathbf{x}} = (1, \mathbf{x})$, and $\tilde{M}$ is an extension to M. Another example is the Gibbs covariance functions specified in Equation (2.4.10) where each $\ell_d(\cdot)$ is an arbitrary positive function which is referred to as a length-scale function. These play an analogous role to the correlation length hyperparameters seen in Equation (2.4.3), but are free to vary across the parameter space. This idea may be extended to the use of other stationary covariance functions with hyperparameters that depend on input settings, whilst another option is modelling the input variance matrix to also be a function of input location, $M(\mathbf{x})$ [148]. A full discussion along with further non-stationary

covariance functions is found in [146].

$$\kappa(\mathbf{x}, \mathbf{x}') = \frac{2}{\pi} \sin^{-1} \left(\frac{2\tilde{\mathbf{x}}^{\mathrm{T}} \tilde{M} \tilde{\mathbf{x}}'}{\sqrt{(1 + 2\tilde{\mathbf{x}}^{\mathrm{T}} \tilde{M} \tilde{\mathbf{x}})(1 + 2\tilde{\mathbf{x}}'^{\mathrm{T}} \tilde{M} \tilde{\mathbf{x}}')}} \right) \tag{2.4.9}$$

$$\kappa(\mathbf{x}, \mathbf{x}') = \prod_{d=1}^{D} \left(\frac{2\ell_d(\mathbf{x})\ell_d(\mathbf{x}'))}{\ell_d^2(\mathbf{x}) + \ell_d^2(\mathbf{x}')} \right)^{1/2} \exp \left\{ - \sum_{d=1}^{D} \frac{(x_d - x_d')^2}{\ell_d^2(\mathbf{x}) + \ell_d^2(\mathbf{x}')} \right\} \tag{2.4.10}$$

A major challenge for designing a new covariance function is ensuring that it will always yield a valid covariance matrix. New and valid covariance functions may be constructed from existing ones: using the binary operations of addition and multiplication; through vertical rescaling via a deterministic function, $a(\mathbf{x})$, to obtain $\tilde{\kappa}(\mathbf{x}, \mathbf{x}') = a(\mathbf{x})\kappa(\mathbf{x}, \mathbf{x}')a(\mathbf{x}')$; and convolutions of covariance kernels. Moreover, in the situation of two disjoint parameter spaces with covariance functions $\kappa_1(\mathbf{x}_1, \mathbf{x}_1')$ for $\mathbf{x}_1, \mathbf{x}_1' \in \mathcal{X}_1$, and $\kappa_2(\mathbf{x}_2, \mathbf{x}_2')$ for $\mathbf{x}_2, \mathbf{x}_2' \in \mathcal{X}_2$, direct sums and tensor products can be used to define a covariance function over the product space $\mathcal{X}_1 \times \mathcal{X}_2$.

Each of the covariance functions discussed in this section requires a set of hyperparameters for which their specification or elicitation is discussed in Section 2.4.3.

2.4.3 Emulator Prior Belief Specification

Before performing a Bayesian update for the emulator given simulation data, a prior specification is required including for the deterministic functions and all of the emulator hyperparameters. In Sections 2.4.4 and 2.4.5, a distinction will be made between this prior specification for a full Bayesian Gaussian process [126] and a Bayes linear emulator respectively [28].

For an emulator of the form shown in Equation (2.4.1), a collection of deterministic functions, $\{g_{ij}(\cdot)\}_{j=1}^{p}$, must be defined and are treated as known, with a prior over their unknown regression coefficients, $\boldsymbol{\beta}_i$, with $\mathrm{Var}[\boldsymbol{\beta}_i] = \Sigma_\beta \in \mathbb{R}^{p \times p}$. Many authors prefer to select a constant mean function [87, 107, 152], whilst a more detailed and structured form is favoured by others [29, 32, 170, 171], with the justification of aiming to reduce the residual variance and in turn decrease the dependence on a

well specified residual process.

The next step is to select a suitable covariance function for the weakly stationary stochastic process, as discussed in Section 2.4.2. The two residual variance hyperparameters may be jointly linked, as in [30, 32, 171], by introducing the proportion $\rho \in [0, 1]$ of the residual variance attributed to the nugget term with $\sigma^2_{u_i} = (1-\rho)\sigma^2_i$ and $\sigma^2_{w_i} = \rho\sigma^2_i$, where σ^2_i is the residual variability after accounting for the global mean function. Two approaches to handling hyperparameters exist: assigning probability distributions to them (which is explored further in Section 2.4.4); or plugging-in values. For the latter approach, hyperparameters are specified a priori, possibly using expert elicitation, and validated using emulator diagnostic techniques [8], as was the case in [27]. For example, when a squared exponential covariance function of the form shown in Equation (2.4.3) is used, correlation lengths are often specified between 0.25 and 0.5 on the scale of each parameter's range, whilst ρ is usually small and often set to be $\rho = 0.05$. Alternatively, Maximum Likelihood or Restricted Maximum Likelihood Estimation (MLE or REML) may be used to estimate any hyperparameters, with the option to simultaneously include $\boldsymbol{\beta}_i$ [34, 153]. Note that issues can arise in maximum likelihood estimation when the covariance function is misspecified [6].

Once these quantities have been specified, and combining with Equation (2.4.1), the prior expectation and covariance for a univariate emulator is obtained from [171]:

$$\mu_i(\mathbf{x}) = \mathbb{E}[f_i(\mathbf{x})] = \sum_{j=1}^{p} \mathbb{E}[\beta_{ij}] g_{ij}(\mathbf{x}_{A_i}) = \mathbf{g}_i(\mathbf{x}_{A_i})^{\mathrm{T}} \boldsymbol{\mu}_{\beta_i} \tag{2.4.11}$$

$$\begin{aligned}
\kappa_i(\mathbf{x}, \mathbf{x}') = {} & \mathrm{Cov}[f_i(\mathbf{x}), f_i(\mathbf{x}')] \\
&= \mathrm{Cov}\left[\sum_{j=1}^{p} \beta_{ij} g_{ij}(\mathbf{x}_{A_i}), \sum_{j=1}^{p} \beta_{ij} g_{ij}(\mathbf{x}'_{A_i})\right] \\
&\quad + \mathrm{Cov}[u_i(\mathbf{x}_{A_i}), u_i(\mathbf{x}'_{A_i})] + \sigma^2_{w_i} \mathbb{1}_{\{\mathbf{x}=\mathbf{x}'\}} \\
&= \mathbf{g}_i(\mathbf{x}_{A_i})^{\mathrm{T}} \Sigma_\beta \mathbf{g}_i(\mathbf{x}'_{A_i}) + \mathrm{Cov}[u_i(\mathbf{x}_{A_i}), u_i(\mathbf{x}'_{A_i})] + \sigma^2_{w_i} \mathbb{1}_{\{\mathbf{x}=\mathbf{x}'\}}
\end{aligned} \tag{2.4.12}$$

For practical applications following a subjective Bayesian analysis, emulator prior belief specification requires elicitation from subject experts. This can be an extremely

challenging process that is problem specific and involves transferring qualitative statements to a quantitative prior specification for which computer programs may be used to assist experts in investigating their prior beliefs [27, 28, 29]. Further methods of prior elicitation within a Bayes linear framework are presented in [147]. Situations where there are multiple experts exhibit further complexities requiring belief aggregation methods. The Sheffield Elicitation Framework, SHELF [72, 130], is a form of behavioural aggregation resulting in a prior belief specification that represents those of a "rational impartial observer". Alternative mathematical means of combining multiple experts' beliefs also exist such as linear pooling in which weights are assigned based on the importance of each expert's opinions. These may be equal or assigned via the classical method which uses seed questions to score experts on their informativeness and calibration ability, although questions arise as to whose beliefs the resulting prior actually represents [24]. Bayesian hierarchical aggregation also uses seed questions with the additional stage of grouping experts to obtain a prior that is designed to represent the prior beliefs of a non-expert decision maker [80].

Within a Bayesian analysis, conclusions depend on the exact choice of both the (likelihood) model and the prior. This is especially true of a full probabilistic specification for which [133] introduces imprecision in the elicited prior specification in order to address this issue. Alternatively, a partial prior specification requiring only low-order moments of the distribution(s), as in a Bayes linear analysis, which are easier to interpret and elicit may be preferable [28]. The robustness of Bayesian analyses may also be investigated by treating them as computer models and using Gaussian process emulators to assess inference sensitivity to the prior specification and choice of likelihoods [172].

2.4.4 Gaussian Process Emulation

Full Bayesian emulation requires the specification of probability distributions for the likelihood and prior. However, for most arbitrary choices of distributions, the

process of updating and sampling from the emulator given simulation data becomes extremely computationally expensive, potentially diminishing any gains made versus the direct use of the simulator within an analysis. In contrast, this is not the case for a Gaussian Process (GP) defined in Definition 2.4.8 as a stochastic process generalisation of the Gaussian probability distribution to functions [107, 146], as long as the variable of interest is also assumed to follow a Gaussian process prior.

Definition 2.4.8. An unknown function $f(\cdot)$ with inputs $\mathbf{x} \in \mathcal{X}$ is a **Gaussian Process (GP)** if and only if for every $n \in \mathbb{N}$, the joint distribution of $(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))$ is a multivariate normal distribution for all $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{X}$.

Imposing a Gaussian process model for the simulation output leads to Gaussian process (GP) emulation. This is a strong probabilistic assertion stating that any finite set of simulator outputs follows a multivariate normal distribution and is frequently not the case. These were initially used effectively within a non-Bayesian setup [34, 146, 152, 153]; but have also been heavily adopted across the Bayesian literature such as in [107, 128, 129, 173]. Notation for a Gaussian process emulator for univariate simulator output $f_i(\cdot)$ is given in Equation (2.4.13), which is fully characterised by the specification of a mean function, $m_i(\cdot)$, and a covariance function, $\kappa_i(\cdot, \cdot)$ defined in Equations (2.4.14) and (2.4.15) respectively.

$$f_i(\cdot) \sim \mathcal{GP}(m_i(\cdot), \kappa_i(\cdot, \cdot)) \tag{2.4.13}$$

$$m_i(\mathbf{x}) = \mathbb{E}[f_i(\mathbf{x})] \tag{2.4.14}$$

$$\begin{aligned} \kappa_i(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f_i(\mathbf{x}) - m_i(\mathbf{x}))(f_i(\mathbf{x}') - m_i(\mathbf{x}'))] \\ &= \mathrm{Cov}[f_i(\mathbf{x}), f_i(\mathbf{x}')] \end{aligned} \tag{2.4.15}$$

Under the framework presented in Section 2.4.1, a consistent specification for the mean and covariance functions is $m_i(\mathbf{x}) = \sum_j \beta_{ij} g_{ij}(\mathbf{x}_{A_i})$ and
$\kappa_i(\mathbf{x}, \mathbf{x}') = \mathrm{Cov}[u_i(\mathbf{x}), u_i(\mathbf{x}')] + \sigma^2_{w_i} \mathbb{1}_{\{\mathbf{x}=\mathbf{x}'\}}$.

Updating a Gaussian process emulator given simulations or observations makes use of the conditional multivariate normality result stated in Lemma 2.4.9 [146]:

Lemma 2.4.9. *Let* $\mathbf{W}_1 \in \mathbb{R}^{n_1}$ *and* $\mathbf{W}_2 \in \mathbb{R}^{n_2}$ *be random vectors and define* $\mathbf{W} = \begin{pmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \end{pmatrix} \in \mathbb{R}^{n_1+n_2}$ *such that:*

$$\mathbf{W} = \begin{pmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \end{pmatrix} \sim \mathcal{N}_{n_1+n_2} \left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} \right) \tag{2.4.16}$$

where $\mu_1 \in \mathbb{R}^{n_1}$, $\mu_2 \in \mathbb{R}^{n_2}$, $\Sigma_{11} \in \mathbb{R}^{n_1 \times n_1}$, $\Sigma_{12} = \Sigma_{21}^{\mathrm{T}} \in \mathbb{R}^{n_1 \times n_2}$ *and* $\Sigma_{22} \in \mathbb{R}^{n_2 \times n_2}$. ***Conditional multivariate normality*** *[146] states that the distribution of* $\mathbf{W}_1$ *given* $\mathbf{W}_2$ *is:*

$$\mathbf{W}_1 \mid \mathbf{W}_2 \sim \mathcal{N}_{n_1} \left(\mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{W}_2 - \mu_2),\, \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} \right) \tag{2.4.17}$$

Notice how the expressions for the conditional mean and variance in Equation (2.4.17) are equivalent to the Bayes linear adjusted expectation and variance in Equations (2.2.3) and (2.2.4) respectively. Since a Gaussian process is fully characterised by its mean and variance, GP emulation is analogous to Bayes linear emulation with the additional assumption of the output following a normal distribution.

Bayesian Gaussian process emulation also requires the specification of a prior distribution over any hyperparameters which are integrated over to obtain the posterior distribution. To ensure computational tractability afforded by the use of Gaussian processes for the prior and likelihood of the variable(s) of interest, a convenient choice is to specify that the hyperparameters are also normally distributed. Note that this is a fairly strong assumption; often only limited knowledge is possessed about the hyperparameters. For the regression coefficients this may be encapsulated via an improper prior, $\mathbb{P}[\boldsymbol{\beta}] \propto 1$, which implies infinite prior variance in the mean function [107, 132]. Most prior choices of covariance function, such as the squared exponential, contain a variance hyperparameter, $\sigma_{u_i}^2$, for which $\mathbb{P}\left[\boldsymbol{\beta}, \sigma_{u_i}^2\right]$ may be assumed to be the conjugate normal-inverse gamma distribution [128, 131]. The weak improper form of this prior is $\mathbb{P}\left[\boldsymbol{\beta}, \sigma_{u_i}^2\right] \propto \sigma_{u_i}^{-2}$ [107, 132]. Elicitation for GP emulation is described in [126], although it is acknowledged that a full probabilistic specification is sometimes impractical and a compromise using the "plug-in" approach is more

appropriate [107].

In some instances, computer simulators may output derivatives with respect to some or all of their inputs. For statistical modelling using a Gaussian process emulator, this additional information can be incorporated to increase the emulator accuracy for the quantity or quantities of interest since the derivatives of a Gaussian process are also a Gaussian process [131]. Assuming the simulator is differentiable everywhere, then the derivatives may be modelled by a Gaussian process where the mean and covariance function are obtained from the relevant derivatives of the mean and covariance functions [125, 131, 162]. This can be particularly useful for calibration, optimisation and decision support tasks.

A limitation to the use of Gaussian process emulation is its scalability as the number of simulation data points, n, on which the Bayesian update is conditioned increases. Computational time grows at rate $\mathcal{O}(n^3)$, whilst memory costs are of the order of $\mathcal{O}(n^2)$. Further computational constraints arise for high-dimensional inputs with p parameters, which may result in more expensive evaluations of the mean and covariance functions. The Sherman-Morrison-Woodbury formula (see Lemma B.0.1 in Appendix B [118] for a statement and proof) can be used to reduce computational costs associated with matrix inversion and multiplication, but these issues remain when both n and p are relatively large.

Reviews of methodology for scalable Gaussian process inference are presented in [81, 115], with examples including: sparse GPs [159, 166]; local approximate GPs (laGP) [73], which uses a nearest-neighbour subset of the data to perform updates; and On-Site Surrogates (OSS) [90]. Local GP methods such as those described in [73] also have their disadvantages. They are based on a mixture of GPs and involve partitioning the covariance kernel, thus they ignore covariance between different blocks of the matrix. This can be an issue if there exists long range dependence. Another option for reducing computational costs is treed Gaussian processes [74] which dichotomise the input parameter space into hypercube regions using divisions which are parallel and perpendicular to the input axes. This leads to very flexible

models, however the averaging over the space of partitions can be slow and is not easily parallelisable and may result in an under-reporting of the uncertainty. A further method presented in [188] uses Importance Sampled Mixture of Experts (IS-MOE) for Gaussian processes in which the proposals are independent ensuring that the algorithm can be easily parallelised.

2.4.5 Bayes Linear Emulation

The Bayes linear paradigm was introduced in Section 2.2.2 and is applied to emulators of the form presented in Section 2.4.1 leading to Bayes linear emulation. Within a second order belief specification it is sufficient to consider how the emulator prior expectation and covariance shown in Equations (2.4.11) and (2.4.12) respectively, update to obtain the Bayes linear emulator adjusted expectation and variance. In this section, these quantities are derived via the Bayes linear adjustment formulae in Equations (2.2.3) to (2.2.5) for an emulator of the general form (including a nugget term) shown in Equation (2.4.1) evaluated at a new parameter vector, $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^D$. For notational clarity, we drop the subscript i and consider $f(\mathbf{x})$ to be a simulator with univariate output, with all emulator terms as defined in Section 2.4.1. Results for an emulator without a nugget term are obtained by setting $\sigma_w = 0$.

Prior Specification and Simulator Evaluations

The prior specification is as follows: $\mathbf{g}\colon \mathcal{X} \mapsto \mathbb{R}^p$, $\mathbb{E}[\boldsymbol{\beta}] = \boldsymbol{\mu}_\beta \in \mathbb{R}^p$; $\mathrm{Var}[\boldsymbol{\beta}] = \Sigma_\beta \in \mathbb{R}^{p\times p}$; $\mathbb{E}[u(\mathbf{x})] = 0$; $\mathrm{Cov}[u(\mathbf{x}), u(\mathbf{x}')] = \sigma_u^2 r(\mathbf{x}, \mathbf{x}')$, where σ_u^2 is the variance of the weakly stationary stochastic process, and $r(\cdot,\cdot)$ is a correlation function; and $\mathrm{Cov}[\boldsymbol{\beta}, u(\mathbf{x})] = \mathbf{0} \in \mathbb{R}^{p\times 1}$. In addition, define $r(0) = r(\mathbf{x}, \mathbf{x})$ noting that for many common choices of correlation function, such as the γ-exponential family, $r(0) = 1$. For the nugget term, the following prior assumptions are made: $\mathbb{E}[w(\mathbf{x})] = 0$; $\mathrm{Cov}[\boldsymbol{\beta}, w(\mathbf{x})] = \mathbf{0} \in \mathbb{R}^{p\times 1}$; $\mathrm{Cov}[u(\mathbf{x}), w(\mathbf{x})] = 0$; and $\mathrm{Cov}[w(\mathbf{x}), w(\mathbf{x}')] = \sigma_w^2 \mathbb{1}_{\{\mathbf{x}=\mathbf{x}'\}}$.

A collection of simulator evaluations are used to perform the Bayes linear update with design points $\mathbf{x}^{(i)}$, $i = 1, \ldots, n$, where $\mathbf{F}$ denotes the vector of simulation values

with $F_i = f(\mathbf{x}^{(i)})$. First introduce the notation $X \in \mathbb{R}^{n\times d}$ for the design matrix where the i^{th} row equals $\mathbf{x}^{(i)}$ and hence for the emulator let $\mathbf{r}(\mathbf{x}, X) \in \mathbb{R}^n$ be a column vector of correlations (computed using the correlation function $r(\cdot,\cdot)$) between $\mathbf{x}$ and each of the design points, where $\mathbf{r}(\mathbf{x}, X)_i = r(\mathbf{x}, \mathbf{x}^{(i)}),\ i = 1, \ldots, n$. In addition, let $G \in \mathbb{R}^{n\times p}$ be the design matrix with $G_{ij} = g_j(\mathbf{x}^{(i)})$, $\mathbf{u} \in \mathbb{R}^n$ be a vector of realisations of the weakly stationary stochastic process at the design inputs, and $\mathbf{w} \in \mathbb{R}^n$ be the vector of realisations of the nugget term. Equations (2.4.18) and (2.4.19) illustrate the decomposition of these simulations.

$$\mathbf{F} = G\boldsymbol{\beta} + \mathbf{u} + \mathbf{w} \tag{2.4.18}$$

$$\begin{pmatrix} f(\mathbf{x}^{(1)}) \\ \vdots \\ f(\mathbf{x}^{(n)}) \end{pmatrix} = \begin{pmatrix} g_1(\mathbf{x}^{(1)}) & \cdots & g_p(\mathbf{x}^{(1)}) \\ \vdots & \ddots & \vdots \\ g_1(\mathbf{x}^{(n)}) & \cdots & g_p(\mathbf{x}^{(n)}) \end{pmatrix} \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} u(\mathbf{x}^{(1)}) \\ \vdots \\ u(\mathbf{x}^{(n)}) \end{pmatrix} + \begin{pmatrix} w(\mathbf{x}^{(1)}) \\ \vdots \\ w(\mathbf{x}^{(n)}) \end{pmatrix} \tag{2.4.19}$$

Given the prior specification and the simulations, define $\text{Var}[\mathbf{u}] = \Sigma \in \mathbb{R}^{n\times n}$, where $\Sigma_{ij} = \sigma_u^2 r(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$, and $\text{Cov}[\boldsymbol{\beta}, \mathbf{u}] = \mathbf{0} \in \mathbb{R}^{p\times n}$. Moreover, $\text{Var}[\mathbf{w}] = \sigma_w^2 I_n$ where I_n is the $n \times n$ identity matrix, $\text{Cov}[\boldsymbol{\beta}, \mathbf{w}] = \mathbf{0} \in \mathbb{R}^{p\times n}$, and $\text{Cov}[\mathbf{u}, \mathbf{w}] = \mathbf{0} \in \mathbb{R}^{n\times n}$. These prior statements are combined to obtain $\mathbb{E}[\mathbf{F}]$, $\text{Var}[\mathbf{F}]$, $\text{Cov}[\boldsymbol{\beta}, \mathbf{F}]$, $\text{Cov}[u(\mathbf{x}), \mathbf{F}]$ and $\text{Cov}[w(\mathbf{x}), \mathbf{F}]$ in eqs. (2.4.20) to (2.4.24) respectively.

$$\begin{aligned} \mathbb{E}[\mathbf{F}] &= \mathbb{E}[G\boldsymbol{\beta} + \mathbf{u} + \mathbf{w}] \\ &= G\,\mathbb{E}[\boldsymbol{\beta}] + \mathbb{E}[\mathbf{u}] + \mathbb{E}[\mathbf{w}] \\ &= G\boldsymbol{\mu}_\beta \end{aligned} \tag{2.4.20}$$

$$\begin{aligned} \text{Var}[\mathbf{F}] &= \text{Var}[G\boldsymbol{\beta} + \mathbf{u} + \mathbf{w}] = \text{Cov}[G\boldsymbol{\beta} + \mathbf{u} + \mathbf{w}, G\boldsymbol{\beta} + \mathbf{u} + \mathbf{w}] \\ &= G\,\text{Cov}[\boldsymbol{\beta}, \boldsymbol{\beta}]G^{\mathrm{T}} + G\,\text{Cov}[\boldsymbol{\beta}, \mathbf{u}] + G\,\text{Cov}[\boldsymbol{\beta}, \mathbf{w}] \\ &\quad + \text{Cov}[\mathbf{u}, \boldsymbol{\beta}]G^{\mathrm{T}} + \text{Cov}[\mathbf{u}, \mathbf{u}] + \text{Cov}[\mathbf{u}, \mathbf{w}] \\ &\quad + \text{Cov}[\mathbf{w}, \boldsymbol{\beta}]G^{\mathrm{T}} + \text{Cov}[\mathbf{w}, \mathbf{u}] + \text{Cov}[\mathbf{w}, \mathbf{w}] \\ &= G\,\text{Var}[\boldsymbol{\beta}]G^{\mathrm{T}} + \text{Var}[\mathbf{u}] + \text{Var}[\mathbf{w}] \\ &= G\Sigma_\beta G^{\mathrm{T}} + \Sigma + \sigma_w^2 I_n \end{aligned} \tag{2.4.21}$$

$$\begin{aligned}
\mathrm{Cov}[\boldsymbol{\beta}, \mathbf{F}] &= \mathrm{Cov}[\boldsymbol{\beta}, G\boldsymbol{\beta} + \mathbf{u} + \mathbf{w}] \\
&= \mathrm{Cov}[\boldsymbol{\beta}, \boldsymbol{\beta}]G^{\mathrm{T}} + \mathrm{Cov}[\boldsymbol{\beta}, \mathbf{u}] + \mathrm{Cov}[\boldsymbol{\beta}, \mathbf{w}] \\
&= \Sigma_{\beta} G^{\mathrm{T}} \qquad (2.4.22) \\
\mathrm{Cov}[u(\mathbf{x}), \mathbf{F}] &= \mathrm{Cov}[u(\mathbf{x}), G\boldsymbol{\beta} + \mathbf{u} + \mathbf{w}] \\
&= \mathrm{Cov}[u(\mathbf{x}), \boldsymbol{\beta}]G^{\mathrm{T}} + \mathrm{Cov}[u(\mathbf{x}), \mathbf{u}] + \mathrm{Cov}[u(\mathbf{x}), \mathbf{w}] \\
&= \sigma_u^2 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X) \qquad (2.4.23) \\
\mathrm{Cov}[w(\mathbf{x}), \mathbf{F}] &= \mathrm{Cov}[w(\mathbf{x}), G\boldsymbol{\beta} + \mathbf{u} + \mathbf{w}] \\
&= \mathrm{Cov}[w(\mathbf{x}), \boldsymbol{\beta}]G^{\mathrm{T}} + \mathrm{Cov}[w(\mathbf{x}), \mathbf{u}] + \mathrm{Cov}[w(\mathbf{x}), \mathbf{w}] \\
&= \sigma_w^2 \left(\mathbb{1}_{\{\mathbf{x}=\mathbf{x}^{(1)}\}}, \ldots, \mathbb{1}_{\{\mathbf{x}=\mathbf{x}^{(n)}\}} \right) \qquad (2.4.24)
\end{aligned}$$

Assuming that $\mathbf{x} \notin \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\}$ yields $\mathrm{Cov}[w(\mathbf{x}), \mathbf{F}] = \mathbf{0} \in \mathbb{R}^{1 \times n}$ in Equation (2.4.24) which leads to important implications for the derivations since it asserts that there is no correlation between the nugget term and the simulations.

We proceed to obtain the Bayes linear emulator adjustment formulae by first deriving $\mathbb{E}_F[\boldsymbol{\beta}]$, $\mathrm{Var}_F[\boldsymbol{\beta}]$, $\mathbb{E}_F[u(\mathbf{x})]$, $\mathrm{Var}_F[u(\mathbf{x})]$, $\mathbb{E}_F[w(\mathbf{x})]$, $\mathrm{Var}_F[w(\mathbf{x})]$, $\mathrm{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})]$, $\mathrm{Cov}_F[\boldsymbol{\beta}, w(\mathbf{x})]$ and $\mathrm{Cov}_F[u(\mathbf{x}), w(\mathbf{x})]$. Throughout these derivations the Sherman-Morrison-Woodbury formula and a related expression in Equations (B.0.1) and (B.0.3) respectively are used for manipulating matrix inverses. For a formal statement and proof, see Lemma B.0.1 and Corollary B.0.3 respectively in Appendix B [118]. In order to make use of these identities, the variance matrices Σ and $\sigma_w^2 I_n$ are frequently grouped together as:

$$\Omega = \Sigma + \sigma_w^2 I_n \qquad (2.4.25)$$

It is assumed that Ω is invertible. Moreover, at each stage the limit as $\mathrm{Var}[\boldsymbol{\beta}] \to \infty$ is considered. Formally, this is the limit as the eigenvalues of Σ_{β} tend to ∞, or equivalently the limit as the eigenvalues of Σ_{β}^{-1} tend to 0. This corresponds to infinite prior uncertainty (equivalently no prior knowledge) of $\boldsymbol{\beta}$.

Derivation of $\mathbb{E}_F[\boldsymbol{\beta}]$

The adjusted expectation of $\boldsymbol{\beta}$ is shown in Equation (2.4.26).

$$\begin{aligned}\mathbb{E}_F[\boldsymbol{\beta}] &= \mathbb{E}[\boldsymbol{\beta}] + \mathrm{Cov}[\boldsymbol{\beta}, \mathbf{F}]\,\mathrm{Var}[\mathbf{F}]^{-1}(\mathbf{F} - \mathbb{E}[\mathbf{F}]) \\ &= \boldsymbol{\mu}_\beta + \Sigma_\beta G^{\mathrm{T}}(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}(\mathbf{F} - G\boldsymbol{\mu}_\beta)\end{aligned}$$

Using Equation (2.4.25) and applying Equation (B.0.3).

$$\begin{aligned}&= \boldsymbol{\mu}_\beta + (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}G^{\mathrm{T}}\Omega^{-1}(\mathbf{F} - G\boldsymbol{\mu}_\beta) \\ &= (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}\{(G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})\boldsymbol{\mu}_\beta + G^{\mathrm{T}}\Omega^{-1}(\mathbf{F} - G\boldsymbol{\mu}_\beta)\} \\ &= (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}(\Sigma_\beta^{-1}\boldsymbol{\mu}_\beta + G^{\mathrm{T}}\Omega^{-1}\mathbf{F}) \qquad (2.4.26)\end{aligned}$$

It is observed that $\mathbb{E}_F[\boldsymbol{\beta}]$ may be written in terms of the Generalised Least Squares (GLS) estimate for $\boldsymbol{\beta}$ in a linear regression model where $u(\mathbf{x}) + w(\mathbf{x})$ is treated as a single residual term with variance matrix Ω. Under this setup $\hat{\boldsymbol{\beta}}_{GLS} = (G^{\mathrm{T}}\Omega^{-1}G)^{-1}G^{\mathrm{T}}\Omega^{-1}\mathbf{F}$, with $\mathbb{E}_F[\boldsymbol{\beta}]$ re-expressed in Equation (2.4.27).

$$\mathbb{E}_F[\boldsymbol{\beta}] = (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}(\Sigma_\beta^{-1}\boldsymbol{\mu}_\beta + G^{\mathrm{T}}\Omega^{-1}G\hat{\boldsymbol{\beta}}_{GLS}) \qquad (2.4.27)$$

The second part is a weighted sum of the prior expectation and the GLS estimate with weights equal to the prior precision[4] of $\boldsymbol{\beta}$ and the precision of the GLS estimate respectively, hence each term is standardised. It will subsequently be shown that the first part is equal to $\mathrm{Var}_F[\boldsymbol{\beta}]$ (see Equation (2.4.29)) which scales the weighted sum. Note that this setup for GLS regression recurs throughout these derivations. A further remark is that assuming a squared exponential covariance function as in Equation (2.4.3) with correlation lengths that are not too long and well spaced simulation design points, then $\Sigma \approx \sigma_u^2 I_n$. The result is that $\hat{\boldsymbol{\beta}}_{GLS} \approx \hat{\boldsymbol{\beta}}_{OLS}$, where $\hat{\boldsymbol{\beta}}_{OLS}$ are the Ordinary Least Squares (OLS) regression estimates for $\boldsymbol{\beta}$.

The limit as $\mathrm{Var}[\boldsymbol{\beta}] \to \infty$ for $\mathbb{E}_F[\boldsymbol{\beta}]$ is given in Equation (2.4.28) and tends towards the classical limit of $\hat{\boldsymbol{\beta}}_{GLS}$. It is noted that $\boldsymbol{\mu}_\beta$ is not present in this formula

[4]The precision (matrix) is defined as the inverse of the variance (matrix).

and hence its specification is unnecessary if there is complete uncertainty regarding the variability of $\boldsymbol{\beta}$.

$$\lim_{\mathrm{Var}[\boldsymbol{\beta}]\to\infty} \mathbb{E}_F[\boldsymbol{\beta}] = \hat{\boldsymbol{\beta}}_{GLS} = (G^{\mathrm{T}}\Omega^{-1}G)^{-1}G^{\mathrm{T}}\Omega^{-1}\mathbf{F} \tag{2.4.28}$$

Derivation of $\mathrm{Var}_F[\boldsymbol{\beta}]$

Derivation of the adjusted variance of $\boldsymbol{\beta}$ is given in Equation (2.4.29).

$$\begin{aligned}\mathrm{Var}_F[\boldsymbol{\beta}] &= \mathrm{Var}[\boldsymbol{\beta}] - \mathrm{Cov}[\boldsymbol{\beta},\mathbf{F}]\,\mathrm{Var}[\mathbf{F}]^{-1}\,\mathrm{Cov}[\mathbf{F},\boldsymbol{\beta}] \\ &= \Sigma_\beta - \Sigma_\beta G^{\mathrm{T}}(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}G\Sigma_\beta\end{aligned}$$

Applying Corollary B.0.3.

$$\begin{aligned}&= \Sigma_\beta - (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}G^{\mathrm{T}}\Sigma^{-1}G\Sigma_\beta \\ &= \{I_p - (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}G^{\mathrm{T}}\Sigma^{-1}G\}\Sigma_\beta \\ &= (G^{\mathrm{T}}\Sigma^{-1}G + \Sigma_\beta^{-1})^{-1}\{(G^{\mathrm{T}}\Sigma^{-1}G + \Sigma_\beta^{-1}) - G^{\mathrm{T}}\Sigma^{-1}G\}\Sigma_\beta \\ &= (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}\end{aligned} \tag{2.4.29}$$

The adjusted precision of $\boldsymbol{\beta}$ is the inverse of Equation (2.4.29) which is observed to be the sum of the precision matrix for $\hat{\boldsymbol{\beta}}_{GLS}$ and the prior precision matrix Σ_β^{-1}. Each term linearly contributes to the adjusted precision of $\boldsymbol{\beta}$ with no interaction between the two. This leads to the interpretation that for large prior uncertainty, the variance of the $\hat{\boldsymbol{\beta}}_{GLS}$ is the dominant term, whilst for very small prior uncertainty, it is expected that Σ_β^{-1} will dominate. This is confirmed by considering the limit as $\mathrm{Var}[\boldsymbol{\beta}] \to \infty$ in Equation (2.4.30) where it is established that $\mathrm{Var}_F[\boldsymbol{\beta}]$ tends towards $\mathrm{Var}\left[\hat{\boldsymbol{\beta}}_{GLS}\right]$ under the above GLS regression setup.

$$\lim_{\mathrm{Var}[\boldsymbol{\beta}]\to\infty} \mathrm{Var}_F[\boldsymbol{\beta}] = \mathrm{Var}\left[\hat{\boldsymbol{\beta}}_{GLS}\right] = (G^{\mathrm{T}}\Omega^{-1}G)^{-1} \tag{2.4.30}$$

Derivation of $\mathbb{E}_F[u(\mathbf{x})]$

$\mathbb{E}_F[u(\mathbf{x})]$ is derived as follows, using the Bayes linear adjustment formula in Equation (2.2.3):

$$\begin{aligned}\mathbb{E}_F[u(\mathbf{x})] &= \mathbb{E}[u(\mathbf{x})] + \mathrm{Cov}[u(\mathbf{x}), \mathbf{F}]\,\mathrm{Var}[\mathbf{F}]^{-1}(\mathbf{F} - \mathbb{E}[\mathbf{F}]) \\ &= \sigma_u^2 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}(\mathbf{F} - G\boldsymbol{\mu}_\beta) \\ &= \sigma_u^2 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\{(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}\mathbf{F} - (G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}G\boldsymbol{\mu}_\beta\}\end{aligned}$$

The term $(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}G\boldsymbol{\mu}_\beta$ may be manipulated as follows to express it in terms of $\mathbb{E}_F[\boldsymbol{\beta}]$ via a double application of the matrix inverse identity in Equation (B.0.3).

$$\begin{aligned}(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}G\boldsymbol{\mu}_\beta &= (G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}G\Sigma_\beta\Sigma_\beta^{-1}\boldsymbol{\mu}_\beta \\ &= \Omega^{-1}G(G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}\Sigma_\beta^{-1}\boldsymbol{\mu}_\beta \\ &= \Omega^{-1}G(G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}\{(\Sigma_\beta^{-1}\boldsymbol{\mu}_\beta + G^{\mathrm{T}}\Omega^{-1}\mathbf{F}) - G^{\mathrm{T}}\Omega^{-1}\mathbf{F}\} \\ &= \Omega^{-1}G\{\mathbb{E}_F[\boldsymbol{\beta}] - (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}G^{\mathrm{T}}\Omega^{-1}\mathbf{F}\} \\ &= \Omega^{-1}G\{\mathbb{E}_F[\boldsymbol{\beta}] - \Sigma_\beta G^{\mathrm{T}}(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}\mathbf{F}\}\end{aligned}$$

Substituting into the equation for $\mathbb{E}_F[u(\mathbf{x})]$ yields the following simplification.

$$\begin{aligned}\mathbb{E}_F[u(\mathbf{x})] &= \sigma_u^2 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\{(I_n + \Omega^{-1}G\Sigma_\beta G^{\mathrm{T}})(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}\mathbf{F} - \Omega^{-1}G\,\mathbb{E}_F[\boldsymbol{\beta}]\} \\ &= \sigma_u^2 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\{(G\Sigma_\beta G^{\mathrm{T}} + \Omega)(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}\mathbf{F} - G\,\mathbb{E}_F[\boldsymbol{\beta}]\} \\ &= \sigma_u^2 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\{\mathbf{F} - G\,\mathbb{E}_F[\boldsymbol{\beta}]\} \qquad (2.4.31)\end{aligned}$$

The form presented in Equation (2.4.31) may be interpreted as a weighted sum of the adjusted standardised residuals for the regression component of the emulator over the simulation design points. The weights, stored in the row vector $\sigma_u^2 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)$, are computed based on the strength of correlation between $\mathbf{x}$ and the design points, scaled by σ_u^2.

In the limit as $\mathrm{Var}[\boldsymbol{\beta}] \to \infty$, $\mathbb{E}_F[\boldsymbol{\beta}] \to \hat{\boldsymbol{\beta}}_{GLS}$ (see Equation (2.4.28)). This yields

the following result:

$$\lim_{\mathrm{Var}[\boldsymbol{\beta}]\to\infty} \mathbb{E}_F[u(\mathbf{x})] = \sigma_u^2 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\{\mathbf{F} - G\hat{\boldsymbol{\beta}}_{GLS}\} \tag{2.4.32}$$

Derivation of $\mathrm{Var}_F[u(\mathbf{x})]$

$\mathrm{Var}_F[u(\mathbf{x})]$ is derived as follows:

$$\begin{aligned}\mathrm{Var}_F[u(\mathbf{x})] &= \mathrm{Var}[u(\mathbf{x})] - \mathrm{Cov}[u(\mathbf{x}), \mathbf{F}]\,\mathrm{Var}[\mathbf{F}]^{-1}\,\mathrm{Cov}[\mathbf{F}, u(\mathbf{x})] \\ &= \sigma_u^2 r(0) - \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}\mathbf{r}(\mathbf{x}, X)\end{aligned}$$

Applying Equation (B.0.1).

$$\begin{aligned}&= \sigma_u^2 r(0) - \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\{\Omega^{-1} - \Omega^{-1}G(G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}G^{\mathrm{T}}\Omega^{-1}\}\mathbf{r}(\mathbf{x}, X) \\ &= \sigma_u^2 r(0) - \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\mathbf{r}(\mathbf{x}, X) \\ &\quad + \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}G(G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X) \\ &= \sigma_u^2 r(0) - \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\mathbf{r}(\mathbf{x}, X) \\ &\quad + \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}G\,\mathrm{Var}_F[\boldsymbol{\beta}]G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X)\end{aligned} \tag{2.4.33}$$

Inspection shows that the first two terms (first line of Equation (2.4.33)) represent the direct contribution from the combined residual process, $u(\cdot) + w(\cdot)$. The first term, $\sigma_u^2 r(0)$ is the prior variance for the weakly stationary stochastic process, whilst the second term which is subtracted is a quantification of the information gained as a consequence of the collection of simulator evaluations via the imposed correlation structure. Notice that the third term (second line of Equation (2.4.33)) incorporates the interaction between the regression part (via the deterministic function evaluation matrix G) and the residual stochastic process part of the emulator structure, with a standardisation by $\mathrm{Var}_F[\boldsymbol{\beta}]$.

The limit as $\mathrm{Var}[\boldsymbol{\beta}] \to \infty$ is again considered. Note that only the third term in Equation (2.4.33) contains Σ_β, whilst the first two terms are invariant to Σ_β. Using

$\lim_{\text{Var}[\boldsymbol{\beta}]\to\infty} \text{Var}_F[\boldsymbol{\beta}] = \text{Var}\left[\hat{\boldsymbol{\beta}}_{GLS}\right]$, the limiting result is as follows:

$$\lim_{\text{Var}[\boldsymbol{\beta}]\to\infty} \text{Var}_F[u(\mathbf{x})] = \sigma_u^2 r(0) - \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\mathbf{r}(\mathbf{x}, X) + \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1} G \,\text{Var}\left[\hat{\boldsymbol{\beta}}_{GLS}\right] G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X) \tag{2.4.34}$$

Derivation of $\text{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})]$

The expression for $\text{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})]$ is derived in Equation (2.4.35). Note that the prior assumption of $\text{Cov}[\boldsymbol{\beta}, \mathbf{u}] = \mathbf{0}$ is used.

$$\begin{aligned}\text{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})] &= \text{Cov}[\boldsymbol{\beta}, u(\mathbf{x})] - \text{Cov}[\boldsymbol{\beta}, \mathbf{F}]\,\text{Var}[\mathbf{F}]^{-1}\,\text{Cov}[\mathbf{F}, u(\mathbf{x})] \\ &= -\sigma_u^2 \Sigma_\beta G^{\mathrm{T}}(G\Sigma_\beta G^{\mathrm{T}} + \Omega)^{-1}\mathbf{r}(\mathbf{x}, X)\end{aligned}$$

Applying Equation (B.0.3).

$$= -\sigma_u^2 (G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1} G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X) \tag{2.4.35}$$

Interpreting the form of $\text{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})]$, the first part, $(G^{\mathrm{T}}\Omega^{-1}G + \Sigma_\beta^{-1})^{-1}G^{\mathrm{T}} = \text{Var}_F[\boldsymbol{\beta}]G^{\mathrm{T}}$ and represents the contribution of the $\boldsymbol{\beta}$-coefficients via a scaling of the design matrix for the simulation inputs by $\text{Var}_F[\boldsymbol{\beta}]$. Similarly, the second term $\sigma_u^2\Omega^{-1}\mathbf{r}(\mathbf{x}, X)$, depicts the correlation between $\mathbf{x}$ and the inputs at which the simulator is evaluated, scaled by σ_u^2 to obtain the covariance. This is standardised by the prior residual variance of the simulations.

The limit as $\text{Var}[\boldsymbol{\beta}] \to \infty$ of $\text{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})]$ is obtained using that $\text{Var}_F[\boldsymbol{\beta}] \to \text{Var}\left[\hat{\boldsymbol{\beta}}_{GLS}\right] = (G^{\mathrm{T}}\Omega^{-1}G)^{-1}$ as $\text{Var}[\boldsymbol{\beta}] \to \infty$ (Equation (2.4.30)), yielding the following expression:

$$\lim_{\text{Var}[\boldsymbol{\beta}]\to\infty} \text{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})] = -\sigma_u^2\,\text{Var}\left[\hat{\boldsymbol{\beta}}_{GLS}\right] G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X) \tag{2.4.36}$$

Derivation of $\mathbb{E}_F[w(\mathbf{x})]$**,** $\text{Var}_F[w(\mathbf{x})]$**,** $\text{Cov}_F[\boldsymbol{\beta}, w(\mathbf{x})]$ **and** $\text{Cov}_F[u(\mathbf{x}), w(\mathbf{x})]$

Several interesting results regarding the adjusted quantities for $w(\mathbf{x})$ are presented which occur as a result of the above stated prior specification. $\mathbb{E}_F[w(\mathbf{x})]$, $\text{Var}_F[w(\mathbf{x})]$, $\text{Cov}_F[\boldsymbol{\beta}, w(\mathbf{x})]$ and $\text{Cov}_F[u(\mathbf{x}), w(\mathbf{x})]$ are given in Equations (2.4.37) to (2.4.40) re-

spectively. Comparing with the prior assumptions, there has been no change between the prior and updated quantities. This invariance is a key property of the emulator nugget term and characterises how its purpose is to account for the remaining variability not captured by either the global regression term or the stochastic process.

$$\begin{aligned}\mathbb{E}_F[w(\mathbf{x})] &= \mathbb{E}[w(\mathbf{x})] + \mathrm{Cov}[w(\mathbf{x}), \mathbf{F}]\,\mathrm{Var}[\mathbf{F}]^{-1}(\mathbf{F} - \mathbb{E}[\mathbf{F}]) \\ &= 0 \qquad (2.4.37)\\ \mathrm{Var}_F[w(\mathbf{x})] &= \mathrm{Var}[w(\mathbf{x})] - \mathrm{Cov}[w(\mathbf{x}), \mathbf{F}]\,\mathrm{Var}[\mathbf{F}]^{-1}\,\mathrm{Cov}[\mathbf{F}, w(\mathbf{x})] \\ &= \sigma_w^2 \qquad (2.4.38)\\ \mathrm{Cov}_F[\boldsymbol{\beta}, w(\mathbf{x})] &= \mathrm{Cov}[\boldsymbol{\beta}, w(\mathbf{x})] - \mathrm{Cov}[\boldsymbol{\beta}, \mathbf{F}]\,\mathrm{Var}[\mathbf{F}]^{-1}\,\mathrm{Cov}[\mathbf{F}, w(\mathbf{x})] \\ &= \mathbf{0} \qquad (2.4.39)\\ \mathrm{Cov}_F[u(\mathbf{x}), w(\mathbf{x})] &= \mathrm{Cov}[u(\mathbf{x}), w(\mathbf{x})] - \mathrm{Cov}[u(\mathbf{x}), \mathbf{F}]\,\mathrm{Var}[\mathbf{F}]^{-1}\,\mathrm{Cov}[\mathbf{F}, w(\mathbf{x})] \\ &= 0 \qquad (2.4.40)\end{aligned}$$

Bayes Linear Adjustment Formulae for a Bayesian Emulator

The above derived quantities are combined to obtain the emulator adjusted expectation and variance in Equations (2.4.41) and (2.4.42) respectively, whilst a factorised form of the adjusted variance is given in Equation (2.4.43). In all of these equations, the term $\mathbf{g}(\mathbf{x}) - \sigma_u^2 G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X)$ occurs which is the difference between the deterministic regression functions evaluated at $\mathbf{x}$ and the adjusted effect from the residuals. Also note that there are no interaction terms between $w(\mathbf{x})$ and either $\boldsymbol{\beta}$ or $u(\mathbf{x})$ which is as a consequence of the nugget term prior specification and fulfils the aim of accounting for the remaining variability beyond the regression and stochastic process parts.

$$\begin{aligned}\mathbb{E}_F[f(\mathbf{x})] &= \mathbb{E}_F[\mathbf{g}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\beta} + u(\mathbf{x}) + w(\mathbf{x})] \\ &= \mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathbb{E}_F[\boldsymbol{\beta}] + \mathbb{E}_F[u(\mathbf{x})] + \mathbb{E}_F[w(\mathbf{x})] \\ &= \mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathbb{E}_F[\boldsymbol{\beta}] + \sigma_u^2\mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}(\mathbf{F} - G\,\mathbb{E}_F[\boldsymbol{\beta}]) \\ &= (\mathbf{g}(\mathbf{x}) - \sigma_u^2 G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X))^{\mathrm{T}}\,\mathbb{E}_F[\boldsymbol{\beta}] + \sigma_u^2\mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\mathbf{F} \qquad (2.4.41)\end{aligned}$$

$$\begin{aligned}\mathrm{Var}_F[f(\mathbf{x})] &= \mathrm{Var}_F[\mathbf{g}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\beta} + u(\mathbf{x}) + w(\mathbf{x})] \\ &= \mathrm{Cov}_F[\mathbf{g}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\beta} + u(\mathbf{x}) + w(\mathbf{x}), \mathbf{g}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\beta} + u(\mathbf{x}) + w(\mathbf{x})] \\ &= \mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathrm{Var}_F[\boldsymbol{\beta}]\mathbf{g}(\mathbf{x}) + \mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathrm{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})] + \mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathrm{Cov}_F[\boldsymbol{\beta}, w(\mathbf{x})] \\ &\quad + \mathrm{Cov}_F[u(\mathbf{x}), \boldsymbol{\beta}]\mathbf{g}(\mathbf{x}) + \mathrm{Var}_F[u(\mathbf{x})] + \mathrm{Cov}_F[u(\mathbf{x}), w(\mathbf{x})] \\ &\quad + \mathrm{Cov}_F[w(\mathbf{x}), \boldsymbol{\beta}]\mathbf{g}(\mathbf{x}) + \mathrm{Cov}_F[w(\mathbf{x}), u(\mathbf{x})] + \mathrm{Var}_F[w(\mathbf{x})]\end{aligned}$$

Simplify using that $\mathrm{Cov}_F[u(\mathbf{x}), \boldsymbol{\beta}]\mathbf{g}(\mathbf{x}) = \mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathrm{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})]$ and the above equations for terms containing $w(\mathbf{x})$.

$$\begin{aligned}&= \mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathrm{Var}_F[\boldsymbol{\beta}]\mathbf{g}(\mathbf{x}) - 2\mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathrm{Cov}_F[\boldsymbol{\beta}, u(\mathbf{x})] \\ &\quad + \mathrm{Var}_F[u(\mathbf{x})] + \mathrm{Var}_F[w(\mathbf{x})] \\ &= \mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathrm{Var}_F[\boldsymbol{\beta}]\mathbf{g}(\mathbf{x}) - 2\sigma_u^2\mathbf{g}(\mathbf{x})^{\mathrm{T}}\,\mathrm{Var}_F[\boldsymbol{\beta}]G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X) \\ &\quad + \sigma_u^2 r(0) - \sigma_u^4\mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\mathbf{r}(\mathbf{x}, X) \\ &\quad + \sigma_u^4\mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}G\,\mathrm{Var}_F[\boldsymbol{\beta}]G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X) + \sigma_w^2 \qquad (2.4.42)\end{aligned}$$

This may be factorised as follows.

$$\begin{aligned}&= (\mathbf{g}(\mathbf{x}) - \sigma_u^2 G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X))^{\mathrm{T}}\,\mathrm{Var}_F[\boldsymbol{\beta}](\mathbf{g}(\mathbf{x}) - \sigma_u^2 G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X)) \\ &\quad + \sigma_u^2 r(0) - \sigma_u^4\mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\mathbf{r}(\mathbf{x}, X) + \sigma_w^2 \qquad (2.4.43)\end{aligned}$$

Limiting Results for a Bayes Linear Emulator

The limit of $\mathbb{E}_F[f(\mathbf{x})]$ as $\mathrm{Var}[\boldsymbol{\beta}] \to \infty$ is given in Equation (2.4.44). Note that $\hat{\boldsymbol{\beta}}_{GLS} = (G^{\mathrm{T}}\Omega^{-1}G)^{-1}G^{\mathrm{T}}\Omega^{-1}\mathbf{F}$ where the variance matrix used is Ω. Consequently the limiting case for $\mathbb{E}_F[f(\mathbf{x})]$ may be viewed as a linear combination of the scaled $\hat{\boldsymbol{\beta}}_{GLS}$ estimate and simulation residual variability.

$$\lim_{\mathrm{Var}[\boldsymbol{\beta}]\to\infty} \mathbb{E}_F[f(\mathbf{x})] = (\mathbf{g}(\mathbf{x}) - \sigma_u^2 G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X))^{\mathrm{T}}\hat{\boldsymbol{\beta}}_{GLS} + \sigma_u^2\mathbf{r}^{\mathrm{T}}(\mathbf{x}, X)\Omega^{-1}\mathbf{F} \quad (2.4.44)$$

The limit of $\mathrm{Var}_F[f(\mathbf{x})]$ as $\mathrm{Var}[\boldsymbol{\beta}] \to \infty$ is as follows, where $\mathrm{Var}\left[\hat{\boldsymbol{\beta}}_{GLS}\right] = (G^{\mathrm{T}}\Omega^{-1}G)^{-1}$.

$$\lim_{\mathrm{Var}[\boldsymbol{\beta}]\to\infty} \mathrm{Var}_F[f(\mathbf{x})] = (\mathbf{g}(\mathbf{x}) - \sigma_u^2 G^{\mathrm{T}}\Omega^{-1}\mathbf{r}(\mathbf{x}, X))^{\mathrm{T}}\,\mathrm{Var}\left[\hat{\boldsymbol{\beta}}_{GLS}\right]$$

$$\cdot (\mathbf{g}(\mathbf{x}) - \sigma_u^2 G^{\mathrm{T}} \Omega^{-1} \mathbf{r}(\mathbf{x}, X))$$
$$+ \sigma_u^2 r(0) - \sigma_u^4 \mathbf{r}^{\mathrm{T}}(\mathbf{x}, X) \Omega^{-1} \mathbf{r}(\mathbf{x}, X) + \sigma_w^2 \qquad (2.4.45)$$

A similar interpretation is that the limiting case for $\mathrm{Var}_F[f(\mathbf{x})]$ is a linear combination of the variance due to the scaled $\hat{\boldsymbol{\beta}}_{GLS}$, and the variability due to the weakly stationary stochastic process and nugget terms, which is partially resolved by the simulations.

2.4.6 Emulator Diagnostics

Well specified Bayesian emulators provide an effective means of statistically approximating complex computer models. However, if an emulator is ill specified, it may yield poor predictions in part or all of the input parameter space leading to potentially problematic consequences within analyses. Issues can arise within the emulator structure including: badly chosen deterministic functions and active input parameters; an invalid assumption of (weak) stationarity for the emulator stochastic process term; as well as inappropriate specification of the prior mean and covariance structure including hyperparameter estimation or prior distribution. For a full Bayesian emulator, further issues can arise due to the underlying distributional assumptions. Even in situations where the specification is reasonable, problems can occur in high-dimensional input parameter spaces with limited computational resources leading to sparse simulation designs and hence limited training data on which to update the emulator. It is therefore imperative to perform diagnostics in order to verify that the emulator is well fitting and that inferences are trustworthy. An emulator diagnostic technique is formally defined in Definition 2.4.10 [8]. Note that all diagnostics are conditional on the prior specification and the simulation data used to fit the emulator.

Definition 2.4.10. Emulator diagnostics consist of a comparison between the output of an emulator, and the simulated response(s) of the scientific model using a test data set, also known as validation data, in order to verify the underlying

assumptions and fit of the emulator. A **validation data set** consists of samples from the model which were not used to construct the emulator.

Diagnostics require a collection of simulation data not used in the fitting of the emulator and should be chosen to provide coverage of the entire input parameter space region of interest. These can be obtained in several ways. The simplest approach is to split the simulations into two parts with designs: $X = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{R}^{n \times D}$, as introduced in Section 2.4.5 and used in the fitting of the emulator; and new input design $X^* = (\mathbf{x}_1^*, \ldots, \mathbf{x}_m^*) \in \mathbb{R}^{m \times D}$ producing validation data $f(X^*) = (f(\mathbf{x}_1^*), \ldots, f(\mathbf{x}_m^*)) \in \mathbb{R}^m$ with which to compare to the emulator output. This has the main advantage of generating two independent sets for fitting and diagnostics. However, it does not optimally use the available computational resources to construct a more accurate emulator; a limitation which is most acutely felt when only limited simulations can be performed [8]. An alternative approach is leave-one-out diagnostics in which predictions are made for each of the n training points in X based on an emulator fitted using the remaining $n-1$ points. Leave-one-out diagnostics can be used to assess the effect of inclusion of each simulation training point used to fit the emulator. In addition, this alleviates the aforementioned concerns on the number of simulations, although comes at the cost of refitting the emulator with each of the n data points removed, which in turn can potentially be computationally expensive, but is usually many orders of magnitude smaller than performing extra simulations [98, 151]. Leave-k-out diagnostics extends this idea further to all simulation subsets of size k, however this requires refitting the emulator $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ times and so is often infeasibly expensive. A related method is k-fold cross-validation, also known as multifold cross-validation, in which the simulation data is randomly partitioned into k (approximately) equal size subsamples with emulator predictions made given the data in all other folds. This greatly reduces the computational costs versus full leave-k-out diagnostics [98].

Before commencing emulation it is useful to examine intermediate steps in the construction. One important aspect is to assess the mean function by inspecting

the corresponding linear regression model predictions, $\hat{f}_{\mathrm{LM}}(\mathbf{x})$. The standardised residuals are defined in Equation (2.4.46), where $\hat{\sigma}_{\mathrm{LM}}$ is the estimated linear model residual standard error.

$$\hat{\varepsilon}_{\mathrm{LM}}(\mathbf{x}) = \frac{f(\mathbf{x}) - \hat{f}_{\mathrm{LM}}(\mathbf{x})}{\hat{\sigma}_{\mathrm{LM}}} \tag{2.4.46}$$

For a collection of simulations, plots of standardised linear model residuals versus the simulator output, each of the inputs and any transformed inputs within the linear model, can be used to identify any patterns. If these exist, they indicate that the linear model, and hence the mean function, do not adequately capture all of the available information. As with standard checks for regression models, too many large standardised residuals; large is usually classified as greater than 2 or 3, implies a poor fitting model. The adjusted R^2 value also provides an indication of the goodness-of-fit with values close to one preferred.

After fitting an emulator, a range of diagnostic techniques may be applied [8]. Let $f^*(\cdot)$ denote the emulator representation for a simulator output $f(\cdot)$ and define a general emulator diagnostic function $D(\cdot)$ to be a function of the validation data from which $D(f(\mathbf{x}^*))$ and $D(f^*(\mathbf{x}^*))$ can be compared based on the distribution of $D(f^*(\mathbf{x}^*))$. Note that these diagnostic functions can also be evaluated within the leave-one-out framework. If there are few failures of the diagnostic tests, then this is indicative of a well specified emulator.

For the univariate simulator output, $f_i(\mathbf{x})$, first introduce the individual standardised prediction errors in Equation (2.4.47) [8].

$$D_{ij}(f_i(\mathbf{x}_j^*)) = \frac{f_i(\mathbf{x}_j^*) - \mathbb{E}_F[f_i^*(\mathbf{x}_j^*)]}{\sqrt{\mathrm{Var}_F[f_i^*(\mathbf{x}_j^*)]}} \tag{2.4.47}$$

Assessment depends on the underlying emulator assumptions. For Gaussian process emulation, these follow a t-distribution with $(n-p)$ degrees of freedom conditional on the training data and the estimated parameters, with asymptotic normality when the residual degrees of freedom is large. An alternative and more robust method used in the Bayes linear approach employs Pukelsheim's 3-sigma rule [142]

(see Section 2.8.3 for a statement and discussion) to assess which validation data points satisfy $|D_{ij}(f_i(\mathbf{x}_j^*))| \leq 3$. An equivalent technique is to test whether $f_i(\mathbf{x}_j^*)$ is contained in the credible interval: $\mathbb{E}_F[f_i^*(\mathbf{x}_j^*)] \pm 3\,\mathrm{Var}_F[f_i^*(\mathbf{x}_j^*)]$.

Standardised prediction errors are useful for assessing individual predictions, but do not account for correlation across multiple input settings. A generalisation is the diagnostic statistic presented in Equation (2.4.48) [8] combining the individual prediction errors whilst standardising by their joint covariance structure. Note that this is a Mahalanobis distance.

$$D_{\mathrm{MD}}(f(X^*) = (f(X^*) - \mathbb{E}_F[f^*(X^*)])^{\mathrm{T}}\,\mathrm{Var}_F[f^*(X^*)]^{-1}(f(X^*) - \mathbb{E}_F[f^*(X^*)]) \tag{2.4.48}$$

For a Gaussian process emulator and conditional on the simulation data, $D_{\mathrm{MD}}(f(X^*))$ follows an Fisher-Snedecor distribution with m and $n-p$ degrees of freedom. This is given in Equation (2.4.49) [8]. For Bayes linear emulation, Equation (2.4.49) can still be used to provide an indication of whether the diagnostics are satisfactory.

$$\frac{n-p}{m(n-p-2)} D_{\mathrm{MD}}(f(X^*)) \sim \mathcal{F}_{m,n-p} \tag{2.4.49}$$

A further diagnostic is the vector of transformed errors presented in Equation (2.4.50), where Γ is a standard deviation matrix such that $\mathrm{Var}_F[f^*(X^*)] = \Gamma\Gamma^{\mathrm{T}}$. This improves the interpretability of the individual correlated prediction errors compared to Equation (2.4.47).

$$D_{\Gamma}(f(X^*)) = \Gamma^{-1}(f(X^*) - \mathbb{E}_F[f^*(X^*)]) \tag{2.4.50}$$

There exist multiple choices of decomposition for Γ including the Eigen, Cholesky and Pivoted Cholesky decompositions with further details in [8]. As for the individual standardised prediction errors, under the Gaussian process assumption and conditional on the fitting data, each of the individual errors follows a t-distribution with $(n-p)$ degrees of freedom. Moreover, $D_{\mathrm{MD}}(f(X^*)) = D_{\Gamma}(f(X^*))^{\mathrm{T}} D_{\Gamma}(f(X^*))$.

Each of these diagnostic statistics can be used to detect problems with an emulator, with both large and small values being potential causes of concern. If there exists many large standardised prediction errors across the parameter space, this suggests a global problem that the emulator is overconfident meaning that its adjusted variance is consistently too small. Similarly, large numbers of particularly small standardised prediction errors should also be treated with caution since this may indicate an emulator which is globally under-confident; its adjusted variance is consistently too large. In general, under-confidence is less of a concern but can limit the effectiveness of an analysis, whilst over-confidence should be addressed by considering the emulator construction.

When several large prediction errors are confined to a particular region of the parameter space, this may suggest a badly chosen mean function or an incorrect stationarity assumption. On the other hand, several small errors within a particular region may indicate that the emulator is too conservative in its error estimation in this part of the space. For validation data points in close proximity to parameter vectors used to fit the emulator, particularly large or small standardised errors can indicate over or under estimation respectively of correlation length hyperparameters.

Graphical methods enable easier identification of patterns amongst residuals. Examples include plots of the individual standardised prediction errors in Equations (2.4.47) and (2.4.50) versus the simulated responses, $f(X^*)$; the emulator expectation, $\mathbb{E}_F[f^*(X^*)]$; the simulation index; and each of the input parameters [8]. None of these plots should show any form of pattern. If one exists, it may indicate problems with the emulator specification such as the requirement for additional deterministic functions within the mean function or non-stationarity within the covariance function. A generalisation for the plots versus inputs beyond [8] is to inspect contour plots of the standardised prediction errors for each pair of inputs. Another useful plot is the credible interval versus the simulated responses, $f(X^*)$, which highlights whether there exists any systematic poor prediction for particular output values. Under the Gaussian process assumption, quantile-quantile plots can

also be a useful tool, and are required to validate the normality assumption of the GP.

There exist many methods for performing emulator diagnostics, yet no single diagnostic is capable of highlighting all possible issues with an emulator. It is therefore best practice to use a combination of diagnostic techniques in order to certify that an emulator is an adequate representation of a computer simulator.

2.5 Multivariate Emulation

Emulation of multiple computer simulator outputs of interest may be performed using the methodology described in Section 2.4 by separately emulating each of the scalar outputs, thus treating them as independent. However, this fails to take into account any correlation between the outputs which may provide valuable insight of the system behaviour leading to more accurate emulators and more robust inferences. Multivariate emulation techniques provide a means of accounting for joint correlation structure by imposing a covariance function over the outputs. Note that this can be challenging to accurately specify, especially when there are a large number of outputs.

Let $f\colon \mathcal{X} \to \mathbb{R}^q$ denote a computer simulator with q-dimensional output and input parameter vector $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^D$. A natural extension to the univariate emulation formulation presented in Section 2.4.1 is given by:

$$f(\mathbf{x}) = B^T \mathbf{g}(\mathbf{x}_A) + \mathbf{u}(\mathbf{x}_A) + \mathbf{w}(\mathbf{x}) \tag{2.5.1}$$

The subscript A denotes the collection of active parameters over all simulator outputs and are input to the vector-valued function $\mathbf{g}(\mathbf{x}_A) \in \mathbb{R}^p$; a generalisation of the deterministic functions which are shared across all outputs, and where $B \in \mathbb{R}^{p \times q}$ is a matrix of coefficients. The weakly stationary stochastic process term extends to $\mathbf{u}(\mathbf{x}_A)$ with $\mathbb{E}[\mathbf{u}(\mathbf{x}_A)] = \mathbf{0} \in \mathbb{R}^q$. A common means of accounting for covariance between multiple outputs is to impose a separability condition such that

$\text{Cov}[\mathbf{u}(\mathbf{x}_A), \mathbf{u}(\mathbf{x}'_A)] = \kappa(\mathbf{x}_A, \mathbf{x}'_X)\Sigma_{\text{MV}}$, where $\kappa(\cdot,\cdot)$ provides the correlation structure over the input parameter space, $\mathcal{X}$, and $\Sigma_{\text{MV}} \in \mathbb{R}^{q\times q}$ is a positive-definite matrix that describes the covariance between the simulator outputs [23]. The final term is a generalisation of the nugget term such that $\mathbb{E}[\mathbf{w}(\mathbf{x})] = \mathbf{0} \in \mathbb{R}^q$ and $\text{Cov}[\mathbf{w}_i(\mathbf{x}), \mathbf{w}_j(\mathbf{x}')] = \sigma^2_{w_i}\mathbb{1}_{\{i=j\}}\mathbb{1}_{\{\mathbf{x}=\mathbf{x}'\}}$. The Bayes linear update equations follow a natural extension to the Bayes linear adjustment formulae in Equations (2.2.3) to (2.2.5) to vectors of outputs.

A separable multivariate Gaussian process emulator is defined in Equation (2.5.2) [23], where $\mathbf{m}(\mathbf{x}) \in \mathbb{R}^q$ is a multivariate emulator mean function, with B, $\mathbf{g}(\cdot)$, $\kappa(\cdot,\cdot)$ and Σ_{MV} as above.

$$f(\cdot) \sim \mathcal{GP}_q(\mathbf{m}(\cdot), \kappa(\cdot,\cdot)\Sigma_{\text{MV}}) \tag{2.5.2}$$

$$\mathbf{m}(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})] = B^{\text{T}}\mathbf{g}(\mathbf{x}) \tag{2.5.3}$$

This includes the distributional assumption of a matrix-variate Gaussian process, an extension of the univariate Gaussian process defined in Definition 2.5.1. The normal conditional formulae in Equation (2.4.17) provide a means of updating given a set of simulations. In general, it can be computationally expensive to update a multivariate emulator, however the separability assumption helps to alleviate this by splitting the costly matrix inversion calculations into two parts dependent on the inputs and outputs respectively using a Kronecker product factorisation [149].

Definition 2.5.1. An unknown multivariate function $f(\cdot)$ with inputs $\mathbf{x} \in \mathcal{X}$ such that $f(\mathbf{x}) \in \mathbb{R}^q$ is a **matrix-variate Gaussian Process** if and only if for every $n \in \mathbb{N}$, the joint distribution of $(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)) \in \mathbb{R}^{q\times n}$ is a matrix-variate normal distribution for all $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{X}$.

The above multivariate emulator structure utilises separability of the input and output spaces within the residual covariance function. This can be extended further to form an outer-product emulator imposing a separability assumption on the regression components contained in $\mathbf{g}(\cdot)$ which achieves further computational savings. Full details can be found in [149], with an application in [151]. It is also noted

that the above multivariate emulation methodology of [23] is a special case of the outer-product emulator.

Separability assumptions can be too restrictive and lead to poor predictive performance with non-separable emulators instead providing more flexible choices for the covariance structure. Examples include computing the covariance function via either convolution or linear model of coregionalisation methods, as discussed in [53]. However, it is noted that non-separable emulators are not necessarily as predictable as multiple independent univariate emulators or a separable multivariate emulator for high-dimensional output, whilst also being much more computationally expensive to fit and evaluate. If computational resources allow, best practice is to fit several types of multivariate emulator using diagnostics to determine the most suitable for use within the analysis.

Another scenario is dynamic simulators which yield time series output with quantities of interest reported at multiple time steps. For an individual output type, this temporal correlation may be accounted for by elevating the time variable to a direct input to the mathematical representation of the simulator [23]. The univariate emulation techniques of Section 2.4 may then be applied and the emulator evaluated for time points and parameter settings of interest. Note that the above described separable emulator framework provides greater flexibility for the choice of covariance structure whilst also reducing computational load.

Simulators may possess an extremely high number of outputs for which it is computationally impractical and potentially unnecessary to construct a complex multivariate emulator over all outputs. Dimensional reduction techniques seek to establish a lower-dimensional representation than the full set of outputs. Principal Component Analysis (PCA) is a common technique used to identify an orthogonal basis which correspond to the largest proportion of the total variation over the outputs, after accounting for all previous principal components. Emulation for high-dimensional output projected onto their principal component basis is implemented in [87]. Another technique is principal variable selection [120]. This is used to compose

a subset of the simulator outputs which best characterise and preserve the original data structure and information by accounting for a large proportion of the total variation [33]. The identified principle variables may be used within emulation [30].

2.6 Multilevel Emulation

Situations arise in which there exist a number of models for the same physical system which act on the same input space as a function to the same set of outputs, with the models forming a hierarchy in terms of their level of complexity. Differing levels of simulator accuracy can occur in applications for several reasons including: modification of the mathematical structure to either simplify or introduce additional or more detailed equations to describe the underlying physics of the system; changing the step size of the numerical solver; and increasing the grid or mesh resolution [32]. It is noted that the different versions of the models usually share many qualitative features. Such practices are regularly observed within the petroleum industry in a process known as upscaling [29, 30, 32]. In general, the ramifications of using more complex models is longer evaluation times, but will yield more accurate results, assuming the model is representative of the system. It is therefore desirable to optimally combine the various models using the cheaper but less accurate models to inform those of greater complexity, but for which only a limited number of simulations may be performed.

Multilevel emulation, also known as multiscale or multi-fidelity emulation, involves constructing a hierarchy of emulators, each of which are informed by both simulations from the corresponding computer model, and prior information provided by the emulator for the preceding coarser model [28, 32, 132]. The starting point for multilevel emulation is to construct an emulator for the most coarse simulator, usually via the standard methodology presented in Section 2.4. The next step involves sequentially linking emulators for each model with one approach being to use a first-order autoregressive type model to link each level of the hierarchy [28, 108].

This is further developed to incorporate a parameter dependence into the autoregression coefficient in [143, 144]. In situations where the various simulators may be viewed as particular cases from an overarching computer model, another approach is the cumulative roughness model which introduces a complexity parameter [108]. An alternative means of multilevel emulation is discussed in [30, 32] which directly uses the coarse emulator structure and introduces multilevel weight hyperparameters for each emulator term. This is based on the prior belief that each part represents some qualitative physical effect. Methodology has also been developed for multilevel emulation of stochastic computer models through an adaption of the autoregressive framework in [106].

Consideration must also be given to the design of simulations in terms of both the number and locations for each version of the computer model. In general, there will be a larger number of simulations conducted for the cheaper computer models with progressively fewer for each level in the hierarchy with the exact distribution depending on the evaluation time and available computational resources. One such method is border-block designs discussed in [32].

2.7 Uncertainty Quantification

Computer models provide a means of studying complex real world physical systems and often include many input parameters that specify the model, and a set of outputs for which it may be possible to observe some or all for the real system. However, computer models do not provide an exact representation of the phenomena for which they describe. Robust analyses therefore require a comprehensive uncertainty quantification in order to assess the difference between the computer model and the real world physical system. Omission of uncertainties is a failure to acknowledge the limitations of the modelling process, measurement errors and the natural variability which exists within a stochastic real world system. This can lead to misleading conclusions. Moreover, it should be noted that an uncertainty quantification is

specific to the application and analysis in question.

In Section 2.7.1, methods of linking computer models to their corresponding physical systems are introduced, with several common types of uncertainties presented in Section 2.7.2. Further discussion of observational error and structural model discrepancy can be found in Sections 2.7.3 and 2.7.4 respectively, along with comments on quantification methods. This is expanded within the context of decision support in Section 3.5.

2.7.1 Linking the Computer Model to the Real Physical System

Uncertainty quantification provides a structured link between the simulator which maps input vector $\mathbf{x}$ to uncertain output $f(\mathbf{x})$, and the true value for the physical system, y, each of which may be a scalar or a vector. A common assumption is for a general additive error structure shown in Equation (2.7.1), where ε_i represents each form of uncertainty considered with a possible dependence on the location in the input parameter space [58, 59, 170, 171, 178]. In addition, it is often assumed that the uncertainties are mutually uncorrelated, denoted as $\varepsilon_i \perp\!\!\!\perp \varepsilon_j$ for $i \neq j$, as well as uncorrelated with the true system value, $y \perp\!\!\!\perp \varepsilon_i$ for each i.

$$y = f(\mathbf{x}) + \sum_i \varepsilon_i(\mathbf{x}) \tag{2.7.1}$$

Computer simulator output $f(\mathbf{x})$ is often approximated using an emulator representation, denoted in this section by $f^*(\mathbf{x})$, which implicitly encompasses the additional uncertainty. The additive error structure is obtained via a substitution as:

$$y = f^*(\mathbf{x}) + \sum_i \varepsilon_i(\mathbf{x}) \tag{2.7.2}$$

Following the second-order Bayes linear formulation, as in Section 2.2.2, it is sufficient to consider the expectation, variance and covariances of each of the uncertainties. In a full Bayesian framework, as in Section 2.2.1, it is necessary to specify probability distributions for each of the uncertainties.

An alternative form of link between the computer model and the physical system is a multiplicative error structure described in Equation (2.7.3), where ε is a source of uncertainty with mean zero. When y and $f(\mathbf{x})$ are vectors, $\varepsilon(\mathbf{x})$ is a vector of multiplicative errors, "1" should be interpreted as a vector of ones, and $\odot$ represents the Hadamard product.

$$y = (1 + \varepsilon(\mathbf{x})) \odot f(\mathbf{x}) \tag{2.7.3}$$

Note that it is possible to obtain multiplicative uncertainties using an additive error structure by considering the logarithms of quantities of interest [178].

The relationship between the real physical system and the corresponding computer model or models may be further abstracted by introducing the concepts of direct and indirect simulators, with the distinction made depending on whether the inputs are measurable or not respectively, and if model discrepancy (see Section 2.7.4) is probabilistically independent of the inputs. Indirect simulators also possess a set of tuning parameters as inputs which: ensure a good fit to the simulated or observed data; account for poorly understood physics; and numerical solver deficiencies. A comprehensive analysis presenting a framework for connecting these along with all other sources of uncertainty is described in [67].

2.7.2 Types of Uncertainties

Computer models can never capture the full detailed nature of a real physical system with commonly occurring sources of uncertainty often categorised into several canonical forms [29, 69, 107, 170, 171, 178]. These include:

- **Parameter uncertainty** which pertains to a lack of knowledge of suitable input parameter values. Where there exist several types of parameters, as discussed in Section 2.1.3, this form of uncertainty may be decomposed further. In addition, there is the dilemma of whether there exists model parameter settings which yield an acceptable match between the model output and the observed system, up to any other sources of uncertainty. This is the focus of

history matching and calibration presented in Section 2.8.

- **Simulator uncertainty** relates to the fact that computer models form a function, either deterministic or stochastic, which map from the input parameter space to a set of outputs. However, the simulator output is unknown except at points in parameter space where the model has been evaluated. Limitations due to the computational expense of evaluating such functions discussed in Section 2.1.2 makes this a large problem, especially in high-dimensional parameter spaces. This form of uncertainty may be addressed through the use of emulators described in Section 2.4 to statistically approximate computer models at as yet unevaluated parameter settings, whilst providing a quantification of the induced uncertainty.
- **Structural model discrepancy**, also known as **structural uncertainty** or **model inadequacy**, represents the difference between the real physical system and the associated computer model. Further discussion is found in Section 2.7.4 [59, 69].
- **Observational or experimental error** concerns inaccuracies in the process of obtaining measurements of outputs from the real system. This can be physical or a consequence of any processing required of the collected data. This form of uncertainty is further described in Section 2.7.3.
- **Initial conditions and forcing function uncertainty**, sometimes referred to as residual variability [107], includes any other part of the model which must be specified before the model can be evaluated. This may be incorporated into the structural uncertainty, as in [171], with alternative means appealing to second-order exchangeability, as discussed in [89].

Other forms of uncertainty exist, but these are specific to each application and analysis.

For a full analysis of any computer model, all major sources of uncertainty should be incorporated. Where it is not possible to accurately quantify them, the best strategy is to devise a conservative estimate for their magnitude in order to

reduce the risk of incorrect inferences. It is always possible at a later time to reduce the uncertainty as a greater state of knowledge becomes available, as is the case in sequential analyses such as history matching (Section 2.8) or iterative decision support (Section 3.6). Once uncertainties are assessed to be smaller than their true value, it is very difficult to undo this choice without recommencing the analysis. Note that the disadvantage of using conservative errors is a reduction in the power to perform meaningful inferences. In history matching or iterative decision support this manifests as an inability to discriminate between parts of the parameter space which should be classified as implausible.

2.7.3 Observational Error

Uncertainty due to observational errors occurs whenever observed data, denoted by either a scalar or vector $\mathbf{z} \in \mathbb{R}^s$, is used to aid inferences for the true state of the system, $\mathbf{y}$. Observational error is defined in Definition 2.7.1 and can be attributed to a number of causes including: errors as a result of preprocessing based on scientific theory which is not incorporated into the model; uncertainty introduced to the system as a consequence of the measuring process; and the inability to measure the quantity (or quantities) of interest directly, thus relying on the interpretation and processing of measurements of some other quantity or quantities [171].

Definition 2.7.1. Observational error is the variability in the measured value of a quantity compared to its true value [178]. Observational error is often denoted by the vector $\mathbf{e}$.

The additive error structure described in Equation (2.7.4) is a prevalent modelling choice [2, 29, 107, 171, 178]. It is often assumed that the observational errors are independent of both $\mathbf{y}$ and $\mathbf{x}$, hence unlike in Section 2.7.1, $\mathbf{e}$ is not a function of $\mathbf{x}$. Further common assumptions depend on whether $\mathbf{z}$ is a scalar or a vector. In the first instance, $\mathbb{E}[e] = 0$ and $\text{Var}[e] = \sigma_e^2$, whilst for a vector of observational errors, it is common to assume that the errors are independent with $\mathbb{E}[\mathbf{e}] = \mathbf{0}$ and

$\text{Var}[\mathbf{e}] = \Sigma_e = \sigma_e^2 I_s$. Quantification of this source of uncertainty is most often based on expert knowledge from the relevant scientific field(s).

$$\mathbf{z} = \mathbf{y} + \mathbf{e} \tag{2.7.4}$$

Substituting the physical system, about which the aim is to make inferences, for a computer model and combining Equations (2.7.1) and (2.7.4) leads to the full additive error structure in Equation (2.7.5). The approximation of the computer model using an emulator (as in Equation (2.7.2)) results in the additive error structure shown in Equation (2.7.6).

$$\mathbf{z} = f(\mathbf{x}) + \sum_i \varepsilon_i(\mathbf{x}) + \mathbf{e} \tag{2.7.5}$$

$$\mathbf{z} = f^*(\mathbf{x}) + \sum_i \varepsilon_i(\mathbf{x}) + \mathbf{e} \tag{2.7.6}$$

Note that for simulators which include model parameters, the settings which lead to the observations is usually unknown with the problem of identifying such parameter setting(s) corresponding to parameter uncertainty introduced in Section 2.7.2 and can be addressed using history matching or calibration in Section 2.8.

2.7.4 Structural Model Discrepancy

A computer model is a mathematical representation of a physical system. As such, no computer model is an exact representation of its corresponding physical system leading to the introduction of structural model discrepancy, also known as structural uncertainty or model inadequacy, which is defined in Definition 2.7.2 and abbreviated to model discrepancy throughout this thesis.

Definition 2.7.2. Structural model discrepancy is the uncertainty due to the differences between the model and the true physical system [29, 107, 171, 178]. It is commonly denoted the vector $\varepsilon_{\text{MD}}(\mathbf{x})$, allowing for the possible dependence on the inputs $\mathbf{x}$.

Causes of model discrepancy can broadly be described within two categories [65].

Firstly, constraints on scientific knowledge lead to the omission or approximation of certain physical processes within the model. The second cause of model discrepancy arises from mathematical or numerical approximations. This includes: model simplification in order to ensure that it is solvable; variability due to the choice of initial conditions; numerical solver considerations; and discretisation approximations of continuous space and time. As long as these uncertainties are acknowledged and assessed within an analysis using the simulator, then the modelling process is valid. Failure to incorporate the modelling limitations constitutes a "perfect model" analysis [61, 64], which can lead to false conclusions such as the belief that the future evolution of a system may be predicted exactly.

Common assumptions for model discrepancy include that $\varepsilon_{\text{MD}}(\cdot)$ is independent of $\mathbf{y}$ and each other source of uncertainty, with $\mathbb{E}[\varepsilon_{\text{MD}}(\mathbf{x})] = 0$. In addition, for a scalar $\varepsilon_{\text{MD}}(\cdot)$ it is assumed that $\text{Var}[\varepsilon_{\text{MD}}(\mathbf{x})] = \sigma^2_{\text{MD}}$, whilst for multiple outputs $\text{Var}[\varepsilon_{\text{MD}}(\mathbf{x})] = \Sigma_{\text{MD}}(\mathbf{x})$ [178]. The covariance structure described in Σ_{MD} may be complex with non-zero covariances between components of $\varepsilon_{\text{MD}}(\cdot)$. Probabilistic or expectation statements about ε_{MD} can be formed and verified by performing exploratory model evaluations for appropriate initial conditions to form a lower bound for the variance σ^2_{MD}. A second approach is to elicit expert judgements. It is particularly effective to combine these two methods iteratively with experts assessing the revised prior specifications in light of simulation data, as in [28]. Accurately quantifying the structure of model discrepancy can be an extremely complex task, especially in high-dimensional input settings with large numbers of outputs, or if there is perceived to be a dependence on $\mathbf{x}$.

Model discrepancy can be split into a sum of internal and external model discrepancy, defined in Definitions 2.7.3 and 2.7.4 and denoted by $\varepsilon_{\text{MD,int}}(\cdot)$ and $\varepsilon_{\text{MD,ext}}(\cdot)$ respectively. The decomposition of model discrepancy is assumed additive as follows:

$$\varepsilon_{\text{MD}}(\mathbf{x}) = \varepsilon_{\text{MD,int}}(\mathbf{x}) + \varepsilon_{\text{MD,ext}}(\mathbf{x}) \tag{2.7.7}$$

Internal Model Discrepancy

Definition 2.7.3. Internal model discrepancy is defined to be any aspect of the structural model discrepancy which can be assessed using experiments on the model. This includes an assessment of the magnitude and the dependency structure [65].

An assessment of the internal model discrepancy can be performed to obtain a lower bound for the induced uncertainty, whilst it is also possible to determine certain structural properties. Sources of internal model discrepancy may be modelled additively and assumed to be mutually uncorrelated with examples including [64, 65]:

- Condition uncertainty from any forcing functions, boundary and initial conditions within the model.
- Stochastic uncertainty arising where the model is either stochastic, or should be designed to be stochastic.
- Solution or numerical uncertainty as a result of numerical solver accuracy leading to a certain level of approximation to the output.
- Parameter uncertainty within the emulator due to inactive variables.
- Population variability within the real world system where individuals exhibit the same behaviour, but for different input parameter settings. For example, in systems biology this manifests as biological variability across individuals [178].

Internal model discrepancy can be assessed via experiments on the model. Let $\mathcal{I}$ be the set of internal model discrepancy types being considered, denoted as $\varepsilon_{\text{MD,int},j}(\cdot)$ for $j \in \mathcal{I}$. Under the Bayes linear approach, it is necessary and sufficient to specify the expectation, variance and covariance structure between them. For $j \in \mathcal{I}$, let $\mathbb{E}[\varepsilon_{\text{MD,int},j}(\mathbf{x})]$ and $\Sigma_{\text{MD,int},j}(\mathbf{x})$ be the expectation and variance (matrix) respectively. Within a general framework, all types of internal model discrepancy are simultaneously assessed using a careful design. A common assumption is the types are mutually uncorrelated, leading the internal model discrepancy expecta-

tion and variance to decompose in Equations (2.7.8) and (2.7.9) respectively, with $\text{Cov}[\varepsilon_{\text{MD,int},i}(\mathbf{x}), \varepsilon_{\text{MD,int},j}(\mathbf{x})] = \Sigma_{\text{MD,int},i}(\mathbf{x})\mathbb{1}_{\{i=j\}}$.

$$\mathbb{E}[\varepsilon_{\text{MD,int}}(\mathbf{x})] = \sum_{j\in\mathcal{I}} \mathbb{E}[\varepsilon_{\text{MD,int},j}(\mathbf{x})] \tag{2.7.8}$$

$$\text{Var}[\varepsilon_{\text{MD,int}}(\mathbf{x})] = \Sigma_{\text{MD,int}}(\mathbf{x}) = \sum_{j\in\mathcal{I}} \Sigma_{\text{MD,int},j}(\mathbf{x}) \tag{2.7.9}$$

Methods for assessing internal model discrepancy involve varying certain quantities within the model to obtain empirical estimates of the necessary expectations [65]. Depending on the speed of the model, this can be infeasible due to computational expense. The expected internal model discrepancy is used to account for bias which results in systematic deviations between the model output and the true system. Moreover, in assessing the variance matrix (or matrices), the magnitude and structure should be analysed separately by splitting the empirical variance matrix into a vector of variances and a correlation matrix which is scale invariant.

External Model Discrepancy

Definition 2.7.4. External model discrepancy refers to all aspects of the structural model discrepancy which are not incorporated into the internal model discrepancy, and specifically refers to the inherent limitations of the modelling process due to missing or misspecified aspects of the physical model [65].

Due to the nature of external model discrepancy, it cannot be quantified by computer experiments on the model, and as such, must be specified by expert judgement, through careful assessment of the reasons for failure of the model to match the observed system. A detailed accounted is provided in [171]. Note that a common assumption is that $\mathbb{E}[\varepsilon_{\text{MD,ext}}(\mathbf{x})] = 0$ for all parameter vectors.

Situations transpire where there is no access to expert judgement regarding external model discrepancy. Instead, the interest lies in determining the magnitude of the external model discrepancy beyond which no meaningful inferences can be drawn from analyses of the model. In these situations it is sometimes reasonable to

assume that the external model discrepancy error has the same dependency structure as the internal model discrepancy error, and hence is treated as a scalar multiple of the internal model discrepancy. That is $\Sigma_{\mathrm{MD,ext}}(\mathbf{x}) = \lambda\Sigma_{\mathrm{MD,int}}(\mathbf{x})$ for some $\lambda > 0$, resulting in Equation (2.7.10) for the overall model discrepancy variance.

$$\Sigma_{\mathrm{MD}}(\mathbf{x}) = (1+\lambda)\Sigma_{\mathrm{MD,int}}(\mathbf{x}) \tag{2.7.10}$$

Other Approaches to Accounting for Model Discrepancy

There exist alternative approaches to assessing model discrepancy. Within a full Bayesian analysis in [17], a GP and a constrained GP prior distribution for model discrepancy are used which are shown to reduce the issues of bias and overconfidence. A six-step procedure for computer model validation is presented in [9] to determine the adequacy of a computer model as a representation of a physical system including an assessment of the accuracy in terms of potential bias and uncertainty. Another expression for some forms of uncertainty, such as induced by variable initial conditions, is via the creation of an ensemble of multiple versions of a computer model. Second-order exchangeability arguments may then be used to assess the induced model discrepancy [89]. A further approach to dealing with model discrepancy described in [163] involves decomposing a computer model into subfunctions and individually assessing the uncertainty propagation through the model. Where such a decomposition is possible, it would be worthwhile to consider methods of emulating networks of simulators, as discussed in [96, 111]. For a careful but complex treatment of model discrepancy via the abstract concept of a reified model and a reified Bayesian emulator, see [68].

2.8 History Matching

History matching concerns the problem of identifying a set of model parameter vectors which yield acceptable matches between the computer model output and the real physical system based on observed historical data and whilst accounting

for model discrepancy and observational errors [27, 29, 58, 170, 171, 178]. This is a means of addressing the statistical inverse problem and may be utilised as a precursor to a full Bayesian model calibration [107] that does not require a full probabilistic specification [171]. Throughout this section the focus is on model parameters, $\mathbf{x}$, since these govern the model behaviour, whilst decision parameters, $\mathbf{d}$, may also be incorporated separately if it is known that a system was controlled in a particular manner prior to taking measurements.

History matching is an iterative procedure which works in waves to eliminate regions of the model parameter space which do not yield satisfactory matches between the computer model output and the observed data, up to the uncertainties present. This requires a comprehensive exploration of the model parameter space and thus Bayesian emulators, described in Section 2.4, are incorporated because of their speed of evaluation, ability to accurately estimate model output, and can be substituted for the model in lengthy calculations. Uncertainty quantification methods (Section 2.7) are also required to link the model with the system and observations. Comparisons are performed using an implausibility measure. Note that if no parameters are found for which the model is compatible with the real observed system, then this may indicate that the model is not a good representation of reality. History matching has been successfully applied across a range of scientific disciplines including: climate modelling [19, 40, 65, 185]; cosmology [170, 171]; epidemiology [2, 3]; systems biology [178]; and petroleum reservoir modelling [27, 29, 30].

This section will begin by explaining the best input assumption in Section 2.8.1, before presenting a selection of implausibility measures in Section 2.8.2, with a means of selecting a suitable cut-off described in Section 2.8.3. The history matching algorithm is presented in Section 2.8.4, with a discussion of methods for resampling from the remaining model parameter space at each iteration in Section 2.8.5, and stopping rules in Section 2.8.6. A comparison with Bayesian calibration methods is presented in Section 2.8.7.

2.8.1 Best Input Assumption

An assumption usually made within calibration, although not strictly necessary in history matching, is the existence of a model parameter vector $\mathbf{x}^* \in \mathcal{X}$, which is unknown prior to analysis and contains the maximum information about the true system state, $\mathbf{y} \in \mathbb{R}^q$, either a scalar or a vector. In addition, it is typically assumed that $\mathbf{x}^*$ and the simulator are uncorrelated, denoted as $\mathbf{x}^* \perp\!\!\!\perp f$, such that $f^* = f(\mathbf{x}^*)$ is sufficient for the simulator f, as well as for the uncertainty about $\mathbf{y}$. This is termed the best input assumption [58, 66, 171, 178]. The model output $f(\mathbf{x}^*)$ may not necessarily match reality $\mathbf{y}$ exactly due to model discrepancy which unlike in Section 2.7 is only considered at the best input setting. For the commonly used additive error structure this leads to the model discrepancy defined in Equation (2.8.1), with an implication being that $\varepsilon_{\text{MD}} \perp\!\!\!\perp \{f, \mathbf{x}^*\}$.

$$\varepsilon_{\text{MD}} = \mathbf{y} - f(\mathbf{x}^*) \tag{2.8.1}$$

Assessment of the model discrepancy may follow a similar line to the ideas portrayed in Section 2.7.4.

History matching requires system observations, $\mathbf{z} \in \mathbb{R}^q$, measured with errors, $\mathbf{e} \in \mathbb{R}^q$. It is also typically assumed that $f(\mathbf{x}^*) \perp\!\!\!\perp \varepsilon_{\text{MD}} \perp\!\!\!\perp \mathbf{e}$. These are the two most relevant forms of uncertainty within history matching, although others may be incorporated if it is deemed necessary. The multivariate additive error structure used for history matching is given by:

$$\mathbf{z} = f(\mathbf{x}^*) + \varepsilon_{\text{MD}} + \mathbf{e} \tag{2.8.2}$$

Note that it is also assumed that for each component y_i of $\mathbf{y}$, there exists a respective z_i from $\mathbf{z}$ measured with error e_i, for which $f_i(\mathbf{x}^*)$ denotes the corresponding simulator output at $\mathbf{x}^*$ with model discrepancy $\varepsilon_{\text{MD},i}$. This leads to the univariate additive error structure in Equation (2.8.3).

$$z_i = f_i(\mathbf{x}^*) + \varepsilon_{\text{MD},i} + e_i \tag{2.8.3}$$

2.8.2 Implausibility Measures

History matching requires a method of comparing the simulator output evaluated at a given model parameter vector $\mathbf{x}$ with observations of the system in order to identify those parameter settings which may reasonably yield an acceptable match up to the uncertainties, and those which are deemed unlikely to do so. This leads to the development of an implausibility measure which is formally defined in Definition 2.8.1, along with the meaning of implausible and non-implausible model parameter settings [27, 29, 171].

Definition 2.8.1. An **implausibility measure** is a function $I(\mathbf{x})$ defined over $\mathcal{X}$ which is large at input $\mathbf{x}$ which are suspected to yield a poor match between simulator output and observed data with respect to an uncertainty quantification and a specified level of tolerance. Input settings for which $I(\mathbf{x})$ is large are described as **implausible**, whilst any other $\mathbf{x}$ are categorised as **non-implausible**.

It is important to stress that even if a model parameter vector $\mathbf{x}$ is classified as non-implausible, it does not necessarily guarantee a close match between the simulator output and the true system value, only that the difference is small relative to the magnitude of the total uncertainty within the analysis. Instead, implausibility measures are used within history matching to formulate a candidate set of values $\mathcal{X}^*$ which should contain $\mathbf{x}^*$. This is termed the non-implausible region or set.

For an individual simulator output $f_i(\mathbf{x})$, an implausibility measure is constructed by assessing how much $f_i(\mathbf{x})$ differs from the true value y_i of the system. At this stage the only uncertainty is the model discrepancy, so the ideal measure of uncertainty would be the square of the standardised distance $\frac{(y_i - f_i(\mathbf{x}))^2}{\mathrm{Var}[\varepsilon_{\mathrm{MD},i}]}$. However, y_i is unknown, thus rendering this calculation impossible, and hence it should be substituted for an observation z_i of y_i. This introduces observational error e_i, leading to $\frac{(z_i - f_i(\mathbf{x}))^2}{\mathrm{Var}[\varepsilon_{\mathrm{MD},i}] + \mathrm{Var}[e_i]}$. It has already been highlighted that simulators are often highly computationally expensive and hence it is infeasible to evaluate for even a modest number of model parameter vectors, let alone across the entire input space $\mathcal{X}$. This is where an

emulator is introduced as a surrogate for the computer model with the univariate implausibility measure defined in Equation (2.8.4) in which it is assumed the model discrepancy and observational error are uncorrelated. Recall that $\mathbb{E}_{F_i}[f_i(\mathbf{x})]$ and $\mathrm{Var}_{F_i}[f_i(\mathbf{x})]$ denote the emulator adjusted expectation and variance respectively, where the subscript F_i denotes conditioning on simulation output for $f_i(\cdot)$ [27, 29, 171, 178].

$$I_i^2(\mathbf{x}) = \frac{(\mathbb{E}_{F_i}[f_i(\mathbf{x})] - z_i)^2}{\mathrm{Var}_{F_i}[f_i(\mathbf{x})] + \mathrm{Var}[\epsilon_{\mathrm{MD},i}] + \mathrm{Var}[e_i]} \tag{2.8.4}$$

Large values of $I_i^2(\mathbf{x})$ are a result of a large difference between the emulator adjusted expectation and the observed data relative to the magnitude of the combined variance of all the uncertainties. When this is the case it is unlikely that the model evaluated at $\mathbf{x}$ will yield an acceptable match with the observed data, hence Equation (2.8.4) defines an implausibility measure, thus $\mathbf{x}$ would be rejected as implausible. Within the history matching algorithm, a threshold value c beyond which an implausibility measure is considered large must be chosen. This means that if $I_i(\mathbf{x}) \leq c$, then $\mathbf{x}$ is classed as non-implausible, and therefore $\mathbf{x} \in \mathcal{X}^*$. Note that a popular choice is $c = 3$ using Pukelsheim's 3-sigma rule [142]. Further details can be found in Section 2.8.3.

Many analyses use simulators which produce complex, often time-dependent, multivariate output, with observational data available for some or all of these. In these situations it is necessary to consider a collection Q of univariate responses where Q does not necessarily consist of all model outputs, rather a subset of those of interest, those which are simplest to emulate at a particular wave of the history match, or those that are deemed to encapsulate the most information about all responses. Note that the size of Q may increase at each wave. For multivariate output, it is then desired to combine the implausibility measures for individual outputs into a single implausibility measure. A simple choice is the maximum over the univariate implausibility measures which is termed the maximum implausibility

measure in [29, 171, 178] and is defined as:

$$I_M(\mathbf{x}) = \max_{i \in Q} I_i(\mathbf{x}) \tag{2.8.5}$$

Note that $I_M(\mathbf{x})$ is strict; it will only classify $\mathbf{x}$ as non-implausible if it yields acceptable matches for all outputs, and hence is sensitive to inaccuracies in the individual emulators, whilst also failing to consider any multivariate structure between the outputs.

Alternative less sensitive implausibility measures for multivariate responses are the second and third maximum of the univariate implausibility measures. These are defined in Equations (2.8.6) and (2.8.7) respectively [171]. Note that this concept can be further generalised to the m-th largest univariate implausibility measure.

$$I_{2M}(\mathbf{x}) = \max_{i \in Q}(\{I_i(\mathbf{x})\} \setminus I_M(\mathbf{x})) \tag{2.8.6}$$

$$I_{3M}(\mathbf{x}) = \max_{i \in Q}(\{I_i(\mathbf{x})\} \setminus \{I_M(\mathbf{x}), I_{2M}(\mathbf{x})\}) \tag{2.8.7}$$

Using a collection of univariate implausibility measures omits the correlation structure between multiple outputs which may provide valuable additional information. An extension is therefore the multivariate implausibility measure [29, 171] defined as:

$$\begin{aligned} I^2_{multi}(\mathbf{x}) &= (\mathbf{z} - \mathbb{E}_F[f(\mathbf{x})])^{\mathrm{T}}(\mathrm{Var}_F[\mathbf{z} - \mathbb{E}[f(\mathbf{x})]])^{-1}(\mathbf{z} - \mathbb{E}_F[f(\mathbf{x})]) \\ &= (\mathbb{E}_F[f(\mathbf{x})] - \mathbf{z})^{\mathrm{T}}(\mathrm{Var}_F[f(\mathbf{x})] + \mathrm{Var}[\varepsilon_{\mathrm{MD}}] + \mathrm{Var}[\mathbf{e}])^{-1}(\mathbb{E}_F[f(\mathbf{x})] - \mathbf{z}) \\ &= (\mathbb{E}_F[f(\mathbf{x})] - \mathbf{z})^{\mathrm{T}}(\mathrm{Var}_F[f(\mathbf{x})] + \Sigma_{\mathrm{MD}} + \Sigma_e)^{-1}(\mathbb{E}_F[f(\mathbf{x})] - \mathbf{z}) \end{aligned} \tag{2.8.8}$$

The notation F corresponds to simulation data for all outputs. This requires a more careful consideration of the covariance structure via the variance matrices, $\mathrm{Var}_F[f(\mathbf{x})]$, $\mathrm{Var}[\varepsilon_{\mathrm{MD}}]$ and $\mathrm{Var}[\mathbf{e}]$, which are generally more difficult to accurately specify than individual variances. The use of the multivariate implausibility measure has the potential to greatly reduce the proportion of the model parameter space deemed non-implausible, and as such, reduce the number of waves of history matching required. However, this comes at the cost of requiring greater certainty about the

multivariate structure between both the simulator outputs and each of the errors.

Irrespective of which form of the implausibility measure is used, denoted in general by $I(\mathbf{x})$, the non-implausible region is defined in Equation (2.8.9).

$$\mathcal{X}^* := \{\mathbf{x} \mid I(\mathbf{x}) \leq c\} \tag{2.8.9}$$

2.8.3 Pukelsheim's 3-sigma Rule

A method of determining an appropriate threshold value for the implausibility is via Pukelsheim's 3-sigma rule [142], a specific case of the more general Vysochanskij-Petunin inequality [181].

Theorem 2.8.2. *Vysochanskij-Petunin Inequality*
Let X be a random variable with a continuous, unimodal distribution with mean μ and finite, non-zero variance σ^2. Then for any $r > \sqrt{\frac{8}{3}} \approx 1.63$,

$$\Pr[|X - \mu| \geq r] \leq \frac{4\sigma^2}{9r^2} \tag{2.8.10}$$

Theorem 2.8.3. *Pukelsheim's 3-sigma Rule*
For any continuous, unimodal distribution with mean μ and finite, non-zero variance σ^2, 95% of the probability must lie within $\pm 3\sigma$ of μ.

Proof. The proof follows by setting $r = 3\sigma$ in Theorem 2.8.2.

$$\Pr[|X - \mu| \geq 3\sigma) \leq \frac{4\sigma^2}{9(3\sigma)^2} = \frac{4}{81} \approx 0.0494 < 0.05 \tag{2.8.11}$$

Hence, the probability that a realisation of X falls outside the region $\mu \pm 3\sigma$ is less than 0.05. It follows that at least 95% of the probability mass lies within plus or minus 3 standard deviations of the mean. □

The assumptions are continuity and unimodality of the probability distribution, hence these results hold regardless of any skewness or asymmetry. These are reasonable assumptions for each of the defined implausibility measures and so

Pukelsheim's 3-sigma rule can be applied to suggest a cut-off of $c = 3$. Alternatively, the Vysochanskij-Petunin Inequality can be used directly to obtain an upper bound for the cut-off for a desired proportion. Moreover, Pukelsheim's 3-sigma rule also allows for the construction of approximate credible intervals for unimodal distributions with a specified upper bound on the probability level for a given width. This is employed within a Bayes linear approach.

2.8.4 History Matching Algorithm

History matching that incorporates an emulator as a surrogate for a complex computer model provides an efficient method of searching high-dimensional model parameter spaces within a formal statistical setting that incorporates all major sources of uncertainty to identify acceptable matches between the computer model output, approximated by an emulator, and observed data from the real physical system. The implausibility measures described in Section 2.8.2 are used to determine what constitutes an acceptable match. History matching is an iterative procedure in which implausible regions of model parameter space are discarded in waves to leave a non-implausible region. At each wave, indexed by k, define $\mathcal{X}_k$ be the current set of non-implausible model parameter vectors, and Q_k to be the outputs considered, and hence emulated. History matching is initialised by defining $\mathcal{X}_0 = \mathcal{X}$, the full model parameter space, and $Q_0 = \emptyset$. Let $I(\mathbf{x})$ denote a chosen implausibility measure, noting that this may change between waves. The history matching algorithm is presented below [27, 29, 58, 170, 171, 178].

1. Generate an initial space filling design of model parameter vectors from $\mathcal{X}_0$ and evaluate the model at these points.
2. Determine if any new and informative outputs exist which can be accurately emulated and were not included for the previous waves. Add them to the set Q_{k-1} to define Q_k.
3. For each output in Q_k, devise an emulator structure. Fit using the simulations and evaluate over $\mathcal{X}_{k-1}$.

4. Compute the implausibility measure(s) $I(\mathbf{x})$ over $\mathcal{X}_{k-1}$ using the emulators formed in 3.
5. Use the threshold c to obtain an updated non-implausible region $\mathcal{X}_k \coloneqq \{\mathbf{x} \mid I(\mathbf{x}) \leq c\}$.
6. If any of the following occur [171]:
   - $\mathcal{X}_k = \emptyset$, meaning that the whole parameter space has been classified as implausible.
   - Computational resources are exhausted.
   - The emulator adjusted variance is sufficiently less than the combined variance of the model discrepancy and observational error. This stopping rule is further detailed in Section 2.8.6.

   Go to step 8, otherwise go to step 7.
7. Formulate a further design from $\mathcal{X}_k$ using methods discussed in Section 2.8.5 and evaluate the simulator. Return to 2.
8. Define $\mathcal{X}^* = \mathcal{X}_k$ to be the final non-implausible region. If $\mathcal{X}^*$ is non-empty, sample a large number of model parameter vectors from $\mathcal{X}^*$ depending on the scientific need. Perform a large number of acceptable model evaluations.

The history matching process successively reduces the volume of the non-implausible parameter space such that the intermediate non-implausible regions satisfy $\mathcal{X}^* \subset \mathcal{X}_{k+1} \subset \mathcal{X}_k$. If this is not the case, then it is an indicator that the emulator(s) is poorly specified. This process is known as iterative refocusing. A major strength of history matching is its ability to quickly dismiss a large proportion of the initial model parameter space based on a relatively limited number of simulations and a small set of carefully chosen simulation outputs which are simple to emulate. At later waves, additional outputs are included which may have been initially difficult to emulate due to their challenging behaviour within certain regions of the space. Note that it is important to carefully consider $\mathcal{X}_0$ in order to avoid unnecessarily exploring an excessively large input space. This choice should be guided by expert

knowledge. Constructing designs for each successive non-implausible region can present a challenging problem. This is further discussed in Section 2.8.5.

At each successive wave it is expected that superior emulator accuracy is achieved for $f(\mathbf{x})$ (or each $f_i(\mathbf{x})$) due to several factors. For deterministic computer models the behaviour will generally be smoother within smaller regions of the model parameter space and hence better approximated by low order polynomials, as found within the regression part of the emulator structure. After each wave, further model evaluations are performed within the new non-implausible region which in general yields a higher density of model parameter settings with known simulator data within this space meaning that each new $\mathbf{x}$ is generally in closer proximity to the nearest design points. In particular, this leads to a better approximation by the stochastic process term of the emulator. As further waves of a history match are performed, the effect of the initial set of active inputs reduces with other previously inactive model parameters being elevated to active inputs as their effect on the simulator output(s) becomes more apparent. In the case of stochastic computer models, smaller non-implausible volumes permits a greater number of repetitions to drive down the emulator uncertainty.

The final step of the history matching algorithm involves sampling from the reported $\mathcal{X}^*$, and performing simulations for a selection of these candidates, as permitted by the available computational resources. These should be used to assess the quality of the history match by evaluating whether the output is a close match to the observed data, again using an implausibility measure, but omitting emulator variability. A history matching diagnostic technique is then obtained by comparing the implausibility measure computed for the simulator and emulator output to ensure that they yield a consistent result for a range of choices of $\mathbf{x}$. This may be applied both at the end of the process, but also at intermediate stages.

2.8.5 Resampling from the Current Non-Implausible Region

The design of simulations for a first wave is often generated using a maximin Latin hypercube sample or other standard design method over a hypercube parameter space discussed in Section 2.3. However, each subsequent wave of a history match requires a design over the new smaller non-implausible parameter space, which is generally of a complex non-analytic shape. This process becomes increasingly difficult as the volume of this region decreases further. Moreover, as the number of waves increases it is necessary to check that any proposed design point also satisfies all previous wave implausibility cut-offs. For the k^{th} wave, a membership function is used to assess whether an input $\mathbf{x}$ is classified as non-implausible for all previous $k-1$ waves or not. Only points which satisfy the membership function may then be sampled for the wave k design.

Uniform rejection sampling over $\mathcal{X}_k$ is a common method used to construct a design. However, a pitfall of uniform sampling is that there is no guarantee of well spaced design points which are required in order to construct accurate emulators at wave k. A preferable approach is to generate a large set of points and draw numerous subsamples of a specified size, depending on computational resources. An approximately optimal design may then be chosen with respect to a design selection criterion such as those detailed in Section 2.3.5. Alternatively, consider the proposed emulator structure for the next wave and note that the Bayes linear adjustment formula for the variance, see Equations (2.2.4) and (2.4.42), does not depend on the actual simulator output, and hence may be evaluated with the design point only. An approximately optimal design is then selected with respect to a monotone increasing function of the emulator adjusted variance. This will usually resolve the issue of poor spacing. Note that for both approaches the design optimisation is approximate. Analytic design optimisation with respect to most criteria is only possible for small designs in low-dimensional spaces of an analytic shape, often a

hypercube. Whilst this approach is simple to implement, it has several disadvantages. Firstly, it very quickly becomes computationally expensive to even identify a non-implausible model parameter vector as the volume of the parameter space becomes very small requiring repeated and increasingly large samples to be formed. This problem is exacerbated at later waves due to the need to evaluate the membership function containing an increasing number of emulators and implausibility measures to compute. Some efficiency gains can be made by uniformly sampling from the smallest hypercube within the initial model parameter space that contains $\mathcal{X}_k$, although these computational issues remain, especially when the non-implausible region is of a complex geometry. Note that uniform rejection sampling also provides a means of accurately quantifying the size of the wave k non-implausible parameter space as a proportion of the volume at wave $k-1$.

Alternative approaches have been developed for constructing designs in subsequent waves of history matching that are more adept at efficiently handling very small non-implausible volumes. One example is the Implausibility Driven Evolutionary Monte Carlo (IDEMC) algorithm developed in [186] which is based on the Evolutionary Monte Carlo (EMC) method discussed in [114]. A temperature ladder is defined based on the implausibility measures of successive waves of history matching. This produces uniform samples over the current non-implausible region.

Another algorithm described in [2] is based on a set of non-implausible model parameter vectors from the last wave of history matching, known as generating points, which are used to estimate the hyperparameters of a D-variate normal proposal distribution with an inflated variance matrix. An approximate q-dimensional normal likelihood for the data is formed based on the uncertainty structure used within the implausibility measure. Moreover, it is assumed that the model parameters possess a flat prior within the non-implausible region. This is justifiable when the volume is very small. This approach proceeds to generate a large collection of proposals that are assigned weights computed as the ratio of the likelihood to the proposal probability. Additional non-implausible posterior samples are then

obtained from the distribution defined by these weights. This technique especially works well in relatively low-dimensions, although the hyper-ellipses defined by the multivariate normal proposal distribution may be particularly large encompassing many implausible points. Another consequence of the normal proposal is that the samples are not exactly uniformly distributed.

The methods discussed in this section were done so in the context of constructing designs. It is also necessary to obtain a large number of uniformly distributed model parameter vectors over the non-implausible region at which to emulate in the subsequent wave of emulation. The uniform rejection sampling and IDEMC algorithm presented above can also be applied to this task.

2.8.6 Stopping Rule

In the history matching algorithm it is necessary to stipulate a condition that when satisfied the process concludes. This is in order to prevent the algorithm continuing when there is no further information to be gained through extra waves of history matching [178]. Several stopping rules are presented within the algorithm detailed in Section 2.8.4. Two of these rules are incontestable. Firstly, when the entire model parameter space has been classified as implausible there is no further parameter settings to be investigated, instead the model should be scrutinised as to whether it accurately represents the real world physical system, as well as assessing the accuracy of the constructed emulators and the uncertainty quantification. A second rule is to stop when there are insufficient computational resources to continue the analysis.

In most scenarios where a scientific model has been carefully constructed with the uncertainties accurately quantified, this first stopping rule is unlikely to be invoked. The second rule will eventually be achieved, although it may be inefficient and unnecessary to wait until this point. A third and most important stopping rule is introduced based on the total uncertainty attributed to the emulator. For a collection of separately emulated univariate outputs in Q, a stopping rule is defined

by the Variance Ratio (VR):

$$\mathrm{VR} = \max_{i \in Q} \left\{ \frac{\mathrm{Var}_{F_i}[f_i(\mathbf{x})]}{\mathrm{Var}[\varepsilon_{\mathrm{MD},i}] + \mathrm{Var}[e_i]} \right\} \tag{2.8.12}$$

A natural extension where the multivariate implausibility measure is used is to consider the trace of the variance matrices:

$$\mathrm{VR}_{multi} = \frac{\mathrm{Tr}(\mathrm{Var}_F[f(\mathbf{x})])}{\mathrm{Tr}(\Sigma_{\mathrm{MD}}) + \mathrm{Tr}(\Sigma_e)} \tag{2.8.13}$$

Note that other multivariate generalisations are available. This statistic is then compared to a cut-off c_{stop} with an appropriate value being $c_{stop} = \frac{1}{10}$. If $\mathrm{VR} < c_{stop}$ (or $\mathrm{VR}_{multi} < c_{stop}$), then the variance due to emulation of the model is at most $\frac{1}{10}$ of the variance due to the combination of all other uncertainties. The implication is that very little further improvement can be made in the history match at subsequent waves through additional simulations alone because the emulator uncertainty is not dominant. If this condition is met, then it would seem sensible to stop the history match.

2.8.7 Comparison with Calibration

The history matching process aims to efficiently identify model parameter settings which yield acceptable matches between the simulator and observed data, whilst making no probabilistic statements regarding which are likely to yield the closest match with the true system. Major benefits include that all calculations are efficient and simple to perform, whilst issues traditionally associated with high-dimensional input and output spaces are mitigated through the use of subsets of the model parameters and outputs. This latter point is a consequence of requiring parameter settings that produce matches with the model for all observed outputs of the real system, hence an acceptable match should be achieved for each individual output. Moreover, the successive reduction in the non-implausible space has the added benefit that at later waves it is often simpler to emulate outputs of interest due to smoothness properties over small volumes of the input space. A criticism which

may be levelled at the history matching algorithm is identifiability of the best model parameter setting. However this is not the case since history matching goes beyond the standard point estimator and reports a class of inputs respecting all incorporated uncertainties, whilst possessing the ability to discard all model parameter settings in the case of no acceptable matches with the simulator. Additional model evaluations may then be performed over a range of non-implausible model parameter settings to further investigate the differences (if any) in the output.

Bayesian calibration techniques share many similarities with history matching, but are fundamentally different. They are fully probabilistic approaches (see Section 2.2.1) which allow for more detailed analyses and yield posterior distributions for the model parameters (as well as the system outputs) from which probabilistic statements can be made in response to scientific questions. Calibration techniques are based on the best input assumption described in Section 2.8.1 and require the specification of a joint prior distribution over all parameters along with a likelihood for the output (and hence observed data), as detailed in Sections 2.2.1 and 2.2.3. The posterior distribution is then obtained via Bayes theorem. There exist calibration techniques which are closely related to history matching using Bayes linear emulators, such as through the use of Gaussian process emulation (as presented in Section 2.4.4) for calibration [9, 10, 87, 107, 132].

There also exists a Bayes linear calibration approach, as presented in [66]. Other examples of likelihood based methods include: maximum likelihood via iterated filtering; data augmentation and/or Reversible-Jump Markov chain Monte Carlo (RJMCMC), whilst likelihood approximation methods which directly use the simulator include: particle filters; Bayesian MCMC and Approximate Bayesian Computation (ABC, which approximates full Bayesian inference) [2, 88, 178, 184].

Probabilistic inference may at first be viewed as an advantage of full Bayesian calibration methods over history matching, however this also presents weaknesses. With the exception of a small class of choices for the joint prior distribution and the likelihood, the posterior distribution may be of an intractable form, requiring

numerical techniques such as Bayesian MCMC to perform inferences and sample the model parameters [178]. MCMC can be extremely computationally expensive and prone to convergence issues [55], especially in high-dimensional settings where the posterior may possess complex correlation structures and be multi-modal, potentially limiting the ability to perform inference. Moreover, MCMC is unable to properly exploit smoothness of the outputs. By comparison, the history matching calculations are efficient and can exploit any smoothness properties. In addition, as discussed in Section 2.2.3, accurate and meaningful specification of the joint prior for all of the inputs can be challenging, raising questions regarding the robustness of such an analysis. This issue is further exacerbated in higher-dimensional settings where all inputs and outputs must be simultaneously incorporated from the beginning of the analysis, unlike in the history matching procedure. See [178] for further discussion.

The aim of scientific modelling is to better understand real world physical systems, with models containing equations which govern many underlying phenomena. Some of the modelled processes may be unimportant in their effect on the observable data leading to limited information gain for certain inputs which govern any such process. The result is some model parameters having a flat marginal posterior distribution and hence an identifiability issue. Within a full probabilistic framework this is typically embodied by a posterior distribution with multiple modes and correlation ridge structures which can obscure these identifiability issues, especially in methods such as MCMC. Conversely, history matching clearly highlights such identifiability issues, and can aid in discerning the limitations of the information contained within the observed data as well as direct towards what additional data is required to learn more about these non-identifiable model parameters [2].

Another scenario which can occur is where a scientific model does not match the real world physical system for any choice of model parameter settings. Consequently there does not exist a best input, contravening an important assumption of calibration. These techniques are therefore unable to identify this non-match situation, instead yielding a posterior distribution with the majority of the probability mass

focused on a very small or narrow region of the model parameter space which corresponds to where there is the closest, but still not acceptable, match between the model and observed data. In contrast, history matching reports an empty non-implausible region suggesting further examination of the model and uncertainty specification is required.

Emulators are used within history matching in order to perform a quick and efficient global parameter search. These can also be employed within approximate inference methods as a surrogate for simulator evaluations where the computer model is too computationally expensive to practically perform the large number of required simulations. However, unlike in history matching where the focus may be limited to a few well understood outputs and a collection of active inputs, this relies on an accurate specification of the joint dependency structure between all system outputs. Moreover, it requires that the emulator(s) is of a tractable form to sample from, which in turn depends on the choice of priors and likelihood with associated issues presented above.

Given the discussed limitations, a fully Bayesian approach should only be implemented where the application warrants the time and effort required to achieve robust conclusions. History matching is most useful to identify model parameters which are likely to yield an acceptable match with the observed data including assessing if such a match exists, but can also be used in more complex tasks such as forecasting [26]. It is therefore advocated to first perform history matching as a precursor to a full calibration in order to refocus on a (potentially very small) region of the initial model parameter space that is expected to contain almost all of the probability mass of the posterior distribution. Full Bayesian calibration including MCMC can then be performed over the final wave non-implausible region at a much lower computational cost and with greater robustness due to more accurate prior belief specifications, modelling choices and uncertainty quantification [29, 65, 171]. Furthermore, history matching has been shown to be effective in situations where it is not practical, and potentially is impossible to perform full probabilistic calibration such as in [2, 3].

Chapter 3

Iterative Decision Support for Well Control Optimisation

This chapter presents novel methodology and a framework that were developed to address the substantial problem of decision support under uncertainty guided by complex and computationally expensive computer models. An example of such a problem is that of well control optimisation within the petroleum industry which is reformulated as a decision support problem. We begin by introducing the TNO OLYMPUS Field Development Optimisation Challenge in Section 3.1, which serves as motivation for the developed methodology throughout this thesis and includes a critique of the challenge and discussion of current petroleum industry methodology and approaches, although the core ideas and techniques presented in this chapter are adaptable and transferable across many scientific and industrial applications. In Section 3.2 we describe several methods for identifying a representative subset of models from an ensemble to decrease computational expense. Certain simulator outputs from the TNO OLYMPUS Field Development Optimisation Challenge possess structure which can be exploited by Bayesian emulators to achieve superior accuracy. This methodology is presented in Section 3.3 within a hierarchical framework for linking emulators for these quantities to obtain an emulator for the expected NPV. Section 3.4 contains a review of various optimisation routines and decision

theoretic approaches which may be applied to the motivating problem. Uncertainty quantification for linking computer models to the real world physical system was first explored in Section 2.7 and is further developed within the context of decision support in Section 3.5. A novel iterative decision support procedure is presented in Section 3.6 which incorporates Bayesian emulators and a comprehensive uncertainty quantification whilst highlighting its advantages compared to optimisation and decision theoretic approaches. Throughout this chapter and the subsequent applications in Chapters 4 and 5, the notation $\mathbf{d}$ is used to represent a vector of decision parameters, as discussed in Section 2.1.3.

3.1 Introduction to the TNO OLYMPUS Field Development Optimisation Challenge

Throughout this thesis the development of methodology was motivated by problems of interest within the petroleum industry. A major challenge is that of field development optimisation under uncertainty during the initial planning stages for a green field which encompasses the two important sub-problems of formulating optimal well placement (also known as field development) and well control strategies. This is often aided by increasingly complex computer simulator representations of real oil fields that are used to assess the potential future effect of implementing various strategies. The enormity and significance of this challenge is evidenced by the establishment of the TNO OLYMPUS Field Development Optimisation Challenge [93] (also abbreviated to TNO OLYMPUS Challenge) which has attracted much interest from across the industry and academic institutions through the EAGE/TNO Workshop on OLYMPUS Field Development Optimization [39].[1]

The TNO OLYMPUS Field Development Optimisation Challenge is formally introduced in Section 3.1.1 including the aims of the challenge organisers, the OLYMPUS model ensemble and a brief description of the three parts of the challenge. In

[1]EAGE is the European Association of Geoscientists and Engineers.

addition, the reformulation of the challenge within a decision support framework and the implications are discussed. Our analysis identifies multiple deficiencies with the challenge which potentially limits the usefulness of any methods that are shown to be effective for the preliminary setup when transferred onto a corresponding real world scenario. These critiques along with suggestions for improvements are debated in Section 3.1.2. Current petroleum industry approaches to field development optimisation are reviewed in Section 3.1.3.

3.1.1 Overview of the TNO OLYMPUS Field Development Optimisation Challenge

The TNO OLYMPUS Field Development Optimisation Challenge was devised by the Netherlands Organisation for Applied Scientific Research (TNO) as part of Integrated Systems Approach for Petroleum Production (ISAPP) research programme which was initiated in 2011 by TNO in collaboration with Delft University of Technology (TU Delft), and industrial partners Eni S.p.A, Statoil (now known as Equinor ASA as of May 15, 2018) and Petrobras.[2] The aim of the ISAPP research programme is to encourage the development of innovative methods for reservoir management in order to improve well productivity, increase hydrocarbon recovery and field life time. Methods include both physical hardware such as advanced sensors and smart wells, and the handling of data and the development of software to implement closed-loop and optimisation technologies based around simulators for oil reservoirs. The ultimate aim being to manage oil recovery from central control rooms with mainly automated systems used to achieve optimal daily production rates and total recovery [93].

The TNO OLYMPUS Field Development Optimisation Challenge is based around the fictitious oil reservoir model named OLYMPUS [4] (further details can be found

[2]The original ISAPP programme began in 2004 as a joint venture between TNO, TU Delft and Royal Dutch Shell with a focus on model based optimisation workflows in a closed-loop. This programme ended in 2009.

below), and was devised as an interactive competition designed to achieve these aims with a focus on oil field management, planning and development. It has attracted much attention from industry and academia with results from the active competition period presented and compared at the EAGE/TNO Workshop on OLYMPUS Field Development Optimization [39]. The challenge is setup to resemble the decision making process when presented with a surveyed green oil field.[3] This is conveyed as the formulation of optimal strategies with respect to an objective function whilst accounting for geological uncertainty, using an ensemble of 50 computer model realisations of the OLYMPUS oil field. The novelty and principal mathematical and statistical component is the methodology used to perform optimisation under uncertainty. TNO aim to investigate the following questions.

- What added value can be expected from applying optimisation techniques?
- Do there exist any good workflows?
- Which are good choices for controls?
- Are there any algorithms which perform optimisation substantially better than other algorithms?
- Should field development and well control optimisation be considered jointly?

In order to address the research questions, the TNO OLYMPUS Field Development Optimisation Challenge is split into three sub-challenges. These are:

- Well control optimisation (see Chapter 4);
- Well placement optimisation, also known as field development optimisation (see Chapter 7);
- Full field development optimisation consisting of joint well placement and well control optimisation.

The first two of these challenges are addressed in this thesis.

In each challenge participants are tasked with developing an optimal strategy with respect to maximising the expected Net Present Value (NPV) objective function

[3] A green oil field is a new subsurface region believed to contain oil or gas which has yet to be exploited meaning that no drilling, production or injection has been performed

over the 20 year field lifetime (starting January 1, 2016) with accumulation and discounting at 3 month intervals. The NPV for an individual OLYMPUS model is defined in Equation (3.1.1) as a function of a vector of decision parameters, $\mathbf{d}$, where index i refers to the time interval $\Delta t_i = t_i - t_{i-1}$, N_t is the total number of time intervals, d is the discount factor, $\tau = 365$ days is the time interval for discounting, and $R_j(\mathbf{d}, t_i)$ is the difference of all revenue and expenditure during the interval Δt_i for which the exact form depends on the sub-challenge. The expected NPV is approximated by the ensemble mean NPV defined in Equation (3.1.2), where $N = 50$ is the number of models in the ensemble.

$$\mathrm{NPV}_j(\mathbf{d}) = \sum_{i=1}^{N_t} \frac{R_j(\mathbf{d}, t_i)}{(1+d)^{\frac{t_i}{\tau}}} \tag{3.1.1}$$

$$\mathbb{E}[\mathrm{NPV}](\mathbf{d}) \approx \overline{\mathrm{NPV}}(\mathbf{d}) = \frac{1}{N} \sum_{j=1}^{N} \mathrm{NPV}_j(\mathbf{d}) \tag{3.1.2}$$

For the well control optimisation challenge a fixed well configuration is provided by TNO based on oil reservoir engineering principles with $R_j(\mathbf{d}, t_i)$ defined in Equation (3.1.3), where $Q_{j,op}(\mathbf{d}, t_i)$, $Q_{j,wp}(\mathbf{d}, t_i)$ and $Q_{j,wi}(\mathbf{d}, t_i)$ are the total oil production, water production and water injection volumes in time interval Δt_i under controls $\mathbf{d}$, and r_{op}, r_{wp} and r_{wi} are the corresponding oil revenue, water production and injection costs. Note that these are fixed at the values specified in Table 4.3 on page 195.

$$R_j(\mathbf{d}, t_i) = Q_{j,op}(\mathbf{d}, t_i) \cdot r_{op} - Q_{j,wp}(\mathbf{d}, t_i) \cdot r_{wp} - Q_{j,wi}(\mathbf{d}, t_i) \cdot r_{wi} \tag{3.1.3}$$

For the well placement and the full field development optimisation challenges the $R_j(\mathbf{d}, t_i)$ formula is modified to that shown in Equation (3.1.4). This includes the additional terms of $P_j(\mathbf{d}, t_i)$ and $D_j(\mathbf{d}, t_i)$ which account for the platform and drilling costs in time interval Δt_i associated with the field development strategy captured in $\mathbf{d}$. Similarly, these costs are fixed at the values specified in Table 7.2 on page 361.

$$\begin{aligned} R_j(\mathbf{d}, t_i) = {} & Q_{j,op}(\mathbf{d}, t_i) \cdot r_{op} - Q_{j,wp}(\mathbf{d}, t_i) \cdot r_{wp} - Q_{j,wi}(\mathbf{d}, t_i) \cdot r_{wi} \\ & - P_j(\mathbf{d}, t_i) - D_j(\mathbf{d}, t_i) \end{aligned} \tag{3.1.4}$$

The OLYMPUS oil reservoir model is a fictitious oil field (inspired by a virgin oil field in the North Sea) created by TNO and was specifically developed for the challenge. It has a size of 9km by 3km, and a depth of 50m, split into 16 layers for modelling purposes. The design was conceived to realistically imitate a real oil field possessing many of the features encountered in actual oil fields including: boundary and minor geological faults; two vertical zones separated by an impermeable shale layer (the top layer contains fluvial channel sands embedded in floodplain shale, whilst the bottom layer consists of alternating layers of coarse, medium and fine sands); as well as multiple types of facies (body of rock of specified characteristics) including channel sands, shale, and multiple types of sand. It is considered a medium complexity oil reservoir model. Within the TNO OLYMPUS Field Development Optimisation Challenge the main aim is to address geological uncertainty (uncertain porosity, permeability, net-to-gross and initial water saturation) for which an ensemble of 50 realisations of the OLYMPUS model are provided which were generated from a stochastic geology model. These are labelled as OLYMPUS 1 to 50. The OLYMPUS oil reservoir model is illustrated in Figure 3.1 which shows a three-dimensional map depicting the initial oil saturation for the OLYMPUS 28 realisation. Full details of the model can be found in [4].

The focus of this chapter and Chapters 4 and 5 will be well control optimisation, whilst well placement optimisation will be explored further in Chapters 6 and 7. For well control optimisation we reformulate the problem as a decision support task treating the expected NPV as a utility function with the well controls within each time interval as decision parameters.

3.1.2 Critique of the TNO OLYMPUS Field Development Optimisation Challenge

The TNO OLYMPUS Field Development Optimisation Challenge has set out a clear aim of investigating and testing methodology designed to perform optimisation under uncertainty. In particular, TNO are interested in the development of effective

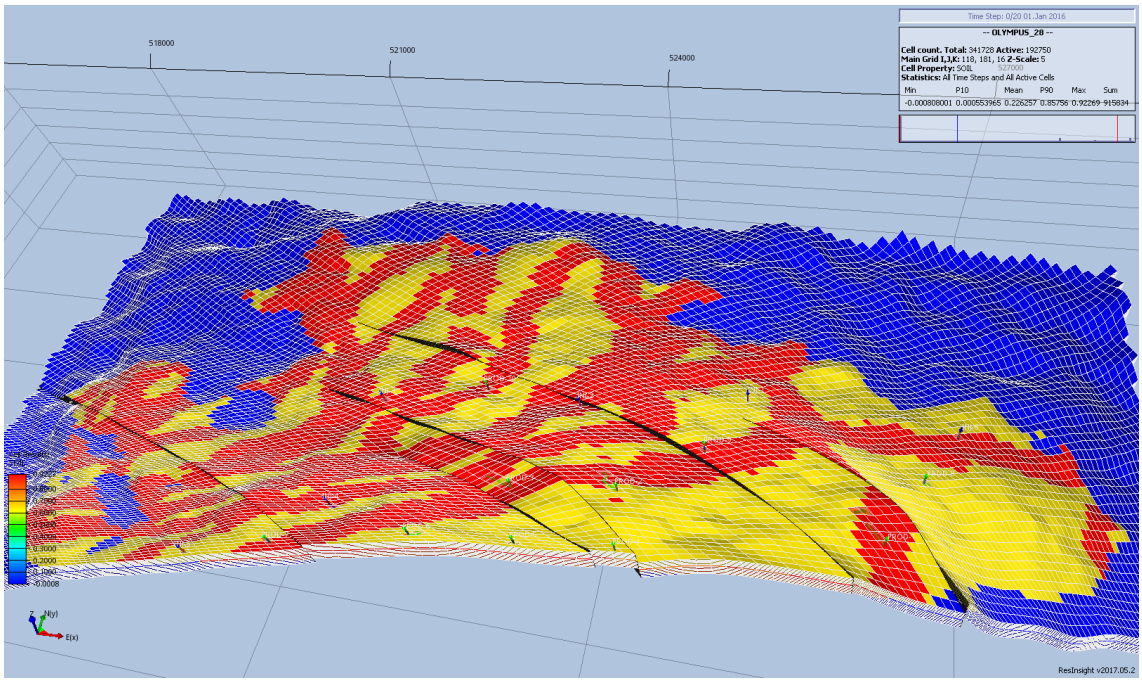


Figure 3.1: Three-dimensional map of the TNO OLYMPUS oil reservoir model depicting the initial oil saturation for the OLYMPUS 28 geological realisation. Red and yellow regions correspond to high and moderate oil saturation respectively, whilst blue regions depict water and thus possess low oil saturation. The green upward arrows and red downward arrows represent the fixed producer and injector well locations respectively within the TNO OLYMPUS Well Control Optimisation Challenge. The lines across the field depict the geological faults which shift neighbouring layers on either side. This image was generated using ResInsight [161].

ensemble optimisers. However, examination reveals numerous omissions and over-simplifications to the extent that the challenge is no longer a realistic representation of the real world process of developing green oil fields. This raises the question as to whether any of the methodology developed to directly address the challenge under its original guise, irrespective of how successful it is judged to be, will have any meaningful benefits when applied to the corresponding physical system? It appears that TNO are more interested in the mathematical methodologies versus their implications for the real world problem. In this section we highlight the main deficiencies with the challenge, discussing the ramifications and means of addressing them.

Firstly, in each of the sub-challenges there is a clear aim of searching for a single optimal choice of well control or field development strategy which maximises the ensemble mean NPV objective function as an approximation to the expected NPV. This stems from the interest in ensemble optimisers. This is potentially limiting in that it fails to recognise that multiple strategies may yield similar objective function values when compared given a comprehensive uncertainty quantification to form an accurate link with the real world process. Furthermore, focusing on a single strategy which is believed to be the global optimum is potentially dangerous due to the well-known risks associated with optimisation routines becoming trapped at local optima, a particular issue for high-dimensional parameter spaces, as is the case in the TNO OLYMPUS Challenge. Instead we propose identifying a class of near-optimal decisions up to the quantified uncertainties via an iterative decision support algorithm presented in Section 3.6.

TNO provide an ensemble of 50 geological realisations of the OLYMPUS model from an underlying stochastic geology model for which participants have no access. This is common practice within the petroleum industry to represent geological uncertainty via an ensemble of models with different underling geologies that are obtained from either a (calibrated) geological model or elicited from reservoir engineers. An initial issue is whether the ensemble is a representative sample obtained via random sampling. Within each of the field development optimisation challenges the objective function is the ensemble mean NPV which is specified to be equal to the expected NPV. Assuming that the geologies are randomly sampled, then the ensemble mean NPV is an unbiased estimator of the expected NPV, however this approach fails to account for the remaining geological uncertainty. Given the infinitely large space of possible geologies, using an ensemble of size 50 only partially represents this geological uncertainty. The ideal solution would be to sample a larger number of geologies as well as obtain knowledge of the sampling distribution of the ensemble mean NPV under a particular strategy. By the central limit theorem, this will be a normal distribution, although the mean and variance parameters must be estimated. These

investigations are not possible under the setup of the TNO OLYMPUS Challenge. A potential option is bootstrapping whilst in our application to the TNO OLYMPUS Well Control Optimisation Challenge we incorporate geological uncertainty into decision support as described in Section 4.6.2.

Another weakness of the TNO setup is the form of the objective function. For decision theoretic and decision support approaches that acknowledge and incorporate sources of uncertainty, the objective function should be re-expressed as a utility function which also encompasses the risk preferences of the decision maker(s). Note that directly interpreting the objective function as equal to the utility function results in the statement that the decision maker(s) is risk neutral; an assumption which is rarely true based on discussions with collaborators in the oil industry, with the exact level of risk aversion often depending on the size of an oil company and their financial status. This may be addressed via a sensitivity analysis to various forms of the utility function that portray different risk preferences. Moreover, within all three sub-challenges, the additional cost parameters introduced to the objective function include: oil price; water production and injection costs; discount factor; platform and drilling costs, are all assumed to be fixed over the 20 year field lifetime. This is an entirely unrealistic prior belief statement which has consequences for which strategy is identified as optimal under the original challenge formulation. Instead we acknowledge the uncertainty induced by time variable NPV cost parameters and demonstrate the quantification in Section 4.6.4, and highlight the implications in Section 4.7.3.

Furthermore, there exist numerous other sources of uncertainty not considered within the TNO OLYMPUS Challenge. At the very least these should be acknowledged to highlight the potentially large and misleading consequences of their omission. The TNO OLYMPUS Challenge is designed to obtain an optimal strategy for an ensemble of computer models rather than the real OLYMPUS oil field (if it existed); a substantial limitation when transferring the methodology developed and tested on the OLYMPUS ensemble to the problem of devising real world field

development strategies. Structural model discrepancy should therefore be incorporated to link the computer model with the real world system [65, 171]. Another form of uncertainty which is not accounted for is implementation error. This pertains to the tolerances adhered to during the execution of a well control or placement strategy, for example, the accuracy with which a control setting is input, as well as its start or changeover time compared to the plan. Further discussion is found in Section 3.5.4. Although these two forms of uncertainty are not assessed as part of the TNO OLYMPUS Well Control Optimisation Challenge, they are incorporated for the commercial application to the Emerson Jade model; for details see Sections 5.5.5 and 5.5.6 respectively. Such uncertainties and any others considered relevant are simple to incorporate within our proposed iterative decision support procedure which produces a class of decisions that are indistinguishable with respect to these uncertainties.

In the TNO problem statement it is asserted that "history matching will not be considered", as such preventing the use of historic data (for which none is provided) to learn about the underlying geology of the OLYMPUS reservoir. History matching or calibration would allow for more accurate predictions of the expected NPV under particular field development strategies leading to better decisions. This is in contrast to activities within the petroleum industry where exploratory investigations are frequently conducted providing a source of historical data for which it seems ill advised to ignore. Although not implemented in this thesis, the effects of incorporating such knowledge could be assessed by simulating a short history from the OLYMPUS model.

A related and also unrealistic stipulation of the challenge is that a well control or placement strategy should be devised for the full 20 year field lifetime at the beginning. Under the "no history matching" rule it is not permitted to use the future historical data to re-assess decisions at intermediate time points given the improved state of knowledge, thus excluding sequential decision making and the potential added value obtained from observed data from the challenge remit. Note

that sequential decision making will also require the introduction of observational errors. This is discussed further in Section 3.4.

3.1.3 Petroleum Industry Approaches to Field Development Optimisation

Developing well control and well placement strategies which maximise the profitability over the field lifetime is a major problem for the petroleum industry. This is highlighted by the instigation of the TNO OLYMPUS Field Development Optimisation Challenge [93] which has received much attention including at the dedicated EAGE/TNO Workshop on OLYMPUS Field Development Optimization [39] as well as beyond, for example in [135, 165]. Current research within the petroleum industry has mainly focused on optimisation of field development plans and well control strategies, in many cases for an ensemble of models characterising geological uncertainty. Decisions are expressed as parameters to the simulator. This section contains a brief discussion of ensemble methods before providing an overview of the methodology developed for or applied to the field development optimisation problem.

Ensemble methods are commonly used within the petroleum industry to perform history matching to past data, but also find application to production optimisation. The Ensemble Kalman Filter (EnKF) [117] is a particularly popular ensemble method with multiple variants developed and applied to the history matching problem including: using both stochastic and deterministic updates [103]; adapted with a variance inflation or localisation [103, 155]; the iterative EnKF for inverse problems [92]; and the EnKF with Multiple Data Assimilations (EnKF-MDA) [43]. Other ensemble methods for inverse problems include; the Ensemble Smoother (ES) [43]; the Ensemble Smoother with Multiple Data Assimilations (ES-MDA) [42]; and the Ensemble Randomised Maximum Likelihood (EnRML) and batch-EnRML algorithms [41]. The EnKF as a derivative-free optimisation technique has also been applied to the field development optimisation problem where ensemble members represent differing stochastic geology realisations [21, 92], as too has the iterative EnKF [21].

The main limitations to the EnKF algorithm and its variants is the underlying linearity assumption for the relationship between the simulator parameters and the function output. In addition, there is the assumption of Gaussian errors whilst in practice users often rely on the properties obtained in the large sample size limit, as opposed to the fixed sample size [155].

Treatment of uncertainty within the petroleum industry for production optimisation is generally fairly limited with efforts mainly concentrated on addressing geological uncertainty. Most often this is characterised through an ensemble of models with differing underlying geologies, as is the case for the TNO OLYMPUS Challenge. The advent of optimisation under uncertainty via ensemble methods has permitted a greater exploration of the impact of uncertainties such as the geology and is also referred to as robust optimisation. Examples include the above mentioned EnKF and EnRML algorithms. A popular method is the Ensemble-based Optimisation scheme (EnOpt) of [21] which uses the EnKF within a closed-loop framework to propagate uncertainty in time. It is empirically shown to be fairly robust and does not require adjoint information, instead using a stochastic gradient approximation. EnOpt was further developed into the modified EnOpt algorithm in [36], and the Covariance-Matrix-Adaptation EnOpt (CMA-EnOpt) algorithm in [51] which stems from the CMA evolutionary strategy, a gradient-free optimisation method (originating within machine learning). This is an adaptive strategy in which the modifications of the covariance matrix at each step allow for exploration in the early stages, with smaller perturbations at later stages when the algorithm nears convergence. It is demonstrated that CMA-EnOpt is more robust than EnOpt to the initial choice of the covariance matrix. Moreover, [36] discuss the relationship between EnOpt and the Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm. A further advancement of robust optimisation techniques is the Stochastic Simplex Approximate Gradient (StoSAG) algorithm of [50, 52] for which there exist several variants: unsmoothed; singly- and doubly-smoothed StoSAG; and singly- and doubly-smoothed cross-covariance StoSAG. Each version makes efficient

use of gradients, however this is challenging for high-dimensional parameter spaces whilst it is shown that StoSAG does not perform well for very large ensembles. Empirical results suggest that StoSAG is generally superior to EnOpt, although no one variant consistently achieves the best outcome. It may therefore be necessary to perform multiple optimisations which further increases computational expense. All of the listed optimisation algorithms yield a single vector of decision parameters which may be non-robust to the uncertainties present; which potentially includes geological uncertainty despite the attempts to incorporate it. This is due to a finite ensemble size as well as underlying statistical questions regarding how the geologies have been sampled or generated. Comparisons of such routines as in [165] also highlight how such optimisation algorithms frequently become stuck at different local optima and are sensitive to their starting location and tuning parameters. Consequently it is very difficult to be certain that the global optimum strategy has been identified, especially in high-dimensional settings. Further complications arise due to the computational costs of such optimisation algorithms which directly use the computer model ensemble, but were not designed for the increasingly expensive simulators used within the petroleum industry.

Gradient- or adjoint-based optimisation routines have also been applied to well control optimisation such as in [97] with examples including steepest descent, conjugate gradient and Sequential Quadratic Programming (SQP). However, these are computationally expensive and limited to low-dimensional decision parameter spaces. Moreover, derivative free stochastic algorithms find application both to well control but also well placement optimisation. Examples include: Genetic Algorithm (GA), Very Fast Simulated Annealing (VFSA) and Particle Swarm Optimisation (PSO) [165]. These approaches are implemented for both sequential and simultaneous optimisation, whilst robustness with respect to geological uncertainty is assessed using what is termed the NPV S-curve; an empirical cumulative distribution function for the NPV. The joint well placement and well control optimisation is substantially more complex to the two separate problems which is largely due to the considerably

higher dimensional decision parameter space making it more computationally challenging. The standard approach is to perform these tasks sequentially, first devising a well placement strategy before formulating an optimal well control plan in what is referred to as a nested well control optimisation routine. One of the key research questions of the TNO OLYMPUS Challenge is to determine what additional value can be achieved through joint optimisation [93]. An example of the sequential approach is presented in [12] where well placement optimisation is performed using a pattern search via a combination of the Hooke-Jeeves Direct Search (HJDS), Generalised Pattern Search (GPS) and a Hybrid Optimization Parallel Search Package (HOPSPACK), before well control optimisation via a SQP algorithm, specifically SNOPT (Sparse Nonlinear OPTimiser), a gradient-based algorithm which are efficiently computed using adjoints.

At the EAGE/TNO Workshop on OLYMPUS Field Development Optimization, various optimisation algorithms including many of those stated above were implemented for each part of the TNO OLYMPUS Challenge [39]. For the TNO OLYMPUS Well Control Optimisation Challenge methods in decreasing order of their reported ensemble mean NPV included:

1. Singly-Smoothed Cross-Covariance StoSAG (ss-cc-StoSAG);
2. GA using Conditional Value at Risk (CVaR) and sequential optimisation;
3. Partitioned Global PSO using 18 partitions (one for each well) and a swarm size of 30;
4. MATLAB Reservoir Simulation Toolbox (MRST) inbuilt function for performing gradient based optimisation using adjoints applied to an ensemble of models;
5. Steepest Ascent using StoSAG;
6. $(\mu + \lambda)$ Evolutionary Strategy (ES).

Approaches to the TNO OLYMPUS Well Placement Optimisation Challenge, also in decreasing order of the reported ensemble mean NPV included:

1. Simultaneous optimisation using GA for Numerical Optimisation of Constraint Problems (Genocop III);
2. Mixed-Integer GA;
3. A sequential (non-automated) approach using GA;
4. Black Hole PSO (BHPSO) with 10 particles per generation;
5. Sequential optimisation first using the Line Search Derivative-Free (LSDF) algorithm for well location optimisation, followed by EnOpt and RMS for joint well path and drilling order optimisation, and then Equinor's internal optimisation tool to optimise drilling order;
6. $(\mu + \lambda)$ ES with the platform location separately optimised;
7. 3D well pattern design process via graphical methods and incorporating reservoir engineering principles;
8. Partitioned Global PSO using 15 partitions (up to two wells per partition), 16 variables per partition, and a swarm size of 30.

Note that some participants also incorporated techniques to obtain a reduced ensemble size with the aim of decreasing computational expense. Due to the complexity of the full field development optimisation challenge, few participants attempted this with most opting for a sequential approach using their devised well placement strategy before optimising the well controls. The only attempt at simultaneous optimisation was via the $(\mu + \lambda)$ ES. This returned a smaller ensemble mean NPV compared to sequential application of the same algorithm, thus highlighting the non-robustness of such approaches when applied to very high-dimensional problems.

The various optimisation techniques applied to the TNO OLYMPUS Challenge reported distinct "optimal" solutions, yet with different ensemble mean NPVs. It is therefore questionable whether any of these strategies are the global optimum, although it is plausible that most are close to the true maximum expected NPV, but are likely to be indistinguishable once uncertainty is taken into consideration, as discussed in Section 3.1.2. In addition, it is noteworthy that the achieved ensemble mean NPVs were not substantially better than those obtained using reservoir

engineering insight, particularly for the more complex well placement optimisation challenge. Moreover, in both instances the method with the highest reported ensemble mean NPV used substantially more simulations for very little increase in the objective function versus all other approaches. This suggests that good results when accounting for uncertainty can be achieved for a much lower computational expense.

Across the petroleum industry a large number of optimisation procedures have been developed and applied to the field development problem, however none of these methods fully explore the decision parameter space and also require a large number of often computationally expensive simulations. Numerous comparisons have shown that different algorithms tend to perform better on certain styles or even specific problems, hence whether one has attained a strategy close to the optimal solution seems to be sensitive to the exact problem. Also, results may be sensitive to the algorithm setup including any hyperparameters and conditioning. For real world problems it is not usually practical to apply a large number of methods and select the solution which returns the highest value of the objective function. As noted for the TNO OLYMPUS Field Development Optimisation Challenge, comparisons between optimisation routines show incremental improvements in the objective function raising questions as to whether any increases are substantial with respect to the underlying uncertainties.

The above described methodology are for handling geological uncertainty which is of principal concern in the petroleum industry, however there exist many other sources of uncertainty for which failure to account for may have potentially large implications. Economic uncertainty, referred to in this thesis as NPV cost parameter uncertainty, is acknowledged in some papers, but often given an overly simplified treatment [158] where oil price is modelled using an AutoRegressive-Moving-Average (ARMA) model. Optimisation is performed with respect to various objective functions including: Mean Optimisation (MO); Mean-Variance Optimisation (MVO); Worst-Case Optimisation (WCO); and Conditional Value-at-Risk (CVaR) Optimisation. This is extended to the joint treatment of geological and oil price uncertainty

(although no other forms of economic uncertainty) in [157] demonstrating two robust optimisation strategies that balance long- and short-term economic objectives as a form of Multi-Objective Optimisation (MOO). These are: a robust hierarchical optimisation technique, a two stage process using a tolerance compared with the maximum field lifetime NPV; and a Robust Weighted-Sum approach yielding a Pareto curve of solutions with decisions selected based on risk preferences. Another consideration is the uncertainty in future development plans, for example: potential drilling of future wells, such as in [97] where well control optimisation is performed using a quasi-Newton method with the gradient generated with an adjoint model (approximate Hessian calculated from the BFGS algorithm), and joint optimisation using the SPSA algorithm. The concept of a quality map is used to aid the identification of promising reservoir regions or locations at which to drill new wells. Throughout the literature there is no comprehensive framework for performing optimisation or decision support which simultaneously incorporates all sources of uncertainty.

3.2 Efficient Geological Ensemble Subsampling

Ensembles of computer models are frequently used throughout the petroleum industry to characterise some forms of uncertainty. A prevalent example is the underlying geology of the field being modelled to reflect the views of the geologist(s). In the TNO OLYMPUS Field Development Optimisation Challenge this consists of 50 versions of the OLYMPUS model realised from an underlying stochastic geology model. Using every model within the ensemble to learn about the real physical system therefore requires a greater number of simulations than for a single model, although this does provide some insight into the effect of the uncertainty represented. However, larger ensembles place a greater strain on computational resources, especially when simulations already take a long time to evaluate. In many analyses the outputs over the ensemble are combined into a single output such as through averaging and is termed the ensemble mean. Whilst such quantities are easier to

analyse and use, the averaging process reduces the benefits of starting with an ensemble by collapsing the uncertainty onto a single value. Consequently it is desirable to establish a subset of the ensemble to use as a surrogate, whilst acknowledging any reduction in information gained from the simulations.

In this section we present an approach for identifying a representative subset of models from the ensemble that includes an initial graphical investigation in Section 3.2.1 combined with a method of prediction and uncertainty quantification in Section 3.2.2. Collectively these are referred to as Efficient Geological Ensemble Subsampling (EGES) techniques which constitute a novel application within the petroleum industry. This is useful for both decision support and optimisation since it permits greater exploration of the decision parameter space for the same or lower computational cost. For example, in well control optimisation a larger number of potential future control strategies can be simulated. This is particularly important in the high-dimensional settings in which these problems are situated.

3.2.1 Efficient Geological Ensemble Subsampling Graphical Investigations

In the process of selecting an appropriate subset of models, and in general for the analysis of both an individual or an ensemble of complex computer models, it is useful to possess a small initial collection of exploratory simulations using all models in the ensemble. This is referred to as a wave 0 of simulations with designs often constructed using a maximin Latin hypercube or other method described in Section 2.3. Wave 0 simulations are then used to asses how well a given subset of models represent the ensemble mean of certain outputs. For the field development problem, a suitable choice of output may be the ensemble mean NPV, whilst other quantities of interest may include the constituent parts of NPV formula in Equation (3.1.1) including: Field Oil Production Total (FOPT); Field Water Production Total (FWPT); Field Water Injection Total (FWIT) within a time interval; as well as the further decomposition into the corresponding well quantities: WOPT, WWPT

and WWIT.

An initial graphical investigation of plots of the ensemble mean for output(s) of interest versus that obtained for each of the individual models provides insight into patterns between them. A strong linear correlation would suggest that an individual model may be a good representative for the ensemble mean. Note that these are two-dimensional plots which are unable to capture the interaction between multiple models' outputs, thus missing where two or more models are jointly able to characterise the ensemble mean, often to a better extent than any one individual model. An extension is to examine 2-dimensional contour plots. For large ensembles, this can be used as a screening method to identify a preliminary subset of the ensemble for further analysis and on which to perform the next step of this EGES technique where it may be computationally impractical from both a time or memory perspective to perform exhaustive investigations over all possible subsets.

3.2.2 Efficient Geological Ensemble Subsampling Linear Modelling

The aim is to select a representative subset of the ensemble for which linear models provide a fast and effective tool as well as a method for predicting the ensemble mean from the output of individual models with a quantification of the induced uncertainty. For an ensemble of N models the aim is to select a subset of $N_{\text{EGES}} < N$ (to be determined) models which characterise output $f(\cdot)$, for example the NPV, where $\bar{f}(\cdot)$ and $f^{(i_k)}(\cdot)$ represent the output ensemble mean and model i_k output respectively. Note that we focus on the ensemble mean NPV, $\bar{f}(\mathbf{d}) = \overline{\text{NPV}}(\mathbf{d})$, as this is the primary focus in the TNO OLYMPUS Challenge, whilst this is linked to the expected NPV, $U(\mathbf{d}) = \mathbb{E}[\text{NPV}](\mathbf{d})$, in Sections 3.5.3 and 3.6.2.

The ensemble mean is a linear combination of the individual models' output, hence an affine linear transformation of a subset of models is expected to produce a good approximation. We therefore propose a linear model of the form in Equation (3.2.1), where α_{EGES} and $\beta_{k,\text{EGES}}$ are unknown regression coefficients to be

estimated.

$$\bar{f}(\mathbf{d}) = \alpha_{\text{EGES}} + \sum_{k=1}^{N_{\text{EGES}}} \beta_{k,\text{EGES}} f^{(i_k)}(\mathbf{d}) + \varepsilon_{\text{EGES}}(\mathbf{d})$$
$$i_k \in \{1, \ldots, N\} \text{ distinct}, \; N_{\text{EGES}} < N \tag{3.2.1}$$

Model selection criteria such as those discussed in Appendix A can be used to compare the linear models and hence choose the most appropriate value for N_{EGES} and the subset of models as predictors. It is found that AIC is suitable for this task, whilst BIC often places an overly strong penalty on the number of linear model terms. The adjusted R^2 value provides a measure of the variation explained by the regression whilst penalising using an increased number of original ensemble members (predictors in the linear model).

In situations where N is relatively small and a prior interval for N_{EGES} is given, it is computationally feasible to exhaustively explore all possible models. Otherwise stepwise selection starting from a linear model containing only an intercept ($N_{\text{EGES}} = 0$) can be performed, potentially with an upper bound on N_{EGES}, to construct a subset of models. As discussed above in Section 3.2.1, another option is to preselect a larger preliminary subset of models based on the graphical analysis and use only these in forming the linear model. This process can be repeated for multiple outputs and assessed for agreement in terms of the models identified. The final selected linear model(s) is referred to as the EGES linear model(s), with a corresponding subset of the ensemble.

At later stages in an analysis the choice of models (and hence EGES linear model) may be revised or expanded if increased accuracy is required. This scenario naturally occurs within iterative procedures such as history matching or the decision support algorithm presented in Section 3.6, where the model or decision parameter space has been reduced in volume at each wave. An alternative subset of the ensemble may offer a superior characterisation over this reduced parameter space versus the initial EGES linear model for the full parameter space.

The presented EGES linear modelling technique for identifying a representative

subset is related to the research on second-order exchangeability of ensemble members in [150] leading to coexchangeability between the output of individual models and the actual system. This is used to establish a link between: the output of individual models; a common "representative simulator", in our formulation this is interpreted as the ensemble mean simulator; the output for the actual system meaning the real world expected NPV with respect to all possible geological configurations; as well as any system observations. Note that our approach does not explicitly assume second-order exchangeability of ensemble members.

3.3 Hierarchical Emulators Exploiting Known Simulator Behaviour

Bayesian emulators were introduced in Section 2.4 as a means of statistically approximating the output of a computer model for as yet unevaluated parameter settings and including a statement of the uncertainty. This approach is based on the assumption that simulators behave like a "black-box" for which the user possesses no insight of the structure or links between individual processes, as well as no knowledge of the model behaviour for specific outputs at a given parameter setting prior to its evaluation. Whilst this makes the above described methodology generalisable, it potentially limits the emulator accuracy in situations where the user has an understanding of how certain outputs are linked. If such prior information is available, then not using it has implications within decision support where more accurate emulators could be used to obtain a smaller decision support region for a lower computational expense.

In this section we present a new modified form of hierarchical emulation which is designed to open the "black-box" and exploit known structures between the simulator outputs and any functions thereof. This is motivated by the TNO OLYMPUS Well Control Optimisation Challenge with discussion in Section 3.3.1 with structured emulation of outputs of a commonly occurring specific form described in Section 3.3.2. In

the context of the well control optimisation problem we introduce an approximation to the NPV for an individual model for which emulation methodology is presented in Section 3.3.3 before linking to the exact NPV in Section 3.3.4. The EGES techniques described in Section 3.2 are used to combine the emulation output for a subset of ensemble members to emulate the ensemble mean NPV in Section 3.3.5. The hierarchical emulator construction is summarised in Section 3.3.6.

3.3.1 Motivation

Many scientific quantities of interest are functions of several simulator outputs. These may be emulated directly using the aforementioned methodology, however an alternative is to decompose the quantity into its constituent parts and emulate these separately before combining by some means to emulate the quantity. This has the possibility of achieving superior accuracy by exploiting structures between the outputs. Such a scenario occurs naturally within oil field development optimisation where the aim is to maximise the expected NPV objective function. For the TNO OLYMPUS Well Control Optimisation Challenge, the formulae given in Equations (3.1.1) and (3.1.2) provide a decomposition into the FOPT, FWPT and FWIT within each discounting time interval, which in turn may be further split by well. Collectively these quantities will be referred to as the NPV constituents or contributors. This suggests the development of a hierarchical emulation framework for the purpose of emulating the ensemble mean or expected NPV.

The first step is to emulate each of the NPV constituents for each model. A standard approach would be to construct a Bayes linear or Gaussian process emulator, as discussed in Sections 2.4.4 and 2.4.5 respectively, however both methods ignore any partial knowledge of a NPV constituent's behaviour for a given decision parameter setting prior to the evaluation of the simulator. For our motivating application using target production and injection rates as well control decision parameters, it would be expected that WOPT within a time interval should be equal to the product of the corresponding target production rate and the length of the interval. It is

observed that this behaviour occurs up to an unknown target production rate where constraints such as Bottom Hole Pressure (BHP) come into effect and thus prevent the assigned target rate from being achieved for the full duration of the control interval. Moreover, there is the interaction with other wells and their target rates within the field and the past production and injection history which makes the point of this behavioural change difficult to exactly determine. Consequently WOPT initially follows a linear trend up to a change point beyond which a ceiling or plateau is reached. An illustration of such behaviour can be seen in Figure 3.2 on page 108. Similar behaviour is also observed for WWIT within a control interval with respect to the corresponding target injection rate for that interval. This raises the question of how best to use this extra knowledge to improve the accuracy of an emulator for simulator output that exhibit such behaviour and subsequently aid decision support, and is the subject of Section 3.3.2 where change points, extrapolation cut-offs, and a modified form of the emulator are introduced. Note that such structured behaviour is not observed for WWPT within a control interval and hence standard emulation methodology is applied.

Once all of the NPV constituents have been emulated for a model in the ensemble, it is necessary to combine them to emulate the NPV. It is noted that within our approach to decision support that the control intervals are formed by the amalgamation of multiple consecutive discounting intervals with further discussion in relation to an application in Section 4.1.1. An intermediate step to the hierarchical emulation structure is therefore introduced in Section 3.3.3 based on the linear combination via an approximate NPV formula of the NPV constituent emulators. To complete the emulation of the NPV for an individual model, an intuitive method of linking the approximate and exact NPV is then discussed in Section 3.3.4. For multiple sub-selected models, the EGES linear model is then used in Section 3.3.5 to combine each of the emulated NPVs by model. Note that the approach presented in this discussion for this motivation is specific to the well control optimisation problem, but the concept is easily adapted to other scenarios where quantities of interest can

be expressed as functions of multiple simulator outputs, each of which potentially exhibits some behavioural structure with respect to a or some of the (decision) parameters.

3.3.2 Emulators Exploiting Known Simulator Behaviour for NPV Constituents

The emulation methodology presented in this section is motivated by the behavioural form observed for WOPT and WWIT outputs within control intervals, as observed in Figure 3.2 and discussed in Section 3.3.1. For each output, this requires the introduction of a change point and an extrapolation cut-off which are used within a modified form of emulation with the decision parameter space split into three regions based on the modes of behaviour.

Change Points

The observed structured behaviour for a particular NPV constituent is with respect to a corresponding decision parameter which affects that particular control interval. For example, for the WOPTPROD2_20180101 output, the corresponding target rate is prod_2_2016_01, as illustrated in Figure 3.2. Theoretically there are two distinct regions of behaviour termed the slope and plateau regions for an output. Within the slope region the target rate decision parameter is adhered to for the full duration of the control interval, whilst within the plateau region this is not the case. The slope therefore possesses a known gradient which is equal to the length of the control interval (in days). In Figure 3.2 the slope is equal to $\Delta t_{20180101} = 731$ days. The exact decision parameter value of the transition from slope to plateau region is designated the change point. This is unknown and is likely to depend on all other decision parameter values. Given only a finite number of simulations it is impossible to determine the exact location of the change point since this would require investigations for every configuration of all other decision parameters. Consequently in practical application there are three distinct regions of behaviour: the slope and

plateau regions separated by an additional uncertain region believed to contain the unknown change point. For a simulator output $f_i(\cdot)$ which follows such behaviour the left-hand slope is precisely known up to a tolerance $\delta_i \geq 0$; this can be used to estimate the mean change point location given a collection of simulations from design $\mathcal{D}$. In the well control optimisation problem, let d_{jk,t_i} be the decision parameter which directly effects NPV constituent $f_i(\cdot)$ with change point c_{jk,t_i}. The index i is condensed notation for the indices tuple (jk, t_i), where $j \in \{P, I\}$ and k refer to the well type (P producer, I injector) and number respectively, whilst t_i is the control interval start date. The gradient of the slope region is equal to the length of the control interval, Δt_i.

There is an implicit dependence of c_{jk,t_i} on all other decision parameters and therefore we introduce methods of estimating an upper and lower bound. A conservative estimate for the change point upper bound, c^u_{jk,t_i}, is defined in Equation (3.3.1), where $f_{i,\max} = \max_{\mathbf{d}\in\mathcal{D}} f_i(\mathbf{d})$, and $\delta_{i,u} \geq 0$ is a tolerance included for numerical stability and to ensure that an upper bound is obtained.

$$c^u_{jk,t_i} = \min_{d_{jk,t_i}} \{d_{jk,t_i} \mid d_{jk,t_i} \cdot \Delta t_i \geq f_{i,\max} + \delta_{i,u}\} \tag{3.3.1}$$

This is the smallest value of the corresponding decision parameter for which if the target was achieved for the entire control interval, then the simulator output would exceed the largest observed value (plus a tolerance) from the collection of simulations. Note that the decision parameter is untransformed and so is on its original scale.

An estimate for the change point lower bound is defined in Equation (3.3.2), where $f_{i,\text{diff}}(\mathbf{d}) = f_{i,\max}(\mathbf{d}) - f_i(\mathbf{d}) = d_{jk,t_i} \cdot \Delta t_i - f_i(\mathbf{d})$, and $\delta_{i,l} \geq 0$ is another tolerance, again used for numerical stability in the comparison of the simulated and the theoretical maximum values.

$$\begin{aligned} c^l_{jk,t} = \frac{1}{2} \Bigg(& \arg\min_{d_{jk,t_i}|\mathbf{d}\in\mathcal{D}} \{f_i(\mathbf{d}) < d_{jk,t_i} \cdot \Delta t_i - \delta_{i,l}\} \\ & + \arg\max_{d_{jk,t_i}|\mathbf{d}\in\mathcal{D}} \left\{ d_{jk,t_i} < \arg\min_{d_{jk,t_i}|\mathbf{d}\in\mathcal{D}} \{f_i(\mathbf{d}) < d_{jk,t_i} \cdot \Delta t_i - \delta_{i,l}\} \right\} \Bigg) \end{aligned} \tag{3.3.2}$$

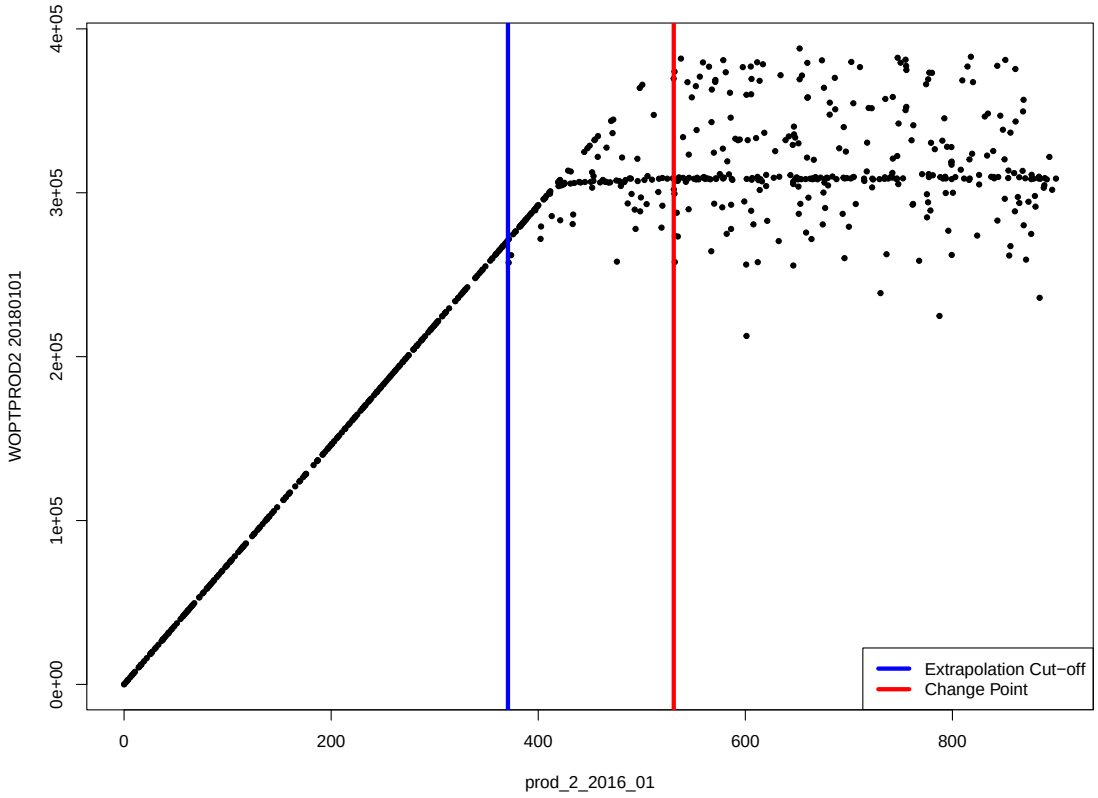


Figure 3.2: OLYMPUS 25 WOPT for producer well 2 during the first two years (ending 01/01/2018) versus the corresponding target production rate, prod_2_2016_01. For small values of prod_2_2016_01 the target is achieved resulting in a perfectly defined linear behaviour up to a change point due to other controls and BHP constraints beyond which the WOPT plateaus as a maximum threshold on the production rate is achieved. The vertical blue and red lines denote the extrapolation cut-off and change point upper bounds respectively.

This is the midpoint between the first simulation decision parameter setting which is not on the slope and hence has $f_{i,\text{diff}}(\mathbf{d}) > \delta_{i,l}$ (first term in Equation (3.3.2)), and the decision parameter setting with the largest value of d_{jk,t_i} which is less than this first departure point previously obtained (second term in Equation (3.3.2)). This is illustrated in Figure 3.3 for the output WOPTPROD2_20180101 versus the target rate prod_2_2016_01. The vertical blue line denotes $c^l_{P2,2016}$ and the red line represents the slope upper bound which is attained if prod_2_2016_01 is adhered to for the full control interval. The first point not on the slope is coloured green and

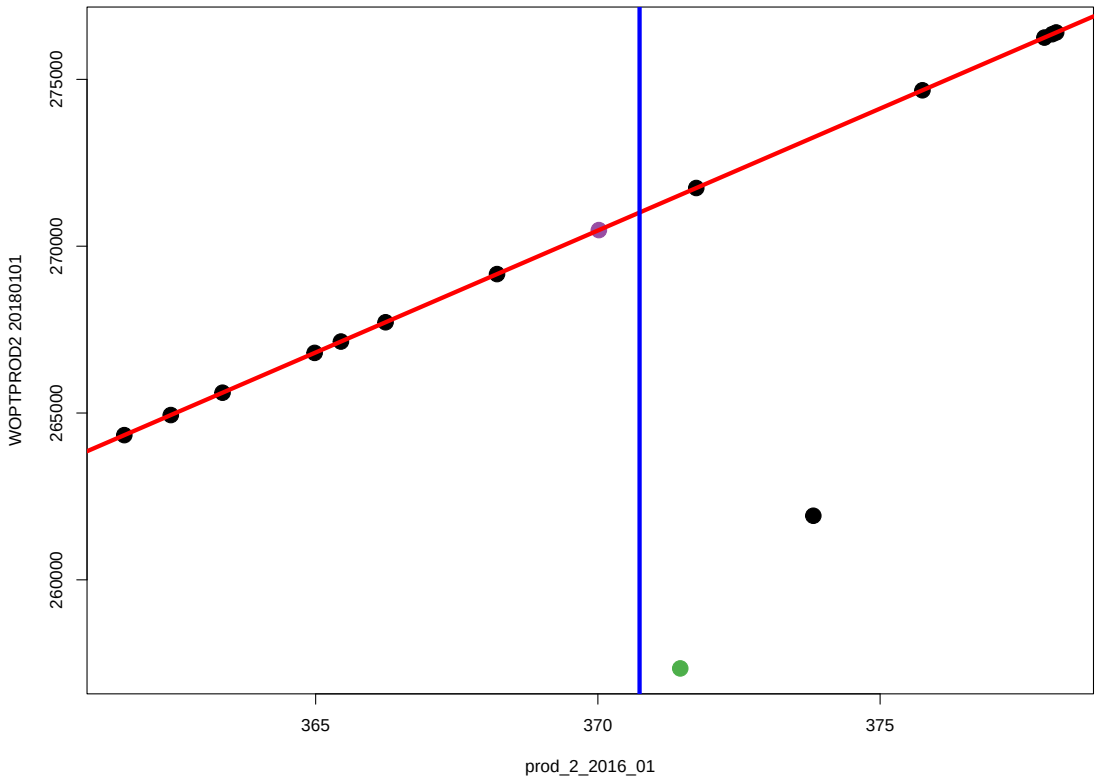


Figure 3.3: OLYMPUS 25 WOPT for producer well 2 during the first two years (ending 01/01/2018) versus the corresponding target production rate, prod_2_2016_01. The plot is zoomed in to focus around the change point lower bound, $c^l_{P2,2016}$, computed using Equation (3.3.2), and denoted by the vertical blue line. The red line represents the slope upper bound which is attained if the target production rate is adhered to for the full control interval. It is shown that $c^l_{P2,2016}$ is the midpoint of the first point not on the slope coloured green, and preceding point which is on the slope coloured magenta.

corresponds to the first term in Equation (3.3.2), whilst the preceding point, which lies on the slope up to a tolerance $\delta_{i,l}$, is coloured magenta.

Extrapolation Cut-offs

The two distinct modes of behaviour for the NPV constituents suggests that the emulator be fitted piecewise using a combination of the more accurate knowledge in the slope region, and the less well understood behaviour in the plateau region.

Noting the uncertainty in the change point location, for the plateau region this leads to the idea of fitting an emulator based only on data which is almost certainly on plateau using the estimate of the change point upper bound. For $f_i(\cdot)$ this is design points with $d_{jk,t_i} \geq c^u_{jk,t_i}$. In order to connect these two regions and learn more about the intermediate space between the slope and plateau requires an extrapolation of the emulator for the plateau region, further described below. During the development of the structured emulation methodology, it was observed that this extrapolation beyond the fitting region resulted in diagnostics issues. Problems arise when there are relatively few fitting points and the slope represents a large proportion of space which initially necessitated large extrapolations far beyond the known plateau region, as is often the case for WOPT during early time control intervals and WWIT during late time control intervals.

An extrapolation cut-off is introduced in order to combat this problem, denoted b_{jk,t_i}, beyond which the emulator should not be extrapolated, which for a simulator output $f_i(\mathbf{d})$ is defined with respect to the same decision parameter d_{jk,t_i} as the change point. This leads to the decision space being split into three distinct regions with the first being a definite slope region where $d_{jk,t_i} < b_{jk,t_i}$ whilst the plateau region remains where $d_{jk,t_i} \geq c_{jk,t_i}$. The intermediate region where $b_{jk,t_i} \leq d_{jk,t_i} < c_{jk,t_i}$ is an uncertain region in which some points may follow the slope and others the plateau modes of behaviour.

The estimation of an extrapolation cut-off is a trade-off between having an excessively cautious small value which fails to alleviate some of the issues that motivate its incorporation, whilst too large values risk points being wrongly classified as on the slope. A suitable and sufficiently conservative approach is to select the extrapolation cut-off to be equal to the change point lower bound, then $b_{jk,t_i} = c^l_{jk,t_i}$, as defined in Equation (3.3.2). Since this is based on the location of the first point to not fall on the slope, it is guaranteed that the smallest value of d_{jk,t_i} for which the simulated data does not lie on the slope will exceed b_{jk,t_i}. Figure 3.2 illustrates the three regions obtained by splitting WOPTPROD2_20180101 versus

prod_2_2016_01 at $b_{P2,2016} = c^l_{P2,2016}$ and $c^u_{P2,2016}$ denoted by the vertical blue and red lines respectively.

Structured Emulation

A method of structured emulation to accurately capture the behaviour of the NPV constituents with respect to a corresponding decision parameter is presented here and is based on the above justification for splitting the simulator output into three regions using an estimate for the extrapolation cut-off and change point. In addition, it is known that it is impossible to exceed a target rate and hence there is a theoretical maximum which can be attained by such outputs, up to a numerical tolerance. Truncation is therefore introduced within the emulator construction in order to impose this constraint. Two versions have been developed and implemented with the main distinction being how this truncation within the uncertain region between the slope and plateau behaviour is handled. The first involves an upper truncation only and is applied in Section 4.5 to the TNO OLYMPUS Well Control Optimisation Challenge, whilst in the second version this is evolved to use a two-sided truncation which is employed in Sections 5.4 and 5.8 for the Emerson Jade model application.

A structured emulator for output which exhibits this form of behaviour is constructed in several steps. Firstly, for (NPV constituent) simulator output $f_i(\mathbf{d})$ the structured emulation approach begins by fitting a preliminary Bayes linear emulator of the form shown in Equation (2.4.1) with the aim of representing the behaviour in the plateau region. Traditionally all available simulations would be utilised in the Bayes linear adjustment step; instead a sub-collection of design parameter vectors, $\mathcal{D}'_i = \{\mathbf{d} \mid \mathbf{d} \in \mathcal{D},\ d_{jk,t_i} > c^u_{jk,t_i}\}$, with corresponding simulator output $\mathbf{F}'_i = \{f_i(\mathbf{d}) \mid \mathbf{d} \in \mathcal{D}'_i\}$ are used. Moreover, preliminary emulation is only applied to $\mathbf{d}$ which satisfy $d_{jk,t_i} \geq b_{jk,t_i}$. By construction, all parameter settings in $\mathcal{D}'_i$ do not adhere to the target rate and hence are in the plateau, thus providing reliable information on which to construct this part of the emulator. Note that there may exist simulated parameter settings with $b_{jk,t_i} \leq d_{jk,t_i} < c^u_{jk,t_i}$ which are not on the

slope, although these are more difficult to accurately determine and less reliable for use in fitting this preliminary emulator. The Bayes linear adjustment formulae are then used to update this emulator.

This preliminary emulator along with b_{jk,t_i} is then used to determine the form of the structured emulator for any new parameter setting in what is referred to as a classification step. At this point we introduce the two versions.

Structured Emulation with Upper Truncation

The final structured emulator adjusted expectation and variance using an upper truncation only is obtained via the following classification step:

- **Slope Region:** If for the preliminary emulator $d_{jk,t_i} < b_{jk,t_i}$ **or** $\mathbb{E}_{\mathbf{F}'_i}[f_i(\mathbf{d})] - 3\sqrt{\mathrm{Var}_{\mathbf{F}'_i}[f_i(\mathbf{d})]} > d_{jk,t_i} \cdot \Delta t_i$, then collapse the emulator such that for the structured emulator $\mathbb{E}_{\mathbf{F}'_i}[f_i(\mathbf{d})] = d_{jk,t_i} \cdot \Delta t_i$ with fixed maximum absolute errors of size δ_i. This corresponds to when the output is strongly believed to be in the slope region.
- **Uncertain Region around Change Point:** If for the preliminary emulator $\mathbb{E}_{\mathbf{F}'_i}[f_i(\mathbf{d})] - 3\sqrt{\mathrm{Var}_{\mathbf{F}'_i}[f_i(\mathbf{d})]} \leq d_{jk,t_i} \cdot \Delta t_i < \mathbb{E}_{\mathbf{F}'_i}[f_i(\mathbf{d})] + 3\sqrt{\mathrm{Var}_{\mathbf{F}'_i}[f_i(\mathbf{d})]}$, a truncated Gaussian process (truncated GP) emulator which is evaluated with mean and variance determined by Equations (3.3.3) and (3.3.4) respectively [99], where $\phi(\cdot)$ and $\Phi(\cdot)$ represent the probability density and cumulative distribution functions respectively of a standard normal distribution. These are computed assuming a preliminary Gaussian process emulator with posterior mean and variance, abbreviated to μ_i and σ_i^2 respectively, equal to the computed adjusted expectation and variance. The truncation bounds are $a_i = -\infty$ and $b_i = d_{jk,t_i} \cdot \Delta t_i$, with $\alpha_i = \frac{a_i - \mu_i}{\sigma_i} = -\infty$, and $\beta_i = \frac{b_i - \mu_i}{\sigma_i}$. This form of emulation is used in the uncertain region around the true location of the change point.
- **Plateau Region:** In all other cases where $\mathbb{E}_{\mathbf{F}'_i}[f_i(\mathbf{d})] + 3\sqrt{\mathrm{Var}_{\mathbf{F}'_i}[f_i(\mathbf{d})]} \leq d_{jk,t_i} \cdot \Delta t_i$, the preliminary emulator output is used which corresponds to when

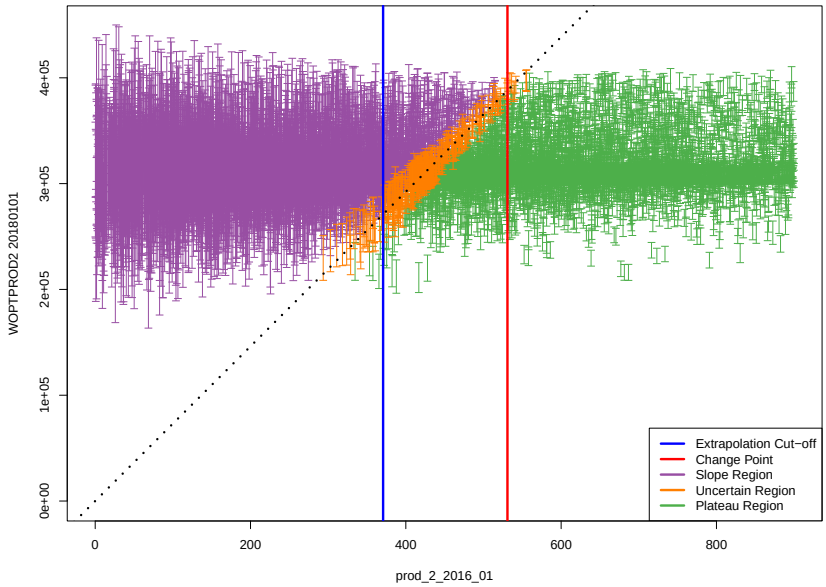
the output is strongly believed to be in the plateau region.

$$\mathbb{E}_{\mathbf{F}_i'}[f_i(\mathbf{d}) \mid a_i < f_i(\mathbf{d}) < b_i] = \mu_i + \sigma \frac{\phi(\alpha_i) - \phi(\beta_i)}{\Phi(\beta_i) - \Phi(\alpha_i)} \tag{3.3.3}$$

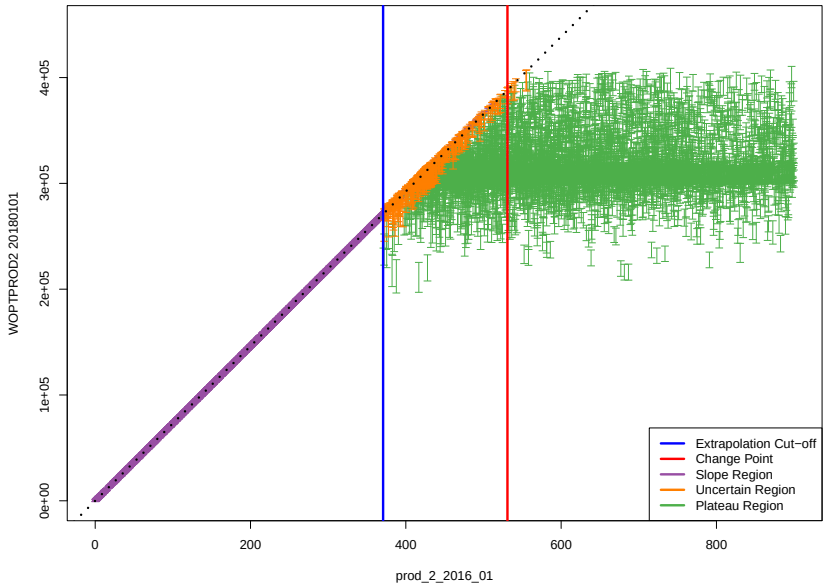
$$\mathrm{Var}_{\mathbf{F}_i'}[f_i(\mathbf{d}) \mid a_i < f_i(\mathbf{d}) < b_i] = \sigma_i^2 \left[1 + \frac{\alpha_i\phi(\alpha_i) - \beta_i\phi(\beta_i)}{\Phi(\beta_i) - \Phi(\alpha_i)} - \left(\frac{\phi(\alpha_i) - \phi(\beta_i)}{\Phi(\beta_i) - \Phi(\alpha_i)}\right)^2\right] \tag{3.3.4}$$

Imposing a truncation upper bound based on a comparison between the preliminary emulator 3-sigma credible interval and the theoretical maximum due to the effective target rate for a particular NPV constituent has the desired effect of ensuring that this physical constraint is not violated by the emulator. This is illustrated in Figure 3.4 for OLYMPUS 25 WOPT for producer well 2 within the first control interval (ending 01/01/2018) versus prod_2_2016_01. First, a preliminary Bayes linear emulator is fitted using only the subset of the wave 1 simulations for which $d_{P2,2016} \geq c^u_{P2,2016}$, as seen in Figure 3.4a. Note that the vertical blue and red lines are situated at $b_{P2,2016}$ and $c^u_{P2,2016}$ respectively, whilst the black dotted line represents the theoretical maximum slope. It is observed that extrapolating the emulator at $\mathbf{d}$ with $d_{P2,2016} < c^u_{P2,2016}$ yields increasingly wide credible intervals with distance below $c^u_{P2,2016}$.

The structured emulation methodology using an upper truncation described above is then applied to obtain Figure 3.4b. Within the slope region where the preliminary emulator credible interval is entirely above this constraint; this is characterised by a collapse onto the constraint slope. This is shown by the purple credible intervals in Figure 3.4 and includes all parameter vectors with $d_{P2,2016} < b_{P2,2016}$, as well as some cases with $d_{P2,2016} \geq b_{P2,2016}$ in which the credible interval lower bound exceeds the slope. In all instances, the credible interval width is very narrow characterising the strong beliefs that these parameter vectors will result in simulations that are on the slope. In the uncertain region this is handled by a truncated GP to reflect the uncertainty in whether the model output is actually on the slope or relatively close, potentially due to a target rate being achieved for a large proportion



(a) Preliminary Bayes linear emulator predictive CI versus prod_2_2016_01.



(b) Structured emulator with upper truncation predictive CI versus prod_2_2016_01.

Figure 3.4: OLYMPUS 25 wave 1 WOPT for producer well 2 during the first two years (ending 01/01/2018) versus the corresponding target production rate, prod_2_2016_01. The top plot shows the wave 1 preliminary Bayes linear emulator predictive 3-sigma credible interval (CI) fitted using only the simulations with $d_{P2,2016} \geq c^u_{P2,2016}$. This is used within the structured emulation algorithm imposing the upper truncation due to the slope (black dotted line) with the CI shown in the bottom plot. The vertical blue and red lines are situated at $b_{P2,2016}$ and $c^u_{P2,2016}$ respectively. The purple, orange and green CI correspond to points in the slope, uncertain and plateau regions respectively.

of the control interval. This is seen by the orange credible intervals in Figure 3.4 for which all points lie close to the black dotted slope line with much narrower credible intervals than the preliminary Bayes linear emulator. For the plateau region the preliminary emulator credible interval is well below this slope and based on arguments using Pukelsheim's 3-sigma rule it is not necessary to impose a truncation due to the very small probability that an emulator realisation actually exceeds this physical constraint. This corresponds to the green credible intervals in Figure 3.4 which are unchanged between the two plots. Alternative width credible intervals may be used depending on the level of conservativeness desired within an analysis with justification based on the Vysochanskij-Petunin inequality [181].

Structured Emulation with Two-Sided Truncation

The above methodology is further developed based on observations during the application to the Emerson Jade model in Chapter 5 to obtain a final structured emulator with a two-sided truncation. A lower truncation a_i is introduced leading to the modified classification step as follows:

- **Slope Region:** If for the preliminary emulator $d_{jk,t_i} < b_{jk,t_i}$ **or** $\mathbb{E}_{\mathbf{F}_i'}[f_i(\mathbf{d})] - 3\sqrt{\mathrm{Var}_{\mathbf{F}_i'}[f_i(\mathbf{d})]} > d_{jk,t_i} \cdot \Delta t_i$, then collapse the emulator such that for the structured emulator $\mathbb{E}_{\mathbf{F}_i'}[f_i(\mathbf{d})] = d_{jk,t_i} \cdot \Delta t_i$ with fixed maximum absolute errors of size δ_i. This corresponds to when the output is strongly believed to be in the slope region.
- **Uncertain Region around Change Point:** As for the upper truncation version above, if for the preliminary emulator $\mathbb{E}_{\mathbf{F}_i'}[f_i(\mathbf{d})] - 3\sqrt{\mathrm{Var}_{\mathbf{F}_i'}[f_i(\mathbf{d})]} \leq d_{jk,t_i} \cdot \Delta t_i < \mathbb{E}_{\mathbf{F}_i'}[f_i(\mathbf{d})] + 3\sqrt{\mathrm{Var}_{\mathbf{F}_i'}[f_i(\mathbf{d})]}$, **or** if the additional criterion of $\mathbb{E}_{\mathbf{F}_i'}[f_i(\mathbf{d})] - 3\sqrt{\mathrm{Var}_{\mathbf{F}_i'}[f_i(\mathbf{d})]} < a_i$ **whilst** $\mathbb{E}_{\mathbf{F}_i'}[f_i(\mathbf{d})] + 3\sqrt{\mathrm{Var}_{\mathbf{F}_i'}[f_i(\mathbf{d})]} \leq d_{jk,t_i} \cdot \Delta t_i$, a truncated GP emulator is evaluated. The mean and variance are determined by Equations (3.3.3) and (3.3.4) respectively and are computed as for the upper truncation version above noting the change in the lower truncation bound a_i and hence α_i. This form of emulation is used in the uncertain region

around the true location of the change point.

- **Plateau Region:** In all other cases where $\mathbb{E}_{\mathbf{F}'_i}[f_i(\mathbf{d})] - 3\sqrt{\mathrm{Var}_{\mathbf{F}'_i}[f_i(\mathbf{d})]} \geq a_i$ **and** $\mathbb{E}_{\mathbf{F}'_i}[f_i(\mathbf{d})] + 3\sqrt{\mathrm{Var}_{\mathbf{F}'_i}[f_i(\mathbf{d})]} \leq d_{jk,t_i} \cdot \Delta t_i$, the preliminary emulator output is used. This corresponds to when the output is strongly believed to be in the plateau region.

The inclusion of a truncation lower bound ensures that other physical constraints are adhered to. For the NPV constituents: WOPT and WWIT, $a_i = 0$, represents that it is not possible to produce a negative quantity of oil or inject a negative volume of water respectively. This is particularly useful where there is large uncertainty in the preliminary emulator for reasons including: few points on which to fit the emulator; evaluating the preliminary emulator a reasonable distance from the points used in the fitting; and limited information provided from the regression part of the emulator structure.

Compared to applying standard methods such as Gaussian process or Bayes linear emulation, the structured approach demonstrates improvements in accuracy. Moreover, an auxiliary benefit is the increased speed of emulator evaluation and efficiency. This is as a consequence of using fewer design points in the fitting (only those which exceed the change point upper bound) leading to inversions and multiplications of smaller matrices, and the restriction to only performing a full emulator update at new decision parameter vectors which exceed the extrapolation cut-off whilst all others are automatically collapsed onto the slope in order to satisfy the physical constraints. The estimation processes for both the change point upper bound and extrapolation cut-off are computationally very cheap, whilst the use of a truncated GP helps reduce the reliance on accurate estimation of the extrapolation cut-off.

The presented methods are also preferred over alternative approaches to emulating functions with distinct modes of behaviour in different regions of the parameter space such as Treed Gaussian processes [74] because for each NPV constituent the form of the behaviour within the slope region is almost exactly known up to a very

small tolerance. This is encapsulated within the developed form of structured emulation where as a Treed GP would maintain a much larger uncertainty over this region than is necessary with potential implications during subsequent analyses. Moreover, in later application it is highlighted how in certain instances the number of points within these distinct regions can be relatively limited. This potentially impacts on the accuracy of a Treed GP.

Both developed forms of the structured emulation technique allow more complex outputs such as the NPV to be deconstructed into their constituents which exhibit features such as slopes and plateaus. These are emulated separately before combining in a "divide and conquer approach" to obtain an emulator of improved accuracy.

3.3.3 Emulation of the Approximate NPV

A method of accurately emulating the NPV constituents was presented in Section 3.3.2 for which the next step towards emulating the NPV for an individual ensemble member is to mathematically synthesise the constituent emulators. An intuitive means of combining the NPV constituent emulators is via the NPV formula. However, first note that it is often the case that control intervals are formed by the amalgamation of multiple consecutive discounting intervals, hence the constituent emulators do not exactly correspond to the terms in the NPV formula expressed in Equations (3.1.1) and (3.1.3) for well control optimisation, labelled as the exact NPV and denoted by $\text{NPV}_j(\mathbf{d})$ for the j^{th} model evaluated at $\mathbf{d}$. An approximation to the exact NPV is obtained by applying a weighted average discounting factor, λ_i, to the respective NPV constituents. For the i^{th} interval this is defined in Equation (3.3.5) where k indexes the exact discounting intervals contained within the longer control (or approximation) interval, N_{t_i} is the total number of such discounting intervals, and $t_{i,0} = t_{i-1}$. The formula for the average discounting approximate NPV, denoted as $\widetilde{\text{NPV}}_j(\mathbf{d})$ and abbreviated through this thesis to the approximate NPV, is shown in Equation (3.3.6) for well control optimisation.

$$\lambda_i = \frac{1}{t_i - t_{i-1}} \sum_{k=1}^{N_{t_i}} \frac{t_{i,k} - t_{i,k-1}}{(1+d)^{\frac{t_{i,k}}{\tau}}} \tag{3.3.5}$$

$$\widetilde{\mathrm{NPV}}_j(\mathbf{d}) = \sum_{i=1}^{N_t} \lambda_i \left[Q_{j,op}(\mathbf{d}, t_i) \cdot r_{op} - Q_{j,wp}(\mathbf{d}, t_i) \cdot r_{wp} - Q_{j,wi}(\mathbf{d}, t_i) \cdot r_{wi} \right] \tag{3.3.6}$$

Note that using an average discounting factor yields much more accurate approximations than a variant of applying the discounting at the end of the control intervals only.

Continuing with the approximate NPV for a fixed ensemble member, an emulator is obtained by substituting the simulator for emulator output for $Q_{j,op}(\mathbf{d}, t_i)$, $Q_{j,wp}(\mathbf{d}, t_i)$ and $Q_{j,wi}(\mathbf{d}, t_i)$. Under the assumption of uncorrelated NPV constituents which are emulated using a collection of univariate emulators, Equations (3.3.7) and (3.3.8) are the formulae for the adjusted expectation and variance respectively for the $\widetilde{\mathrm{NPV}}_j(\mathbf{d})$ where $\mathbf{F}_j$ denotes the collection of all data used to fit the NPV constituents emulators.

$$\begin{aligned} \mathbb{E}_{\mathbf{F}_j}\left[\widetilde{\mathrm{NPV}}_j(\mathbf{d})\right] = \sum_{i=1}^{N_t} \lambda_i \Big\{ & r_{op}\, \mathbb{E}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d}, t_i)] \\ & - r_{wp}\, \mathbb{E}_{\mathbf{F}_j}[Q_{j,wp}(\mathbf{d}, t_i)] - r_{wi}\, \mathbb{E}_{\mathbf{F}_j}[Q_{j,wi}(\mathbf{d}, t_i)] \Big\} \end{aligned} \tag{3.3.7}$$

$$\begin{aligned} \mathrm{Var}_{\mathbf{F}_j}\left[\widetilde{\mathrm{NPV}}_j(\mathbf{d})\right] = \sum_{i=1}^{N_t} \lambda_i^2 \Big\{ & r_{op}^2\, \mathrm{Var}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d}, t_i)] \\ & + r_{wp}^2\, \mathrm{Var}_{\mathbf{F}_j}[Q_{j,wp}(\mathbf{d}, t_i)] + r_{wi}^2\, \mathrm{Var}_{\mathbf{F}_j}[Q_{j,wi}(\mathbf{d}, t_i)] \Big\} \end{aligned} \tag{3.3.8}$$

For each WOPT and WWIT contributor where structured emulation is applied, for new $\mathbf{d}$ identified as either in the plateau region or the uncertain region around the change point, the adjusted expectation and variance are obtained from the preliminary Bayes linear or truncated GP emulator respectively. In the slope region where the emulator is collapsed onto the slope, the expectation is known whilst the adjusted variance is approximated by treating δ_i as equal to 3 standard deviations.

This methodology can also be extended to permit correlated NPV constituents

via multivariate emulation, as presented in Section 2.5. The adjusted expectation formula remains as in Equation (3.3.7), whilst the adjusted variance is extended to include all the relevant covariance terms between the contributors to the NPV as follows:

$$\begin{aligned}\mathrm{Var}_{\mathbf{F}_j}\left[\widetilde{\mathrm{NPV}}_j(\mathbf{d})\right] = \sum_{i=1}^{N_t}\sum_{l=1}^{N_t}\lambda_i\lambda_l\Big\{&r_{op}^2\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d},t_i),Q_{j,op}(\mathbf{d},t_l)]\\ &-r_{op}r_{wp}\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d},t_i),Q_{j,wp}(\mathbf{d},t_l)]\\ &-r_{op}r_{wi}\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d},t_i),Q_{j,wi}(\mathbf{d},t_l)]\\ &+r_{wp}^2\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,wp}(\mathbf{d},t_i),Q_{j,wp}(\mathbf{d},t_l)]\\ &+r_{wp}r_{wi}\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,wp}(\mathbf{d},t_i),Q_{j,wi}(\mathbf{d},t_l)]\\ &+r_{wi}^2\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,wi}(\mathbf{d},t_i),Q_{j,wi}(\mathbf{d},t_l)]\Big\}\end{aligned} \tag{3.3.9}$$

An alternative form of the adjusted variance is displayed in Equation (3.3.10) where: the first summation contains all of the terms included under the uncorrelated assumption; the second summation gives the temporal correlations for fixed well and NPV contributor type; and the final summation accounts for both the spatial and temporal correlation across contributors of different NPV constituent types.

$$\begin{aligned}\mathrm{Var}_{\mathbf{F}_j}\left[\widetilde{\mathrm{NPV}}_j(\mathbf{d})\right] = &\sum_{i=1}^{N_t}\lambda_i^2\Big\{r_{op}^2\,\mathrm{Var}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d},t_i)]\\ &\qquad+r_{wp}^2\,\mathrm{Var}_{\mathbf{F}_j}[Q_{j,wp}(\mathbf{d},t_i)]+r_{wi}^2\,\mathrm{Var}_{\mathbf{F}_j}[Q_{j,wi}(\mathbf{d},t_i)]\Big\}\\ &+2\sum_{i=1}^{N_t-1}\sum_{l=i+1}^{N_t}\lambda_i\lambda_l\Big\{r_{op}^2\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d},t_i),Q_{j,op}(\mathbf{d},t_l)]\\ &\qquad+r_{wp}^2\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,wp}(\mathbf{d},t_i),Q_{j,wp}(\mathbf{d},t_l)]\\ &\qquad+r_{wi}^2\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,wi}(\mathbf{d},t_i),Q_{j,wi}(\mathbf{d},t_l)]\Big\}\\ &+2\sum_{\substack{i=1\\l\neq i}}^{N_t}\lambda_i\lambda_l\Big\{-r_{op}r_{wp}\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d},t_i),Q_{j,wp}(\mathbf{d},t_l)]\end{aligned}$$

$$-r_{op}r_{wi}\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,op}(\mathbf{d},t_i),Q_{j,wi}(\mathbf{d},t_l)]$$

$$+r_{wp}r_{wi}\,\mathrm{Cov}_{\mathbf{F}_j}[Q_{j,wp}(\mathbf{d},t_i),Q_{j,wi}(\mathbf{d},t_l)]\Big\} \quad (3.3.10)$$

Implementing a multivariate emulator constructed with the assumption of a separable covariance function between the input decision parameters and the NPV constituents as outputs [23] results in an uncoupling of their correlation structures. For the NPV constituents it is reasonable to expect: negative temporal correlations for fixed contributor types (increased production or injection within an interval generally results in a lower production or injection for a later control interval); and positive spatio-temporal correlation between oil production and either water contributor type (higher water injection is executed with the aim of achieving a higher oil production, although with an anticipated side-effect being higher water production). From the form of Equation (3.3.10) it is evident that this can yield a reduction in the adjusted variance compared to that computed under the uncorrelated assumption. Note that there is also the possibility of increased uncertainty, such as due to the expected positive spatio-temporal correlation between water production and injection (the final term in Equation (3.3.10), although investigations show that this is likely to be negligible compared with other terms), however this is unlikely in this scenario given the aforementioned physical considerations.

The primary difficulty in constructing a separable multivariate emulator over all of the NPV constituents is accurately determining their joint correlation structure. This is particularly challenging under the structured emulation approach where different subsets of the output are used in the fitting of the emulator from which to learn the correlations and obtain a valid correlation matrix. In contrast, for Gaussian process or Bayes linear emulation, all of the data is used in the fitting and hence learning about the multivariate correlation structure is simpler, but this is at the cost of much less accurate individual emulators because the partially known behaviour of the NPV constituents is not exploited.

Another method considered was the use of a linear model to link the approximate NPV to a subset of the NPV contributors since the NPV formula is a perfect linear combination of its constituents. It was believed that this would provide a suitable approximation whilst reducing computational costs by only requiring the emulation of a subset of the NPV constituents chosen via forward stepwise selection. Evaluation of this methodology found it to be substantially less effective versus the above approach of directly using the NPV formula and so exact details are omitted.

The NPV constituents are obtained from a single simulator, however the way in which they feed into the NPV calculation suggests that they may also be viewed as from a network of computer simulators. Alternative methodology for emulating simulator output which can be decomposed into a network of component simulators can be seen in [96, 111]. This is best applied where the link between each of the simulator outputs is less well understood. In this instance there is a natural means of combining the emulators for each of the NPV constituents which is intuitive to understand, simple to construct, and quick to evaluate as a function of the individual contributors emulator output. The methodology for emulating networks of component simulators is therefore not employed.

3.3.4 Linking the Exact and Approximate NPV

It is observed that there exists a very strong linear relationship between the approximate and exact NPV (introduced in Section 3.3.3). A statistical link via a simple linear regression model of the form shown in Equation (3.3.11) is proposed to connect them, in order to capture the uncertainties induced by the approximation.

$$\mathrm{NPV}_j(\mathbf{d}) = \beta_{0,\widetilde{\mathrm{NPV}_j}} + \beta_{1,\widetilde{\mathrm{NPV}_j}} \widetilde{\mathrm{NPV}}_j(\mathbf{d}) + \varepsilon_{\widetilde{\mathrm{NPV}_j}} \tag{3.3.11}$$

The adjusted expectation and variance are then computed using Equations (3.3.12) and (3.3.13) respectively where the regression coefficients, their variances and covariance are estimated using the simulation data. In addition, they are assumed to be mutually independent of the emulated approximate NPV. Under the linear

model assumption, $\varepsilon_{\widetilde{\mathrm{NPV}}_j}$ is treated as independent of the regression coefficients and $\widetilde{\mathrm{NPV}}_j(\mathbf{d})$, with $\sigma_{\widetilde{\mathrm{NPV}}_j}$ denoting the residual standard error.

$$\mathbb{E}_{\mathbf{F}_j}[\mathrm{NPV}_j(\mathbf{d})] = \hat{\beta}_{0,\widetilde{\mathrm{NPV}}_j} + \hat{\beta}_{1,\widetilde{\mathrm{NPV}}_j}\,\mathbb{E}_{\mathbf{F}_j}\left[\widetilde{\mathrm{NPV}}_j(\mathbf{d})\right] \tag{3.3.12}$$

$$\begin{aligned}\mathrm{Var}_{\mathbf{F}_j}[\mathrm{NPV}_j(\mathbf{d})] = {} & \mathrm{Var}\left[\hat{\beta}_{0,\widetilde{\mathrm{NPV}}_j}\right] + 2\,\mathrm{Cov}\left[\hat{\beta}_{0,\widetilde{\mathrm{NPV}}_j}, \hat{\beta}_{1,\widetilde{\mathrm{NPV}}_j}\right]\mathbb{E}_{\mathbf{F}_j}\left[\widetilde{\mathrm{NPV}}_j(\mathbf{d})\right] \\ & + \left\{\hat{\beta}^2_{1,\widetilde{\mathrm{NPV}}_j} + \mathrm{Var}\left[\hat{\beta}_{1,\widetilde{\mathrm{NPV}}_j}\right]\right\}\mathrm{Var}_{\mathbf{F}_j}\left[\widetilde{\mathrm{NPV}}_j(\mathbf{d})\right] \\ & + \mathrm{Var}\left[\hat{\beta}_{1,\widetilde{\mathrm{NPV}}_j}\right]\left(\mathbb{E}_{\mathbf{F}_j}\left[\widetilde{\mathrm{NPV}}_j(\mathbf{d})\right]\right)^2 + \sigma^2_{\widetilde{\mathrm{NPV}}_j}\end{aligned} \tag{3.3.13}$$

3.3.5 Emulation of the Ensemble Mean NPV

Up to this point, emulation of the NPV for individual models within an ensemble is described. In decision support or optimisation problems the aim is frequently to maximise the expected NPV with respect to some uncertainty, with the ensemble mean NPV, denoted $\overline{\mathrm{NPV}}$, often treated as a surrogate, as in the TNO OLYMPUS Challenge. The next step is to present a framework for combining these individual emulators in order to obtain an emulator for the expected NPV. A reasonable assumption is that the results for individual models are independent given the complexity of the different underlying geologies.

Emulation of the Ensemble Mean NPV when Simulations are Available for all Ensemble Members

First consider when simulations are performed for the entire ensemble. This is the case when: simulations are relatively quick to evaluate; large amounts of computing resources are available; or there is a desire to minimise the uncertainty due to the underlying geology. This final point is particularly relevant when the ensemble mean NPV is viewed as equal to the true expected NPV. In this situation the ensemble mean NPV is computed as either the arithmetic mean or a weighted mean (when there is a prior probability distribution over the geology models) of the individual model NPVs. This provides a natural method to emulate $\overline{\mathrm{NPV}}$. For an ensemble of size N, the adjusted expectation and variance are defined in Equations (3.3.14)

and (3.3.15), where $\mathbf{F}$ denotes all simulation data incorporating each of the $\mathbf{F}_j$, and ω_j are the mean weights with $\omega_j = \frac{1}{N}$ for the arithmetic mean.

$$\mathbb{E}_{\mathbf{F}}\left[\overline{\text{NPV}}(\mathbf{d})\right] = \sum_{j=1}^{N} \omega_j \, \mathbb{E}_{\mathbf{F}_j}[\text{NPV}_j(\mathbf{d})] \tag{3.3.14}$$

$$\text{Var}_{\mathbf{F}}\left[\overline{\text{NPV}}(\mathbf{d})\right] = \sum_{j=1}^{N} \omega_j^2 \, \text{Var}_{\mathbf{F}_j}[\text{NPV}_j(\mathbf{d})] \tag{3.3.15}$$

The variance formula is simple to adapt when it is believed that the outcome of different models are correlated by introducing the relevant covariance terms in Equation (3.3.15).

Emulation of the Ensemble Mean NPV via the EGES Linear Model

A more realistic and practical scenario is that simulations are only performed for a subset of the ensemble selected using the techniques described in Section 3.2. We propose using the EGES linear model presented in Equation (3.2.1) to emulate $\overline{\text{NPV}}$ with the emulated NPV for each of the sub-selected models as inputs, where the estimated EGES linear model regression coefficients are denoted by $\hat{\alpha}_{\text{EGES}}$ and $\hat{\beta}_{k,\text{EGES}}$. It is assumed that the individual emulator outputs and the regression coefficients are uncorrelated. This is justifiable if two distinct simulation data sets are used to construct the EGES linear model and to fit the emulators of the NPV for individual models. Under this formulation the adjusted expectation is shown in Equation (3.3.16).

$$\begin{aligned}\mathbb{E}_{\mathbf{F}}\left[\overline{\text{NPV}}(\mathbf{d})\right] &= \mathbb{E}_{\mathbf{F}}\left[\hat{\alpha}_{\text{EGES}} + \sum_{k=1}^{N_{\text{EGES}}} \hat{\beta}_{k,\text{EGES}} \text{NPV}_{j_k}(\mathbf{d}) + \varepsilon_{\text{EGES}}(\mathbf{d})\right] \\ &= \hat{\alpha}_{\text{EGES}} + \sum_{k=1}^{N_{\text{EGES}}} \hat{\beta}_{k,\text{EGES}} \, \mathbb{E}_{\mathbf{F}_{j_k}}[\text{NPV}_{j_k}(\mathbf{d})]\end{aligned} \tag{3.3.16}$$

Define $\hat{\boldsymbol{\beta}}_{\text{EGES}} = (\hat{\alpha}_{\text{EGES}}, \hat{\beta}_{1,\text{EGES}}, \ldots, \hat{\beta}_{N_{\text{EGES}}})^{\text{T}} \in \mathbb{R}^{N_{\text{EGES}}+1}$ with $\Sigma_{\beta,\text{EGES}} = \text{Var}\left[\hat{\boldsymbol{\beta}}_{\text{EGES}}\right]$, and $X_{\text{EGES}}(\mathbf{d}) = (1, \text{NPV}_{j_1}(\mathbf{d}), \ldots, \text{NPV}_{j_{N_{\text{EGES}}}}(\mathbf{d}))^{\text{T}} \in \mathbb{R}^{N_{\text{EGES}}+1}$ with uncorrelated components. The adjusted variance is derived in Equation (3.3.17), where $\hat{\sigma}_{\text{EGES}}$ is

the residual standard error of the EGES linear model.

$$\begin{aligned}\mathrm{Var}_{\mathbf{F}}\left[\overline{\mathrm{NPV}}(\mathbf{d})\right] &= \mathrm{Var}_{\mathbf{F}}\left[\hat{\alpha}_{\mathrm{EGES}} + \sum_{k=1}^{N_{\mathrm{EGES}}} \hat{\beta}_{k,\mathrm{EGES}}\mathrm{NPV}_{j_k}(\mathbf{d}) + \varepsilon_{\mathrm{EGES}}(\mathbf{d})\right] \\ &= \mathrm{Var}_{\mathbf{F}}\left[X^{\mathrm{T}}_{\mathrm{EGES}}(\mathbf{d})\hat{\boldsymbol{\beta}}_{\mathrm{EGES}}\right] + \mathrm{Cov}_{\mathbf{F}}\left[X^{\mathrm{T}}_{\mathrm{EGES}}(\mathbf{d})\hat{\boldsymbol{\beta}}_{\mathrm{EGES}}, \varepsilon_{\mathrm{EGES}}(\mathbf{d})\right] \\ &\quad + \mathrm{Cov}_{\mathbf{F}}\left[\varepsilon_{\mathrm{EGES}}(\mathbf{d}), X^{\mathrm{T}}_{\mathrm{EGES}}(\mathbf{d})\hat{\boldsymbol{\beta}}_{\mathrm{EGES}}\right] + \mathrm{Var}_{\mathbf{F}}\left[\varepsilon_{\mathrm{EGES}}(\mathbf{d})\right]\end{aligned}$$

Assuming that $\boldsymbol{\beta}_{\mathrm{EGES}}$ and $X_{\mathrm{EGES}}(\mathbf{d})$ are uncorrelated, and noting that $\mathrm{Cov}_{\mathbf{F}}\left[X^{\mathrm{T}}_{\mathrm{EGES}}(\mathbf{d})\hat{\boldsymbol{\beta}}_{\mathrm{EGES}}, \varepsilon_{\mathrm{EGES}}(\mathbf{d})\right] = \mathbf{0} \in \mathbb{R}^{N_{\mathrm{EGES}}+1}$.

$$\begin{aligned}&= \mathbb{E}_{\mathbf{F}}\left[X^{\mathrm{T}}_{\mathrm{EGES}}(\mathbf{d})\Sigma_{\beta,\mathrm{EGES}}X_{\mathrm{EGES}}(\mathbf{d})\right] \\ &\quad + \hat{\boldsymbol{\beta}}^{\mathrm{T}}_{\mathrm{EGES}}\,\mathrm{Var}_{\mathbf{F}}\left[X_{\mathrm{EGES}}(\mathbf{d})\right]\hat{\boldsymbol{\beta}}_{\mathrm{EGES}} + \hat{\sigma}^2_{\mathrm{EGES}}\end{aligned}$$

Assuming that the components of $X_{\mathrm{EGES}}(\mathbf{d})$ are uncorrelated.

$$\begin{aligned}&= \mathrm{Var}\left[\hat{\alpha}_{\mathrm{EGES}}\right] \\ &\quad + \sum_{k=1}^{N_{\mathrm{EGES}}} \mathrm{Var}\left[\hat{\beta}_{k,\mathrm{EGES}}\right]\left(\mathrm{Var}_{\mathbf{F}_{j_k}}\left[\mathrm{NPV}_{j_k}(\mathbf{d})\right] + \mathbb{E}_{\mathbf{F}_{j_k}}\left[\mathrm{NPV}_{j_k}(\mathbf{d})\right]^2\right) \\ &\quad + 2\sum_{k=1}^{N_{\mathrm{EGES}}} \mathrm{Cov}\left[\hat{\alpha}_{\mathrm{EGES}}, \hat{\beta}_{k,\mathrm{EGES}}\right]\mathbb{E}_{\mathbf{F}_{j_k}}\left[\mathrm{NPV}_{j_k}(\mathbf{d})\right] \\ &\quad + \sum_{\substack{k\neq l \\ k,l=1,\ldots,N_{\mathrm{EGES}}}} \Big(\mathrm{Cov}\left[\hat{\beta}_{k,\mathrm{EGES}}, \hat{\beta}_{l,\mathrm{EGES}}\right] \\ &\qquad\qquad \cdot \mathbb{E}_{\mathbf{F}_{j_k}}\left[\mathrm{NPV}_{j_k}(\mathbf{d})\right]\mathbb{E}_{\mathbf{F}_{j_l}}\left[\mathrm{NPV}_{j_l}(\mathbf{d})\right]\Big) \\ &\quad + \sum_{k=1}^{N_{\mathrm{EGES}}} \hat{\beta}^2_k\,\mathrm{Var}_{\mathbf{F}_{j_k}}\left[\mathrm{NPV}_{j_k}(\mathbf{d})\right] + \hat{\sigma}^2_{\mathrm{EGES}}\end{aligned} \tag{3.3.17}$$

Note that this formula may also be applied more generally where linear models are used to combine uncorrelated emulator output and for which it is reasonable to assume that the estimates of the linear model coefficients are uncorrelated with the emulator output.

3.3.6 Summary of the Hierarchical Emulator Construction

In this section we have developed a hierarchical Bayesian emulation framework to exploit structure within the utility function constituent parts, as motivated by the ensemble mean NPV within the general well control optimisation problem, such as that presented in the TNO OLYMPUS Challenge. This consists of:

- Decomposing the utility function into its constituent pieces, for example, the components of the NPV formula in Section 3.3.1.
- Formulating structured emulators which exploit prior knowledge of constrained behaviour for certain parts of the utility function, in particular, for the WOPT and WWIT within each control interval, as detailed in Section 3.3.2.
- Amalgamating constituent emulators to approximate the NPV using longer control and discounting time intervals for an individual ensemble member in Section 3.3.3.
- Linking the emulator for the approximate with the exact NPV for an individual ensemble member in Section 3.3.4.
- Efficiently using a small subset of the ensemble to emulate the ensemble mean NPV in Section 3.3.5.

We will subsequently link the ensemble mean NPV to the expected NPV via ensemble uncertainty in Section 3.5.3.

3.4 Optimisation and Decision Theoretic Approaches

Up to this point we have presented methodology which enables more accurate emulation by taking into account certain structures within the response of interest. At this stage we move our focus to consider how to make better decisions providing an overview in this section of optimisation, Bayesian optimisation, decision theoretic and decision support frameworks in Sections 3.4.1 to 3.4.4 respectively.

3.4.1 Optimisation

From an applied Mathematics perspective, the aim of optimisation for an objective function $f(\mathbf{d})$ is to identify $\mathbf{d}_{\max}$ defined as follows [15]:

$$\mathbf{d}_{\max} = \underset{\mathbf{d}\in\Omega\subset\mathbb{R}^D}{\arg\max}\, f(\mathbf{d}) \tag{3.4.1}$$

In the context of field development optimisation, $f(\mathbf{d})$ is the expected NPV for a particular well control and well placement strategy.

There exists large swathes of literature focusing on maximising such non-linear functions. When $f(\mathbf{d})$ possesses an explicit mathematical form it may be possible to solve this analytically, whilst for all other situations there are a large selection of numerical optimisation algorithms from which to choose. However, there exist several potential limitations to these approaches since they implicitly assume that:

- The function of interest can be evaluated and has a reasonable computation time,
- The function is either the quantity to be optimised or perfectly represents the real world process to be maximised,
- There are no other sources of uncertainty.

Many simulators (or functions) of interest are of an unknown form and have a long evaluation time rendering such algorithms computationally intractable, whilst many optimisation routines exploit derivative information which is often unknown. Moreover, simulators are used as a surrogate for the real world physical systems since they are simpler to experiment on, however the differences between the simulator and reality, usually in the form of structural model discrepancy, must be acknowledged for the optimisation process to be meaningful. Perhaps most importantly, the potential effect of other uncertainties on the optimised parameter setting is not accounted for by these optimisation techniques.

3.4.2 Bayesian Optimisation

A Bayesian statistical approach to optimisation is achieved through the use of emulators to approximate the expensive computer simulator in order to mitigate the computational concerns; the most common choice being Gaussian process emulators. The Bayesian framework also permits the comprehensive treatment of the uncertainties. A large proportion of the methodology focuses on sequentially selecting the next design point at which to evaluate the simulator through the use of a Bayesian optimisation criterion, also known as an acquisition function. This is a myopic or one-step ahead process. Note that the introduction of such criteria shifts the often computationally demanding optimisation problem from the original function to the chosen criterion, many of which are either of an analytic form or are much cheaper to evaluate; as well as satisfy conditions such as continuity enabling traditional optimisation procedures to be applied in selecting the next design point [15, 102, 134].

Examples of Bayesian optimisation criteria which exploit the probabilistic properties of the GP to yield a closed form and cheap to evaluate expression include: Probability Of Improvement (POI); Expected Improvement (EI) [102], with an extension being to consider the expectation of higher powers of the improvement function; (Gaussian Process) confidence bound criteria including Lower & Upper Confidence Bounds ((GP-) LCB & UCB respectively) [15]. Several other myopic optimisation strategies compared in [56, 134] include: maximising uncertainty, as characterised by standard deviation for which there is no bias towards high performance regions, although is inefficient in practice; the Stepwise Uncertainty Reduction (SUR) strategy, also known as the Informational Approach to Global Optimisation (IAGO) [54, 179]; Radial Basis Function (RBF) method [79]; and the Dividing Rectangles (DIRECT) optimisation algorithm [101]. The Gaussian Process Global Optimisation (GPGO) procedure [134] may also be used for myopic searches and shares similarities with the use of the EI criterion.

The specific choice of criterion is application dependent and is selected to achieve

a trade-off between exploration and exploitation. Exploration involves placing a design point within a region of the decision parameter space for which there is high function uncertainty. This is usually where there are few other design points. Exploitation selects new design points within a region where the function is expected to be high based on the Gaussian process model. Once sufficient design points have been explored, this is often characterised by a clustering of design points around where the function is expected to be maximised. The EI criterion is a particularly popular choice [15, 56, 76, 102, 113, 160], with the optimisation routine using GPs and EI known as the Efficient Global Optimisation (EGO) algorithm [102, 134]. This "vanilla" version of Bayesian optimisation has many advantages including: ease of implementation; efficient single point criteria; incorporates smoothness; constructs a surrogate representation of the objective function; and can handle multiple modes. However, there exist several weaknesses to such an approach including: an over-reliance on the specific form of the emulator which may also lead to boundary issues, particularly in higher dimensions; does not incorporate derivative information or batches of simulation output; and is designed for a myopic search [176]. Note that these advantages and disadvantages also apply to many of the other criterion based methods presented in this section, with the main distinction being computational tractability, whilst one-step ahead searches remain a constant issue. An identified limitation specific to the EI criterion is that it over-exploits the fitted GP model resulting in an under-exploration of the parameter space, potentially with the consequence that the procedure becomes stuck at local optima and thus fails to converge to a global optimum [18, 145]. A proposed solution is the Hierarchical Expected Improvement (HEI) acquisition function which allows for greater exploration whilst maintaining efficiency. This is is achieved through the use of priors over the GP hyperparameters leading to a t-distribution and a finite sample correction, as well as a closed form expression for HEI. Further details can be found in [22].

The Bayesian optimisation methodology has been extended to account for noisy experimental data and constraints, as reviewed in [113, 134]. Examples of information

based acquisition functions from [113] which are designed to handle noisy data include: Noisy Expected Improvement (NEI) [138]; entropy based methods including entropy search [83], Predictive Entropy Search (PES) [84], and Predictive Entropy Search with Constraints (PESC) [86], although their practical application is limited by the intractability of calculations requiring difficult to implement approximations: knowledge gradient [156]; and Thompson sampling for GP Bayesian optimisation [85]. Two further algorithms discussed in [134] are Implicit Filtering (IF) [105] and the Stable Noisy Optimisation by Branch and Fit (SNOBFIT) algorithm [91]. Another approach is quantile-based optimisation of noisy computer experiments [139, 140] which uses an Expected Quantile Improvement (EQI) criterion. This is applicable for optimisation of stochastic simulators, or for deterministic simulators with uncertain model, environmental or tuning variables.

The above described Bayesian optimisation methodology is for a myopic search to sequentially select the next individual new design point. Only evaluating the simulator for a single parameter setting at each step is very often a sub-optimal use of the available computational resources, especially when parallel computing architecture is accessible. A natural extension is therefore to a non-myopic strategy in the form of batch optimisation via the multi-points (q-points) Expected Improvement (q-EI) criterion within the GPGO algorithm [134]. This compares favourably versus the myopic strategies for both noise-free and noisy functions. Another option is multi-step optimisation [94] through the use of loss functions such as: constrained quadratic loss, constrained Laplace losses, and an extended Laplace loss, with a Forward Filtering, Backward Smoothing (FFBS) algorithm used for Bayesian Expectation Maximisation (EM).

Multi-Objective Optimisation occurs when there exist multiple objective functions which cannot be combined into a single function. Pareto sets (or bounds or curves) form a useful tool and are composed of decisions for which increasing one objective cannot be achieved without decreasing any one of the others. Such decisions are labelled as non-dominated [56, 157]. An example is the trade-off between the

updated mean and standard deviation in GP Bayesian optimisation seen in [56].

3.4.3 Decision Theoretic Approaches

Decision theory provides a coherent framework for identifying a decision strategy which is optimal with respect to both the objectives of the user whilst incorporating all relevant sources of uncertainty [35]. Both decision theory and decision support approaches to the problem require the introduction of a utility function. Whilst an objective function as discussed in the context of optimisation often represents a specific numerical quantity which is an output of the system or simulator (for petroleum field development optimisation this is the expected NPV), utility functions are able to combine multiple facets that are important to decision making including both social and economic considerations. Such quantities do not necessarily have common units or physical interpretations; thus they are not usually comparable. By converting onto units of utility, the decision theoretic approach permits their comparison such that decisions which yield higher utility values are considered preferable. Moreover, a utility function is also able to incorporate the risk preferences of the decision maker(s). Under certain scenarios, the utility function may be specified as equal to the original objective function and in situations where the objective function is a monetary value, this corresponds to expected monetary value with the decision maker being risk neutral. Similar to the optimisation setup, the aim is to maximise the expected utility. Note that an equivalent formulation is in terms of a loss function which is the negative of the utility function and hence should be minimised.

Utility functions are commonly denoted by $U(\cdot)$. These are functions of simulator output $f(\mathbf{x}, \mathbf{d})$ which may explicitly depend on the beliefs about the model parameters, the specified decisions encapsulated in $\mathbf{d}$, and some additional utility function cost parameters $\mathbf{c}$ that are not inputs to the simulator, written as $U(f(\mathbf{x}, \mathbf{d}), \mathbf{d}, \mathbf{c})$. For a decision theoretic approach within a full Bayesian paradigm the expected utility, written as $U(\mathbf{d}, \mathbf{c})$ where there is an explicit dependence on the utility function

cost parameters, is computed as an integral with respect to the model parameters, as shown in Equation (3.4.2), where $\pi_x(\mathbf{x})$ is the prior distribution for the model parameters. When $f(\cdot)$ represents a complex computer model then it is often not possible to evaluate this integral analytically and so may be approximated by a Monte Carlo estimate shown in Equation (3.4.3) where $\mathbf{x}_i,\ i = 1, \ldots, n_x$, is a sample of model parameters. This corresponds to using the ensemble mean NPV as a surrogate for the expected NPV within the TNO OLYMPUS Challenge, as well as more generally within the petroleum industry, where the model parameters represent various geological configurations and $n_x = N$, the number of ensemble members. A weakness to the Monte Carlo approximation is the requirement to evaluate $f(\cdot)$ a large number of times in order to obtain a sufficiently small Monte Carlo estimate variance. This is exacerbated when the simulator is highly computationally expensive.

$$U(\mathbf{d}, \mathbf{c}) = \mathbb{E}_{\mathbf{x}}[U(f(\mathbf{x}, \mathbf{d}), \mathbf{d}, \mathbf{c})] = \int_{\mathbf{x}\in\mathcal{X}} U(f(\mathbf{x}, \mathbf{d}), \mathbf{d}, \mathbf{c})\pi_x(\mathbf{x})\, d\mathbf{x} \tag{3.4.2}$$

$$\approx \frac{1}{n_x}\sum_{i=1}^{n_x} U(f(\mathbf{x}_i, \mathbf{d}), \mathbf{d}, \mathbf{c}) \tag{3.4.3}$$

Note that where there is no model parameter uncertainty, the explicit dependence of $U(\cdot)$ on $f(\mathbf{x}, \mathbf{d})$ is omitted for notational clarity. In this situation $U(\mathbf{d}, \mathbf{c}) = \mathbb{E}_{\mathbf{x}}[U(\mathbf{d}, \mathbf{c})]$. For the petroleum industry application this corresponds to using only one model representing a single geological realisation which is believed to be representative of the corresponding real oil field.

When the utility function cost parameters are also uncertain, a second expectation is computed in Equation (3.4.4) to obtain $U(\mathbf{d})$, where $\pi_c(\mathbf{c})$ is the prior distribution for the utility function cost parameters and $\mathcal{C}$ is the cost parameter space. Depending on the form of $U(\mathbf{d}, \mathbf{c})$ and the specified $\pi_c(\cdot)$, this integral may be computed analytically. Alternatively, it is similarly approximated using Monte Carlo in Equation (3.4.5) where $\mathbf{c}_i,\ i = 1, \ldots, n_c$, is a sample of cost parameters. Since $\mathbf{c}$ is not an input to $f(\cdot)$ and depending on the computational expense of evaluating $U(\mathbf{d}, \mathbf{c})$, it is often possible to evaluate the Monte Carlo approximation using a much

larger n_c and hence achieve a substantially smaller Monte Carlo error.

$$U(\mathbf{d}) = \mathbb{E}_{\mathbf{c}}[U(\mathbf{d}, \mathbf{c})] = \int_{\mathbf{c}\in\mathcal{C}} U(\mathbf{d}, \mathbf{c})\pi_c(\mathbf{c})\, d\mathbf{c} \tag{3.4.4}$$

$$\approx \frac{1}{n_c}\sum_{i=1}^{n_c} U(\mathbf{d}, \mathbf{c}_i) \tag{3.4.5}$$

An example of the cost parameters in the TNO OLYMPUS Well Control Optimisation Challenge is the NPV cost parameters of oil price, water production and injection costs, and the discount factor. The notation $U(\mathbf{d})$ is also used when there are implicitly no utility function cost parameters.

The decision theoretic optimal decision $\mathbf{d}^*$ is then defined in Equation (3.4.6), although this can be challenging to solve, particularly when $\mathbf{d}$ is high-dimensional. This constitutes a single node decision problem and is representative of the setup in the TNO OLYMPUS Challenge.

$$\mathbf{d}^* = \underset{\mathbf{d}\in\Omega\subset\mathbb{R}^D}{\arg\max}\, U(\mathbf{d}) \tag{3.4.6}$$

For a more complete treatment of statistical decision theory, see [154] for example.

An alternative approach which links to the idea of decision support explored in Section 3.4.4 naturally arises when there is insufficient information to accurately specify $\pi_c(\cdot)$. Instead, we propose exploring feasible values for $\mathbf{c}$ encapsulated in the finite set $\mathcal{C}' \subset \mathcal{C}$ to obtain $\mathbf{d}^*(\mathbf{c}) = \arg\max_{\mathbf{d}\in\Omega} U(\mathbf{d}, \mathbf{c})$ for each $\mathbf{c} \in \mathcal{C}'$, and then construct a set of decision theoretic optimal decisions with respect to one of the specified utility function cost parameters settings Ω^* defined in Equation (3.4.7) [175]. This is the spirit of decision support where the aim is to provide a class of near-optimal decisions.

$$\Omega^* \coloneqq \{\mathbf{d}^*(\mathbf{c}) \mid \mathbf{c} \in \mathcal{C}'\} \tag{3.4.7}$$

Each $\mathbf{d} \in \Omega^*$ is a viable optimal decision strategy under a particular utility function cost parameter scenario. The final decision maker(s) can then examine Ω^* and its attributes to gain a more detailed understanding of the decision problem and is a particularly useful approach where there is a committee of decision makers separate

from the analysts, who may possess differing views on the expression of utility, potentially captured via the choice of cost parametrisation.

Sequential decision making or multi-node decision problems, which are more realistic of the actual process used within oil field development, concern devising a decision strategy for an initial time period in the knowledge that future decisions will then be taken at later times in light of the observed implications of all previous actions. This sequence of decisions may be illustrated using a decision tree. The statistically correct way to address such problems is via backwards induction starting with the decision(s) at the final time point and working backwards to propose the initial decision(s). For each additional layer or time step this process requires the calculation of an integral over the potential final observed utility conditional on selecting any future decisions which maximise this function given the proposed current decision. Consequently backwards induction quickly becomes prohibitively computationally expensive for all but problems of modest dimensionality due to the complex form of the integrals and numerous optimisation sub-problems [31, 185]. A more tractable and feasible approach is forwards induction in which decisions are optimised at each step without fully accounting for the potential future decisions [175]. Note that much of the existing petroleum industry literature on "closed loop" optimisation such as [21] is actually a form of forwards induction, rather than backwards induction implied by the title.

Both optimisation and decision theoretic approaches yield a single optimal decision parameter choice with neither framework directly highlighting any sensitivity in the objective or utility function evaluated for this decision due to uncertainties present in the process. This can be assessed via a sensitivity analysis. Note that there is a distinction between a traditional probabilistic sensitivity analysis which focuses on the effects of alternative prior distributions, and a decision theoretic sensitivity analysis. A comprehensive review of the latter can be found in [127] with details of sensitivity measures including: Expected Value of Perfect Information (EVPI) and Expected Value of Sample Information (EVSI), whilst further examples discussed

in [182] include: Value at Risk (VaR, also known as Quantile-Loss); Conditional Value at Risk (CVaR, also known as Upper-Trimmed Mean Utility); and Cumulative Expected Loss (CEL). The use of computationally expensive computer models for aiding decision makers requires the use of emulators in order for many of the sensitivity analysis techniques to be tractable, whilst Bayesian quadrature may be used to approximate integrals [127]. In addition, the effects of model misspecification on the expected utility should be investigated. An example of a method within a full Bayesian framework is the use of the Kullback-Leibler (KL) divergence to define a neighbourhood (KL ball of given radius) around the model parameter posterior distribution and subsequently explore to obtain a local-minimax distribution [182]. A decision analysis is said to be robust if perturbations (based on the uncertainties) in the decision setting do not affect the optimality of the decision [182], although this may be extended to also consider when the resultant change in the utility function is small relative to the magnitude of its uncertainties.

3.4.4 Decision Support

Up to this point we have considered optimisation and decision theoretic techniques which yield a single decision strategy. It is identified that the various optimisation algorithms frequently suffer from several major limitations which include: relying on a large number of potentially computationally intractable function evaluations; some necessitate gradient information which may be unavailable for black-box models; challenges in high-dimensional decision parameter spaces which lead to a tendency for algorithms to become stuck at local-optima without first fully exploring the space; and only yield a single "optimal" decision strategy which may be sensitive to the uncertainties. Further challenges in decision theory include an inability to efficiently compute (or approximate) integrals required to obtain the expected utility. We therefore develop a decision support framework to obtain a class of decisions which are near to the optimal decision that would be obtained via utility maximisation, whilst respecting the various uncertainties within the analysis. The important delineation

between decision theory and decision support is the treatment of uncertainty. In decision theory, all uncertainties are resolved via integration. However, in decision support the sources of uncertainty are divided into two mutually exclusive and exhaustive sub-collections distinguished by whether the uncertainties: will never be resolved by the decision maker(s) and so are integrated over, as in decision theory; or can in principle be resolved by the decision maker(s) where these uncertainties remain within the analysis and are used to define a class of acceptable decisions. A related concept is Ω^* defined in Equation (3.4.7) where the uncertainty in the cost parameter scenarios is used to define a collection of potentially optimal decisions for the decision maker to select from.

Within the decision theoretic approach it is assumed that the utility function is known and fixed, however this is not an entirely realistic viewpoint. Firstly the utility function must be elicited from the decision maker(s); a task which is known to be difficult to perform accurately and subsequently leads to the criticism of the subjective choice of utility function which may only be addressed by investigating the consequences of alternative specifications via a form of utility function sensitivity analysis. This is analogous to the criticism of subjective Bayesianism regarding the specification of the prior distribution. Further challenges occur in many real-world governmental and commercial applications where there is a committee of decision maker(s), each of whom possesses there own individual beliefs. In order to implement a decision theoretic approach, it is necessary for the committee to reach an agreement and propose a single utility function, potentially via analogous methods to those of belief aggregation discussed in Section 2.4.3. The resultant uncertainty in the utility function must be accounted for, however it is not clear how do so robustly without solving the problem for each individual utility function which in turn may also be uncertain. A critically important point is that the statistician(s) or analyst(s) may not have access to the committee of decision makers, as is the case in the petroleum industry, leading to utility function uncertainty due to unknown judgements and risk preferences which cannot be resolved, as required within a decision theoretic

approach, before returning the suggested optimal decision strategy to the committee. This is not an isolated setup pertaining to the oil industry; the separation of analysts from the decision makers occurs in many other scientific disciplines, for example, epidemiology and climate science. In addition, this is related to the concept of imprecise utility discussed in [45, 46, 47], which is used to obtain Pareto optimal choices of parameter settings (for designs) under the imprecise specification.

All of the highlighted weaknesses of the decision theoretic approach within the context of real-world decision making problems suggest that instead of a single, potentially non-robust decision, the analyst(s) should return a class of decisions which respect all of the uncertainties related to the decision structure. This is a decision support framework which is further developed in Section 3.6.

3.5 Uncertainty Quantification within Decision Support

A comprehensive uncertainty quantification forms a vital part of any robust analysis of computer models, as introduced in Section 2.7. This is further developed in this section for robust decision support where the object of interest is the utility function, $U(f(\mathbf{x}, \mathbf{d}), \mathbf{d}, \mathbf{c})$, $U(\mathbf{d}, \mathbf{c})$ or $U(\mathbf{d})$ depending on the context, instead of the computer model, although evaluation of the utility will often implicitly require simulation from the model. Much of the methodology and framework presented in Section 2.7 remains relevant including the additive structure for linking computer models to reality in Section 2.7.1 and the various sources of uncertainty described in Section 2.7.2, whilst in order to make meaningful inference for decisions to enact on the real physical system, structural model discrepancy described in Section 2.7.4 plays a crucial role.

This section advances the uncertainty quantification framework for decision support. Firstly, it is important to analyse how an uncertainty affects the utility surface and consequently whether it is important from the perspective of decision support which is discussed in Section 3.5.1. In Section 3.5.2 an assessment of the utility func-

tion cost parameter uncertainty is presented including a method for reducing the effect of such uncertainties to achieve more precise decision support results. Where ensembles of models are used, it is important to account for the induced uncertainty due to using a sample of the possible models, as described in Section 3.5.3. Another form of uncertainty which is relevant to decision making is implementation error which is discussed in Section 3.5.4, whilst utility function uncertainty is considered in Section 3.5.5. Much of this framework and methodology is first introduced in [135] where it is applied to the TNO OLYMPUS Well Control Optimisation Challenge, with some further developments also presented in this section.

3.5.1 Effect of Uncertainties on the Utility Surface

The aim of decision support is to identify vectors of decision parameters which yield a utility close to the maximum value within a tolerance defined by the uncertainties pertaining to the decision structure. Any uncertainty about the decision process which potentially changes the location in decision space of the maximum utility (a priori this is unknown) has the potential to redefine the decision support region, whilst changes in the height or a positive scaling of the utility function do not affect this region, as long as other uncertainties are also modified appropriately. Consequently it is important to differentiate between sources of uncertainty that result in a shift or a positive scaling of the utility surface, and those which produce any other change of shape. A shift and positive scaling of the surface constitute a form of linear transformation which do not affect the location of the optimal solution assuming that the actual utility function is in use by the properties of decision theory. Linear transformation effects should therefore be removed in order to obtain an accurate uncertainty quantification for the purpose of decision support. Note that the specific linear transformation becomes important when answering questions about actual utility and hence should be stored. A method of performing such removal of linear transformation effects is presented in the context of the utility function cost parameters in Section 3.5.2.

Note that this discussion is also relevant within an optimisation and decision theoretic setup where the aim is to identify the global maximum objective or utility in which only the shape, rather than the actual height is of relevance.

3.5.2 Utility Function Cost Parameter Uncertainty

The utility function often includes as inputs a collection of additional cost parameters which are not directly passed to the simulator resulting in uncertainty ε_{C}. Often these are economic parameters which may be time variable and are frequently fixed throughout an analysis. For example, in the TNO OLYMPUS Well Control Optimisation Challenge with the utility function assumed to be the expected NPV, the cost parameters of price per barrel of oil; the cost per barrel of water produced and injected; and the discount factor over time, are the NPV and utility function cost parameters. These are assumed to be fixed in the standard TNO OLYMPUS Challenge setup. The expected NPV utility function is obtained by integrating out stochastic variability. However, only a single fixed vector of cost values is provided, yet a committee of decision makers may possess more refined beliefs; hence this uncertainty can in principle be resolved at a later stage and so is treated as such within decision support. The same pretence is relevant to other decision making applications.

For decision support the treatment of this uncertainty depends on the exact nature of the cost parameters and usually involves some element of expert elicitation. It may be appropriate when there is a relatively limited number of cost parameters to elicit either a prior distribution or the induced variance in the utility. In scenarios such as for large numbers of time variable economic parameters, a simulation based approach may be preferred drawing samples of the utility function cost parameters from a model or probability distribution.

Since the cost parameters are not necessarily inputs to the often computationally expensive simulator, whilst the utility function is usually much cheaper to compute, large numbers of experiments can be performed on the utility function given a fixed

collection of simulations over a design $\mathcal{D}$. It is highlighted in the above section that linear transformations of the utility surface do not affect the location of the optimal decision strategy and hence the decision support region for which a method of accounting for such effects is presented within the context of utility function cost parameter uncertainty.

Let $U(\mathbf{d}_j, \mathbf{c}_i)$ be the utility function evaluated for decision parameter vector $\mathbf{d}_j \in \mathcal{D},\ j = 1, \ldots, n$, for a generated cost parameter setting $\mathbf{c}_i,\ i = 1, \ldots, n_c$. Note that there is an implicit dependence on the simulator. To remove the effects of linear transformations, define a base cost parameter scenario $\mathbf{c}_0$ at which the utility is also computed. A convenient choice for $\mathbf{c}_0$ is the fixed value used within the analysis which may be the mean obtained from either the simulation method or the probability distribution, or elicited from the expert(s). We employ linear regression models of the form shown in Equation (3.5.1) as a means of removing any linear transformation effects for arbitrary choice of $\mathbf{d} \in \Omega$. For each $\mathbf{c}_i$ a distinct linear model is fitted using all $\mathbf{d}_j \in \mathcal{D}$ to transform the new utility value obtained onto that achieved using $\mathbf{c}_0$. Note that the magnitude of the residuals $\varepsilon_{\mathrm{C}}(\mathbf{d}, \mathbf{c}_i)$ may also depend on the location in decision space.

$$U(\mathbf{d}, \mathbf{c}_0) = \beta_{\mathrm{C},0,i} + \beta_{\mathrm{C},1,i} U(\mathbf{d}, \mathbf{c}_i) + \varepsilon_{\mathrm{C}}(\mathbf{d}, \mathbf{c}_i) \quad i = 1, \ldots, n_c \tag{3.5.1}$$

It is these residuals which quantify the non-linear variability in the utility function induced by the uncertainty on $\mathbf{c}$. Note that by linearly mapping $U(\mathbf{d}, \mathbf{c}_i)$ onto $U(\mathbf{d}, \mathbf{c}_0)$, the utility function evaluated for a fixed cost parameter scenario, ensures that the residuals are directly comparable. Estimates for the residuals over $\mathcal{D}$ are denoted by $\hat{\varepsilon}_{\mathrm{C}}(\mathbf{d}_j, \mathbf{c}_i),\ j = 1, \ldots, n$. For each fixed $\mathbf{d}_j \in \mathcal{D}$, the sample mean and variance of the residuals over the linear models, and hence the different cost parameter settings, written as $m_{\mathrm{C}}(\mathbf{d}_j)$ and $s^2_{\mathrm{C}}(\mathbf{d}_j)$, are computed in Equations (3.5.2) and (3.5.3) as estimates of $\mathbb{E}[\varepsilon_{\mathrm{C}}(\mathbf{d}_j)]$ and $\mathrm{Var}[\varepsilon_{\mathrm{C}}(\mathbf{d}_j)]$ respectively.

$$\mathbb{E}[\varepsilon_{\mathrm{C}}(\mathbf{d}_j)] \approx m_{\mathrm{C}}(\mathbf{d}_j) = \frac{1}{n_c} \sum_{i=1}^{n_c} \hat{\varepsilon}_{\mathrm{C}}(\mathbf{d}_j, \mathbf{c}_i) \tag{3.5.2}$$

$$\mathrm{Var}[\varepsilon_{\mathrm{C}}(\mathbf{d}_j)] \approx s^2_{\mathrm{C}}(\mathbf{d}_j) = \frac{1}{n_c - 1} \sum_{i=1}^{n_c} \left(\hat{\varepsilon}_{\mathrm{C}}(\mathbf{d}_j, \mathbf{c}_i) - m_{\mathrm{C}}(\mathbf{d}_j)\right)^2 \tag{3.5.3}$$

There remains a potential dependency on $\mathbf{d}_j$. For decision support a conservative estimate of the uncertainty is given by $\max_{\mathbf{d}_j \in \mathcal{D}} \mathrm{Var}[\varepsilon_{\mathrm{C}}(\mathbf{d}_j)]$. Alternatively an empirical quantile may be sufficient. If there is large variability in $\mathrm{Var}[\varepsilon_{\mathrm{C}}(\mathbf{d}_j)]$ over the decision space then variance emulation techniques may be applied [2, 3]. This methodology for quantifying uncertainty after the removal of linear transformation effects generalises to any other source of uncertainty which is easily parametrised and for which the simulator and utility function are sufficiently quick to evaluate.

3.5.3 Ensemble Uncertainty

Many scientific disciplines and commercial applications use an ensemble of models to characterise a particular form of uncertainty. Within the petroleum industry and the TNO OLYMPUS Challenge it is common practice to represent geological uncertainty via an ensemble generated from a stochastic geology model or obtained through expert elicitation. The uncertainty in the expected NPV induced by geological uncertainty is denoted by ε_{G}. This form of uncertainty may be considered as analogous to model parameter uncertainty where the inputs represent various geological properties of the field and within specific regions. However, the high-dimensionality of the model parameter space makes this challenging to handle and hence it is difficult to accurately express a prior distribution that realistically represents the expert's beliefs. An alternative approach is therefore to formulate an ensemble of (geological) model parameter realisations generated from an underlying stochastic (geology) model. This is the case in the TNO OLYMPUS Challenge. The ensemble mean should be treated as an unbiased estimator for the true mean whilst acknowledging the induced uncertainty. For geological uncertainty and assuming an additive error structure, that is $\mathbb{E}[\varepsilon_{\mathrm{G}}] = 0$. The realisations may be treated as an exchangeable sample from an infinite number of geologies for which a more complex method of handling this form of uncertainty is discussed in [89]. We propose a simpler

approach using an initial collection of simulations over the entire ensemble of size N to produce a conservative estimate for $\mathrm{Var}[\varepsilon_{\mathrm{G}}]$ of $\frac{s^2}{N}$, where s is the sample standard deviation of the function of interest, such as the utility function. It is acknowledged that the ensemble mean NPV is an unbiased estimator for the expected NPV and so could be integrated out. Although a committee of decision makers may resolve this uncertainty by ordering simulations to be evaluated for a larger ensemble to obtain a substantial reduction in the uncertainty.

3.5.4 Decision Implementation Error

Decision implementation error, often abbreviated to implementation error, ε_{I}, refers to the uncertainty induced as a consequence of the difference between the controls suggested to the real world user and those which are actually enacted. This is often ignored in practice [58]. For example, in well control optimisation there may be a difference between the target production and injection rates provided and what the corresponding controls are set to on an oil platform. This is another form of uncertainty that can in principle be resolved through a superior understanding of the typical implementation errors, as well as by its practical reduction in the real world. Note that there are several similarities with observational errors for historic data. The relationship between the proposed decisions, $\mathbf{d}$, and those actually implemented, $\boldsymbol{\delta}_{\mathrm{I}}$, is modelled as in [31] using an additive error structure shown in Equation (3.5.4), where $\mathbf{e}_{\mathrm{I}}(\mathbf{d})$ is the uncertainty in the decisions themselves. This induces the implementation error uncertainty in the expected utility shown in Equation (3.5.5) which may depend on the decision.

$$\boldsymbol{\delta}_{\mathrm{I}} = \mathbf{d} + \mathbf{e}_{\mathrm{I}}(\mathbf{d}) \tag{3.5.4}$$

$$U(f(\mathbf{x}, \boldsymbol{\delta}_{\mathrm{I}}), \boldsymbol{\delta}_{\mathrm{I}}, \mathbf{c}) = U(f(\mathbf{x}, \mathbf{d}), \mathbf{d}, \mathbf{c}) + \varepsilon_{\mathrm{I}}(\mathbf{d}) \tag{3.5.5}$$

Specification of $\mathbf{e}_{\mathrm{I}}(\mathbf{d})$ requires elicitation from the relevant experts and often takes the form of a prior distribution, $\pi_{\mathrm{I}}(\mathbf{e}_{\mathrm{I}} \mid \mathbf{d})$, noting the possible explicit dependence on $\mathbf{d}$ due to other constraints such as practical bounds on the decision parameter.

A reasonable assumption is $\mathbb{E}[\mathbf{e}_{\mathrm{I}}] = \mathbf{0}$ representing that the users will attempt to implement the suggested decision strategy.

Unlike the utility function cost parameters, $\mathbf{d}$ is an input to both the utility function and the computer model rendering direct experimentation in the assessment of implementation error much more computationally demanding. Assuming that a sufficiently accurate emulator representation has been constructed for $f(\mathbf{x}, \mathbf{d})$ or $U(f(\mathbf{x}, \mathbf{d}), \mathbf{d}, \mathbf{c})$, and using the notation $\mathbb{E}_F[U(\mathbf{d})]$ irrespective of whether it is the simulator or utility function that has been emulated, then a simulation based method of approximately assessing this form of uncertainty up to a second-order belief specification is as follows.

1. Construct a space filling design, $\mathcal{D}_{\mathrm{I}}$, of decisions at which to assess the implementation error.
2. For each $\mathbf{d}_i \in \mathcal{D}_{\mathrm{I}}$, sample $\mathbf{e}_{\mathrm{I},i,j} \sim \pi_{\mathrm{I}}(\mathbf{e}_{\mathrm{I}} \mid \mathbf{d}_i)$ and compute $\boldsymbol{\delta}_{i,j} = \mathbf{d}_i + \mathbf{e}_{\mathrm{I},i,j}$ for $j = 1, \ldots, n_{\mathrm{I}}$.
3. Evaluate $\mathbb{E}_{\mathbf{F}}[U(\boldsymbol{\delta}_{i,j})]$, treating this as a surrogate for the actual utility value.
4. Investigate the bias due to implementation error by computing the sample mean and then approximating $\mathbb{E}[\varepsilon_{\mathrm{I}}(\mathbf{d}_i)] \approx m_{\mathrm{I},i}(\mathbf{d}_i) = \frac{1}{n_{\mathrm{I}}} \sum_{j=1}^{n_{\mathrm{I}}} \mathbb{E}_{\mathbf{F}}[U(\boldsymbol{\delta}_{i,j})]$.
5. Assess the variance of $U(\mathbf{d}_i)$ induced by implementation error via the sample variance. This is $\mathrm{Var}[\varepsilon_{\mathrm{I}}(\mathbf{d}_i)] \approx s^2_{\mathrm{I},i}(\mathbf{d}_i) = \frac{1}{n_{\mathrm{I}}-1} \sum_{j=1}^{n_{\mathrm{I}}} \left(\mathbb{E}_{\mathbf{F}}[U(\boldsymbol{\delta}_{i,j})] - m_{\mathrm{I},i}\right)^2$.

Graphical methods such as histograms and contour plots can be used to examine whether there exists a dependency of $\mathbb{E}[\varepsilon_{\mathrm{I}}]$ or $\mathrm{Var}[\varepsilon_{\mathrm{I}}]$ on $\mathbf{d}$. An assessment of $m_{\mathrm{I},i}$ can be used to detect any systematic uncertainty, with $\frac{m_{\mathrm{I},i}}{s_{\mathrm{I},i}}$ useful for identifying if the bias is of a substantial magnitude. Depending on the findings of the investigation, summary statistics such as high-valued quantiles may be used as an approximate upper bound on $\mathrm{Var}[\varepsilon_{\mathrm{I}}]$, alternatively variance emulation may be necessary [2, 3].

3.5.5 Utility Function Uncertainty

Utility function uncertainty concerns a lack of knowledge or understanding of the form of the true utility function. This is discussed in Section 3.4.4 as part of the justification for implementing a decision support framework, as opposed to an optimisation or decision theoretic approach. There are many reasons for the existence of utility function uncertainty including: difficulties in its elicitation; possible undisclosed judgements and risk preferences arising from the separation of decision makers and the analysts leading to uncertainties which cannot be resolved by the analysts; and the existence of numerous candidate utility functions provided by either an individual or a committee of decision makers. Note under certain formulations it may be possible to treat utility function uncertainty within a parametrised framework, in which case an uncertainty assessment similar to in Section 3.5.2 may be appropriate.

3.6 Iterative Decision Support Procedure

We present a novel iterative decision support procedure inspired by the history matching process described in Section 2.8 which provides the end user with a class of robust decisions with respect to the decision structure uncertainties included within the analysis. This makes explicit use of Bayesian emulators (of either a standard format discussed in Section 2.4, or of the hierarchical form exploiting known simulator behaviour presented in Section 3.3) to efficiently search the decision parameter space, as well as the uncertainty quantification techniques (see Sections 2.7 and 3.5) to robustly compare different strategies. This follows the research presented in our paper [135].

We begin by formalising the connection between decision support and real world decision making in Section 3.6.1. In Section 3.6.2 we introduce a decision support implausibility measure which is used within the iterative decision support algorithm presented in Section 3.6.3. The benefits of the procedure are then discussed including a comparison with the optimisation and decision theoretic approaches in

Section 3.6.4.

3.6.1 Link to Real World Decision Making

Decision support is performed for real world systems guided by the use of computer models that are encompassed within the utility function. Following the framework presented in Section 2.8, we formally link the committee of decision makers' actual utility function, denoted $U^y(\mathbf{d})$, an analogy of the true system state, $\mathbf{y}$, discussed in Section 2.8.1, to the analysts approximation of the utility function, $U(\mathbf{d})$. This is achieved through the introduction of the random quantity, $\varepsilon_{\text{DS}}(\mathbf{d})$, which represents the collective uncertainty on the decision making process and is defined as follows:

$$\varepsilon_{\text{DS}}(\mathbf{d}) = U^y(\mathbf{d}) - U(\mathbf{d}) \tag{3.6.1}$$

This is an amalgamation of the uncertainties regarding the decision framework and their implementation discussed in Section 3.5, with those concerning the computer model described in Section 2.7.

An example of the structure for $\varepsilon_{\text{DS}}(\mathbf{d})$ is to assume that the various sources of uncertainty are mutually uncorrelated and additive, as presented in Equation (3.6.2), where model discrepancy, $\varepsilon_{\text{MD}}(\cdot)$, is explicitly distinguished from the uncertainties pertaining to the decision structure.

$$\varepsilon_{\text{DS}}(\mathbf{d}) = \text{Var}[\varepsilon_{\text{MD}}(\mathbf{d})] + \sum_i \text{Var}[\varepsilon_i(\mathbf{d})] \tag{3.6.2}$$

Each source of uncertainty must be carefully quantified following procedures such as those discussed in Sections 2.7 and 3.5. For generality, this is presented for an explicit dependence on $\mathbf{d}$, although an uncertainty assessment may suggest a uniform specification is sufficient. Note that it is unnecessary to consider observations of the true utility in a similar way to system measurements in Section 2.8.1 because such information is only available after decision support is delivered and implemented.

3.6.2 Decision Support Implausibility Measure

In an analogous way to the history matching procedure of [29, 171], we define a decision support implausibility measure in Equation (3.6.3) based on the formal framework linking the analysis with the real world decision problem presented in Section 3.6.1. This is an analogue of the univariate history matching implausibility measure in Equation (2.8.4) and is used to facilitate decision support by eliminating regions of the decision parameter space which are unlikely to yield high values of $U(\cdot)$ within a certain tolerance. This involves comparing the adjusted expectation of the emulator for $U(\mathbf{d})$ with the maximum utility achieved across all simulations contained in design $\mathcal{D}$ for fixed utility function cost parameters, denoted by $U_{\max} = \max_{\mathbf{d}\in\mathcal{D}} U(f(\mathbf{x},\mathbf{d}),\mathbf{d},\mathbf{c})$. The indicator function, $\mathbb{1}_{\{.\}}$, is incorporated so that when the emulator adjusted expectation exceeds $U_{\max}$ the implausibility measure is equal to 0. If the emulator adjusted expectation does not exceed $U_{\max}$ there remains the possibility that $U(\mathbf{d}) > U_{\max}$ due to the uncertainties present. The magnitude of this deviation relative to the uncertainties is assessed by standardising with respect to the variance of the combined uncertainty. In Equation (3.6.3) it is assumed that the uncertainties on the decision making process and the emulator follow a mutually uncorrelated additive error structure, as obtained by combining Equations (2.7.2) and (3.6.2), where $\mathrm{Var}_F[U(\mathbf{d})]$ is the adjusted variance of the emulator for $U(\mathbf{d})$, $\varepsilon_{\mathrm{MD}}(\cdot)$ is the model discrepancy as described in Section 2.7.4, and each $\varepsilon_i(\cdot)$ represents any further sources of uncertainty included within the analysis such as those listed in Sections 2.7.2 and 3.5. This formula may be modified to reflect other uncertainty modelling structures.

$$I^2_{\mathrm{DS}}(\mathbf{d}) = \frac{(\mathbb{E}_F[U(\mathbf{d})] - U_{\max})^2}{\mathrm{Var}_F[U(\mathbf{d})] + \mathrm{Var}[\varepsilon_{\mathrm{MD}}(\mathbf{d})] + \sum_i \mathrm{Var}[\varepsilon_i(\mathbf{d})]} \cdot \mathbb{1}_{\{\mathbb{E}_F[U(\mathbf{d})]<U_{\max}\}} \qquad (3.6.3)$$

An alternative implausibility measure is seen in [31].

3.6.3 Iterative Decision Support Algorithm

We develop an iterative decision support procedure utilising the implausibility measure to identify a class of decisions up to a tolerance relative to the included uncertainties. At each wave a new non-implausible region is obtained as a subset of the previous using the implausibility measure and a threshold c. This is represented by a collection of decision parameter vectors selected using the Bayesian emulator. After the final wave the non-implausible region is termed the decision support region and consists of a collection of equivalent decisions up to the uncertainties present which may be qualified by performing simulations at these decision parameter settings. The decision maker can then apply their expert judgment to make the final selection, potentially resolving any residual uncertainty such as those discussed in relation to the utility function. Decisions outside this region are unlikely to be of interest to the decision maker as they are expected to yield low utilities given all the uncertainties.

The algorithm is initialised by defining the wave 0 non-implausible decision parameter space $\Omega_0 = \Omega$ before generating an initial design $\mathcal{D}_1 \subset \Omega_0$ at which simulations are performed and the utility function evaluated. In addition, a preliminary uncertainty analysis is performed. Compute $U_{\max} = \max_{\mathbf{d}\in\mathcal{D}_1} U(\mathbf{d})$. At each wave the algorithm proceeds as follows:

1. Construct an emulator for the utility function using simulations over the design $\mathcal{D}_k$ and evaluate over a large collection of $\mathbf{d} \in \Omega_{k-1}$.
2. Compute the decision support implausibility measure $I_{\mathrm{DS}}(\mathbf{d})$.
3. Use threshold c to determine a new non-implausible region $\Omega_k = \{\mathbf{d} \in \Omega_{k-1} \mid I_{\mathrm{DS}}(\mathbf{d}) \leq c\}$.
4. If either of the following conditions are satisfied:
   - Computational resources are exhausted.
   - There is negligible change in the volume of the non-implausible region.

   Return Ω_k, otherwise go to step 5.
5. Construct the next wave design $\mathcal{D}_{k+1} \subset \Omega_k$. There is the option to augment

the design with $\{\mathbf{d} \mid \mathbf{d} \in \mathcal{D}_k \cap \Omega_k\}$. Evaluate the simulator and utility function over $\mathcal{D}_{k+1}$. Redefine $U_{\max} = \max_{\mathbf{d} \in \bigcup_{i=1}^{k+1} \mathcal{D}_i} U(\mathbf{d})$. Optionally reformulate the uncertainty specification for the current non-implausible decision space. Return to 1.

Note that at each iteration $U_{\max}$ used within $I_{\mathrm{DS}}(\mathbf{d})$ is non-decreasing with the threshold c often chosen by appealing to Pukelsheim's 3-sigma rule, as discussed in Section 2.8.3 [142].

3.6.4 Comparison with Optimisation and Decision Theoretic Approaches

In this section we discuss the merits of the presented decision support methodology, before comparing to optimisation and decision theoretic approaches. The presented iterative decision support procedure shares many advantages with the history matching process and include:

- It is computationally efficient through the use of emulators and an inexpensive decision support implausibility measure,
- Only requires a second order belief specification of the expectations, variances and covariances within a Bayes linear framework, ensuring that this approach is robust to the challenges encountered in accurately specifying prior probability distributions and (conditional) likelihoods for all quantities involved, although the algorithm may also be implemented within a full Bayesian paradigm using the relevant summary statistics from the posterior distribution for the utility function,
- The use of a utility function allows for the trade-off between multiple quantities of interest which are not directly comparable whilst also incorporating the risk preferences of the decision maker(s),
- Permits the comprehensive exploration of the entire decision parameter space,
- Scalable to high-dimensional decision parameter spaces by allowing for the

introduction of new active decision parameters at each iteration and as a consequence of the iterative refocusing on the current non-implausible region,

- Provides a coherent framework in which to incorporate all relevant major sources of uncertainty, including accounting for how each uncertainty impacts on the utility surface and hence its relevance to decision support,
- Allows for the uncertainty quantification to be revised at each wave (in an analogous way to in the history matching process), hence it is expected that the magnitude of the uncertainty will decrease at later waves as the volume of the non-implausible decision space is reduced,
- Yields a class of non-implausible decision parameter vectors with respect to all of the included uncertainties,
- The committee of decision makers, who the analyst may not have access to, are able to incorporate and resolve any additional uncertainties at a later time point which were either undisclosed, unobtainable or initially unknown during the preliminary consultation phase of the analysis,
- It is quick and simple to assess the consequences of alternative uncertainty specifications, as well as reduce the uncertainty variances at a later stage when additional information may become available.

In contrast, optimisation techniques can be extremely computationally expensive requiring large numbers of evaluations of the computer model, particularly as the dimensionality of the problem increases. This is exacerbated for supposed robust optimisation addressed via ensemble optimisation routines, as discussed in Section 3.1.3, for which the use of ensembles to represent certain uncertainties further increases the number of required simulations, whilst still only providing a limited account of such uncertainties. Furthermore, optimisation techniques generally fail to solve the actual problem of interest by treating output of the simulator as equal to that for the real physical system. Some of these issues are addressed in Bayesian optimisation through the use of a surrogate model, however all of these methods are susceptible to the problem of becoming stuck at local optima which is negated in the iterative

decision support procedure via a comprehensive search of the decision space.

Decision theory provides a much more comprehensive treatment of uncertainty compared with optimisation, although this is via the expected utility function through the integration with respect to the prior uncertainty distribution(s) and so relies on accurate specification. This may be suitable where the analyst is the decision maker, but is inappropriate if there is a separation between the analyst and decision maker. In addition, these integrals are often computationally expensive to evaluate, frequently requiring approximations, whilst the maximisation of the expected utility still necessitates the use of optimisation routines with their associated limitations. Only decision support is able to properly handle unresolved uncertainties stemming from the real world consultation process such as those due to the specification of a utility function through the provision of a class of acceptable decision parameter vectors. This ensures that any decision selected from this collection is robust both to all of the uncertainties incorporated within the analysis, but also any additional unknown uncertainties for which there may exist further information that is unknown to the analyst.

Chapter 4

TNO OLYMPUS Well Control Optimisation

In this chapter we apply the statistical methodology developed and presented in Chapter 3 to the TNO OLYMPUS Well Control Optimisation Challenge which is reformulated as a decision support problem. This application is based on research presented at the 17$^{\text{th}}$ European Conference on the Mathematics of Oil Recovery (ECMOR XVII) 2020 and published in [135].

The TNO OLYMPUS Well Control Optimisation Challenge is introduced in Section 4.1 expanding on the overview provided in Section 3.1.1 as motivation for the developed methodology and including an exploratory analysis. An implementation of efficient geological ensemble subsampling is demonstrated in Section 4.2 with a method for constructing a targeted Bayesian design described and used in Section 4.3. Bayes linear emulation methodology presented in Section 2.4.5 is applied directly to the utility function in Section 4.4 before comparing with the potential improvements which can be achieved by exploiting known structures in the behaviour of simulator output in Section 4.5. An uncertainty analysis combining the frameworks laid out in Sections 2.7 and 3.5 is applied to the TNO OLYMPUS model in Section 4.6. Both forms of emulator along with the uncertainty quantification are then incorporated within the novel iterative decision support framework in Section 4.7 along

with an assessment and commentary on the consequences of alternative uncertainty specifications. This is summarised in a conclusion in Section 4.8.

4.1 Introduction to the TNO OLYMPUS Well Control Optimisation Challenge

The TNO OLYMPUS Field Development Optimisation Challenge [93] is first introduced in Section 3.1.1 to serve as motivation for the development of an iterative decision support procedure along with the associated uncertainty quantification techniques, and emulation methodology exploiting known simulator behaviour in Chapter 3. Further details of the TNO OLYMPUS Well Control Optimisation Challenge including a setup specific to our application are provided in Section 4.1.1 with a discussion of an exploratory analysis of a preliminary batch of simulations in Section 4.1.2.

4.1.1 TNO OLYMPUS Well Control Optimisation Challenge

The aim of the TNO OLYMPUS Well Control Optimisation Challenge is to design a well control strategy, represented by a vector of decision parameters, for the fictitious TNO OLYMPUS oil reservoir model which maximises the expected NPV objective function over the 20 year lifetime of the field (starting January 1, 2016) with discounting applied at 3 month intervals, under the uncertainty captured by the $N = 50$ geological realisations contained within the OLYMPUS ensemble. For each ensemble member, the same fixed well configuration provided by TNO is used. The NPV objective function with decision parameters $\mathbf{d}$ as inputs for each individual model is defined in Equation (3.1.1), with $R_j(\mathbf{d}, t_i)$ encapsulating the difference between all revenue accrued and expenditure incurred within each discounting time interval defined in Equation (3.1.3). Full details of the terms contained within each

formula can be found in Section 3.1.1. Once the NPV has been computed for each OLYMPUS model, the expected NPV is approximated by the ensemble mean NPV defined in Equation (3.1.2).

The TNO OLYMPUS Well Control Optimisation Challenge is framed as a test of ensemble optimisers, however for our application we reformulate the challenge as a decision support problem aiming to address many of the criticisms discussed in Section 3.1.2. This includes a comprehensive uncertainty quantification to highlight the effects of omitting many of the relevant sources of uncertainty from the analysis, as is the case in the original setup, which is then incorporated within our novel iterative decision support procedure to identify a collection of decision parameter vectors that are robust to the uncertainty specification. We specify our utility function as $U(\mathbf{d}) = \mathbb{E}[\text{NPV}](\mathbf{d})$, noting that this represents a position of risk neutrality. In addition, note that TNO do not provide access to the underlying stochastic geology model for the OLYMPUS reservoir and hence this limits our ability to perform experiments directly on this model, as well as to validate any identified decision strategies.

For demonstrative purposes, we focus on the control of a subset of the wells that are contained between two partial fault boundaries and are in close proximity consisting of two producer wells: 2 & 10, and two injector wells 2 & 3. Collectively these wells are referred to as the Controlled Wells Group (CWG) which provides a sub-problem of interacting wells on which to illustrate the methodology presented in this thesis. All remaining wells within the OLYMPUS model use the fixed controls specified in the TNO reference strategy [93]. The utility and objective functions are both computed from contributions of wells in the CWG only.

A clear distinction should be made between discounting and control intervals with the latter formed by amalgamating multiple consecutive discounting intervals. For wave 1 of our application we choose to use eight control intervals of lengths 2, 2, 2, 2, 2, 2, 4 and 4 years. This corresponds to decisions starting on January 1, 2016, 2018, 2020, 2022, 2024, 2026, 2028 and 2032 respectively, resulting in a total

of 32 decision parameters across the four wells of the CWG. Constructing control intervals which combine several discounting intervals has the effect of reducing the dimensionality of the original optimisation or our decision support problem. Within our iterative decision support procedure, these control intervals may be split into shorter intervals at later waves (if it is deemed necessary) in order to address the full challenge.

Participants in the TNO OLYMPUS Well Control Optimisation Challenge may also select the well control types for which we elect to use target production and injection rates as decision parameters with daily operational rate constraints detailed in Table 4.1.

Type	Minimum	Maximum	Unit
Platform Liquid Production Rate	0	14000	m^3/day
Well Oil Production Rate	0	900	m^3/day
Well Water Injection Rate	0	1600	m^3/day

Table 4.1: TNO OLYMPUS Well Control Optimisation Challenge operational rate constraints.

Throughout this application the notation "prod_x_yyyy_mm" and "inj_x_yyyy_mm" are used to denote the target production and injection rate respectively for well number "x" starting on the 1^{st} of month "mm" in year "yyyy".

Numerous types of simulator output are introduced, discussed and modelled throughout this thesis, particularly in relation to computing the NPV. We adopt a simulator output naming convention which is relevant to both the TNO OLYMPUS model and the Emerson Jade model applications where acronyms are constructed in the form shown:

{Component}{Fluid-Type}{Attribute}

1. Component – F (field), G (group) or W (well).
2. Fluid-Type – O (oil), W (water) or G (gas).
3. Attribute – Properties reported by the simulator, for example: PR or PT for Production Rate or Total respectively; and IR or IT for Injection Rate or Total

respectively.

An example is FOPT for Field Oil Production Total. This is extended to specifically refer to simulator output within a certain time (or discounting or control) interval as follows:

{Output}{Well}{Number}_{Date}

1. Output – Simulator output type of the above specified form.
2. Well – Well type, either PROD (producer) or INJ (injector).
3. Number – Well number.
4. Date – Date of end of interval in the form "yyyymmdd" or "yyyymm" when it is implicitly assumed that the end-date is the 1st of the month.

An example is WOPTPROD2_20180101 which refers to the WOPT for producer well 2 in the control period ending 1st January 2018, noting that the control interval started on 1st January 2016. Further useful acronyms in the petroleum industry are listed in Table 4.2.

Acronym	Definition
NPV	Net Present Value
(W)BHP	(Well) Bottom Hole Pressure
FWCT	Field Water Cut
FOIP	Field Oil In Place
PROD# or PROD.#	Producer Well Number #
INJ# or INJ.#	Injector Well Number #

Table 4.2: Acronyms used in the analysis of the TNO OLYMPUS Field Development Optimisation Challenge.

In our analysis, all simulations of the OLYMPUS ensemble were performed using the open source simulation software, Open Porous Media (OPM) Flow [161]. On average, each simulation takes approximately 30 minutes to complete, hence a single decision parameter vector simulated for all 50 OLYMPUS models takes approximately 25 hours.

4.1.2 OLYMPUS Exploratory Analysis

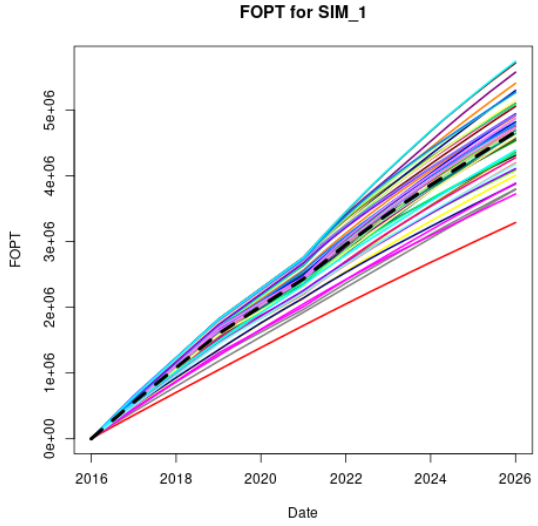
In the analysis of any computer model, it is important to perform an exploratory analysis of a small, preliminary batch of simulations, if possible, with the option to use a shorter time horizon. This should be undertaken prior to constructing a design for either the main collection of simulations or the wave 1 simulations within an iterative analysis; the exploratory simulations may be considered as a wave 0. The aim of an exploratory analysis is to identify any features of the model which may be relevant for constructing a more informative design. Furthermore, in situations where an ensemble of models is used, as is the case in the TNO OLYMPUS Well Control Optimisation Challenge, a preliminary batch of simulations enables investigations into whether there exists a representative subset of ensemble members. This leads to a more efficient use of the available computer resources.

For reasons of available time and limits on computational resources, we restrict to a relatively small exploratory design of 20 decision parameter vectors over a shorter time period of the first 10 years of the OLYMPUS field lifetime. This is justifiable within the challenge setup as TNO stipulate a high discounting rate of 8%, hence after 10 years the income accrued and expenditure incurred have already decreased in value to only 46.32% versus present day. Consequently any gross revenue in the second half of the OLYMPUS field lifetime accounts for less than half the value of monetary gains and costs during the early time periods. This agrees with conventional petroleum engineering principles where the aim is usually to extract larger volumes of oil at early times within the field lifetime to mitigate high discounting effects. For the exploratory simulations the decision parameters are target rates for wells in the CWG at times of January 1, 2016, 2017, 2019, 2021 and 2023, corresponding to control intervals of lengths 1, 2, 2, 2 and 3 years respectively. This is a total of 20 decision parameters, for which it is not reasonable to use more due to the size of the design. Maximin Latin hypercube sampling, as discussed in Section 2.3, was used to generate the design. Note that a larger collection of simulations would be preferable if more computational resources were available. Simulation output for each decision

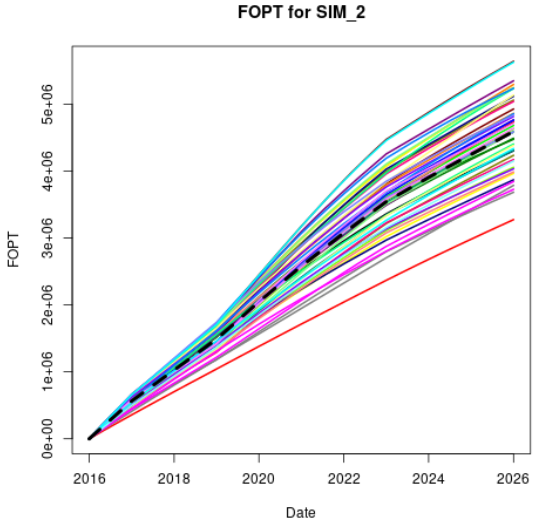
parameter vector is labelled as SIM_1, . . . , SIM_20.

Exploratory simulations can be used to provide a better understanding of the mechanics of the system and its computer model. Firstly, these are used to highlight the variation over the OLYMPUS ensemble for a selection of cumulative field outputs with time in Figure 4.1, and for WOPT for producer well 10 versus time in Figure 4.2. In each plot the various coloured lines represent one of the 50 ensemble members evaluated for a single choice of decision parameter vector, whilst the black dashed line represents the ensemble mean output. It is observed that each output type diverges from its mean at later times, although the sample standard deviation at a fixed time point corresponds to approximately 10% of the mean value. A comparison of Figures 4.1a and 4.1b, as well as Figures 4.2a and 4.2b, illustrate the difference in the observed behaviour depending on the well control strategy. Moreover, it is evident that oil production rate generally decreases with time whilst water production rate increases, as seen in Figure 4.1c. Meanwhile, the water injection rate is approximately constant in order to maintain the pressure in the well, as seen in Figure 4.1d.

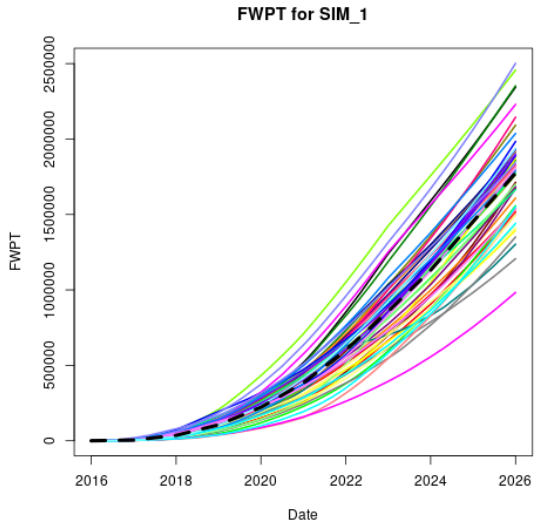
An interesting facet with potential ramifications for emulation and decision support is the vastly different relative absolute contributions of oil and water to the NPV objective function. An assessment of these absolute contributions approximated within one year intervals for the OLYMPUS 25 NPV is shown in Figure 4.3 where each of the 20 exploratory analysis decision parameter vectors are represented by different colours. The solid lines depict the oil contribution, $|Q_{j,op}(\mathbf{d}, t_i) \cdot r_{op}|$ in Equation (3.1.3), for which it is immediately evident that this is dominant versus both the absolute water production contribution (dot-dashed lines), $|Q_{j,wp}(\mathbf{d}, t_i) \cdot r_{wp}|$ in Equation (3.1.3), and injection contribution (dotted lines), $|Q_{j,wi}(\mathbf{d}, t_i) \cdot r_{wi}|$ in Equation (3.1.3), as well as their sum (dashed lines), $|Q_{j,wp}(\mathbf{d}, t_i) \cdot r_{wp} + Q_{j,wi}(\mathbf{d}, t_i) \cdot r_{wi}|$. For earlier time intervals the magnitude of the oil contribution to the NPV is typically of the order of 100 times the combined water contribution which decays towards 10 times larger for later time intervals. Plotting on the logarithmic scale in Figure 4.3b facilitates an easier comparison of the water contributions. It is observed



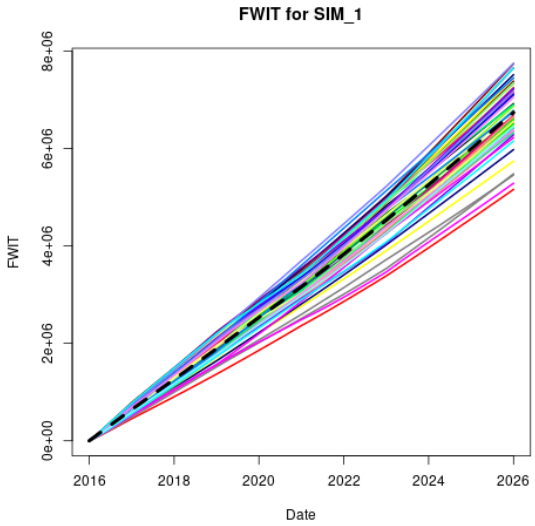
(a) FOPT for SIM_1.



(b) FOPT for SIM_2.



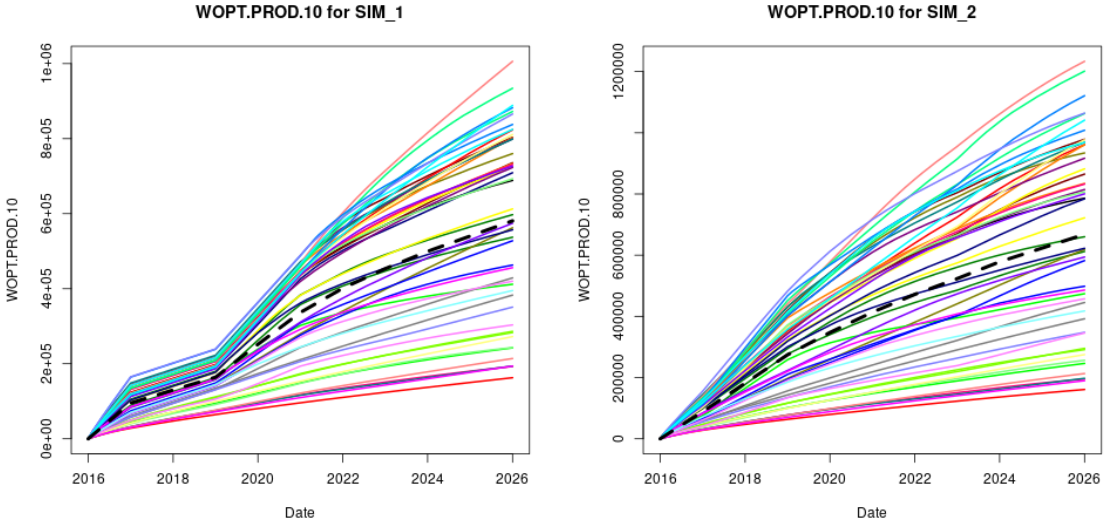
(c) FWPT for SIM_1.



(d) FWIT for SIM_1.

Figure 4.1: OLYMPUS cumulative field outputs versus time for exploratory simulations where each line shows the variation over the full ensemble of 50 geological realisations. The black dashed line indicates the ensemble mean cumulative output.

that water injection contributes a much larger amount to the NPV, particularly for earlier time intervals. This is to be expected since production wells are drilled within regions containing a high oil concentration, hence at initial times there should be very little water production. At later times the contribution becomes more alike as an increased quantity of water is produced in order to maintain oil production,

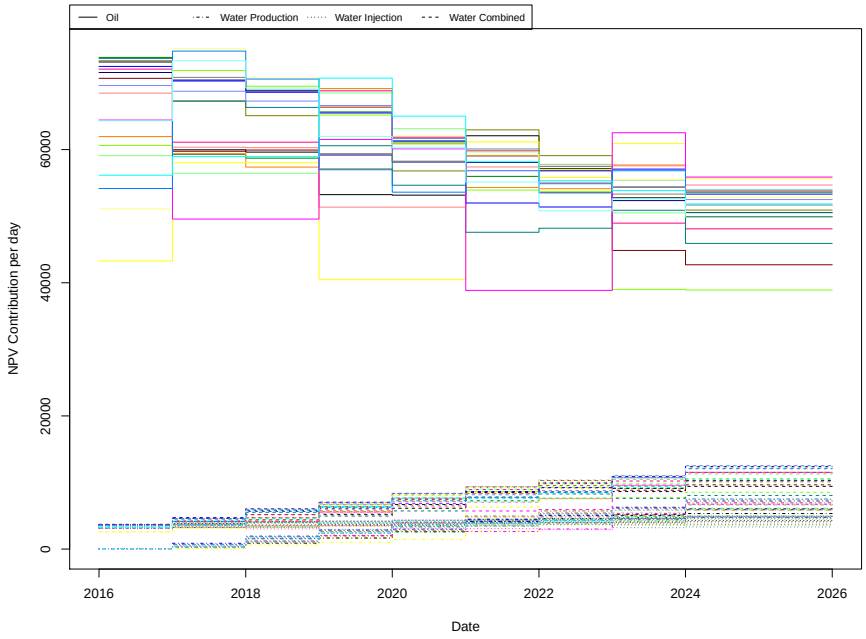


(a) WOPTPROD10 for SIM_1. (b) WOPTPROD10 for SIM_2.

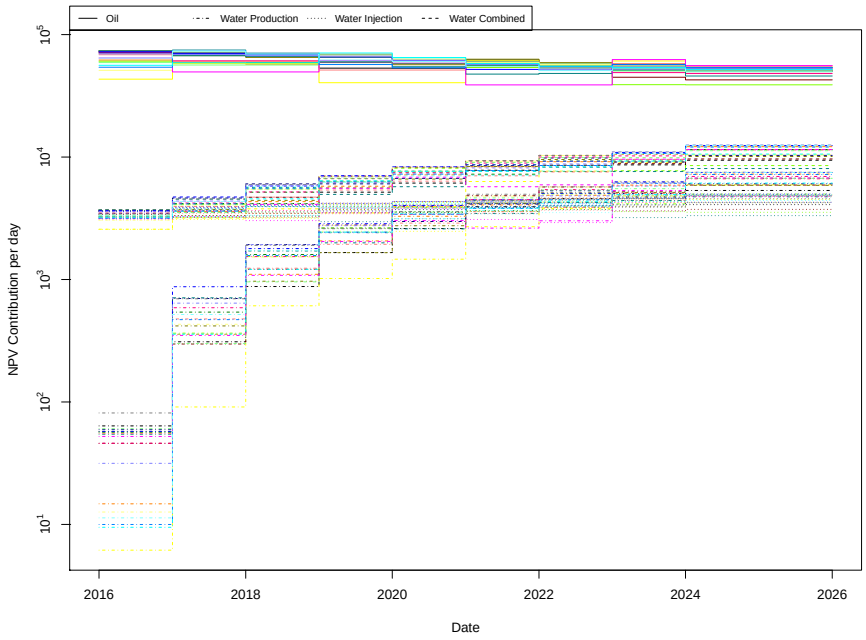
Figure 4.2: OLYMPUS WOPTPROD10 versus time for the exploratory simulations where each line shows the variation over the full ensemble of 50 geological realisations. The black dashed line indicates the ensemble mean WOPTPROD10.

whilst also noting the higher fixed cost per barrel of water produced versus injected. Similar observations are made for other OLYMPUS models.

Another feature which was analysed using the exploratory simulations is the adherence of the simulator to target production and injection rates. Examples of the plots used are shown in Figure 4.4 of the simulated rates versus time for the 50 ensemble members, with the decision targets denoted by the black dashed lines. It is immediately evident in all plots that the input targets are not strictly adhered to; a consequence of the underlying physics programmed into the simulator including imposed constraints such as on BHP resulting in deviations between observed and selected control values. For producer wells the BHP must not fall too low (minimum of 150 bar), hence it is easier to achieve a desired target production rate at earlier times, and whilst the injection rates are sufficiently high. This can be seen up to the start of 2021 in Figures 4.4a and 4.4b, noting that there exists some variation due to the differing geologies, especially when a high target production rate is requested. In later time intervals there is a persistent struggle to maintain BHP, particularly for

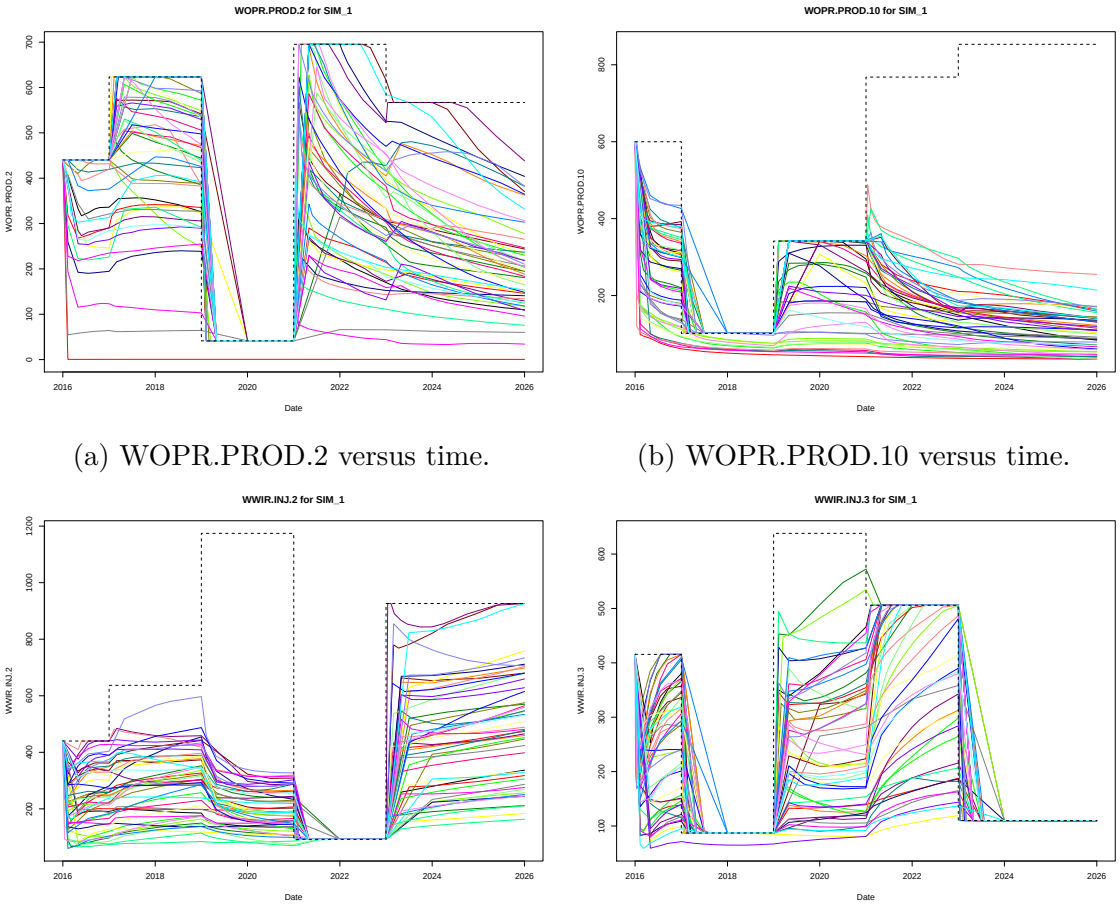


(a) Raw scale.



(b) Logarithmic scale.

Figure 4.3: OLYMPUS 25 approximate absolute contribution to the NPV per year for each of the exploratory simulations represented by different coloured lines. The NPV is decomposed into the oil production (solid line), absolute water production (dot-dashed line) and injection (dotted line), and the total water contribution (dashed line), with each scaled by the respective fixed NPV cost parameter. These are $|Q_{j,op}(\mathbf{d}, t_i) \cdot r_{op}|$, $|Q_{j,wp}(\mathbf{d}, t_i) \cdot r_{wp}|$, $|Q_{j,wi}(\mathbf{d}, t_i) \cdot r_{wi}|$ and $|Q_{j,wp}(\mathbf{d}, t_i) \cdot r_{wp} + Q_{j,wi}(\mathbf{d}, t_i) \cdot r_{wi}|$ in Equation (3.1.3) respectively. The top and bottom plots are on the raw and logarithmic scale respectively.



(a) WOPR.PROD.2 versus time.

(b) WOPR.PROD.10 versus time.

(c) WWIR.INJ.2 versus time.

(d) WWIR.INJ.3 versus time.

Figure 4.4: Comparison of the target and achieved rate controls for the OLYMPUS model using the first exploratory simulation. Each plot shows the simulator output as coloured lines for each ensemble member of the well oil production rates (top) or water injection rates (bottom) within the CWG versus time. This is compared with the black dashed lines that show the corresponding decision parameters of target production or injection rates respectively over time. The differences between the output traces and the inputs highlights the deviations between the target control and what is achieved due to physical constraints applied in the model.

producer well 10. For injector wells the BHP must not be too high (maximum of 235 bar) which would be exceeded as a result of high water injection whilst not extracting sufficient quantities of fluid from the field. Consequently at earlier time intervals it is harder to maintain a requested injection rate, whilst at later times this can be achieved. Examples of this behaviour can be seen in Figures 4.4c and 4.4d. Such

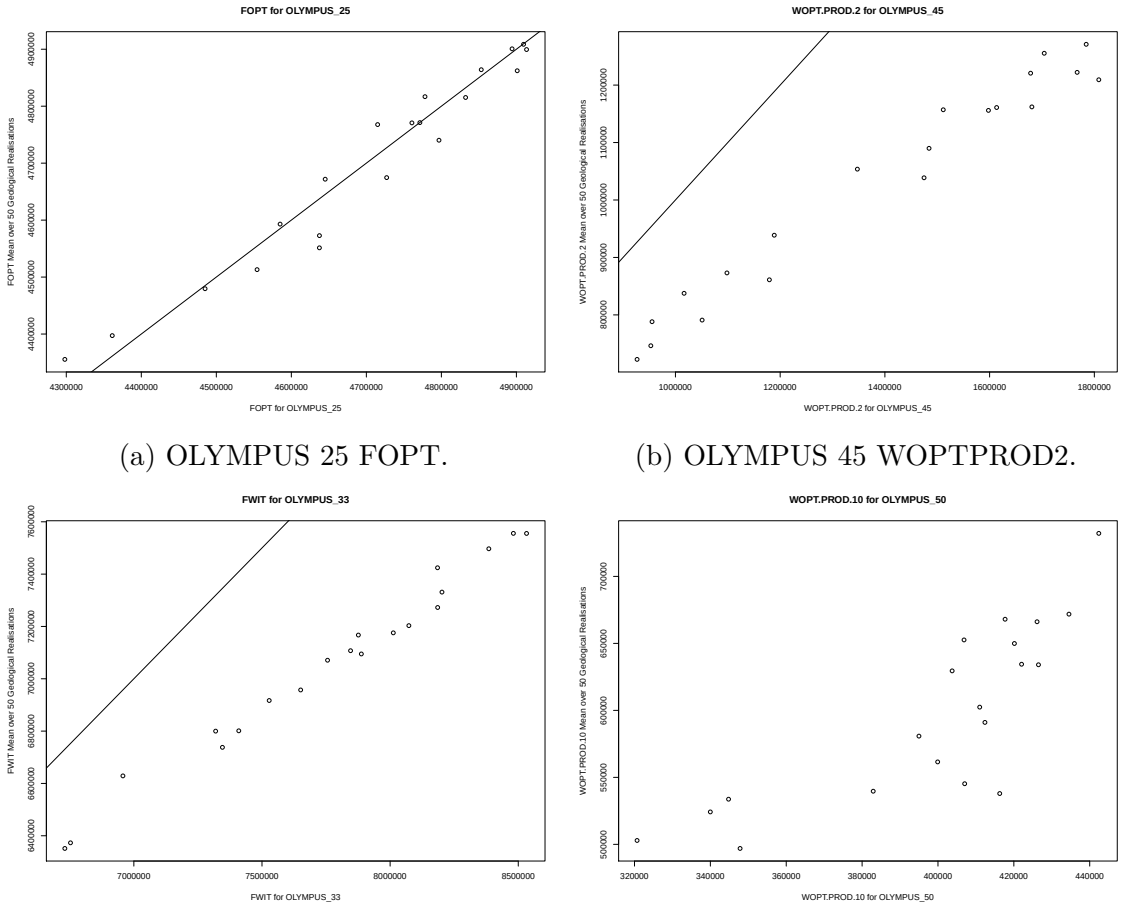
observations of two distinct modes of behaviour motivates structured emulation of these outputs within time intervals in Section 4.5. Note that another characteristic which is observed in these plots is an occasional apparent lag between the new target rate and it being recorded in the simulator output. This is not actually the case and is a consequence of the numerical solver step-sizes which is addressed prior to the first wave of simulations for decision support.

4.2 Application of the EGES Techniques

In this section we apply the efficient geological ensemble subsampling (EGES) techniques described in Section 3.2 utilising the exploratory simulations discussed in Section 4.1.2 to identify a representative subset of the $N = 50$ OLYMPUS models. This is a novel application of such ensemble subsampling techniques within the petroleum industry.

Preliminary graphical investigations utilise plots of the ensemble mean versus the individual model outputs for which a selection are shown in Figure 4.5, where the black line denotes equality between the ensemble mean and individual ensemble member model output. The main outputs of interest stem from the NPV objective function and include: the ensemble mean NPV, oil production, water production and injection total, both for the field and for wells within the CWG.

It is not necessary to sub-select models for which each of the individual model outputs are close to the ensemble mean, but instead identify cases where the relationship is easier to model, for example, there is a preference for linear associations. It is also desirable that the output variation is relatively small. Note that this is a preliminary graphical assessment which is limited to identifying one-dimensional relationships. This is observed in Figures 4.5a to 4.5c which show strong linear relationships with fairly limited variation providing evidence that even as individual models, OLYMPUS 25, 33 & 45 are potentially representative for the ensemble mean. An appropriate (linear) transformation may be applied in the cases seen in



(a) OLYMPUS 25 FOPT.

(b) OLYMPUS 45 WOPTPROD2.

(c) OLYMPUS 33 FWIT.

(d) OLYMPUS 50 WOPTPROD10.

Figure 4.5: Preliminary graphical investigations for Efficient Geological Ensemble Subsampling consisting of plots of the OLYMPUS ensemble mean versus individual model outputs. The black line denotes equality between the ensemble mean and individual model outputs.

Figures 4.5b and 4.5c. In contrast OLYMPUS 50 does not appear to be a good representative model, at least individually, as seen in Figure 4.5d where the relationship is more challenging to model. This graphical investigation is also useful as a preliminary screening technique to obtain a subset of models to investigate further and is particularly useful for large ensembles, especially when there are limited distinct decision parameter vectors, as is the case for our application to the TNO OLYMPUS Well Control Optimisation Challenge. Based on an analysis of all outputs of interest, this leads to the proposal of ensemble members OLYMPUS 2, 6, 11, 23, 25, 33, 35, 37 & 38 for further investigation. Note that this does not contain OLYMPUS 45

which will be discussed below.

In order to capture the interacting effects of the different OLYMPUS models, the efficient geological ensemble subsampling technique using the linear model described in Section 3.2.2 is implemented. This is first applied to a proposed subset of the OLYMPUS models, before extending to all possible models via both directions stepwise selection starting from the full model and using AIC as the model selection criterion, as presented in Appendix A. It is established that a subset of only $N_{\text{EGES}} = 3$ models is sufficient for a large number of the investigated outputs, as demonstrated in Figure 4.6 of the linear model adjusted R^2 values for various outputs. All are high with most greater than 0.95 which implies the majority of the variation in the ensemble mean is explained by a small subset of the OLYMPUS models. These are OLYMPUS 25, 33 & 45 which are subsequently used for wave 1 of iterative decision support. It is noted that OLYMPUS 25 & 33 were identified as part of the proposed subset of models, whilst OLYMPUS 45 was not. This is because for certain outputs it was judged that OLYMPUS 45 was not a sufficiently good representative for the ensemble mean, however in combination with OLYMPUS 25 & 33 via the EGES linear models it is found that this subset characterises the ensemble mean very well, and in particular, the ensemble mean NPV. This demonstrates the value of the EGES linear model construction for identifying combinations of ensemble members which together form an adequate representation of the ensemble mean. The linear model constructed for predicting the ensemble mean NPV using the simulated NPV for OLYMPUS 25, 33 & 45 only is referred to as the EGES linear model which is used as an efficient method of prediction with a corresponding quantification of the uncertainty induced by only using 3 ensemble members instead of the all 50.

4.3 Targeted Bayesian Design

In this section we present a novel Bayesian approach to constructing a design of simulations tailored to decision support which develops on traditional space filling

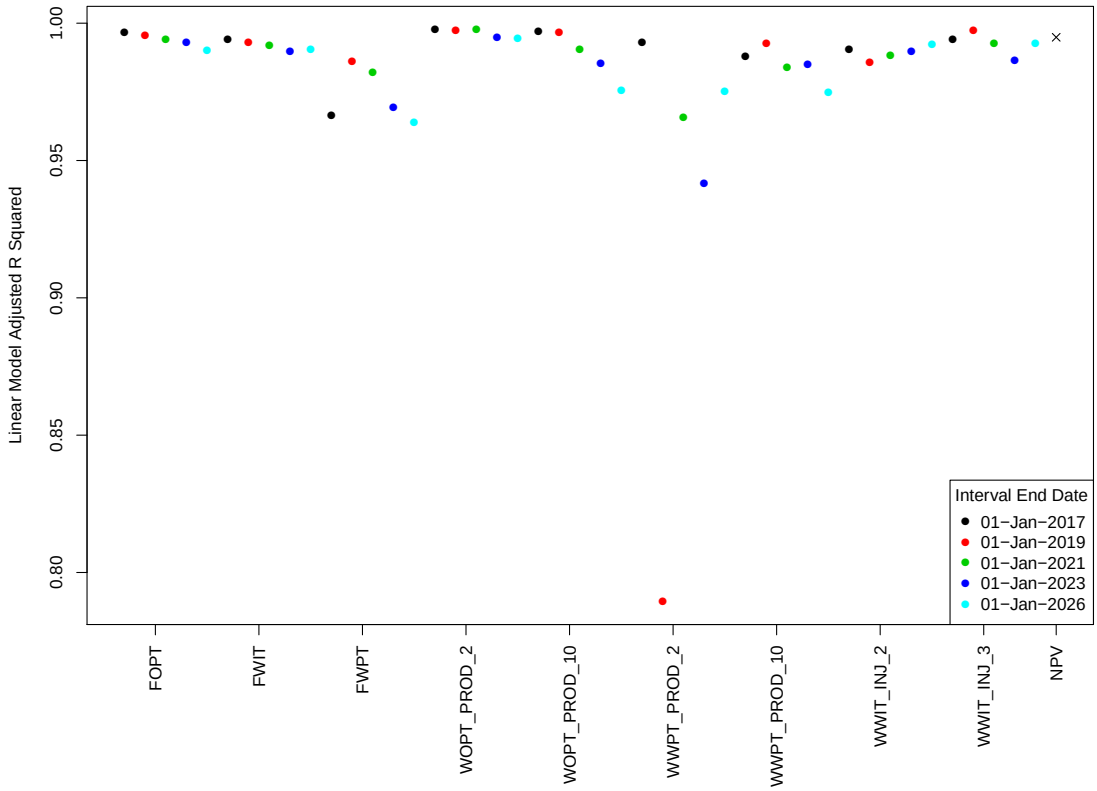


Figure 4.6: Adjusted R^2 values for the OLYMPUS EGES linear models of the form in Equation (3.2.1) for the ensemble mean of various outputs within control intervals using the same subset of $N_{\text{EGES}} = 3$ OLYMPUS models as predictors.

designs (see Section 2.3) to incorporate additional constraints and sample based on prior beliefs regarding the optimal decision. This is useful but not essential for providing decision support by reducing the number of simulations in regions which a priori are not believed to contain the best decision, thus reducing the overall computational expense. Two different types of constraints relevant to well control optimisation are described in Section 4.3.1 followed by an algorithm for sampling based on these constraints and incorporating prior beliefs. Results for the TNO OLYMPUS Well Control Optimisation Challenge are then presented in Section 4.3.2.

4.3.1 Prior Constraints and Targeted Bayesian Design Methodology

Within the analysis of computer experiments, both within a Bayesian or frequentist framework, it is common practice to specify a finite range for each input parameter. In the context of optimisation or decision support for petroleum reservoir well control optimisation, these take the form of operational range constraints providing a minimum and maximum value for each decision parameter. The result is a (decision) parameter space which is a hypercube for which there exists extensive literature for constructing (approximately) optimal (space filling) designs, with a review found in Section 2.3.

Traditional space filling designs are useful where parameters are considered mutually independent however this is not always the case, as seen in the well control optimisation problem where parameters are target rates or pressures for which large temporal variation seems unphysical and is deemed poor practice by oil reservoir engineers. In addition to operational range constraints, this introduces a difference constraint between time consecutive control parameters where for time ordered decision parameters $|d_i - d_{i-1}| \leq \Delta_i,\ i = 2, \ldots, D$. Note that the decision parameter space is no longer a hypercube.

Adhering to such constraints motivates the use of a Bayesian design informed by these prior stipulations involving sampling the sum of the parameters and their consecutive differences. Since each parameter is bounded it can be linearly transformed to any range, thus assume each $d_i \in [0, 1]$. Define $t = \sum_{i=1}^{D} d_i \in [0, D]$ to be the sum of the parameters and $\delta_i = \frac{d_i - d_{i-1}}{\sqrt{2}}$ for $i = 2, \ldots, D$. Scaling the differences by $\frac{1}{\sqrt{2}}$ is required due to the rotation of the parameter space represented by this alternative parametrisation. Note that new parameters are mutually orthogonal with $t \in [0, D]$ and $|\delta_i| \leq \Delta_i' = \frac{\Delta_i}{\sqrt{2}}$. This re-parametrisation is the linear transformation $(t, \boldsymbol{\delta})^{\mathrm{T}} = L\mathbf{d}$ summarised in Equation (4.3.1) where L represents the transformation matrix. By sampling in the re-parametrised space the difference constraints are

automatically satisfied with a sample for $\mathbf{d}$ obtained via $\mathbf{d} = L^{-1}(t, \boldsymbol{\delta})^{\mathrm{T}}$ described in Equation (4.3.1). The range constraints, $d_i \in [0, 1]$, must also then be verified.

$$\begin{pmatrix} t \\ \delta_2 \\ \delta_3 \\ \delta_4 \\ \vdots \\ \delta_D \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 & 1 \\ -1/\sqrt{2} & 1/\sqrt{2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & -1/\sqrt{2} & 1/\sqrt{2} & 0 & \cdots & 0 & 0 \\ 0 & 0 & -1/\sqrt{2} & 1/\sqrt{2} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ \vdots \\ d_D \end{pmatrix} \tag{4.3.1}$$

Application of a uniform rejection sampling algorithm for this alternative parametrisation, it is observed that very few samples for small or large sums of the parameters, t, are accepted, thus failing to properly explore along the t-direction that is thought to be important. Instead we propose preserving an initial sample of the parameter sums, uniformly resampling a vector of the δ_i until both constraints are satisfied. There is freedom to choose the sampling distribution of t with probability density function denoted by $f_T(t)$ dependent on the aim of the analysis. Examples include: uniform; mixture of uniforms; truncated normal; or transformed beta distributions. Orthogonal projection of the samples onto the line $d_1 = d_2 = \cdots = d_D$ will approximately follow this distribution. The process of generating a sample of size n is described in the rejection style Algorithm 1 yielding matrix B in which each column is a sampled vector of decision parameters. An efficiency improvement is made by sampling the differences conditional on t where the rejection step can be computationally time consuming if t is close to 0 or D. This technique may be applied separately to independent subgroups of decision parameters for improved efficiency and combined using random permutations to optimise the final design with respect to a chosen design criterion such as those detailed in Section 2.3.5.

Algorithm 1: Sampling of parameter sums and differences preserving the initial sum of parameters sample.

Result: Matrix B of columns of sampled parameter vectors
Let $\mathbf{t}$ be a vector of n samples of $t \sim f_T(t)$;
Let B be an empty matrix of D rows;
Define $dimension(\cdot)$ to be a function to obtain the length of a vector;
while $dimension(\mathbf{t}) > 0$ **do**
  foreach t *in* $\mathbf{t}$ **do**
    Set $\epsilon_i = \min\{t, D - t, \Delta_i'\}$;
    Generate $\delta_i \mid t \sim \mathcal{U}[-\epsilon_i, \epsilon_i]$, for $i = 2, \ldots, D$;
  end
  Row bind $\mathbf{t}, \boldsymbol{\delta}_2, \ldots, \boldsymbol{\delta}_D$ to form matrix $B_{r,\text{prop}}$;
  Compute $B_{\text{prop}} = L^{-1} B_{r,\text{prop}}$;
  foreach *Column in* B_{prop} **do**
    if *Range conditions of parameter vector are satisfied* **then**
      Join *Column* to B;
      Remove corresponding t from $\mathbf{t}$;
    else
      Discard *Column*;
    end
  end
end

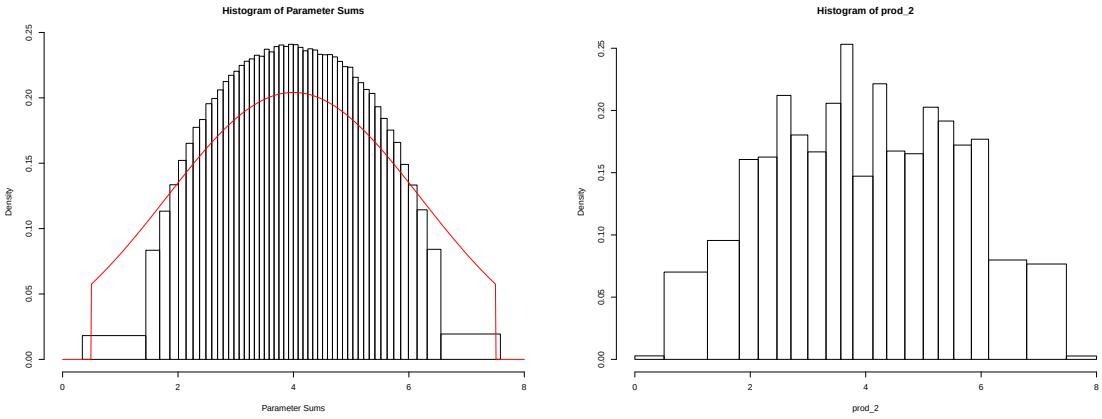
4.3.2 Wave 1 Design for Simulations

In this section we present a wave 1 design for the TNO OLYMPUS Well Control Optimisation Challenge including the two types of constraints and the design algorithm introduced in Section 4.3.1. As discussed in Section 4.1.1, for wave 1 we use a total of $D = 32$ decision parameters corresponding to each of the four controlled wells with eight control intervals starting on January 1, 2016, 2018, 2020, 2022, 2024, 2026, 2028 & 2032. TNO specify the operational ranges to be $[0, 900]\text{m}^3/\text{day}$ and $[0, 1600]\text{m}^3/\text{day}$ for production and injection rates respectively. Additionally we impose a difference constraint between time consecutive parameters for the same well to obtain physically realistic control strategies. A conservative choice is that the maximum permitted change over a two year time interval is $\Delta = \frac{1}{3}$ of the parameter range. Naturally this splits the design into four independent subgroups of eight decision parameters for each well. Moreover, note that imposing the difference constraint results in a reduction of the volume to approximately 3.4529% of the

initial hypercube due to the range constraints only.

Algorithm 1 is implemented to generate a $n = 700$ point design for each of the four subgroups of eight parameters which are approximately optimised with respect to the minimax design selection criterion (see Equation (2.3.4)) given a large $20,000$ point uniform random sample with which to compare a large collection of candidate designs. A priori a 700 point design is justified by the heuristic of sampling at least 10 times the dimension of the parameter space, $n \geq 10 \cdot D$ [116]. The normalised sum of parameters t is sampled from a truncated normal distribution with mean $\mu_t = 4$, standard deviation $\sigma_t = 2.2$, and truncation bounds of $[0.5, 7.5]$, noting that the maximal range is $[0, 8]$. Although this prevents samples within the corners where all decision parameters are set to their minimum or maximum respectively, this choice of truncation bound is justified via uniform sampling over the constrained decision space yielding 99.9995% of 10^6 points which lie within this interval, hence very little of the volume is contained within these two corner regions. Moreover, the choice of an absolute distance from these corners of 0.5 is based on prior beliefs that a reasonable correlation length for each parameter is half of its parameter range. Note that the correlation length will be re-assessed during the construction of the emulators once simulation data is available.

The mean hyperparameter is selected to be $\mu_t = 4$ corresponding to the mid-point of both the range of the normalised sum and the truncation interval, and matches the expectation under uniform sampling. The standard deviation hyperparameter σ_t is selected so as to ensure a reasonable number of samples are obtained for both low and high parameter sums with an assessment performed using histograms to achieve a desired shape for the sampling distribution. An example of this assessment is presented in Figure 4.7a comparing to uniform sampling. It is not expected that results of targetted sampling from $f_T(t)$ and from the uniform distribution align with the choice of $f_T(t)$ dependent on the aims of the analysis and any prior beliefs. We selected a value of $\sigma_t = 2.2$, corresponding to 0.275 per parameter, with the corresponding samples for producer well 2 shown in Figure 4.7b. This differs from



(a) Histogram of the sum of normalised parameters obtained via uniform sampling over the constrained decision parameter space for an individual well with the truncated normal density overlaid in red.

(b) Histogram of the sum of normalised parameters for producer well 2 sampled from a truncated normal distribution.

Figure 4.7: Histograms of the wave 1 design sum of normalised parameters. This includes a comparison of the chosen truncated normal distribution with truncation bounds $[0.5, 7.5]$, mean $\mu_t = 4$, and standard deviation $\sigma_t = 2.2$, this is $f_T(t)$ in Algorithm 1, with the results of uniform sampling over the constrained decision parameter space. The nature of employing a targetted design means that these are not expected to align.

uniform sampling over the constrained decision parameter space which tends to select parameter sums close to the mean as a result of the central limit theorem, and thus provides a nuanced way to sample assigning a modest preference to high and low values of t, which a priori is thought to be a major driver of the NPV constituent outputs. A comparison with the selected truncated normal density presented in Figure 4.7a illustrates this point. For decision support this allows for both the quick elimination of regions of low NPV, whilst also exploring regions that a priori are expected to produce high NPVs.

Figure 4.8 is a pairs plot of the 2-dimensional projection of the wave 1 design for producer well 2. The plots next to the axis labels clearly highlight the result is imposing the difference constraint as points are clustered between two clearly defined diagonal parallel bounds. Since the final two control intervals are of length 4 years a greater change is permitted, hence the wider bands. The effects of imposing the

difference constraints can also be seen between decision parameters with a larger time separation with fewer points away from the main diagonal, although it is less noticeable for greater time separations. Compared to uniform sampling there are more points sampled for smaller and larger parameter values, thus there is better exploration of the corner and edge regions of the decision space, but still sufficiently many points near the centre for constructing Bayesian emulators. This design is evaluated for only the subset of 3 OLYMPUS models identified in Section 4.2, with the EGES linear model used to predicted the ensemble mean NPV.

In Figure 4.8, the points are coloured by the simulated expected NPV predicted from EGES linear model where green, yellow and red points correspond to high, moderate and low NPVs respectively. Observations confirm reservoir engineering insight with high early time target production rates required to achieve a high NPV. As time progresses, the effect of discounting results in an increasing spread of red and yellow points across each parameters range until the final control interval, prod_2_2032_01, which encompasses the full range and hence a high NPV may be achieved for any control value when combined with suitable other decisions. A similar pairs plot is generated and checked for producer well 10 which is omitted for space, although the apparent association between high early time target rates is less substantive. This is indicative of the behaviour for producer well 2 being more important than producer well 10 in determining the overall NPV. Pairs plot for the two injector wells also possess the structure induced by the difference constraints, but with no dominant behavioural insight into which values are likely to yield a high expected NPV. Histograms of the sampled individual parameters are presented in Figure 4.9. Both plots are fairly uniform, although there is an apparent greater weighting on central values for inj_2_2022_01 in Figure 4.9b. This highlights that whilst the sampling method described in Algorithm 1 is not designed to produce uniform marginal samples, a close approximation is obtained.

The four independent sub-designs of size 700 and each of dimension 8 are combined using random permutations and approximately optimised with respect to the

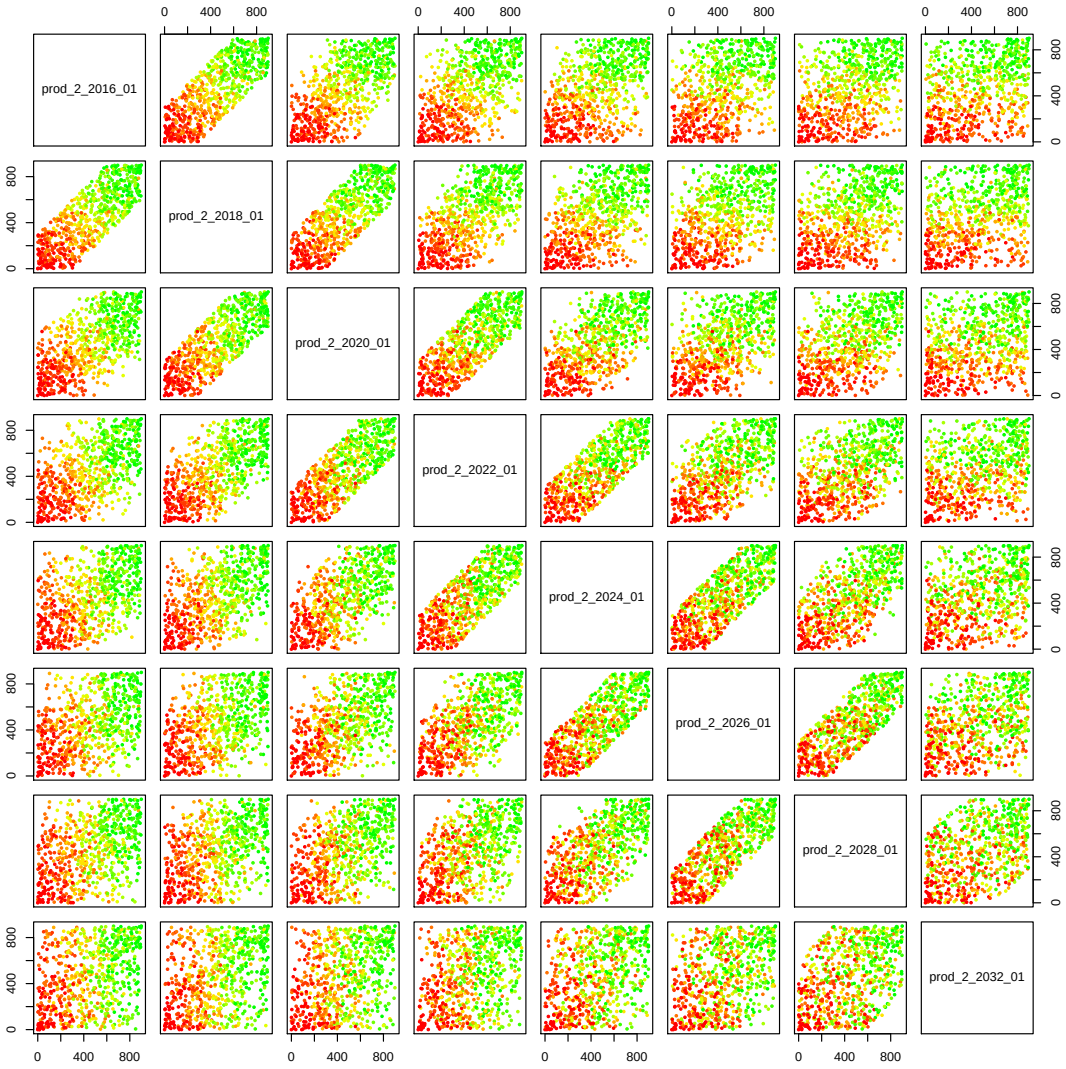
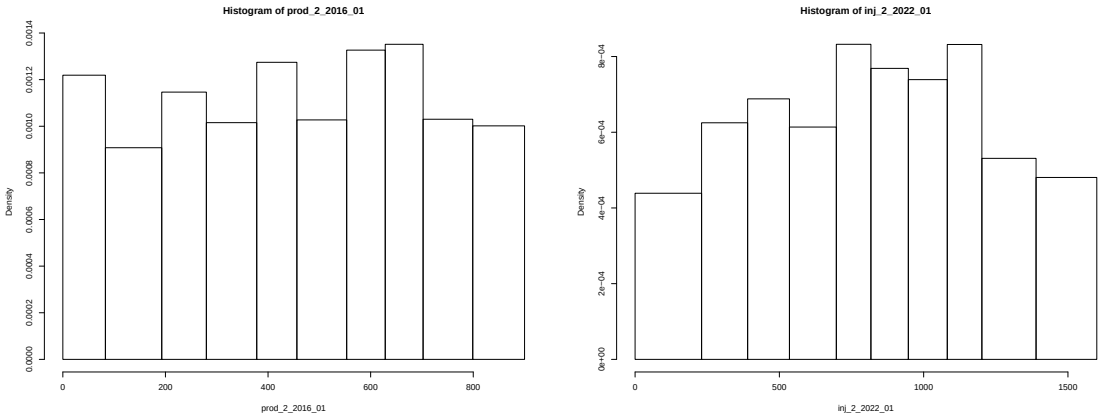


Figure 4.8: Wave 1 design pairs plots of the 702 well control strategies for producer well 2 for a design generated using Algorithm 1 and selected using the minimax distance criterion. A fixed initial sample of parameter sums from a truncated normal distribution was used when forming all candidate designs. Points are coloured according to the simulated NPV obtained from the EGES linear model with green, yellow and red corresponding to high, moderate and low NPVs respectively.

minimax design selection criterion. Moreover, the final selected design of size 700 is augmented to include two further decision parameter vectors with either all parameters set to their minimum or maximum values since it is of interest to observe the model behaviour at these extremes whilst also providing some exploration of these two extreme corner regions removed when sampling from the truncated nor-



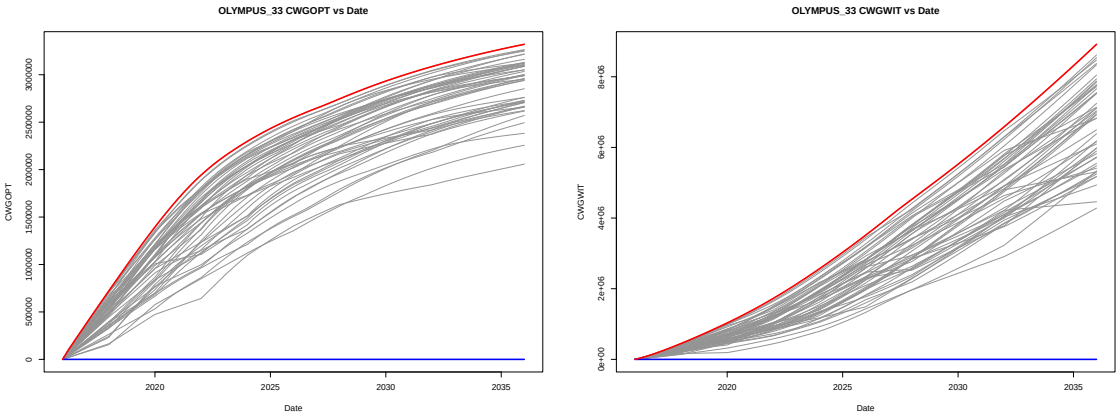
(a) prod_2_2016_01 (b) inj_2_2022_01

Figure 4.9: OLYMPUS wave 1 design marginal density histograms for prod_2_2016_01 and inj_2_2022_01 over the 702 well control strategies generated using Algorithm 1 and selected using the minimax distance criterion.

mal distribution, leading to a final wave 1 design of size 702 and evaluated for the identified subset of 3 OLYMPUS models. The CWGOPT and CWGWIT simulation outputs for OLYMPUS 33 are illustrated in Figure 4.10 for the first 52 design points, where the results for the minimum and maximum decision parameter settings are emphasised by the blue and red lines respectively, whilst the outcome of the next 50 simulations are represented by grey traces. These plots demonstrate the large variation in the outputs with time as a consequence of different decision parameter settings.

4.4 Bayes Linear Emulation of the Expected NPV

In this section we present the direct Bayes linear emulation of the expected NPV in order to explore the 32-dimensional wave 1 decision space following the methodology reviewed in Section 2.4.5. This analysis makes use of the EGES linear model to predict the expected NPV from Section 4.2 and the wave 1 design constructed in Section 4.3.2. Results are compared to the hierarchical emulation approach



(a) CWGOPT. (b) CWGWIT.

Figure 4.10: OLYMPUS 33 CWGOPT (left) and CWGWIT (right) versus date for the first 52 wave 1 simulations. The blue and red lines represent the simulation output where all decision parameters are set to the minimum or maximum respectively. All other simulations are represented by grey traces.

exploiting known simulator behaviour in Section 4.5.5, as well as for decision support in Sections 4.7.1 and 4.7.2.

An emulator with a nugget term of the form shown in Equation (2.4.1) is used where $f(\mathbf{d}) = U(\mathbf{d}) = \mathbb{E}[\text{NPV}](\mathbf{d})$ with decision parameters $\mathbf{d} \in \Omega \subset \mathbb{R}^{32}$ as inputs. The choice of active decision parameters and low order polynomial terms for the $g_j(\cdot)$ functions are investigated using linear regression modelling and stepwise selection techniques with the AIC criterion. Alternative model selection criteria are considered, although AIC is found to be parsimonious in identifying a sufficiently large set of active decision parameters and constructing a global mean function. Equation (4.4.1) shows the form of the linear model identified where $A_{\text{BL}} = \{$prod_2_2016_01, prod_2_2018_01, prod_2_2020_01, prod_2_2022_01, prod_2_2024_01, prod_2_2026_01, prod_2_2028_01, prod_2_2032_01, prod_10_2016_01, prod_10_2018_01, prod_10_2020_01, prod_10_2026_01$\}$ denotes the set of active decision parameters. It is noted that Equation (4.4.1) contains no interaction terms since investigations show negligible improvement in the adjusted R^2 value for the linear model and hence they explain very little additional variation in the expected NPV. Let σ_{lm} be the estimate for the

residual standard error of this linear model. All decision parameters are transformed onto the range $[-1, 1]$. The unknown regression coefficients are assumed to have prior expectation $\boldsymbol{\mu}_\beta = \mathbf{0}$ and an infinite prior uncertainty, with emulator updates exploiting the limiting results as $\text{Var}[\boldsymbol{\beta}] \to \infty$ in Equations (2.4.44) and (2.4.45) derived in Section 2.4.5.

$$\mathbb{E}[\text{NPV}](\mathbf{d}_{A_{\text{BL}}}) = \beta_0 + \sum_{d_i \in A_{\text{BL}}} \{\beta_{i,1} d_i + \beta_{i,2} d_i^2\} + \varepsilon \tag{4.4.1}$$

For the residual process it is assumed that $\mathbb{E}[u(\mathbf{d}_{A_{\text{BL}}})] = 0$ and $\mathbb{E}[w(\mathbf{d})] = 0$ with a squared exponential covariance structure (Equation (2.4.3)) using a single common correlation length hyperparameter. Following the substitution approach for the hyperparameters: $\sigma_u^2 = (1-\rho)\sigma_{lm}^2$ and $\sigma_w^2 = \rho\sigma_{lm}^2$ where $\rho = 0.05$; whilst the correlation length parameter is set to half of the parameter range, hence $\theta = 1$. These choices are validated via emulator diagnostics discussed below.

Bayes linear adjustment for the emulator expectation and variance is performed following Equations (2.4.44) and (2.4.45) respectively. In comparison with the direct use of the simulator we observe a substantial reduction in computation time with over 2000 emulations for new decision parameter settings per second using a single core compared to approximately 30 minutes per OLYMPUS model, or 25 hours using the entire ensemble. The combination of EGES and Bayes linear emulation equates to an efficiency gain of the order of 10^8, thus highlighting one of the key benefits of emulation. Moreover the process is easily parallelisable to achieve even greater increases in speed.

Leave-one-out diagnostics suggest that the emulator fits well across the decision space, as shown in Figure 4.11 of the emulator adjusted expectation with credible interval of width 3 adjusted standard deviations versus the expected NPV where 691 of the 702 (98.4%) credible intervals contain the simulated expected NPV, as highlighted by the red dashed line representing equality. For a Bayes linear emulator this is an approximate 95% credible interval and so the Frequentist coverage is satisfied, thus validating the emulator. Moreover, if the additional assumption of a

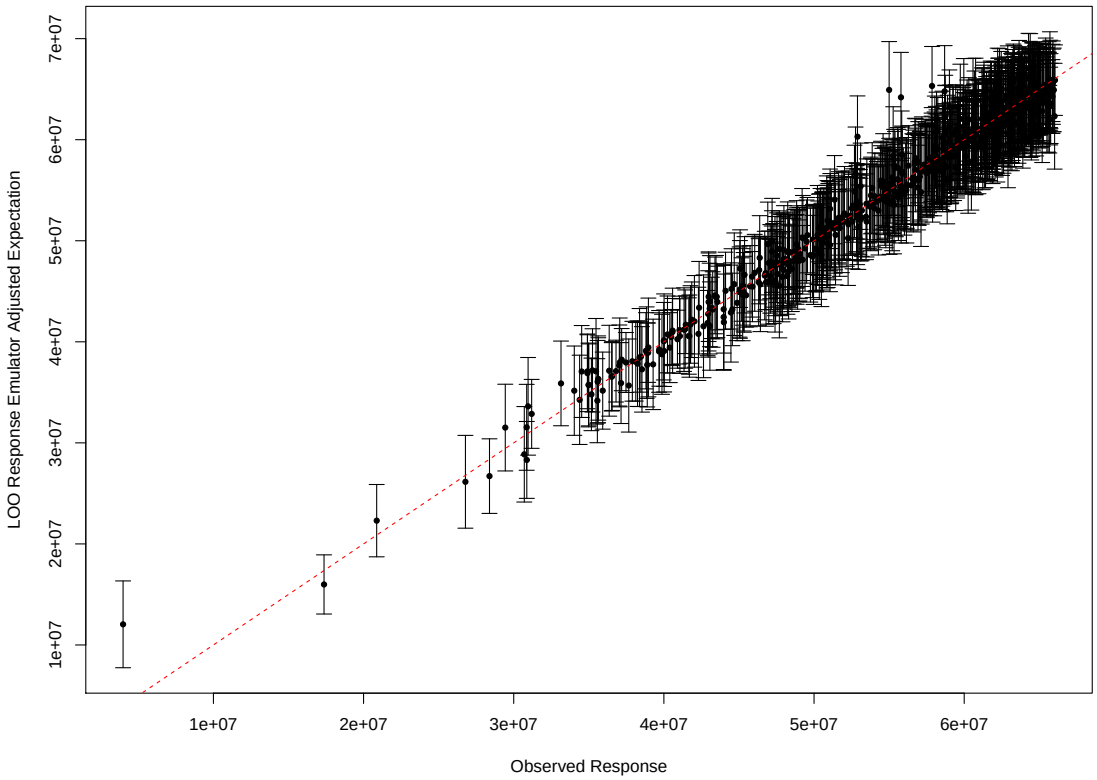


Figure 4.11: Bayes linear emulator for expected NPV leave-one-out diagnostic plot of the emulator adjusted expectations with 3 adjusted standard deviation credible interval error bars versus the simulated expected NPV. The red dashed line denotes equality of the emulator prediction and the observed value.

Gaussian process emulator was imposed, then the 95% credible would be of width 1.96 adjusted standard deviations. It is found that 679 of the simulated expected NPVs are contained within the respective credible intervals, corresponding to a Frequentist coverage of 96.7%, even under these stronger assumptions. It is noted that the few cases where these diagnostics are not satisfied tend to yield over-prediction which is safe when we are ruling out parts of the parameter space that yield low expected NPV. This is also less of a concern with a view to decision support where refinement at later waves will likely correct this. Other leave-one-out diagnostics for the emulator residuals such as those discussed in Section 2.4.6 highlight no major issues or misspecification.

4.5 Hierarchical Emulation Exploiting Known Simulator Behaviour

In this section we present an application of our developed methodology for hierarchical emulation exploiting known simulator behaviour detailed in Section 3.3 to the TNO OLYMPUS Well Control Optimisation Challenge. As for the Bayes linear emulation, this analysis makes use of the EGES linear model to predict the expected NPV, using only 3 geological realisations from Section 4.2 and the wave 1 design constructed in Section 4.3.2. In Section 4.5.1 we decompose the NPV formula by model into its constituent parts and perform structured emulation. These are combined to obtain the approximate NPV for an individual model in Section 4.5.2 before linking to the exact NPV in Section 4.5.3. The utility function is then emulated in Section 4.5.4. A comparison is made with the results of standard Bayes linear emulation in Section 4.5.5.

4.5.1 Structured Emulators Exploiting Known Simulator Behaviour for NPV Constituents

In the TNO OLYMPUS Well Control Optimisation Challenge the expected NPV utility function can be decomposed into three parts: FOPT, FWPT and FWIT for each control interval and may be divided further into the corresponding well quantities. As discussed in Section 3.3.1, for each OLYMPUS model, the WOPT and WWIT within a control interval are observed to follow structured behaviour where the quantity is equal to the corresponding target rate decision parameter multiplied by the length of the time interval up to an unknown change point beyond which there is a plateau in the behaviour due to the BHP constraint. This is illustrated for the OLYMPUS 25 WOPT for producer well 2 over the first two years of the field lifetime (this is the first control interval) in Figure 4.12. The red points correspond to those which exceed the change point upper bound for the target production rate for producer well 2 during this interval, $c^u_{P2,2016}$, calculated using Equation (3.3.1)

with $\delta_{i,u} = 10$. It is also noted that such structured behaviour is not observed for WWPT within a control interval since there is no corresponding target rate; its behaviour is a consequence of attempting to achieve a given target production rate subject to the BHP constraint with water present within the oil field.

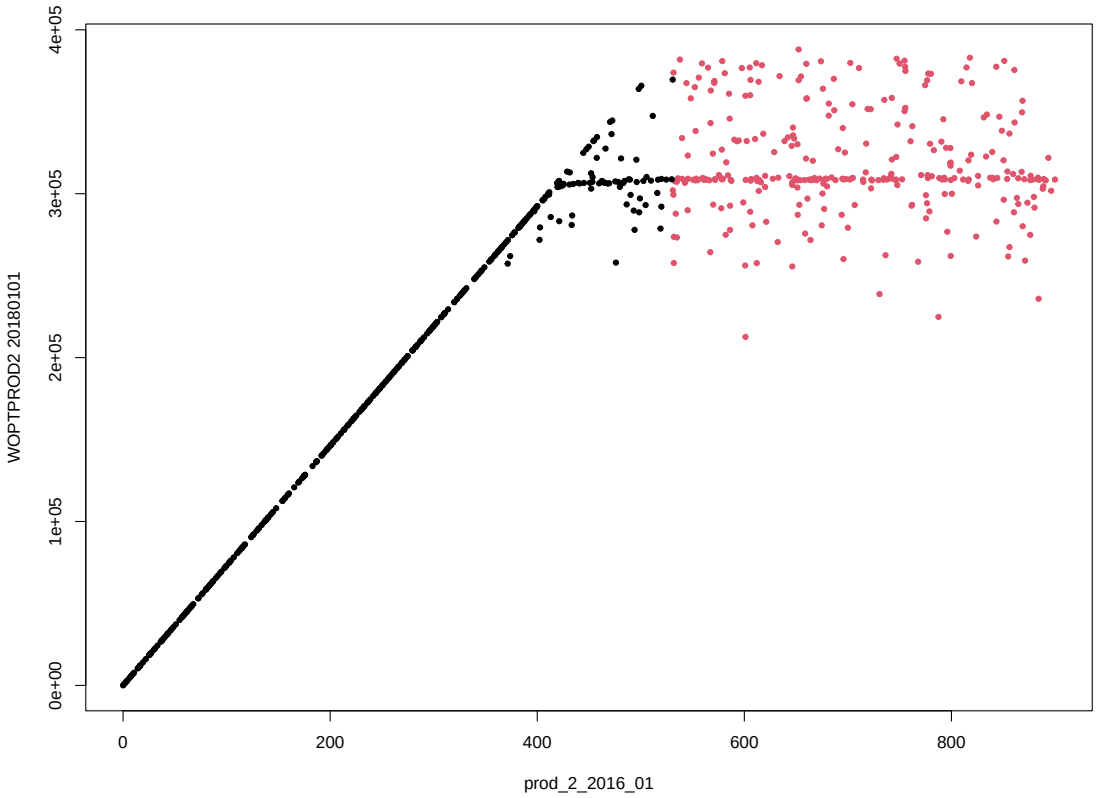


Figure 4.12: OLYMPUS 25 WOPT for producer well 2 during the first two years (ending 01/01/2018) versus the corresponding decision parameter, the target production rate, prod_2_2016_01. For small values of prod_2_2016_01 the target is achieved resulting in a perfectly defined linear behaviour up to a change point due to other controls and BHP constraints beyond which the WOPT plateaus as a maximum threshold on the production rate is achieved. The red points correspond to where prod_2_2016_01 $= d_{P2,2016} > c^u_{P2,2016}$.

The structured emulation technique described in Section 3.3.2 is applied separately for each of the OLYMPUS models to the WOPT and WWIT within each control interval for wells in the CWG. These simulator outputs will be considered as $f_i(\mathbf{d})$ in the construction of the respective emulator. Firstly, for each of these NPV

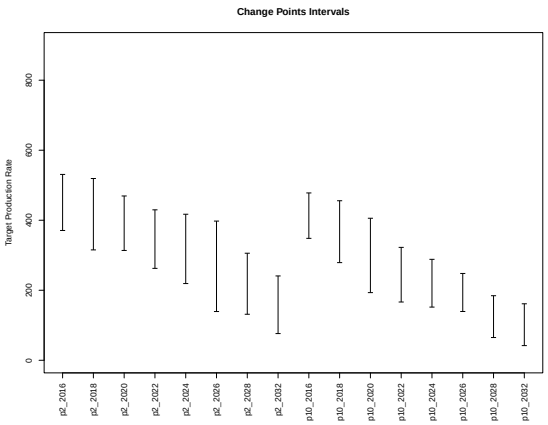
constituents, a conservative estimate for the change points is obtained via the change point upper bounds calculated from the wave 1 simulations using Equation (3.3.1), each time with $\delta_{i,u} = 10$. This ensures numerical stability and that an upper bound is obtained with all points exceeding this definitely in the plateau region. See Figure 4.12 for an example. Next the extrapolation cut-offs are estimated as the change point lower bounds from the wave 1 simulations using Equation (3.3.2) with $\delta_{i,l} = 10$ to account for numerical precision within the simulations. The change point upper bounds and extrapolation cut-offs are illustrated for each of the wave 1 sub-selected OLYMPUS models in Figure 4.13 highlighting the region in which the "true" change point is believed to be situated.

For each NPV constituent, a preliminary Bayes linear emulator is fitted of the form shown in Equation (2.4.1) with deterministic functions, $m_i(\mathbf{d}_{A_i})$, consisting of a constant as well as all linear and quadratic terms in the active decision parameters as follows:

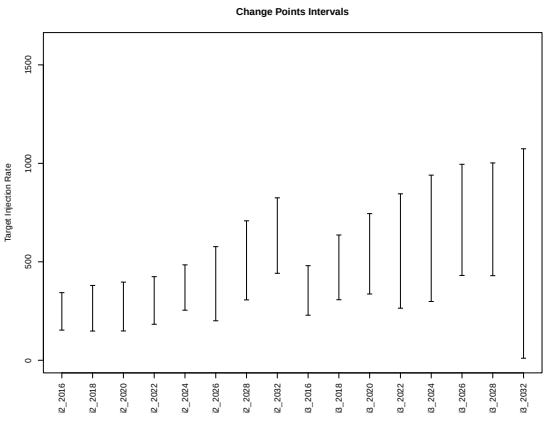
$$m_i(\mathbf{d}_{A_i}) = \mathbf{g}_i(\mathbf{d}_{A_i})^{\mathrm{T}}\boldsymbol{\beta}_i = \beta_{i,0} + \sum_{d_j \in A_i} \{\beta_{i,j,1} d_j + \beta_{i,j,2} d_j^2\} \tag{4.5.1}$$

It is assumed that the active decision parameters comprise all decisions which take place in the past of the output, for example, for WOPTPROD2 these are $A_i = \{$prod_2_2016_01, prod_10_2016_01, inj_2_2016_01, inj_3_2016_01$\}$. This is a logical choice since future decisions are physically unable to impact on an output up to the current time, however any past decisions may potentially have an effect.

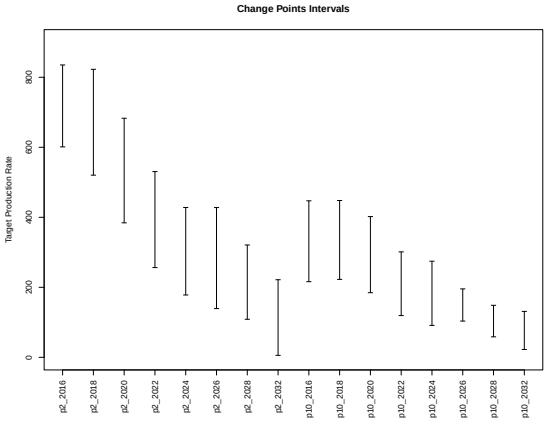
The remainder of each emulator's prior specification is obtained via the same means as for the expected NPV Bayes linear emulator in Section 4.4 with the distinction being that only those simulation points in $\mathcal{D}_i' = \{\mathbf{d} \mid \mathbf{d} \in \mathcal{D},\ d_{jk,t_i} > c^u_{jk,t_i}\}$ with output $\mathbf{F}_i' = \{f_i(\mathbf{d}) \mid \mathbf{d} \in \mathcal{D}_i'\}$ are used. That is a linear model of the form shown in Equation (4.5.1) is fitted using these points with all decision parameters transformed onto $[-1, 1]$. An estimate of the residual standard error is denoted by $\sigma_{i,lm}$. It is assumed that the unknown regression coefficients have prior expectation $\boldsymbol{\mu}_\beta = \mathbf{0}$ and an infinite prior uncertainty, again employing the limiting results in



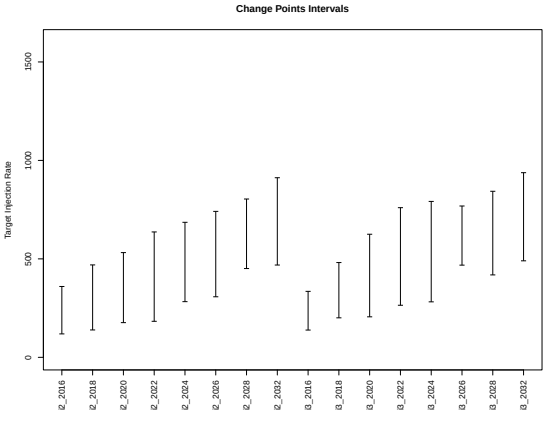
(a) OLYMPUS 25 WOPT



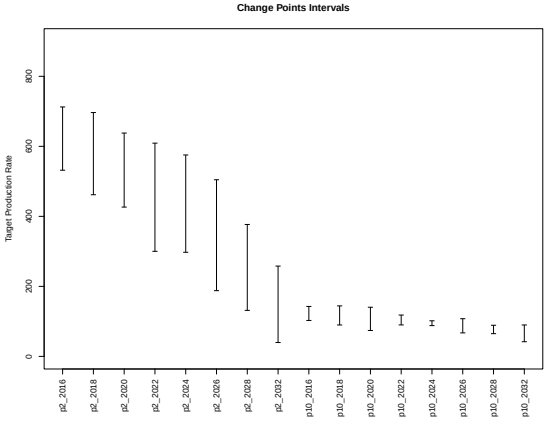
(b) OLYMPUS 25 WWIT



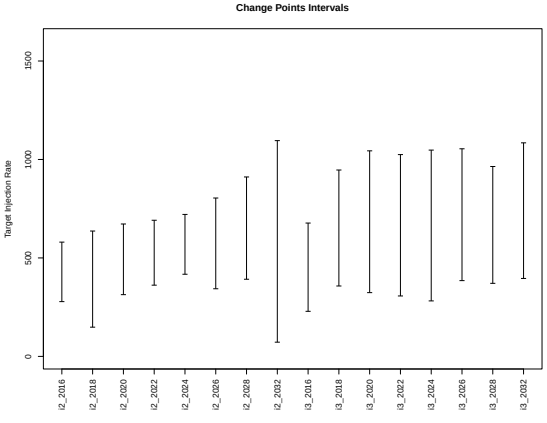
(c) OLYMPUS 33 WOPT



(d) OLYMPUS 33 WWIT



(e) OLYMPUS 45 WOPT



(f) OLYMPUS 45 WWIT

Figure 4.13: OLYMPUS wave 1 change point upper bound and extrapolation cut-off intervals for WOPT and WWIT within each control interval with respect to their corresponding decision parameter for each of the three sub-sampled OLYMPUS models.

Equations (2.4.44) and (2.4.45) derived in Section 2.4.5. For the residual process it is assumed that $\mathbb{E}[u_i(\mathbf{d}_{A_i})] = 0$ and $\mathbb{E}[w_i(\mathbf{d})] = 0$ with a squared exponential covariance structure (Equation (2.4.3)) using a single common correlation length hyperparameter. Following the substitution approach for the hyperparameters: $\sigma^2_{u_i} = (1-\rho)\sigma^2_{i,lm}$ and $\sigma^2_{w_i} = \rho\sigma^2_{i,lm}$ where $\rho = 0.05$; whilst the correlation length parameter is set to half of the parameter range, hence $\theta = 1$.

Given this preliminary emulator and a theoretical upper bound based on the target rate for the time interval, the emulation output type is determined with points being classified as in the slope, plateau, or close to the change point following the structured emulation with upper truncation method presented in Section 3.3.2. Note that it is not necessary to use the two-sided truncation version as for each OLYMPUS model and NPV constituent there are a sufficient number of points in $\mathcal{D}'_i$ which combined with the structured regression term within the preliminary Bayes linear emulator leads to reasonable predictive performance over the extrapolation region with low-to-moderate uncertainty.

Leave-one-out diagnostics are performed yielding plots such as those shown in Figures 4.14 and 4.15 for OLYMPUS 25 NPV constituents where Figures 4.14a, 4.14c, 4.15a and 4.15c show the structured emulator adjusted expectation with credible intervals of width 3 adjusted standard deviations versus the corresponding simulated output. In each case it is observed that for smaller values of each NPV constituent that the emulator is exceptionally accurate corresponding to where the target rate is adhered to for the entire control interval. For larger values of the simulated output believed to be on plateau, the credible interval is of a greater width, whilst the implementation of a truncated GP emulator for intermediate values in the uncertain region around the location of the change point demonstrates a reduction in the uncertainty in these locations. The type of emulator employed within the classification step is best observed in Figures 4.14b, 4.14d, 4.15b and 4.15d which show the credible intervals versus each NPV constituents' corresponding decision parameter. Another observation is that structured emulators for a fixed output type

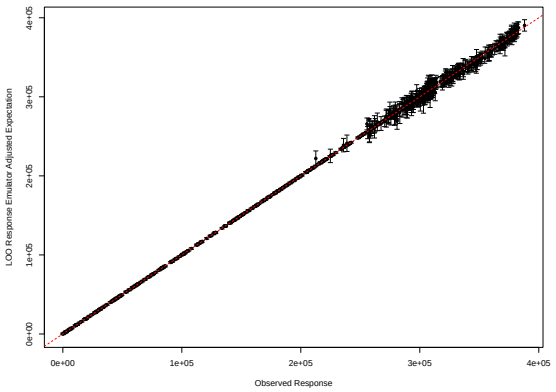
are generally more accurate where the truncated GP or Bayes linear emulator is applied for earlier time intervals. This is to be expected since there is less uncertainty of the preceding oil field behaviour, as well as fewer decision parameters which may affect the output. For all NPV constituents, the majority of the credible intervals contain the simulated value and no issues are detected in other leave-one-out diagnostic analyses. It is therefore demonstrated that it is possible to achieve very accurate emulators for the WOPT and WWIT NPV constituents based on a relatively small number of simulations whilst also capturing the change in behaviour.

As commented above, WWPT within a control interval does not follow the same structured behaviour as WOPT and WWIT, hence it is emulated separately using a Bayes linear emulator. This follows the same construction as for the preliminary Bayes linear emulator within the structured emulation approach implemented above with active decision parameters consisting of all those which occur in the past of the output. The prior specification is determined as above with deterministic functions of the same form as in Equation (4.5.1), but with all simulations in $\mathcal{D}$ used to fit the respective emulators since there is no obvious and easily exploitable structure.

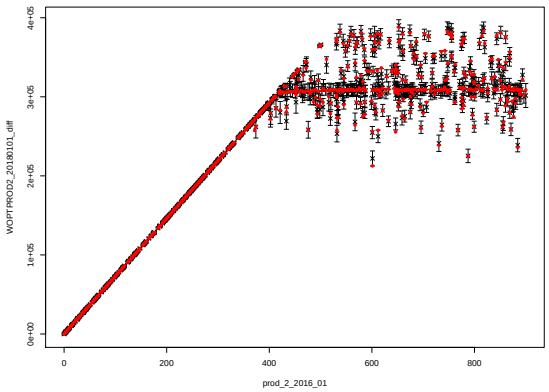
Within this application, for each OLYMPUS model, this requires the fitting of 48 separate emulators: 32 of the structured type; and 16 Bayes linear emulators. This leads to a total of 144 emulators over the three sub-selected OLYMPUS models. The next stage of this analysis is to combine these constituent structured and Bayes linear emulators to first emulate the NPV for a particular model, before subsequently emulating the ensemble mean NPV and linking this to the expected NPV.

4.5.2 Emulation of the Approximate NPV

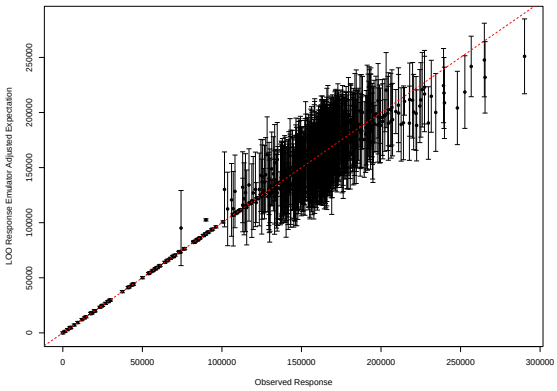
The methodology presented in Section 3.3.3 is applied to combine the emulators for the NPV constituents within the control intervals to obtain an emulator for the average discounting approximate NPV. Decisions are implemented over time periods constructed by amalgamating consecutive discounting intervals, as described by Equations (3.3.5) and (3.3.6). In this application, the 80 3-month discounting



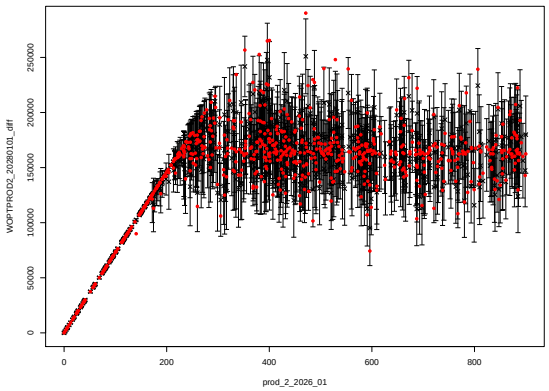
(a) WOPTPROD2_2018_01 emulator CI versus simulated output.



(b) WOPTPROD2_2018_01 emulator CI versus prod_2_2016_01.

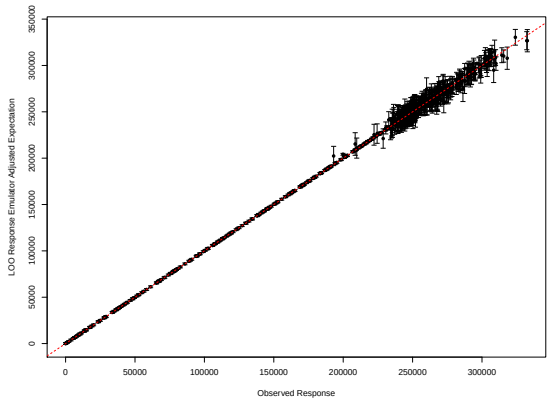


(c) WOPTPROD2_2028_01 emulator CI versus simulated output.

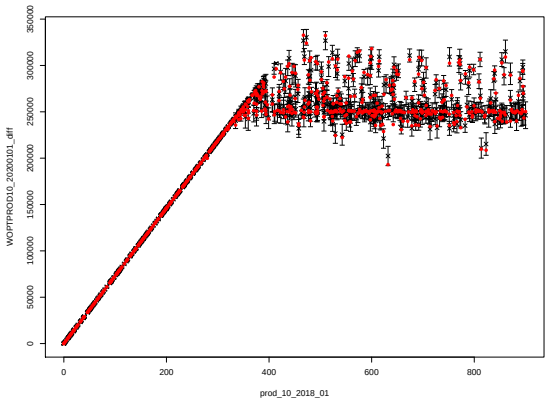


(d) WOPTPROD2_2028_01 emulator CI versus prod_2_2026_01.

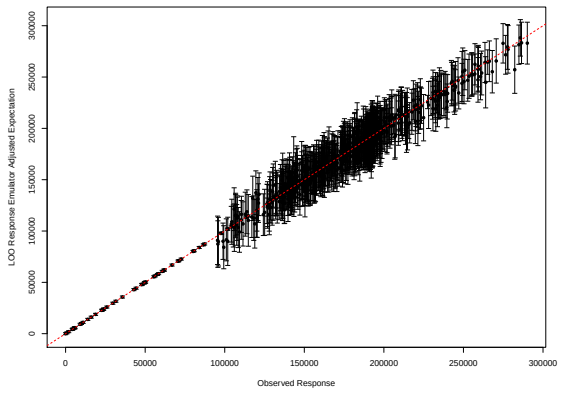
Figure 4.14: Leave-one-out diagnostic plots for the structured emulation of OLYMPUS 25 WOPTPROD2_2018_01 (top) and WOPTPROD2_2028_01 (bottom). The left column shows the adjusted expectation with credible intervals (CI) of width 3 adjusted standard deviations error bars versus the simulated value where the red dashed line denotes equality of the structured emulator and simulator. The right column shows the adjusted expectation with credible intervals of width 3 adjusted standard deviations error bars versus the output's corresponding decision parameter, the target production rate, with red points denoting the simulated values.



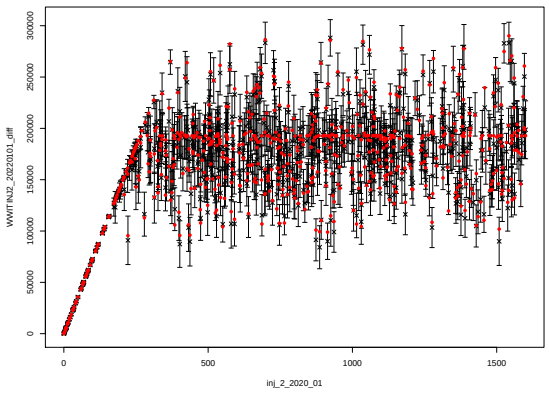
(a) WOPTPROD10_2020_01 emulator CI versus simulated output.



(b) WOPTPROD10_2020_01 emulator CI versus prod_10_2018_01.



(c) WWITINJ2_2022_01 emulator CI versus simulated output.

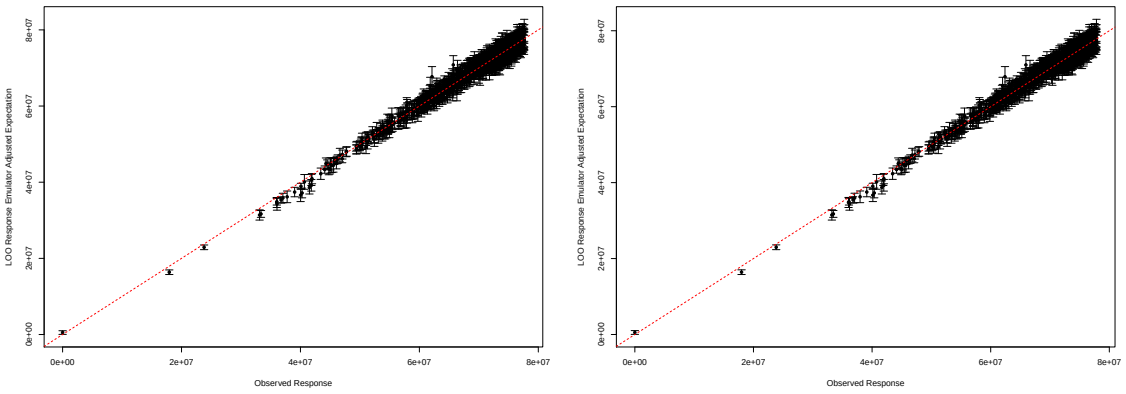


(d) WWITINJ2_2022_01 emulator CI versus inj_2_2020_01.

Figure 4.15: Leave-one-out diagnostic plots for the structured emulation of OLYMPUS 25 WOPTPROD10_2020_01 (top) and WWITINJ2_2022_01 (bottom). The left column shows the adjusted expectation with credible intervals (CI) of width 3 adjusted standard deviations error bars versus the simulated value where the red dashed line denotes equality of the structured emulator and simulator. The right column shows the adjusted expectation with credible intervals of width 3 adjusted standard deviations error bars versus the output's corresponding decision parameter, the target production or injection rate, with red points denoting the simulated values.

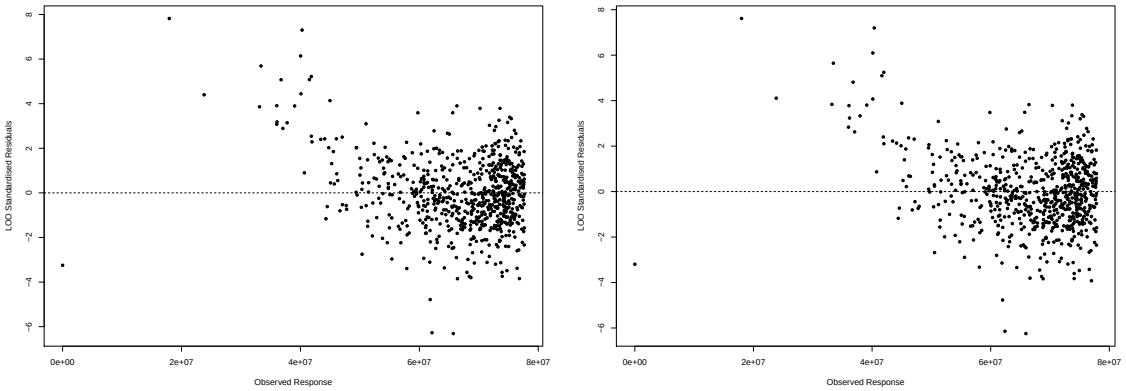
intervals are merged to form the 8 control intervals per well where the splits naturally occur at the start times of each successive decision. The adjusted expectation and variance of the emulator for the approximate NPV is then obtained employing Equations (3.3.7) and (3.3.8) with the NPV cost parameters fixed at the values specified by TNO in Table 4.3 on page 195, treating the NPV constituents as independent. For WOPT and WWIT outputs where the structured emulator is collapsed onto the slope with fixed maximum absolute error, the adjusted standard deviation is approximated by treating this uncertainty interval as equivalent to a 3-sigma credible interval. Intuition and reservoir engineering insight suggest there exists temporal (and potentially spatial) correlation between the NPV constituents. However, a consequence of the individual emulator constructions and the differing subsets of simulation design points on which they are fitted results in it being challenging to meaningfully assess the correlation structure required for a separable multivariate emulator. Realisation of a multivariate framework via Equations (3.3.7) and (3.3.9) leads to various approximate NPV emulator variances depending on the chosen correlation structure and hence is non-robust whilst also demonstrating little or no improvement in uncertainty versus the implemented approach, therefore justifying the independence assumption.

Leave-one-out diagnostics for the approximate NPV emulator for OLYMPUS 25 are shown in Figures 4.16a and 4.16c. There exists a strong linear trend between the adjusted expectation and simulated approximate NPV with the majority of points being situated close to the red dashed line depicting equality with the simulated output. Moreover, the uncertainty generally increases with the approximate NPV. This can be explained by the nature of the behaviour of the WOPT and WWIT constituents and the imposition of an extrapolation cut-off within their structured emulators leading to generally larger uncertainty when higher NPVs are simulated since these generally correspond to higher oil production rates and moderate to low water injection rates. A further observation is that it is reasonably common for multiple (and even the majority) of the NPV constituents to fall on their slope regions



(a) Average discounting approximate NPV credible intervals versus simulated values.

(b) Exact NPV credible intervals versus simulated values.



(c) Average discounting approximate NPV standardised residuals versus simulated values.

(d) Exact NPV standardised residuals versus simulated values.

Figure 4.16: Leave-one-out diagnostics for emulators of the OLYMPUS 25 average discounting approximate NPV (left column) and the exact NPV (right column) via a simple linear model on the emulated approximate NPV. The top row shows the emulator adjusted expectation with 3 adjusted standard deviation credible intervals versus simulated values where the red dashed line denotes when the emulator and simulator coincide. The bottom row shows the emulator standardised residuals versus the simulated values.

for a specified decision parameter vector leading to individually smaller structured emulator uncertainties. These linearly combine to produce a small uncertainty for the approximate NPV. Figure 4.16c shows the emulator standardised residuals versus the simulated approximate NPV. There is evidence of a linearly decreasing pattern, particularly for smaller values of the simulated approximate NPV where there exist

a number of larger, mainly positive residuals corresponding to underprediction. For moderate to high simulated approximate NPVs, the majority of the standardised residuals have an acceptable magnitude less than three. Comparing Figures 4.16a and 4.16c, it is evident that these occur for smaller simulated approximate NPVs and hence with a view to decision support, it is expected that these will be discarded at wave 1 and so are possibly not a concern. There are also a few cases with large negative residuals due to overprediction. The iterative nature of decision support enables further investigation of these parameter vectors at later waves. In addition, the overall aim is to construct an accurate emulator for the final utility function, where the approximate NPV for each OLYMPUS model are intermediate quantities. Assessment of their leave-one-out diagnostics therefore provides valuable insight into the potential sources of any failures in the leave-one-out diagnostics for $U(\mathbf{d})$.

4.5.3 Linking the Exact and Approximate NPV

The next step within the hierarchical emulator construction is to link the exact and approximate NPV in order to account for the discrepancy induced by coalescing the discounting intervals. This follows the simple linear regression framework presented in Section 3.3.4 with the adjusted expectation and variance computed using Equations (3.3.12) and (3.3.13) where for each OLYMPUS model, $\hat{\beta}_{0,\widetilde{\text{NPV}}_j}, \hat{\beta}_{1,\widetilde{\text{NPV}}_j}$ and $\sigma_{\widetilde{\text{NPV}}_j}$ are estimated using the wave 1 simulation data.

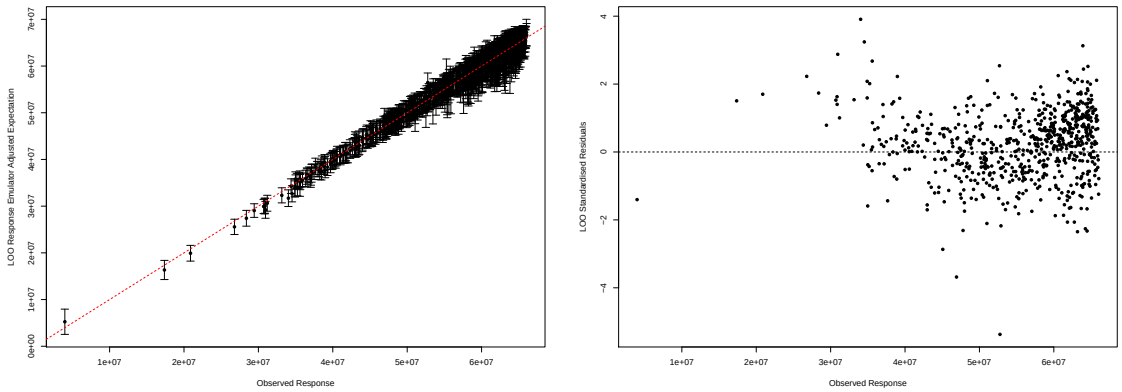
Leave-one-out diagnostics are performed for the OLYMPUS 25 NPV with plots shown in Figures 4.16b and 4.16d. The results are extremely similar to those for the average discounting approximate NPV in Figures 4.16a and 4.16c respectively with commentary mirroring that in Section 4.5.2. Figure 4.16d exhibits a linearly decreasing trend for smaller values of the simulated approximate NPV and occurs due to underprediction highlighted by a number of larger, mainly positive residuals. Within decision support this is potentially not a concern as it is expected that these will be discarded at wave 1. This highlights how the simple linear regression approach to linking the approximate and exact NPV introduces very little additional

uncertainty compared to the initial combination of the NPV constituent emulators.

4.5.4 Emulation of the Ensemble Mean NPV

The process of building structured emulators for each of the NPV constituents, their combination via the NPV formula to obtain the average discounting approximate NPV, and subsequent linking to the exact NPV is repeated for each of the three sub-selected OLYMPUS models. For both the TNO OLYMPUS Well Control Optimisation Challenge and our setup for decision support, the ensemble mean NPV is the objective and utility function respectively. This is emulated using the EGES linear model devised in Section 4.2 to combine the emulators for the OLYMPUS 25, 33 & 45 NPVs, as described in Section 3.3.5. The adjusted expectation and variance are computed using the formulae in Equations (3.3.16) and (3.3.17) respectively which account for the covariance structure when estimating the regression coefficients.

It is not possible to perform leave-one-out diagnostics because simulations have only been performed for a subset of three of the OLYMPUS models, thus the true ensemble mean NPV cannot be computed. Note that this additional uncertainty pertaining to the EGES is already accounted for within the hierarchical emulator construction. Instead we compare the hierarchical emulator output with the predicted ensemble mean NPV, as shown in Figure 4.17. It is evident in Figure 4.17a that the hierarchical emulator produces accurate predictions for the ensemble mean NPV. Moreover, the increase in the uncertainty compared to individually emulating the OLYMPUS 25 NPV in Figure 4.16b is modest; thus demonstrating that the EGES linear model contributes relatively little additional uncertainty versus the structured emulation of the NPV constituents for each model. This will be explored further in Section 4.5.5 where the magnitude of the emulator uncertainty is compared to the other sources of uncertainty. Figure 4.17b shows no distinguishable pattern in the pseudo standardised residuals, whilst the majority are of magnitude less than three.



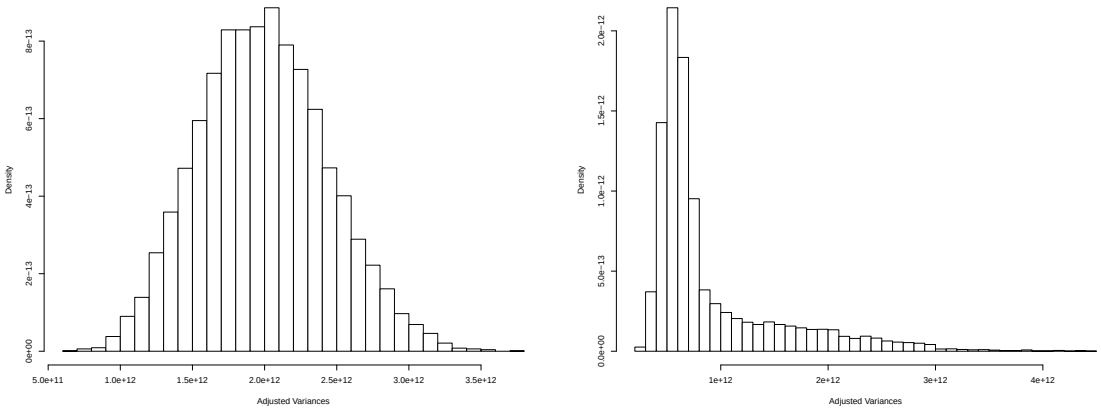
(a) Hierarchical emulator adjusted expectation with 3 adjusted standard deviation credible intervals versus the simulated ensemble mean NPV predicted via the EGES linear model where the red dashed line denotes equality of the emulator and simulator.

(b) Hierarchical emulator standardised residuals versus the simulated ensemble mean NPV.

Figure 4.17: OLYMPUS wave 1 hierarchical emulation diagnostic plots for the predicted ensemble mean NPV via the EGES linear regression model combining the emulation output for the exact NPV of the three sub-selected OLYMPUS models.

4.5.5 Comparison of Bayes Linear and Hierarchical Emulation for the Ensemble Mean NPV

Two approaches have been implemented for emulating the ensemble mean NPV: a direct Bayes linear emulator; and a hierarchical emulator exploiting known simulator behaviour. Firstly, it is observed that the hierarchical emulator achieves a discernible reduction in the uncertainty versus the Bayes linear emulator, as characterised by the emulator adjusted variances. This is evident in a comparison of the leave-one-out diagnostics plots in Figures 4.11 and 4.17a which show a reduction in the credible interval widths, and is confirmed in Figure 4.18 which highlights an average reduction in the adjusted variance of more than a half. Note that there exist a small number of cases where there is a moderate increase in the uncertainty, although this is outweighed by the gains achieved across the majority of sampled locations within the decision parameter space. For context, the seemingly large variances are inline



(a) Bayes linear emulator. (b) Hierarchical emulator.

Figure 4.18: Comparison of the wave 1 Bayes linear and hierarchical emulators for the OLYMPUS ensemble mean NPV via histograms of their adjusted variances. For context, the seemingly large variances are inline with the scale of the ensemble mean NPV which is typically of order of 5.0×10^7 \$ to 6×10^7 \$.

with the scale of the ensemble mean NPV which is typically of order of 5.0×10^7 \$ to 6.0×10^7 \$. The implications for decision support are discussed in Sections 4.7.1 and 4.7.2.

Bayes linear emulation is known to be a very fast and efficient means of constructing emulators, as highlighted in 4.4 where a performance of emulating at over 2000 new decision parameter vectors per second using a single core was achieved. In comparison, the performance for the entire process of hierarchical emulation yielded a much more modest performance of emulating at approximately 4 new decision parameter vectors per second using a single core. However, in comparison with direct simulation from the OLYMPUS ensemble where a single evaluation takes approximately 25 hours, this is still a considerable efficiency improvement of the order of 10^4 which is sufficient for comprehensively exploring the decision parameter space. Moreover, it is believed that this additional emulator computational expense is justified by the reduction in the emulator uncertainty since it will be shown in Section 4.6 that for wave 1 decision support the emulator is the dominant source of uncertainty. Reducing the emulator uncertainty is therefore imperative in order to efficiently reduce the decision support non-implausible region and hence avoid

additional waves of extremely expensive simulations at locations that will be eliminated after the construction of a more accurate emulator. As this can be achieved by considering the known structure of simulator output, the additional computational cost is therefore offset versus extra simulations. Note that both emulation processes are easily parallelisable, thus permitting further efficiency gains.

4.6 Uncertainty Analysis for Well Control Optimisation

The TNO OLYMPUS Field Development Optimisation Challenge has been formulated with the main aim of addressing the problem of geological uncertainty represented via the ensemble of 50 OLYMPUS models with no other sources of uncertainty considered. As emphasised in Section 3.1.2, we feel that this does not adequately represent the real problem of devising optimal well control strategies and so extend the uncertainty quantification to illustrate the importance of a comprehensive uncertainty analysis. For this we work with the utility function $U(\mathbf{d}) = \mathbb{E}[\text{NPV}](\mathbf{d})$ and implement the techniques presented and developed in Sections 2.7 and 3.5. Assuming that the OLYMPUS oil reservoir exists, the computer model is linked to reality in Section 4.6.1. The geological uncertainty is given a more thorough treatment in Section 4.6.2, whilst the additional uncertainty arising from efficient geological ensemble subsampling is considered in Section 4.6.3. An assessment of the NPV cost parameter uncertainty is presented in Section 4.6.4 demonstrating the importance of accounting for the way the uncertainty affects the utility (or objective) function. Comments are also made on structural model discrepancy and decision implementation error in Sections 4.6.5 and 4.6.6 respectively. A summary of the uncertainty analysis is presented in Section 4.6.7. The magnitude of the assessed uncertainties are compared to the utility function emulator uncertainty in Section 4.6.8.

4.6.1 Linking the OLYMPUS Model to the Real World System

For our application we use an uncorrelated additive error structure described in Equations (2.7.1) and (3.6.2) to link the OLYMPUS model with the real world oil reservoir that it is designed to represent. Note that the OLYMPUS reservoir is fictitious, thus limiting the ability to provide a meaningful uncertainty specification, particularly for structural model discrepancy and decision implementation error, due to the lack of a real world system with which to link the OLYMPUS model. For the purposes of this analysis, we assume that the OLYMPUS reservoir exists in order to demonstrate how the decision support process would be performed for a real world system. Since the OLYMPUS ensemble is too computationally expensive to evaluate for a large number of choices of decision parameter settings, we introduce an emulator for the utility function extending the additive error link as shown in Equation (2.7.2). Uncertainties included within this analysis include: geological uncertainty; EGES uncertainty; and NPV cost parameter uncertainty, whilst comments are also made regarding structural model discrepancy and decision implementation error. It is believed that each of these uncertainties are mutually uncorrelated, and have a potentially additive effect on the utility function, thus justifying the choice of link between the model and reality whilst also permitting a dependence on the location in the decision parameter space. Since this uncertainty quantification will be used to perform decision support via the implausibility measure introduced in Equation (3.6.3), we assess the variance of each source as a characterisation of the induced uncertainty.

4.6.2 Geological Uncertainty

Geological uncertainty within the TNO OLYMPUS Field Development Optimisation Challenge is an example of ensemble uncertainty discussed in Section 3.5.3 and concerns the lack of knowledge regarding the geology of the OLYMPUS model. Com-

pared to the TNO OLYMPUS Challenge, the treatment of geological uncertainty, ε_{G}, is broadened beyond the 50 models in the ensemble to consider the variability in the utility function if an alternative sample of geologies were produced from the underlying stochastic geology model. Assessment of ensemble uncertainty, and specifically geological uncertainty, requires simulations to be performed for the entire ensemble, thus we use the preliminary exploratory collection of simulations in this analysis. From these simulations we obtain the sample variance, s^2, of the ensemble mean NPV computed over the exploratory decisions. It is assumed that $\mathbb{E}[\varepsilon_{\mathrm{G}}] = 0$ and $\mathrm{Var}[\varepsilon_{\mathrm{G}}]$ is estimated by $\frac{s^2}{N} = \frac{s^2}{50}$. In our analysis at wave 1, $\mathrm{Var}[\varepsilon_{\mathrm{G}}] = 6.381 \times 10^{11}$. This may be recalculated at later waves as the size of the non-implausible decision space decreases and thus restricts the decision parameters used in this assessment. A geological uncertainty term is incorporated to account for that the decision makers could in principle resolve this uncertainty by requesting a large number of geological realisations to be evaluated over the final decision support region before committing to a final strategy. An alternative approach not implemented in this thesis is to use bootstrap sampling of the ensemble to estimate the utility function variance induced by geological uncertainty.

4.6.3 Efficient Geological Ensemble Subsampling Uncertainty

A subset of 3 of the 50 models are used for the wave 1 simulations chosen via the EGES techniques implemented in Section 4.2. This induces some additional uncertainty in the utility function, $\varepsilon_{\mathrm{EGES}}$. Note that this differs from the geological uncertainty which concerns the step from 50 to a potentially infinite number of geological configurations, whilst EGES uncertainty accounts for using a subset of the ensemble. Moreover, the exact treatment of the EGES uncertainty is distinguished by the form of the utility function emulator.

Bayes linear emulation of the utility function is applied in Section 4.4 and requires an explicit treatment of the EGES uncertainty. The implemented EGES technique

uses a linear model from which it is assumed that $\mathbb{E}[\varepsilon_{\text{EGES}}] = 0$ whilst $\text{Var}[\varepsilon_{\text{EGES}}]$ is estimated by the square of the linear model residual standard error which for wave 1 is 1.149×10^{11}. This represents 18% of the magnitude of the variance due to geological uncertainty and thus justifies the ensemble subsampling due to the greatly reduced computational costs for only a moderate increase in uncertainty compared to other sources.

The hierarchical emulation approach implemented in Section 4.5 directly incorporates the EGES uncertainty within its structure during the step between the NPV of the three sub-selected models and the ensemble mean NPV described in Section 4.5.4. This formulation accounts for the uncertainty in the regression coefficient estimates whilst also adding a residual term which is assessed in the same manner as above.

4.6.4 NPV Cost Parameter Uncertainty

For the TNO OLYMPUS Well Control Optimisation Challenge the NPV cost parameters consist of the oil price, water production and injection costs, and discount factor which are required in order to compute the ensemble mean NPV. Since these are not necessary inputs to the OLYMPUS computer model, they represent a form of utility function cost parameter. In the TNO OLYMPUS Well Control Optimisation Challenge these are assumed to be fixed at the values stipulated in Table 4.3. However, a more realistic scenario is to permit the NPV cost parameters to vary with time introducing uncertainty, ε_{C}, for the utility function. Following the techniques presented in Section 3.5.2, we demonstrate an assessment of the NPV cost parameter uncertainty via a simulation experiment where we account for the transformation type of the utility function.

Simulation of NPV Cost Parameters

To illustrate the effect on decision support of time variable cost parameters a simulation based approach is implemented. A stationary Gaussian process is used to

Cost Parameter	Notation	Value
Oil Price	r_{op}	45 \$ per bbl
Water Produced Cost	r_{wp}	6 \$ per bbl
Water Injected Cost	r_{wi}	2 \$ per bbl
Discount Factor	d	0.08

Table 4.3: Expected NPV fixed cost parameters within the TNO OLYMPUS Well Control Optimisation Challenge.

independently simulate the oil price, water produced and injected costs with samples drawn at 3 month intervals. For consistency, the Gaussian process has a constant mean equal to the respective TNO fixed cost parameter value, each using an exponential covariance kernel with correlation length equal to 365 days and standard deviation parameter taken to be $\sqrt{15}$, 0.12 and 0.04 respectively. This is based on the belief that economic quantities do not vary smoothly with time. Such a model is commonly used within financial mathematics to model commodity prices. The described NPV cost parameter simulation model is not designed to be representative of our true beliefs for how the actual cost parameters will vary over the 20 year field lifetime, but is instead used for demonstrative purposes. Actual oil companies can substitute for their own more complex, stochastic economic model for the time evolution of the NPV cost parameters. A total of $n_c = 1000$ NPV cost parameter time series samples are generated with the first 10 oil price scenarios illustrated in Figure 4.19.

NPV Cost Parameter Uncertainty Before the Removal of Linear Transformation Effects

The ensemble mean NPV is extremely quick to evaluate given a collection of simulations from the OLYMPUS models. For each of the 702 wave 1 decision parameter vectors, the NPV is evaluated for the subset of 3 OLYMPUS models using the 1000 generated cost parameter scenarios with the EGES linear model used to predict the ensemble mean NPV. With a view to decision support, the naïve approach would be to then assess the variability in the expected NPV due to cost parameters. Examples

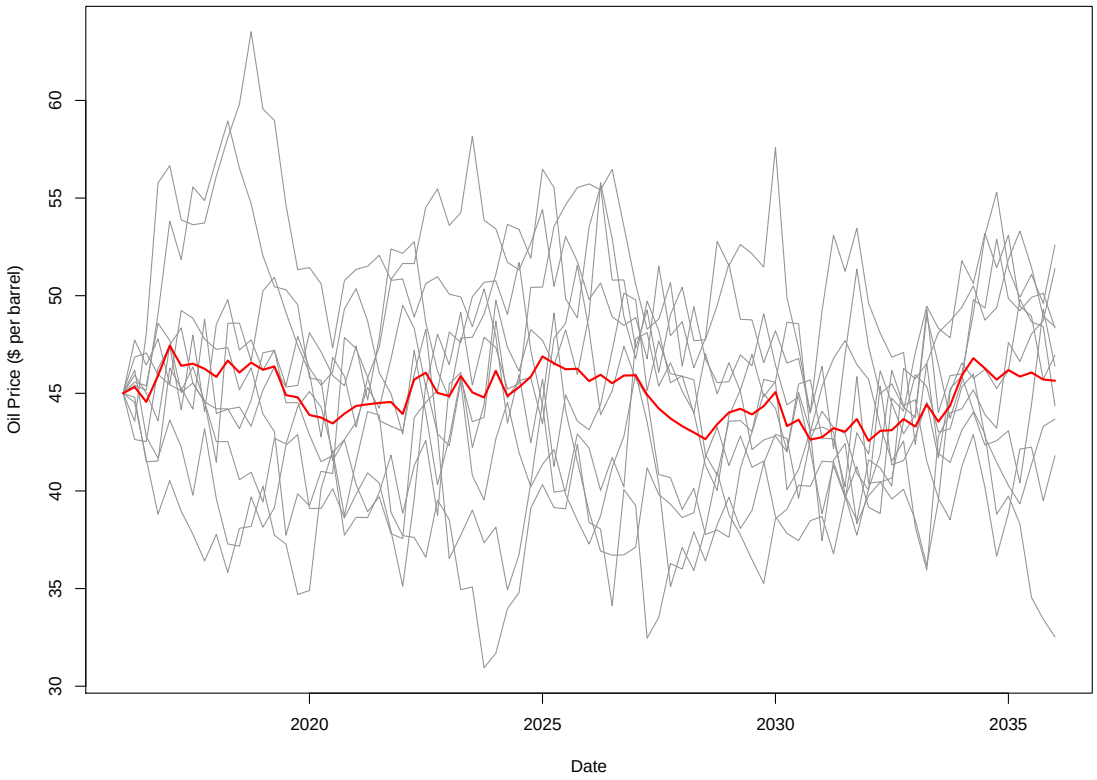
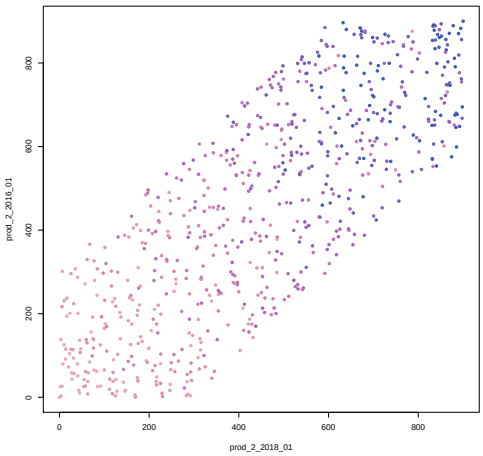
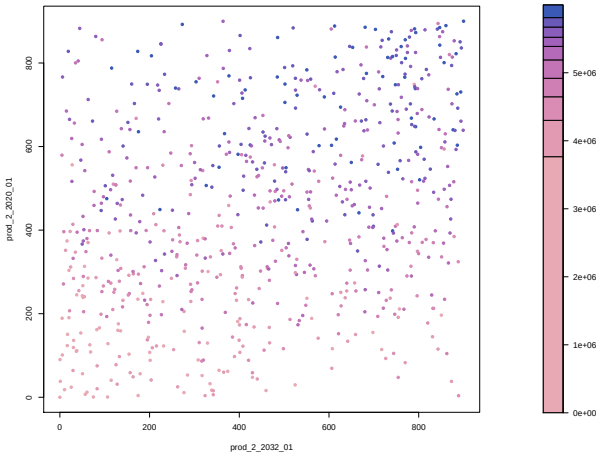


Figure 4.19: Trace plot of 10 simulated oil prices ($ per barrel) from a Gaussian process with constant mean of $45 per barrel and an exponential covariance kernel with correlation length of 365 days, and standard deviation $\sqrt{15}$. The red line denotes the sample mean over the 10 simulated scenarios.

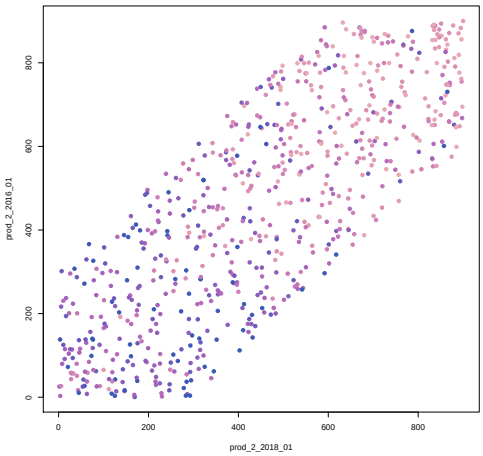
of plots illustrating the two-dimensional projection onto the decision space coloured by the standard deviation of ε_{C} using this approach are shown in Figures 4.20a and 4.20b. It is observed that before the removal of the linear effects, the largest uncertainty occurs for higher marginal parameter values and can be explained as a priori it is anticipated that selecting higher target rates will lead to larger ensemble mean NPVs for which the variance is also larger. This is also evident in Figure 4.21a which shows the NPV cost parameter uncertainty versus the predicted ensemble mean NPV for which there is a strong positive linear association. A robust choice in the assessment of the NPV cost parameter uncertainty is therefore the maximum variance over the decision space; $\mathrm{Var}[\varepsilon_{\mathrm{C}}] = 3.598 \times 10^{13}$. Construction of an emu-



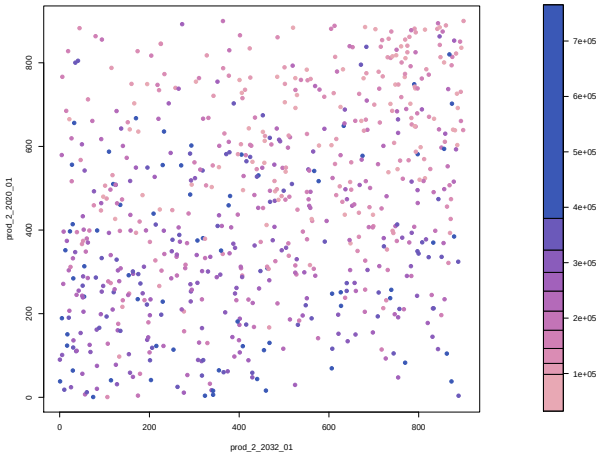
(a) prod_2_2016_01 versus prod_2_2018_01 prior to the removal of linear effects.



(b) prod_2_2020 versus prod_2_2032_01 prior to the removal of linear effects.



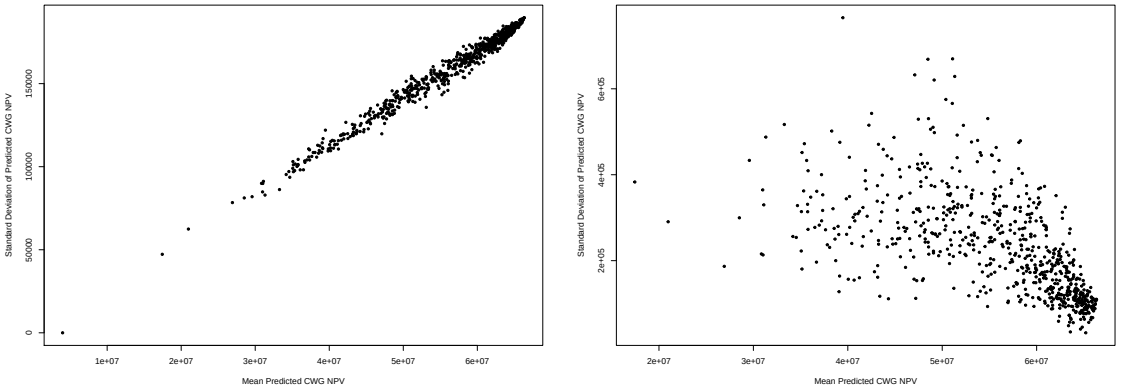
(c) prod_2_2016_01 versus prod_2_2018_01 after the removal of linear effects.



(d) prod_2_2020_01 versus prod_2_2032_01 after the removal of linear effects.

Figure 4.20: OLYMPUS wave 1 uncertainty quantification two-dimensional projections of the decision space coloured by the standard deviations of the ensemble mean NPV utility function due to the cost parameter variability. The top and bottom rows illustrate the uncertainty before and after the removal of linear effects respectively. The left column shows prod_2_2016_01 versus prod_2_2018_01, and the right column is of prod_2_2020_01 versus prod_2_2032_01. The colour scheme denotes the magnitude of the uncertainty on a relative scale with light pink through to dark purple corresponding to low through to high standard deviations.

lator or model for $\text{Var}[\varepsilon_C(\mathbf{d})]$ is necessitated to achieve a reduction in this uncertainty, whilst any uniform decrease would potentially yield non-robust conclusions. This is problematic for decision support because it makes it difficult to distinguish between decisions in this region where prior knowledge indicates that strategies are likely to yield the largest values of ensemble mean NPV.



(a) Before the removal of linear effects. (b) After the removal of linear effects.

Figure 4.21: OLYMPUS wave 1 ensemble mean NPV standard deviation induced by NPV cost parameter variability versus the ensemble mean NPV over the wave 1 design computed using the $n_c = 1000$ simulated cost parameter scenarios.

NPV Cost Parameter Uncertainty After the Removal of Linear Transformation Effects

For our uncertainty quantification we instead employ the more sophisticated approach detailed in Section 3.5.2 to first remove the linear transformation effects of this uncertainty since these do not alter the location of the optimal decision. For each of the n_c simulated cost parameter scenarios, linear models of the form shown in Equation (3.5.1) are fitted with a natural choice for $\mathbf{c}_0$ being the TNO fixed cost parameter settings. Assessing the variability in the residuals for fixed decisions via $s^2_C(\mathbf{d}_j)$ defined in Equation (3.5.3) provides a quantification of this uncertainty with the analogous projection plots shown in Figures 4.20c and 4.20d. The same colour scheme is used across the two versions of each plot which clearly illustrates the

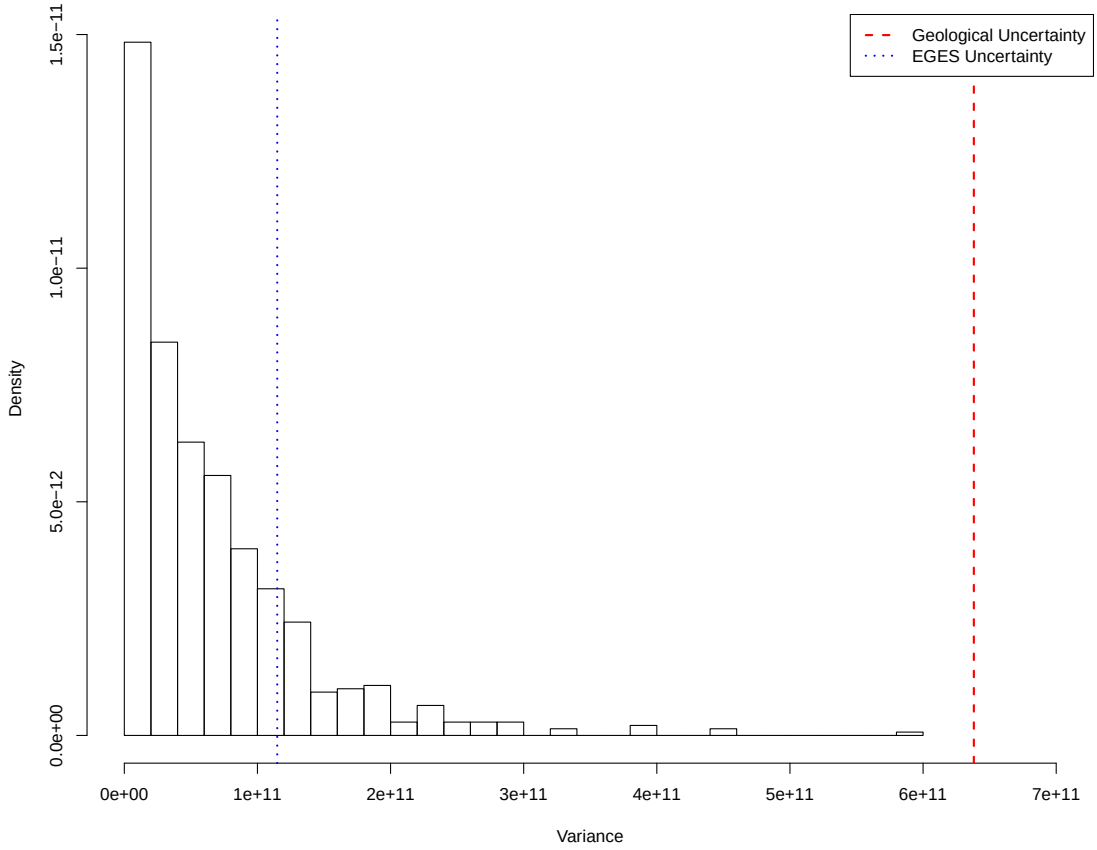


Figure 4.22: Histogram of the variances of the OLYMPUS ensemble mean NPV over the generated cost parameter scenarios after the linear effect is removed. For comparison, the dashed red and dotted blue vertical lines correspond to the assessed values for $\mathrm{Var}[\varepsilon_{\mathrm{G}}]$ and $\mathrm{Var}[\varepsilon_{\mathrm{EGES}}]$ respectively.

reversal in the direction of the association between larger parameter values and the level of uncertainty due to the cost parameter variation after linear effects have been accounted for. In contrast to the first row of plots, after the removal of the linear effects, the largest uncertainty is for smaller marginal parameter values which is a region thought to be of less interest and is expected to be ruled out as implausible at wave 1. An examination of the mean residuals over the cost parameters using $m_{\mathrm{C}}(\mathbf{d}_j)$ (see Equation (3.5.2)) highlights that these are typically at least an order of magnitude smaller than $s_{\mathrm{C}}(\mathbf{d}_j)$, and thus have negligible effect on the overall analysis.

The distribution of the variances of the utility function due to NPV cost para-

meter variation is shown in Figure 4.22 with the distribution centred at 6.639×10^{10}. It is noted that this uncertainty is not insubstantial and is of comparable magnitude to the EGES uncertainty. Since the observed pattern is that smaller uncertainties tend to occur for larger values of the ensemble mean NPV, $\text{Var}[\varepsilon_{\text{C}}]$ is estimated by the median over the decision space for wave 1 decision support, so $\text{Var}[\varepsilon_{\text{C}}] = 4.525 \times 10^{10}$. This is reinforced by the observations in Figure 4.21b where the standard deviation due to NPV cost parameter uncertainty is generally larger for smaller values of the predicted ensemble mean NPV; a reversal versus the observations in Figure 4.21a. A comparison with the results prior to removing the linear effects uses the more conservative maximum due to the association between high uncertainty and high ensemble mean NPV, so $\text{Var}[\varepsilon_{\text{C}}] = 3.598 \times 10^{13}$ (the median is 2.693×10^{13}), thus demonstrating a three order of magnitude reduction in this uncertainty. The implication for decision support is that a much smaller class of decisions may be obtained that are still robust to this uncertainty.

4.6.5 Structural Model Discrepancy

The aim in the TNO OLYMPUS Challenge is to maximise the expected NPV for the simulator rather than the actual field, hence there exists no model discrepancy within the challenge. Whilst it is important to provide a comprehensive assessment of the model discrepancy in order to link the computer model and the real world oil field, we also omit this from our analysis since the OLYMPUS reservoir is fictitious making it difficult to meaningfully assess, especially without access to the necessary reservoir engineering experts. Future work may introduce model discrepancy to provide a link between the modelling world and the real world physical problem and decision making. For an illustration of the assessment of model discrepancy, this is performed for the Emerson Jade model application in Sections 5.5.5 and 5.9.4.

4.6.6 Decision Implementation Error

As for model discrepancy, decision implementation error is omitted from both the TNO OLYMPUS Challenge and our analysis due to the OLYMPUS reservoir model being fictitious. Prior expert knowledge from reservoir engineers would be required in order to meaningfully assess the implementation error with this being an area for future work. An uncertainty analysis for the decision implementation error [31] following the approach described in Section 3.5.4 is presented for the Emerson Jade model application in Sections 5.5.6 and 5.9.5.

4.6.7 Uncertainty Analysis Summary

A summary of the wave 1 uncertainty analysis is shown in Table 4.4 which includes the variances and the proportion of the total uncertainty both including and excluding the EGES uncertainty corresponding to the use of a Bayes linear or the novel hierarchical emulator for the utility function in the decision support analyses in Sections 4.7.1 and 4.7.2 respectively. The individual uncertainties, excluding that due to the emulator, are combined following the uncorrelated additive error structure described in Equation (2.7.1) to obtain a total uncertainty variance of 6.834×10^{11}. Note that this variance should be compared to the scale of the ensemble mean NPV which is typically of order of 5.0×10^7 \$ to 6×10^7 \$.

4.6.8 Comparison of Uncertainty Quantification with the Utility Function Emulator Uncertainty

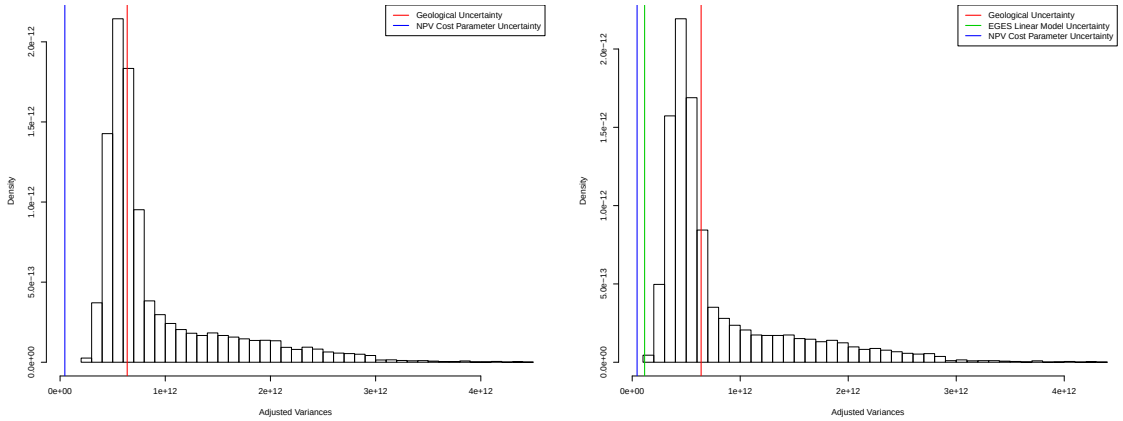
In this section we compare the uncertainty due to the hierarchical emulator for the ensemble mean NPV with the assessment of each of the major sources of uncertainty relevant to decision support. Figure 4.23a is a histogram of the hierarchical emulator adjusted variances where red and blue lines denote the variance in the utility function induced by the geological and NPV cost parameter uncertainty respectively. It is observed that geological uncertainty is substantially larger than the NPV cost

Source of Uncertainty	Variance	Proportion of Total Uncertainty (inc EGES)	Proportion of Total Uncertainty (exc EGES)
Geological Uncertainty	6.381×10^{11}	79.9%	93.4%
EGES Uncertainty	1.15×10^{11}	14.4%	–
NPV Cost Parameter Uncertainty	4.525×10^{10}	5.7%	6.6%
Total Uncertainty	6.834×10^{11}	–	–

Table 4.4: OLYMPUS wave 1 uncertainty analysis results including the variances and proportion of total variances for TNO OLYMPUS Well Control Optimisation Challenge obtained using an additive error structure described in Equation (2.7.1). Columns three and four show the proportions of the total uncertainty either including or excluding the EGES uncertainty which corresponds to the decision support analysis using the Bayes linear or hierarchical emulator in Sections 4.7.1 and 4.7.2 respectively. For comparison, the ensemble mean NPV is typically of the order of 5.0×10^7 \$ to 6×10^7 \$.

parameter uncertainty, although the hierarchical emulator uncertainty for many new decision parameter vectors is the largest source of uncertainty within the analysis. Following on from previous discussion in Section 4.5.4, the decision scenarios which yield higher utility function values also tend to correspond to the largest uncertainties, hence the emulator uncertainty is expected to dominate the wave 1 analysis. Note that there exist some scenarios where the geological uncertainty is larger or of a comparable order of magnitude to the emulator uncertainty, thus highlighting the need for its inclusion and rigorous treatment. In Section 4.5.5 it was identified that the Bayes linear approach generally gave rise to larger adjusted variances than the hierarchical emulator, hence these comments are also true when a Bayes linear emulator is in use. Moreover, this justifies the additional effort in constructing the hierarchical emulator in order to reduce the uncertainty to closer to the limiting wave 1 uncertainty constituted by this analysis.

The construction of the hierarchical emulator implicitly incorporates the EGES



(a) Full hierarchical emulator adjusted variances, including the EGES residual uncertainty.

(b) Hierarchical emulator adjusted variances separating out the EGES residual uncertainty, as denoted by the green line.

Figure 4.23: Histograms of the OLYMPUS wave 1 ensemble mean NPV hierarchical emulator adjusted variances compared with the variances for each assessed source of uncertainty. The red and blue lines denote the variance in the utility function induced by the geological and NPV cost parameter uncertainty respectively. The additional green line in Figure 4.23b denotes the EGES linear model squared residual standard error.

uncertainty for linking the NPV of the three sub-selected OLYMPUS models to the ensemble mean NPV. In order to additionally assess the relative contribution of the EGES uncertainty, the residual error of the EGES linear model is disentangled from the emulator formulation for which the reduced version of the hierarchical emulator adjusted variances along with the three sources of uncertainty are illustrated in Figure 4.23b. This corresponds to the removal of the $\hat{\sigma}^2_{\text{EGES}}$ term within Equation (3.3.17) which is marked by the green line. It is observed that $\hat{\sigma}^2_{\text{EGES}}$ is in all cases dominated by the remaining hierarchical emulator adjusted variances; generally comprising of between 6% and 25% of the total hierarchical emulator adjusted variance. The overall uncertainty is therefore dominated by sources pertaining to the NPV constituents.

4.7 Iterative Decision Support for TNO OLYMPUS Well Control Optimisation

The TNO OLYMPUS Field Development Optimisation Challenge instructs participants to locate the decision parameters setting which yields the maximum ensemble mean NPV as a representative for the expected NPV. In this section we instead address the reformulated challenge as a decision support problem where $U(\mathbf{d}) = \mathbb{E}[\text{NPV}](\mathbf{d})$ demonstrating the iterative decision support procedure developed in Section 3.6. This incorporates the EGES technique, construction of a wave 1 targeted Bayesian design and the uncertainty quantification implemented in Sections 4.2, 4.3 and 4.6 respectively. Decision support is performed using two forms of the emulator: the Bayes linear emulator (see Section 4.4) in Section 4.7.1; and the hierarchical emulator exploiting known simulator behaviour (see Section 4.5) in Section 4.7.2. Results are compared under alternative uncertainty analyses in Section 4.7.3.

4.7.1 Decision Support Using a Bayes Linear Emulator

The first wave of the iterative decision support procedure is presented for the expected NPV using the decision support implausibility measure described in Equation (3.6.3). In this part of the analysis we use the Bayes linear emulator constructed in Section 4.4 for the utility function, and incorporate the geological, EGES and NPV cost parameter uncertainties as quantified in Section 4.6. Note that it is necessary to explicitly include the EGES uncertainty as unlike the hierarchical emulator, this is not directly accounted for within the emulator formulation. The non-implausible region, which would be explored further in wave 2, is defined as points satisfying $I_{\text{DS}}(\mathbf{d}) \leq c = 3$ for which it is found that 34.69% of the constrained decision space is classified as non-implausible. Note that imposing the difference constraints has already reduced the decision space volume to approximately 3.4529% of the initial hypercube volume, hence this actually corresponds to a reduction to 1.20% of the

full hypercube volume.

Results for wave 1 are illustrated in Figure 4.24, which presents a pairs plot for the active decision parameters of the Bayes linear emulator displaying the wave 1 decision support implausibility measures below the diagonal, the implausibility optical density above the diagonal, and marginal density histograms along the diagonal. Note that for the implausibility plots, the axes are swapped to match the corresponding optical density plot with the colours green, yellow and red indicating where $0 < I_{\mathrm{DS}}(\mathbf{d}) \leq 2$, $2 < I_{\mathrm{DS}}(\mathbf{d}) \leq 3.5$ and $I_{\mathrm{DS}}(\mathbf{d}) > 3.5$ respectively. Blue triangles denote where the implausibility measure is zero and hence the emulator for the expected NPV adjusted expectation exceeds the best simulated value. For the optical density plots, dark blue through to yellow depict low to high density of points satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c$. These plots aid the visualisation of the locations of the best decisions, as well as any relationships between the non-implausible decision parameter settings. It is evident from the implausibility plots that early time production rates, particularly for producer well 2 seen in the top-left block of 4 variables, are most important for achieving a high expected NPV and so are required to take high values. This is also supported by the marginal density histograms where the majority of the mass is placed on higher parameter values with a decreasing trend at later times. Moreover, the blue triangles denoting $I_{\mathrm{DS}}(\mathbf{d}) = 0$ are mostly clustered for higher decision parameter values.

Later time target production rates are more free to vary, at least within this wave 1 analysis, as illustrated in the central block of variables 5–8. For example, the value of prod_2_2032_01 seems to have little impact; there exist decision vectors for most values of prod_2_2032_01 which are classified as non-implausible, as long as the other decision parameters, particularly those at early times, take suitably large values. This is apparent within its marginal density histogram which is the closest to uniform. Note that these histograms must take into account the edge effects due to imposing the difference constraints and so are not actually uniform for the initial parameter space, as would usually occur for a hypercube volume. The

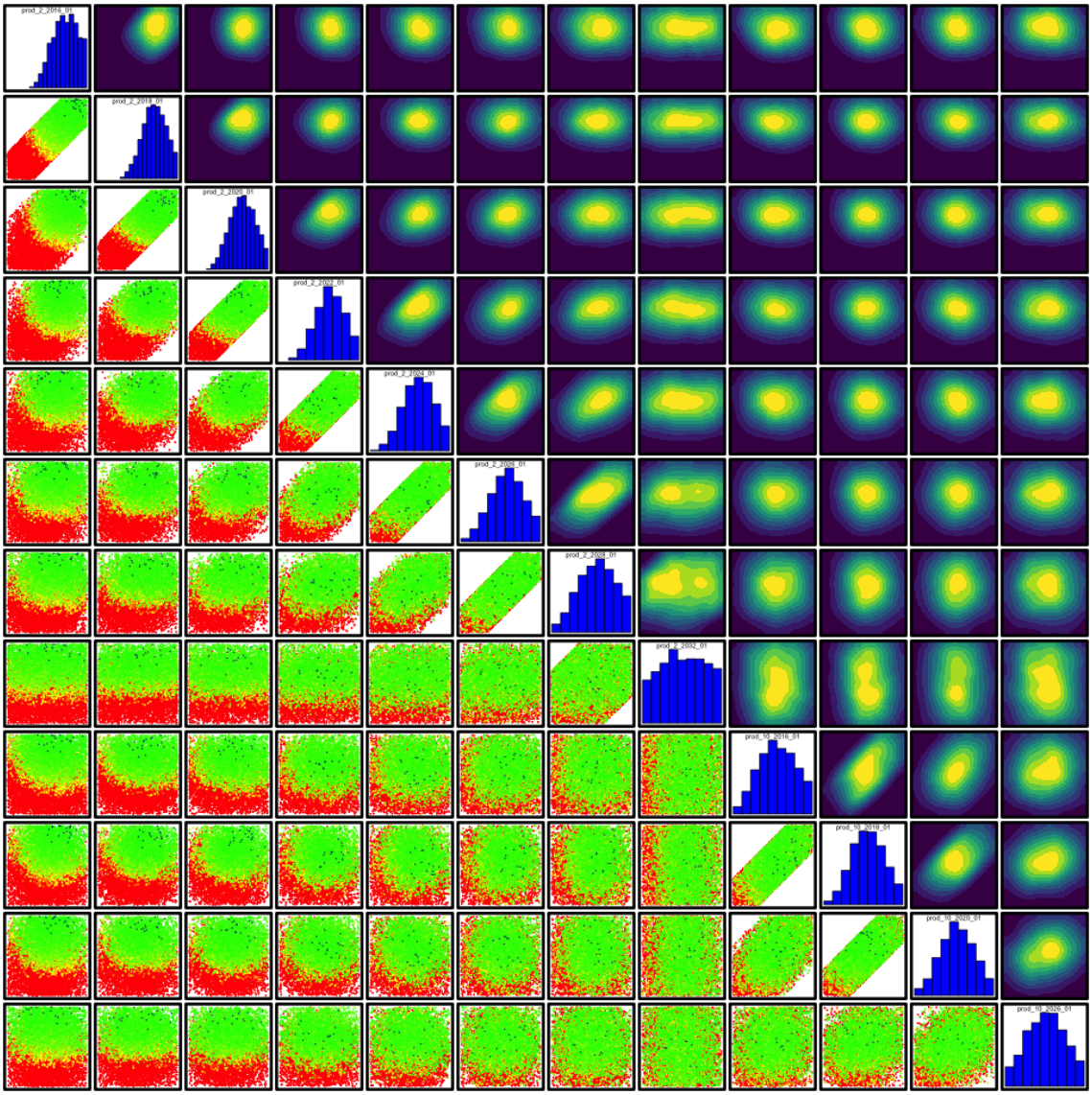


Figure 4.24: OLYMPUS wave 1 decision support implausibility measure (below the diagonal) and optical density (above the diagonal) pairs plot, with marginal density histograms (along the diagonal) for the active decision parameters. This is for the analysis using a Bayes linear emulator for the utility function and incorporating geological, EGES and NPV cost parameter uncertainties. For the implausibility plots, the axes are swapped to match the corresponding optical density plot. The colours green, yellow and red indicate where $0 < I_{\mathrm{DS}}(\mathbf{d}) \leq 2$, $2 < I_{\mathrm{DS}}(\mathbf{d}) \leq 3.5$ and $I_{\mathrm{DS}}(\mathbf{d}) > 3.5$ respectively, whilst blue triangles denote where $I_{\mathrm{DS}}(\mathbf{d}) = 0$ which occurs when $\mathbb{E}_F[U(\mathbf{d})] \geq U_{\max}$. For the optical density plots, dark blue through to yellow depict low to high density of points satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c$.

unimportance of prod_2_2032_01 is likely due to the heavy effect of discounting within the NPV calculations. In addition, it is evident that decision parameters for the same well which are temporally close are correlated in their values which potentially lead to high expected NPV and is best highlighted by the implausibility plots just below the diagonal. Decision support implausibility plots can also be used to screen for whether controls for certain wells have a stronger effect on the expected NPV as seen by comparing variables 1–3 and 6, with 9–12 respectively in Figure 4.24 where it is evident that controls for producer well 2 have a greater effect on the expected NPV than those producer well 10. Such patterns can also be seen in the optical density plots which possess a higher mass of points within regions where both decision parameters take higher values. Furthermore, the effect of imposing the difference constraints is also visible with the narrow elliptical contours for temporally close points, and relatively low density within the corner regions. This corroborates reservoir engineering insight and the postulated beliefs based on Figure 4.8 that early time production rates are dominant in determining whether a high utility is achieved, whilst decisions for producer well 2 are of greater importance than for producer well 10. In addition, this analysis confirms the effect of a high discount factor.

4.7.2 Decision Support Using a Hierarchical Emulator Exploiting Known Simulator Behaviour

The first wave of the iterative decision support procedure is performed using the hierarchical emulator exploiting known simulator behaviour, as devised in Section 4.5 for the utility function. This version of the analysis explicitly incorporates the geological and NPV cost parameter uncertainties as quantified in Sections 4.6.2 and 4.6.4 respectively, whilst the EGES uncertainty is encompassed within the hierarchical emulator formulation. As in Section 4.7.1, the wave 1 non-implausible region is defined as decision parameter vectors which satisfy $I_{\mathrm{DS}}(\mathbf{d}) \leq c = 3$. It is established that 25.30% of the decision space is classified as non-implausible; a

marked reduction on the 34.69% in the analysis using the Bayes linear emulator and necessary alteration to the uncertainty quantification. This is a consequence of the general decrease in the emulator uncertainty discussed in Section 4.5.5 and thus highlights the benefits of the additional efforts and computational costs in constructing the hierarchical emulator which mitigates the requirement for further exploration in a wave 2 analysis via additional expensive computer simulations within this extra volume that would otherwise have been deemed non-implausible. Moreover, due to the difference constraint this equates to 0.87% of the full hypercube volume remaining after wave 1, compared to 1.20% using the Bayes linear emulator.

Visualisation of the wave 1 decision support non-implausible region is conducted in the same manner with the implausibility pairs plot for producer well 2 and for all decision parameters starting in 2016 and 2018 shown in Figures 4.25 and 4.26 respectively, where the plot types and colour schemes are the same as in Figure 4.24. Note that the target production rates in Figure 4.25 correspond to the first eight active decision parameters in Figure 4.24. Examination of these plots leads to similar conclusions including the importance of early time production rates, particularly for producer well 2, which must be set to high values, whilst the blue triangles designating where $I_{\mathrm{DS}}(\mathbf{d}) = 0$ are mostly clustered where both decision parameters have high values in each 2-dimensional plot. In Figure 4.25 it is also evident that later time target production rates have greater flexibility in their values, again emphasising the impact of the high assumed discount factor. This is perceivable from the close to uniform marginal histogram for prod_2_2032_01. Exploration of the relationship between decision parameters for different wells such as in Figure 4.26 illustrates that target rates for producer well 2 have a dominant effect on determining whether a high utility is achieved versus those for producer well 10. Furthermore, investigations show that within the wave 1 decision support analysis the target injection rates are relatively free to vary, as observed in the implausibility plots for variables 5–8 in Figure 4.26 compared both internally and with the target production rates. From the optical density plots and marginal density histograms, the only condition is that

for early time their value is not too small. Moreover, the implausibility plots for prod_2_2016_01 and prod_2_2018_01 highlight that high target production rates are unsustainable if these are not matched by sufficiently high target injection rates, as indicated by the red points within this region for these four decision parameters. This is explained since low injection leads to insufficient fluid within the field at later times to maintain a sufficiently high BHP at the producer wells. Consequently high, and in some cases even modest assigned target production rates cannot be achieved in much the same way as is observed in the exploratory analysis in Figure 4.4 which severely inhibits the ability of a well control strategy to attain a high expected NPV. This observation is also apparent for later time target injection rates, although to a lesser extent, also a consequence of the high discount factor and the implications of aiming for higher early production.

Application of the first wave of iterative decision support to the TNO OLYMPUS Well Control Optimisation Challenge provides insight into which decision parameter settings are likely to yield high expected NPV utility function values. It is identified that target rates for producer well 2 need to be relatively high, especially for earlier times, whilst for producer well 10, these are less important but should still not be too low. In addition, target injection rates are permitted to vary more freely, but must still be of sufficient value to support the stipulated target production rates, both at the current time and in the future. This aligns with the beliefs of reservoir engineers. Our analysis is restricted to wave 1 of the developed iterative decision support framework because of limitations on time and access to the necessary computing resources. The next step would be to design and evaluate a second wave of simulations before emulating the utility function over the reduced decision parameter space. Along with a reassessment of the uncertainties, this would be used to compute the decision support implausibility measure in order to achieve further reductions in the non-implausible volume and gain greater insight into this region.

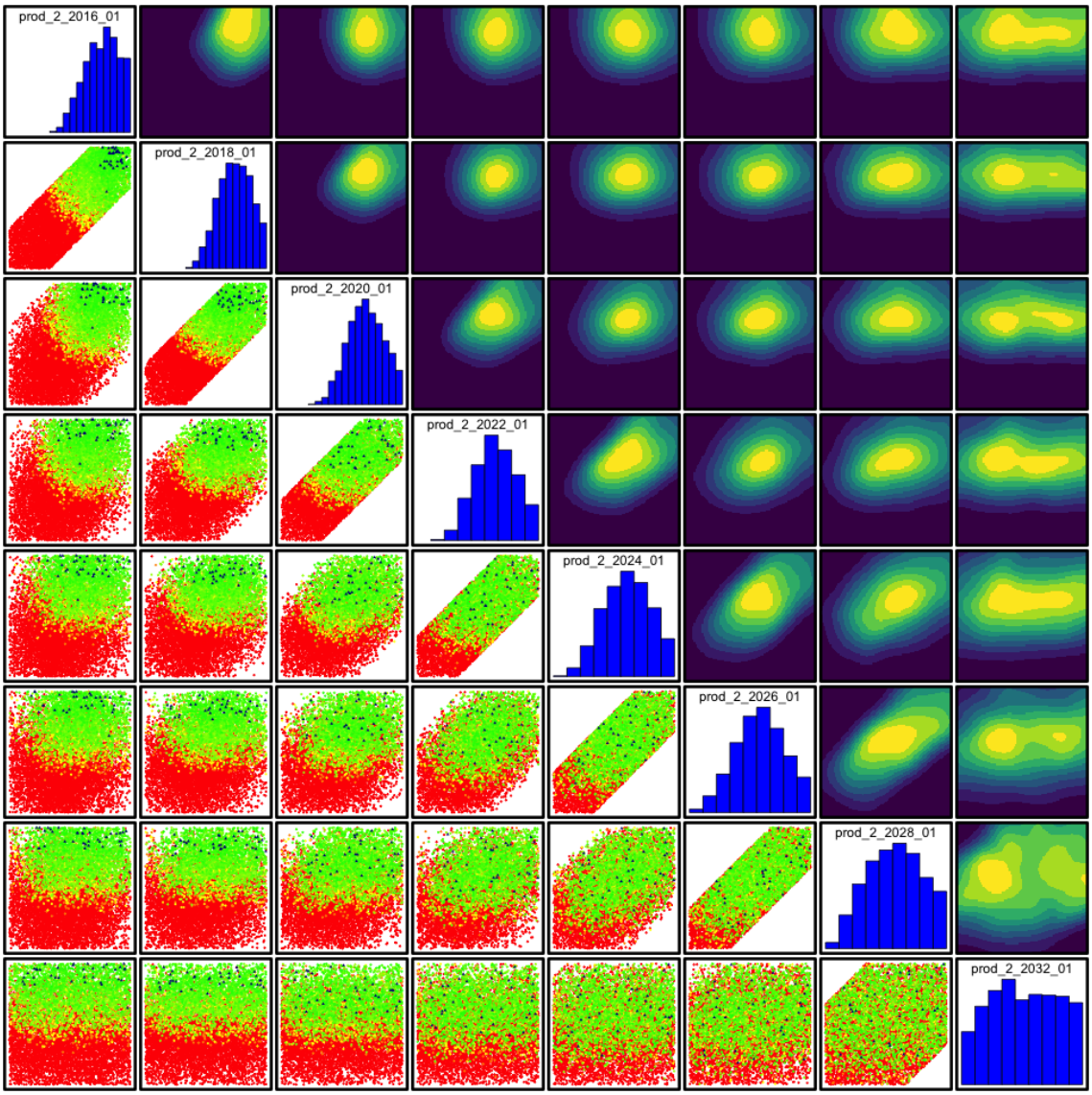


Figure 4.25: OLYMPUS wave 1 decision support implausibility measure (below the diagonal) and optical density (above the diagonal) pairs plot, with marginal density histograms (along the diagonal) for the decision parameters for producer well 2. This is for the analysis using a hierarchical emulator exploiting known simulator behaviour for the utility function and explicitly incorporating the geological and NPV cost parameter uncertainties. For the implausibility plots, the axes are swapped to match the corresponding optical density plot. The colours green, yellow and red indicate where $0 < I_{\mathrm{DS}}(\mathbf{d}) \leq 2$, $2 < I_{\mathrm{DS}}(\mathbf{d}) \leq 3.5$ and $I_{\mathrm{DS}}(\mathbf{d}) > 3.5$ respectively, whilst blue triangles denote where $I_{\mathrm{DS}}(\mathbf{d}) = 0$ which occurs when $\mathbb{E}_F[U(\mathbf{d})] \geq U_{\max}$. For the optical density plots, dark blue through to yellow depict low to high density of points satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c$.

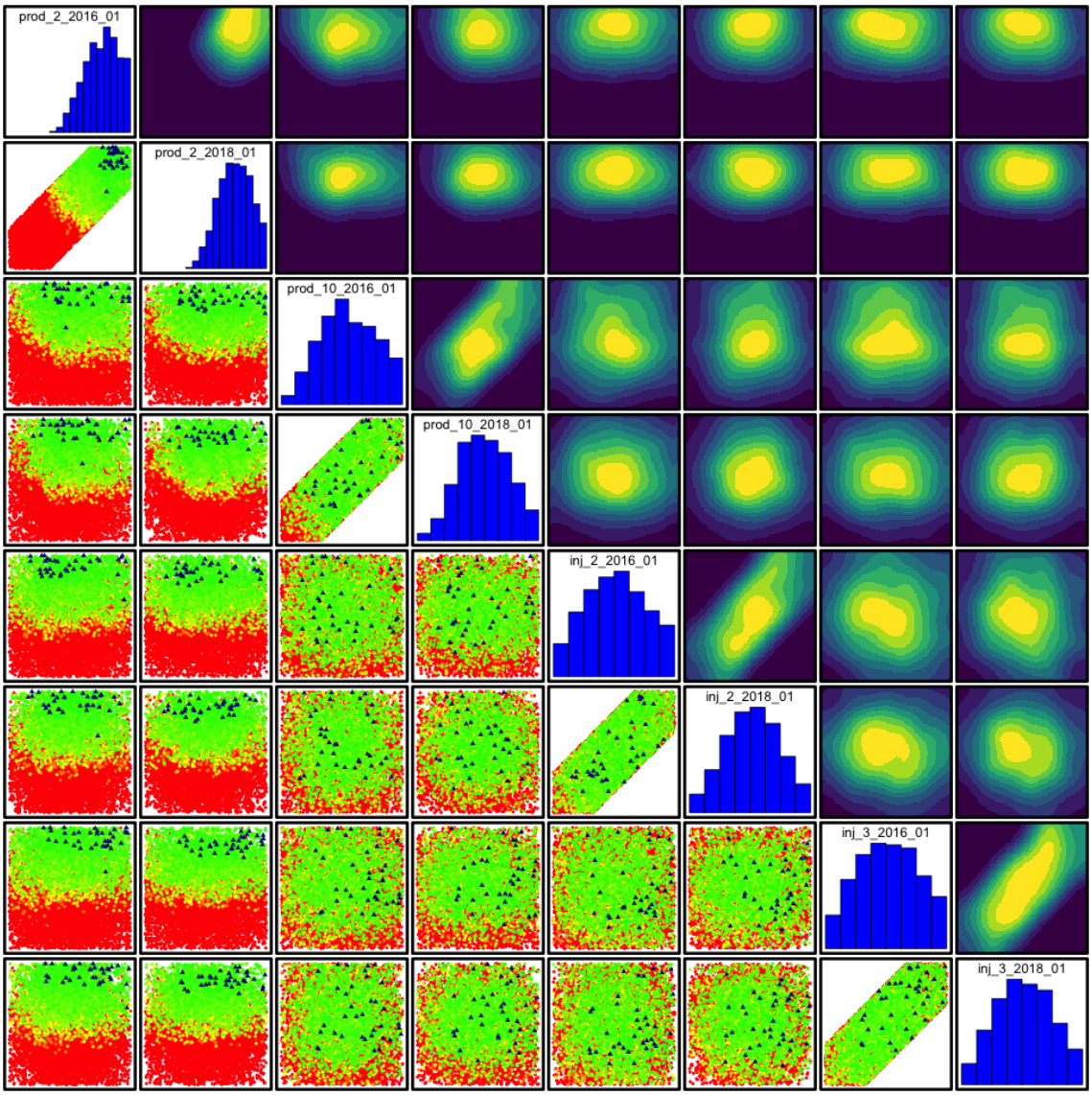


Figure 4.26: OLYMPUS wave 1 decision support implausibility measure (below the diagonal) and optical density (above the diagonal) pairs plot, with marginal density histograms (along the diagonal) for all decision parameters starting in 2016 and 2018. This is for the analysis using a hierarchical emulator exploiting known simulator behaviour for the utility function and explicitly incorporating the geological and NPV cost parameter uncertainties. For the implausibility plots, the axes are swapped to match the corresponding optical density plot. The colours green, yellow and red indicate where $0 < I_{\mathrm{DS}}(\mathbf{d}) \leq 2$, $2 < I_{\mathrm{DS}}(\mathbf{d}) \leq 3.5$ and $I_{\mathrm{DS}}(\mathbf{d}) > 3.5$ respectively, whilst blue triangles denote where $I_{\mathrm{DS}}(\mathbf{d}) = 0$ which occurs when $\mathbb{E}_F[U(\mathbf{d})] \geq U_{\max}$. For the optical density plots, dark blue through to yellow depict low to high density of points satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c$.

4.7.3 Decision Support Results for Alternative Uncertainty Quantifications

The iterative decision support procedure using an implausibility measure developed in Section 3.6 has the major advantage compared to optimisation and decision theoretic approaches that it is computationally quick and efficient to perform a decision support sensitivity analysis with respect to the uncertainty quantification. In this section we highlight the importance of a comprehensive uncertainty quantification for the TNO OLYMPUS Well Control Optimisation Challenge by repeating the wave 1 decision support application under alternative uncertainty specifications. The results are summarised in Table 4.5 where the hierarchical emulator exploiting known simulator behaviour is employed for the utility function due to the established superior decision support performance versus the implementation using the Bayes linear emulator; this is also included for comparison. For clarity, we exhibit the non-implausible proportion of the wave 1 decision space formed by imposing both the range and difference constraints in the middle column, and portray the non-implausible proportion of the entire hypercube volume prior to the inclusion of the difference constraints in the right column. All subsequent commentary is framed in terms of the fully constrained decision space.

In Section 4.6.4 we consider the differences before and after accounting for the linear transformation effect of the NPV cost parameter uncertainty on the utility function obtaining an assessment of $\mathrm{Var}[\varepsilon_{\mathrm{C}}] = 3.598 \times 10^{13}$ and $\mathrm{Var}[\varepsilon_{\mathrm{C}}] = 4.525 \times 10^{10}$ respectively. With all other uncertainties as previously specified, it is ascertained that prior to removing the linear effects of the NPV cost parameter uncertainty then the non-implausible proportion is 91.60%. This makes it extremely difficult to distinguish between almost all of the decision parameter settings, thus compromising the ability to provide meaningful decision support. Consequently it is necessary to consider how the uncertainty quantification is performed within the context of the wider objective, as demonstrated by the added value to the analysis. In this

Uncertainty Specification	Non-Implausible Volume	Non-Implausible Volume of Hypercube
Bayes Linear Emulator with All Uncertainties (see Section 4.7.1)	34.69%	1.20%
Hierarchical Emulator with All uncertainties (see Section 4.7.2)	25.30%	0.87%
Before Removing NPV Cost Parameter Uncertainty Linear Effect	91.60%	3.16%
Emulator Uncertainty Only	16.11%	0.56%
No Emulator Uncertainty	10.68%	0.37%
No Uncertainties or Emulator Uncertainty	0.36%	0.012%

Table 4.5: Summary of the proportion of decision space classified as non-implausible after wave 1 decision support for TNO OLYMPUS Well Control Optimisation Challenge for various included uncertainties within the decision support implausibility measure using threshold value $c = 3$. The first row specifies the proportion using a Bayes linear emulator. All subsequent rows use the hierarchical emulator for the utility function. The middle column shows the non-implausible proportion of the volume of the decision space formed by imposing both the range and difference constraints. The right column portrays the non-implausible proportion of the entire hypercube volume prior to the inclusion of the difference constraints. Imposition of the difference constraints results in a reduction to 3.4529% of the initial hypercube constituted by the range constraints only.

case uncertainties which linearly transform the utility function have no effect on the location of the maximum and hence should be removed. This result is shown in the third row of Table 4.5.

Omitting all uncertainty except for those due to the emulator as a surrogate for the OLYMPUS expected NPV yields a non-implausible volume of 16.11% of the decision space (see the fourth row of Table 4.5). This corresponds to setting $\mathbb{E}[\varepsilon_i] = 0$ and $\text{Var}[\varepsilon_i] = 0$ for all uncertainties representing the prior belief that up to emulator uncertainty, there is no difference between the real world OLYMPUS field and the model; specifically their underlying geology is known and can be represented

by the prediction of the ensemble mean NPV obtained via the EGES linear model, whilst the future time evolution of NPV cost parameters is also precisely known. The results highlight a marked difference versus the 25.30% initially achieved, thus demonstrating that it is necessary to perform a thorough uncertainty quantification to circumvent the risks of falsely eliminating regions of the decision space which may contain good choices of well controls. At later stages within the iterative decision support procedure, if it is believed that these uncertainties are smaller (or negligible) then they may be subsequently re-assessed (or removed) to obtain a reduced non-implausible volume. Realistic and representative uncertainty quantifications should be incorporated during earlier waves of the analysis.

If the emulator is assumed to be a perfect representation of the OLYMPUS model but including all other analysed sources of uncertainty, then wave 1 decision support classifies 10.68% of the decision space as non-implausible (see the fifth row of Table 4.5). This is equivalent to treating the emulator expectation as equal to the simulator output. Such an investigation constitutes an approximation of the decision support analysis using the simulator directly assuming there are sufficient computational resources available. The wave 1 decision support result provides a theoretical lower bound on the proportion of space remaining at wave 1 given the current uncertainty quantification and suggests that improvement in the Bayesian emulator accuracy has the potential to greatly reduce this proportion further at wave 1 from the current 25.30% closer to the much smaller 10.68% without the use of any additional simulations. This motivates further research to improve the accuracy of the emulator for the expected NPV.

A final extreme situation is to suppose that there are no uncertainties present from either the emulator or the uncertainty quantification and search for where the emulator yields predictions that exceed the current maximum simulated expected NPV. This is equivalent to the proportion of points with $I_{\mathrm{DS}}(\mathbf{d}) = 0$ for which results show only 0.36% of the decision space satisfy this condition, as seen in the sixth row of Table 4.5. This is analogous to assuming a perfect representation of

the real world system within the modelling framework (including the emulator), for which the emulator matches the OLYMPUS model output and there are no uncertainties. In this setup many potentially good decision strategies are falsely rejected, although it also provides insight of which well control strategies may lead to expected improvement versus the current best simulated strategy.

4.8 Decision Support Conclusion

In this chapter we have presented an analysis of the TNO OLYMPUS Well Control Optimisation Challenge reformulated as a decision support problem to which we have applied the methodology developed in Chapter 3. This begins with a demonstration of the computational efficiency gains which can be achieved using the developed efficient geological ensemble subsampling techniques, before devising and implementing a method for constructing targeted Bayesian designs for decision support. Two surrogates for the complex and computationally expensive OLYMPUS ensemble are devised including Bayes linear emulators and a hierarchical emulation framework which exploits known structure of the simulator outputs. An uncertainty quantification is performed within the context of decision support. This toolkit is utilised for a first wave application of the developed iterative decision support procedure to yield a robust class of decisions respecting all major sources of uncertainty. Much of the methodology developed for well control optimisation as a decision support problem including the application to the TNO OLYMPUS model was presented at ECMOR XVII 2020 and published in the conference paper [135]. Along with subsequent further research results presented in Sections 4.5, 4.7.2 and 4.7.3, a complete application is in preparation for publication.

The TNO OLYMPUS Well Control Optimisation Challenge uses an ensemble of models to represent geological uncertainty, a form of ensemble uncertainty and a common practice within the petroleum industry. We highlight the limitations of using an ensemble, particularly in terms of computational expense involved in

exploring the decision parameter space implementing the EGES techniques described in Section 3.2 to identify a small representative subset of three models for only a modest increase in the uncertainty relative to other sources. A careful method is formulated for constructing a targeted Bayesian design that incorporates prior knowledge from oil reservoir engineers regarding feasible and physically realistic well control strategies in the form of time consecutive difference constraints. Application yields a decrease to approximately 3.4529% of the original hypercube volume whilst also achieving good sampling coverage.

In the reformulation of the TNO OLYMPUS Well Control Optimisation Challenge as a decision support problem the expected NPV objective function is treated as a utility function. Bayesian emulators are used as fast and efficient statistical approximations for the expected NPV enabling the full exploration of the wave 1 decision parameter space given only a collection of 702 simulations in a process that is easy to execute in parallel. Initially Bayes linear emulators are used, however whilst these are effective, they fail to incorporate the structure within the contributors to the NPV formula. The Bayesian emulation framework is extended in Section 3.3 to exploit known simulator behaviour by decomposing the utility function which yields generally greater accuracy in Section 4.5.

Uncertainty quantification is performed for various sources relevant to decision support. It is first highlighted that geological uncertainty is the most substantial source after the emulator and thus warrants a more careful treatment than is conventionally given within the petroleum industry, whilst the introduced EGES uncertainty is shown to be approximately an order of magnitude smaller. Moreover, this analysis recognises how only uncertainties which affect the location of the optimal decision need to be included achieving a three order of magnitude reduction in the variance due to NPV cost parameter uncertainty after accounting for linear transformation effects. Model discrepancy and decision implementation error are not incorporated as the OLYMPUS model is of a fictitious reservoir, hence these are difficult to meaningfully assess without access to a reservoir engineer.

Implementation of the first wave of the iterative decision support procedure detailed in Section 3.6 produced a robust class of decisions respecting all major sources of uncertainty whilst providing insight into the structure of the region which is believed to yield high expected NPVs. The greater accuracy of the hierarchical versus Bayes linear emulator is mirrored in decision support by a noticeable additional reduction in the non-implausible decision space. Furthermore, the necessity for an accurate and comprehensive uncertainty quantification is emphasised in Section 4.7.3. The omission of all uncertainties except the emulator leads to a drastic reduction in the non-implausible volume, whilst accounting for linear transformation effects due to the NPV cost parameter uncertainty is highlighted as an important step in obtaining useable results. Emulator uncertainty is shown to be the largest uncertainty which may be reduced in a second wave via additional simulations within the smaller non-implausible decision space. Additional reductions to the non-implausible volume, either in this wave 1 analysis or for a wave 2, may be achieved through a re-assessment of the conservative upper bound estimate for the variance due to the geological uncertainty through an application of variance emulation techniques such as in [2, 3]. The EGES uncertainty requires extra OLYMPUS models to be used for simulations in order to reduce this uncertainty. This comes at the computational expense of additional simulations which may inhibit the ability to simulate over a large range of decision parameter vectors which in turn will affect the accuracy of the fitted emulator.

The next stage of this analysis is to perform subsequent waves of decision support to further reduce the non-implausible decision space utilising the iterative nature of the procedure to refocus on the current non-implausible region. This was not performed during this PhD since it requires the evaluation of additional simulations at a large computational expense. Moreover, we used the open source simulation software, OPM Flow [161], with limited access to technical support, exacerbated by the Covid-19 pandemic and changing cluster facilities related to moving to a new building for the Department of Mathematical Sciences, that would have been

afforded by access to the expensive commercial software suggested for the TNO OLYMPUS Challenge. Another area for future investigations is to validate the well control strategies which lie within the decision support non-implausible region. At this stage it is not possible because the underlying stochastic OLYMPUS geology model has not been made available.

Chapter 5

Commercial Client Placement Analysis: Emerson Jade Model Well Control Optimisation Application

We present an application of the methodology developed during this PhD for the commercial client placement part of my iCASE PhD studentship with Emerson. This is to demonstrate how the conducted research benefits our industry partner by enabling further development of their software capability to incorporate novel statistical techniques for decision support. The ability to perform such an analysis is of increasing importance across the petroleum industry, as seen by the establishment of the TNO OLYMPUS Field Development Optimisation Challenge [93] and participation in the EAGE/TNO Workshop on OLYMPUS Field Development Optimization [39]. In this chapter we apply the methodology described in Chapter 3 to the Emerson Jade model [164] for decision support, and therefore serves as a complete example of the iterative decision support procedure. The initial part of this analysis follows a similar structure to that presented for the TNO OLYMPUS Well Control Optimisation Challenge in Chapter 4 which is then extended to also include:

the quantification of structural model discrepancy and decision implementation error; construction of a wave 2 design; a second wave of iterative decision support which encompasses a reassessment of several of the uncertainties; and validation of the decision support results. These supplementary parts to the application are possible due to access to sufficient computational resources, simulation software support, reservoir engineering insight for the Jade model, and access to the underlying stochastic geology model; all of which were missing for the TNO OLYMPUS Well Control Optimisation Challenge application.

The Emerson Jade model is introduced in Section 5.1 including an exploratory analysis. Efficient geological ensemble subsampling is implemented in Section 5.2 with a wave 1 targeted Bayesian design constructed in Section 5.3. Given the observed results for the TNO OLYMPUS Well Control Optimisation Challenge, the expected NPV utility function is emulated using the hierarchical Bayesian emulator which exploits known structures in the behaviour of simulator output in Section 5.4. A wave 1 uncertainty analysis is performed for the Jade model in Section 5.5 which is utilised with the hierarchical emulator in the novel iterative decision support procedure with wave 1 results presented in Section 5.6. Extending on the TNO OLYMPUS Well Control Optimisation Challenge application, a wave 2 design over the wave 1 non-implausible decision space is constructed in Section 5.7 before constructing a new Bayesian emulator over this space in Section 5.8 and reassessing the uncertainties in Section 5.9. The results of a second and final wave of iterative decision support are described in Section 5.10 with the identified non-implausible region validated in Section 5.11. A comparison of the results of simulations performed at each wave of the iterative decision support procedure is presented in Section 5.12 with a summary and conclusion in Section 5.13.

5.1 Introduction to the Emerson Jade Model

In this section we introduce the Emerson Jade model for which decision support will be performed. This includes further details of the Jade model and the setup for decision support in Section 5.1.1 with a discussion of an exploratory analysis of a preliminary batch of simulations in Section 5.1.2.

5.1.1 Overview of the Emerson Jade Model

The Emerson Jade model is a fictitious oil reservoir model that is used by Emerson to test their software and to demonstrate its capabilities to prospective clients. We use the Jade model to demonstrate our developed decision support framework for well control optimisation which may then be subsequently incorporated in Emerson's Tempest software packages and in demonstrations to clients. The aim is to identify a collection of well control strategies, each represented by a vector of decision parameters, $\mathbf{d}$, for the Jade model which yield near maximum values for the utility function with respect to all major sources of uncertainty that are quantified within the analysis. For this application we specify the utility function to be the expected NPV, $U(\mathbf{d}) = \mathbb{E}[\text{NPV}](\mathbf{d})$, which represents a position of risk neutrality. This is calculated by first computing the NPV for each individual Jade model using Equation (3.1.1), with $R_j(\mathbf{d}, t_i)$ as specified in Equation (3.1.3) with full details of these formulae in Section 3.1.1. For the Emerson Jade model the field lifetime is 10 years starting January 1, 2022, discounting is applied at 3 month intervals with a discounting time period of 1 year, and the NPV cost parameters are fixed at the suggested values of \$30, \$6 and \$2 per barrel for oil produced, water produced and injected respectively, with a discount factor of 10%, $d = 0.10$ (see Table 5.2 on page 249 for a summary). This is a higher discount factor than in the TNO OLYMPUS Well Control Optimisation Challenge and thus encourages aiming for larger oil production volumes at early times due to the higher penalty at later times. Uncertainty due to variability in the NPV cost parameters is investigated in Section 5.5.4. Once the NPV has

been computed for each Jade model, the expected NPV is approximated by the ensemble mean NPV defined in Equation (3.1.2). As for the TNO OLYMPUS Field Development Optimisation Challenge and inline with standard industry practice, geological uncertainty is characterised by the Jade ensemble of $N = 50$ geological realisations. A major difference versus the setup of TNO OLYMPUS Challenge is that the Jade model has been history matched to simulated historical data to produce a posterior ensemble of 50 realisations. These are labelled as Jade $1, \ldots,$ Jade 50. Another major difference is access to this underlying stochastic geology model that will permit the validation of the identified decision strategies for which a further 150 Jade models have been generated; labelled Jade $51, \ldots,$ Jade 200.

The Jade model setup used in this application is the same for each ensemble member and is illustrated in Figure 5.1. There are four producer wells labelled A10, B4, C5 and C6, and two injector wells named A16 and C7, which are distinguished on the map by green and blue points respectively. The Jade model also contains numerous full and partial faults with varying levels of fault permeability ensuring that this is a representative problem of that faced for a real world oil field. Note that the location of one of these faults down the centre of the field effectively splits the reservoir into two separate regions, each containing an injector and two producer wells. Control intervals for the decision parameters are formed by amalgamating multiple consecutive discounting intervals. For wave 1 of our application we choose to use six control intervals of lengths 1, 1, 2, 2, 2 and 2 years corresponding to decisions starting on January 1, 2022, 2023, 2024, 2026, 2028 and 2030 respectively resulting in a total of 36 decision parameters across the four wells. Note that decisions are permitted at one year intervals, however constructing control intervals which combine several discounting intervals has the advantage of reducing the dimensionality of the decision support problem. The longer wave 1 intervals may be subsequently split into the original shorter control intervals at a later wave if it is deemed necessary. In addition, target production and injection rates are used as decision parameters with daily operational rate constraints detailed in Table 5.1.

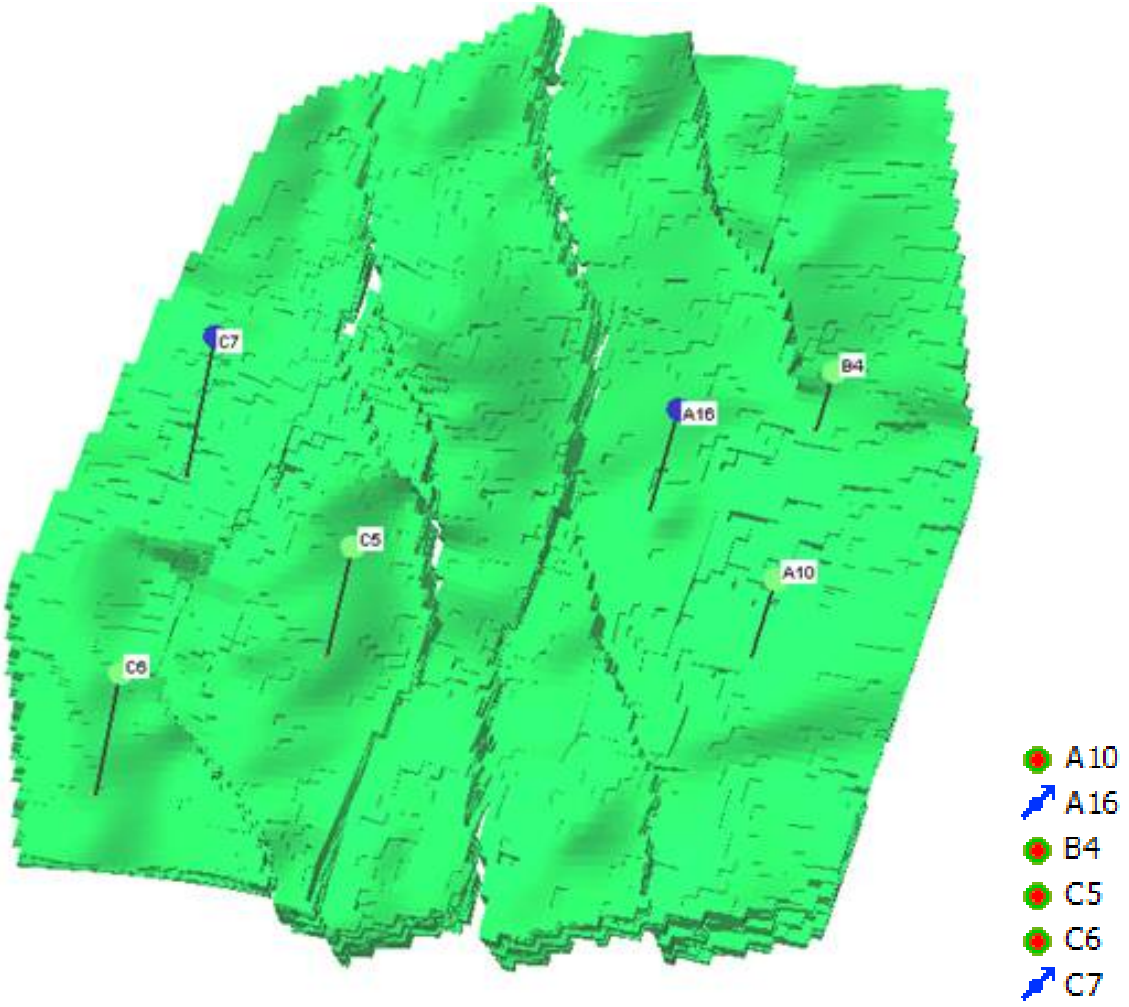


Figure 5.1: Emerson Jade model map with key of decision parameter well IDs and types. The green points represent the four producer wells with IDs A10, B4, C5 and C6. The blue points represent the two injector wells with IDs A16 and C7.

Type	Minimum	Maximum	Unit
Well Oil Production Rate	0	10,000	m^3/day
Well Water Injection Rate	0	15,000	m^3/day

Table 5.1: Emerson Jade model application operational rate constraints.

Throughout this application the same naming conventions and notation for decision parameters and various Jade computer model output will be used as in the application to the TNO OLYMPUS Well Control Optimisation Challenge in Chapter 4. The only modification is that a well ID instead of a well number will be used, for example: “prod_x_yyyy_mm” and “inj_x_yyyy_mm” are used to denote the tar-

get production and injection rate respectively for well ID "x" starting on the 1st of month "mm" in year "yyyy".

In our analysis, all simulations of the Jade ensemble were performed using the Emerson Tempest ENABLE software [44]. On average, each simulation takes approximately 30 minutes to complete using one cluster node, hence for a single decision parameter vector simulated for all 50 Jade models takes approximately 25 hours.

5.1.2 Emerson Jade Model Exploratory Analysis

An exploratory analysis based on a smaller preliminary wave 0 batch of simulations is conducted prior to decision support. The aim is to identify any interesting features of the model which may aid the construction of informative designs as well as to facilitate the establishment of a representative subset of the ensemble allowing for greater exploration of the decision parameter space and more efficient use of the available computing resources. Exploratory simulations are performed for the full 10 year Jade field lifetime. A design for this preliminary batch of simulations is constructed for all 36 decision parameters specified in Section 5.1.1 using maximin Latin hypercube sampling described in Section 2.3 to generate 50 decision parameter vectors from the hypercube volume defined by the operational range constraints. This is a larger than the exploratory design used in the TNO OLYMPUS Well Control Optimisation Challenge application due to access to additional computing resources and technical support for evaluating them. Furthermore, the constructed 50-point maximin Latin hypercube design was formulated such that it contains a smaller 25-point maximin Latin hypercube design. This enables the later testing on whether similar decision support results can be achieved using fewer exploratory simulations when fitting the EGES linear model in Section 5.2 and thus induces additional uncertainty. The simulation output for each decision parameter vector is labelled as SIM_1, . . . , SIM_50.

The spread of Jade exploratory analysis NPV over the 50 Jade model realisations for each of the exploratory simulations are illustrated by the boxplots in Figure 5.2.

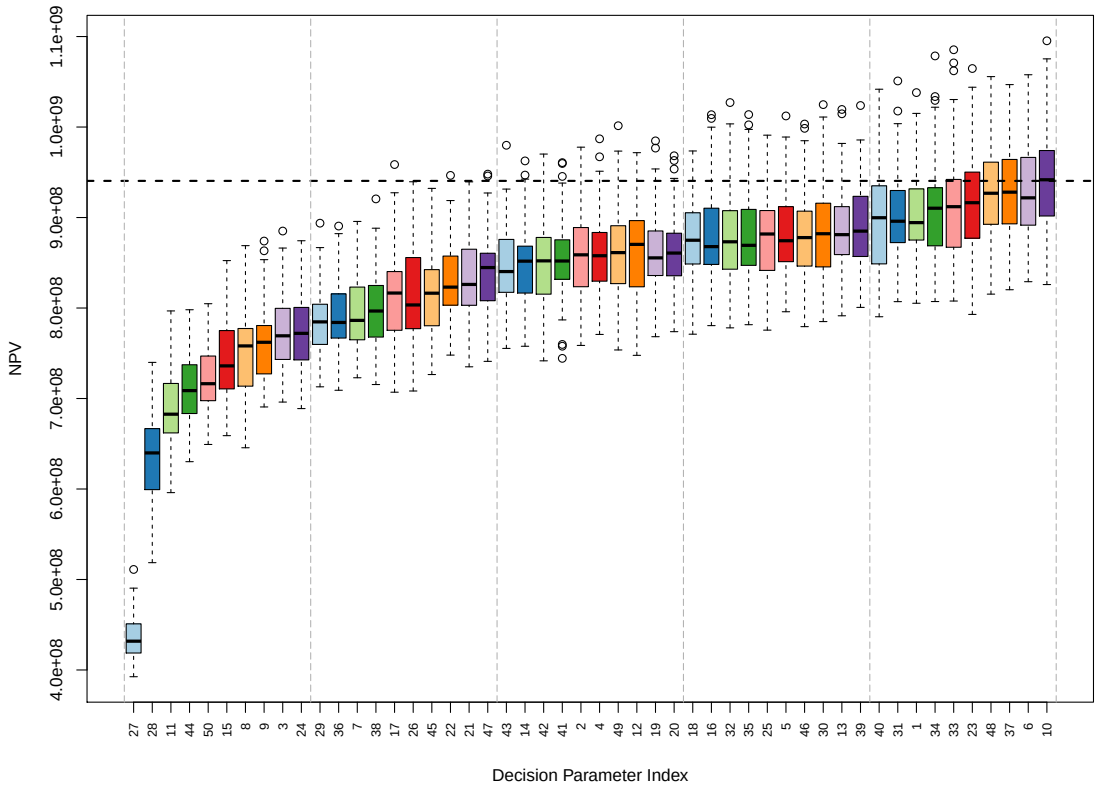


Figure 5.2: Boxplots of the Jade exploratory analysis simulated NPV illustrating the spread over the 50 Jade model realisations. These are ordered from smallest to largest ensemble mean NPV, where the horizontal black dashed line represents the maximum simulated ensemble mean NPV from the exploratory simulations.

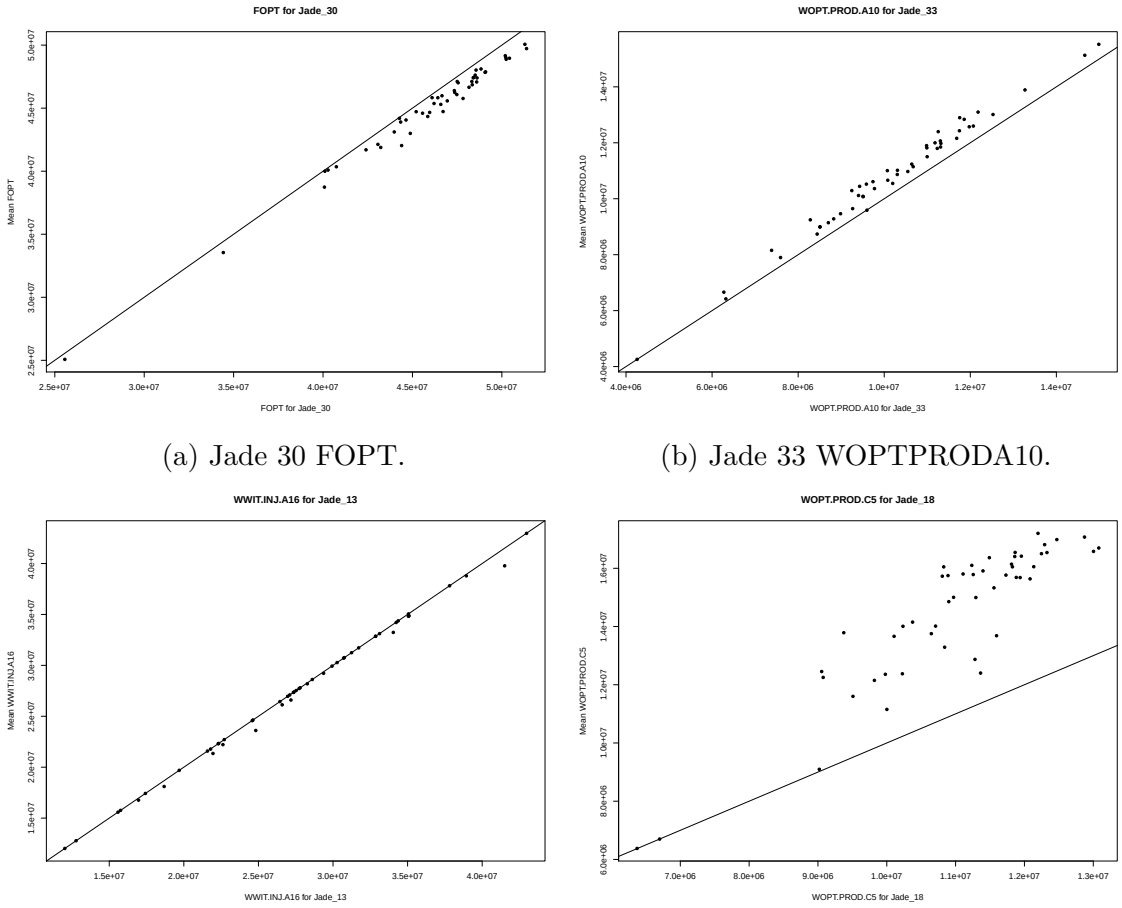
These are ordered from the smallest to the largest ensemble mean NPV, and the horizontal black dashed line represents the maximum simulated ensemble mean NPV from the exploratory simulations. It is observed that there is a large spectrum of NPVs for the different decision parameter vectors, but there are also numerous cases which yield similar ensemble mean NPVs with largely overlapping boxplots, therefore exemplifying their similarity when compared with respect to the uncertainty captured by the geological ensemble. Moreover, 33 of the 50 decision parameters vectors possess a maximum NPV over the ensemble which exceeds the largest ensemble mean NPV. This highlights how a large range of decisions could potentially lead to a good expected NPV on actual implementation once all uncertainties have

been realised. A decision support approach is thus justified since it permits such uncertainties to be resolved at a later time.

5.2 Application of the EGES Techniques

In this section we apply the efficient geological ensemble subsampling (EGES) techniques described in Section 3.2 utilising the exploratory simulations discussed in Section 5.1.2 to identify a representative subset of the $N = 50$ Jade models. This is a novel application of such ensemble subsampling techniques within the petroleum industry.

Preliminary graphical investigations are performed using plots of the ensemble mean versus the individual model outputs for which a selection are shown in Figure 5.3, where the black line denotes equality between the ensemble mean and individual model output. The focus is on the ensemble mean NPV and any simulator outputs which are inputs to the NPV formula including: oil production, water production and injection total, both for the field and individually by well. It is desirable that the chosen Jade models possess a relationship with the ensemble mean output (or NPV) that is easy to model, has minimal uncertainty, and with a preference for linear associations. Note that this is a preliminary graphical assessment which is limited to identifying one-dimensional relationships. A strong linear relationship between the individual model output and the ensemble mean with limited variation is observed in Figures 5.3a to 5.3c indicating that Jade 13, 30 & 33 are potentially representative for the ensemble mean, at least as individual models. In contrast, Jade 18 does not appear to be representative as seen in Figure 5.3d which depicts a non-linear relationship with large residual variation in the ensemble mean versus the output of Jade 18. Using this graphical assessment as a preliminary screening technique leads to a proposed subset of models for further investigations and is particularly useful for large ensemble. These are Jade 1, 2, 3, 6, 7, 10, 11, 12, 14, 15, 17, 23, 24, 27, 30, 31, 37, 40, 44, 47 & 50. Note that this does not



(a) Jade 30 FOPT.

(b) Jade 33 WOPTPRODA10.

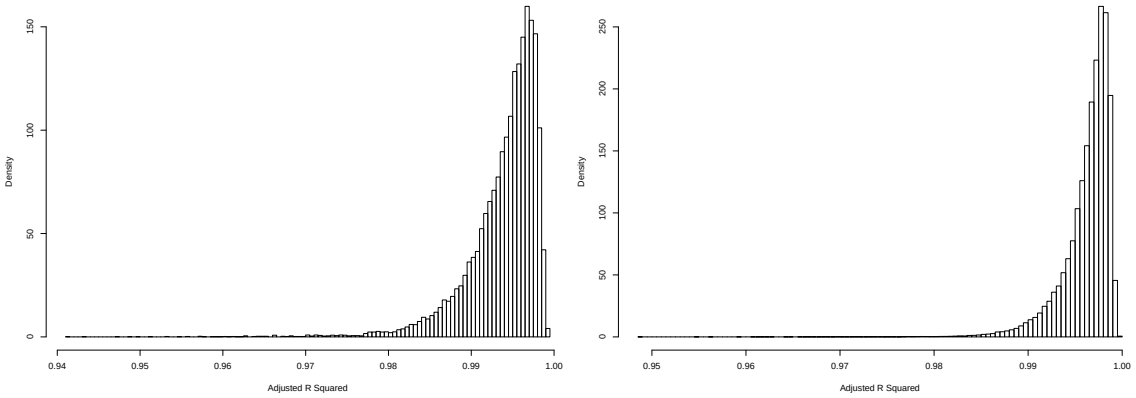
(c) Jade 13 WWITINJA16.

(d) Jade 18 WOPTPRODC5.

Figure 5.3: Preliminary graphical investigations for Efficient Geological Ensemble Subsampling consisting of plots of the Jade ensemble mean versus individual model outputs. The black line denotes equality between the ensemble mean and individual model outputs.

contain Jade 13 or 33 introduced above as good representatives for certain outputs of interest. More detail on their consideration will be presented below.

Linear models are used to encapsulate the interactions between different Jade models in the efficient geological ensemble subsampling technique presented in Section 3.2.2. An initial investigation using the above proposed Jade model subset suggests that either $N_{\text{EGES}} = 3$ or $N_{\text{EGES}} = 4$ is sufficient to accurately capture the relationship between the ensemble mean NPV and the NPV for individual models, with very little improvement in the adjusted R^2 for larger values of N_{EGES}. The search is expanded to consider all Jade models and fit linear models to all subsets of

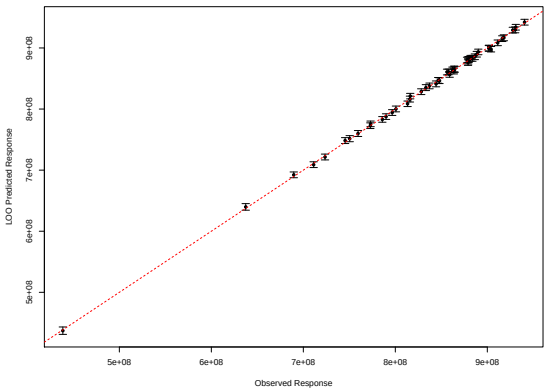


(a) $N_{\text{EGES}} = 3$ (b) $N_{\text{EGES}} = 4$

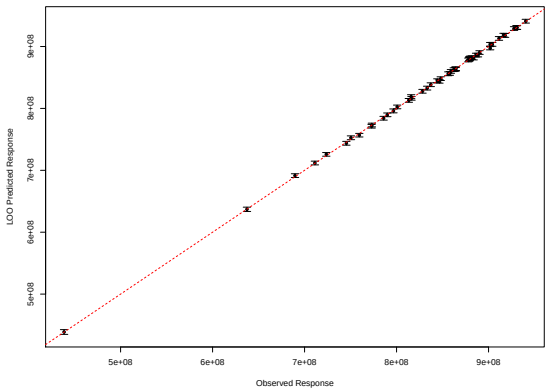
Figure 5.4: Histograms of linear models ensemble mean NPV adjusted R^2 values over all subsets of Jade models of size N_{EGES}.

each size, noting that this is a relatively lengthy but worthwhile calculation in order to obtain a suitable EGES linear model and subset of Jade models. Histograms of the adjusted R^2 values obtained over these linear models for the ensemble mean NPV are shown in Figure 5.4. Both show similarly high adjusted R^2 values and that for most subsets of the respective size, a high value is attained.

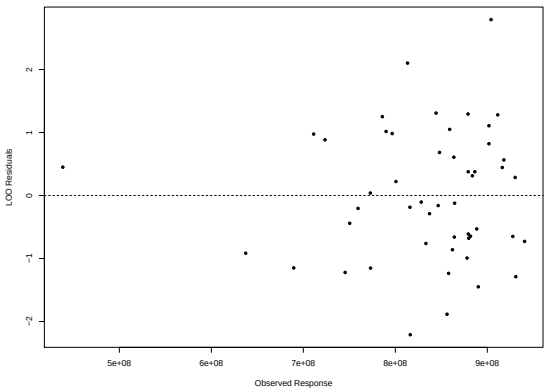
Selecting the candidate EGES linear model with the maximum adjusted R^2 for $N_{\text{EGES}} = 3$ and $N_{\text{EGES}} = 4$; leave-one-out diagnostics are performed with plots shown in Figure 5.5. It is evident in Figures 5.5a and 5.5b that both linear models possess good predictive performance, as indicated by their high adjusted R^2 values, with slightly tighter predictive intervals in Figure 5.5b. Comparison of the standardised residuals in Figures 5.5c and 5.5d highlights that whilst the candidate linear model for $N_{\text{EGES}} = 4$ produces excellent predictions, the uncertainty is potentially too small with some standardised residuals of magnitude greater than 3 reflecting that this linear model may not perform so well for new NPVs for the four Jade models. In contrast, for $N_{\text{EGES}} = 3$ all standardised residuals are less than 3, with most less than 2, hence this is the EGES linear model which will be used in this application containing Jade 13, 30 & 33 as predictors. This comes with the additional benefit of one less Jade model at which to perform simulations, thus reducing the computational expense. It is noted above that both Jade 13 & 33 were not part of the proposed



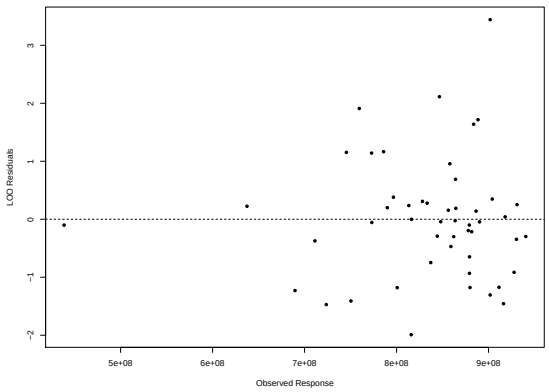
(a) $N_{\text{EGES}} = 3$ linear model with three standard error predictive interval versus simulated ensemble mean NPV.



(b) $N_{\text{EGES}} = 4$ linear model with three standard error predictive interval versus simulated ensemble mean NPV.



(c) $N_{\text{EGES}} = 3$ linear model standardised predictive residuals versus simulated ensemble mean NPV.



(d) $N_{\text{EGES}} = 4$ linear model standardised predictive residuals versus simulated ensemble mean NPV.

Figure 5.5: Leave-one-out diagnostics plots for the candidate ensemble mean NPV EGES linear models with $N_{\text{EGES}} = 3$ (left) and $N_{\text{EGES}} = 4$ (right). The top row shows plots of the linear model three standard error predictive intervals versus the simulated ensemble mean NPV. The bottom row shows plots of the linear model standardised predictive residuals versus the simulated ensemble mean NPV.

subset of models since it was judged that for certain outputs these were not as good representatives for the ensemble mean as some of the other suggested models. However, in combination with Jade 30 via the EGES linear models it is found that this subset characterises the ensemble mean NPV best with respect to the adjusted R^2 with $N_{\text{EGES}} = 3$. This serves as a warning for placing too much emphasis on

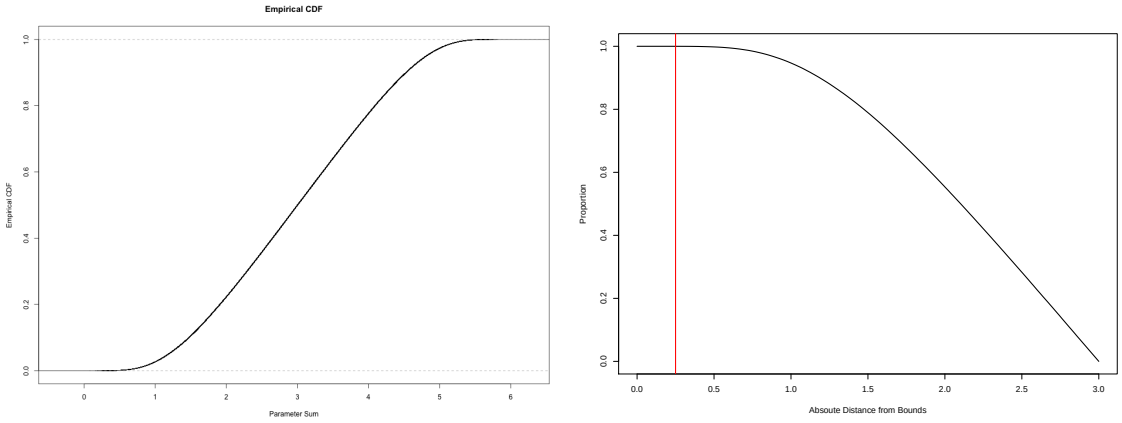
one dimensional plots for a multivariate analysis and demonstrates the value of the EGES linear model construction. The EGES linear model is used as an efficient method of prediction whilst providing a quantification of the induced uncertainty.

5.3 Targeted Bayesian Design Wave 1

The first wave of iterative decision support requires a space filling design in order to construct an accurate emulator representation for the utility function. This follows the approach described in Section 4.3.1 first imposing two types of constraints on the decision parameters with the wave 1 targetted Bayesian design constructed using Algorithm 1. For the wave 1 design we use the setup described in Section 5.1.1 with $D = 36$ decision parameters corresponding to six control intervals starting on January 1, 2022, 2023, 2024, 2026, 2028 & 2030 for each of the six wells.

An initial hypercube decision parameter space is formed by imposing finite ranges for each well control in the form of operational range constraints. These are stated in Table 5.1. A priori reservoir engineering insight suggests that the decision parameters should not be treated as mutually independent, with large temporal variations being unphysical and unlikely to lead to good reservoir production performance. In order to account for these beliefs and restrict our attention to the meaningful part of the initial decision parameter space, we impose difference constraints between time consecutive control parameters where for time ordered decision parameters $|d_i - d_{i-1}| \leq \Delta_i$ for $i = 2, \ldots, D'$ resulting in a non-hypercube decision parameter space. A conservative choice for the maximum permitted change per year is $\Delta = \frac{1}{6}$ of the parameter range with the design divided into six independent subgroups of $D' = 6$ decision parameters by well. This reduces the volume of the decision space to approximately 1.8895% of the initial hypercube volume due to the range constraints only with the auxiliary benefit of reducing the computational expense.

For each of the six subgroups of six decision parameters a wave 1 targeted Bayesian design of $n = 700$ points is constructed using Algorithm 1. These are



(a) Empirical Cumulative Distribution Function (ECDF) for the normalised parameter sum for an individual well based on uniform random sampling.

(b) Proportion of constrained wave 1 decision parameter space contained within the truncation bounds defined by an inset. The vertical red line corresponds to the chosen case of 0.25.

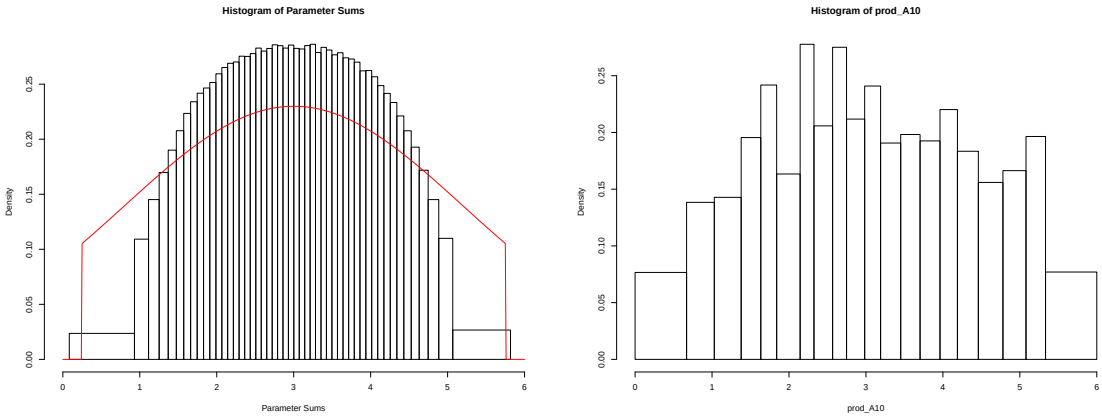
Figure 5.6: Visualisation tools in selecting the truncation bounds for sampling the normalised sum of parameters from a truncated normal distribution within the construction of the wave 1 targetted Bayesian design via Algorithm 1 for the Emerson Jade model.

approximately optimised with respect to the minimax design selection criterion (see Equation (2.3.4)) compared to a large $20,000$ point uniform random reference sample with which to compare a large collection of candidate designs. Note that a 700 point design is justified by the heuristic of sampling such that $n \geq 10 \cdot D$ [116]. Within Algorithm 1 the normalised sums of parameters are sampled from a truncated normal distribution with mean $\mu_t = 3$, standard deviation $\sigma_t = 2.2$, and truncation bounds of $[0.25, 5.75]$, whilst the maximal range is $[0, 6]$. This restriction is validated since 99.9960% of the normalised parameter sums for uniform samples over the decision space lie within this interval. Visualisation tools shown in Figure 5.6 are useful for selecting an appropriate truncation interval. The empirical cumulative distribution function (ECDF) for the normalised sum of parameters for an individual well is shown in Figure 5.6a highlighting very little tail probability mass, whilst the proportion of the decision parameter space between two bounds of equal absolute inset is shown in Figure 5.6b where the chosen case of 0.25 is depicted by the red line.

The truncated normal distribution mean hyperparameter is specified to be $\mu_t = 3$

corresponding to the mid-point of both the range of the normalised sum and the truncation interval. This matches the expectation under uniform sampling. The standard deviation hyperparameter is selected in order to achieve a trade-off between the results of uniform sampling where the majority of the probability mass is centred close to the sum of parameters mean, and increased exploration of the two extreme corners that give rise to smaller and larger normalised sums of parameters. This assessment is conducted through the use of histograms to compare a proposed truncated normal density with the results of uniform sampling. An example is shown in Figure 5.7a for the selected value of $\sigma_t = 2.2$, corresponding to 0.367 per parameter. Note that it is not expected that results of targetted sampling from $f_T(t)$ and from the uniform distribution align with the choice of $f_T(t)$ dependent on the aims of the analysis and any prior beliefs. The corresponding samples for producer well A10 are shown in Figure 5.7b. The benefit of targetted sampling for decision support is that it enables regions of low utility function values to be quickly eliminated, whilst providing greater coverage of regions where reservoir engineering insight suggests high NPVs are likely to occur.

Figure 5.8 is a pairs plot of the 2-dimensional projection of the wave 1 design for producer well B4. The clustering of points between two diagonal parallel bands for the plots next to the axis labels illustrates the effect of the difference constraint, with longer control intervals resulting in wider bands. This effect is also evident at larger time separations with fewer points away from the main diagonal, although it is less noticeable as the time lag increases. As expected given the above comments, there are more points sampled for smaller and larger parameter values compared to uniform sampling achieving better exploration of the corner and edge regions of the decision space whilst maintaining sufficient coverage of the central region with a view to constructing Bayesian emulators. Moreover, the points are coloured by the simulated expected NPV predicted from the EGES linear model where green, yellow and red points correspond to high, moderate and low NPVs respectively. Observations confirm reservoir engineering insight with high early time target production rates are



(a) Histogram of the sum of normalised parameters obtained via uniform sampling over the constrained decision parameter space for an individual well with the truncated normal density overlaid in red.

(b) Histogram of sum of normalised parameters for producer well A10 sampled from a truncated normal distribution.

Figure 5.7: Histograms of wave 1 design sum of normalised parameters. This includes a comparison of the chosen truncated normal distribution with truncation bounds $[0.25, 5.75]$, mean $\mu_t = 3$, and standard deviation $\sigma_t = 2.2$, this is $f_T(t)$ in Algorithm 1, with the results of uniform sampling over the constrained decision parameter space. The nature of employing a targetted design means that these are not expected to align.

required to achieve a high NPV whilst at later times the effect of the high discount factor results in greater spread of red and yellow points, although small values still generally correspond to a low expected NPV. Similar patterns are observed for the other producer wells, whilst for the injector wells, the association is reversed but less pronounced. These plots are omitted for space. The sampling marginal density of the individual parameters are illustrated by the histograms in Figure 5.9. Both appear to have a slightly higher density in the central region, but are still relatively close to uniform. Note that Algorithm 1 is not designed to produce uniform marginal samples, but does yield a close approximation for this choice of $f_T(t)$.

The six independent sub-designs of size 700 and each of dimension 6 are combined using random permutations and approximately optimised with respect to the minimax design selection criterion. Two further points of interest not explored by the constructed design are when all parameters are set to either their minimum

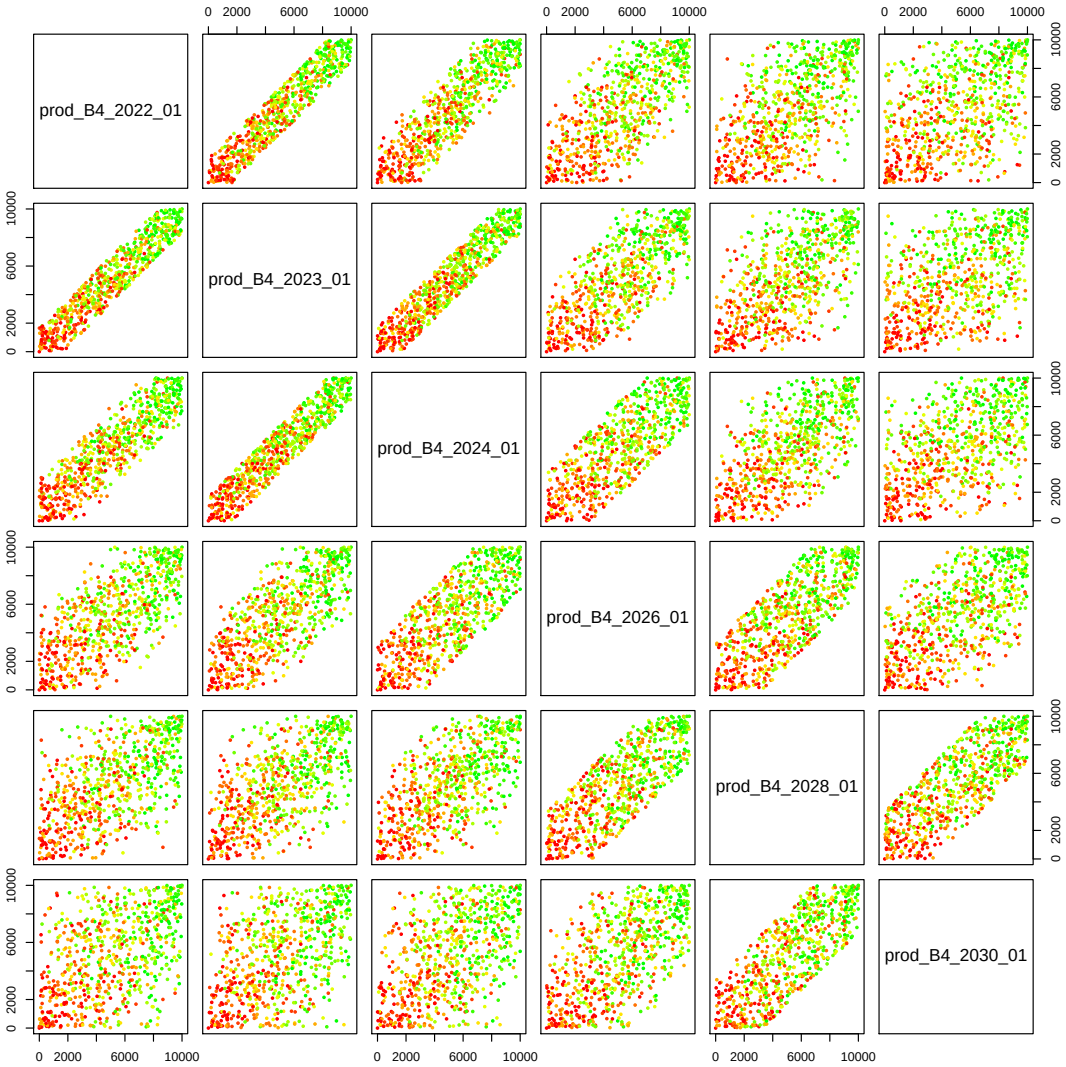
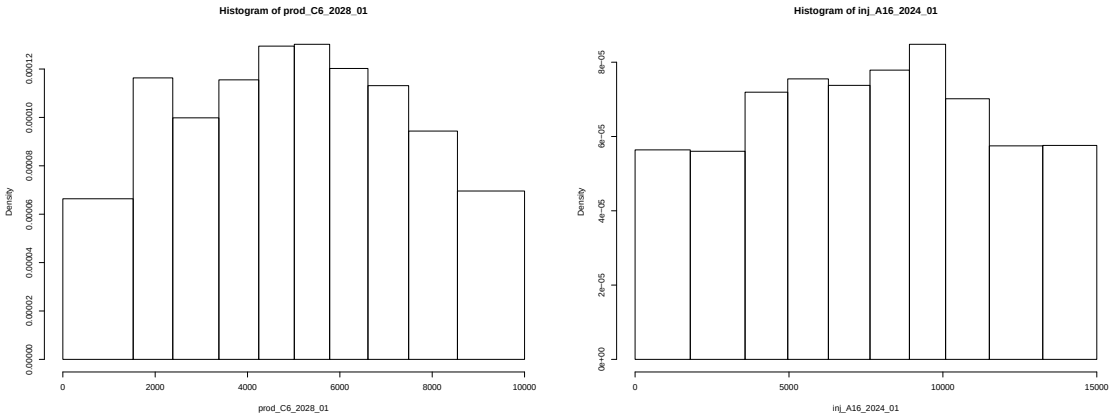


Figure 5.8: Wave 1 design pairs plots of the 702 well control strategies for producer well B4 for a design generated using Algorithm 1 and selected using the minimax distance criterion. A fixed initial sample of parameter sums from a truncated normal distribution was used when forming all candidate designs. Points are coloured according to the simulated NPV obtained from the EGES linear model with green, yellow and red corresponding to high, moderate and low NPVs respectively.

or maximum values which lie within the two extreme corner regions removed during sampling from the truncated normal distribution. The final selected design is augmented to include these two decision parameter vectors.



(a) prod_C6_2028_01 (b) inj_A16_2024_01

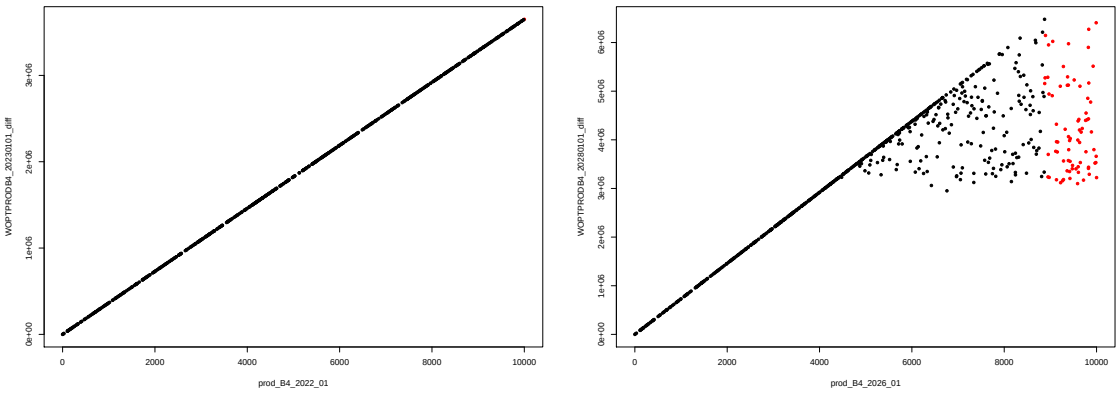
Figure 5.9: Jade wave 1 design marginal density histograms for prod_C6_2028_01 and inj_A16_2024_01 over the 702 well control strategies generated using Algorithm 1 and selected using the minimax distance criterion.

5.4 Hierarchical Emulation Exploiting Known Simulator Behaviour Wave 1

In this section we fit a hierarchical emulator exploiting known simulator behaviour as described in Section 3.3 to the Jade model utility function in order to explore the 36-dimensional decision space. The EGES linear model constructed in Section 5.2 is used to predict the expected NPV with the emulator fitted using the wave 1 design constructed in Section 5.3. We begin by decomposing the NPV formula for each model into its constituents and fit emulators which incorporate their structure in Section 5.4.1 which are combined to obtain the approximate NPV for an individual model in Section 5.4.2. This is linked to the exact NPV in Section 5.4.3. The final step is to emulate the utility function in Section 5.4.4.

5.4.1 Structured Emulators Exploiting Known Simulator Behaviour for NPV Constituents

The expected NPV utility function can be decomposed into contributions from oil production, water production and injection by well and control interval, each of



(a) WOPTPRODB4_2023_01 versus prod_B4_2022_01.

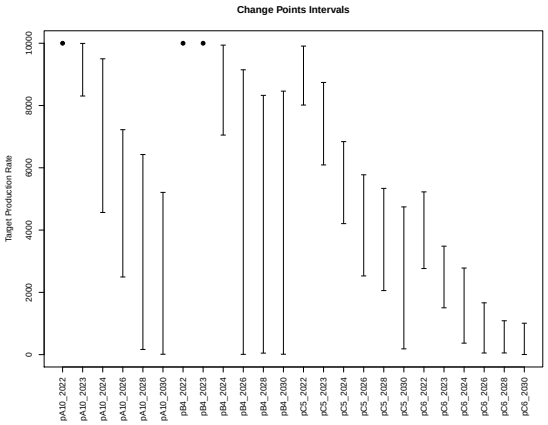
(b) WOPTPRODB4_2028_01 versus prod_B4_2026_01.

Figure 5.10: Jade 30 wave 1 WOPT for producer well B4 during the intervals January 1, 2022 – 2023 and 2026 – 2028 versus prod_B4_2022_01 and prod_2026_01 respectively. The red points correspond to where $d_{PB4,2022} > c^u_{PB4,2022}$ and $d_{PB4,2026} > c^u_{PB4,2026}$ respectively. In the left plot all values of prod_B4_2022_01 are achieved for the full control interval. In the right plot prod_B4_2026_01 is achieved for small values yielding perfectly defined linear behaviour up to an unknown change point beyond which there is plateau region due to BHP constraints.

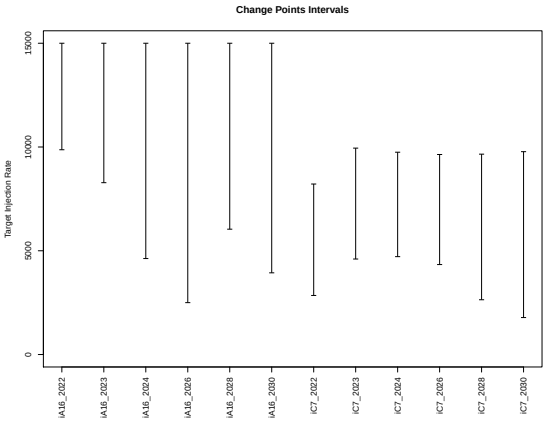
which can be extracted as simulation output from the Jade model. As was the case for the OLYMPUS model, WOPT and WWIT within a control interval adhere to a known and constrained structural form of behaviour where the quantity is equal to the corresponding target rate multiplied by the length of the control interval up to an unknown change point, beyond which there is a plateau due to BHP constraints. Two variants are shown for Jade 30 in Figure 5.10 for producer well B4 in the control intervals where the red points indicate those which exceed the estimate of the change point upper bound for the respective input. This is calculated using Equation (3.3.1) with $\delta_{i,u} = 10$. For some wells during the first few control intervals where all values of a target rate are achieved for the full time period, as seen in Figure 5.10a. This structure should be exploited to obtain extremely accurate emulators. The slope-plateau structure is illustrated for a later control interval in Figure 5.10b. Note that structured behaviour is not observed for WWPT within a control interval.

Structured emulation is applied separately for each Jade model with estimates of the change point upper bounds using Equation (3.3.1) with $\delta_{i,u} = 10$ and the wave 1 simulations for each WOPT and WWIT by well and control interval. An example of the result is displayed in Figure 5.10b. The extrapolation cut-offs are estimated as the change point lower bounds from the wave 1 simulations using Equation (3.3.2) with $\delta_{i,l} = 10$ to account for numerical precision within the simulations. These are illustrated for each of the wave 1 sub-selected Jade models in Figure 5.11 highlighting the region in which the "true" change point is believed to be situated. Compared with the TNO OLYMPUS Well Control Optimisation Challenge, there exist numerous constituents for which the estimate of the change point upper bound and extrapolation cut-off coincide, as is the case for Jade 30 WOPTPRODB4_2023_01 shown in Figure 5.10a. In addition, it is established that there are many cases where the change point upper bound estimate is close to the maximum of the operational range and hence there are relatively few design points for which $d_{jk,t_i} > c^u_{jk,t_i}$. These observations have implications for the preliminary Bayes linear emulator.

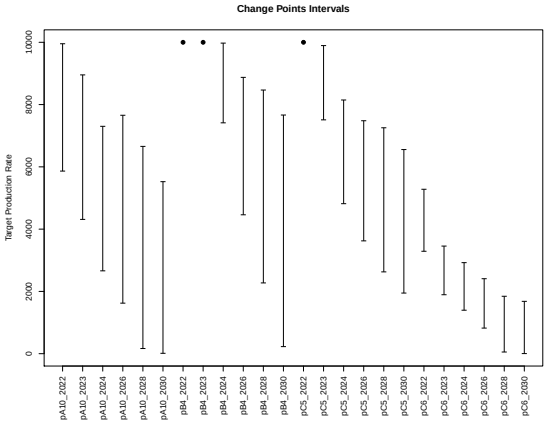
A modified setup compared to in the TNO OLYMPUS Well Control Optimisation Challenge is used to perform preliminary Bayes linear emulation for each WOPT and WWIT constituent, represented by $f_i(\mathbf{d})$, and still assuming the structure shown in Equation (2.4.1). Firstly, for constituents where $c^u_{jk,t_i} = b^{cp^l}_{jk,t_i}$ corresponding to belief that all points lie on the slope, we impose that the emulator should collapse onto the slope for all new decision parameter vectors, thus skipping the preliminary Bayes linear emulation step and jumping straight to the classification. It is acknowledged that simulations have only been performed for a sample of possible scenarios, hence there is the potential that a new simulation may yield output which is not on the slope. This is deemed unlikely given the relatively large size of the wave 1 design. In all other situations a preliminary emulator is fitted, however the form of the deterministic functions, $m_i(\mathbf{d}_{A_i})$, depends on the number of points in $\mathcal{D}'_i = \{\mathbf{d} \mid \mathbf{d} \in \mathcal{D},\ d_{jk,t_i} > c^u_{jk,t_i}\}$. Let A_i be the set of active decision parameters for the i^{th} constituent which is assumed to consist of all decisions which take place in the past



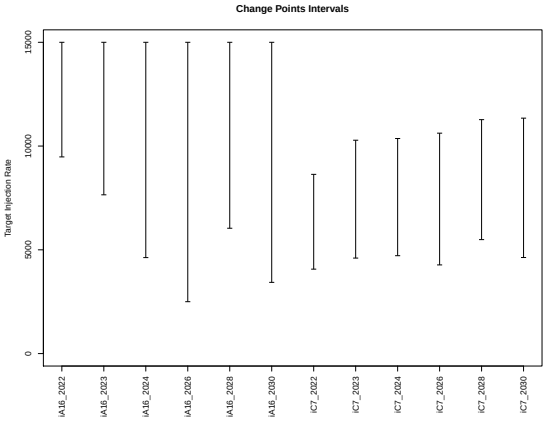
(a) Jade 13 WOPT



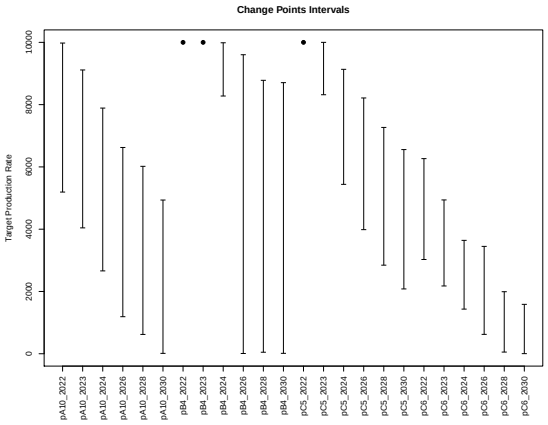
(b) Jade 13 WWIT



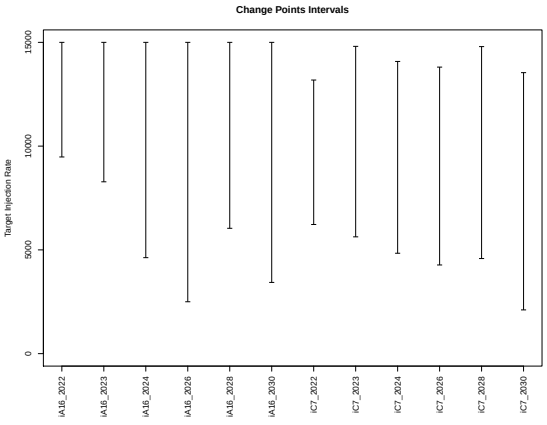
(c) Jade 30 WOPT



(d) Jade 30 WWIT



(e) Jade 33 WOPT



(f) Jade 33 WWIT

Figure 5.11: Jade wave 1 change point upper bound and extrapolation cut-off intervals for WOPT and WWIT within each control interval with respect to their corresponding decision parameter for each of the three subsampled Jade models. Black points correspond to where the change point upper bound and extrapolation cut-off coincide for a particular NPV constituent and occurs for the maximum value of the target rate.

of the output, and $|A_i|$ denote its cardinality. If $|\mathcal{D}_i'| \leq \max\{40, 1 + 2|A_i|\}$, then the deterministic function is specified to be a constant mean in order to reduce the number of emulator hyperparameters to fit. Otherwise the deterministic functions are of the following form:

$$m_i(\mathbf{d}_{A_i}) = \mathbf{g}_i(\mathbf{d}_{A_i})^{\mathrm{T}}\boldsymbol{\beta}_i = \beta_{i,0} + \sum_{d_j \in A_i} \{\beta_{i,j,1} d_j + \beta_{i,j,2} d_j^2\} \tag{5.4.1}$$

The remainder of the emulator structure consisting of the weakly stationary stochastic process and nugget terms remain unaltered. This construction ensures that the fitting process can be automated. Note that the limited number of fitting points issue for particular constituents was not observed for the OLYMPUS model and hence this approach was not necessary in that application.

The rest of each preliminary emulator's prior specification is obtained via linear models fitted only using simulation points in $\mathcal{D}_i'$ with output $\mathbf{F}_i' = \{f_i(\mathbf{d}) \mid \mathbf{d} \in \mathcal{D}_i'\}$. The form of the linear model is either a constant or as shown in Equation (5.4.1) given the above considerations with all decision parameters transformed onto $[-1, 1]$. An estimate of the residual standard error is denoted by $\sigma_{i,lm}$. It is assumed that the unknown regression coefficients have prior expectation $\boldsymbol{\mu}_\beta = \mathbf{0}$ and an infinite prior uncertainty, again employing the limiting results in Equations (2.4.44) and (2.4.45) derived in Section 2.4.5. For the residual process it is assumed that $\mathbb{E}[u_i(\mathbf{d}_{A_i})] = 0$ and $\mathbb{E}[w_i(\mathbf{d})] = 0$ with squared exponential covariance structure (Equation (2.4.3)) with a single common correlation length hyperparameter. Following the substitution approach for the hyperparameters: $\sigma_{u_i}^2 = (1-\rho)\sigma_{i,lm}^2$ and $\sigma_{w_i}^2 = \rho\sigma_{i,lm}^2$ where $\rho = 0.05$; whilst the correlation length is set to half of the parameter range, hence $\theta = 1$.

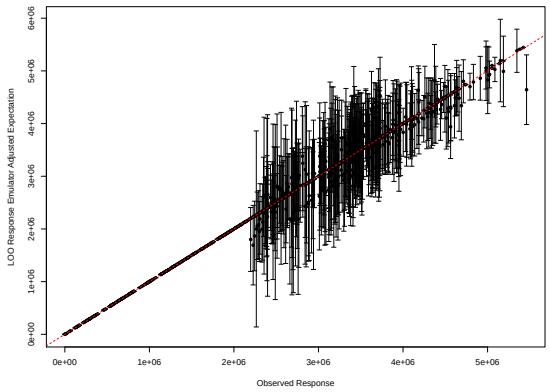
Given this preliminary emulator and a theoretical upper bound based on the target rate for the time interval, the emulation output type is determined with points classified as in the slope, plateau, or close to the change point following the structured emulation with two-sided truncation method presented in Section 3.3.2. The use of the two-sided truncation is necessitated by the limited number of points in $\mathcal{D}_i'$ for some constituents combined with the extrapolation due to higher estimates

of c^u_{jk,t_i}, both of which results in greater additional uncertainty.

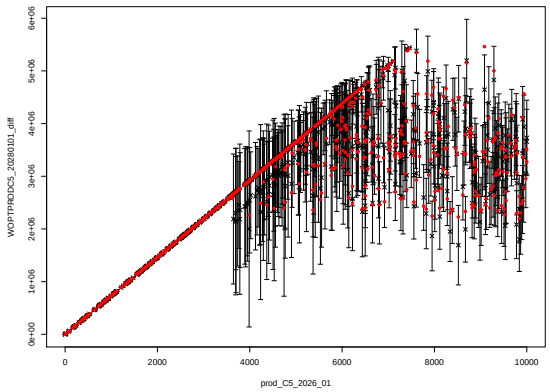
Examples of the leave-one-out diagnostics plots are shown in Figure 5.12 for Jade 30 NPV constituents. Figures 5.12a and 5.12c show the structured emulator adjusted expectation with credible intervals of width 3 adjusted standard deviations versus the corresponding simulated output. These are plotted versus the corresponding target rate in Figures 5.12b and 5.12d respectively. It is observed that the structured emulation approach is exceptionally accurate for smaller response values where the target rate is adhered to for the entire control interval. For larger values believed to be on plateau the credible interval is of a greater width. The two-sided truncated GP emulator is employed for intermediate values in the uncertain region around the change point location. The benefits are best observed in Figure 5.12b for Jade 30 WOPTPRODC5_20280101_01. There exist some cases close to the extrapolation cut-off where the emulator under-predicts, but the credible interval is sufficiently wide to account for this lack of knowledge. This arises as there is a noticeable difference between $c^u_{PC5,2028}$ and $b^{cp^l}_{PC5,2028}$ in Figure 5.11c leading to a relatively large extrapolation. A potential means of addressing this is to impose a smaller credible interval width in the truncation classification step detailed in Section 3.3.2. It is established that for all NPV constituents, the majority of the credible intervals contain the simulated value with no further issues detected in other leave-one-out diagnostic analyses. Highly accurate emulators for the WOPT and WWIT NPV constituents have been constructed based on a relatively small number of simulations by exploiting the known structure of their behaviour.

WWPT within a control interval does not follow the same structured behaviour as WOPT and WWIT and is emulated separately using a Bayes linear emulator. This follows the same construction as the above preliminary Bayes linear emulator with deterministic functions of the form specified in Equation (5.4.1). The rest of the prior specification is obtained via linear models fitted using all simulations in $\mathcal{D}$.

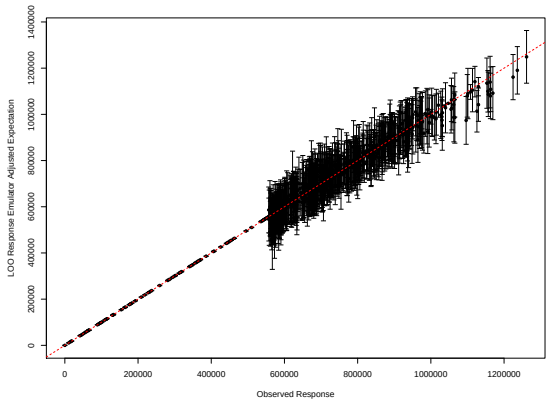
For each Jade model, this structured approach requires the fitting of 60 separate emulators: 36 of the structured type for WOPT and WWIT; and 24 Bayes linear



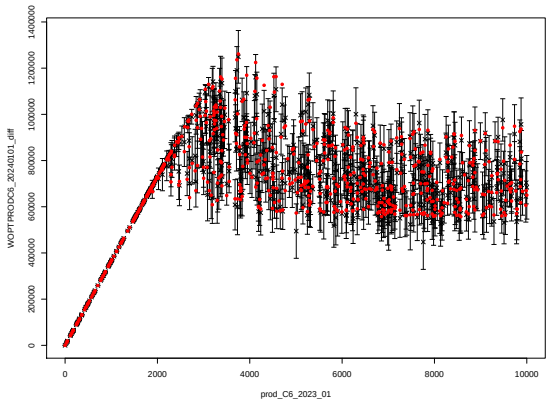
(a) WOPTPRODC5_2028_01 emulator CI versus simulated output.



(b) WOPTPRODC5_2028_01 emulator CI versus prod_C5_2026_01.



(c) WOPTPRODC6_2024_01 emulator CI versus simulated output.



(d) WOPTPRODC6_2024_01 emulator CI versus prod_C6_2023_01.

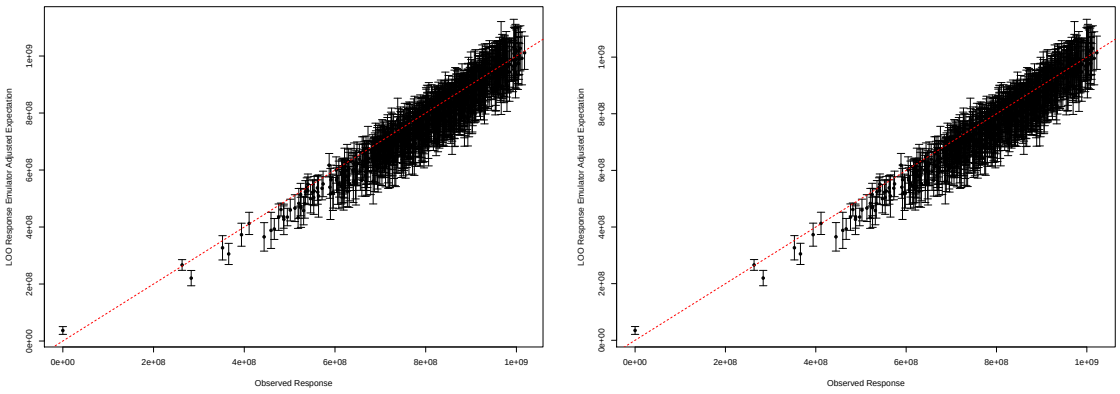
Figure 5.12: Leave-one-out diagnostic plots for the wave 1 structured emulation of Jade 30 WOPTPRODC5_2028_01 (top) and WOPTPRODC6_2024_01 (bottom). The left column shows the adjusted expectation with credible intervals (CI) of width 3 adjusted standard deviations error bars versus the simulated value where the red dashed line denotes equality of the structured emulator and simulator. The right column shows the adjusted expectation with credible intervals of width 3 adjusted standard deviations error bars versus the output's corresponding target production rate with red points denoting the simulated values.

emulators for WWPT, and a total of 180 emulators over the three sub-selected Jade models. The next stage of this analysis is to combine these constituent structured and Bayes linear emulators to first emulate the NPV for each model, before subsequently emulating the ensemble mean NPV and linking this to the expected NPV.

5.4.2 Emulation of the Approximate NPV

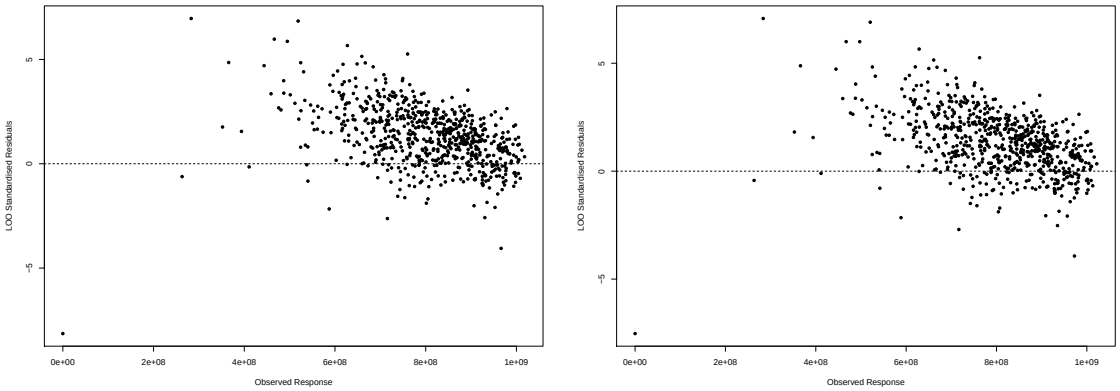
The six control intervals used in this application are formed by merging multiple of the consecutive discounting intervals. An emulator for the average discounting approximate NPV computed via Equations (3.3.5) and (3.3.6) is constructed following the methodology in Section 3.3.3 using the adjusted expectation and variance formulae in Equations (3.3.7) and (3.3.8) respectively. These are evaluated with respect to the suggested fixed NPV cost parameters in Table 5.2 on page 249 with the NPV constituents regarded as independent. As for the TNO OLYMPUS Well Control Optimisation Challenge, when a structured emulator is collapsed onto its slope, the fixed maximum absolute error is treated as equivalent to a 3-sigma credible interval. Multivariate emulators for the NPV constituents are not employed for the same reasons as outlined in Section 4.5.2.

Leave-one-out diagnostics for the approximate NPV emulator for Jade 30 are shown in Figures 5.13a and 5.13c. It is observed that the majority of the credible intervals contain the simulated value of the approximate NPV, as illustrated by the red dashed line which passes through most of the intervals in Figure 5.13a. An analysis of the standardised residuals in Figure 5.13c highlights that there is a slight tendency for under-prediction for smaller approximate NPVs shown by the positive residuals and decreasing pattern. This is explained by the greater uncertainty within this intermediate region for certain NPV constituents arising from the previously highlighted issues regarding change point and extrapolation cut-off estimation. Classification for the preliminary Bayes linear emulator most often leads to two-sided truncation resulting in adjusted expectations pulled below the slope values. Where the simulated output also lies on the slope, it is shown that the credible interval is sufficiently wide leading to few or no violations of emulator diagnostics. Another observation for the magnitude of the standardised residuals is that whilst many are less than three, there exist larger cases. These tend to occur for smaller approximate NPVs; a priori these are unlikely to correspond to well control strategies that will yield near optimal expected NPV. Consequently this is



(a) Average discounting approximate NPV credible intervals versus simulated values.

(b) Exact NPV credible intervals versus simulated values.



(c) Average discounting approximate NPV standardised residuals versus simulated values.

(d) Exact NPV standardised residuals versus simulated values.

Figure 5.13: Leave-one-out diagnostics for wave 1 emulators of the Jade 30 average discounting approximate NPV (left column) and the exact NPV (right column) via a simple linear model on the emulated approximate NPV. The top row shows the emulator adjusted expectation with 3 adjusted standard deviation credible intervals versus simulated values where the red dashed line denotes when the emulator and simulator coincide. The bottom row shows the emulator standardised residuals versus the simulated values.

not an issue with a view to decision support, particularly given that this is wave 1 of the analysis. In addition, where multiple (and in some cases the majority) of the NPV constituents have their emulators collapsed onto the slope region yields tighter credible intervals for the approximate NPV. This highlights the benefits of the hierarchical construction.

5.4.3 Linking the Exact and Approximate NPV

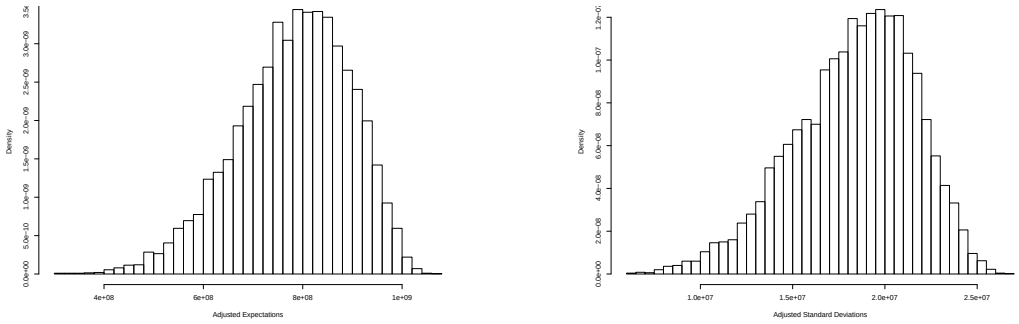
An emulator for exact NPV, as defined in Section 3.3.3, is obtained following the simple linear regression framework described in Section 3.3.4 on the approximate NPV which accounts for the uncertainty induced by amalgamating discounting intervals. The adjusted expectation and variance are computed using Equations (3.3.12) and (3.3.13) respectively where for each Jade model, $\hat{\beta}_{0,\widetilde{\text{NPV}}_j}, \hat{\beta}_{1,\widetilde{\text{NPV}}_j}$ and $\sigma_{\widetilde{\text{NPV}}_j}$ are estimated using the wave 1 simulation data.

Leave-one-out diagnostics plots for the Jade 30 NPV are shown in Figures 5.13b and 5.13d with similar results and appraisal to those for the average discounting approximate NPV in Figures 5.13a and 5.13c. This shows that very little additional uncertainty is introduced by the link model.

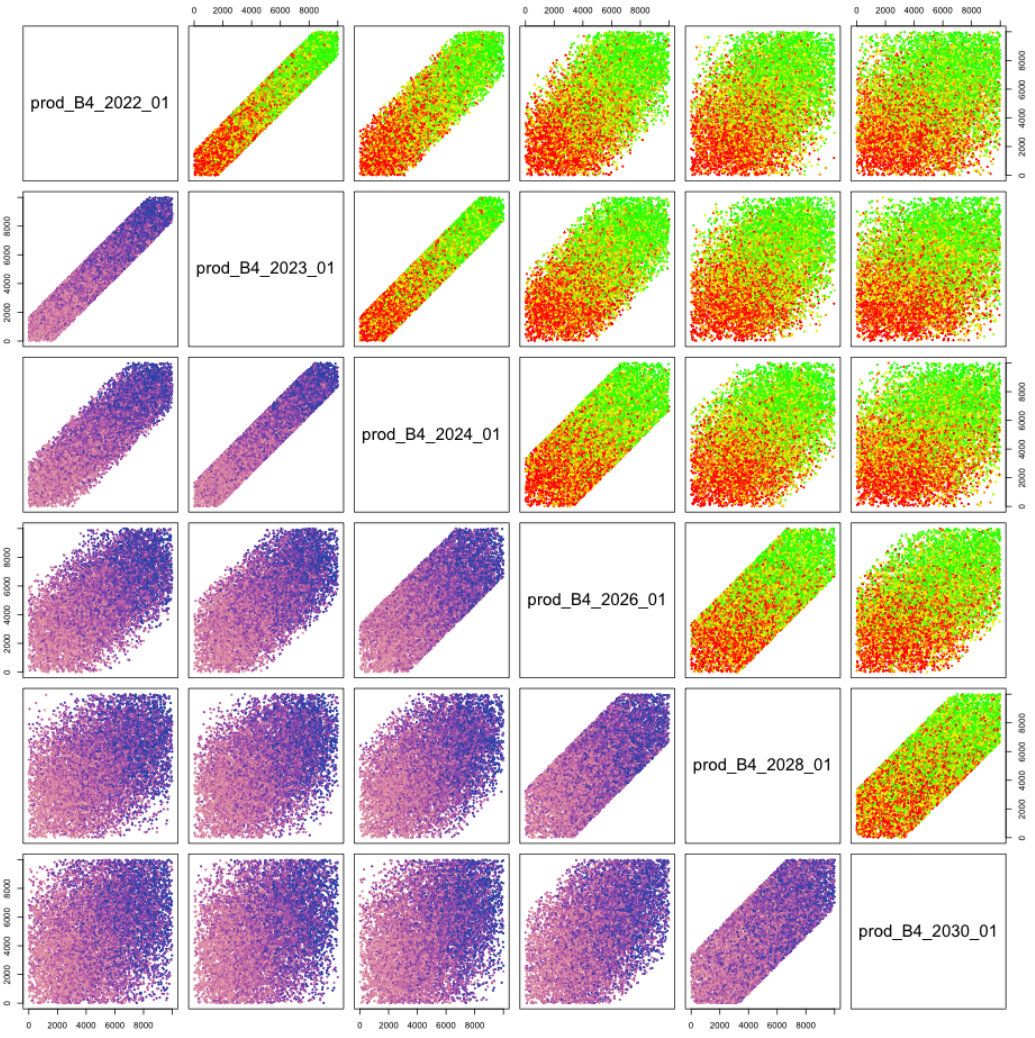
5.4.4 Emulation of the Ensemble Mean NPV

An emulator for the Jade ensemble mean NPV is constructed using the EGES linear model formulated in Section 5.2 to combine the emulators for the Jade 13, 30 & 33 NPV. This follows the methodology described in Section 3.3.5 with the adjusted expectation and variance computed using the formulae in Equations (3.3.16) and (3.3.17) respectively. This accounts for the covariance structure when estimating the regression coefficients.

True leave-one-out diagnostics cannot be performed as simulations have only been evaluated for a subset of three of the Jade models. It is believed that an accurate emulator with a reasonable magnitude of uncertainty has been constructed since the leave-one-out diagnostics for each sub-selected Jade model NPV emulator and EGES linear model diagnostics support a well fitted hierarchical emulator. The adjusted expectations and standard deviations evaluated over the space are illustrated by the histograms in Figures 5.14a and 5.14b respectively from which it is evident that the standard deviations are at least an order of magnitude smaller than the expectations, with a reasonable spread of both emulator outputs.



(a) Histogram of the adjusted expectations. (b) Histogram of the adjusted standard deviations.



(c) Hierarchical emulator for the Jade ensemble mean NPV pairs plot for producer well B4. The plots above the diagonal are coloured by the adjusted expectation with green, yellow and red corresponding to high, moderate and low values respectively. The plots below the diagonal are coloured by the adjusted standard deviation with light pink, purple and blue corresponding to low, moderate and high values respectively.

Figure 5.14: Jade ensemble mean NPV wave 1 hierarchical emulator output plots.

Figure 5.14c shows a pairs plot for the decision parameters associated with producer well B4. Above diagonal plots are coloured by the emulator adjusted expectation where green, yellow and red correspond to high, moderate and low values respectively. This is in agreement with Figure 5.8 with high adjusted expectations for the ensemble mean NPV corresponding to larger decision parameter values, as advocated by reservoir engineers. The below diagonal plots are coloured by the emulator adjusted standard deviation where light pink, purple and blue correspond to low, moderate and high values respectively. For each pair of decision parameters, the observed pattern is very similar to that in the corresponding above diagonal plot suggesting that the largest uncertainty occurs for the highest ensemble mean NPVs.

The fitting and evaluation of a hierarchical emulator for the ensemble mean NPV delivers noticeable computational performance gains versus the direct simulation from the Jade model achieving approximately 4 new decision parameter vector emulator evaluations per second using a single core, compared to the 25 hours for the Jade model. This result is comparable to that for the TNO OLYMPUS Well Control Optimisation Challenge, despite the construction of additional structured emulators for the NPV constituents from 144 to 180.

5.5 Wave 1 Uncertainty Analysis for Well Control Optimisation

A comprehensive uncertainty quantification linking the computer model and the corresponding real world physical system forms a vital part of any decision support analysis, as illustrated for the TNO OLYMPUS Well Control Optimisation Challenge in Section 4.7.3. For this application we assume the computer model represents an actual Jade oil reservoir and that the utility function is $U(\mathbf{d}) = \mathbb{E}[\text{NPV}](\mathbf{d})$. Following the techniques discussed in Sections 2.7 and 3.5 we implement a thorough wave 1 uncertainty quantification noting that this may be revised in later waves of iterative decision support, as is the case in Section 5.9.

The Jade computer model is first linked to reality in Section 5.5.1. Geological uncertainty is assessed in Section 5.5.2 which is extended to include the additional uncertainty arising from efficient geological ensemble subsampling in Section 5.5.3, with NPV cost parameter uncertainty quantified in Section 5.5.4 including the important step of handling linear transformation effects. The uncertainty quantification is extended beyond that seen for the TNO OLYMPUS Well Control Optimisation Challenge to incorporate structural model discrepancy and decision implementation error in Sections 5.5.5 and 5.5.6 respectively. This is made possible by insight from product specialists and users at Emerson. A summary of the wave 1 uncertainty analysis is presented in Section 5.5.7.

5.5.1 Linking the Jade Model to the Real World System

For the purpose of demonstrating a complete uncertainty analysis it is assumed that the fictitious Jade model represents an actual real world oil reservoir. We link these using an uncorrelated additive error structure detailed in Equations (2.7.1) and (3.6.2). Due to the computational expense of evaluating the Jade ensemble, emulators are used as a surrogate for both the model and the utility function with the additive error structure extended as described in Equation (2.7.2). Uncertainties included within this analysis include: geological uncertainty; EGES uncertainty; NPV cost parameter uncertainty; structural model discrepancy; and decision implementation error. The choice of uncertainty model is justified as the uncertainties are perceived to be mutually uncorrelated with an additive effect on the utility function. Decision support using an implausibility measure proposed in Equation (3.6.3) requires that we assess the variance of the decision maker's utility function induced by each of the uncertainties.

5.5.2 Geological Uncertainty

Geological uncertainty, ε_{G}, is partially represented through the Jade model ensemble, however this is limited to the sample of $N = 50$ geological realisations. Our treatment

extends from these 50 models to consider the utility function uncertainty arising from possible alternative posterior samples from the underlying history matched stochastic geology model. This may be thought of as stepping between the finite sample of 50 geologies and the potentially infinite collection of valid geological configurations. Quantification of ensemble uncertainty requires simulations for all ensemble members for which we use the collection of exploratory simulations described in Section 5.1.2 to compute the sample variance, s^2, of the ensemble mean NPV over the exploratory decisions. It is assumed that $\mathbb{E}[\varepsilon_{\mathrm{G}}] = 0$, whilst $\mathrm{Var}[\varepsilon_{\mathrm{G}}]$ is estimated by $\frac{s^2}{N} = \frac{s^2}{50}$. Wave 1 results convey $\mathrm{Var}[\varepsilon_{\mathrm{G}}] = 1.555 \times 10^{14}$ as a reasonable assessment.

5.5.3 Efficient Geological Ensemble Subsampling Uncertainty

A subset of 3 of the 50 Jade models are used for the wave 1 simulations chosen via the EGES techniques implemented in Section 5.2 in order to achieve simulation computational savings. A consequence of EGES is that it induces some additional uncertainty in the utility function, denoted as $\varepsilon_{\mathrm{EGES}}$, and pertains to the use of a subset of the Jade model ensemble. EGES uncertainty is explicitly handled within the hierarchical emulation formulation in linking the emulators for the NPV for each sub-selected Jade model to the ensemble mean NPV in Section 5.4.4 which accounts for both the correlated uncertainties in the regression coefficient estimates and the residual uncertainty. For this residual uncertainty we assume that $\mathbb{E}[\varepsilon_{\mathrm{EGES}}] = 0$ whilst $\mathrm{Var}[\varepsilon_{\mathrm{EGES}}]$ is estimated by the square of the EGES linear model residual standard error which for wave 1 is 5.046×10^{12}. This represents 3.24% of the magnitude of the geological uncertainty variance and hence is a relatively small contributor to the overall uncertainty implying that the computational savings achieved by EGES are justified by the small increase in the uncertainty.

5.5.4 NPV Cost Parameter Uncertainty

As for the TNO OLYMPUS Well Control Optimisation Challenge, fixed NPV cost parameters are suggested for the oil price, water production and injection costs, and discount factor which are summarised in Table 5.2. These are used to compute the ensemble mean NPV, however they are not inputs to the Jade model so constitute a form of utility function cost parameter. In this section we investigate the more realistic scenario of time variable NPV cost parameters introducing NPV (or utility function) cost parameter uncertainty, ε_{C}. This is illustrated via a simulation study which follows the techniques presented in Section 3.5.2 that accounts for the linear transformation effects on the utility function.

Cost Parameter	Notation	Value
Oil Price	r_{op}	30 $ per bbl
Water Produced Cost	r_{wp}	6 $ per bbl
Water Injected Cost	r_{wi}	2 $ per bbl
Discount Factor	d	0.10

Table 5.2: Expected NPV fixed cost parameters within the Emerson Jade model application.

Simulation of NPV Cost Parameters

We introduce a stochastic economic model for time variable NPV cost parameters in the form of a stationary Gaussian process which is used to independently simulate the oil price, water produced and injected costs, and the discount factor (an extension on the TNO OLYMPUS Well Control Optimisation Challenge), with samples at 3 month intervals. In an alternative approach breaking with the TNO OLYMPUS Well Control Optimisation Challenge application and in a scenario which is more befitting of the realistic situation whereby a committee of decision makers may possess differing views on the long term average oil price depending on their level of optimism, we simulate multiple scenarios with fixed mean oil prices of $20, $25, $30, $35 and $40 per barrel, noting that these values are chosen such that their average is the proposed fixed value of $30 per barrel. This extra layer of the uncertainty

quantification is only performed for the oil price due to the previously highlighted importance of the contribution of oil to the expected NPV compared with water contributions. For all other NPV cost parameters the mean hyperparameter is fixed at their suggested values given in Table 5.2. Each simulation model uses an exponential covariance kernel with correlation length equal to 365 days and standard deviation parameter of 0.75, 0.12, 0.04 and 0.002 respectively. Compared to the TNO OLYMPUS Well Control Optimisation Challenge the choice of standard deviation hyperparameter for the time variable oil price of 0.75 is much smaller. This represents the greater certainty which members of a decision making committee may hold regarding their beliefs on the mean parameter. Moreover, the choice of NPV cost parameter simulation model is based on the belief that economic quantities do not vary smoothly with time. Note that this is an illustration and the described model is not designed to be representative of our true beliefs for how the actual cost parameters will vary over the field lifetime, but instead highlight the importance of including NPV cost parameter uncertainty by demonstrating its effect on decision support. This may be substituted for any other stochastic economic model. A total of $n_c = 1000$ NPV cost parameter time series samples are generated which is split into 200 for each mean oil price. The simulated oil price scenarios are illustrated in Figure 5.15 which shows time evolving traces for each of the fixed mean hyperparameter values.

NPV Cost Parameter Uncertainty Before the Removal of Linear Transformation Effects

The ensemble mean NPV is extremely quick to evaluate for a collection of Jade model simulations. For each of the 702 wave 1 decision parameter vectors, the NPV is evaluated for the subset of 3 Jade models using the 1000 generated cost parameter scenarios with the EGES linear model used to predict the ensemble mean NPV. As for the TNO OLYMPUS Well Control Optimisation Challenge, we illustrate the differing uncertainty quantification results before and after the removal of the

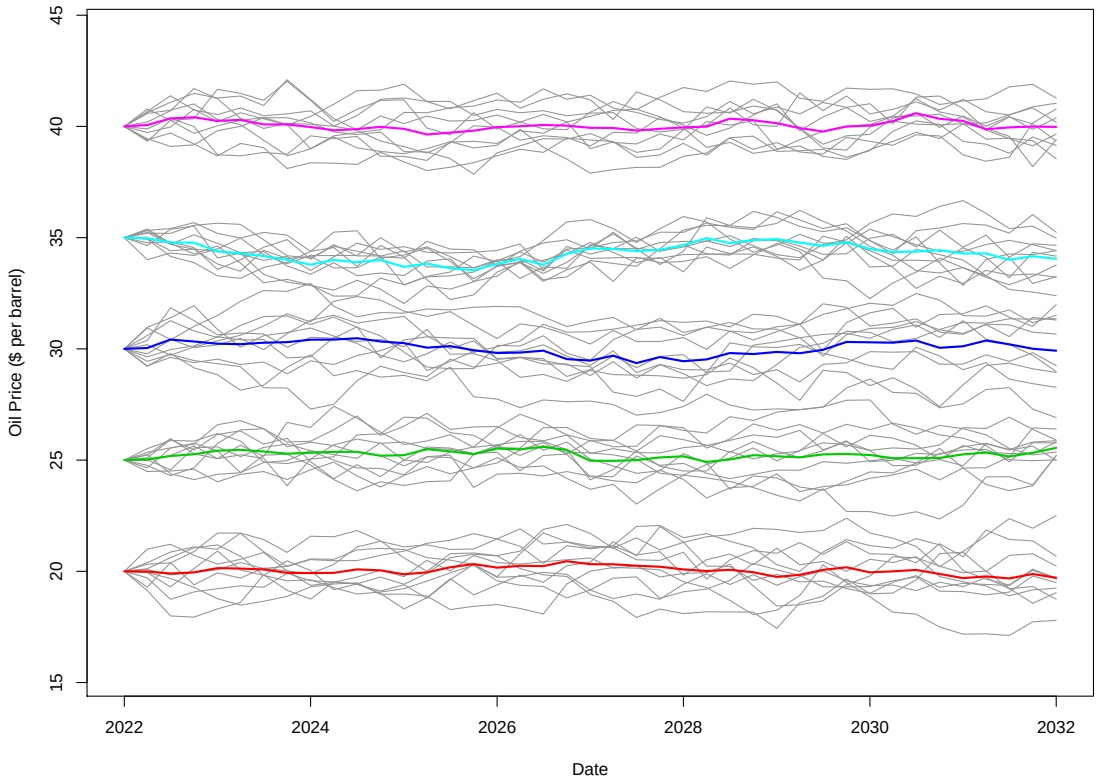


Figure 5.15: Trace plot of 10 simulated oil prices ($ per barrel) from a Gaussian process for each constant mean of $20, $25, $30, $35 and $40 per barrel, using an exponential covariance kernel with correlation length of 365 days, and standard deviation 0.75. The red, green, dark blue, light blue and pink lines denote the sample mean over the 10 simulated scenarios for each mean hyperparameter respectively.

linear transformation effects. The direct approach yields pairs plots such as those shown in Figure 5.16 for producer well B4 coloured by the standard deviation of ε_{C} where the largest uncertainty occurs for higher marginal parameter values. This is also evident in Figure 5.18a which shows the NPV cost parameter uncertainty versus the predicted ensemble mean NPV for which there is a strong positive linear association. A robust choice in the assessment of the NPV cost parameter uncertainty is therefore the maximum variance over the decision space; $\mathrm{Var}[\varepsilon_{\mathrm{C}}] = 7.879 \times 10^{16}$. This is problematic for decision support since this region is anticipated to contain good decision strategies whilst large uncertainties create difficulties in distinguishing

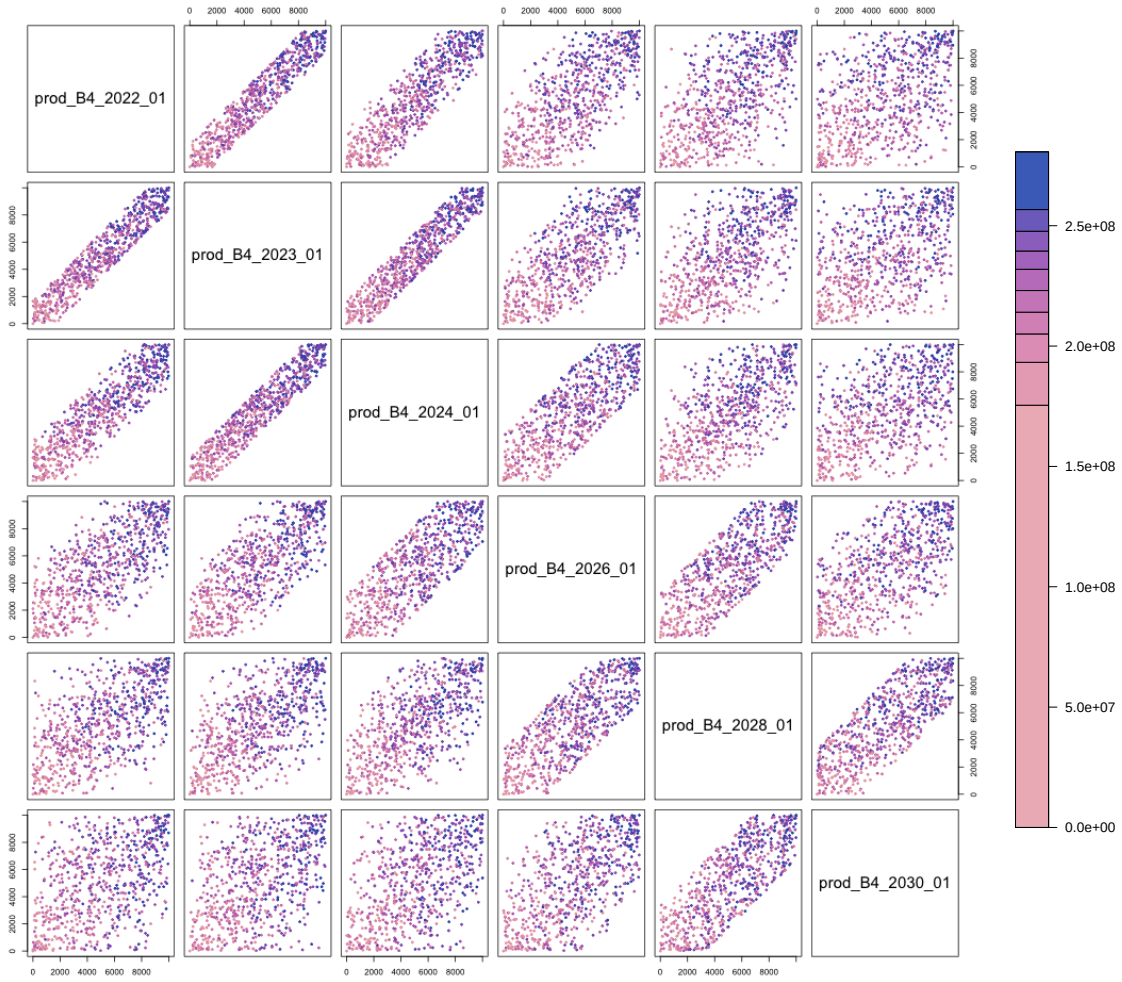


Figure 5.16: Jade wave 1 uncertainty quantification two-dimensional projections of the decision space for producer well B4 coloured by the standard deviations of the ensemble mean NPV utility function due to the cost parameter variability before the removal of linear effects. The colour scheme denotes the magnitude of the uncertainty on a relative scale with light pink through to dark purple corresponding to low through to high standard deviations.

between such decisions.

NPV Cost Parameter Uncertainty After the Removal of Linear Transformation Effects

Removal of the linear transformation effects is performed following the procedure described in Section 3.5.2 since these do not alter the location of the optimal decision.

For each of the n_c simulated cost parameter scenarios, linear models of the form shown in Equation (3.5.1) are fitted with a natural choice for $\mathbf{c}_0$ being the suggested fixed cost parameter settings listed in Table 5.2. The residual variability, estimated by $s^2_{\mathrm{C}}(\mathbf{d}_j)$ (see Equation (3.5.3)), is illustrated with a projection pairs plot for producer well B4 shown in Figure 5.17. The same colour scheme is used in Figures 5.16 and 5.17 to highlight the differences. Unlike for the TNO OLYMPUS Well Control Optimisation Challenge where there was a reversal in the direction of the association between larger decision parameter values and the magnitude of the uncertainty due to the cost parameter variability after linear effects have been accounted for, we observe a breaking of the association with a spread of standard deviation values across the entire parameter range. This is confirmed by the lack of association in Figure 5.18b of the NPV cost parameter uncertainty versus the predicted ensemble mean NPV, although it is noted that there is evidence of negative correlation indicating that the uncertainty has been reduced for the region that a priori is of most interest for optimal well control. The differences with the TNO OLYMPUS Challenge analysis can be explained by the alternative method of simulating the NPV cost parameter scenarios where the oil price is the main driver for high NPVs, however this leads to output which is more challenging to model via linear regression and hence the change in the pattern. Note that the mean residuals over the cost parameter scenarios are also examined using $m_{\mathrm{C}}(\mathbf{d}_j)$ defined in Equation (3.5.2). As in the TNO OLYMPUS Challenge these are typically at least an order of magnitude smaller than $s_{\mathrm{C}}(\mathbf{d}_j)$, and thus have negligible effect on the decision analysis.

Figure 5.19 shows a histogram of the variances of the utility function induced by NPV cost parameter uncertainty with the distribution centred at 1.152×10^{14}. This remains a substantial source of uncertainty that is generally of a larger magnitude than the EGES uncertainty. Based on the observations in Figure 5.18b, a suitable estimate for $\mathrm{Var}[\varepsilon_{\mathrm{C}}]$ is the median over the initial decision parameter space, leading to $\mathrm{Var}[\varepsilon_{\mathrm{C}}] = 6.465 \times 10^{13}$ as a robust choice. The magnitude of the uncertainty for many instances of large Jade ensemble mean NPV is comparable with this

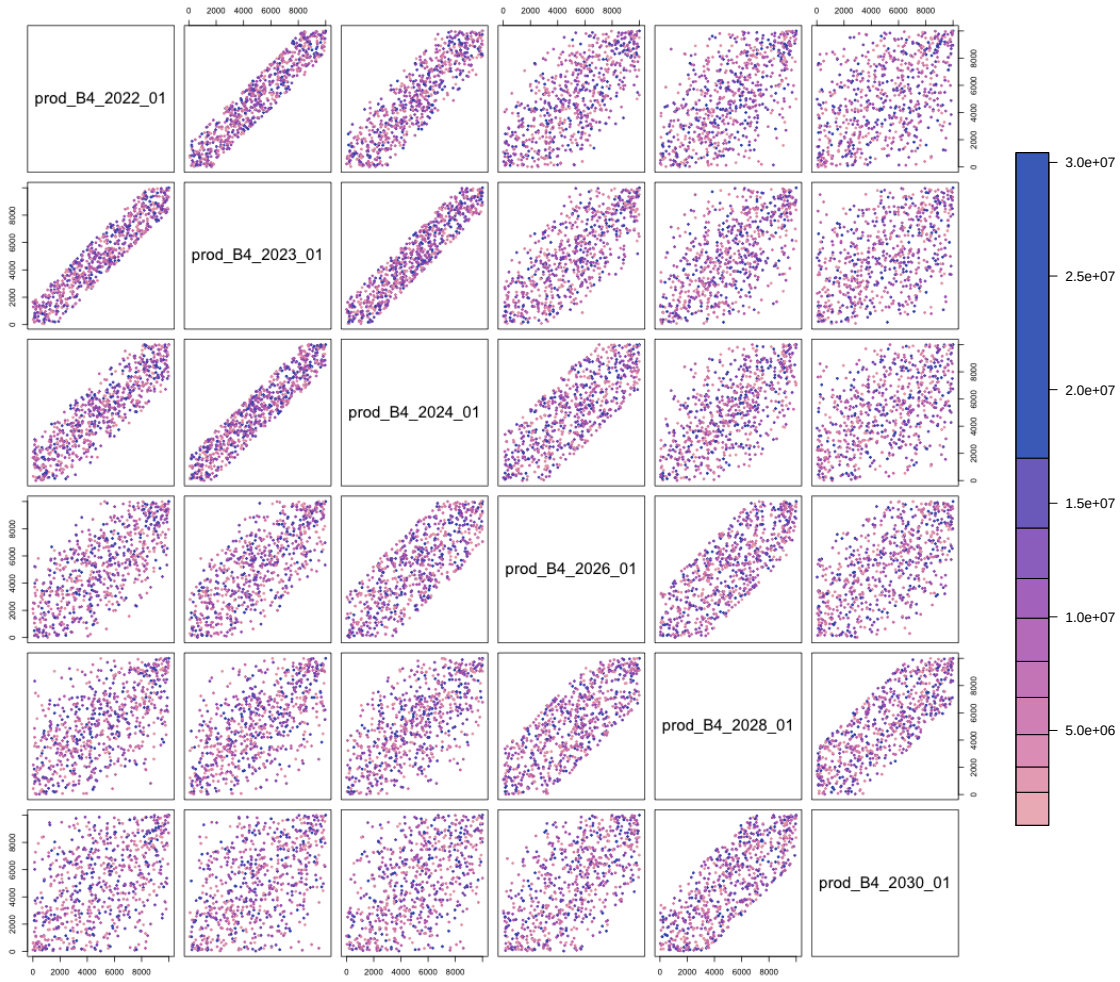
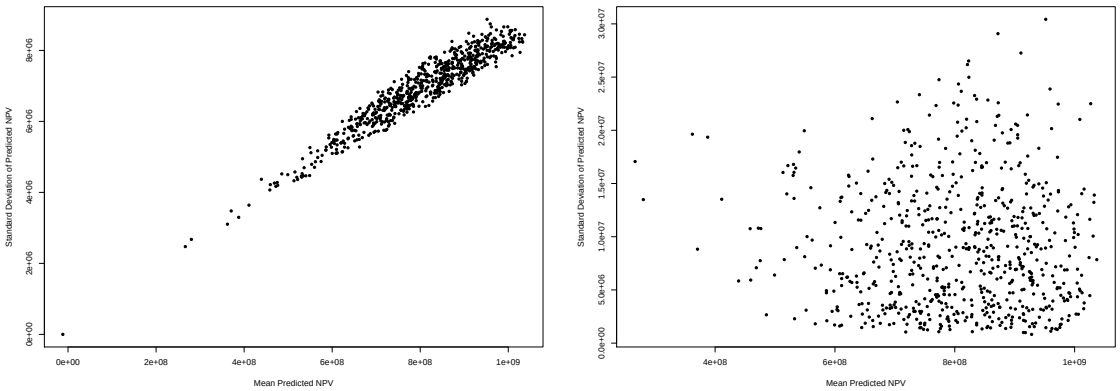


Figure 5.17: Jade wave 1 uncertainty quantification two-dimensional projections of the decision space for producer well B4 coloured by the standard deviations of the ensemble mean NPV utility function due to the cost parameter variability after the removal of linear effects. The colour scheme denotes the magnitude of the uncertainty on a relative scale with light pink through to dark purple corresponding to low through to high standard deviations.

uniform estimate. Prior to removing the linear effects this estimate is given by $\mathrm{Var}[\varepsilon_{\mathrm{C}}] = 7.879 \times 10^{16}$ (note that the median is 4.976×10^{16}), thus demonstrating a three order of magnitude reduction in this uncertainty. In addition, note that the maximum uncertainty after the removal of the linear transformation effects is 9.260×10^{14}, which still represents a two order of magnitude reduction. The implication for decision support is that a much smaller class of decisions may be



(a) Before the removal of linear effects. (b) After the removal of linear effects.

Figure 5.18: Jade wave 1 ensemble mean NPV standard deviation induced by NPV cost parameter variability versus the ensemble mean NPV over the wave 1 design computed using the $n_c = 1000$ simulated cost parameter scenarios.

obtained that are still robust to this uncertainty.

5.5.5 Structural Model Discrepancy

Model discrepancy provides a structured link between the computer model output and the real-world physical system. Whilst the Jade model is fictitious and is used to test and demonstrate Emerson's software, we introduce a model discrepancy term, ε_{MD}, as part of a comprehensive and realistic analysis guided by expert input. In our approach we represent model discrepancy as a proportion of the simulated ensemble mean NPV based on the intuition that this is: unrelated to the NPV distribution due to different geological ensembles (as characterised by geological uncertainty in Section 5.5.2); and that model discrepancy is expected to be proportional to the size of the expected NPV obtained for the Jade model due to the differences between the model and reality. A conservative estimate is obtained by considering the maximum simulated Jade ensemble mean NPV with a suitable proportion for the difference between model and reality being 0.1% representing one standard deviation. This assumes that: the ensemble mean NPV is a good representative for the expected NPV (see geological uncertainty in Section 5.5.2); and that the

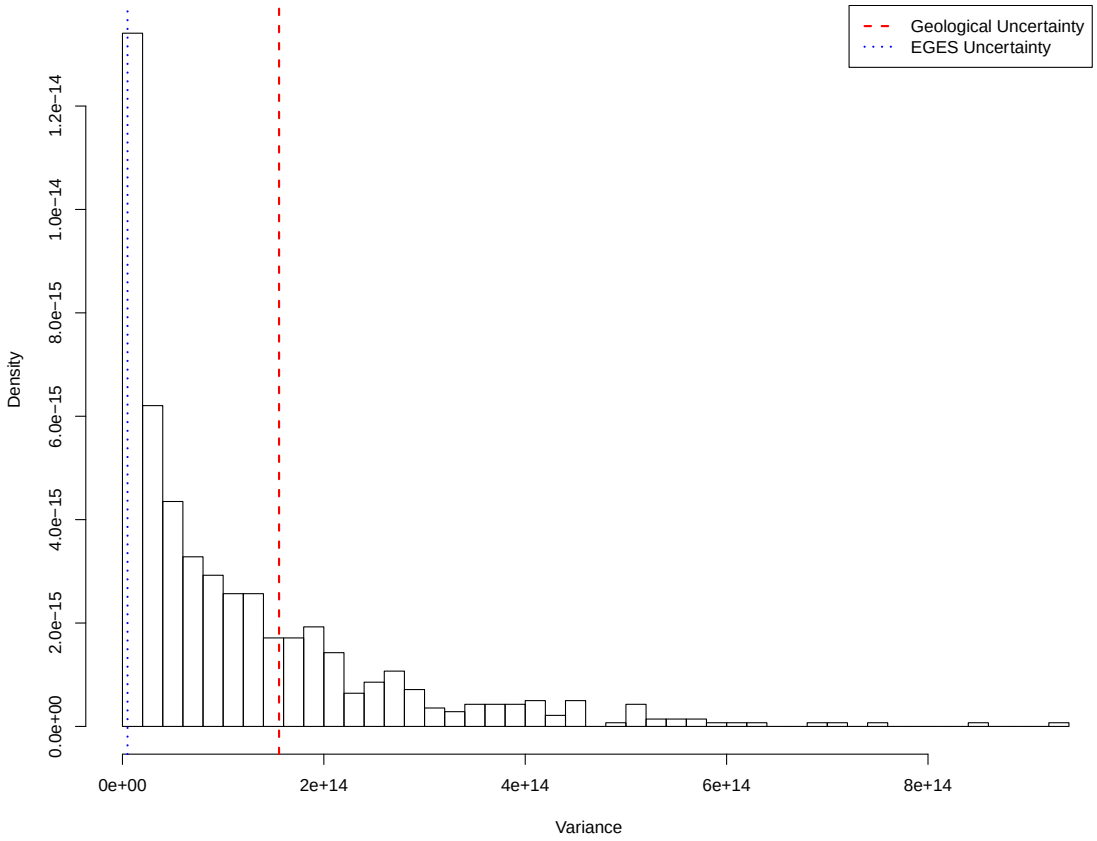


Figure 5.19: Histogram of the wave 1 variances of the Jade ensemble mean NPV over the generated cost parameter scenarios after the linear effect is removed. For comparison, the red dashed and blue dotted vertical lines correspond to the assessed values for $\text{Var}[\varepsilon_{\text{G}}]$ and $\text{Var}[\varepsilon_{\text{EGES}}]$ respectively.

specified decisions are executed precisely with any variation in the controls accounted for via the decision implementation error in Section 5.5.6. Note that whilst there is uncertainty in the predicted Jade ensemble mean NPV, this is already accounted for by the EGES uncertainty in Section 5.5.3. Under this setup we specify that $\mathbb{E}[\varepsilon_{\text{MD}}] = 0$, since a reasonable prior belief is that the Jade model was constructed to accurately represent the corresponding real world NPV under a particular decision strategy, and $\text{Var}[\varepsilon_{\text{MD}}] = 1.076 \times 10^{12}$, commenting that this is of comparable size to the EGES uncertainty, but smaller than the geological and NPV cost parameter uncertainties. For real world analyses we may be guided by the knowledge of experts including the model developers such as regarding the important approximations

and differences with the physical system. In cases where the prior belief is that $\mathbb{E}[\varepsilon_{\text{MD}}] \neq 0$ two modifications to the decision support implausibility measure in Equation (3.6.3) are necessary. Firstly, the numerator becomes $(\mathbb{E}_F[U(\mathbf{d})] - (U_{\max} + \mathbb{E}[\varepsilon_{\text{MD}}]))^2$, whilst the indicator function is modified to $\mathbb{1}_{\{\mathbb{E}_F[U(\mathbf{d})] < U_{\max} + \mathbb{E}[\varepsilon_{\text{MD}}]\}}$. A more thorough analysis would involve emulating the model discrepancy based on this setup. An alternative formulation would be to use a multiplicative error structure utilising the emulator expectation to obtain an approximation for the model discrepancy at different decision parameter settings.

5.5.6 Decision Implementation Error

Decision implementation error occurs due to the difference between the prescribed controls and those which are implemented for the real-world oil field. As for the model discrepancy we note that the Jade model is fictitious, although we include decision implementation error, ε_{I}, in order to illustrate its quantification and the effects on a comprehensive and realistic analysis. There exists two forms of implementation error: discrepancies in the values of the enacted controls, as described in Section 3.5.4; and uncertainty in the time of decision execution, often in the form of a delay. An assessment of the implementation error requires the evaluation of the computer model for a large number of decision parameter settings rendering such experimentation computationally infeasible, even for the Jade model which is of modest complexity and moderate run-time. Instead we propose using the fitted hierarchical emulator for the utility function as a substitute within these investigations to mitigate such computational concerns. This can be used to analyse the effects of the first form of decision implementation error, however it is not possible to assess the effect of time variable decisions since the emulator possesses decision parameters at a fixed time as inputs. If it was believed that this was important then this should also be assessed via direct experimentation using the Jade model, although real-world protocol should also be considered to mitigate this form of decision implementation uncertainty. An assessment of the first form is sufficient to demonstrate the inclusion

of implementation error and to provide a meaningful assessment for decision support.

The methodology described in Section 3.5.4 is used to assess decision implementation error using the hierarchical emulator constructed in Section 5.4 to first assess the variability in the NPV for each of the three sub-selected Jade models, before combining to emulate the Jade ensemble mean NPV. This begins by generating an initial 200 point space-filling design over the wave 1 decision parameter space. For each point, 100 decision parameter vectors are obtained via perturbations centred at the proposed $\mathbf{d}$ with each individual implementation error $e_{\mathrm{I},i}$ independently sampled conditional on the respective proposed d_i from $\pi_{\mathrm{I},i}(e_{\mathrm{I},i} \mid d_i) \sim \mathcal{TN}(0, \sigma^2_{e_{\mathrm{I},i}}, -a_{\mathrm{I},i}, a_{\mathrm{I},i})$, a truncated normal distribution, where $\sigma_{e_{\mathrm{I},i}} = \frac{M_i - m_i}{300}$, and truncation bounds given by $a_{\mathrm{I},i} = \min\{\frac{M_i - m_i}{100}, d_i - m_i, M_i - d_i\}$, with m_i and M_i denoting the minimum and maximum of the i^{th} parameter range respectively. The choice of a truncated normal distribution represents that operatives will attempt to achieve the stipulated control value, but with some potential for error with decreasing probability as the magnitude increases. The hyper-parametrisation represents a permitted tolerance of 1% of each parameter's range which is treated as a three standard deviation credible interval in specifying $\sigma_{e_{\mathrm{I},i}}$. In cases where an individual decision parameter is within 1% of its respective minimum or maximum, the truncation bounds are contracted to the difference with the nearest range end-point in the second or third terms for $a_{\mathrm{I},i}$, although there is no change to $\sigma_{e_{\mathrm{I},i}}$. This is to ensure consistency with the mean equal to the proposed decision parameter for both the joint and marginal distributions. A total of 2×10^4 emulator evaluations are performed. Preliminary investigations show that 100 perturbations for each initial $\mathbf{d}$ are sufficient to obtain an accurate characterisation of the ε_{I} with additional hierarchical emulator evaluations not merited as they only yield a minor decrease in the induced variance. This enables exploration of a greater number of proposed decision parameter settings. Assessment of the mean emulator adjusted expectation for each fixed decision parameter location provides no evidence against the assumption that $\mathbb{E}[\varepsilon_{\mathrm{I}}(\mathbf{d})] = 0$ for all $\mathbf{d} \in \Omega_0$, whilst a conservative assessment of the variance due to decision implementation error as the 95^{th}

percentile yields $\text{Var}[\varepsilon_{\text{I}}] = 6.821 \times 10^{12}$. This is justified based on the histogram of the variances over the initial 200 sampled decision parameter vectors in Figure 5.20.

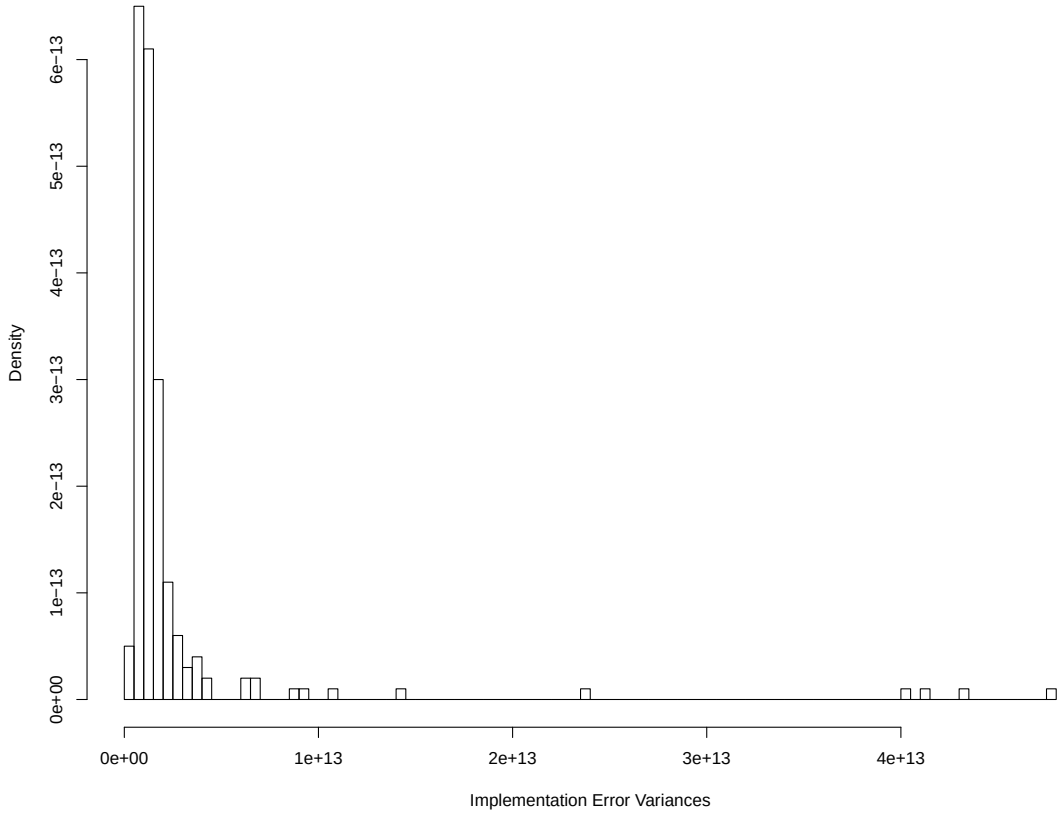


Figure 5.20: Histogram of the wave 1 variances of the Jade ensemble mean NPV due to decision implementation error.

5.5.7 Uncertainty Analysis Summary

A summary of the wave 1 uncertainty analysis is presented in Table 5.3 which shows the variances and the proportion of the total wave 1 uncertainty variance. A comparison with the wave 2 uncertainty analysis is given in Table 5.5 on page 291. The individual uncertainties, excluding that due to the emulator, are combined following the uncorrelated additive error structure described in Equation (2.7.1) to obtain a total uncertainty variance of 2.281×10^{14}. Moreover, note that this excludes the EGES uncertainty which is encompassed within the novel hierarchical emulator for the utility function during the decision support analysis. It is observed

Source of Uncertainty	Variance	Proportion of Total Uncertainty
Geological Uncertainty	1.555×10^{14}	68.2%
EGES Uncertainty	5.046×10^{12}	–
NPV Cost Parameter Uncertainty	6.465×10^{13}	28.3%
Structural Model Discrepancy	1.076×10^{12}	0.5%
Decision Implementation Error	6.821×10^{12}	3.0%
Total Uncertainty	2.281×10^{14}	–

Table 5.3: Emerson Jade model application wave 1 uncertainty analysis results including the variances and proportion of total variances for the Emerson Jade model application obtained using an additive error structure described in Equation (2.7.1). Note that the hierarchical emulator construction implicitly encompasses the EGES uncertainty to predict the ensemble mean NPV, hence it is omitted from the total uncertainty. A comparison with the wave 2 uncertainty analysis is given in Table 5.5 on page 291.

that geological uncertainty is the largest contributor to the overall uncertainty, and NPV cost parameter uncertainty represents a reasonable amount. In contrast, structural model discrepancy and decision implementation error are relatively minor contributors within wave 1 of our analysis. A decision support sensitivity analysis may be performed to assess whether this conclusion is a result of the modelling choices. For the structural model discrepancy, under the same framework as used in Section 5.5.5, it would be necessary to inflate the proportion of the maximum simulated Jade ensemble mean NPV by a factor of 60 and 144 in order to obtain an uncertainty of comparable size to that induced by the NPV cost parameters or the underlying geology respectively. This corresponds to proportions of 6% and 14.4% respectively as representing one standard deviation of structural model discrepancy; both of which seem excessively large given the efforts which go into accurately constructing such models.

5.6 Decision Support Wave 1 for Well Control Optimisation

The first wave of iterative decision support for well control optimisation applied to the Emerson Jade model is performed in this section using $U(\mathbf{d}) = \mathbb{E}[\text{NPV}](\mathbf{d})$ within the procedure formulated in Section 3.6. This incorporates the EGES technique, construction of a wave 1 targeted Bayesian design, the hierarchical emulator exploiting known simulator behaviour and the wave 1 uncertainty analysis implemented in Sections 5.2 to 5.5 respectively. Results are presented in Section 5.6.1 with a comparison under alternative uncertainty analyses discussed in Section 5.6.2.

5.6.1 Results

Wave 1 of iterative decision support is performed using the hierarchical emulator exploiting known simulator behaviour devised in Section 5.4 for the utility function incorporating all uncertainties discussed in Section 5.5. Note that the EGES uncertainty is encompassed within the hierarchical emulator formulation. The wave 1 non-implausible region is defined as decision parameter vectors which satisfy $I_{\text{DS}}(\mathbf{d}) \leq c = 3$ for which it is established that 3.72% of the decision space obtained by imposing both the operational range and difference constraints is classified as non-implausible, thus highlighting the immense benefits of using an emulator to focus in on a small region that contains potentially good control strategies. Moreover, this equates to 7.029×10^{-2}% of the full hypercube volume formed by the operational range constraints only. Compared to the analysis of the TNO OLYMPUS Well Control Optimisation Challenge, this demonstrates how the incorporation of structure into the emulator can yield vast improvements for decision support.

An example of the visualisation of the wave 1 decision support non-implausible region is seen in the implausibility pairs plot for the decision parameters for producer well B4 in Figure 5.21. This includes implausibility measure plots below the diagonal, the implausibility optical density above the diagonal, and marginal density

histograms along the diagonal. Note that for the implausibility plots, the axes are swapped to match the corresponding optical density plot whilst the colours green, yellow and red convey where $0 < I_{\mathrm{DS}}(\mathbf{d}) \leq 2$, $2 < I_{\mathrm{DS}}(\mathbf{d}) \leq 3.5$ and $I_{\mathrm{DS}}(\mathbf{d}) > 3.5$ respectively. Blue triangles denote where the implausibility measure is zero and hence the emulator for the expected NPV adjusted expectation exceeds the current maximum simulated NPV. For the optical density plots, dark blue through to yellow depict low to high density of points satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c$. Examination of these plots already emphasises the importance of setting very high early time target production rates, whilst unlike in the TNO OLYMPUS Well Control Optimisation Challenge, the middle and later time rates must also be relatively high. Similar patterns are observed for the other producer wells with a reversal seen for the injector wells where lower target rates are favoured at earlier times with increasing values at later times to sustain the desired production rates. This is explained by the effect of a high discounting factor.

These results are corroborated by decision support directly using the simulation output in which the emulator uncertainty is omitted from the implausibility measure formula. Figure 5.22 shows plots of the wave 1 simulations for the predicted ensemble mean NPV computed using the EGES linear model versus several decision parameters. The horizontal blue line denotes the fixed threshold with any decision parameter vectors classified as non-implausible if its simulated ensemble mean NPV exceeds this line, as is the case for the red points. It is evident in Figure 5.22a for prod_C5_2024_01 that a relatively high target rate is preferred for this middle-time target production rate, whilst in Figure 5.22b there is less restriction over the exact value of the target injection rate, although there is a slightly higher concentration for smaller values which lead to lower injection costs. The height of the blue line is computed as $U_{\max} - c\sqrt{\mathrm{Var}[\varepsilon_{\mathrm{MD}}] + \sum_i \mathrm{Var}[\varepsilon_i]}$, where $c = 3$. This provides an illustration of how iterative decision support proceeds to identify decision parameter vectors which yield a utility function value that exceeds a threshold defined with respect to the incorporated uncertainties related to the unknown decision maker's

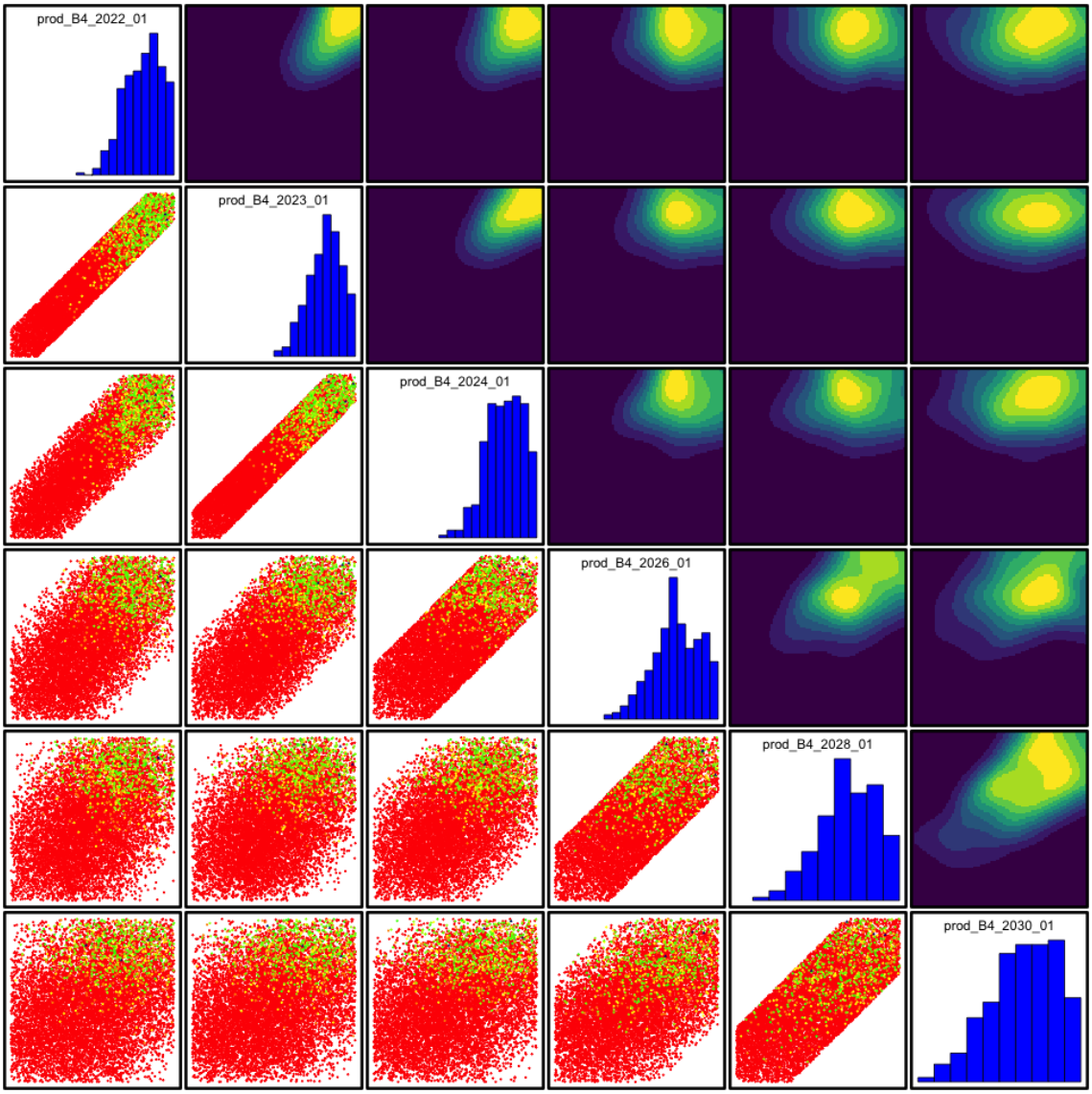
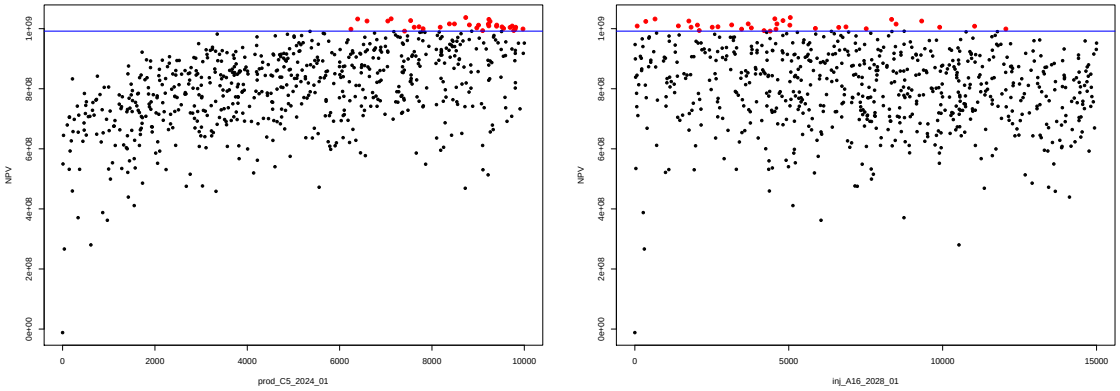


Figure 5.21: Jade wave 1 decision support implausibility measure (below the diagonal) and optical density (above the diagonal) pairs plot, with marginal density histograms (along the diagonal) for the decision parameters for producer well B4. In the implausibility plots, the axes are swapped to match the corresponding optical density plot in which the colours green, yellow and red indicate where $0 < I_{\mathrm{DS}}(\mathbf{d}) \leq 2$, $2 < I_{\mathrm{DS}}(\mathbf{d}) \leq 3.5$ and $I_{\mathrm{DS}}(\mathbf{d}) > 3.5$ respectively, whilst blue triangles denote where $I_{\mathrm{DS}}(\mathbf{d}) = 0$ which occurs when $\mathbb{E}_F[U(\mathbf{d})] \geq U_{\max}$. For the optical density plots, dark blue through to yellow depict low to high density of points satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c$.

utility. It is the magnitude of these uncertainties which thus determines the size of the non-implausible region. Any changes to their assessment results in a shift of this threshold where the utility is a function of the decision parameters. Further exploration of the non-implausible decision space is conducted for the wave 2 analysis in Section 5.10.1.



(a) Predicted ensemble mean NPV versus prod_C5_2024_01.

(b) Predicted ensemble mean NPV versus inj_A16_2028_01.

Figure 5.22: Jade wave 1 simulations predicted ensemble mean NPV via the EGES linear model versus decision parameters. The horizontal blue line denotes the fixed threshold defined with respect to the wave 1 uncertainty analysis at height $U_{\max} - c\sqrt{\mathrm{Var}[\varepsilon_{\mathrm{MD}}] + \sum_i \mathrm{Var}[\varepsilon_i]}$, where $c = 3$, above which decision parameter vectors are classified as non-implausible. These are highlighted by the red points

5.6.2 Decision Support Wave 1 Results for Alternative Uncertainty Quantifications

A decision support sensitivity analysis is performed in this section to assess the effect of various uncertainty specifications exploiting the computational efficiency of the developed iterative decision support procedure. The first wave of decision support is repeated for several alternative uncertainty specifications with the results summarised in Table 5.4. A comparison with the wave 2 results is shown in Table 5.6 on page 298. For clarity, column two shows the non-implausible volume of the wave 1

decision space obtained by imposing both the range and difference constraints, whilst column three depicts the non-implausible volume of the entire hypercube formed by the range constraints only. All comments are with respect to the fully constrained decision space.

Uncertainty Specification	Non-Implausible Volume	Non-Implausible Volume of Hypercube
Hierarchical Emulator with All Uncertainties	3.72%	7.03×10^{-2}%
Omitting Geological Uncertainty	2.80%	5.29×10^{-2}%
Before Removing NPV Cost Parameter Uncertainty Linear Effect	99.86%	1.89%
Emulator Uncertainty Only	2.33%	4.40×10^{-2}%
No Emulator Uncertainty	0.99%	1.87×10^{-2}%
No Uncertainties or Emulator Uncertainty	0.04%	7.56×10^{-4}%

Table 5.4: Summary of the proportion of decision space classified as non-implausible after wave 1 decision support for the Emerson Jade model for various included uncertainties within the decision support implausibility measure using threshold value $c = 3$. The first row displays the results using the hierarchical emulator for the utility function and a comprehensive uncertainty quantification with all subsequent rows omitting various forms of uncertainty. The second and third columns show the non-implausible proportion of the initial decision space formed by imposing both the range and difference constraints, and the hypercube formed by only the range constraints respectively. Note that imposition of the difference constraints before the wave 1 analysis results in a reduction to 1.8895% of the initial hypercube volume due to the range constraints only. A comparison with the wave 2 results is shown in Table 5.6 on page 298.

Geological uncertainty is of principal concern within the oil industry, as evidenced by the TNO OLYMPUS Field Development Optimisation Challenge and general petroleum reservoir engineering practice in generating ensembles with differing underlying geologies. Our more comprehensive treatment beyond a fixed ensemble of finite size linking to the potentially infinite number of geological configurations in Section 5.5.2 yields a wave 1 assessment of $\mathrm{Var}[\varepsilon_{\mathrm{G}}] = 1.555 \times 10^{14}$. This is an uncer-

tainty of a substantial magnitude, as highlighted in Table 5.3 (page 260). Omission from the decision support analysis results in a non-implausible proportion of 2.80%, as seen in the second row of Table 5.4 representing approximately a one-quarter decrease in the volume compared with when geological uncertainty is fully incorporated. Consequently, some decision parameter settings which may be deemed of interest were the decision makers to order the evaluation of further geological ensemble members, could be falsely eliminated at wave 1.

In Section 5.5.4 we consider the differences before and after accounting for the linear transformation effect of the NPV cost parameter uncertainty on the utility function obtaining a wave 1 assessment of $\text{Var}[\varepsilon_{\text{C}}] = 7.879 \times 10^{16}$ and $\text{Var}[\varepsilon_{\text{C}}] = 6.465 \times 10^{13}$ respectively. The uninformed version dominates the uncertainty quantification leading to a non-implausible proportion of 99.86%. It is almost impossible to distinguish between the decision parameter settings with such a large source of uncertainty and hence meaningful decision support cannot be provided. As for the TNO OLYMPUS Well Control Optimisation Challenge, the importance of performing an uncertainty quantification whilst taking into account the wider objective of the analysis is emphasised, with linear transforms of the utility function presenting no change to the decision support analysis in terms of the location of good decisions. This result is shown in the third row of Table 5.4.

Omitting all uncertainty except for those due to the emulator as a surrogate for the Jade expected NPV, as represented by specifying $\mathbb{E}[\varepsilon_i] = 0$ and $\text{Var}[\varepsilon_i] = 0$ for all uncertainties, yields a wave 1 non-implausible volume of 2.33% of the decision space (see the fourth row of Table 5.4). This corresponds to the prior belief that up to emulator uncertainty, there is no difference between the Jade model and the real world field. As in the case of omitting the geological uncertainty, there is a considerable decrease from the initial 3.72% non-implausible volume, thus accentuating the need for a realistic and comprehensive uncertainty quantification to avoid falsely eliminating decision settings which may produce high expected NPVs. These uncertainties may be re-assessed and subsequently potentially decreased in

later waves of iterative decision support.

The assumption that the emulator is a perfect representation of the Jade model is characterised by omitting the emulator uncertainty, whilst retaining the rest of the uncertainty quantification. This results in a wave 1 decision support non-implausible volume of 0.99% in the fifth row of Table 5.4. This investigation is equivalent to treating the emulator expectation as equal to the simulator output, and approximates decision support directly using the Jade model if sufficient computational resources were available. The wave 1 result provides a theoretical lower bound on the proportion of space remaining after wave 1 for the current uncertainty quantification suggesting that further refinement of the emulator is capable of a further reduction from 3.72% to close to 0.99% without any additional simulations. In addition, this non-implausible volume is smaller than that achieved by omitting the uncertainty quantification whilst retaining the emulator uncertainty. This implies that at wave 1 the emulator represents the largest single source of uncertainty, and hence a wave 2 analysis is necessary to further reduce the non-implausible space. Note that for the majority of decision parameter vectors the wave 1 emulator uncertainty is greater than the largest resolvable uncertainty, geological uncertainty, from the wave 1 uncertainty quantification (see Table 5.3).

Neglecting all sources of uncertainty including both the emulator and all of those detailed in Section 5.5 and identifying cases where $I_{\mathrm{DS}}(\mathbf{d}) = 0$ results in a non-implausible volume of 0.04%. This is shown in the sixth row of Table 5.4. This analysis represents the situation where it is assumed that the Jade model is a perfect representation of the corresponding real world oil field, and that the emulator is an exact portrayal of the model, with no other sources of uncertainty. This is unrealistic and thus highlights the dangers of inadequately linking the various layers of the modelling framework with the real world physical system. Within decision support this leads to many potentially good decision strategies being falsely rejected, although it also provides insight of which well control strategies are expected to yield improvements versus the current best simulated strategy.

5.7 Wave 2 Design of Simulations

Wave 1 of the iterative decision support procedure has substantially reduced the decision parameter space which is classified as non-implausible to only 3.72% of the initial decision space formed by imposing both constraint types. A second design over this smaller space must be constructed in order to perform a second wave of decision support. A method of constructing a design for wave 2 is described in Section 5.7.1 with the results of its application to the Jade model presented in Section 5.7.2.

5.7.1 Wave 2 Design Methodology

Iterative decision support provides an insight into the size, shape and location of the non-implausible region, however it does not yield a probability distribution over the parameters, as would be the case within a full Bayesian analysis. It is therefore desirable to construct a uniform design over the wave 1 non-implausible decision space, which may be of a complex shape, in order to achieve good exploration and coverage. The following algorithm is used to generate a design of size n which is approximately optimised with respect to a design selection criterion such as the minimax criterion (see Equation (2.3.4)) via a comparison with a reference design of at least n_{ref} points, where $n_{\text{ref}} \gg n$.

1. Uniformly sample batches of n_{prop} decision parameter vectors over the initial decision space, $\mathbf{d} \in \Omega_0$. Use the wave 1 hierarchical emulator and uncertainty analysis to obtain a collection which are in the wave 1 non-implausible region. This is any $\mathbf{d}$ satisfying $I_{\text{DS}}(\mathbf{d}) \leq c$, where c is the threshold value used within the iterative decision support procedure. Repeat this process until at least $10n$ non-implausible decision parameter vectors are obtained.
2. Formulate a second uniform sample of at least n_{ref} points following the same procedure as for the proposal points in step 1.
3. Identify any additional decision parameter vectors from the wave 1 analysis that are of interest for evaluating the simulator and with which to augment

the wave 2 design. Examples may include:

- $\mathbf{d}_{\max,\exp} = \arg\max_{\mathbf{d}\in\Omega} \mathbb{E}_F[U(\mathbf{d})]$, the decision parameter vector which maximises the utility function emulator adjusted expectation and hence is expected to yield simulator output with a utility function value that exceeds the current highest utility function evaluation.
- $\mathbf{d}_{\max,3\text{CIupr}} = \arg\max_{\mathbf{d}\in\Omega} \mathbb{E}_F[f(\mathbf{d})] + 3\sqrt{\text{Var}_F[U(\mathbf{d})]}$, the decision parameter vector which maximises the 3 adjusted standard deviation credible interval upper bound where there is a possibility of a high utility function value taking into consideration both the adjusted expectation and the emulator uncertainty. Credible interval upper bounds of alternative widths may also be considered.

Note that this step is optional. Also, it is possible that multiple of these augmentation decision parameter vectors may coincide.

4. Formulate n_{gen} candidate designs by randomly selecting n points from those generated in step 1 and augment to include any additional decision parameter vectors from step 3.
5. Approximately optimise the design with respect to the minimax design selection criterion by comparing to the reference design formulated in step 2.

The algorithm naturally extends to designs for later waves by sequentially evaluating the decision support implausibility measure and checking the cut-off criterion for each wave in steps 1 and 2. In situations where either the current non-implausible region represents a very small volume of the initial decision parameter space, or many waves of decision support have already been performed, this can become computationally challenging. Solutions to this include the techniques discussed in Section 2.8.5.

Another use for the algorithm is to assess the proportion of the decision space classified as non-implausible at wave 1 by exploiting the recursive structure of generating batches of points to test versus the decision support implausibility criterion cut-off in both steps 1 and 2. The large number of sampled points produces an accurate estimate for the non-implausible proportion summary statistics. Moreover,

each batch can be viewed as an individual estimate of this statistic and hence used to measure its variability resulting from alternative proposed uniform samples.

5.7.2 Wave 2 Design

In this section we present the construction of a wave 2 design for the Jade model application following the algorithm described in Section 5.7.1. This is the first step in a second wave of iterative decision support for which certain aspects of the setup may be reviewed including: the sub-selected Jade models from the EGES analysis; and the choice of decision parameters.

The same three Jade models as in wave 1 identified using the EGES techniques are used for wave 2 simulations. This is justified since the EGES uncertainty was comparatively small versus all other sources of uncertainty in the wave 1 analysis; noticeably it is two orders of magnitude smaller than the geological uncertainty, and hence the additional computational costs of simulations for additional ensemble members is not warranted. Moreover, a re-assessment over the wave 1 non-implausible decision space to identify a new and potentially larger representative subset would require an additional collection of simulations for all 50 Jade models. Similarly, the additional computational expense is not believed to be of sufficient benefit.

For the wave 2 decision support analysis the same $D = 36$ decision parameters as for wave 1 are used, namely six control intervals starting on January 1, 2022, 2023, 2024, 2026, 2028 & 2030 for each of the six wells. In addition, there are no changes to the operational range or difference constraints defined for wave 1 in Section 5.3.

Given this setup, the aim is to formulate a wave 2 design of size $n = 700$ by applying the algorithm in Section 5.7.1. In step 1 we uniformly sample $\mathbf{d} \in \Omega_0$ in batches of size $n_{\text{prop}} = 10^4$ until we possess at least $10n = 7000$ decision parameter vectors satisfying $I_{\text{DS}}(\mathbf{d}) \leq 3$. The choice of $n_{\text{prop}} = 10^4$ is justified based on the wave 1 analysis where 3.72% of the decision space was classified as non-implausible and for reasons of emulator computational efficiency and memory. A total of 18 iterations were necessary to yield 7283 proposal points in Ω_1. The same process was

repeated in step 2 to construct a reference design containing $n_{\text{ref}} \geq 10^4$ points in Ω_1 requiring 26 batches yielding $n_{\text{ref}} = 10405$. Note that this analysis indicates that the actual non-implausible volume is approximately 4.02% with an estimated standard error for batches of $n_{\text{prop}} = 10^4$ points of 0.22. Note that the initial wave 1 estimate of 3.72% therefore lies within two standard deviations of this value.

Design augmentation is considered as in step 3 for $\mathbf{d}_{\text{max,exp}}$ and $\mathbf{d}_{\text{max,3CIupr}}$, with the maximisation performed approximately with respect to a large number of $\mathbf{d} \in \Omega_0$. It is discovered that $\mathbf{d}_{\text{max,exp}}$ and $\mathbf{d}_{\text{max,3CIupr}}$ (approximately) coincide, hence the wave 2 design is augmented to include only one extra decision parameter vector. A collection of $n_{\text{gen}} = 10^4$ candidate designs are constructed by randomly selecting $n = 700$ of the proposed non-implausible $\mathbf{d}$, with each design augmented to also include $\mathbf{d}_{\text{max,exp}}$, as in step 4. Approximate design optimisation is then performed with respect to the minimax design selection criterion through a comparison with the large reference design to conclude step 5.

The wave 2 design is illustrated and compared with the wave 1 design in Figures 5.23 and 5.24. These are pairs plots for the decision parameters for producer well B4 and injector well A16 respectively where the above diagonal plots encompass both the wave 1 and 2 designs which are denoted by grey and red points respectively. The below diagonal plots show the wave 1 design only. In all plots the blue points indicate the wave 1 design points that are classified as non-implausible for decision support directly using the simulation output along with the wave 1 uncertainty quantification. For producer well B4 the pattern closely resembles the non-implausible region in Figure 5.21 corresponding to generally high target production rates, especially for earlier time control intervals. Moreover, it is evident that this covers a much smaller region than the wave 1 design whilst noting that the red points are situated in similar locations to the blue wave 1 design points, although with slightly greater spread arising due to the targeted sampling of a larger number of points classified as non-implausible in wave 1. The pattern is less well defined for injector well A16, although it remains the case that lower target injection rates are favoured

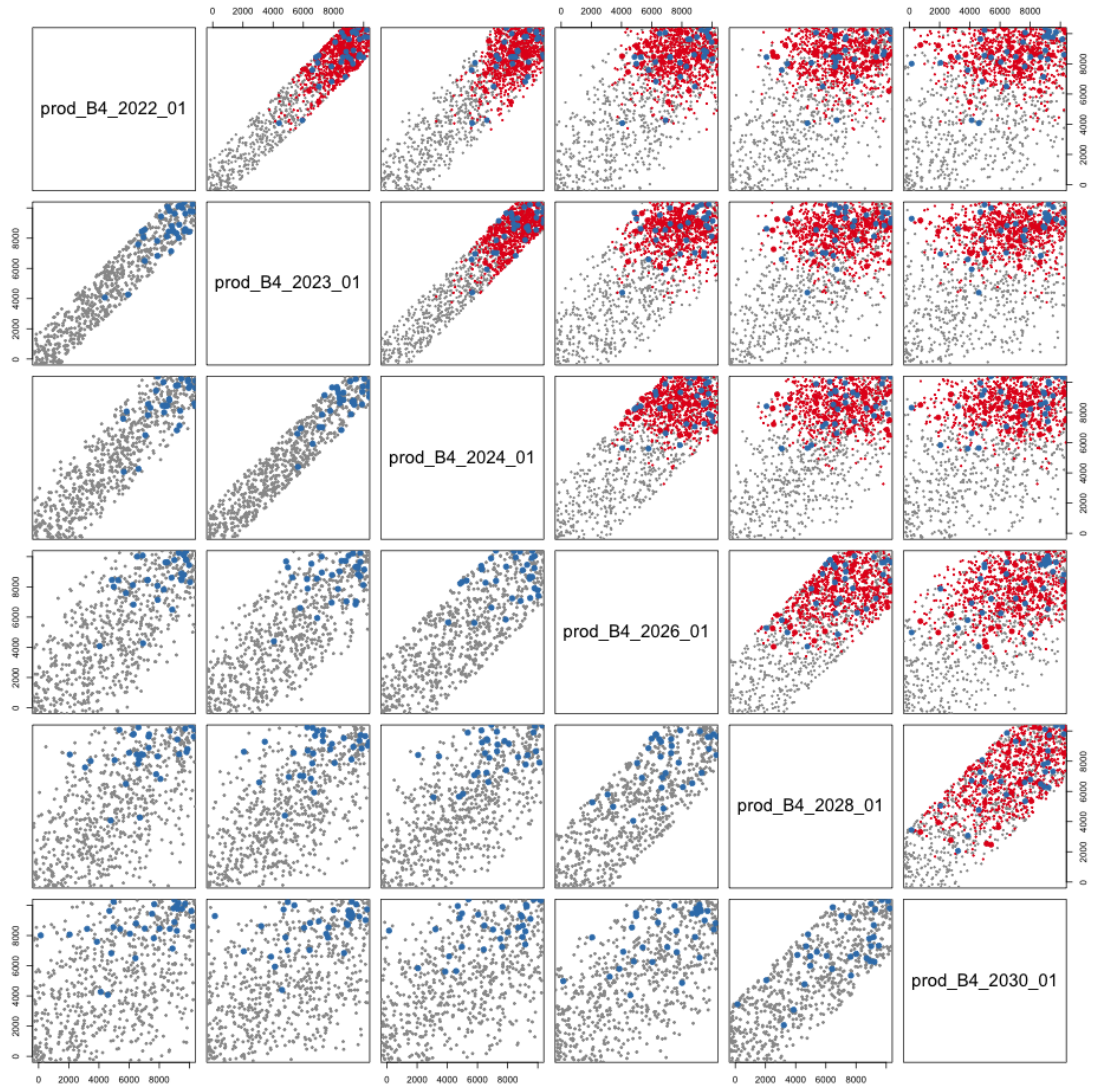


Figure 5.23: Comparison of the designs for Jade wave 1 and 2 pairs plot for producer well B4 decision parameters. The wave 2 design consists of 701 points generated using the algorithm presented in Section 5.7.1 with the minimax distance criterion. The above diagonal plots show the wave 1 and 2 design points in grey and red respectively, whilst the below diagonal plots show the wave 1 design only. In all plots the blue points indicate the wave 1 design points that are classified as non-implausible for decision support directly using the simulation output along with the wave 1 uncertainty quantification.

in order to attain higher expected NPVs. As for producer well B4, the red wave 2 design points are in close proximity to the blue wave 1 design points.

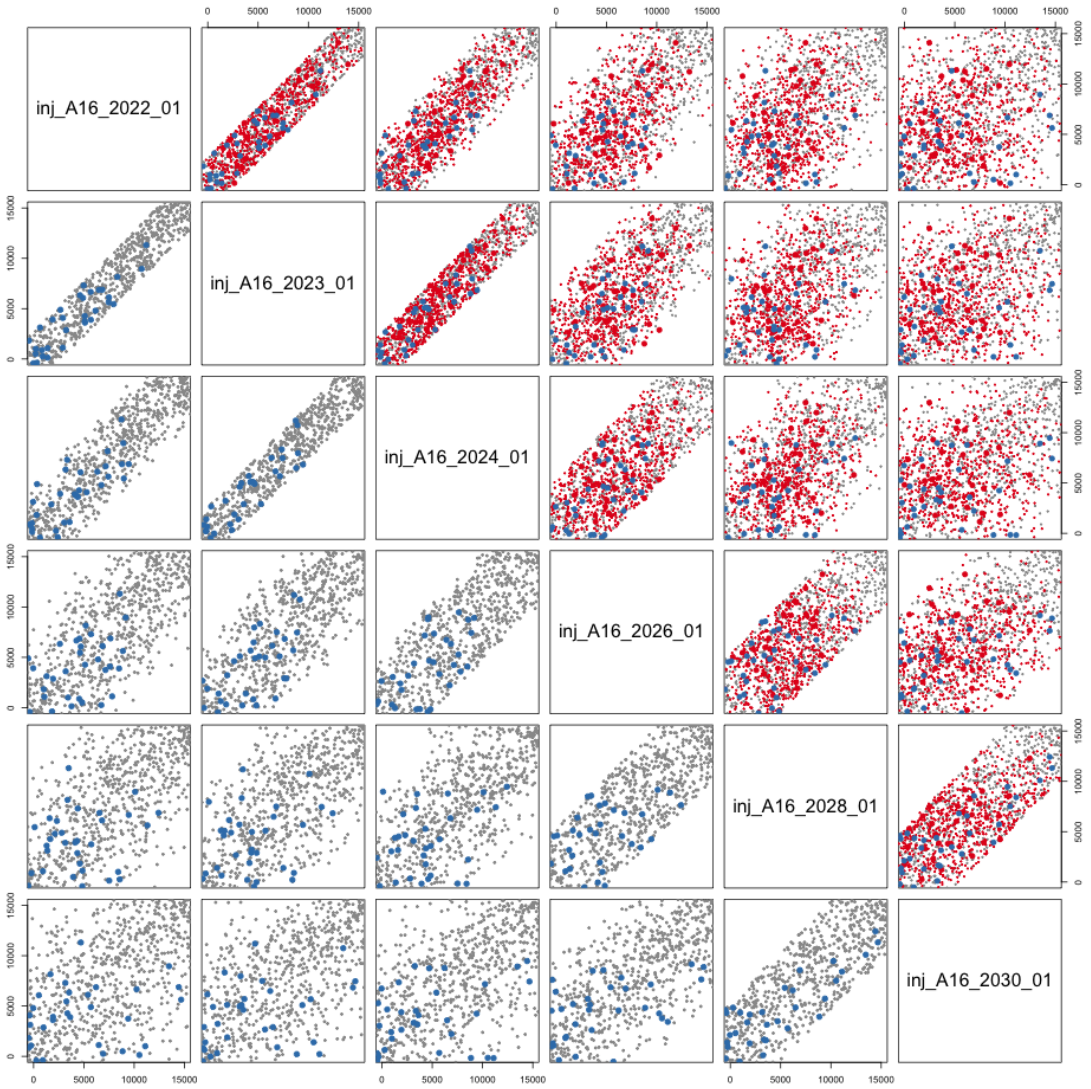


Figure 5.24: Comparison of the designs for Jade wave 1 and 2 pairs plot for injector well A16 decision parameters. The wave 2 design consists of 701 points generated using the algorithm presented in Section 5.7.1 with the minimax distance criterion. The above diagonal plots show the wave 1 and 2 design points in grey and red respectively, whilst the below diagonal plots show the wave 1 design only. In all plots the blue points indicate the wave 1 design points that are classified as non-implausible for decision support directly using the simulation output along with the wave 1 uncertainty quantification.

5.8 Hierarchical Emulation Exploiting Known Simulator Behaviour Wave 2

At each iteration of the decision support procedure it is necessary to re-fit the utility function emulator using the simulations evaluated over the wave 2 design constructed in Section 5.7.2 which is defined and valid for the substantially smaller wave 1 non-implausible decision space. A wave 2 hierarchical emulator which exploits known simulator behaviour is constructed following the methodology in Section 3.3 in the same manner as in wave 1, again utilising the EGES linear model constructed in Section 5.2 to predict the expected NPV. This process is sequentially examined by first decomposing the NPV formula for each Jade model into its constituent parts with emulators fitted that account for the known behavioural structure in Section 5.8.1. For each individual model, these are combined to form an emulator for the approximate NPV in Section 5.8.2 and then linked to the exact NPV in Section 5.8.3. The EGES linear model is then used to obtain an emulator for the utility function in Section 5.8.4.

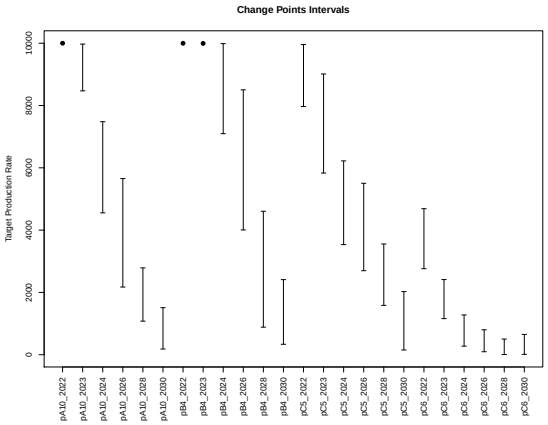
5.8.1 Structured Emulators Exploiting Known Simulator Behaviour for NPV Constituents

The first step is to decompose the expected NPV utility function into contributions from oil production, water production and injection by well and control interval, for which it is known that WOPT and WWIT outputs within a control interval follow the same structured behavioural form of slopes and plateaus as in wave 1. This will be exploited to construct accurate emulators. This structure is not observed for WWPT within a control interval.

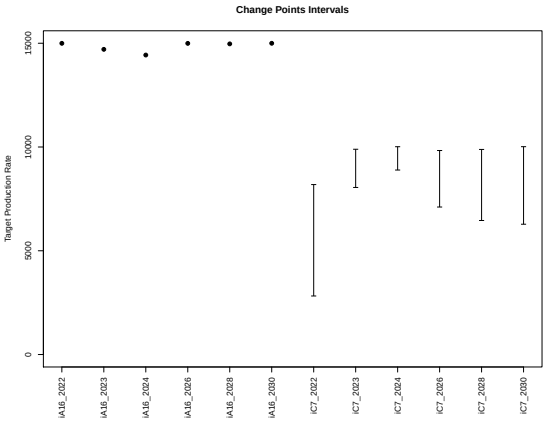
Structured emulation for each Jade model is performed in the same manner as in wave 1. Estimates of the change point upper bounds using Equation (3.3.1) with $\delta_{i,u} = 10$ and the extrapolation cut-offs using Equation (3.3.2) with $\delta_{i,l} = 10$ given the wave 2 simulations for each WOPT and WWIT by well and control interval

are displayed in Figure 5.25. The plots highlight the region in which the "true" change point is believed to be situated. A comparison with the wave 1 estimates in Figure 5.11 shows that this interval between the extrapolation cut-off and change point upper bound is generally tighter and is always contained by the wave 1 interval. Moreover, there are a greater number of instances where the two estimates coincide, particularly for the WWIT constituents. This is a consequence of the refocussing on a smaller non-implausible region in which the target production and injection rates are generally specified to be of moderate-to-high or moderate-to-low values respectively. The result is a reduction in the uncertainty surrounding the "true" change point location due to fewer instances of large variation between parameters of the same type for different wells.

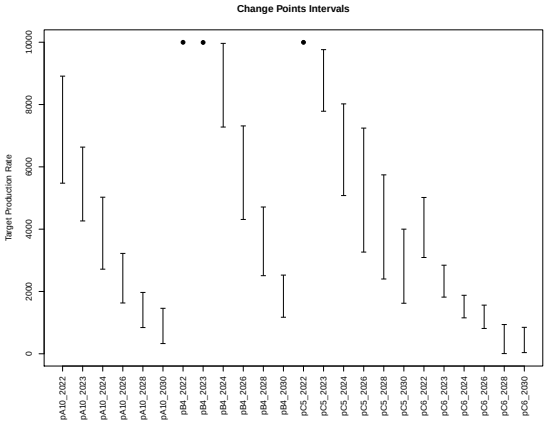
Preliminary Bayes linear emulation proceeds for each WOPT and WWIT constituent, represented by $f_i(\mathbf{d})$, as described in Section 5.4.1. Note that for constituents where $c^u_{jk,t_i} = b^{cp^l}_{jk,t_i}$, this stage is skipped and the output is directly collapsed onto the slope given the corresponding decision parameter value and length of the control interval. In all other cases, a Bayes linear emulator of the structure shown in Equation (2.4.1) is fitted given $\mathcal{D}'_i = \{\mathbf{d} \mid \mathbf{d} \in \mathcal{D},\ d_{jk,t_i} > c^u_{jk,t_i}\}$ and $\mathbf{F}'_i = \{f_i(\mathbf{d}) \mid \mathbf{d} \in \mathcal{D}'_i\}$, with A_i consisting of all decisions which take place in the past of the output. The deterministic functions consist of either a constant or the form presented in Equation (5.4.1) depending on $|A_i|$. The remainder of the preliminary emulator specification is as discussed for wave 1. In summary: $\boldsymbol{\mu}_\beta = \mathbf{0}$; $\mathrm{Var}[\boldsymbol{\beta}] = \infty$; $\mathbb{E}[u_i(\mathbf{d}_{A_i})] = 0$; and $\mathbb{E}[w_i(\mathbf{d})] = 0$; whilst $u_i(\cdot)$ possesses a squared exponential covariance structure defined in Equation (2.4.3) with hyperparameters specified by the substitution approach. These are $\sigma^2_{u_i} = (1-\rho)\sigma^2_{i,lm}$ and $\sigma^2_{w_i} = \rho\sigma^2_{i,lm}$, where $\rho = 0.05$, whilst $\sigma_{i,lm}$ is obtained from a linear model of the relevant form. The correlation length hyperparameter is stipulated to be $\theta = 1$, noting that all decision parameters are transformed to $[-1, 1]$. The final step is to classify points as either in the slope, plateau, or close to the change point using the two-sided truncation method presented in Section 3.3.2.



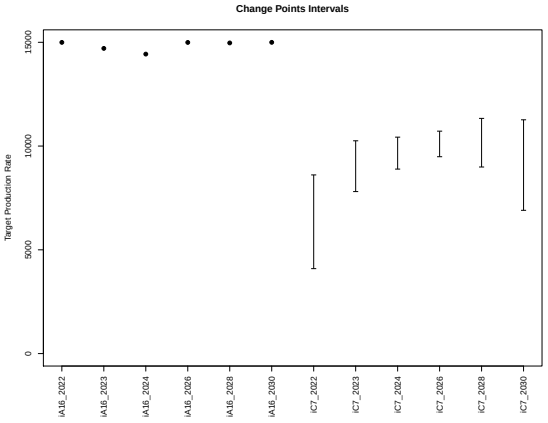
(a) Jade 13 WOPT



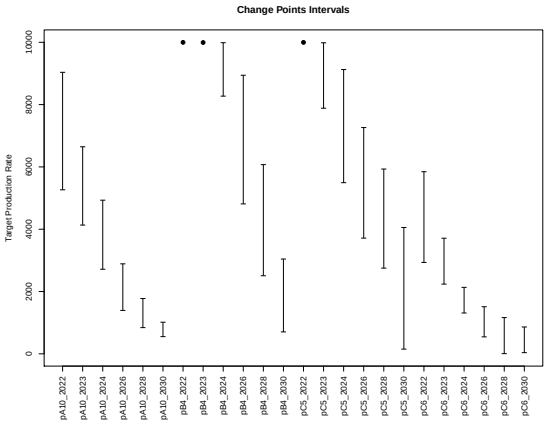
(b) Jade 13 WWIT



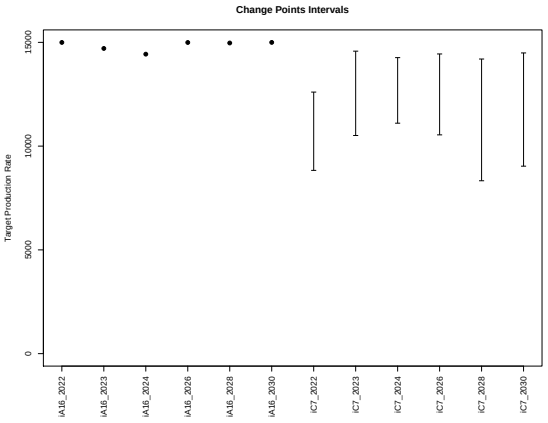
(c) Jade 30 WOPT



(d) Jade 30 WWIT



(e) Jade 33 WOPT

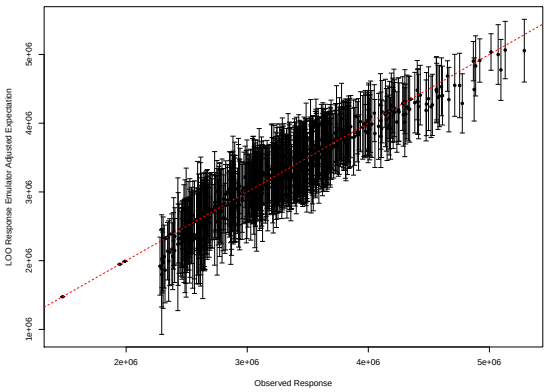


(f) Jade 33 WWIT

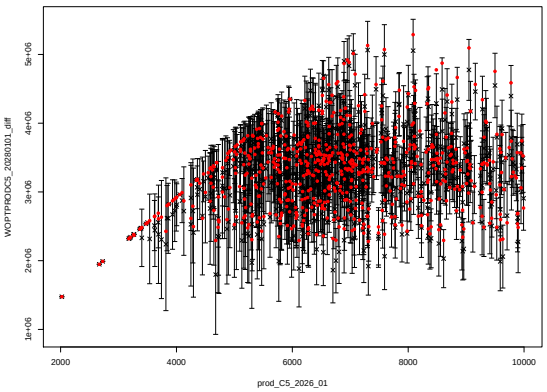
Figure 5.25: Jade wave 2 change point upper bound and extrapolation cut-off intervals for WOPT and WWIT within each control interval with respect to their corresponding decision parameter for each of the three subsampled Jade models. Black points correspond to where the change point upper bound and extrapolation cut-off coincide for a particular NPV constituent and occurs for the maximum value of the target rate.

Leave-one-out diagnostics plots are illustrated for the same Jade 30 NPV constituents as for wave 1 in Figure 5.26 in order to portray the development from waves 1 (see Figure 5.12) to 2. For a direct comparison, note that the range of the simulated expected NPVs in Figures 5.26a and 5.26c are smaller and shifted to higher values than for the corresponding wave 1 plots in Figures 5.12a and 5.12c respectively. Similarly, note that the decision parameter range is also restricted in Figures 5.26b and 5.26d versus Figures 5.12b and 5.12d respectively. This is a result of the iterative refocussing to a smaller region of the decision parameter space which is anticipated to yield higher expected NPVs. For all plots in Figure 5.26 it is observed that the credible intervals are narrower representing a reduction in the wave 2 emulator uncertainty versus wave 1. Moreover, for WOPTPRODC5_2028_01 the remaining space mainly consists of the plateau region, whilst for WOPTPRODC6_2024_01 there is also a reduction in the number of points classified as on-slope with small expected NPVs. Another distinguishing feature between waves 1 and 2 for WOPTPRODC6_2024_01 is observed in a comparison of Figures 5.12d and 5.26d; the wave 2 emulator is more accurately able to capture the shape of the behaviour and reveals a peak utility function for prod_C6_2023_01 in the range of $[2500, 4000]\text{m}^3/\text{day}$ followed by a gradual decay. Such observations demonstrate the power of emulators to provide insight into the behaviour of the response with respect to the decision parameters that would otherwise be unattainable through direct use of the simulator alone.

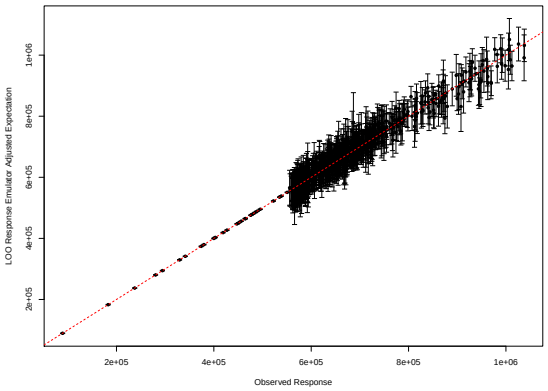
The WWPT within a control interval constituents are emulated separately using a Bayes linear emulator following the same approach as for the preliminary Bayes linear emulator with deterministic functions of the form specified in Equation (5.4.1). The rest of the prior specification is obtained via linear models fitted using all simulations in $\mathcal{D}$.



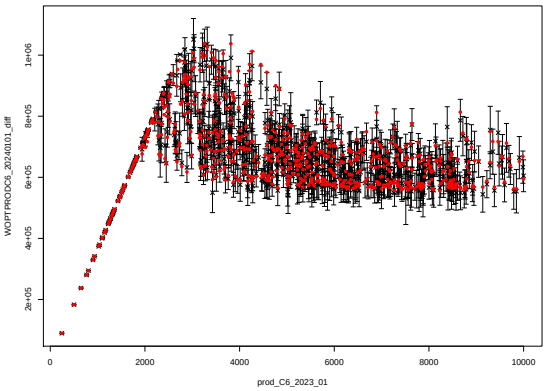
(a) WOPTPRODC5_2028_01 emulator CI versus simulator.



(b) WOPTPRODC5_2028_01 emulator CI versus prod_C5_2026_01.



(c) WOPTPRODC6_2024_01 emulator CI versus simulated output.



(d) WOPTPRODC6_2024_01 emulator CI versus prod_C6_2023_01.

Figure 5.26: Leave-one-out diagnostic plots for the wave 2 structured emulation of Jade 30 WOPTPRODC5_2028_01 (top) and WOPTPRODC6_2024_01 (bottom). The left column shows the adjusted expectation with credible intervals (CI) of width 3 adjusted standard deviations error bars versus the simulated value where the red dashed line denotes equality of the structured emulator and simulator. The right column shows the adjusted expectation with credible intervals of width 3 adjusted standard deviations error bars versus the output's corresponding target production rate with red points denoting the simulated values.

5.8.2 Emulation of the Approximate NPV

For each Jade model the emulators for the NPV constituents are first combined to obtain an emulator for the average discounting approximate NPV of the form shown in Equations (3.3.5) and (3.3.6) following the methodology in Section 3.3.3. This

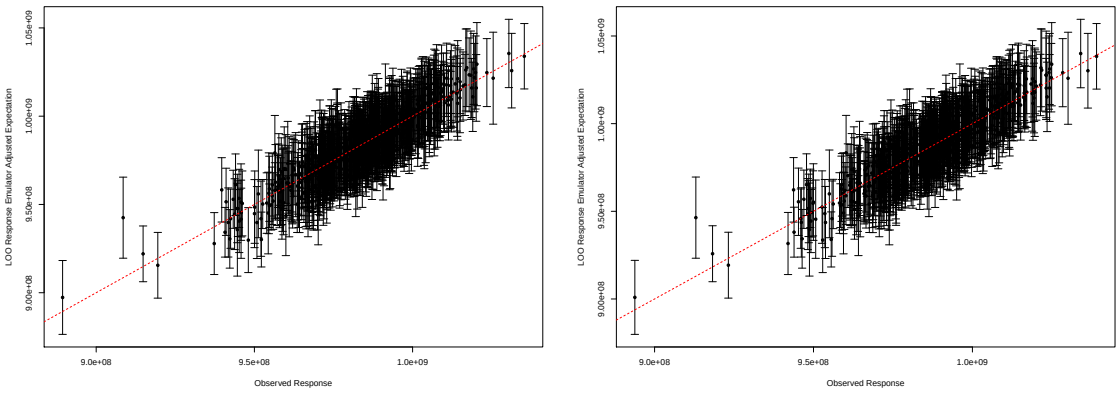
involves the same process as for wave 1 where the six control intervals are formed by merging multiple consecutive discounting intervals. The adjusted expectation and variance formulae are displayed in Equations (3.3.7) and (3.3.8) respectively.

A comparison of the leave-one-out diagnostics for the approximate NPV emulator for Jade 30 for wave 2 in Figures 5.27a and 5.27c with wave 1 in Figures 5.13a and 5.13c respectively highlights similar features as for the NPV constituents. The results in Figure 5.27a concentrates on a much smaller region corresponding to the top-right corner of Figure 5.13a for which it is observed that there is a reduction in the uncertainty. Comparing Figure 5.27c with Figure 5.13c we note that there is a reduction in the magnitude of the standardised residuals indicating a better emulator fit. Moreover, for wave 2 there is no evidence against homoscedasticity as well as no visible pattern, unlike in wave 1 where the pattern was caused by the enormous variation in the approximate NPV over the decision space. This has been rectified by focusing only on the non-implausible region demonstrating the benefit of an iterative approach to emulation and decision support.

5.8.3 Linking the Exact and Approximate NPV

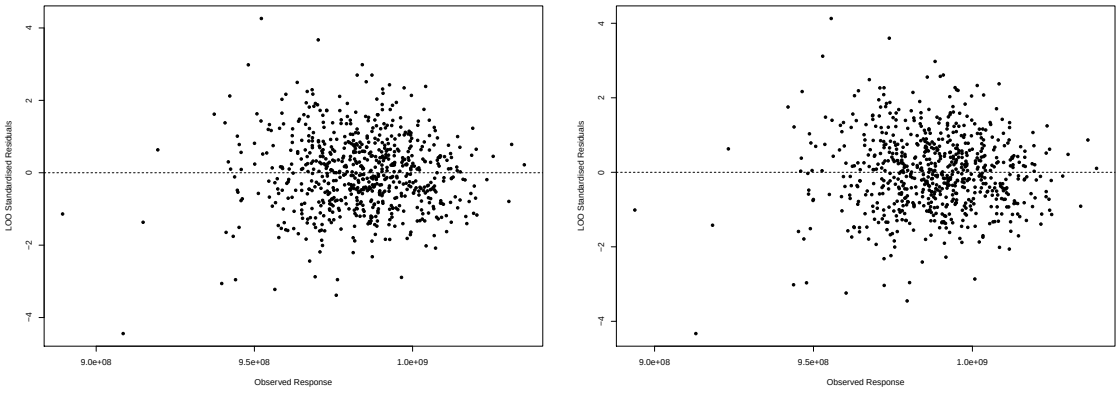
The exact NPV defined in Section 3.3.3 is emulated using a simple linear regression model on the approximate NPV described in Section 3.3.4 to handle the uncertainty due to amalgamating discounting intervals. The adjusted expectation and variance are computed using Equations (3.3.12) and (3.3.13) respectively where for each Jade model, $\hat{\beta}_{0,\widetilde{\text{NPV}}_j}, \hat{\beta}_{1,\widetilde{\text{NPV}}_j}$ and $\sigma_{\widetilde{\text{NPV}}_j}$ are estimated using the wave 2 simulation data.

Leave-one-out diagnostics plots for the Jade 30 NPV are shown in Figures 5.27b and 5.27d. The results are similar to those for the average discounting approximate NPV in Figures 5.27a and 5.27c respectively showing how little additional uncertainty is introduced by the link model. A comparison with the wave 1 emulation of the Jade 30 NPV in Figures 5.13b and 5.13d leads to the same remarks as for the approximate NPV in Section 5.8.2.



(a) Average discounting approximate NPV credible intervals versus simulated values.

(b) Exact NPV credible intervals versus simulated values.



(c) Average discounting approximate NPV standardised residuals versus simulated values.

(d) Exact NPV standardised residuals versus simulated values.

Figure 5.27: Leave-one-out diagnostics for wave 2 emulators of the Jade 30 average discounting approximate NPV (left column) and the exact NPV (right column) via a simple linear model on the emulated approximate NPV. The top row shows the emulator adjusted expectation with 3 adjusted standard deviation credible intervals versus simulated values where the red dashed line denotes when the emulator and simulator coincide. The bottom row shows the emulator standardised residuals versus the simulated values.

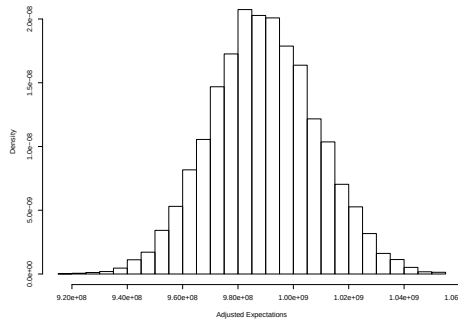
5.8.4 Emulation of the Ensemble Mean NPV

The final stage is to construct an emulator for the Jade ensemble mean NPV using the EGES linear model formulated in Section 5.2 to combine the wave 2 emulators for the Jade 13, 30 & 33 NPV. This follows the methodology described in Sec-

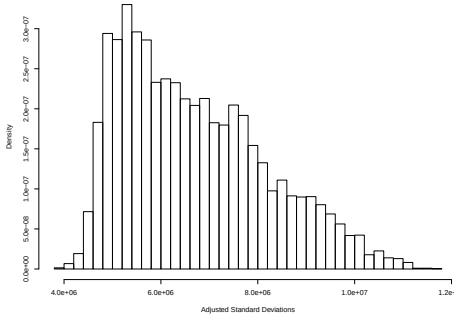
tion 3.3.5 with the adjusted expectation and variance computed using the formulae in Equations (3.3.16) and (3.3.17) respectively.

As was the case in wave 1, true leave-one-out diagnostics cannot be performed because this would require simulations for the entire ensemble. The use of the Jade ensemble mean NPV is justified since the leave-one-out emulator diagnostics for each sub-selected Jade model NPV show no problems, whilst the EGES linear model diagnostics also provide no evidence against a well fitted hierarchical emulator. Histograms in Figures 5.28a and 5.28b show the adjusted expectations and standard deviations respectively over the wave 1 non-implausible decision space. A comparison with Figures 5.14a and 5.14b respectively for wave 1 shows an increase in the adjusted expectations to new higher values as well as a general reduction in the uncertainty along with a shift in the distribution to positive skew. Both features are beneficial from the perspective of decision support.

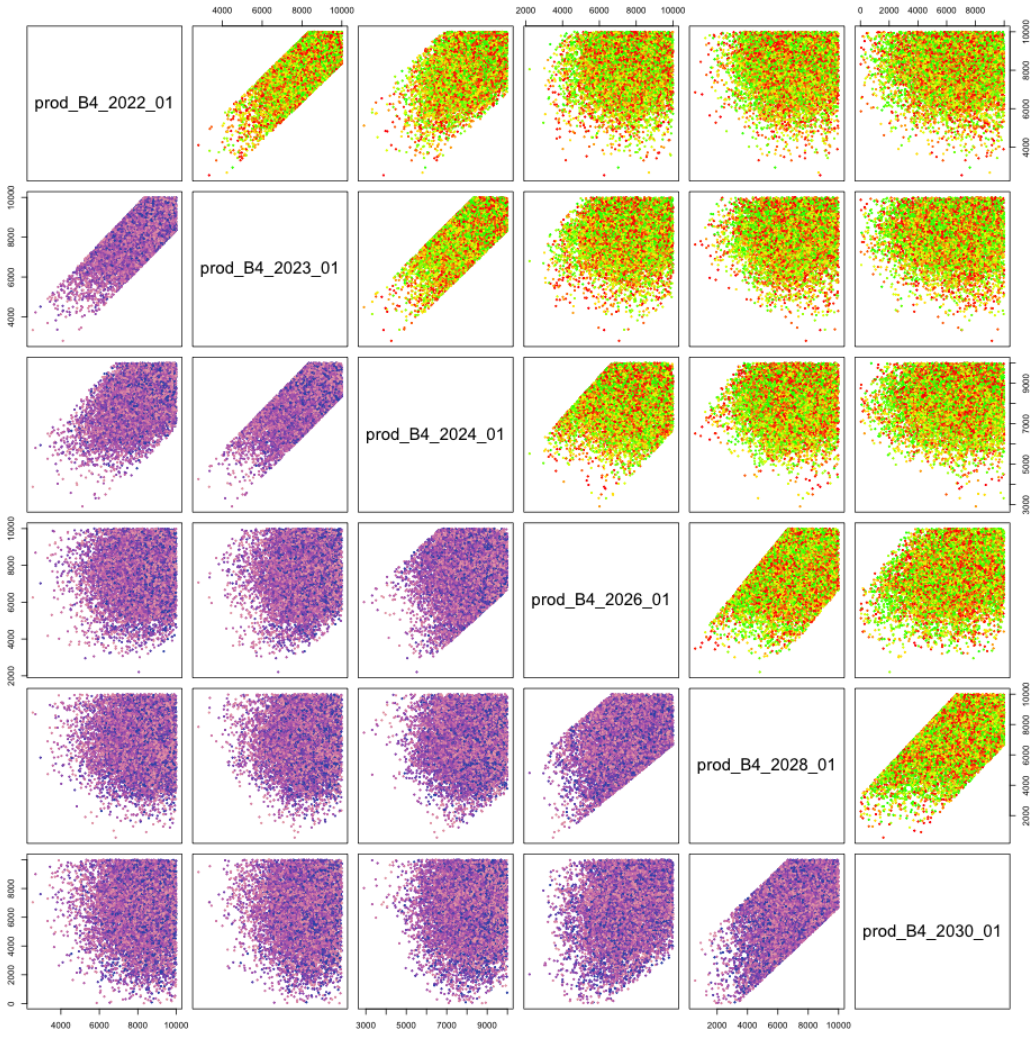
Figure 5.28c shows a pairs plot evaluated at wave 1 non-implausible decision parameters associated with producer well B4. The above diagonal plots are coloured by the emulator adjusted expectation where green, yellow and red correspond to high, moderate and low values respectively. Unlike in wave 1 (see Figure 5.14c) the association is less apparent between the control values and high ensemble mean NPVs. This is a consequence of producer well B4 being identified as the most dominant for determining the utility function value. The wave 1 analysis has already focused in on the region of good producer well B4 decision parameter settings, hence these induce much less variability in the ensemble mean NPV over the remaining non-implausible space. In contrast, an assessment of the corresponding pairs plots of the decision parameters for the other wells now illustrate stronger relationships and are thus more active in determining the utility function value. An example is shown in Figure 5.29 for producer well C5. The below diagonal plots are coloured by the emulator adjusted standard deviation where light pink, purple and blue correspond to low, moderate and high values respectively. A similar pattern between control values and the uncertainty is observed as for the emulator adjusted expectation.



(a) Histogram of the adjusted expectations.



(b) Histogram of the adjusted standard deviations.



(c) Hierarchical emulator for the Jade ensemble mean NPV pairs plot for producer well B4. The plots above the diagonal are coloured by the adjusted expectation with green, yellow and red corresponding to high, moderate and low values respectively. The plots below the diagonal are coloured by the adjusted standard deviation with light pink, purple and blue corresponding to low, moderate and high values respectively.

Figure 5.28: Jade ensemble mean NPV wave 2 hierarchical emulator output plots.

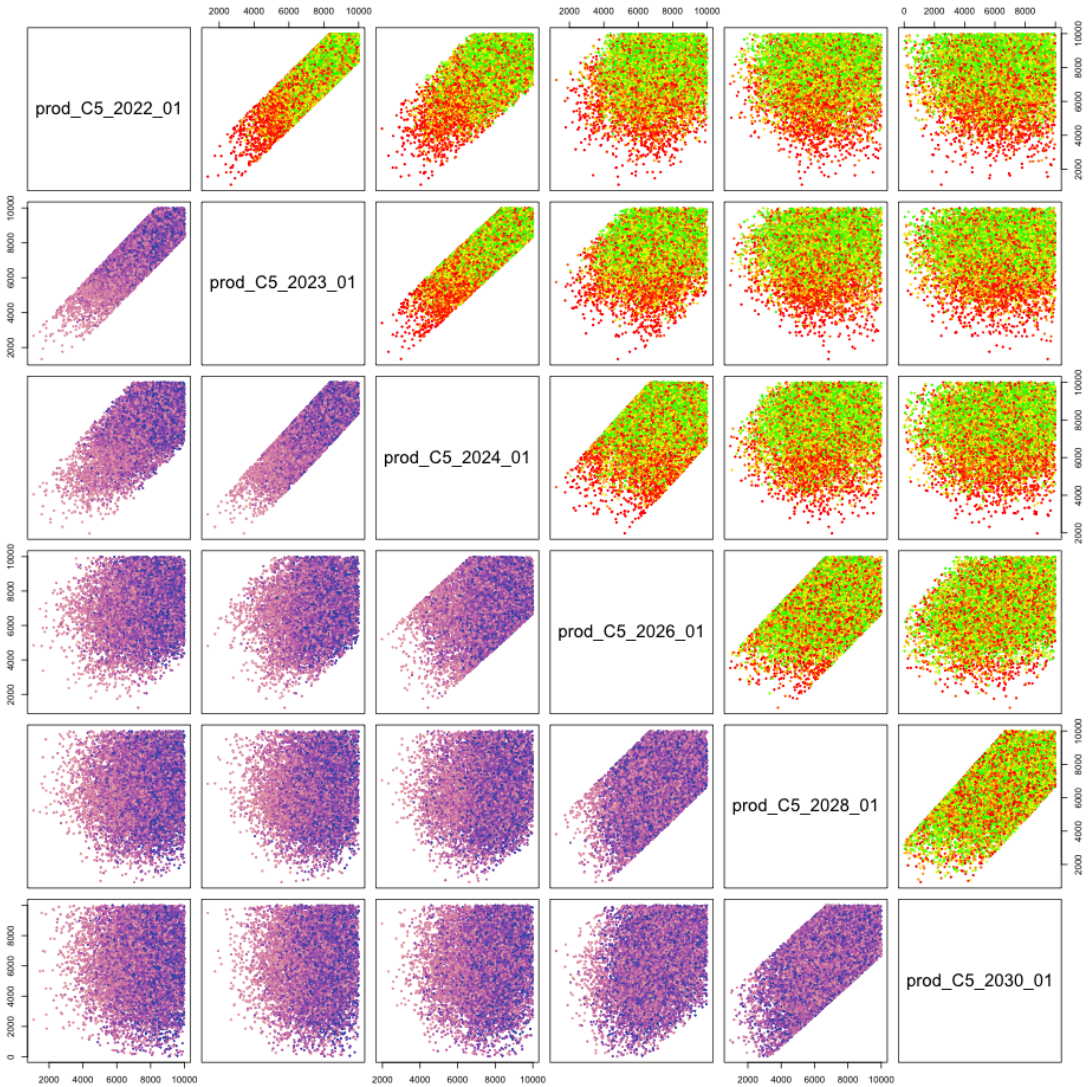


Figure 5.29: Jade ensemble mean NPV wave 2 hierarchical emulator pairs plot of the decision parameters for producer well C5. The plots above the diagonal are coloured by the adjusted expectation with green, yellow and red corresponding to high, moderate and low values respectively. The plots below the diagonal are coloured by the adjusted standard deviation with light pink, purple and blue corresponding to low, moderate and high values respectively.

Note that both colour schemes are defined with respect to the emulator output and are not directly comparable with the wave 1 colours. The process of fitting and evaluation of the hierarchical emulator for the ensemble mean NPV achieved similar performance in terms of computational efficiency to wave 1 delivering large savings versus direct Jade model simulations.

5.9 Wave 2 Uncertainty Analysis for Well Control Optimisation

Iterative decision support allows for a reassessment of the uncertainties at each wave. In this section we update the uncertainty quantification over the wave 1 non-implausible decision parameter space using the same uncorrelated additive error structure to link between the Jade model and the corresponding real world physical system, as in Section 5.5.1. Geological and the efficient geological ensemble subsampling uncertainties remain unchanged from wave 1 for the reasons discussed in Sections 5.9.1 and 5.9.2. All other sources are revised including the: NPV cost parameter uncertainty in Section 5.9.3; structural model discrepancy quantified in Section 5.9.4; and decision implementation error in Section 5.9.5. A summary of the waves 1 and 2 uncertainty analyses is presented in Section 5.9.6. This analysis makes use of the techniques described in Sections 2.7 and 3.5.

5.9.1 Geological Uncertainty

The wave 2 assessment of geological uncertainty is the same as in wave 1 (see Section 5.5.2 for details) where it is assumed that $\mathbb{E}[\varepsilon_{\mathrm{G}}] = 0$ and $\mathrm{Var}[\varepsilon_{\mathrm{G}}] \approx \frac{s^2}{N} = \frac{s^2}{50} = 1.555 \times 10^{14}$. This is as the assessment of the geological uncertainty requires simulations for the entire ensemble for which only the exploratory set exists. If it was deemed necessary, then additional simulations for all 50 Jade models may be performed for decision parameter vectors in the wave 1 non-implausible space with the calculations repeated over this region of the decision space.

5.9.2 Efficient Geological Ensemble Subsampling Uncertainty

The EGES linear model fitted in Section 5.2 and subsequently used within the hierarchical emulator structure in Sections 5.4.4 and 5.8.4 was formulated based on an

analysis of the exploratory simulations using all 50 Jade models. Consequently it is not possible to revise the uncertainty quantification at wave 2 without further simulations. It is expected that the EGES uncertainty will reduce between waves as the volume of the non-implausible decision space reduces. Due to constraints on computational resources, it is impractical to re-evaluate this uncertainty. In addition, since the EGES uncertainty represented only a small proportion of the total uncertainty and was of a substantially smaller magnitude than the emulator uncertainty, it is deemed unnecessary to expend computational resources on its reassessment. The wave 2 analysis uses the wave 1 setup: $\mathbb{E}[\varepsilon_{\text{EGES}}] = 0$ and $\text{Var}[\varepsilon_{\text{EGES}}] \approx 5.046 \times 10^{12}$.

5.9.3 NPV Cost Parameter Uncertainty

The NPV cost parameter uncertainty is examined over the wave 1 decision support non-implausible region following the methodology presented for utility function uncertainty quantification detailed in Section 3.5.2. For consistency, the same collection of $n_c = 1000$ simulated cost parameter scenarios are used as in Section 5.5.4 with the ensemble mean NPV evaluated via the EGES linear model for $\mathbf{d} \in \Omega_1$. As in wave 1 results are presented both before and after the removal of the linear transformation effect.

NPV Cost Parameter Uncertainty Before the Removal of Linear Transformation Effects

The wave 2 uncertainty induced by NPV cost parameter uncertainty is illustrated in Figure 5.31a which shows the NPV cost parameter uncertainty versus the predicted ensemble mean NPV over the wave 1 non-implausible space. There is evidence of positive correlation and hence a robust assessment of the NPV cost parameter uncertainty is the maximum variance for $\mathbf{d} \in \Omega_1$; $\text{Var}[\varepsilon_{\text{C}}] = 7.725 \times 10^{16}$. This is marginally smaller than in wave 1, however remains the dominant source of uncertainty with large implications for decision support due to difficulties in distinguishing between the ensemble mean NPV for various decision parameter vectors.

NPV Cost Parameter Uncertainty After the Removal of Linear Transformation Effects

The linear transformation effects are removed using the procedure described in Section 3.5.2 with $\mathbf{c}_0$ equal to the suggested fixed cost parameter settings listed in Table 5.2. The residual variability, estimated by $s^2_{\mathrm{C}}(\mathbf{d}_j)$ (see Equation (3.5.3)), is illustrated with a projection pairs plot for producer well B4 in Figure 5.30 where there is no evidence of association between the standard deviations and decision parameter values. Figure 5.31b shows the standard deviations due to NPV cost parameter uncertainty versus the predicted ensemble mean NPV which initially exhibits decreasing uncertainty for larger NPVs before a slight increase. This is indicative of some non-linearity in the utility function due to changes in the cost parameters which occurs for the very highest values and may be explained by the subtle differences in the well control strategies required to achieve these such as possessing high decision parameter values for all control intervals, rather than only the first few prior to the heavy discounting coming into full effect. An investigation of the mean residuals over the cost parameter scenarios using $m_{\mathrm{C}}(\mathbf{d}_j)$ (see Equation (3.5.2)) again shows these are at least an order of magnitude smaller than $s_{\mathrm{C}}(\mathbf{d}_j)$, and thus have negligible effect on the decision analysis.

Figure 5.32 shows a histogram of the wave 2 variances of the utility function induced by NPV cost parameter uncertainty with the distribution centred at 3.296×10^{13}. A comparison with the results of wave 1 in Figure 5.19 highlights a general shift of the entire uncertainty distribution to the left, with a higher density placed on smaller uncertainties. It is also noted that the NPV cost parameter uncertainty remains larger than the EGES uncertainty. Given these observations and those for Figure 5.31b, $\mathrm{Var}[\varepsilon_{\mathrm{C}}]$ is estimated by the median over $\mathbf{d} \in \Omega_1$: $\mathrm{Var}[\varepsilon_{\mathrm{C}}] = 1.731 \times 10^{13}$. This represents a 73.2% decrease compared with wave 1 due to refocusing on a smaller non-implausible region of the decision space. Moreover, there is a three order of magnitude reduction versus prior to removing the linear transformation effects where $\mathrm{Var}[\varepsilon_{\mathrm{C}}] = 7.725 \times 10^{16}$ (note that the median is 6.877×10^{16}). In addition, note that

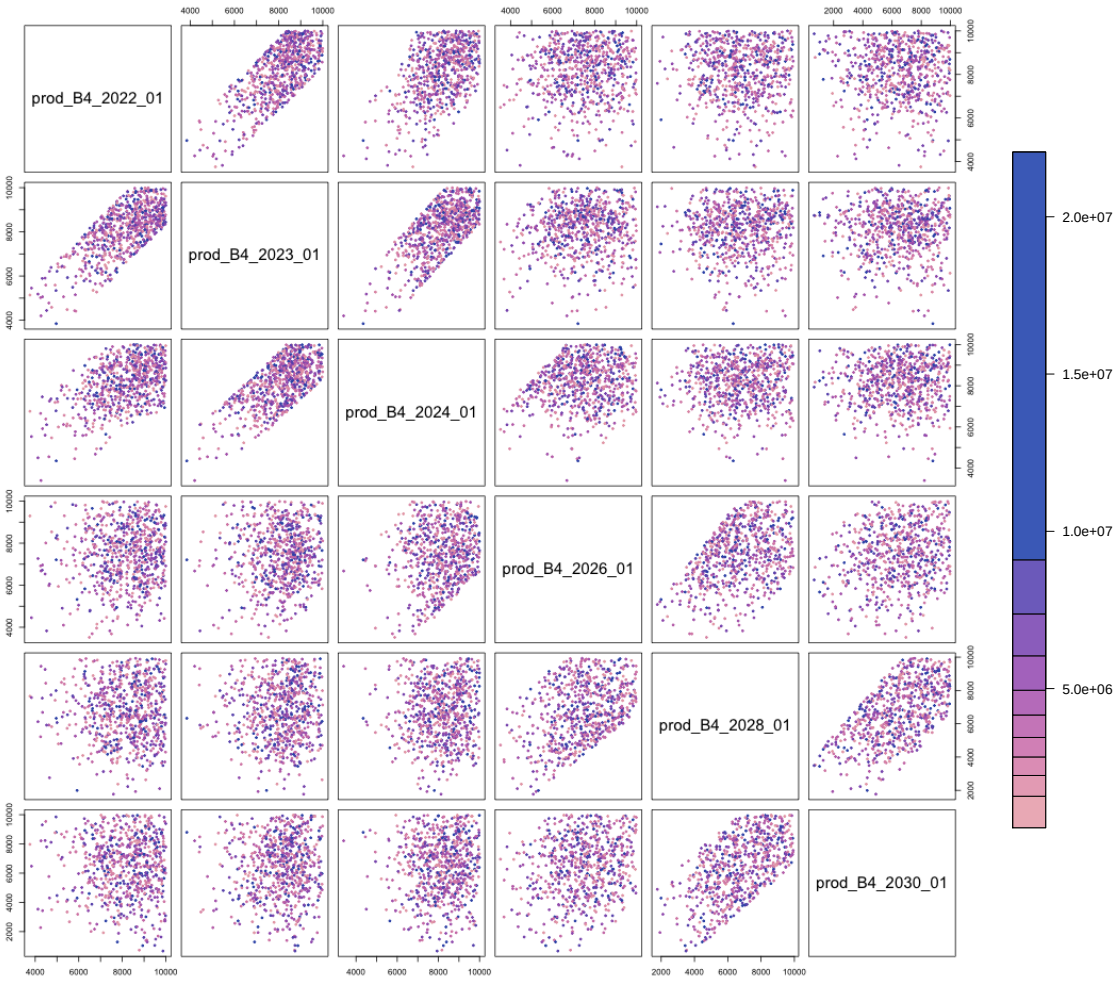
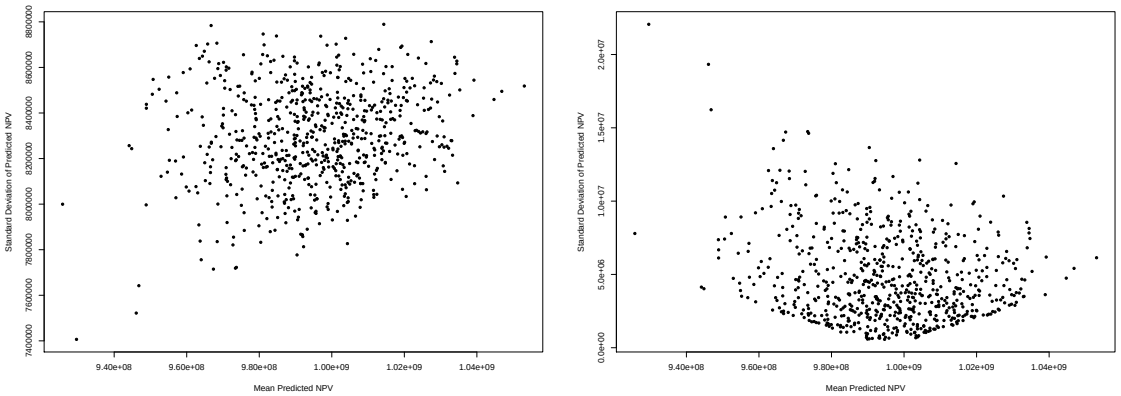


Figure 5.30: Jade wave 2 uncertainty quantification two-dimensional projections of the decision space for producer well B4 coloured by the standard deviations of the ensemble mean NPV utility function due to the cost parameter variability after the removal of linear effects respectively. The colour scheme denotes the magnitude of the uncertainty on a relative scale with light pink through to dark purple corresponding to low through to high standard deviations.

the maximum uncertainty after the removal of the linear transformation effects is 4.867×10^{14}, a decrease of two orders of magnitude. As was the case for wave 1, this will have major benefits for wave 2 decision support by enabling the effective comparison of the emulator (or simulator) output for different decision parameter vectors in an analysis which is robust to this uncertainty.



(a) Before the removal of linear effects. (b) After the removal of linear effects.

Figure 5.31: Jade wave 2 ensemble mean NPV standard deviation induced by NPV cost parameter variability versus the ensemble mean NPV over the wave 2 design computed using the $n_c = 1000$ simulated cost parameter scenarios from wave 1.

5.9.4 Structural Model Discrepancy

Structural model discrepancy forms an integral part of a comprehensive uncertainty analysis. With the caveat that the Jade model is artificial, model discrepancy is represented as a proportion of the simulated ensemble mean NPV with the intuition discussed in Section 5.5.5. A conservative estimate is obtained by considering the maximum simulated Jade ensemble mean NPV within the current non-implausible region over both waves of simulations. For consistency with the wave 1 analysis, the proportion is stipulated to be 0.1% as representative of a variation of one standard deviation due to model discrepancy. This assumes: the ensemble mean NPV is a good representative for the expected NPV; and the specified decisions are executed without error with this potential uncertainty treated in Section 5.9.5. Note that the uncertainty due to prediction of the Jade ensemble mean NPV via the EGES linear model is handled in Section 5.9.2. We specify that $\mathbb{E}[\varepsilon_{\text{MD}}] = 0$ whilst the wave 2 assessment of the model discrepancy yields $\text{Var}[\varepsilon_{\text{MD}}] = 1.109 \times 10^{12}$. This is very similar to the wave 1 assessment. Furthermore, model discrepancy remains much smaller than both the geological or NPV cost parameter uncertainties.

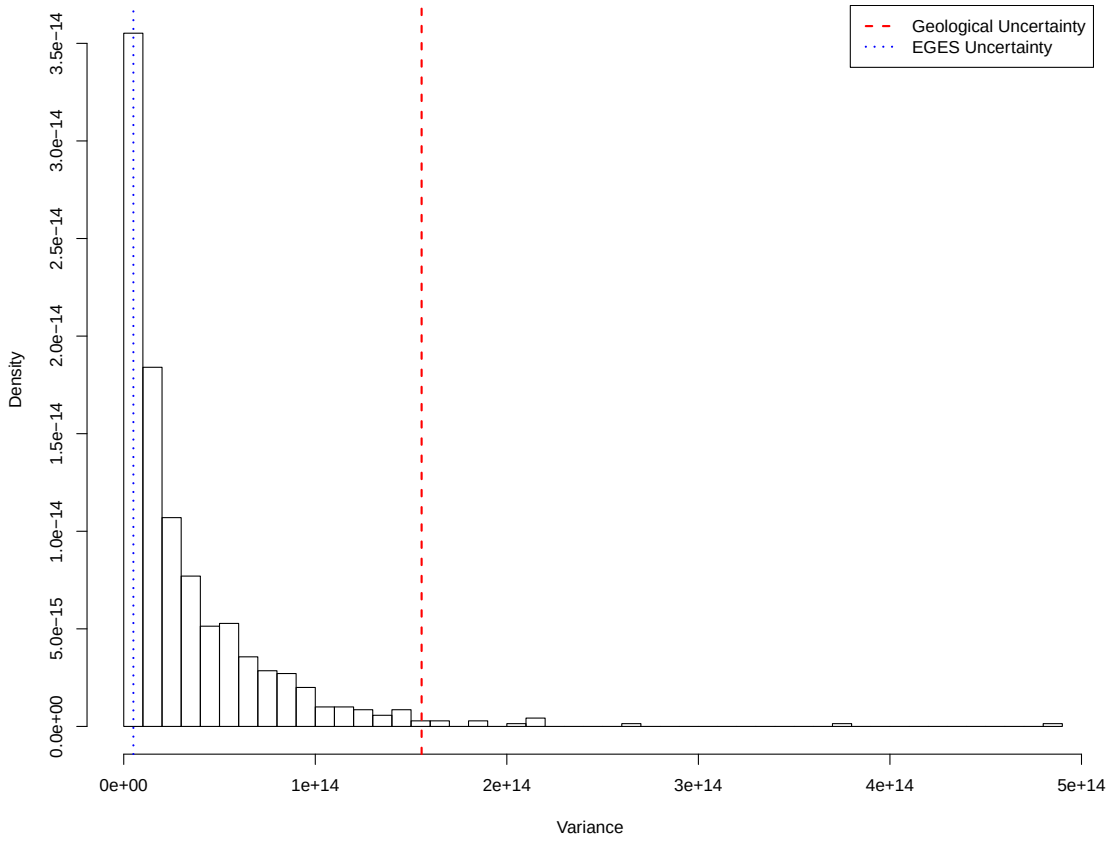


Figure 5.32: Histogram of the wave 2 variances of the Jade ensemble mean NPV over the generated cost parameter scenarios after the linear effect is removed. For comparison, the red dashed and blue dotted vertical lines correspond to the assessed values for $\mathrm{Var}[\varepsilon_{\mathrm{G}}]$ and $\mathrm{Var}[\varepsilon_{\mathrm{EGES}}]$ respectively.

5.9.5 Decision Implementation Error

Decision implementation error due to deviations from the stipulated well controls is addressed in the wave 2 analysis using the methodology presented in Section 3.5.4 along with the hierarchical emulator constructed in Section 5.8. As in wave 1, the first step is to generate a 200 point space-filling design over the wave 1 non-implausible region with 100 decision parameter vectors sampled according to the same perturbation distribution as proposed in Section 5.5.6. Note that it is possible that some of these decision parameter vectors may be outside the wave 1 non-implausible region, although this is representative of the real world scenario where such constraints may not be properly adhered to. Moreover, the final decision

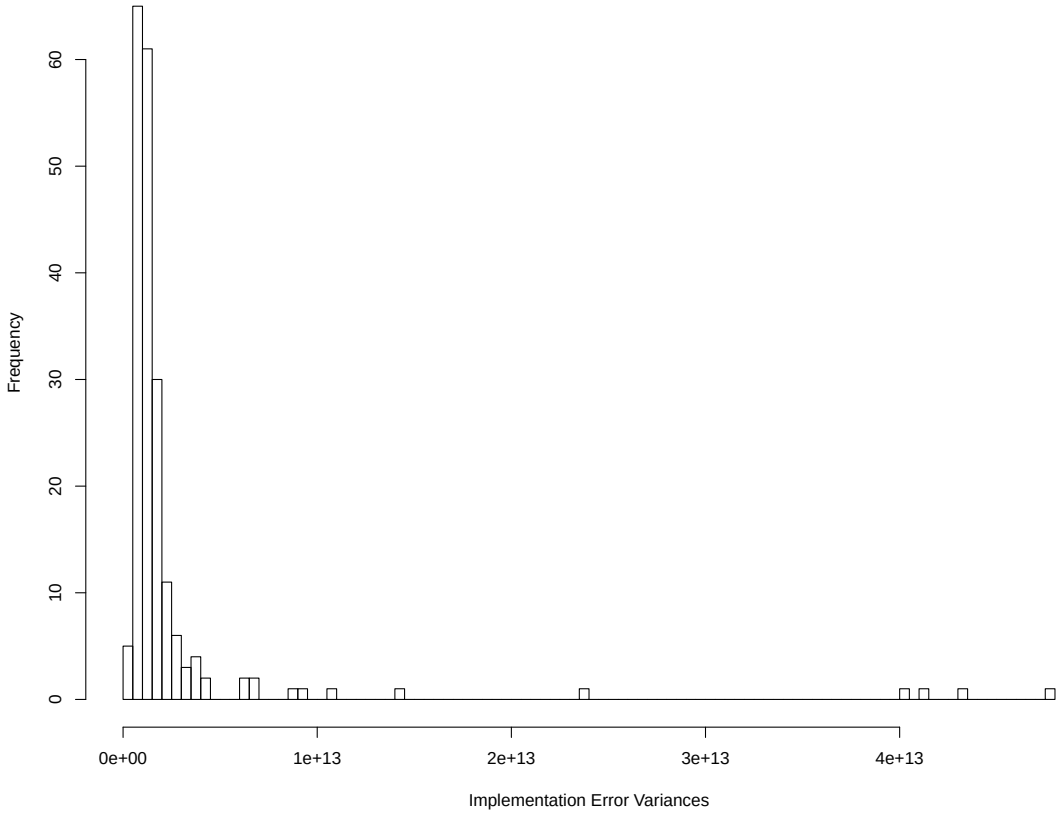


Figure 5.33: Histogram of the wave 2 variances of the Jade ensemble mean NPV due to decision implementation error.

support region may be obtained using stricter threshold values, and hence any such cases will still lie within the more conservative region. An assessment of the mean emulator adjusted expectation for each fixed decision parameter location provides no evidence against the assumption that $\mathbb{E}[\varepsilon_{\mathrm{I}}(\mathbf{d})] = 0$ for all $\mathbf{d} \in \Omega_1$. A conservative assessment of the variance due to decision implementation error as the 95^{th} percentile yields $\mathrm{Var}[\varepsilon_{\mathrm{I}}] = 6.464 \times 10^{11}$ and is justified by the histogram of the variances over the initial 200 sampled decision parameter vectors in Figure 5.33. Compared with wave 1 this constitutes an order of magnitude reduction in the uncertainty.

5.9.6 Uncertainty Analysis Summary

A summary of the wave 1 and wave 2 uncertainty analyses is presented in Table 5.5 including the variances and the proportion of the total uncertainty variances for each wave. For both wave 1 and 2, the individual uncertainties, excluding that due

to the emulator, are combined following the uncorrelated additive error structure described in Equation (3.6.2) to obtain a total uncertainty variance of 2.281×10^{14} and 1.746×10^{14} respectively. Moreover, the EGES uncertainty is encompassed within the hierarchical emulator structure for the utility function. In both waves the geological uncertainty is the largest source of uncertainty, and is especially dominant at wave 2. The re-assessment of the NPV cost parameter uncertainty at wave 2 yields a substantial reduction and hence a decrease in the percentage of the overall uncertainty. Both structural model discrepancy and decision implementation error remain small sources of uncertainty in wave 2 of the analysis.

Source of Uncertainty	Wave 1 Variance	Proportion of Total Wave 1 Uncertainty	Wave 2 Variance	Proportion of Total Wave 2 Uncertainty
Geological Uncertainty	1.555×10^{14}	68.2%	1.555×10^{14}	89.1%
EGES Uncertainty	5.046×10^{12}	–	5.046×10^{12}	–
NPV Cost Parameter Uncertainty	6.465×10^{13}	28.3%	1.731×10^{13}	9.9%
Structural Model Discrepancy	1.076×10^{12}	0.5%	1.109×10^{12}	0.6%
Decision Implementation Error	6.821×10^{12}	3.0%	6.464×10^{11}	0.4%
Total Uncertainty	2.281×10^{14}	–	1.746×10^{14}	–

Table 5.5: Emerson Jade model application wave 1 and 2 uncertainty analysis results including the variances and proportion of total variances for the Emerson Jade model application obtained using an additive error structure described in Equation (2.7.1). Note that the hierarchical emulator construction implicitly encompasses the EGES uncertainty to predict the ensemble mean NPV, hence it is omitted from the total uncertainty.

5.10 Decision Support Wave 2 for Well Control Optimisation

In this section we perform the second wave of iterative decision support for well control optimisation applied to the Emerson Jade model. This incorporates the EGES technique detailed in Section 5.2, as well as the wave 2 design; hierarchical emulator exploiting known simulator behaviour; and revised uncertainty quantification described in Sections 5.7 to 5.9 respectively. Results are presented in Section 5.10.1 with a comparison under alternative uncertainty analyses discussed in Section 5.10.2.

5.10.1 Results

The second wave of iterative decision support is performed using the hierarchical emulator exploiting known simulator behaviour for the utility function devised in Section 5.8 and incorporates the wave 2 uncertainty analysis described in Section 5.9. As in wave 1, the EGES uncertainty is encompassed within the hierarchical emulator formulation. In order to evaluate the decision support implausibility measure it is necessary to recompute $U_{\max}$ where for the second wave $U_{\max} = \max_{\mathbf{d} \in \mathcal{D}_1 \cup \mathcal{D}_2} U(\mathbf{d})$. The result is $U_{\max} = 1.053 \times 10^9$ at wave 2, compared with $U_{\max} = 1.037 \times 10^9$ in wave 1; hence it is a wave 2 simulation that leads to this increase. This illustrates how the refocusing on the wave 1 non-implausible region leads to a higher concentration of simulations using decision parameter vectors that are judged to be more likely to produce higher expected NPVs.

The wave 2 non-implausible region is defined as decision parameter vectors which satisfy $I_{\mathrm{DS}}(\mathbf{d}) \leq c = 3$ resulting in a further decrease to 15.87% of the wave 1 non-implausible region. Moreover, this corresponds to 0.567% of the initial decision space formed by imposing the operational range and difference constraints, whilst also equating to 0.0112% of the full hypercube volume formed by the operational range constraints only. This emphasises how the iterative approach to decision support is capable of sequentially refocusing on smaller regions of the decision space

which contain potentially good control strategies through the exploitation of a higher concentration of simulation output information at each wave.

Implausibility pairs plots are used to visualise the wave 2 decision support non-implausible region with results shown for the decision parameters for producer well B4 and for all decision parameters in the first two control intervals starting in 2022 and 2023 shown in Figures 5.34 and 5.35 respectively. The plot types and colour schemes are the same as in Figure 5.21. The evolution from Figure 5.21 to Figure 5.34 further emphasises the importance of high early and mid-time target production rates, as evidenced by the general shift to higher values observed in the marginal distribution in the histograms and the contraction of the non-implausible region in the below diagonal implausibility plots. Moreover, there exists a more prominent peak for the later time (last three) controls which is situated slightly below the maximum of the parameter range. This implies that to achieve a high expected NPV does not necessarily require maximum target production rates for all time intervals; as explained by the increasing costs due to water production incurred by attempting to sustain such an oil production rate. These observations are mirrored in Figure 5.35 which also illustrates a strong dependency between production targets across wells and time points within the optical density plots. The decision parameters for certain producer wells such as B4 and C5 are more constrained by the non-implausible region and are thus perceived to have a greater influence on achieving a high expected NPV. In contrast, the pattern between target injection rates is less well defined showing the dominance of target production rates in determining the utility function value, although there is a tendency to favour lower injection rates to avoid the associated water costs. It is also noted that the controls for injector well A16 appear to play a more significant role than those for injector C7. Similar developments between waves 1 and 2 are also observed for the other wells and subsets of control intervals across wells, with the plots omitted for space.

Results for a decision support analysis directly using the wave 2 simulation output omitting emulator uncertainty are shown in Figure 5.36. The red points highlight

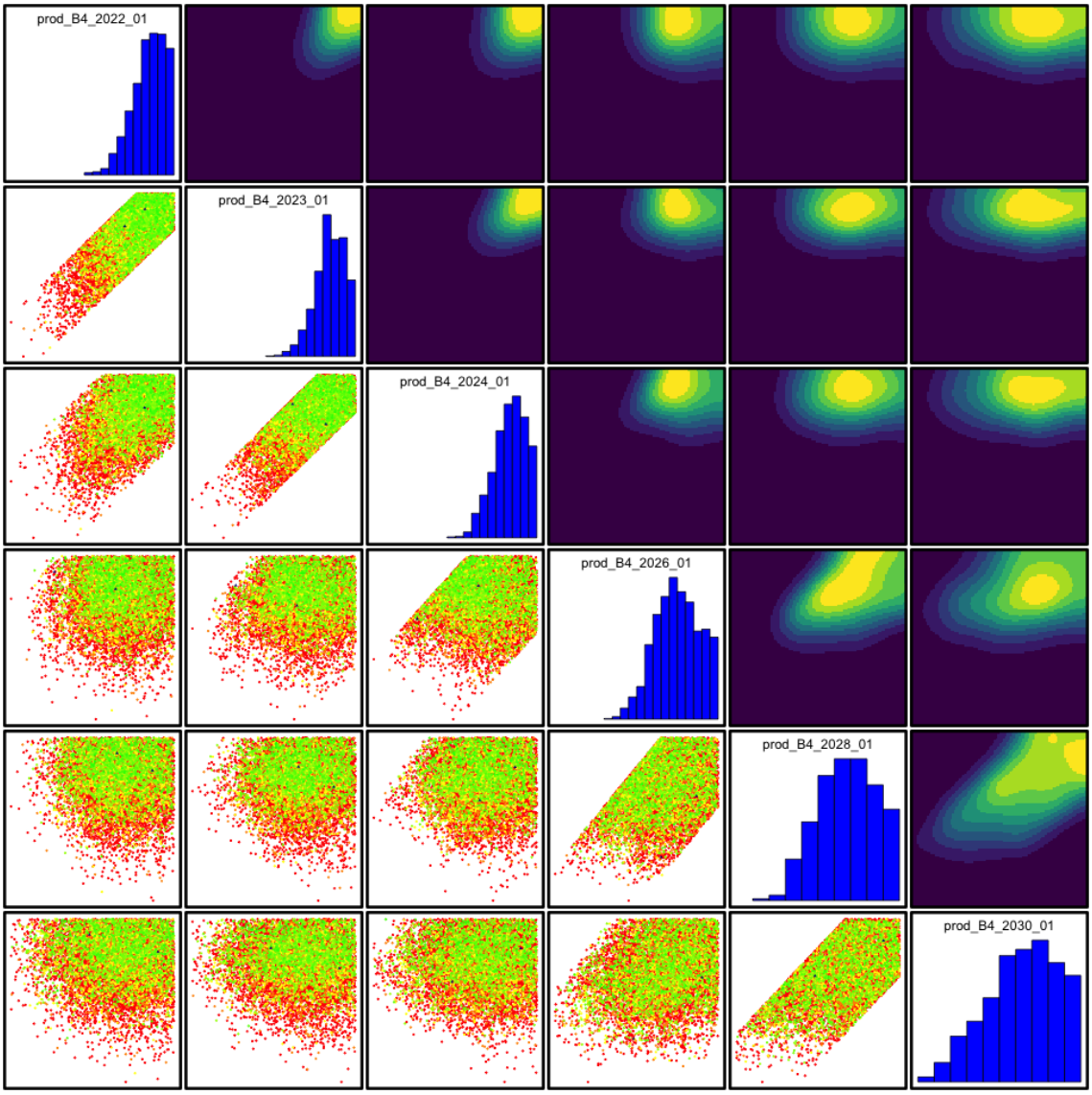


Figure 5.34: Jade wave 2 decision support implausibility measure (below the diagonal) and optical density (above the diagonal) pairs plot, with marginal density histograms (along the diagonal) for the decision parameters for producer well B4. In the implausibility plots, the axes are swapped to match the corresponding optical density plot in which the colours green, yellow and red indicate where $0 < I_{\mathrm{DS}}(\mathbf{d}) \leq 2$, $2 < I_{\mathrm{DS}}(\mathbf{d}) \leq 3.5$ and $I_{\mathrm{DS}}(\mathbf{d}) > 3.5$ respectively, whilst blue triangles denote where $I_{\mathrm{DS}}(\mathbf{d}) = 0$ which occurs when $\mathbb{E}_F[U(\mathbf{d})] \geq U_{\max}$. For the optical density plots, dark blue through to yellow depict low to high density of points satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c$.

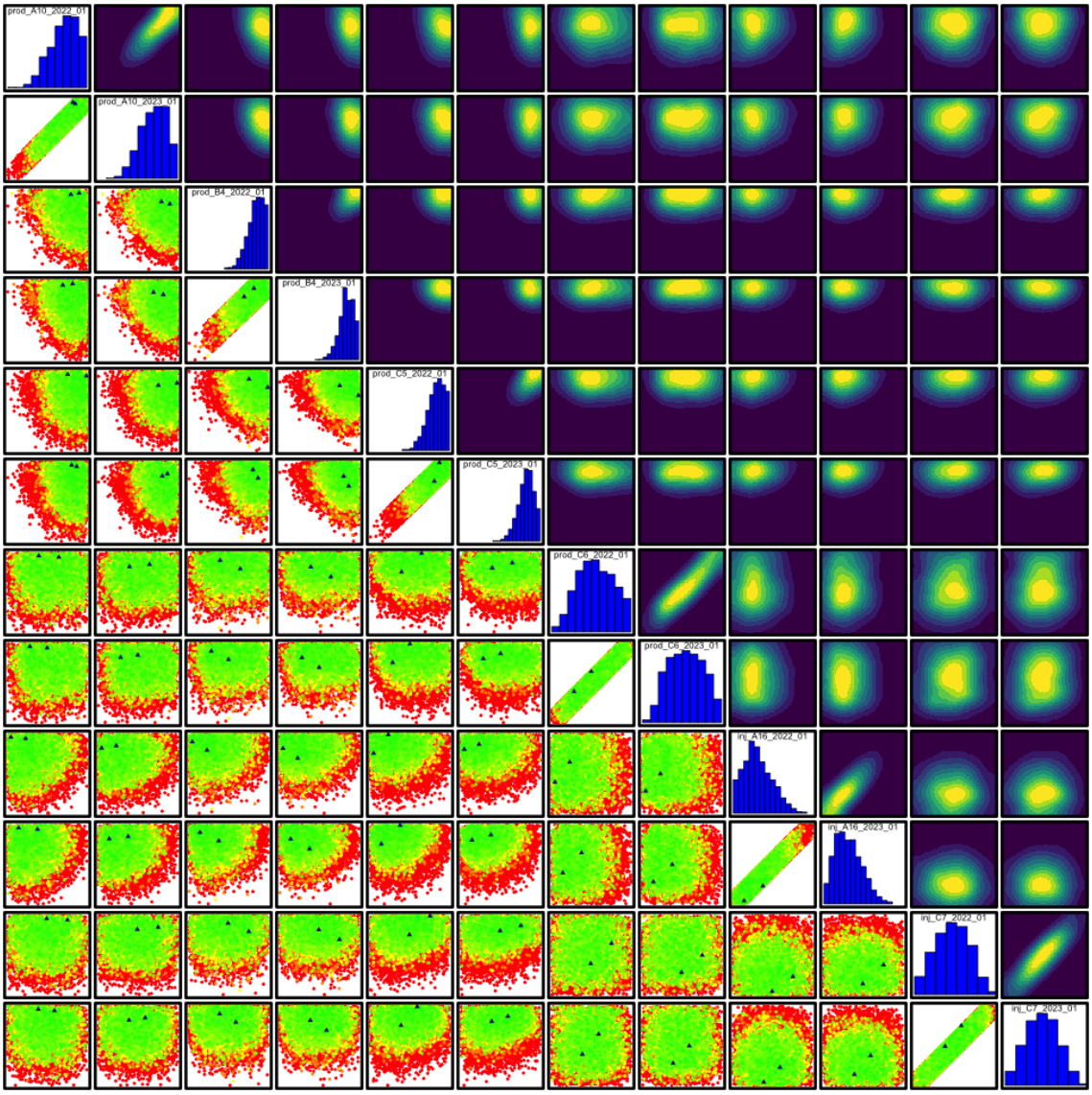
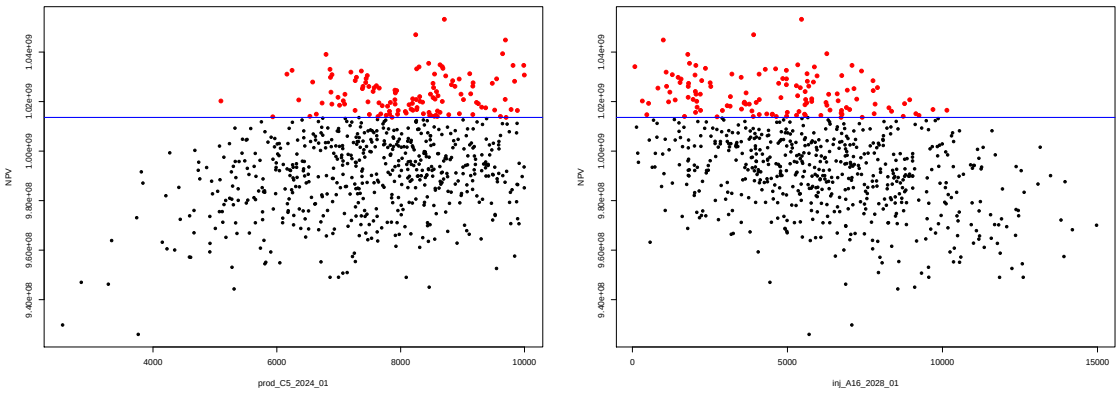


Figure 5.35: Jade wave 2 decision support implausibility measure (below the diagonal) and optical density (above the diagonal) pairs plot, with marginal density histograms (along the diagonal) for all decision parameters starting in 2022 and 2023. In the implausibility plots, the axes are swapped to match the corresponding optical density plot in which the colours green, yellow and red indicate where $0 < I_{\mathrm{DS}}(\mathbf{d}) \leq 2$, $2 < I_{\mathrm{DS}}(\mathbf{d}) \leq 3.5$ and $I_{\mathrm{DS}}(\mathbf{d}) > 3.5$ respectively, whilst blue triangles denote where $I_{\mathrm{DS}}(\mathbf{d}) = 0$ which occurs when $\mathbb{E}_F[U(\mathbf{d})] \geq U_{\max}$. For the optical density plots, dark blue through to yellow depict low to high density of points satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c$.



(a) Predicted ensemble mean NPV versus prod_C5_2024_01.

(b) Predicted ensemble mean NPV versus inj_A16_2028_01.

Figure 5.36: Jade wave 2 simulations predicted ensemble mean NPV via the EGES linear model versus decision parameters. The horizontal blue line denotes the fixed threshold defined with respect to the wave 2 uncertainty analysis at height $U_{\max} - c\sqrt{\text{Var}[\varepsilon_{\text{MD}}] + \sum_i \text{Var}[\varepsilon_i]}$, where $c = 3$, above which decision parameter vectors are classified as non-implausible. These are highlighted by the red points

non-implausible decision parameter vectors which exceed the horizontal blue line denoting the fixed uncertainty threshold computed as $U_{\max} - c\sqrt{\text{Var}[\varepsilon_{\text{MD}}] + \sum_i \text{Var}[\varepsilon_i]}$, where $c = 3$. This supports the above results and developments from wave 1. Figure 5.36a for prod_C5_2024_01 confirms that a relatively high target production rate is necessary, even for intermediate control periods, whilst Figure 5.36b for inj_A16_2028_01 shows the greater restriction of target injection rates compared with the first wave analysis. It is evident that wave 1 is necessary to restrict the more dominant target production rates, with the second wave used to refine the target injection rates.

The iterative decision support procedure stopping rule in step 5 of the algorithm presented in Section 3.6.3 is based on either: the exhaustion of computational resources; or negligible change in the volume of the non-implausible region between waves. An equivalent expression for the latter is negligible decrease in the overall uncertainty. It is observed in Figure 5.28b that the emulator adjusted standard deviation is generally much smaller than the geological uncertainty standard deviation

of $\text{sd}[\varepsilon_{\text{G}}] = 1.247 \times 10^7$ which is fixed between waves. Reformulation of the emulator over the wave 2 non-implausible region given further simulations will only result in further decreases in the emulator uncertainty. Consequently, a third wave of decision support is unlikely to yield a substantially smaller non-implausible region with the additional computational expense being unjustified. Instead a non-implausible decision space defined by the stricter criterion of $I_{\text{DS}}(\mathbf{d}) \leq c = 2$ is returned to the committee of decision makers with a volume of 3.94% of the wave 1 non-implausible region, which corresponds to 0.147% of the initial decision space and 2.77×10^{-5}% of the full hypercube volume. It is then the decision of this committee to select their preferred well control strategy given any additional knowledge or preferences that the analysts are not informed of.

5.10.2 Decision Support Wave 2 Results for Alternative Uncertainty Quantifications

The second wave of iterative decision support is repeated in order to perform a sensitivity analysis for the effects of various uncertainty specifications starting from the wave 2 uncertainty analysis in Section 5.9. A summary of the results detailed in columns four and five of Table 5.6 which also illustrates the progression from wave 1. For clarity, column four shows the non-implausible volume of the wave 1 non-implausible decision space, whilst column five depicts the non-implausible proportion of the entire hypercube and is directly comparable to the wave 1 results in column three. This exploits the computational efficiency of the developed procedure. All comments in this discussion are with respect to the proportion of the wave 1 non-implausible decision space.

The detailed treatment of geological uncertainty yields $\text{Var}[\varepsilon_{\text{G}}] = 1.555 \times 10^{14}$ and is shown to be the largest single source of uncertainty in Table 5.5 on page 291. Omission of geological uncertainty results in a non-implausible volume of 1.96%, as seen in the second row of Table 5.6 representing a substantial decrease in the proportion compared with the analysis using the full uncertainty quantification. It

Uncertainty Specification	Wave 1 Non-Implausible Volume	Wave 1 Non-Implausible Volume of Hypercube	Wave 2 Non-Implausible Volume of Wave 1 Space	Wave 2 Non-Implausible Volume of Hypercube
Hierarchical Emulator with All Uncertainties	3.72%	7.03×10^{-2}%	15.87%	1.12×10^{-2}%
Omitting Geological Uncertainty	2.80%	5.29×10^{-2}%	1.96%	1.04×10^{-3}%
Before Removing NPV Cost Parameter Uncertainty Linear Effect	99.86%	1.89%	100%	1.89%
Emulator Uncertainty Only	2.33%	4.40×10^{-2}%	1.17%	5.16×10^{-4}%
No Emulator Uncertainty	0.99%	1.87×10^{-2}%	10.75%	2.01×10^{-3}%
No Uncertainties or Emulator Uncertainty	0.04%	7.56×10^{-4}%	∼0%	∼0%

Table 5.6: Summary of the proportion of decision space classified as non-implausible after wave 1 and wave 2 of decision support for the Emerson Jade model for various included uncertainties within the decision support implausibility measure using threshold value $c = 3$. The first row displays the results using the hierarchical emulator for the utility function and a comprehensive uncertainty quantification with all subsequent rows omitting various forms of uncertainty. Wave 1 results are presented in the second and third columns showing the non-implausible proportion of the initial decision space formed by imposing both the range and difference constraints and the hypercube due to only the range constraints respectively. Wave 2 results are exhibited in the fourth and fifth columns for the non-implausible proportion of the wave 1 decision space and of the initial hypercube decision space respectively. Note that imposition of the difference constraints before the wave 1 analysis results in a reduction to 1.8895% of the initial hypercube volume due to the range constraints only.

is perceived that many decision parameter vectors may be falsely discarded.

In the wave 1 analysis the importance of considering the context of the uncer-

tainty quantification was demonstrated; for decision support this entails accounting for linear transformation effects. The wave 2 before and after assessments of the NPV cost parameter uncertainty in Section 5.9.3 of $\text{Var}[\varepsilon_{\text{C}}] = 7.725 \times 10^{16}$ and $\text{Var}[\varepsilon_{\text{C}}] = 3.296 \times 10^{13}$ respectively illustrates this substantial difference with the former being much larger than any other source of uncertainty, including the emulator. Consequently, decision support using the initial assessment yields no further reduction in the non-implausible decision space: it is impossible to distinguish between any of the decision parameter settings rendering this wave of decision support ineffective. This result is shown in the third row of Table 5.6.

Failure to perform an uncertainty quantification beyond the emulator yields a wave 2 non-implausible volume of 1.17% of the wave 1 non-implausible decision space, as seen in the fourth row of Table 5.6. This characterises the belief that the modelling framework is an exact representation of the real world physical system and any interactions that the user has with it; the only uncertainty pertains to the statistical construction of the emulator as a surrogate for the Jade model ensemble mean NPV. The result of this inaccurate belief is a considerable difference versus the 15.87% obtained using a full uncertainty quantification, hence it is believed that many reasonable choices for $\mathbf{d}$ are erroneously rejected.

Retaining the uncertainty analysis but omitting the emulator uncertainty yields a reduction to 10.75% of the wave 1 non-implausible region in the fifth row of Table 5.6; much closer to the 15.87% using the full uncertainty quantification. This constitutes treating the emulator expectation as equal to the Jade model output and provides a theoretical lower bound on the proportion of space remaining after wave 2 given the current uncertainty analysis. This suggests that there is some improvements which may be achieved, however it is fairly limited compared with resolving geological uncertainty; the dominant wave 2 source of uncertainty. This justifies the decision to end the algorithm after two iterations.

A final consideration is the consequence of excluding all sources of uncertainty both from the emulator and those detailed in Section 5.9. This is equivalent to

identifying $\mathbf{d}$ for which $I_{\mathrm{DS}}(\mathbf{d}) = 0$ which results in a non-implausible volume of effectively 0%, as shown in the sixth row of Table 5.6. This assumption is unrealistic with the sensitivity analysis emphasising the complications of inadequately linking the model to the real world physical system through a comprehensive network of uncertainties. For decision support this has resulted in the omission of effectively all well control strategies with a very small focus only around the current best simulated utility function, as represented by $U_{\max}$.

5.11 Validation of the Decision Support Non-Implausible Region

The purpose of this section is to validate the results of emulation and iterative decision support applied to the Emerson Jade model. Access to the underlying Jade geology model enables such a validation and evaluation of the decision support non-implausible region identified after two iterations of the procedure through the use of previously unseen realisations.

A design for this analysis is constructed and discussed in Section 5.11.1 differentiating between simulations for the initial Jade ensemble of 50 models and for an extra 150 geological realisations, explaining how these are necessary for the subsequent validation analyses. Simulations for the initial ensemble are first used to assess the efficient geological ensemble subsampling linear model in Section 5.11.2 before validating the performance of the emulator for the Jade ensemble mean NPV in Section 5.11.3. The results of decision support are analysed in Section 5.11.4 for the Jade expected NPV utility function which requires simulations from previously unseen realisations.

5.11.1 Design of Validation Simulations

A validation wave of simulations may be performed using either the initial ensemble of 50 models, or for a larger collection which also incorporates say an extra 150

previously unseen geological realisations, termed the full ensemble which consists of 200 models. The type of assessments and validations which are subsequently performed depends on the ensemble used. Within this analysis, simulations for the initial ensemble are used to assess the EGES linear model and emulator for the Jade ensemble mean NPV in Sections 5.11.2 and 5.11.3, whilst the full ensemble is used to validate the decision support non-implausible region in Section 5.11.4. The number of simulations using each ensemble depends on constraints on the computational budget. Simulations for the full and initial ensemble are evaluated for 10 and 40 decision parameter vectors respectively, constituting 2000 simulations each, and a total of 4000. Note that as the full ensemble contains the initial ensemble, all 50 decision parameter vectors are evaluated for the initial ensemble. A large collection of candidate decision parameter vectors are uniformly sampled from the wave 2 non-implausible region satisfying $I_{\mathrm{DS}}(\mathbf{d}) \leq c = 2$ from which to formulate the design for this third wave of simulations.

It is of interest to investigate decision parameter settings which (approximately) maximise $\mathbb{E}_F[U(\mathbf{d})]$, $\mathbb{E}_F[U(\mathbf{d})] + 2\sqrt{\mathrm{Var}_F[U(\mathbf{d})]}$ and $\mathbb{E}_F[U(\mathbf{d})] + 3\sqrt{\mathrm{Var}_F[U(\mathbf{d})]}$ for the wave 2 emulator for $U(\mathbf{d})$. Note that these may coincide, although for this analysis the identified decision parameter vectors are distinct and denoted as $\mathbf{d}'_{\mathrm{max,exp}}$, $\mathbf{d}'_{\mathrm{max,2CIupr}}$, and $\mathbf{d}'_{\mathrm{max,3CIupr}}$ respectively. The design for the full ensemble includes 8 distinct decision parameter vectors which are randomly chosen from the above collection of candidate decision strategies. This is augmented to include $\mathbf{d}'_{\mathrm{max,exp}}$ and $\mathbf{d}'_{\mathrm{max,2CIupr}}$ as these are where the emulator indicates that a higher ensemble mean NPV is likely to be achieved, with the latter also accounting for some emulator uncertainty. Similarly, the design of simulations for the initial ensemble only consists of a further 39 distinct randomly selected decision parameter vectors from this collection which is augmented to include $\mathbf{d}'_{\mathrm{max,3CIupr}}$ since this corresponds to greater emulator uncertainty.

The validation design is illustrated and compared with the wave 2 design in Figure 5.37 which displays a pairs plot for decision parameters for producer well

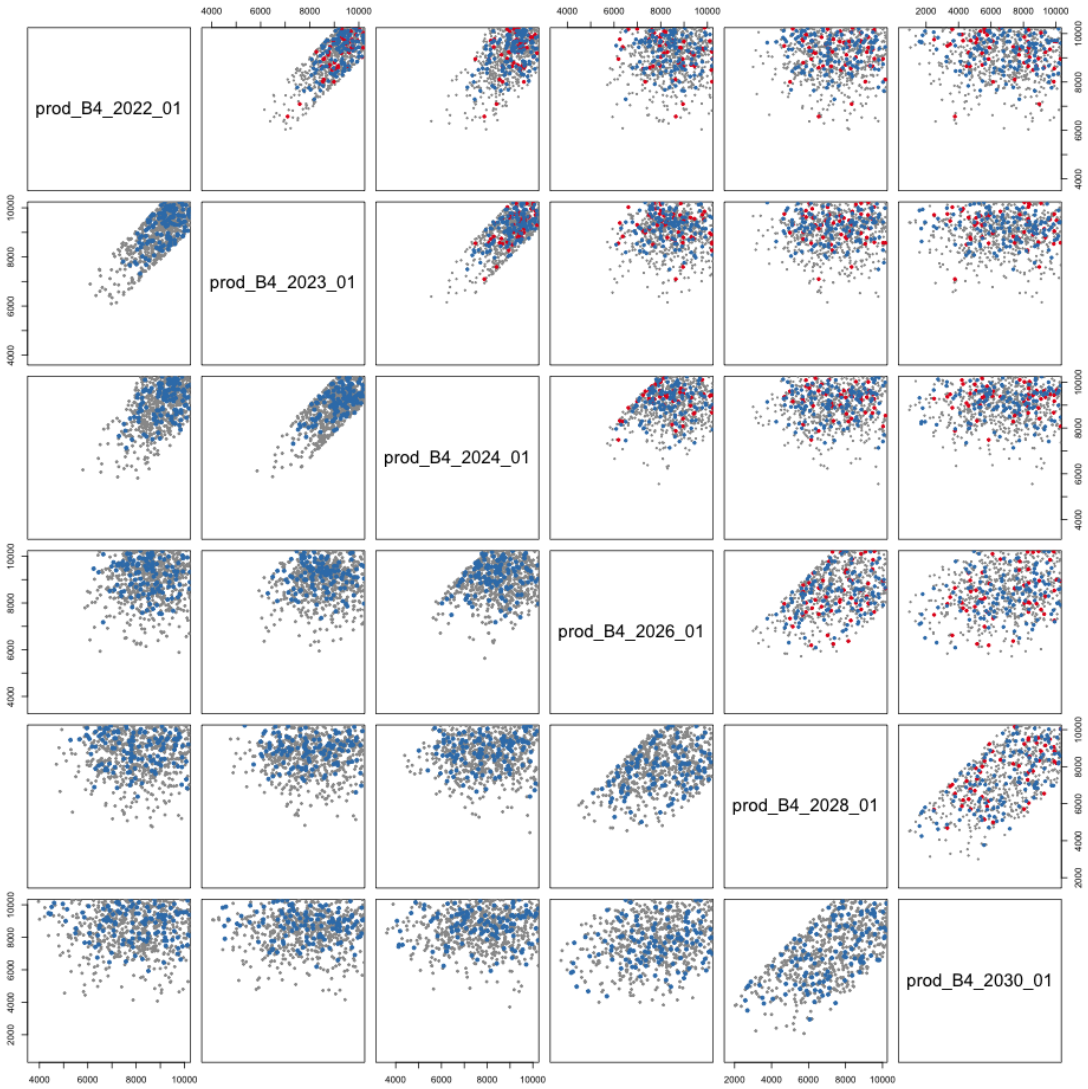


Figure 5.37: Comparison of the designs for Jade wave 2 and validation pairs plot for producer well B4 decision parameters. The validation design consists of 50 points split into groups of 10 and 40 using the full and initial ensemble respectively, as presented in Section 5.11.1. The above diagonal plots show the wave 2 and validation design points in grey and red respectively, whilst the below diagonal plots show the wave 2 design only. In all plots the blue points indicate the wave 2 design points that are classified as non-implausible for decision support directly using the simulation output along with the wave 2 uncertainty quantification.

B4. The above diagonal plots encompass both the wave 2 and validation designs which are denoted by grey and red points respectively with the below diagonal plots illustrating the wave 2 design only. In all plots the blue points indicate the wave 2

design points that are classified as non-implausible based on the simulation output and the wave 2 uncertainty quantification. The pattern closely resembles the non-implausible region in Figure 5.34 corresponding to very high target production rates for the first four control intervals, and fairly high for the last two control intervals. Compared with the wave 2 design, these cover a much smaller region, as expected due to the further decrease in the non-implausible volume between waves 1 and 2 of decision support. It is observed that the blue wave 2 simulations non-implausible points are also situated in similar locations to the validation results that are sampled from the non-implausible space devised using the wave 2 hierarchical emulator and uncertainty analysis, but with threshold $c = 2$.

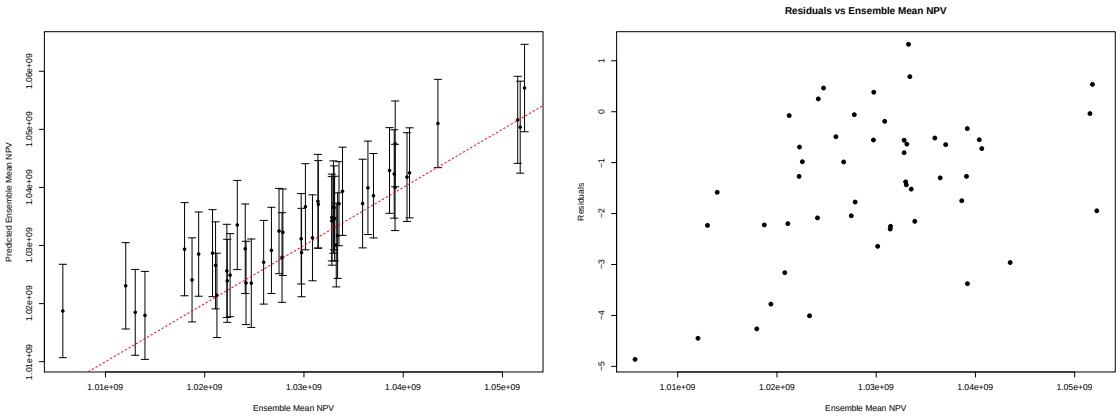
5.11.2 Validation of EGES Linear Model for the Ensemble Mean NPV

In Section 5.2 the efficient geological ensemble subsampling techniques presented in Section 3.2 are applied to an exploratory batch of 50 Jade model simulations yielding a reduction from the initial ensemble of 50 geological realisations to a subset of 3 models. This includes the construction of an EGES linear model to efficiently predict the ensemble mean NPV given the NPV for these 3 models. It is used throughout the wave 1 and 2 analyses via the hierarchical emulator construction in Sections 5.4.4 and 5.8.4 respectively, which benefits from the substantial computational savings for only a modest increase in the uncertainty.

The third wave of simulations uses all of the initial ensemble which are employed to validate the EGES linear model over the non-implausible region with the Jade ensemble mean NPV predicted given the Jade 13, 30 and 33 NPV. Validation diagnostic plots are shown in Figure 5.38 including: the three standard deviation prediction interval versus the simulated ensemble mean NPV in Figure 5.38a; and the standardised prediction residuals versus the simulated and predicted ensemble mean NPV in Figures 5.38b and 5.38d respectively, and the simulation indices in Figure 5.38c. The EGES linear model has a tendency to over-predict the Jade

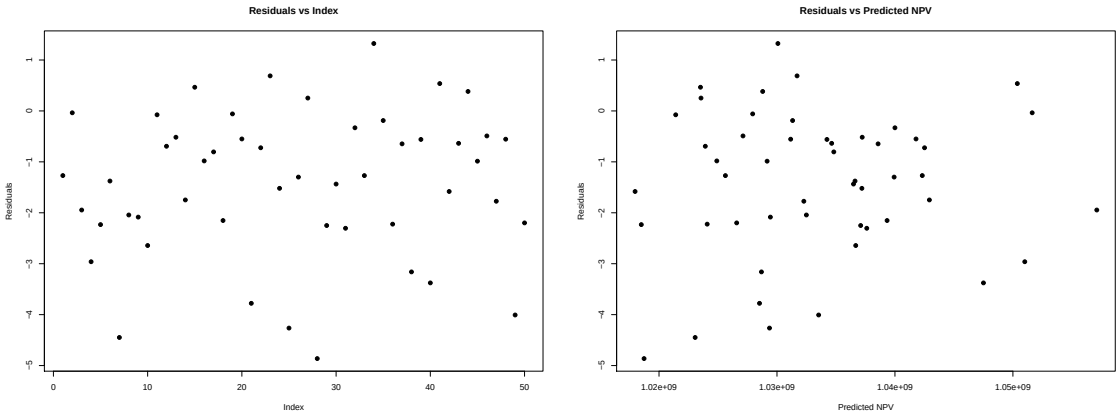
ensemble mean NPV, although many of the prediction intervals do still contain the simulated values. This behaviour is most noticeable for smaller NPVs whilst there are no other patterns evident in the residuals plots. This is explained as the EGES linear model was originally fitted using the 50 decision parameters distributed over the full 36-dimensional hypercube decision parameter space; hence they are very sparse, particularly within the non-implausible region which represents a tiny proportion of this space from which these diagnostic runs are drawn. Consequently prediction is achieved by some degree of extrapolation. Furthermore, there exists variability in the fitting of the linear model due to the choice of design and subsequent simulation output, although it is noted in Section 5.2 that a very high adjusted R^2 is obtained for regression on almost all subsets of three Jade model NPVs. The over-prediction is consistent and not of a substantial size and so for the reasons highlighted when considering utility function parameter uncertainty in the form of NPV cost parameter uncertainty in Sections 5.5.4 and 5.9.3, this does not affect the validity of the EGES linear model, or the results of decision support.

An assessment of the results of decision support based on the wave 2 uncertainty analysis and directly using the validation simulation output is performed with examples of the results exhibited in Figure 5.39. Outcomes using the EGES linear model with the NPV for the three sub-selected Jade models and the actual initial ensemble mean NPV are shown in the top and bottom rows respectively. The blue point denotes $\mathbf{d}$ from wave 2 that yields $U_{\max} = \max_{\mathbf{d}\in\mathcal{D}_2} U(\mathbf{d})$. A fixed decision support threshold (from wave 2) is used because there is no emulator uncertainty and the uncertainty analysis does not encompass variability for different choices of $\mathbf{d}$. This is represented by the horizontal red line and is calculated as $U_{\max} - 3\sqrt{\sum_i \mathrm{Var}[\varepsilon_i]}$ where each $\mathrm{Var}[\varepsilon_i]$ is the variances of the sources of uncertainty assessed in Section 5.9. Note that the additional uncertainty induced by the EGES linear model is handled via the inclusion of $\mathrm{Var}[\varepsilon_{\mathrm{EGES}}]$ in the top row only. The ramifications of the EGES linear model over-prediction are observed in Figures 5.39a and 5.39b where all decision strategies are classified as non-implausible and are well above the fixed threshold.



(a) Prediction interval versus simulated ensemble mean NPV.

(b) Standardised residuals versus the simulated ensemble mean NPV.



(c) Standardised residuals versus index.

(d) Standardised residuals versus the predicted ensemble mean NPV.

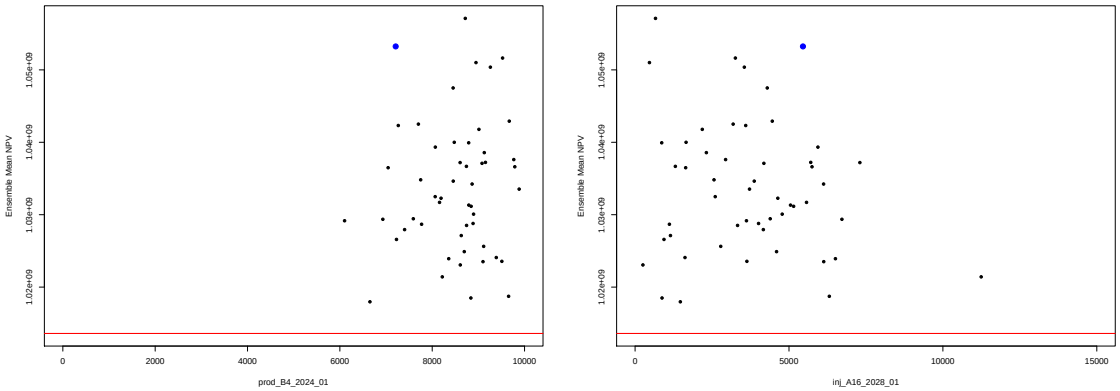
Figure 5.38: Validation diagnostics for Jade EGES linear model over the decision support non-implausible space. Figure 5.38a shows the three standard deviation prediction intervals versus the simulated ensemble mean NPV where the red dashed line denotes when the emulator and simulator coincide. The standardised prediction residuals are plotted versus the simulated and predicted ensemble mean NPV in Figures 5.38b and 5.38d respectively, and versus the simulation index in Figure 5.38c.

Performing decision support using the Jade ensemble mean NPV in Figures 5.39c and 5.39d results in 47 out of 50 cases being classified as non-implausible. These results are unsurprising since the validation simulations design is constructed over the wave 2 non-implausible region identified using emulators but with the stricter cut-off of $c = 2$. A further point is that the maximum wave 2 EGES linear model

prediction for the ensemble mean NPV exceeds all wave 3 ensemble mean NPVs. This is again due to a tendency of the EGES linear model to over-predict. However, this is used consistently throughout the analysis and within the hierarchical emulator structure and so does not invalidate the results of decision support.

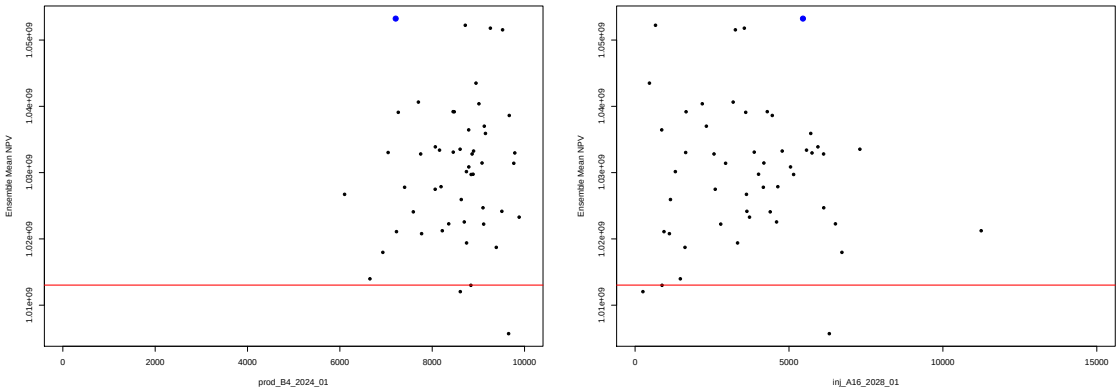
5.11.3 Validation of the Emulator for the Ensemble Mean NPV

At each wave of iterative decision support a hierarchical emulator is constructed for the Jade ensemble mean NPV where validation requires simulations for the entire initial ensemble. Focusing on the wave 2 hierarchical emulator which is used to identify the final decision support non-implausible region, validation diagnostics are performed by evaluating the emulator adjusted expectation and variance for the wave 3 simulations with the results exhibited in Figure 5.40. The hierarchical emulator three adjusted standard deviation credible intervals (an approximate 95% credible interval) versus the simulated Jade ensemble mean NPV is illustrated in Figure 5.40a where the red dashed line corresponds to where the emulator prediction and Jade ensemble mean NPV coincide. It displays that there exist only 2 cases (out of 50, thus this represents 4% of points) where the simulated value does not lie within the credible interval. Similarly, this is evident in the standardised residuals versus the simulated ensemble mean plot in Figure 5.40b where the majority of the residuals have magnitude less than three. Note that the residuals tend to be large and negative for smaller values of the Jade ensemble mean NPV seen in the bottom-left corner; this is where the two cases of diagnostics failure are situated. This suggests some difficulties in the emulator accurately portraying smaller values of the utility function and is potentially a consequence of the smaller uncertainty for some of the NPV constituents where truncation is imposed within the structured emulators exploiting known simulator behaviour. There is also evidence of a positive linear trend, although this is strongly influenced by the aforementioned points in the bottom-left corner. Excluding these points and focusing on the region of greatest



(a) EGES linear model predicted ensemble mean NPV versus prod_B4_2024_01.

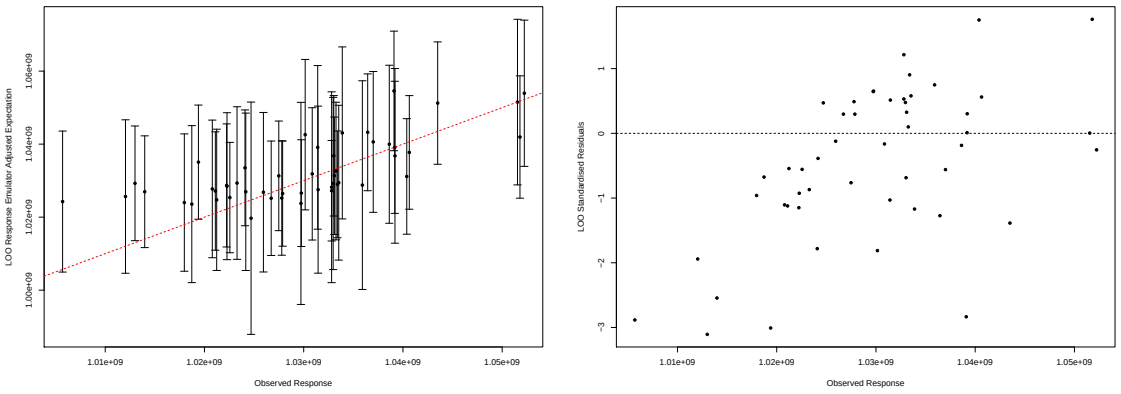
(b) EGES linear model predicted ensemble mean NPV versus inj_A16_2028_01.



(c) Simulated ensemble mean NPV versus prod_B4_2024_01.

(d) Simulated ensemble mean NPV versus inj_A16_2028_01.

Figure 5.39: Decision support for the Jade initial ensemble mean NPV validation simulations. Results are shown in the top row using the EGES linear model predicted ensemble mean NPV, whilst in the bottom row using the simulated ensemble mean NPV directly. The left and right columns display plots versus prod_B4_2024_01 and inj_A16_2028_01 respectively. In each plot the blue point corresponds to the decision parameter vector that gives rise to $U_{\max} = \max_{\mathbf{d} \in \mathcal{D}_2} U(\mathbf{d})$ from the wave 2 simulations. The horizontal red line represents the wave 2 fixed threshold above which a decision parameter vector is classified as non-implausible. This is calculated as $U_{\max} - 3\sqrt{\sum_i \mathrm{Var}[\varepsilon_i]}$ where each ε_i corresponds to a source of uncertainty. For the top row, this encompasses EGES uncertainty via $\mathrm{Var}[\varepsilon_{\mathrm{EGES}}]$, whilst this is omitted in the bottom row.



(a) Ensemble Mean NPV credible intervals versus simulated values.

(b) Ensemble Mean NPV standardised residuals versus simulated values.

Figure 5.40: Validation diagnostics for the wave 2 hierarchical emulators for the Jade ensemble mean NPV over the decision support non-implausible space. The left plot shows the emulator adjusted expectation with 3 adjusted standard deviation credible intervals versus simulated values where the red dashed line denotes when the emulator and simulator coincide. The right plot shows the emulator standardised residuals versus the simulated values.

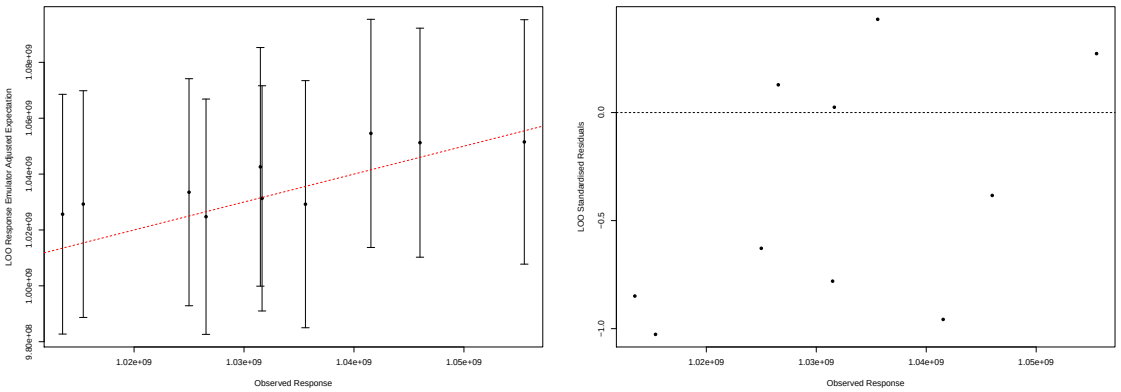
interest where the highest ensemble mean NPVs are returned, this pattern is greatly weakened, but not eliminated, whilst most of the standardised residuals are within $[-2, 2]$. These observations highlight that the emulator performs well over most of the non-implausible region, hence the results of decision support are valid.

The potential concerns raised in Section 5.11.2 regarding the tendency of the EGES linear model to over-predict are alleviated since its incorporation within the hierarchical emulator does not exhibit such effects in the validation diagnostics, partly due to the inflation of the uncertainty by emulating the NPV for a subset of ensemble members. This is evident in decision support where all cases are classified as non-implausible using the emulator which is to be expected by the nature of the design of validation simulations.

5.11.4 Validation of the Emulator for the Expected NPV

Practical application of decision support is performed for the real world Jade oil field (if it existed) rather than for the model, as is the aim in the TNO OLYMPUS Challenge. Throughout the analysis emulators are constructed for the ensemble mean NPV which is linked to the expected NPV via geological uncertainty, as discussed in Sections 5.5.2 and 5.9.1. It is not feasible to compute the true Jade expected NPV because of the large number of underlying geological model parameters which lead to an intractable integral. Instead, validation is performed using simulations for the larger full ensemble of 200 models which serves as a proxy for the Jade expected NPV. This consists of the 50 models from the initial ensemble, plus an extra 150 previously unseen geological realisations, with 10 simulations evaluated from the design described in Section 5.11.1.

The wave 2 hierarchical emulator is extended to represent the Jade expected NPV through the addition of geological uncertainty which is judged to be independent of the existing emulator uncertainty with no change to the adjusted expectation. Validation diagnostic plots are displayed in Figure 5.41. It is evident in Figure 5.41a that the three adjusted standard deviation credible intervals are all sufficiently wide to contain the approximation to the expected NPV, as portrayed by the containment of the red dashed line and hence no failures of validation diagnostics. Note also that the emulator adjusted expectation remains relatively close to the simulated value. These observations are confirmed in Figure 5.41b where almost all of the standardised residuals have magnitude less than one. Compared to emulating the Jade (initial) ensemble mean NPV, there is a marked increase in the overall emulator uncertainty due to the underlying geology. This is shown to be the largest source of uncertainty and is the reason for stopping iterative decision support after two waves, although we could continue with more geological realisations and greater computational resources as discussed below. The validation results highlight that both the emulator is sufficiently accurate whilst the quantification of the geological uncertainty is adequately able to capture the magnitude of the potential variability.



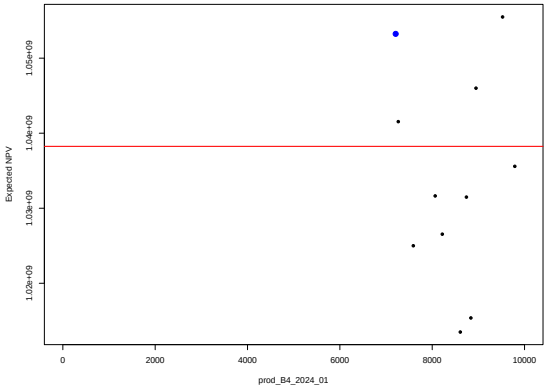
(a) Ensemble Mean NPV credible intervals versus simulated values.

(b) Ensemble Mean NPV standardised residuals versus simulated values.

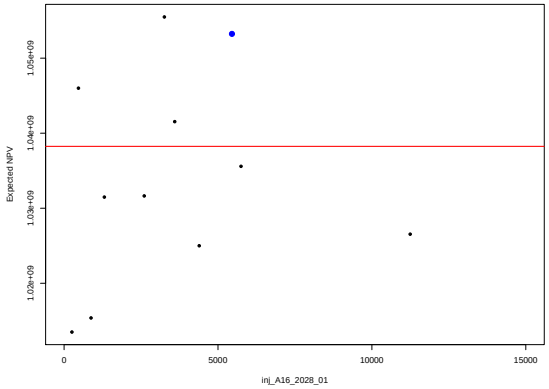
Figure 5.41: Validation diagnostics for the wave 2 hierarchical emulators for the Jade expected NPV over the decision support non-implausible space. The Jade full ensemble mean NPV over the 200 Jade models including the previously unseen realisations are used as a proxy for the Jade expected NPV. The left plot shows the emulator adjusted expectation with 3 adjusted standard deviation credible intervals versus simulated values where the red dashed line denotes when the emulator and simulator coincide. The right plot shows the emulator standardised residuals versus the simulated values.

However, more sophisticated uncertainty quantification techniques are required to further reduce the geological uncertainty and enable subsequent refinement of the decision support non-implausible region.

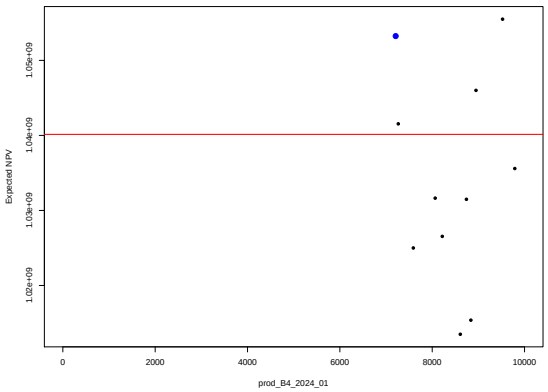
Decision support is performed for the full ensemble mean NPV as a surrogate for the true expected NPV with results illustrated in Figure 5.42 where the top row includes a revised assessment of the geological uncertainty to account for this approximation, whilst the bottom row demonstrates the consequences of omitting geological uncertainty. The blue point denotes $\mathbf{d}$ from wave 2 that yields $U_{\max} = \max_{\mathbf{d}\in\mathcal{D}_2} U(\mathbf{d})$; noting that this uses the EGES linear model prediction for the initial ensemble mean NPV, with the fixed decision support threshold (from wave 2) represented by the horizontal red line and computed as $U_{\max} - 3\sqrt{\sum_i \mathrm{Var}[\varepsilon_i]}$. Note that for the full ensemble, $\mathrm{Var}[\varepsilon_{\mathrm{G}}] = 8.605 \times 10^{11}$, a 99.4% reduction versus the original assessment using the exploratory batch of simulations. This highlights the



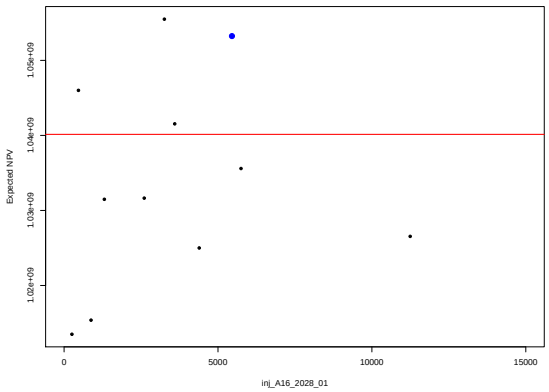
(a) Simulated full ensemble mean NPV versus prod_B4_2024_01.



(b) Simulated full ensemble mean NPV versus inj_A16_2028_01.



(c) Simulated full ensemble mean NPV versus prod_B4_2024_01 omitting geological uncertainty from the decision support threshold.



(d) Simulated full ensemble mean NPV versus inj_A16_2028_01 omitting geological uncertainty from the decision support threshold.

Figure 5.42: Decision support for the Jade full ensemble mean NPV validation simulations. In the top row a revised assessment of the geological uncertainty that accounts for the larger ensemble size as a representative for the Jade expected NPV is used, whilst the bottom row highlights the consequences of omitting geological uncertainty. The left and right columns display plots versus prod_B4_2024_01 and inj_A16_2028_01 respectively. In each plot the blue point corresponds to the decision parameter vector that gives rise to $U_{\max} = \max_{\mathbf{d}\in\mathcal{D}_2} U(\mathbf{d})$ from the wave 2 simulations using the EGES linear model to predict the initial ensemble mean NPV. The horizontal red line represents the wave 2 fixed threshold above which a decision parameter vector is classified as non-implausible. This is calculated as $U_{\max} - 3\sqrt{\sum_i \mathrm{Var}[\varepsilon_i]}$ where each ε_i corresponds to a source of uncertainty.

benefit of a larger ensemble size as well as revising the uncertainty assessment using simulations over the substantially smaller non-implausible region at later waves. Furthermore, an assessment based on the initial ensemble only over the wave 2 non-implausible region yields $\text{Var}[\varepsilon_{\text{G}}] = 1.971 \times 10^{12}$, a 98.7% reduction, and hence is no longer dominant versus emulator uncertainty. Consequently, if further waves of decision support were performed using this revised geological uncertainty it would be expected that additional parts of the decision parameter space would be eliminated as implausible.

Extension of decision support to the full ensemble mean NPV in Figures 5.42a and 5.42b illustrates the limitations of using a relatively small and fixed ensemble size, as is the case in the TNO OLYMPUS Field Development Optimisation Challenge. Only 3 of the 10 simulations are classified as non-implausible. In addition, the omission of geological uncertainty from the uncertainty assessment in Figures 5.42c and 5.42d raises the height of the threshold (red line). No further $\mathbf{d}$ are rejected as implausible, however, 10 is a small sample size and it would be expected that further decision strategies would be incorrectly classified as implausible at this stage of the analysis. The results exhibited in Figure 5.42 therefore highlight the risks of using a fixed and small ensemble as representative for the expected NPV.

5.12 Comparison of the Waves of Simulations

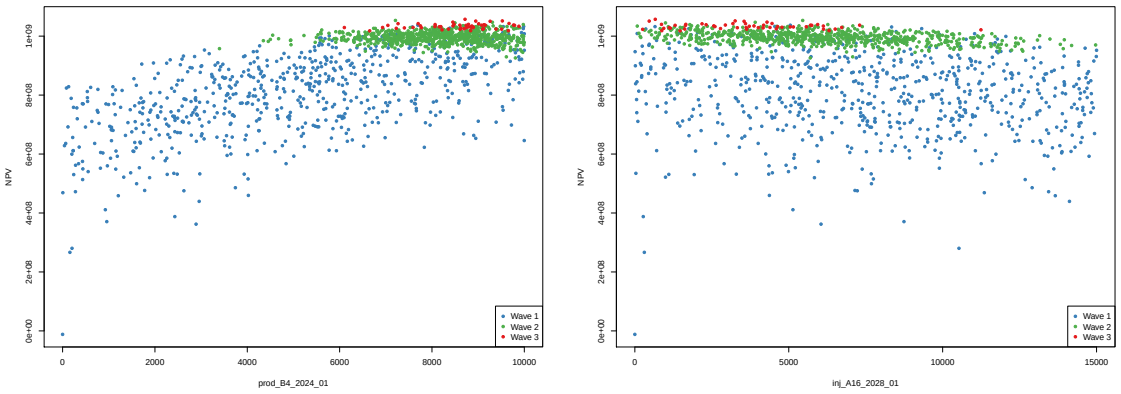
The decision support analysis for well control optimisation applied to the Emerson Jade model has been performed in three waves: the first two for decision support; and the third for validation of the emulators and the suggested decision support non-plausible region. The aim of this section is to both illustrate the output of the Jade simulations, and to demonstrate how the model behaviour and outputs vary across the waves as the region of decision space in which simulations are evaluated is iteratively refocused. Progression of the EGES linear model predictions for the ensemble mean NPV over the three waves is discussed in Section 5.12.1 and the

variation for important simulator output is detailed in Section 5.12.2.

5.12.1 Progression of the Ensemble Mean NPV

The aim of decision support is to identify a class of well control strategies that yield high expected NPV with respect to the uncertainty analysis. In this section we consider the EGES linear model prediction for the ensemble mean NPV as a surrogate for the expected NPV which is computationally intractable to evaluate, whilst a subset of three of the Jade models were used for simulations in the two waves of iterative decision support, hence it is not possible to calculate the ensemble mean NPV.

The progression of the NPV is illustrated over the three waves in Figure 5.43. Note that for consistency, the same EGES linear model constructed in Section 5.2 is used throughout. Moreover, as discussed in Section 5.9.2, each reformulation of the EGES linear model requires simulations over all 50 Jade models and is thus computationally infeasible within this analysis. For wave 1, simulations are performed across the full range of decision parameter values yielding a large spread of NPVs with high variability, some of which are high by chance. Wave 2 simulations are performed within the wave 1 decision support non-implausible region with the iterative refocusing restricting the permitted individual decision parameter values to scenarios that are more likely to yield a high NPV. This is evident with the green points corresponding to generally higher NPVs with a much smaller spread than the blue wave 1 points, although there remain some cases which are not as high. After further refinement in wave 2 decision support, the wave 3 validation simulations occupy a greatly constrained region of the decision space and produce very high NPVs. These appear to be close to the maximum attainable value. Moreover, there is a further reduction in the variability. It is expected that similar patterns would be observed for both the expected NPV and the initial ensemble mean NPV. The generally increasing NPV between waves is a consequence of the decision support algorithm and the elimination of well control strategies that are not expected to



(a) prod_B4_2024_01 (b) inj_A16_2028_01

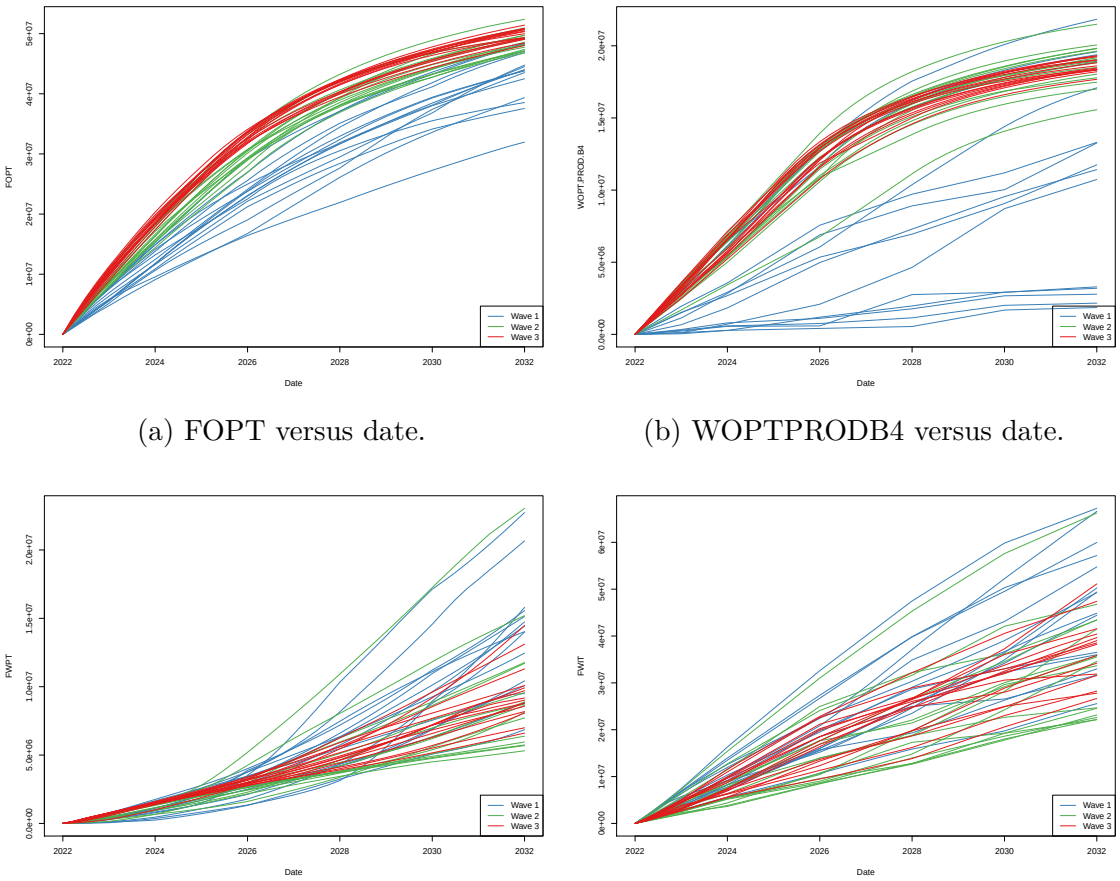
Figure 5.43: Progression of the EGES linear model predictions for the Jade ensemble mean NPV over the three waves. Blue, green and red points denote simulations from waves 1, 2 and 3 respectively.

deliver a high NPV.

5.12.2 Evolution of Simulator Outputs

This section provides an illustration of the Jade model outputs and their evolution over the waves of simulations to highlight the effect of restricting to increasingly small regions of the decision parameter space. Time series plots for Jade 30 FOPT, FWPT, FWIT and WOPT for producer well B4 in Figure 5.44 display the form of several important simulation outputs as contributors to the NPV formula with similar behaviour also observed for Jade 13 and 33. Firstly, it is noted that for all four outputs there is increasing variability with time. Another distinguishing feature in all four plots is the decreasing variability in successive waves following the same trend as the NPV in Section 5.12.1. Similarly, this is a consequence of the refocusing on decision parameter vectors that are expected to result in a high NPV.

Figure 5.44a shows the time evolution of FOPT where it is observed that simulations in later waves tend to cluster on strategies that consistently produce larger volumes of oil at early time points with decreasing variability at each fixed time point between waves 1, 2 and 3. This is to be expected as the contribution of oil is viewed as the main driver in achieving a high expected NPV. Decomposition of



(a) FOPT versus date.

(b) WOPTPRODB4 versus date.

(c) FWPT versus date.

(d) FWIT versus date.

Figure 5.44: Evolution of Jade 30 simulation outputs versus dates for a subset of wave 1, 2 and 3 simulations, denoted by blue, green and red lines respectively.

the FOPT by well in Figure 5.44b similarly shows a decreasing spread. However, there also exist strategies in wave 1 and 2 that yield a higher WOPT at later time points than in wave 3, although these strategies routinely deliver a higher return at early times prior to the impact of the high discounting factor. Also, there are four producer wells in the Jade field, hence FOPT must be distributed between these wells for which the optimal split is not necessarily even: it depends on the interaction between all of the wells.

Analysis of FWPT in Figure 5.44c reveals that the best well control strategies from wave 3 generally do not yield the smallest water production at any time point. The only condition being that FWPT is not too high. Similarly, in Figure 5.44d, the wave 3 simulations return FWIT that are not necessarily minimal across all waves,

but are also not too high. Both instances are explained by the preference to achieve a high FOPT, especially at earlier time steps, which may require increased water production or injection with the incurred costs offset by the increase in revenue due to the higher oil price. Moreover, at later time points the high discount factor further exacerbates the situation leading to the observed moderately high FWPT and FWIT in the final few years.

5.13 Decision Support Conclusion

The purpose of this chapter is to demonstrate the commercial value for my industry partner, Emerson, of the Bayesian emulation, uncertainty quantification and decision support methodology developed throughout this PhD and presented in Chapter 3. This is fulfilled via an application to their Jade model for the well control optimisation problem construed as a decision support task. Greater access to computing resources, technical support, and domain expert knowledge enables a more comprehensive analysis compared with the TNO OLYMPUS Well Control Optimisation Challenge in Chapter 4, including a demonstration of multiple waves of iterative decision support with a refinement of the uncertainty quantification, and a validation of the results using previously unseen geological realisations of the Emerson Jade model.

The first step in the analysis involves an implementation of the efficient geological ensemble subsampling techniques to achieve computational savings by sub-selecting 3 of the 50 ensemble members utilising a collection of exploratory simulations. A wave 1 targeted Bayesian design is constructed following the same methodology presented for the TNO OLYMPUS Well Control Optimisation Challenge. In preparation for the first wave of iterative decision support, the hierarchical emulation framework exploiting known forms of simulator output behaviour is applied to the complex and computationally expensive Jade ensemble for the ensemble mean NPV. An uncertainty quantification is performed for all major sources including structural model discrepancy and decision implementation error. These results are utilised for a first

wave of the iterative decision support procedure to obtain a robust class of decisions respecting all major sources of uncertainty. Extending on the application to the TNO OLYMPUS Well Control Optimisation Challenge, methodology for the design of subsequent waves of simulations is presented and applied to the Jade model before reformulating the emulators and revising the uncertainty quantification over the substantially smaller wave 1 decision support non-implausible region. These are used to perform a second wave of decision support at which point the stopping criterion indicates that no distinguishable improvements can be achieved by further iterations. Validation diagnostics are performed for the: EGES linear model; emulators for the ensemble mean and expected NPV; and the decision support non-implausible region identified in each case. The progression of the ensemble mean NPV results and the Jade simulation output are also illustrated.

A routine convention in the petroleum industry is to use an ensemble of models to represent geological uncertainty; a practice adhered to in this application by initially devising a posterior sample of Jade models. In order to address the computational expense of evaluating the full ensemble, we implement the EGES techniques described in Section 3.2. Investigations reveal that either three or four models form an adequate representative subset, with linear model diagnostics used to show that superior extrapolation predictions are achieved using three models. This results in only a modest increase in the uncertainty relative to other sources; in particular, the geological uncertainty. A wave 1 targeted Bayesian design is constructed that incorporates prior knowledge from oil reservoir engineers regarding feasible and physically realistic well control strategies in the form of time consecutive difference constraints. This yields a decrease to approximately 1.8895% of the original hypercube volume whilst also achieving good sampling coverage.

In the reformulation of the well control optimisation problem as a decision support task, the utility function is specified to be the expected NPV. Bayesian emulators are used as fast and efficient statistical approximations for the expected NPV enabling the full exploration of the decision parameter space given only a small collection of

simulations in a process that is easy to execute in parallel. For this application, the hierarchical Bayesian emulation framework which exploits known simulator behaviour by decomposing the utility function into its constituent pieces detailed in Section 3.3 is implemented in Section 5.4. This achieves highly accurate results. Moreover, this utilises the extension of the structured emulation technique to incorporate the two-sided truncation. This is to handle the limited number of fitting points for some NPV constituents as well as issues arising from extrapolation due to higher estimates of the change point upper bound between the two modes of behaviour.

An uncertainty quantification is performed for wave 1 decision support. It is first highlighted that geological uncertainty is the largest single source after the emulator, whilst the EGES uncertainty is shown to be approximately two orders of magnitude smaller. Moreover, it is vital that the analysis recognises how only uncertainties which affect the location of optimal decisions in the form of non-linear transformations of the utility surface need to be included in order to achieve useful and meaningful results. Application of the detailed methodology to identify and quantify such non-linear transformation effects results in a three order of magnitude reduction in the variance induced by NPV cost parameter uncertainty. This uses an alternative simulation strategy based on differing views of mean oil price over the field lifetime for members of the committee of decision makers. Although the Jade model is fictitious, for completeness an upper bound for structural model discrepancy is estimated utilising the wave 1 simulations whilst commenting on some of the assumptions regarding geological ensemble and EGES uncertainty. Similarly, decision implementation error is assessed via the fast Bayesian emulators. These are both of a similar magnitude to the EGES uncertainty.

The hierarchical emulator and uncertainty quantification are combined within the first wave of the iterative decision support procedure detailed in Section 3.6 to produce a robust class of decisions respecting all major sources of uncertainty whilst providing insight into the structure of the region which is believed to yield high expected NPVs. This constitutes a volume of 3.72% of the initial decision

parameter space and $7.03 \times 10^{-2}\%$ of the full hypercube. The necessity for an accurate and comprehensive uncertainty quantification is emphasised in Section 5.6.2. Failure to account for geological uncertainty yields a discernible reduction in the non-implausible volume, with a small further decrease after omitting all other uncertainties. This is explained as the geological uncertainty is the dominant form within the wave 1 uncertainty analysis. A much larger reduction is observed after omitting the emulator uncertainty only, thus indicating that another wave is necessary to further reduce the non-implausible region. These investigations highlight the potential for decision strategies to be falsely rejected in the first wave. Meanwhile, not accounting for linear transformation effects due to the NPV cost parameter uncertainty produces unusable results.

Wave 2 of iterative decision support firstly requires a collection of simulations covering the wave 1 non-implausible region. A method is presented and applied to uniformly sample from this region with the option to augment the design with wave 1 decision parameter vectors of interest. These are used to fit a hierarchical emulator to the ensemble mean NPV which yields generally higher adjusted expectations than in wave 1 along with a reduction in the uncertainty. The developed iterative decision support framework enables the continual refinement of the uncertainty quantification as new information becomes available and the volume of the decision support non-implausible region decreases. For wave 2, the geological and EGES uncertainty analyses remain unchanged since the necessary simulations for the entire ensemble are not performed. Re-assessment of the NPV cost parameter uncertainty using the same simulated cost parameter scenarios yields a considerable 73.2% reduction versus wave 1, whilst there is negligible change in the magnitude of uncertainty due to structural model discrepancy and decision implementation error.

Performing a wave 2 decision support analysis incorporating the reformulated emulator and refined uncertainty quantification yields a further reduction in the non-implausible region to 15.87% of the wave 1 volume, corresponding to $1.12 \times 10^{-2}\%$ of the full hypercube. The decision support sensitivity analysis is repeated which again

highlights the importance of a comprehensive uncertainty quantification to avoid incorrectly eliminating decision parameter vectors. At this stage it is identified that geological uncertainty is the largest single source of uncertainty. It is approximately one order of magnitude larger than the emulator uncertainty, thus instigating the algorithm stopping rule because very little further improvement can be achieved by constructing more accurate emulator representations without first addressing the geological uncertainty.

Access to the underlying stochastic geology model and additional computing resources enables the validation of decision support outcomes for the well control optimisation problem applied to the Emerson Jade model. This uses a new collection of simulations over the wave 2 non-implausible region for both the initial Jade ensemble of 50 models, and an extra 150 geological realisations. Validation diagnostics for the EGES linear model shows a tendency to over-predict, although this does not appear to be a cause for concern when incorporated within the hierarchical emulator for the Jade ensemble mean or expected NPV. Similarly, validation of the decision support region using the unseen realisations confirms that the emulator, uncertainty quantification and decision support algorithm are capable of delivering robust results. Examination of the progression of the ensemble mean NPV over the three waves of simulations illustrates the iterative refocusing as consistently higher ensemble mean NPVs are returned across the waves. Moreover, the evolution of key simulator outputs such as FOPT, FWPT and FWIT are investigated with a decrease in the variability for each output over the waves. In general, field and well oil production totals increase, whilst water production and water injection totals decrease, however, these do not necessarily attain their lowest simulated values. This is in order to sustain higher oil production rates which yield a greater contribution towards profitability.

A robust and comprehensive decision support analysis including validation of the results is achieved for the Emerson Jade model application utilising the developed iterative decision support procedure incorporating Bayesian emulators and an uncer-

tainty quantification. After two waves the largest source of uncertainty is due to the underlying geology and thus warrants a more careful treatment than is conventionally given within the petroleum industry if further reductions in the non-implausible volume are to be achieved. This can only be resolved using a larger ensemble with simulations performed in the current non-implausible region, as evidenced during the validation diagnostics for the expected NPV. Note that this does not necessarily reduce the EGES uncertainty. The iterative framework enables such an investigation and refinement of the uncertainty quantification in response to the improvements in emulator accuracy and a decreasing non-implausible volume. In addition, more sophisticated uncertainty representation techniques including variance emulation are required if decision dependent uncertainties are to be quantified. In a real-world application, the next stage would be to communicate these results to the committee of decision makers.

Chapter 6

Bayesian Emulation of Computer Models with Known Structured Partial Discontinuities

This chapter presents novel methodology for the Bayesian emulation of computer models possessing general forms of structured partial discontinuities with endpoints which may lie within the input parameter space and of varying strength along them. It is assumed that the discontinuity locations and form are known and fixed. Motivation stems from the well placement optimisation problem within the petroleum industry where geological fault boundaries induce structured partial discontinuities in the expected NPV objective function with respect to well locations. This methodology is fully generalisable and applicable across a multitude of applications employing complex and computationally expensive computer models that possess such structured discontinuities and includes embedding within the iterative decision support procedure described in Section 3.6. The research discussed in this chapter along with the application to the TNO OLYMPUS Well Placement Optimisation Challenge in Chapter 7 are presented in our paper [174] which is currently in submission to a journal.

We begin by introducing the TNO OLYMPUS Well Placement Optimisation

Challenge in Section 6.1 as an example of a commercially important industry challenge that motivates the development of methodology for Bayesian emulation with structured discontinuities. In Section 6.2 we review the existing methodology for Bayesian emulation of computer models possessing discontinuities. The formulation and mathematical derivation of the novel emulation with structured discontinuities methodology is presented in Section 6.3 utilising two examples to elucidate the steps. We name this the Torn Embedding Non-Stationary Emulation (TENSE) approach. The generality and flexibility of this methodology is further demonstrated in Section 6.4 for a known scalar function possessing a complex collection of non-linear partial discontinuities with respect to a 2-dimensional input parameter space. Throughout this chapter and the subsequent application in Chapter 7, the notation $\mathbf{x}$ is used to represent a vector of (decision) parameters, as discussed in Section 2.1.3.

6.1 Motivation of the TNO OLYMPUS Well Placement Optimisation Challenge

The TNO OLYMPUS Field Development Optimisation Challenge [93] was first introduced in Section 3.1 as motivation for the development of methodology for decision making under uncertainty. An overview of the challenge aims and setup are presented in Section 3.1.1 as a representation of the commercially important problem exhibiting substantial difficulties in field development optimisation within the petroleum industry. This includes a description of the fictitious OLYMPUS model ensemble which is computationally expensive to evaluate and possesses many of the features encountered for real-world oil reservoirs. Results of industry and academic research to address these challenges were presented at the EAGE/TNO Workshop on OLYMPUS Field Development Optimization [39].

The aim of the TNO OLYMPUS Well Placement Optimisation Challenge is to design a well placement strategy which includes the: location; trajectory; and type of each well, along with the number of wells, drilling sequence and oil platform location,

for the TNO OLYMPUS oil reservoir model which maximises the expected NPV over the field lifetime under the uncertainty captured by the 50 geological realisations. A reactive well control strategy is implemented. For each OLYMPUS model, the NPV with respect to decision parameters $\mathbf{d}$ is defined in Equation (3.1.1), where $R_j(\mathbf{d}, t_i)$ encapsulates the difference between all revenue accrued and expenditure incurred within each discounting time interval, and is defined in Equation (3.1.4). The expected NPV objective function is approximated by the ensemble mean NPV defined in Equation (3.1.2). We reformulate this challenge as a decision support problem and focus on the important problem of maximising the ensemble mean NPV as a function of well location. Bayesian emulators are employed in order to mitigate the computational expense of evaluating the OLYMPUS ensemble during the iterative exploration of the high-dimensional location parameter space.

An important feature of the OLYMPUS model is the existence of geological fault boundaries. These inhibit the flow of oil and water across them with potentially variable levels of fluid transmissibility at different locations. This is a common occurrence in real oil fields presenting difficulties in petroleum reservoir engineering (as well as for other geological modelling applications). Within the TNO OLYMPUS Challenge the location of these fault boundaries are treated as known and fixed for the entire OLYMPUS ensemble. An illustration of the physical OLYMPUS map is shown in Figure 6.1 where the five black lines depict the fault boundaries. There are four faults with an endpoint within the location parameter space, whilst one is a sealing fault (furthest left in the plot) that acts as a full discontinuity dividing the reservoir into two distinct regions. The map is coloured by mean oil concentration over the ensemble with dark blue corresponding to the highest concentration. An alternative parameterisation of the OLYMPUS reservoir map is used in the analysis of the TNO OLYMPUS Well Placement Optimisation Challenge which is illustrated in Figure 6.2. This is obtained via a bijective transformation defined by TNO and involves a reflection in the vertical axis, rotation through 90° anti-clockwise, and then a non-linear transformation is applied to produce straight, parallel, partial fault

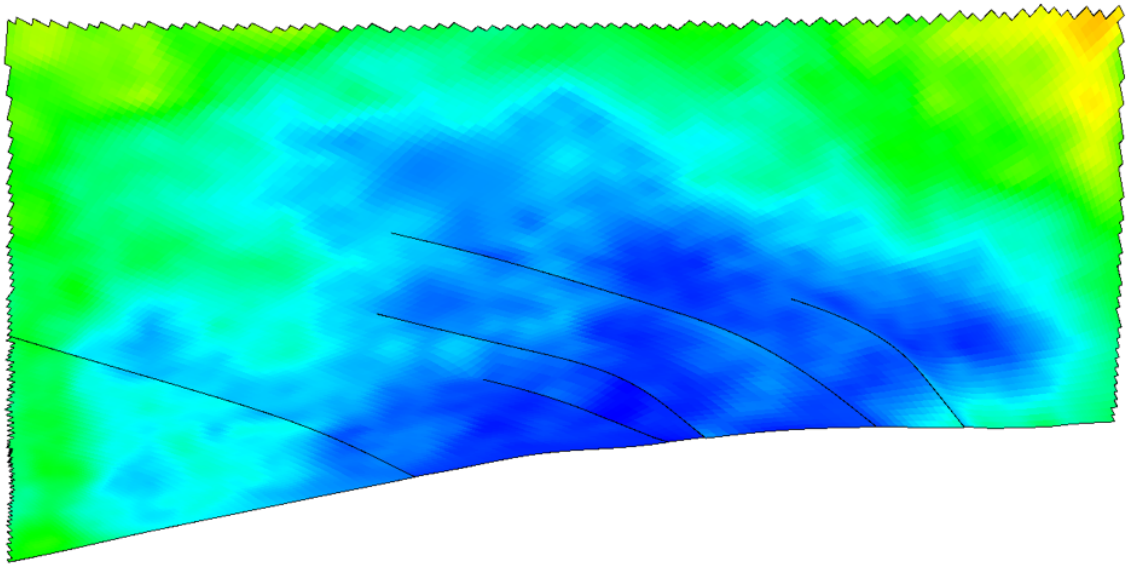


Figure 6.1: Physical map of the TNO OLYMPUS oil reservoir model coloured by mean oil concentration. Dark blue through to green represents high to low oil concentration respectively. The black lines depict the partial fault boundaries.

boundaries. The map is also coloured by mean oil concentration over the ensemble where orange through to black represents high to low mean oil concentration, whilst blue corresponds to no oil; these are regions of water only.

For the TNO OLYMPUS Well Placement Optimisation Challenge, the ensemble mean NPV objective as a function of well location possesses a form of structured partial discontinuity at known locations induced by the partial geological fault boundaries within the reservoir. Consequently, standard emulator constructions as discussed in Section 2.4 should not be applied because these assume that the function is at least continuous, and often possesses several continuous derivatives. Implementation will potentially yield an inaccurate emulator, particularly near to the discontinuities. This motivates the development of methodology for the Bayesian emulation of computer models possessing general forms of structured discontinuities with known endpoints which lie within the input parameter space and of varying strength along them. Whilst motivated by the TNO OLYMPUS Well Placement Optimisation Challenge, this methodology is fully generalisable to a multitude of applications where computer models possess such structured discontinuities.

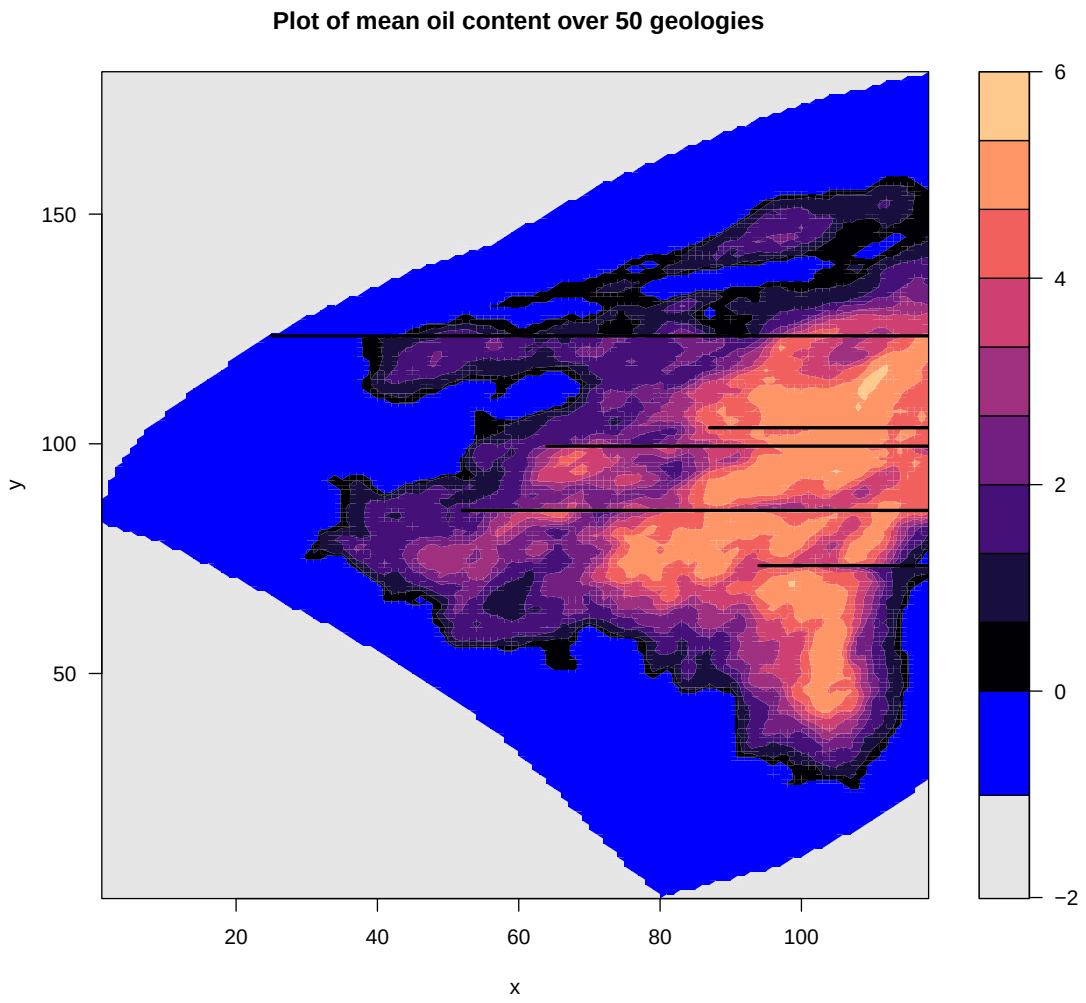


Figure 6.2: Transformed map of the TNO OLYMPUS oil reservoir model coloured by mean oil content over the ensemble. Orange through to black represents high to low mean oil concentration, whilst blue corresponds to no oil. Compared to Figure 6.1 the map is reflected in the vertical axis, rotated through 90° anti-clockwise, and a non-linear transformation applied to produce straight, parallel, partial fault boundaries.

6.2 Current Approaches to Bayesian Emulation of Computer Models with Discontinuities

In this section we review existing methodology for the Bayesian emulation of computer models possessing discontinuities and highlight their limitations. In particular, more general forms of structured partial discontinuities that occur when one or both

ends of the discontinuity lie within the parameter space with potentially variable levels of discontinuity along them necessitate the development of emulators presented in Section 6.3. An example of a function exhibiting a partial discontinuity is introduced in Equation (6.2.1), where $\mathbb{1}_{\{A\}}$ is an indicator function that takes value 1 when statement A is true, and 0 otherwise.

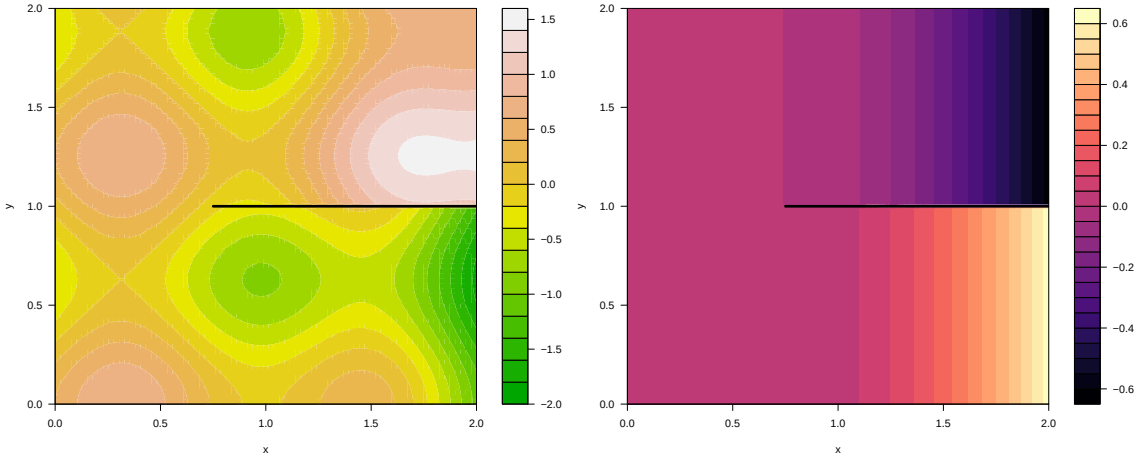
$$f(x,y) \coloneqq 0.4\sin(5x) + 0.4\cos(5y) + 0.8(x-0.75)^2\,\text{sign}(y-1)\mathbb{1}_{\{x>0.75\}} \qquad (6.2.1)$$

This is illustrated in Figure 6.3a for the parameter space $\mathcal{X} = \{(x,y) \mid 0 < x < 2,\ 0 < y < 2\}$. It possesses a discontinuity across the line $y = 1$, for $x > 0.75$, as shown by the black horizontal line. The discontinuity begins in the interior of $\mathcal{X}$ at the point $(x,y) = (0.75, 1)$, and ends on the boundary at $(x,y) = (2,1)$. In addition, the function is smooth everywhere else apart from this partial discontinuity, an attribute that should be exploited by an emulator. This is used to highlight the failures of existing methodology.

A naïve approach is to apply the standard Gaussian process (GP) or Bayes linear emulation procedures detailed in Sections 2.4.4 and 2.4.5 respectively. These fail for functions possessing discontinuities as they attempt to smooth over any discontinuity resulting in two problems:

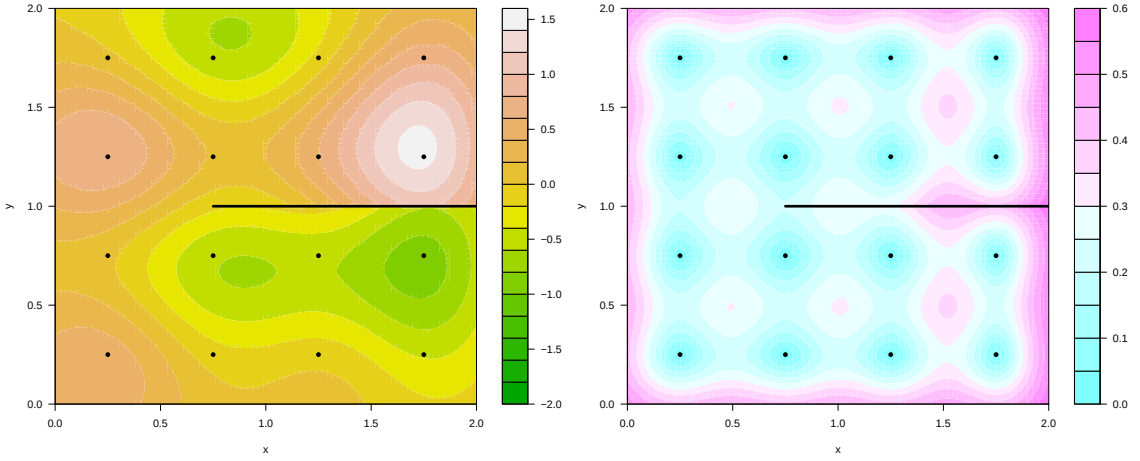
1. Emulator predictions close to the discontinuity will be highly inaccurate leading to poor emulator diagnostics,
2. Estimation of global emulator hyperparameters, for example, covariance function variance and correlation lengths, may produce results that are very sensitive to the design, invoking potential global prediction and diagnostics issues with the emulator.

For $f(x,y)$ in Equation (6.2.1), if the design does not contain points close to the discontinuity, its effect may be unnoticed, whilst also leading to moderately sized estimates of the correlations lengths. Alternatively, if the design contains points either side and close to the discontinuity, it would be detected. Moreover, very small estimates of the correlation lengths may occur depending on the rest of the design



(a) True 2-dimensional function, $f(x, y)$.

(b) Embedding surface, $v(x, y)$.



(c) The emulator adjusted expectation, $\mathbb{E}_D[f(x, y)]$, before controlling for the warping effects.

(d) The emulator adjusted standard deviation, $\sqrt{\text{Var}_D[f(x, y)]}$, before controlling for the warping effects.

Figure 6.3: Emulation of a 2D function $f(x, y)$ with a single partial discontinuity presented in Equation (6.2.1). This is illustrated in Figure 6.3a with the partial discontinuity located along the black horizontal line. The embedding surface $v(x, y)$ defined in Equation (6.3.6) is torn along the location of this discontinuity, as seen in Figure 6.3b. The emulator adjusted expectation $\mathbb{E}_D[f(x, y)]$ and standard deviation $\sqrt{\text{Var}_D[f(x, y)]}$ with the induced partial discontinuity are shown in Figures 6.3c and 6.3d respectively, where black points represent the 16 point grid design. Note the horizontal compression for larger values of x.

as an example of the second problem. In either scenario, the emulator would be non-robust and inaccurate for prediction. The main issue is that discontinuities severely violate the stationarity assumption and some degree of smoothness which

are implicit within the standard emulator covariance structure.

An option is to relax the stationarity assumption, as in [123], where a transformation of the input parameter space, termed warping, is applied, as well as the use of the non-stationary neural network and Gibbs covariance functions (see Equations (2.4.9) and (2.4.10) respectively [146]). The latter allows for input dependant correlation lengths. For $f(x, y)$ in Equation (6.2.1), this leads to high frequency fluctuations in $f(x, y)$ close to the discontinuity. This is only partially successful because the correlation lengths must change rapidly when approaching the discontinuity. Similar issues occur for the various methods of modelling non-stationary covariance functions in GP emulation reviewed in [148]. This includes the use of stationary covariance functions with input dependent hyperparameters, as well as the direct modelling of the input parameter variance matrix via matrix-valued anisotropy processes. In addition, several methods to approximate non-stationary GP emulators in applications with high-dimensional data are considered including: the Sparse General Vecchia (SGV); and nearest neighbour Gaussian process techniques.

Treed Gaussian processes developed in [74] are a popular approach to emulating functions that exhibit distinct modes of behaviour within different regions of the parameter space, as is the case across discontinuities. This is achieved by first recursively partitioning the parameter space perpendicular to single inputs to form non-overlapping (hyper-) rectangular subregions, known as the leaves of the tree. At each iteration, a new leaf is formed by splitting an existing leaf. Secondly, independent stationary GP emulators are fitted for each leaf. A major disadvantage is the requirement for a large number of simulations, especially if the parameter space is partitioned into many subregions so as to obtain sufficiently many points to accurately fit each GP emulator. This is of particular concern if the computer model is computationally expensive. Moreover, treed GPs fail to fully exploit any global behaviour of the function. In the context of functions with discontinuities, implementing a treed GP proxy assumes that the discontinuities are parallel and/or perpendicular to the defined input axes. This is not necessarily true, as discussed

for the example in Section 6.4 where the function exhibits non-linear discontinuities.

A similar approach to treed GPs is the application of Voronoi tessellations to partition the parameter space with independent GP emulators fitted within each polygonal cell, as presented in [141]. Compared with treed GPs, the Voronoi tessellation structure enables more complex forms of discontinuities to be modelled, although still not as general as observed for $f(x, y)$ in Equation (6.2.1) and in the TNO OLYMPUS model discussed in Section 6.1. Furthermore, Reversible Jump Markov Chain Monte Carlo (RJMCMC) [77] is used to update the Voronoi tessellation, however, this is a computationally expensive process, before fitting separate GP emulators to model the function in the partitioned parameter space.

Local Gaussian process emulators discussed in [73] is another related methodology where for new input points, a local approximate GP is fitted using a nearest-neighbour subset of the data to perform Bayesian updates. This is designed to reduce the computational costs associated with GP emulation, but also captures the local behaviour of the function. As for treed GPs, local GP emulation has the limitation that it models the function on a local scale and thus fails to utilise knowledge of global behaviour, whilst there are also difficulties in defining a nearest-neighbour subset of the data in the presence of discontinuities.

Further approaches employing a classification step to separate the parameter space into regions by identifying different modes of function behaviour, potentially due to discontinuities, include the use of: latent GPs in [110]; and through a diagnostic-led approach to fitting non-stationary GP emulators using region-specific covariance kernel mixture models (note that the resulting emulator is continuous) in [180]. However, these methods assume a particular form of discontinuity resulting in regions that are completely separated, hence they are unable to handle the more general forms of discontinuities such as those exhibited by $f(x, y)$ in Equation (6.2.1). Likewise, GP emulation with tipping points employed in [19] seek to first identify the location of the discontinuities before fitting separate GP emulators within each region of the partition. The process of locating the boundaries of any discontinuities is

computationally expensive and scales poorly to high-dimensions. Most importantly, this structure is unable to handle partial discontinuities.

Another option is to consider a deep Gaussian process, for example [37], with multiple layers, whereby either the correlation lengths or inputs are modelled by a second layer GP with inputs or dependant parameters in turn modelled by the next layer GP within a neural network structure. Compared with the above approaches, deep GPs possess the flexibility to achieve a given accuracy tolerance for representing functions close to discontinuities. However, there is a penalty of the additional emulator structure which is not easy to update, to estimate, or to formally incorporate the corresponding uncertainties induced by such a structure. Moreover, a deep GP constructed where smoothness is assumed for all layers may still fail emulator diagnostics. This is due to the impact of the discontinuity percolating down the layers with the potential to remain evident at each level. As an illustration, to permit a rapid change in $f(\mathbf{x})$ due to a discontinuity on the top layer requires rapid, also discontinuous, change of the inputs or correlation lengths on the second layer. This pattern continues for all subsequent layers, however it is not achievable due to the smoothness assumptions.

A further approach to examine is the direct modification of the correlation structure for the emulator to reduce the correlation between outputs on either side of a discontinuity. For example, a common suggestion is to use the geodesic distance between pairs of input parameter points in the correlation function. This is computed as the minimum for a distance metric which does not cross the discontinuity, hence it must go around it. However, this fails as it does not provide a valid covariance structure. This is demonstrated by constructing the 4×4 covariance matrix for $f(x, y)$ in Equation (6.2.1) evaluated at the four input points $(0.5, 1), (0.75, 1), (1, 1^+), (1, 1^-)$ (where the superscripts $+$ and $-$ denote the point above and below the discontinuity respectively) using a geodesic distance within a squared exponential covariance function shown in Equation (2.4.3), with $\sigma^2 = 1$ and $\theta = 0.5$. The resulting covariance matrix given in Equation (6.2.2) is not positive semi-definite; it possesses

a negative eigenvalue.

$$\Sigma = \begin{pmatrix} 1 & 0.779 & 0.368 & 0.368 \\ 0.779 & 1 & 0.779 & 0.779 \\ 0.368 & 0.779 & 1 & 0.368 \\ 0.368 & 0.779 & 0.368 & 1 \end{pmatrix} \tag{6.2.2}$$

This exemplifies the dangers associated with altering the covariance structure of an emulator to incorporate a discontinuity which are further exacerbated for multiple discontinuities of possibly complex shape.

We instead propose a parsimonious emulation structure that addresses the problem of the discontinuity directly and is described in Section 6.3. This approach will guarantee the validity of the emulator's covariance structure, even in the presence of multiple discontinuities of arbitrary shape, while also providing flexible emulator forms to deal with a wide class of computer models.

6.3 Methodology for Bayesian Emulation of Computer Models with Structured Partial Discontinuities

Bayesian emulation of computer models with structured partial discontinuities is a complex task which is inadequately handled by current methodology discussed in Section 6.2. Novel Bayesian emulators are presented to address this problem and are designated the Torn Embedding Non-Stationary Emulation (TENSE) approach. The derivation begins in Section 6.3.1 by embedding the parameter space on a hypersurface in higher dimensions using an embedding function to characterise the structured discontinuities. A consequence of the embedding function is an induced warping effect with the general form of the covariance matrix derived in Section 6.3.2 to control for the local effect. Non-stationary Gaussian process emulators are then used to obtain a global emulator in Section 6.3.3. For clarity, the derivation for our

novel approach is performed for a computer model with a 2-dimensional parameter space embedded in 3-dimensions, noting that the methodology generalises to higher dimensions.

6.3.1 Torn Embedding Surfaces in Higher Dimensions

In Section 6.2 it is emphasised that partial discontinuities present a fundamental challenge to the emulation process. This is overcome by employing the following procedure:

1. Embed the emulator's (and computer model's or function's) 2-dimensional input parameter space $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^2$ into a higher 3-dimensional input space $\mathbf{v}(\mathbf{x}) \in \mathcal{V} \subset \mathbb{R}^3$ using the embedding surface $v(x, y)$ such that:

$$\mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix} \quad \text{and} \quad \mathbf{v}(\mathbf{x}) = \begin{pmatrix} x \\ y \\ v(x,y) \end{pmatrix} \tag{6.3.1}$$

   The embedding $\mathbf{v}(\mathbf{x})$ describes a hypersurface in 3-dimensions.
2. In order to induce the discontinuities, the otherwise smooth 2-dimensional embedding surface $v(x, y)$ is torn along the known locations of the (partial) discontinuities.
3. An emulator is constructed following the methodology in Section 2.4 but in the 3-dimensional space with input $\mathbf{v}(\mathbf{x})$. The design is constructed in 2-dimensions and then raised into 3-dimensions via an application of $\mathbf{v}(\cdot)$.
4. The emulator's adjusted expectation and variance evaluated at a new point $\mathbf{x}$ are obtained from the emulator on the projection of $\mathbf{x}$ onto the embedding surface; that is evaluate $\mathbb{E}_D[f(\mathbf{v}(\mathbf{x}))]$ and $\mathrm{Var}_D[f(\mathbf{v}(\mathbf{x}))]$.

The tears in the embedding surface will induce discontinuities, of as yet uncertain sizes, in the output $f(\mathbf{x})$, as required.

The covariance structure for the embedded emulator is obtained from the desired covariance structure for the original 2-dimensional emulator. This is shown for

the squared exponential covariance function where it is assumed that the original 2-dimensional emulator possesses no regression terms. Firstly, using an isotropic distance metric with a single common correlation length hyperparameter $\theta_{2\mathrm{D}}$ (see Equation (2.4.3)), the resulting covariance structure is:

$$\mathrm{Cov}[f(\mathbf{x}), f(\mathbf{x}')] = \sigma^2 \exp\left\{-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{\theta_{2\mathrm{D}}^2}\right\} \tag{6.3.2}$$

After embedding in 3-dimensions, the covariance structure naturally extends to that shown in Equation (6.3.3). This depends on the Euclidean distance in the new 3-dimensional parameter space via $\mathbf{v}(\mathbf{x}) \in \mathcal{V}$.

$$\mathrm{Cov}[f(\mathbf{v}(\mathbf{x})), f(\mathbf{v}(\mathbf{x}'))] = \sigma^2 \exp\left\{-\frac{\|\mathbf{v}(\mathbf{x})-\mathbf{v}(\mathbf{x}')\|^2}{\theta_{3\mathrm{D}}^2}\right\} \tag{6.3.3}$$

For a general anisotropic squared exponential covariance function, the covariance of the 2-dimensional emulator (with no regression terms) is defined in Equation (6.3.4) where $\Sigma_{2\mathrm{D}}^{-1}$ governs the Mahalanobis distance between input points $\mathbf{x}$ and $\mathbf{x}'$.

$$\mathrm{Cov}[f(\mathbf{x}), f(\mathbf{x}')] = \sigma^2 \exp\left\{-(\mathbf{x}-\mathbf{x}')^{\mathrm{T}}\Sigma_{2\mathrm{D}}^{-1}(\mathbf{x}-\mathbf{x}')\right\} \tag{6.3.4}$$

Similarly, after embedding, the 3-dimensional version is stated in Equation (6.3.5) where $\Sigma_{3\mathrm{D}}^{-1}$ governs the Mahalanobis distance in 3-dimensions.

$$\mathrm{Cov}[f(\mathbf{v}(\mathbf{x})), f(\mathbf{v}(\mathbf{x}'))] = \sigma^2 \exp\left\{-(\mathbf{v}(\mathbf{x})-\mathbf{v}(\mathbf{x}'))^{\mathrm{T}}\Sigma_{3\mathrm{D}}^{-1}(\mathbf{v}(\mathbf{x})-\mathbf{v}(\mathbf{x}'))\right\} \tag{6.3.5}$$

There is freedom to choose from various permitted forms for $\Sigma_{3\mathrm{D}}$. This is an important part in the full TENSE emulator development with further discussion in Section 6.3.2.

This procedure is illustrated for the example function introduced in Equation (6.2.1) in Section 6.2, and displayed in Figure 6.3a. We specify the embedding function in Equation (6.3.6) which is shown in Figure 6.3b.

$$v(x, y) = -0.4(x-0.75)^2 \,\mathrm{sign}(y-1)\mathbb{1}_{\{x>0.75\}} \tag{6.3.6}$$

The main requirement of the embedding surface is that it is locally smooth, whilst

also being torn along the discontinuity such that the regions above and below the discontinuity are sufficiently different in height in the third dimension to decorrelate outputs on either side. It is unnecessary for $v(x, y)$ to track the form of the actual computer model function $f(x, y)$: in this example $v(x, y)$ above and below the discontinuity goes low and high respectively, whilst the function $f(x, y)$ does the opposite. The tears in the embedding surface are only present to induce the discontinuities within the embedded emulator structure.

In order to train the emulator whilst illustrating the features of its construction, a grid design of 16 runs $\mathbf{x}^{(i)},\ i = 1, \ldots, 16$, in the 2-dimensional region $\mathcal{X}$ is constructed at which $f(x, y)$ is evaluated.[1] These are shown as the black points in Figures 6.3c and 6.3d, which are raised into 3-dimensions using $\mathbf{v}^{(i)} = \mathbf{v}(\mathbf{x}^{(i)})$. In this example it is assumed that there is only a constant regression term and the isotropic form of the squared exponential covariance function shown in Equation (6.3.2) is used. Emulator updates over the embedded 3-dimensional parameter space are performed using Equations (2.4.41) and (2.4.42), with $\sigma = 0.7$, $\theta = 0.5$, $\rho = 10^{-5}$, and $\mathbf{D} = \{f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(16)})\}$, with $\Sigma_{3\mathrm{D}} = \mathrm{diag}\{\theta, \theta, \theta\}$.

The emulator adjusted expectation, $\mathbb{E}_D[f(\mathbf{x})] \equiv \mathbb{E}_D[f(\mathbf{v}(\mathbf{x}))]$, evaluated over an 80×80 grid of points in $\mathcal{X}$, is shown in Figure 6.3c. It is evident that the emulator adjusted expectation is smooth away from the discontinuity and displays a suitable jump across the discontinuity, hence mimicking $f(x, y)$ shown in Figure 6.3a reasonably well. The emulator adjusted standard deviation, $\sqrt{\mathrm{Var}_D[f(\mathbf{x})]} \equiv \sqrt{\mathrm{Var}_D[f(\mathbf{v}(\mathbf{x}))]}$, is shown in Figure 6.3d. Similarly, this shows the desired behaviour with the further along the discontinuity (in the positive x direction), the more uncorrelated the two regions (above and below the discontinuity) become. For example, the point $(1.75, 1^-)$ just below the discontinuity has a similarly high emulator adjusted standard deviation as the point $(1.75, 0)$ on the lower boundary. The emulator is therefore as uninformed at both locations, hence almost no information is passed from the

[1] For general applications the choice of a grid design is not optimal or appropriate. This is used to highlight the features induced by this emulator construction and illustrate why the subsequent derivation steps are necessary.

runs above the discontinuity and so the emulator output is almost independent of them. A more detailed discussion of this can be found in Section 7.3. However, the naïve embedding has induced a slight compression in the x-direction for the emulator adjusted standard deviation (and expectation) evaluated at larger values of x.

For the purpose of further explicating this issue, a second example function is introduced in Equation (6.3.7) and shown in Figure 6.4a. It is defined over $\mathcal{X} = \{(x, y) \mid 0 < x < 2,\ 0 < y < 2\}$. This function possesses two parallel partial discontinuities of different lengths, each with one endpoint on the right boundary of the input space.

$$\begin{aligned} f(x,y) \coloneqq\ & 0.4\sin(5x) + 0.4\cos(5y) + 1.2(x-1)^2 \mathbb{1}_{\{x>1\}} \mathbb{1}_{\{y>1.25\}} \\ & - 0.6(x-0.6)^2 \mathbb{1}_{\{x>0.6\}} \mathbb{1}_{\{y<0.75\}} \end{aligned} \tag{6.3.7}$$

This requires a more complex embedding function to accommodate the discontinuities of differing lengths defined in Equation (6.3.8), where $b(y)$ represents the x-coordinate of the line that interpolates the two interior end points, $(0.6, 0.75)$ and $(1, 1.25)$, of the partial discontinuities, as defined in Equation (6.3.9).

$$\begin{aligned} v(x,y) =\ & 0.6(x-b(y))^2 \mathbb{1}_{\{x>b(y)\}} \mathbb{1}_{\{y<1.25\}} \mathbb{1}_{\{y>0.75\}} \\ & - 0.6(x-0.6)^2 \mathbb{1}_{\{x>0.6\}} \mathbb{1}_{\{y<0.75\}} \end{aligned} \tag{6.3.8}$$

where

$$b(y) = 0.6 + \frac{(1-0.6)(y-0.75)}{(1.25-0.75)} \tag{6.3.9}$$

This is illustrated in Figure 6.4b. The use of a piecewise quadratic form for the embedding surface will become evident following the explication of the TENSE methodology with an explanation provided along with general advice on specifying the form of the embedding surface in Section 7.2.

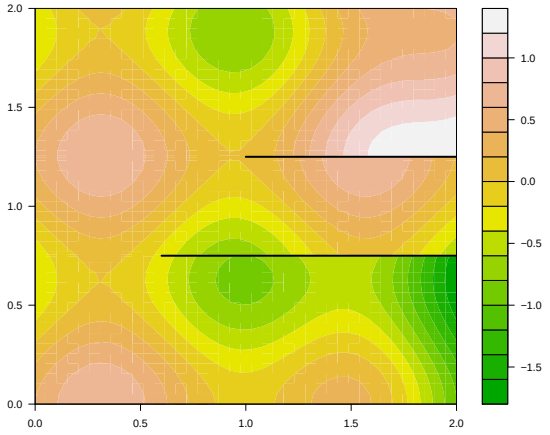
A 12 point grid design is used. An emulator is constructed with a constant mean function using the isotropic form of the squared exponential covariance function shown in Equation (6.3.2) with hyperparameters specified as $\sigma = 0.7$, $\theta = 0.5$,

$\rho = 10^{-5}$, and $\mathbf{D} = \{f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(12)})\}$, with $\Sigma_{3\mathrm{D}} = \mathrm{diag}\{\theta, \theta, \theta\}$. The emulator adjusted standard deviation, $\sqrt{\mathrm{Var}_D[f(\mathbf{x})]}$, is shown in Figure 6.4d and displays clear compression and warping effects in the middle and lower regions; a direct consequence of the chosen form of $v(x, y)$ shown in Figure 6.4b. This compression arises due to the use of a stretched embedding surface for which arc length is not conserved, as well as using a stationary 3-dimensional covariance structure. Consequently, paths on steep regions of $v(x, y)$ move "too quickly" into the third dimension leading to an induced compression on projection into 2-dimensions. Equivalently, for two pairs of points that are equidistant apart in 2-dimensions, the distance in 3-dimensions increases for regions where the embedding surface possesses larger derivatives; hence a decrease in the correlation leading to the warping effect.

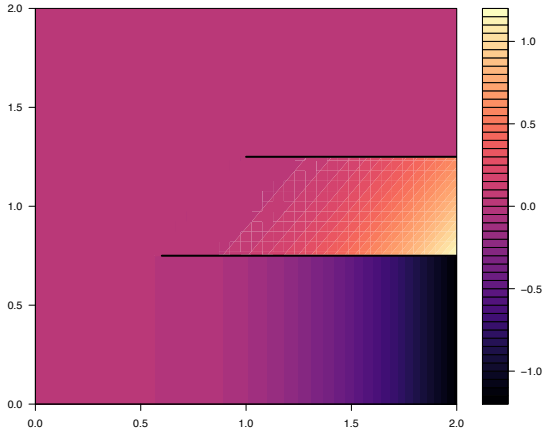
It is desired that the choice of embedding surface is sufficiently flexible to permit such stretches of the inter-point distance to enable sufficient jumps across the discontinuities, as well as to handle more complex forms of discontinuities. This includes: discontinuities with endpoints that both begin and end within the input space; and finite collections of discontinuities of a non-linear form. Neither of these cases can be addressed using an arc-length conserving embedding which would mitigate such compression effects, although not remove them entirely. The warping effect therefore constitutes a substantial problem which is resolved in Sections 6.3.2 and 6.3.3.

6.3.2 Controlling for the Warping Effect Induced by the Embedding – Local Effect

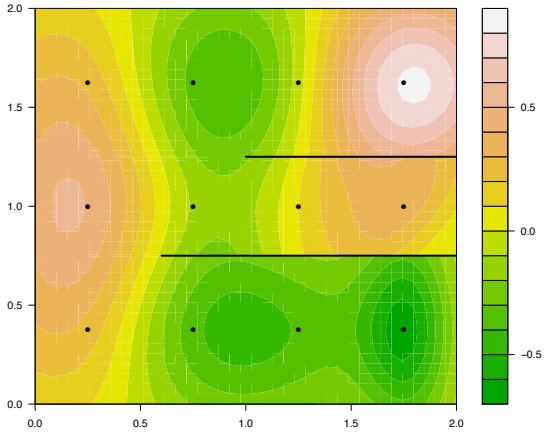
Embedding the input parameter space in higher dimensions enables the characterisation of structured partial discontinuities. However, it is shown in Section 6.3.1 that general forms of stretched embedding functions $v(x, y)$ result in a warped emulator on projection back onto the original parameter space by inducing $\mathbf{x}$ dependent correlation lengths. Consequently, these emulators are inefficient and do not reflect actual prior beliefs about the computer model. This issue is addressed using a carefully constructed non-stationary Gaussian process emulator over the 3-dimensional parameter



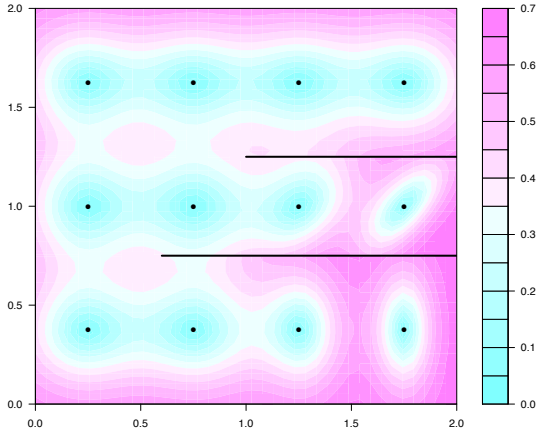
(a) True 2-dimensional function, $f(x, y)$.



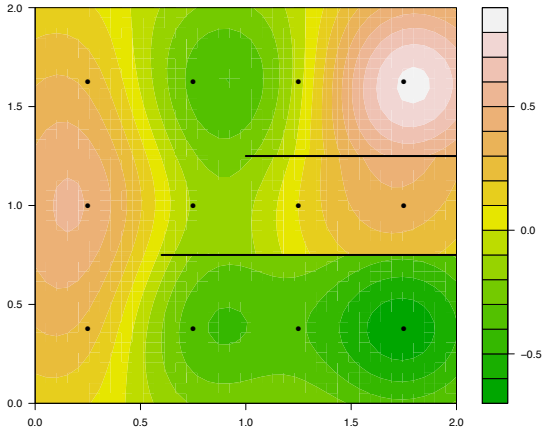
(b) Embedding surface, $v(x, y)$.



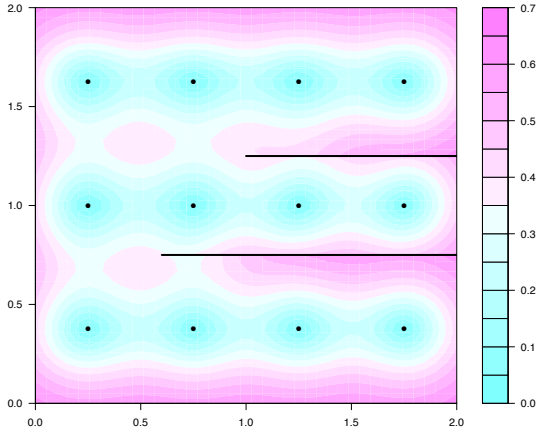
(c) The emulator adjusted expectation, $\mathbb{E}_D[f(x, y)]$, before controlling for the warping effects.



(d) The emulator adjusted standard deviation, $\sqrt{\text{Var}_D[f(x, y)]}$, before controlling for the warping effects.



(e) The TENSE emulator adjusted expectation, $\mathbb{E}_D[f(x, y)]$.



(f) The TENSE emulator adjusted standard deviation, $\sqrt{\text{Var}_D[f(x, y)]}$.

Figure 6.4: Emulation of a 2D function $f(x, y)$ with two parallel partial discontinuities of different lengths presented in Equation (6.3.7). In each plot the partial discontinuities are located along the black horizontal lines. The black points represent the 12 point grid design.

space. First, we detail a proposed form of the 3-dimensional correlation matrix, Σ_{3D}, that is guaranteed to reverse the local effect of the embedding. This is to handle the situation where input points are close together compared to the curvature of the embedding surface. This is extended to general choices of parameter vectors over the whole input space using necessarily non-stationary GPs in Section 6.3.3. This will ensure the freedom to choose from a wide variety of embedding surfaces $v(x, y)$ by removing the induced warping effect.

The derivation is performed for a reference parameter vector, $\mathbf{x}_0$, at which the aim is to specify the form of Σ_{3D} such that it induces the desired covariance structure locally around this point on projection onto the original parameter space. For the squared exponential covariance function, this is to ensure that $\mathrm{Cov}[f(\mathbf{x}), f(\mathbf{x}_0)]$ approximately has the form displayed in Equation (6.3.4) for parameters $\mathbf{x}$ that are close to $\mathbf{x}_0$.

$$\mathrm{Cov}[f(\mathbf{x}), f(\mathbf{x}_0)] \simeq \sigma^2 \exp\left\{-(\mathbf{x} - \mathbf{x}_0)^{\mathrm{T}} \Sigma_{2D}^{-1} (\mathbf{x} - \mathbf{x}_0)\right\} \tag{6.3.10}$$

This is illustrated for the commonly chosen form for Σ_{2D} given in Equation (6.3.11) where it is assumed that the 2-dimensional inputs are uncorrelated with common correlation length θ.

$$\Sigma_{2D} \simeq \begin{pmatrix} \theta^2 & 0 \\ 0 & \theta^2 \end{pmatrix} \tag{6.3.11}$$

Note that the derivation also applies to general Σ_{2D}, as outlined at the end of this section.

The actual covariance structure is calculated from the embedded emulator via $\mathbf{v}(\mathbf{x})$, hence the condition given in Equation (6.3.12) for the squared exponential covariance function must be satisfied.

$$\begin{aligned} & \mathrm{Cov}[f(\mathbf{v}(\mathbf{x})), f(\mathbf{v}(\mathbf{x}_0))] \simeq \mathrm{Cov}[f(\mathbf{x}), f(\mathbf{x}_0)] \\ \iff \sigma^2 \exp\left\{-(\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{x}_0))^{\mathrm{T}} \Sigma_{3D}^{-1} (\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{x}_0))\right\} & \simeq \\ \sigma^2 \exp\left\{-(\mathbf{x} - \mathbf{x}_0)^{\mathrm{T}} \Sigma_{2D}^{-1} (\mathbf{x} - \mathbf{x}_0)\right\} & \end{aligned}$$

$$\iff (\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{x}_0))^{\mathrm{T}} \Sigma_{3\mathrm{D}}^{-1} (\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{x}_0)) \simeq (\mathbf{x} - \mathbf{x}_0)^{\mathrm{T}} \Sigma_{2\mathrm{D}}^{-1} (\mathbf{x} - \mathbf{x}_0) \tag{6.3.12}$$

The next step is to approximate $\mathbf{v}(\mathbf{x})$ by its linear Taylor expansion around the point $\mathbf{x}_0$, as presented in Equation (6.3.13), where $v_x = \frac{\partial v(x,y)}{\partial x}$ and $v_y = \frac{\partial v(x,y)}{\partial y}$ are the partial derivatives of $v(x, y)$ evaluated at $\mathbf{x}_0$. This is equivalent to approximating the embedding surface $v(x, y)$ by the tangent plane to $v(x, y)$ at the point $\mathbf{x}_0$ and is required to construct $\Sigma_{3\mathrm{D}}^{-1}$.

$$v(x, y) - v(x_0, y_0) = v_x(x - x_0) + v_y(y - y_0) + \mathcal{O}(x^2) \tag{6.3.13}$$

For the embedding vector function $\mathbf{v}(\mathbf{x})$ this is described in Equation (6.3.14). Since $\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{x}_0)$ lies on the tangent plane, the third component can be re-expressed as a linear combination of the 2-dimensional inputs, as shown in Equation (6.3.15). The matrix A represents the linear embedding operator for raising the 2-dimensional position vector $\mathbf{x}$ onto the corresponding location on the 3-dimensional tangent plane given by $A\mathbf{x}$.

$$\begin{aligned}
\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{x}_0) &= \begin{pmatrix} x - x_0 \\ y - y_0 \\ v(x, y) - v(x_0, y_0) \end{pmatrix} \\
&= \begin{pmatrix} x - x_0 \\ y - y_0 \\ v_x(x - x_0) + v_y(y - y_0) + \mathcal{O}(x^2) \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ v_x & v_y \end{pmatrix} \begin{pmatrix} x - x_0 \\ y - y_0 \end{pmatrix} + \mathcal{O}(x^2)
\end{aligned} \tag{6.3.14}$$

$$= A(\mathbf{x} - \mathbf{x}_0) + \mathcal{O}(x^2), \quad \text{where} \quad A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ v_x & v_y \end{pmatrix} \tag{6.3.15}$$

Substituting Equation (6.3.15) into the exponent in Equation (6.3.12), and omitting

the second order terms which are of negligible magnitude for $\mathbf{x}$ close to $\mathbf{x}_0$, yields the equality shown in Equation (6.3.16). This constitutes a preservation of the Mahalanobis distance when mapping from the original to the higher-dimensional parameter space. Note that whilst this derivation is for the squared exponential covariance function, the same results are valid for other stationary isotropic covariance functions.

$$(\mathbf{x}-\mathbf{x}_0)^{\mathrm{T}}A^{\mathrm{T}}\Sigma_{3\mathrm{D}}^{-1}A(\mathbf{x}-\mathbf{x}_0) \simeq (\mathbf{x}-\mathbf{x}_0)^{\mathrm{T}}\Sigma_{2\mathrm{D}}^{-1}(\mathbf{x}-\mathbf{x}_0) \tag{6.3.16}$$

$$\iff A^{\mathrm{T}}\Sigma_{3\mathrm{D}}^{-1}A \simeq \Sigma_{2\mathrm{D}}^{-1} \tag{6.3.17}$$

It is evident that in order to counteract the linear effects of the embedding surface in the vicinity of $\mathbf{x}_0$, it is sufficient to construct $\Sigma_{3\mathrm{D}}$ such that it satisfies Equation (6.3.17).

Constructing $\Sigma_{3\mathrm{D}}$ for diagonal $\Sigma_{2\mathrm{D}}$

There are several available forms for $\Sigma_{3\mathrm{D}}$, however many of these will not facilitate sufficient decorrelation of the emulator across discontinuities in the embedding surface $v(x,y)$. The chosen form for $\Sigma_{3\mathrm{D}}$ is aligned with the tangent plane to $v(x,y)$ at the point $\mathbf{x}_0$ and is specifically selected to provide the desired behaviour across discontinuities.

In order to derive $\Sigma_{3\mathrm{D}}$, it is necessary to establish an orthonormal basis, $\{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\}$. This will be referred to as the w-basis and is contrived by stipulating $g(x,y,z) = z - v(x,y)$, noting that the level set $g(x,y,z) = 0$ defines the embedding surface $z = v(x,y)$. Application of standard vector calculus shows that $\nabla g(x,y,z)$ evaluated at $\mathbf{x}_0$ results in a vector that is normal to the embedding surface, and hence is also normal to the tangent plane. This is selected to be the unit vector $\mathbf{w}_3$ in Equation (6.3.18), where $v_x = \frac{\partial v(x,y)}{\partial x}$ and $v_y = \frac{\partial v(x,y)}{\partial y}$ are again the partial derivatives of $v(x,y)$ evaluated at $\mathbf{x}_0$. Throughout $\mathbf{e}_x$, $\mathbf{e}_y$ and $\mathbf{e}_z$ are the canonical basis in $\mathbb{R}^3$.

$$\mathbf{w}_3 \propto \nabla g(x,y,z) = -v_x\mathbf{e}_x - v_y\mathbf{e}_y + \mathbf{e}_z \tag{6.3.18}$$

The unit basis vector $\mathbf{w}_1$ is chosen to lie in the tangent plane, but pointing in the direction of steepest increase of $v(x, y)$. This dictates that the first two components of $\mathbf{w}_1$ must be parallel to $\nabla v(x, y) = v_x \mathbf{e}_x + v_y \mathbf{e}_y$ leading to the following form:

$$\mathbf{w}_1 \propto v_x \mathbf{e}_x + v_y \mathbf{e}_y + \gamma \mathbf{e}_z \tag{6.3.19}$$

The setup of the w-basis where $\mathbf{w}_1$ lies on the tangent plane whilst $\mathbf{w}_3$ is normal leads to the condition $\mathbf{w}_1 \cdot \mathbf{w}_3 = 0$, from which it is determined that $\gamma = v_x^2 + v_y^2$. It is then necessary for $\mathbf{w}_2$ to be orthogonal to both $\mathbf{w}_1$ and $\mathbf{w}_3$. Since $\mathbf{w}_1$ is chosen to be in the direction of maximally increasing $v(x, y)$, this implies that $\mathbf{w}_2$ must have zero component in the 3$^{\text{rd}}$ dimension, hence is of the form:

$$\mathbf{w}_2 \propto \beta \mathbf{e}_x + \delta \mathbf{e}_y \tag{6.3.20}$$

A second application of the orthogonality relation $\mathbf{w}_2 \cdot \mathbf{w}_1 = 0$ implies that $\beta v_x + \delta v_y = 0$ which is satisfied by the choice of $\beta = -v_y$ and $\delta = v_x$ up to an overall normalising constant. A summary of the orthonormal w-basis, $\{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\}$, is presented in Equations (6.3.21) to (6.3.23) respectively.

$$\mathbf{w}_1 = \frac{1}{c_1}\left[v_x \mathbf{e}_x + v_y \mathbf{e}_y + (v_x^2 + v_y^2)\mathbf{e}_z\right], \quad \text{where } c_1^2 = v_x^2 + v_y^2 + (v_x^2 + v_y^2)^2 \tag{6.3.21}$$

$$\mathbf{w}_2 = \frac{1}{c_2}\left[-v_y \mathbf{e}_x + v_x \mathbf{e}_y\right], \quad \text{where } c_2^2 = v_x^2 + v_y^2 \tag{6.3.22}$$

$$\mathbf{w}_3 = \frac{1}{c_3}\left[-v_x \mathbf{e}_x - v_y \mathbf{e}_y + \mathbf{e}_z\right], \quad \text{where } c_3^2 = v_x^2 + v_y^2 + 1 \tag{6.3.23}$$

Proposition 6.3.1. *If $\Sigma_{3\text{D}}$ is specified to be diagonal in the w-basis, then it will satisfy the desired projection constraint given by Equation* (6.3.17)*.*

Proof. We proceed to certify this proposition subject to some additional conditions.

Suppose that $\Sigma_{3\text{D}}$ is diagonal with respect to the $\{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\}$ basis with eigenvalues $\{\alpha_1^2, \alpha_2^2, \alpha_3^2\}$ respectively, then the eigendecomposition of $\Sigma_{3\text{D}}$ is presented in Equation (6.3.24) where $W \in \mathbb{R}^{3\times 3}$ is such that the i^{th} column contains basis vector $\mathbf{w}_i$, and $\Lambda \in \mathbb{R}^{3\times 3}$ is a diagonal matrix with $\Lambda_{ii} = \alpha_i^2$. Since $\{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\}$ form an

orthonormal basis, $W^{-1} = W^{\mathrm{T}}$ in the eigendecomposition.

$$\Sigma_{3\mathrm{D}} = W\Lambda W^{\mathrm{T}} = \alpha_1^2 \mathbf{w}_1 \mathbf{w}_1^{\mathrm{T}} + \alpha_2^2 \mathbf{w}_2 \mathbf{w}_2^{\mathrm{T}} + \alpha_3^2 \mathbf{w}_3 \mathbf{w}_3^{\mathrm{T}} \tag{6.3.24}$$

This is used to obtain $\Sigma_{3\mathrm{D}}^{-1}$ in Equation (6.3.25).

$$\Sigma_{3\mathrm{D}}^{-1} = W\Lambda^{-1}W^{\mathrm{T}} = \frac{1}{\alpha_1^2}\mathbf{w}_1 \mathbf{w}_1^{\mathrm{T}} + \frac{1}{\alpha_2^2}\mathbf{w}_2 \mathbf{w}_2^{\mathrm{T}} + \frac{1}{\alpha_3^2}\mathbf{w}_3 \mathbf{w}_3^{\mathrm{T}} \tag{6.3.25}$$

To determine the form of $\Sigma_{3\mathrm{D}}$ such that Equation (6.3.17) is satisfied, it is necessary to evaluate $A^{\mathrm{T}}\Sigma_{3\mathrm{D}}^{-1}A$. This is performed in stages by considering the result of $\mathbf{w}_i^{\mathrm{T}}A$ for $i = 1, 2, 3$, where it is first noted that $\mathbf{w}_3^{\mathrm{T}}A = \begin{pmatrix} 0 & 0 \end{pmatrix}$ in Equation (6.3.26).

$$\mathbf{w}_3^{\mathrm{T}}A = \frac{1}{c_3}\begin{pmatrix} -v_x & -v_y & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ v_x & v_y \end{pmatrix} = \begin{pmatrix} 0 & 0 \end{pmatrix} \tag{6.3.26}$$

Substituting Equation (6.3.26) into $A^{\mathrm{T}}\Sigma_{3\mathrm{D}}^{-1}A$ yields the following result:

$$\begin{aligned} A^{\mathrm{T}}\Sigma_{3\mathrm{D}}^{-1}A &= A^{\mathrm{T}}\left(\frac{1}{\alpha_1^2}\mathbf{w}_1\mathbf{w}_1^{\mathrm{T}} + \frac{1}{\alpha_2^2}\mathbf{w}_2\mathbf{w}_2^{\mathrm{T}} + \frac{1}{\alpha_3^2}\mathbf{w}_3\mathbf{w}_3^{\mathrm{T}}\right)A \\ &= \frac{1}{\alpha_1^2}A^{\mathrm{T}}\mathbf{w}_1\mathbf{w}_1^{\mathrm{T}}A + \frac{1}{\alpha_2^2}A^{\mathrm{T}}\mathbf{w}_2\mathbf{w}_2^{\mathrm{T}}A \end{aligned} \tag{6.3.27}$$

It is evident that $A^{\mathrm{T}}\Sigma_{3\mathrm{D}}^{-1}A$ is independent of α_3 for which the interpretation and implications are further discussed after this proof.

Completing the derivation of $\Sigma_{3\mathrm{D}}$ requires the remaining terms in Equation (6.3.27) to be evaluated. The results for the $\mathbf{w}_2$ component is derived in Equations (6.3.28) and (6.3.29), introducing the notation $r^2 = v_x^2 + v_y^2 = c_2^2$.

$$\mathbf{w}_2^{\mathrm{T}}A = \frac{1}{c_2}\begin{pmatrix} -v_y & v_x & 0 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ v_x & v_y \end{pmatrix} = \frac{1}{c_2}\begin{pmatrix} -v_y & v_x \end{pmatrix} \tag{6.3.28}$$

$$\implies \frac{1}{\alpha_2^2}A^{\mathrm{T}}\mathbf{w}_2\mathbf{w}_2^{\mathrm{T}}A = \frac{1}{\alpha_2^2 c_2^2}\begin{pmatrix} -v_y \\ v_x \end{pmatrix}\begin{pmatrix} -v_y & v_x \end{pmatrix} = \frac{1}{\alpha_2^2 r^2}\begin{pmatrix} v_y^2 & -v_x v_y \\ -v_x v_y & v_x^2 \end{pmatrix} \tag{6.3.29}$$

The results for the $\mathbf{w}_1$ component are presented in Equations (6.3.30) and (6.3.31)

where $c_1^2 = r^2 + r^4 = r^2(1 + r^2)$.

$$\begin{aligned}\mathbf{w}_1^{\mathrm{T}} A &= \frac{1}{c_1}\begin{pmatrix} v_x & v_y & v_x^2 + v_y^2 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ v_x & v_y \end{pmatrix} \\ &= \frac{1}{c_1}\begin{pmatrix} v_x(1+r^2) & v_y(1+r^2) \end{pmatrix} = \frac{1+r^2}{c_1}\begin{pmatrix} v_x & v_y \end{pmatrix} \qquad (6.3.30)\end{aligned}$$

$$\implies \frac{1}{\alpha_1^2} A^{\mathrm{T}} \mathbf{w}_1 \mathbf{w}_1^{\mathrm{T}} A = \frac{(1+r^2)^2}{\alpha_1^2 c_1^2}\begin{pmatrix} v_x \\ v_y \end{pmatrix}\begin{pmatrix} v_x & v_y \end{pmatrix} = \frac{(1+r^2)}{\alpha_1^2 r^2}\begin{pmatrix} v_x^2 & v_x v_y \\ v_x v_y & v_y^2 \end{pmatrix} \qquad (6.3.31)$$

Combining Equations (6.3.27), (6.3.29) and (6.3.31) and substituting into the projection constraint in Equation (6.3.17) yields the result in Equation (6.3.32).

$$\begin{aligned}&\Sigma_{2\mathrm{D}} = A^{\mathrm{T}} \Sigma_{3\mathrm{D}}^{-1} A \\ \iff &\begin{pmatrix} \frac{1}{\theta^2} & 0 \\ 0 & \frac{1}{\theta^2} \end{pmatrix} = \frac{(1+r^2)}{\alpha_1^2 r^2}\begin{pmatrix} v_x^2 & v_x v_y \\ v_x v_y & v_y^2 \end{pmatrix} + \frac{1}{\alpha_2^2 r^2}\begin{pmatrix} v_y^2 & -v_x v_y \\ -v_x v_y & v_x^2 \end{pmatrix} \\ \iff &\begin{pmatrix} \frac{1}{\theta^2} & 0 \\ 0 & \frac{1}{\theta^2} \end{pmatrix} = \frac{1}{r^2}\begin{pmatrix} \frac{v_y^2}{\alpha_2^2} + \frac{(1+r^2)v_x^2}{\alpha_1^2} & \left(\frac{(1+r^2)}{\alpha_1^2} - \frac{1}{\alpha_2^2}\right) v_x v_y \\ \left(\frac{(1+r^2)}{\alpha_1^2} - \frac{1}{\alpha_2^2}\right) v_x v_y & \frac{v_x^2}{\alpha_2^2} + \frac{(1+r^2)v_y^2}{\alpha_1^2} \end{pmatrix} \qquad (6.3.32)\end{aligned}$$

Equation (6.3.33) shows the results of equating the off-diagonal terms in order to determine α_1 and α_2.

$$\left(\frac{(1+r^2)}{\alpha_1^2} - \frac{1}{\alpha_2^2}\right) v_x v_y = 0 \qquad (6.3.33)$$

$$\iff \textbf{Case 1: } \alpha_1^2 = \alpha_2^2(1+r^2), \text{ or } \textbf{Case 2: } v_x = 0, \text{ or } \textbf{Case 3: } v_y = 0$$

For **Case 1**, inserting $\alpha_1^2 = \alpha_2^2(1+r^2)$ into Equation (6.3.32) yields Equation (6.3.34) where $\alpha_2^2 = \theta^2$.

$$\begin{pmatrix} \frac{1}{\theta^2} & 0 \\ 0 & \frac{1}{\theta^2} \end{pmatrix} = \frac{1}{r^2}\begin{pmatrix} \frac{v_y^2}{\alpha_2^2} + \frac{(1+r^2)v_x^2}{\alpha_2^2(1+r^2)} & 0 \\ 0 & \frac{v_x^2}{\alpha_2^2} + \frac{(1+r^2)v_y^2}{\alpha_2^2(1+r^2)} \end{pmatrix} = \begin{pmatrix} \frac{1}{\alpha_2^2} & 0 \\ 0 & \frac{1}{\alpha_2^2} \end{pmatrix} \iff \alpha_2^2 = \theta^2 \qquad (6.3.34)$$

For **Case 2**, $v_x = 0$ implies $r^2 = v_y^2$ and inserting into Equation (6.3.32) yields Equation (6.3.35) which is satisfied by choosing $\alpha_2^2 = \theta^2$ and $\alpha_1^2 = \alpha_2^2(1+r^2)$. This

is the same result as for **Case 1**.

$$\begin{pmatrix} \frac{1}{\theta^2} & 0 \\ 0 & \frac{1}{\theta^2} \end{pmatrix} = \begin{pmatrix} \frac{1}{\alpha_2^2} & 0 \\ 0 & \frac{(1+r^2)}{\alpha_1^2} \end{pmatrix} \iff \alpha_2^2 = \theta^2 \text{ and } \alpha_1^2 = \alpha_2^2(1+r^2) \tag{6.3.35}$$

For **Case 3** the same answer is obtained due to the symmetry between x and y, but with $r^2 = v_x^2$. The projection constraint is therefore adhered to by specifying the first two eigenvalues α_1^2 and α_2^2 of $\Sigma_{3\mathrm{D}}$ as in Equation (6.3.36), with $r^2 = v_x^2 + v_y^2$.

$$\alpha_1^2 = \theta^2(1+r^2) \text{ and } \alpha_2^2 = \theta^2 \tag{6.3.36}$$

This demonstrates that the postulated form of $\Sigma_{3\mathrm{D}}$ in Equation (6.3.24) is valid. □

The proposed form of $\Sigma_{3\mathrm{D}}$ has a geometric interpretation with respect to the w-basis. Firstly, since $\mathbf{w}_2$ points in the direction for which the embedding surface $v(x,y)$ is not locally increasing at $\mathbf{x}_0$, there will be no warping or compression of the emulator along this direction and hence α_2^2 must equal the desired 2D correlation length, θ^2. Conversely, $\mathbf{w}_1$ is defined to point in the direction of steepest increase of $v(x,y)$ at $\mathbf{x}_0$ where the gradient in this direction is $\|\nabla v(x,y)\| = r$. This induces a warping along $\mathbf{w}_1$ due to the use of a stretched embedding surface and is annulled by increasing α_1^2 accordingly. Only points that lie on the surface described by $v(x,y)$ are of interest for the embedded emulator. As $\mathbf{w}_3$ is stipulated to be orthogonal to $v(x,y)$ at $\mathbf{x}_0$, there must be no constraint imposed on α_3^2 by this construction, as confirmed in Equation (6.3.26). It is therefore a free parameter which can be chosen or inferred to govern the decorrelation of the emulator across the discontinuities. Note that this is a major motivation for choosing the proposed form of $\Sigma_{3\mathrm{D}}$ given in Equation (6.3.24).

Proposition 6.3.2. *The form of $\Sigma_{3\mathrm{D}}$ for the embedded emulator close to $\mathbf{x}_0$ in the standard Cartesian basis which induces the desired projected covariance is:*

$$\Sigma_{3\mathrm{D}}(\mathbf{x}_0) = \begin{pmatrix} \theta^2 + \frac{\alpha_3^2 v_x^2}{r^2+1} & \frac{\alpha_3^2 v_x v_y}{r^2+1} & v_x\left(\theta^2 - \frac{\alpha_3^2}{r^2+1}\right) \\ \frac{\alpha_3^2 v_x v_y}{r^2+1} & \theta^2 + \frac{\alpha_3^2 v_x^2}{r^2+1} & v_y\left(\theta^2 - \frac{\alpha_3^2}{r^2+1}\right) \\ v_x\left(\theta^2 - \frac{\alpha_3^2}{r^2+1}\right) & v_y\left(\theta^2 - \frac{\alpha_3^2}{r^2+1}\right) & \theta^2 r^2 + \frac{\alpha_3^2}{r^2+1} \end{pmatrix} \tag{6.3.37}$$

Proof. The components of the Σ_{3D} eigendecomposition in Equation (6.3.24) are computed in Equations (6.3.38) to (6.3.40) respectively, using the defined w-basis in Equations (6.3.21) to (6.3.23), along with the constrained eigenvalues in Equation (6.3.36).

$$\alpha_1^2 \mathbf{w}_1 \mathbf{w}_1^{\mathrm{T}} = \frac{\alpha_1^2}{c_1^2} \begin{pmatrix} v_x \\ v_y \\ v_x^2 + v_y^2 \end{pmatrix} \begin{pmatrix} v_x & v_y & v_x^2 + v_y^2 \end{pmatrix} = \frac{\theta^2}{r^2} \begin{pmatrix} v_x^2 & v_x v_y & v_x r^2 \\ v_x v_y & v_y^2 & v_y r^2 \\ v_x r^2 & v_y r^2 & r^4 \end{pmatrix} \tag{6.3.38}$$

$$\alpha_2^2 \mathbf{w}_2 \mathbf{w}_2^{\mathrm{T}} = \frac{\alpha_2^2}{c_2^2} \begin{pmatrix} -v_y \\ v_x \\ 0 \end{pmatrix} \begin{pmatrix} -v_y & v_x & 0 \end{pmatrix} = \frac{\theta^2}{r^2} \begin{pmatrix} v_y^2 & -v_x v_y & 0 \\ -v_x v_y & v_x^2 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{6.3.39}$$

$$\alpha_3^2 \mathbf{w}_3 \mathbf{w}_3^{\mathrm{T}} = \frac{\alpha_3^2}{c_3^2} \begin{pmatrix} -v_x \\ -v_y \\ 1 \end{pmatrix} \begin{pmatrix} -v_x & v_y & 1 \end{pmatrix} = \frac{\alpha_3^2}{r^2 + 1} \begin{pmatrix} v_x^2 & v_x v_y & -v_x \\ v_x v_y & v_y^2 & -v_y \\ -v_x & -v_y & 1 \end{pmatrix} \tag{6.3.40}$$

Substitution into Equation (6.3.24) yields the expression for $\Sigma_{3D}(\mathbf{x}_0)$ in Equation (6.3.37). □

The expression for $\Sigma_{3D}(\mathbf{x}_0)$ possesses an explicit dependence on $\mathbf{x}_0$ and the chosen embedding surface, $v(x, y)$, via its partial derivatives. This expression is substituted into the covariance structure of the embedded emulator, for example, the squared exponential covariance function in Equation (6.3.5). For a specified embedding function and any $\mathbf{x}$ close to $\mathbf{x}_0$, Equation (6.3.37) induces the desired covariance structure, Σ_{2D}, in Equation (6.3.11). Note that for embeddings with zero curvature, this correction is exact. In Section 6.3.3 $\Sigma_{3D}(\mathbf{x}_0)$ is incorporated within non-stationary emulators to obtain a global emulator.

Constructing Σ_{3D} for general Σ_{2D}

The framework of constructing emulators over a higher-dimensional embedded parameter space is derived above for the common scenario where Σ_{2D} is of the form specified in Equation (6.3.11); that is diagonal with eigenvalue θ^2 of multiplicity two. We now extend this to general choices of Σ_{2D} via a transformation of the 2-dimensional parameter space to which the above derivation is applied, thus obtaining the corresponding Σ_{3D}.

First, define a rotated t-basis in Equations (6.3.41) and (6.3.42), where the angle $\phi \in [0, \pi)$ is chosen such that the transformed parameters are uncorrelated.

$$\mathbf{t}_1 = \cos(\phi)\mathbf{e}_x + \sin(\phi)\mathbf{e}_y \tag{6.3.41}$$

$$\mathbf{t}_2 = -\sin(\phi)\mathbf{e}_x + \cos(\phi)\mathbf{e}_y \tag{6.3.42}$$

The rotation from the standard basis to the t-basis is described in Equation (6.3.43), where $\mathbf{p} = \binom{p_1}{p_2}$ and $\mathbf{x} = \binom{x}{y}$ denote the same parameter vector expressed in the t- and standard basis respectively.

$$\begin{pmatrix} p_1 \\ p_2 \end{pmatrix} = \begin{pmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{6.3.43}$$

Similarly, this rotational transformation is reversed using Equation (6.3.44).

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \tag{6.3.44}$$

Operating in the t-basis decorrelates the parameters, hence the corresponding covariance matrix, denoted by $\Sigma_{2D}^{\mathbf{t}}$, is of the following form:

$$\Sigma_{2D}^{\mathbf{t}} = \begin{pmatrix} \theta_1^2 & 0 \\ 0 & \theta_2^2 \end{pmatrix} \tag{6.3.45}$$

The form of $\Sigma_{2D}^{\mathbf{t}}$ is diagonal, but has eigenvalues θ_1^2 and θ_2^2, which are potentially distinct. This is not yet in the required setup described above for which the derived embedding correction applies.

In order to obtain a Σ_{2D} variance matrix of the form presented in Equation (6.3.11), it is also necessary to perform a scaling to attain two equal eigenvalues. For this purpose, define the scaled h-basis by multiplying the t-basis by their respective eigenvalues in Equation (6.3.46).

$$\mathbf{h}_1 = \theta_1 \mathbf{t}_1 \quad \text{and} \quad \mathbf{h}_2 = \theta_2 \mathbf{t}_2 \tag{6.3.46}$$

A vector, $\mathbf{p}$, in the t-basis may then be expressed as $\mathbf{q} = \left(\begin{smallmatrix} q_1 \\ q_2 \end{smallmatrix}\right)$ in the h-basis as follows:

$$\begin{pmatrix} p_1 \\ \\ p_2 \end{pmatrix} = p_1 \mathbf{t}_1 + p_2 \mathbf{t}_2 = \frac{p_1}{\theta_1}\mathbf{h}_1 + \frac{p_2}{\theta_2}\mathbf{h}_2 = \begin{pmatrix} \frac{p_1}{\theta_1} \\ \\ \frac{p_2}{\theta_2} \end{pmatrix} = \begin{pmatrix} q_1 \\ \\ q_2 \end{pmatrix} \tag{6.3.47}$$

The transformation between the t- and h-bases is described in Equation (6.3.48). Combining with Equation (6.3.43) yields a transformation between the standard and h-bases presented in Equation (6.3.49).

$$\begin{pmatrix} q_1 \\ \\ q_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{\theta_1} & 0 \\ \\ 0 & \frac{1}{\theta_2} \end{pmatrix} \begin{pmatrix} p_1 \\ \\ p_2 \end{pmatrix} \tag{6.3.48}$$

$$= \begin{pmatrix} \frac{1}{\theta_1} & 0 \\ \\ 0 & \frac{1}{\theta_2} \end{pmatrix} \begin{pmatrix} \cos(\phi) & \sin(\phi) \\ \\ -\sin(\phi) & \cos(\phi) \end{pmatrix} \begin{pmatrix} x \\ \\ y \end{pmatrix} \tag{6.3.49}$$

The complete transformation incorporates a rotation followed by a scaling of the parameter space when starting from the standard basis. These facets are encapsulated by the scaling and rotation matrices in 2-dimensions, denoted S_{2D} and R_{2D} respectively, which are defined in Equation (6.3.50). Any transformation is then uniquely defined by the appropriate choices of ϕ, θ_1 and θ_2.

$$S_{2D} = \begin{pmatrix} \frac{1}{\theta_1} & 0 \\ \\ 0 & \frac{1}{\theta_2} \end{pmatrix} \quad \text{and} \quad R_{2D} = \begin{pmatrix} \cos(\phi) & \sin(\phi) \\ \\ -\sin(\phi) & \cos(\phi) \end{pmatrix} \tag{6.3.50}$$

Transforming to the h-basis has the desired effect of obtaining a parameterisation where the variance matrix, denoted $\Sigma_{2D}^{\mathbf{h}}$, is both diagonal and has a single eigenvalue with multiplicity two. This is given in Equation (6.3.51), where $\Sigma_{2D}^{\mathbf{h}}$ has a repeated

eigenvalue 1.

$$\Sigma_{2\mathrm{D}}^{\mathbf{h}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{6.3.51}$$

In the h-basis we can embed in 3-dimensions and obtain the variance matrix, $\Sigma_{3\mathrm{D}}^{\mathbf{h}}$, that induces $\Sigma_{2\mathrm{D}}^{\mathbf{h}} = I_2$. This is achieved by applying the above methodology to $\mathbf{q}$ using Equation (6.3.37), but using $v(q_1, q_2)$ instead of $v(x, y)$, and substituting v_x and v_y for v_{q_1} and v_{q_2} respectively. Computation of these partial derivatives will be addressed below in Equation (6.3.57). Also note that since $\Sigma_{2\mathrm{D}}^{\mathbf{h}} = I_2$, this corresponds to $\theta = 1$.

Elevating from 2-dimensions to 3-dimensions is achieved via $\mathbf{q} \mapsto \mathbf{u} = \left(\begin{smallmatrix} q_1 \\ q_2 \\ z \end{smallmatrix}\right)$, where $z = v(q_1, q_2)$, whilst in the corresponding standard basis representation, $\mathbf{x} \mapsto \mathbf{v} = \left(\begin{smallmatrix} x \\ y \\ z \end{smallmatrix}\right)$. Note that $z = v(q_1, q_2) = v(x, y)$ signifying that the embedding surface remains the same irrespective of which 2-dimensional basis is used to evaluate $v(\cdot)$. This setup leads to the scaling and rotation transformation matrices for the embedded space shown in Equation (6.3.52).

$$S_{3\mathrm{D}} = \begin{pmatrix} \frac{1}{\theta_1} & 0 & 0 \\ 0 & \frac{1}{\theta_2} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad R_{3\mathrm{D}} = \begin{pmatrix} \cos(\phi) & \sin(\phi) & 0 \\ -\sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{6.3.52}$$

Note that the upper-left 2×2 sub-matrices correspond to $S_{2\mathrm{D}}$ and $R_{2\mathrm{D}}$ respectively, as defined in Equation (6.3.50). The final row and column represents that there is no change to the third dimension to respect that the embedding surface is the same for both parameterisations.

Proposition 6.3.3. *The variance matrix inverse, $\Sigma_{3\mathrm{D}}^{-1}$, for the embedding of the parameter representation in the standard basis is:*

$$\Sigma_{3\mathrm{D}}^{-1} = R_{3\mathrm{D}}^{\mathrm{T}} S_{3\mathrm{D}}^{\mathrm{T}} \Sigma_{3\mathrm{D}}^{\mathbf{h}^{-1}} S_{3\mathrm{D}} R_{3\mathrm{D}} \tag{6.3.53}$$

where $\Sigma_{3\mathrm{D}}^{\mathbf{h}}$ is the variance matrix in the h-basis representation obtained above.

Proof. Rotation and scaling transformations of the 2-dimensional parameter space

must yield the same covariance structure for the embedded emulator. As in Equation (6.3.17), this constitutes a requirement on the Mahalanobis distance relating Σ_{3D} and $\Sigma_{3D}^{\mathbf{h}}$ for the embedding of the standard and h-basis representations. This is presented in Equation (6.3.54).

$$\mathbf{u}^{\mathrm{T}}\Sigma_{3D}^{\mathbf{h}^{-1}}\mathbf{u} = \mathbf{v}^{\mathrm{T}}\Sigma_{3D}^{-1}\mathbf{v} \tag{6.3.54}$$

Substituting $\mathbf{u} = S_{3D}R_{3D}\mathbf{v}$ into Equation (6.3.54) yields the requirement shown in Equation (6.3.55).

$$\iff \mathbf{v}^{\mathrm{T}}R_{3D}^{\mathrm{T}}S_{3D}^{\mathrm{T}}\Sigma_{3D}^{\mathbf{h}^{-1}}S_{3D}R_{3D}\mathbf{v} = \mathbf{v}^{\mathrm{T}}\Sigma_{3D}^{-1}\mathbf{v} \tag{6.3.55}$$

$$\implies \Sigma_{3D}^{-1} = R_{3D}^{\mathrm{T}}S_{3D}^{\mathrm{T}}\Sigma_{3D}^{\mathbf{h}^{-1}}S_{3D}R_{3D} \tag{6.3.56}$$

This is the form presented in Equation (6.3.53). □

The final step in evaluating $\Sigma_{3D}^{\mathbf{h}}$ via Equation (6.3.37) in the h-basis is to compute v_{q_1} and v_{q_2}. This is presented in the following proposition.

Proposition 6.3.4. *The partial derivatives $v_{q_1} = \frac{\partial v}{\partial q_1}$ and $v_{q_2} = \frac{\partial v}{\partial q_2}$ in the h-basis representation are obtained via:*

$$\begin{pmatrix} v_{q_1} \\ v_{q_2} \end{pmatrix} = \begin{pmatrix} \theta_1\cos(\phi) & \theta_1\sin(\phi) \\ -\theta_2\sin(\phi) & \theta_2\cos(\phi) \end{pmatrix} \begin{pmatrix} v_x \\ v_y \end{pmatrix} \tag{6.3.57}$$

Proof. The partial derivatives v_{q_1} and v_{q_2} are computed in Equations (6.3.58) and (6.3.59) respectively, with an equivalent matrix representation in Equation (6.3.60).

$$v_{q_1} = \frac{\partial v}{\partial q_1} = \frac{\partial v(x,y)}{\partial x}\frac{\partial x}{\partial q_1} + \frac{\partial v(x,y)}{\partial y}\frac{\partial y}{\partial q_1} \tag{6.3.58}$$

$$v_{q_2} = \frac{\partial v}{\partial q_2} = \frac{\partial v(x,y)}{\partial x}\frac{\partial x}{\partial q_2} + \frac{\partial v(x,y)}{\partial y}\frac{\partial y}{\partial q_2} \tag{6.3.59}$$

$$\iff \begin{pmatrix} v_{q_1} \\ v_{q_2} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial y}{\partial q_1} \\ \frac{\partial x}{\partial q_2} & \frac{\partial y}{\partial q_2} \end{pmatrix} \begin{pmatrix} v_x \\ v_y \end{pmatrix} \tag{6.3.60}$$

Define M to be to be a 2-dimensional Jacobian matrix which characterises the transformation between the partial derivatives of $v(\cdot)$ with respect to the two basis

representations:

$$M = \begin{pmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial y}{\partial q_1} \\ \frac{\partial x}{\partial q_2} & \frac{\partial y}{\partial q_2} \end{pmatrix} \tag{6.3.61}$$

The construction $\mathbf{q} = S_{2D}R_{2D}\mathbf{x}$ is inverted to obtain $\mathbf{x} = R_{2D}^{-1}S_{2D}^{-1}\mathbf{q}$ and is used to evaluate the partial derivatives and derive a general expression for M for transformations composed of a scaling and rotation. This is given in Equation (6.3.62).

$$\begin{aligned} M &= \begin{pmatrix} \left(\frac{\partial \mathbf{x}}{\partial q_1}\right)^{\mathrm{T}} \\ \left(\frac{\partial \mathbf{x}}{\partial q_2}\right)^{\mathrm{T}} \end{pmatrix} = \left(R_{2D}^{-1}S_{2D}^{-1}\right)^{\mathrm{T}} = S_{2D}^{-1}\left(R_{2D}^{-1}\right)^{\mathrm{T}} \\ &= \begin{pmatrix} \theta_1 & 0 \\ 0 & \theta_2 \end{pmatrix} \begin{pmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{pmatrix}^{\mathrm{T}} = \begin{pmatrix} \theta_1 & 0 \\ 0 & \theta_2 \end{pmatrix} \begin{pmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{pmatrix} \\ &= \begin{pmatrix} \theta_1\cos(\phi) & \theta_1\sin(\phi) \\ -\theta_2\sin(\phi) & \theta_2\cos(\phi) \end{pmatrix} \end{aligned} \tag{6.3.62}$$

Combining Equations (6.3.60) and (6.3.62) yields the formula for v_{q_1} and v_{q_2} in Equation (6.3.57), and thus completes the computation of $\Sigma_{3D}^{\mathbf{h}}$. □

The above derivation, summarised in Propositions 6.3.3 and 6.3.4, shows how to obtain Σ_{3D} for a general choice of Σ_{2D} by first implementing rotation and scaling transformations to decorrelate the parameters and rescale to a common correlation length, as in Equation (6.3.49). The previously described methodology yielding Equation (6.3.37) is then directly applicable utilising Equation (6.3.57) to evaluate the partial derivatives, before substituting into Equation (6.3.53) to construct the general Σ_{3D}.

6.3.3 Controlling for the Global Impact of the Embedding Using Non-Stationary Emulation

In Section 6.3.2 we derived the general form of $\Sigma_{3D}(\mathbf{x}_0)$ (see Equation (6.3.37)) to correct for the impact of the embedding surface on the emulator's covariance structure. However, this is a local correction for $\mathbf{x}$ near to $\mathbf{x}_0$, as v_x, v_y and r^2 are

all evaluated at $\mathbf{x}_0$. This is insufficient for computing covariances between pairs of input parameter settings and is thus extended to a global emulator by employing a non-stationary covariance structure.

For generality, define $\Sigma_{3\mathrm{D}}(\mathbf{x})$ to be an $\mathbf{x}$ dependent covariance matrix of the form given in Equation (6.3.37) with $\mathbf{x}_0 = \mathbf{x}$. This will vary over the parameter space for general embeddings $v(x, y)$, except for a linear embedding. The non-stationary apparatus presented in [37, 136] are employed to define a valid covariance structure. These are defined via a non-stationary extension to valid isotropic covariance functions in a form of averaging of the $\mathbf{x}$ dependent covariance matrix $\Sigma(\mathbf{x})$,[2] whilst guaranteeing a valid covariance structure over the entire parameter space. In general, let $\Sigma\colon \mathbb{R}^d \to \mathbb{R}^{d\times d}$ such that $\Sigma(\mathbf{x})$ is symmetric and positive definite for all $\mathbf{x} \in \mathbb{R}^d$. First introduce the quadratic form $Q(\mathbf{x}, \mathbf{x}')$ in Equation (6.3.63).

$$Q(\mathbf{x}, \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^{\mathrm{T}} \left(\frac{\Sigma(\mathbf{x}) + \Sigma(\mathbf{x}')}{2} \right)^{-1} (\mathbf{x} - \mathbf{x}'), \quad \mathbf{x}, \mathbf{x}' \in \mathbb{R}^d \tag{6.3.63}$$

Let $\kappa_{\mathrm{S}}(\cdot, \cdot)$ be a valid stationary isotropic covariance function corresponding to $\Sigma(\cdot)$. The non-stationary analogue, denoted by $\kappa_{\mathrm{NS}}(\cdot, \cdot)$, is then defined in Equation (6.3.64) which is positive definite on $\mathbb{R}^{d\times d}$ and thus establishes a valid covariance function.

$$\kappa_{\mathrm{NS}}(\mathbf{x}, \mathbf{x}') = \frac{2^{\frac{d}{2}} |\Sigma(\mathbf{x})|^{\frac{1}{4}} |\Sigma(\mathbf{x}')|^{\frac{1}{4}}}{|\Sigma(\mathbf{x}) + \Sigma(\mathbf{x}')|^{\frac{1}{2}}} \kappa_{\mathrm{S}}(\mathbf{x}, \mathbf{x}') \tag{6.3.64}$$

For example, the corresponding non-stationary squared exponential covariance function for $\mathbf{x} \in \mathbb{R}^3$ is:

$$\mathrm{Cov}[f(\mathbf{x}), f(\mathbf{x}')] = \sigma^2 \frac{2^{\frac{3}{2}} |\Sigma(\mathbf{x})|^{\frac{1}{4}} |\Sigma(\mathbf{x}')|^{\frac{1}{4}}}{|\Sigma(\mathbf{x}) + \Sigma(\mathbf{x}')|^{\frac{1}{2}}} \exp\{-Q(\mathbf{x}, \mathbf{x}')\} \tag{6.3.65}$$

Within the TENSE methodology, this non-stationary structure is used to evaluate the covariance on the embedding surface $v(x, y)$ in the 3-dimensional space, hence the quadratic form is defined in terms of the position vector on the embedding

[2] $\Sigma(\mathbf{x})$ corresponds to $\Sigma_{3\mathrm{D}}(\mathbf{x})$ in the TENSE approach derivation.

surface, $\mathbf{v}(\mathbf{x})$, as follows:

$$Q(\mathbf{v}(\mathbf{x}), \mathbf{v}(\mathbf{x}')) = (\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{x}'))^{\mathrm{T}} \left(\frac{\Sigma_{3\mathrm{D}}(\mathbf{v}(\mathbf{x})) + \Sigma_{3\mathrm{D}}(\mathbf{v}(\mathbf{x}'))}{2} \right)^{-1} (\mathbf{v}(\mathbf{x}) - \mathbf{v}(\mathbf{x}')) \tag{6.3.66}$$

The corresponding non-stationary squared exponential covariance function over the embedded 3-dimensional parameter space is described in Equation (6.3.67) where $\Sigma_{3\mathrm{D}}(\mathbf{v}(\mathbf{x})) = \Sigma_{3\mathrm{D}}(\mathbf{x})$ is given by Equation (6.3.37), with $\mathbf{x}_0 = \mathbf{x}$.

$$\mathrm{Cov}[f(\mathbf{v}(\mathbf{x})), f(\mathbf{v}(\mathbf{x}'))] = \sigma^2 \frac{2^{\frac{3}{2}} |\Sigma_{3\mathrm{D}}(\mathbf{v}(\mathbf{x}))|^{\frac{1}{4}} |\Sigma_{3\mathrm{D}}(\mathbf{v}(\mathbf{x}'))|^{\frac{1}{4}}}{|\Sigma_{3\mathrm{D}}(\mathbf{v}(\mathbf{x})) + \Sigma_{3\mathrm{D}}(\mathbf{v}(\mathbf{x}'))|^{\frac{1}{2}}} \exp\{-Q(\mathbf{v}(\mathbf{x}), \mathbf{v}(\mathbf{x}'))\} \tag{6.3.67}$$

This guarantees a valid covariance structure throughout both the 3-dimensional embedded and the induced 2-dimensional parameter spaces.

For any pair of input points $\mathbf{x}$ and $\mathbf{x}'$ that are close together relative to the curvature of the embedding surface, the non-stationary covariance structure in Equation (6.3.67) yields a first order correction to the local warping effect of the chosen embedding surface. It is possible that for pairs of input points with a greater separation that the non-linear effects may become noticeable. However, for modest choices of correlation length, θ, these effects will typically be suppressed as the covariance tends towards zero for separations greater than θ, thus in practical application this is unlikely to be an issue. This complete framework using the non-stationary covariance structure in Equation (6.3.67) is referred to as the Torn Embedding Non-Stationary Emulator (TENSE) approach. It enables the flexible choice from a wide class of torn embedding surfaces $v(x, y)$ to handle finite collections of structured partial discontinuities that ensure the emulator is decorrelated across them. Moreover, a TENSE emulator approximately induces the desired stationary 2-dimensional covariance structure across local regions that do not contain discontinuities, and guarantees a valid covariance structure everywhere.

In Figures 6.4e and 6.4f the TENSE framework is applied to the second example function in Equation (6.3.7) and discussed in Section 6.3.1. Comparing with the

uncorrected version, as seen in Figures 6.4c and 6.4d respectively, it is evident that the emulator adjusted standard deviation now exhibits no perceivable warping effects and maintains the expected symmetry close to each design point (denoted by the black points), whilst also displaying suitable behaviour across the discontinuities. Furthermore, in locations such as the top-left corner where there are no discontinuities, it is observed that neighbouring design points are informative between them. In contrast, along the partial discontinuities the influence of design points decreases as the distance from the discontinuity endpoint increases. Similar behaviour is observed for the emulator adjusted expectation in Figure 6.4e which exhibits no apparent distortion or warping, whilst the magnitude of discontinuity increases with the distance to the right starting from the left-endpoint of each of the discontinuities.

The framework developed in Section 6.3 is demonstrated for a 2-dimensional parameter space embedded in 3-dimensions. The methodology can be extended to handle various scenarios. For example, if a non-stationary induced 2-dimensional covariance structure is desired, this can be achieved using a similar strategy. Likewise, the torn embedding strategy generalises to higher-dimensional parameter spaces possessing more complex discontinuities by embedding a d-dimensional parameter space in which discontinuities reside on $(d-1)$-dimensional hypersurfaces, in to $(d+1)$-dimensions. For more complex networks of discontinuities, this may require embedding in a higher $(d+m)$-dimensional space to avoid unwanted effects across neighbouring discontinuities. Note that as extra dimensions are used locally, this does not create computational issues from working in high-dimensional spaces.

6.4 2-Dimensional Example with Non-Linear Partial Discontinuities

Two examples of known scalar functions with 2-dimensional input spaces possessing partial discontinuities are introduced in Equations (6.2.1) and (6.3.7), with results illustrated in Figures 6.3 and 6.4 respectively, in order to motivate and explicate the

development of the TENSE approach. In both instances the partial discontinuities are straight lines that are parallel to the x-axis. The flexibility and generality of this methodology is emphasised in this section for a substantially more complex function possessing a collection of four non-linear partial discontinuities. These are described using a region identifier function $m(\mathbf{x})$ in Equation (6.4.1) via the intersection of several circles centred on the points $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, $\mathbf{d}$, and the origin.

$$m(\mathbf{x}) = \begin{cases} 0 & \text{if} \quad |\mathbf{x}| < 0.4 \\ 1 & \text{if} \quad |\mathbf{x}| > 0.4 \cap |\mathbf{x}-\mathbf{a}|^2 < 1 \cap |\mathbf{x}-\mathbf{b}|^2 > 1 \\ 2 & \text{if} \quad |\mathbf{x}| > 0.4 \cap |\mathbf{x}-\mathbf{b}|^2 < 1 \cap |\mathbf{x}-\mathbf{c}|^2 > 1 \\ 3 & \text{if} \quad |\mathbf{x}| > 0.4 \cap |\mathbf{x}-\mathbf{c}|^2 < 1 \cap |\mathbf{x}-\mathbf{d}|^2 > 1 \\ 4 & \text{if} \quad |\mathbf{x}| > 0.4 \cap |\mathbf{x}-\mathbf{d}|^2 < 1 \cap |\mathbf{x}-\mathbf{a}|^2 > 1 \end{cases} \tag{6.4.1}$$

$$\text{with} \qquad \mathbf{a} = (1,0),\ \mathbf{b} = (0,1),\ \mathbf{c} = (-1,0),\ \mathbf{d} = (0,-1)$$

The scalar function is defined in Equation (6.4.2) for $\mathbf{x} \in \mathcal{X} = \{(x,y) \mid -1 < x < 1,\ -1 < y < 1\}$ and is illustrated in Figure 6.5a.

$$f(x,y) \coloneqq \begin{cases} 0.5(\sin(3x) + \cos(3.5y)) & \text{if} \quad m(\mathbf{x}) = 0 \\ 0.5(\sin(3x) + \cos(3.5y)) + (-1)^{m(\mathbf{x})+1}(|\mathbf{x}| - 0.4)^2 & \text{if} \quad m(\mathbf{x}) > 0 \end{cases} \tag{6.4.2}$$

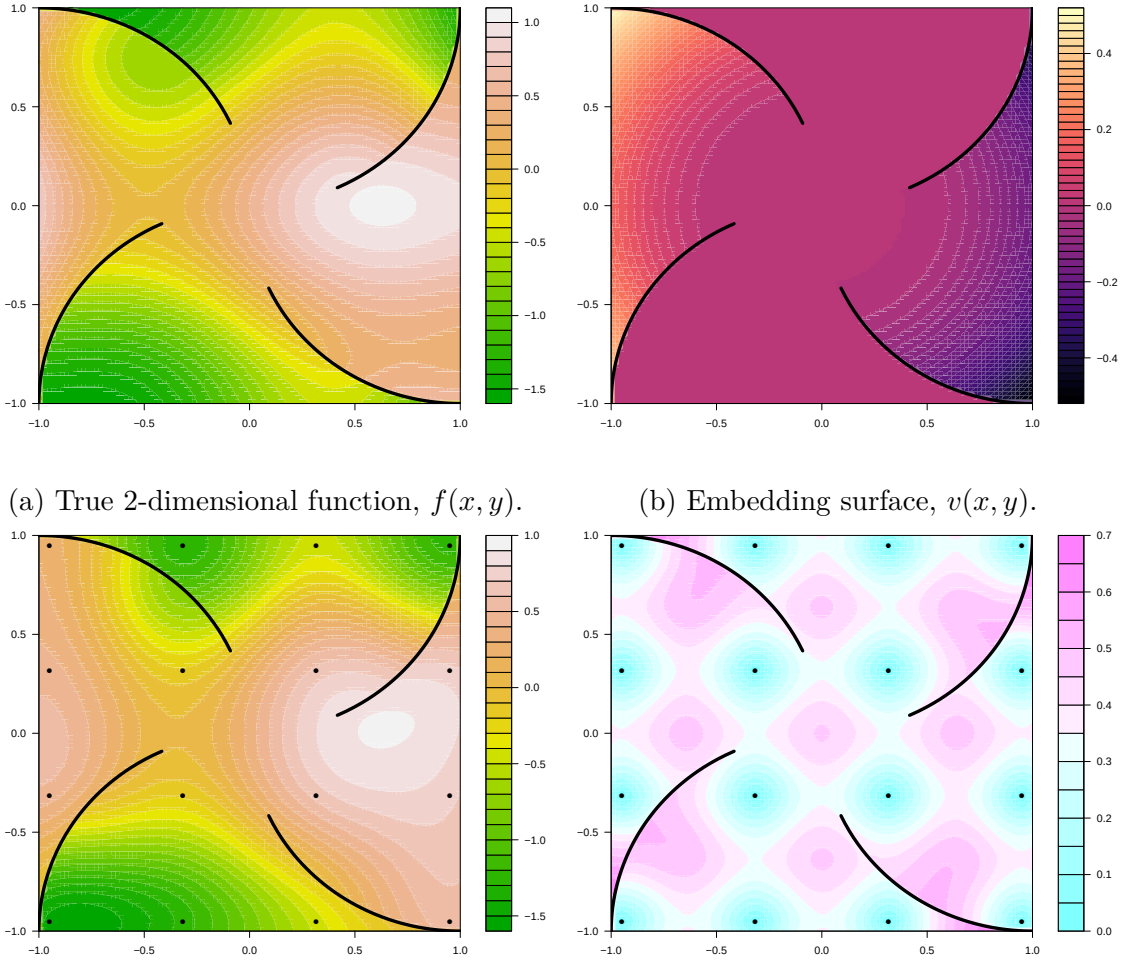
An embedding surface $v(x,y)$ is specified in Equation (6.4.3) to ensure suitable jumps over the locations of the discontinuities. This is shown in Figure 6.5b.

$$v(x,y) = \begin{cases} 0 & \text{if} \quad m(\mathbf{x}) = 0, 2 \text{ or } 4 \\ \frac{1}{2}(m(\mathbf{x}) - 2)(|\mathbf{x}| - 0.4)^2 & \text{if} \quad m(\mathbf{x}) = 1 \text{ or } 3 \end{cases} \tag{6.4.3}$$

General advice on specifying the form of the embedding surface and the suitability of a piecewise quadratic function is provided in Section 7.2.

The TENSE emulation approach is implemented using the non-stationary squared exponential covariance function in Equation (6.3.67), where $\Sigma_{3D}(\mathbf{x})$ is derived using Equation (6.3.37) for the specified embedding function with the desired $\Sigma_{2D} = \text{diag}\{\theta, \theta\}$. This is fitted based on a 16 point grid design to highlight the effects

of the discontinuities, where $\mathbf{D} = \{f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(16)})\}$. The TENSE emulator adjusted expectation, $\mathbb{E}_D[f(\mathbf{x})]$, and standard deviation, $\sqrt{\text{Var}_D[f(\mathbf{x})]}$, are shown in Figures 6.5c and 6.5d respectively. Both plots show an emulator that is continuous everywhere except across the discontinuities where the magnitude of the discontinuities increases with the distance from the interior-endpoint towards the corners of $\mathcal{X}$. Moreover, there is no perceivable warping effect that was previously observed for a naïve embedding approach. A comparison of Figures 6.5a and 6.5c illustrates that the TENSE approach has delivered an accurate emulator based on a small (and non-optimal) design. It is therefore emphasised how the TENSE approach is successfully able to encapsulate complex and non-linear forms of the discontinuities. In contrast, an application of existing methodologies such as treed GPs [74] to $f(x, y)$ would perform poorly due to the inefficiencies in representing curved discontinuities via rectangular regions. To achieve sufficient levels of accuracy would require a large number of simulations rendering this approach computationally intractable, even for functions of modest computation time.



(a) True 2-dimensional function, $f(x, y)$.

(b) Embedding surface, $v(x, y)$.

(c) The TENSE emulator adjusted expectation, $\mathbb{E}_D[f(x, y)]$.

(d) The TENSE emulator adjusted standard deviation, $\sqrt{\text{Var}_D[f(x, y)]}$.

Figure 6.5: Emulation of a 2D function $f(x, y)$ with four curved partial discontinuities presented in Equation (6.4.2) and illustrated in Figure 6.5a. In each plot the partial discontinuities are located along the curved black lines. The embedding surface $v(x, y)$ defined in Equation (6.4.3) is torn along the location of these discontinuities, as seen in Figure 6.5b. The TENSE emulator adjusted expectation, $\mathbb{E}_D[f(x, y)]$, and standard deviation, $\sqrt{\text{Var}_D[f(x, y)]}$, with the warping induced by $v(x, y)$ corrected using NS-GP are shown in Figures 6.5c and 6.5d respectively, where the black points represent the 16 point grid design.

Chapter 7

TNO OLYMPUS Well Placement Optimisation

In this chapter we apply the novel Bayesian emulation of computer models possessing general forms of structured discontinuities methodology developed and presented in Chapter 6 and [174] to the TNO OLYMPUS Well Placement Optimisation Challenge which is reformulated as a decision support problem. A general overview of the TNO OLYMPUS Field Development Optimisation Challenge is provided in Section 3.1.1 for which the TNO OLYMPUS Well Placement Optimisation Challenge is introduced in Section 6.1 as an illustration of a substantial commercial problem requiring such emulation methodology. Further details of the challenge and the exact setup used in this application are presented in Section 7.1. The choice of embedding surface is described in Section 7.2 with the prior covariance structure discussed in Section 7.3. Bayesian emulators that respect the structured partial discontinuities are then implemented to sequentially choose the location of each well in a forwards induction decision analysis. The results for wells 1 to 3 are presented in Sections 7.4 to 7.7, which includes an application of iterative decision support (see Section 3.6) for well 1 and the formulation of sequential designs. This is summarised in a conclusion in Section 7.8. This research is in collaboration with Prof Jonathan Carter, University of Coventry, who has provided petroleum reservoir engineering insight and facilit-

ated the evaluation of simulations for the OLYMPUS ensemble. The problem setup, choice of embedding surface, prior covariance structure, and the results for the first well are also presented in [174].

7.1 Introduction to the TNO OLYMPUS Well Placement Optimisation Challenge

The TNO OLYMPUS Well Placement Optimisation Challenge [93] is formally introduced in Section 6.1 as motivation for the development of Bayesian emulation of computer models with structured discontinuities methodology. This is due to the existence of five geological fault boundaries with potentially variable levels of fluid transmissibility across them at different locations, as illustrated in Figures 6.1 and 6.2, where the endpoint locations are specified in Table 7.1.

Fault Index	Interior Endpoint Location		Boundary Endpoint Location	
	x	y	x	y
1	94	73.5	118	73.5
2	52	85.5	118	85.5
3	64	99.5	118	99.5
4	87	103.5	118	103.5
5	25	123.5	118	123.5

Table 7.1: OLYMPUS geological fault boundary locations used in the TNO OLYMPUS Well Placement Optimisation Challenge corresponding to the transformed map in Figure 6.2 (page 327). The faults are ordered in terms of increasing y-coordinate from the bottom to top of the map. These are expressed in terms of the endpoint coordinates where the interior and boundary endpoints correspond to the left and right endpoints respectively. Note that the 5^{th} geological fault is a sealing fault which divides the entire oil containing region of the OLYMPUS oil field.

The aim is to design a well placement strategy starting from a green oil field which includes the: location; trajectory; and type of each well, along with the number of wells, drilling order of completion and oil platform location, for the fictitious TNO OLYMPUS oil reservoir model which maximises the expected NPV over the 20

year field lifetime (starting January 1, 2016) with discounting applied at 3 month intervals, under the uncertainty captured by the $N = 50$ geological realisations contained within the OLYMPUS ensemble. In order to compare different well placement strategies, a reactive well control strategy is implemented. This consists of maximum production and injection from all wells with shut-in of each producer well connection occurring when the water-cut for that well is breached, and hence it becomes uneconomical. The economic water cut is calculated given the fixed oil production price, water production and injection costs listed in Table 7.2, and is specified to be 88%. If the water as a proportion of total fluid production for a given producer well exceeds 88%, then the well is deactivated and cannot resume operations. Once all producer wells are shut-in, all water injection also stops. This constitutes a complex optimisation problem with a large number of potential decisions represented by parameters to the computer model.

Contribution	Notation	Value
Platform Investment	–	5×10^8 \$
Drilling and Completion	–	$5000 \cdot \lvert\Delta Z\rvert + 1000 \cdot \lvert\Delta XY\rvert$\$
Oil Price	r_{op}	45 \$ per bbl
Water Produced Cost	r_{wp}	6 \$ per bbl
Water Injected Cost	r_{wi}	2 \$ per bbl
Discount Factor	d	0.08

Table 7.2: Expected NPV fixed contribution costs within the TNO OLYMPUS Well Placement Optimisation Challenge. For the drilling and completion contribution to the expected NPV, $|\Delta Z|$ is the change in vertical drilling depth, whilst $|\Delta XY|$ is the lateral drilling length, both measured in metres.

The NPV with decision parameters $\mathbf{d}$ as inputs is computed separately for each OLYMPUS model via the formula in Equation (3.1.1), where $R_j(\mathbf{d}, t_i)$ computes the difference between the revenue accrued from oil production, and the expenditure incurred due to: water production and injection; platform costs; and drilling costs, within each discounting time interval, as defined in Equation (3.1.4). Full details of the terms contained within each formula can be found in Section 3.1.1. Once the NPV has been computed for each OLYMPUS model, the expected NPV is

approximated by the ensemble mean NPV defined in Equation (3.1.2).

The TNO OLYMPUS Well Placement Optimisation Challenge is devised as a test of ensemble optimisers for which we highlight numerous criticisms in Section 3.1.2. As for the TNO OLYMPUS Well Control Optimisation Challenge application in Chapter 4, we believe that this challenge should be reformulated as a decision support problem in terms of a utility function which contains both the financial gains and costs, as well as the decision maker's risk preferences. For the purpose of this application we treat the objective function as equal to the utility function representing a position of risk neutrality. Note that TNO do not provide access to the underlying stochastic OLYMPUS geology model, thus preventing more accurate validation of any well placement strategies. For consistency with the notation introduced for the general TENSE methodology in Chapter 6, a drilling time ordered vector of well locations are denoted by $\mathbf{x}$, where the 2-dimensional location of each well i is such that $(x_i, y_i) \in \mathcal{X} = \{(x, y) \mid 0 < x < 118,\ 0 < y < 181\}$, and the utility function by $f(\mathbf{x}) = \mathbb{E}[\text{NPV}](\mathbf{x}) \approx \overline{\text{NPV}}(\mathbf{x})$. The geological fault boundaries described in Table 7.1 induce discontinuities in $f(\mathbf{x})$ where the first four represent partial discontinuities with an endpoint situated in the interior of the location parameter space, whilst the 5^{th} fault boundary is a sealing fault that divides the OLYMPUS oil field into two separate regions.

Our primary goal in this chapter is to demonstrate the application of the TENSE methodology to the OLYMPUS model to help address the decision support problem. We employ forwards induction for the decision problem which involves selecting successive well locations. This approach is exercised because there exists a very large number of decision parameters, hence it is not computationally feasible to attempt the full backwards induction problem. Moreover, a benefit of the forwards induction approach is that it permits experts to intervene and perturb the locations of previously selected groups of wells in order to obtain a superior solution within the iterative decision support framework. It is acknowledged that this approach is not expected to identify the optimal solution, if this can be properly defined with

respect to the uncertainties present, but will yield a good solution that is close by due to the heavy discounting. Note that in the limit as the discount factor tends to infinity, $d \to \infty$, the forwards induction approach will yield the optimum solution. The TENSE methodology is employed for the ensemble mean NPV as a function of well location which possesses structured partial discontinuities induced by the geological fault boundaries, with potentially variable levels of fluid transmissibility across them at different locations. This is in order to achieve accurate and robust emulation output which respects the structured partial discontinuities across the geological fault boundaries, whilst providing insight into the optimal well locations. The full decision support problem is left to future work, with the focus of this chapter being to sequentially select the first three well locations and types.

7.2 Choice of the Embedding Surface

The first stage of the application of the TENSE framework to the TNO OLYMPUS Well Placement Optimisation Challenge is to embed the 2-dimensional well location parameter vector $\mathbf{x} = \binom{x}{y}$ in 3-dimensions, as in Equation (6.3.1), necessitating the specification of an embedding surface, $v(\mathbf{x}) = v(x, y)$. This is achieved by tearing along the five partial discontinuities, shown in Figure 6.2 and described in Table 7.1, and curving alternate regions higher and lower into the 3$^{\text{rd}}$ dimension using quadratic forms, exploiting a similar strategy to that employed in Section 6.3.1 for the second example function (see Equation (6.3.7) with embedding surface defined in Equations (6.3.8) and (6.3.9)). There is the option is to use information on the fault transmissibilities, obtained via expert elicitation or performing simulations, to guide the specification of the magnitude of the tearing along each discontinuity.

There are several considerations when specifying an embedding surface which are satisfied by the choice of a piecewise quadratic form for $v(x, y)$, which may be curved between the partial discontinuities and flat everywhere else. Firstly, constant shifts are irrelevant as the non-stationary covariance function is expressed in terms

of $Q(\mathbf{v}(\mathbf{x}), \mathbf{v}(\mathbf{x}'))$ (see Equation (6.3.66)) which is invariant to such transformations. Secondly, $v(x, y)$ must contain no linear terms since the derivative must exist at the joining edge (this is the left edge for the OLYMPUS reservoir) of each curved region, beyond which the embedding surface is flat. Any function which includes a linear term will possess a non-zero derivative along this edge when approached along the curved region, however, $v(x, y)$ has zero derivative when approached along the flat region. This is satisfied by the choice of a quadratic polynomial term. Moreover, this ensures that $v(x, y)$ has a locally linear derivative which is desirable for the TENSE framework, where the non-stationary covariance function employed accounts for the use of an embedding and is a good approximation up to first order. Functions beyond polynomials may be considered, although these must also satisfy the conditions on the derivative. For a general function, the Taylor expansion near to any point along the joining (left) edge of a region will therefore have a quadratic leading order term, since the constant and linear polynomial terms are zero for the above described reasons. The use of a quadratic thus serves as an adequate approximation for a general function for the purpose of embedding. In addition, for the desired stationary squared exponential covariance function induced by the corresponding non-stationary form (see Equation (6.3.67)), the covariance structure must rapidly tend towards zero as the separation between parameter vectors increases. This is achieved using a piecewise quadratic form of embedding surface.

The embedding surface is displayed in Figure 7.1, with the full definition in Equation (7.2.1).

$$v(x,y) = \begin{cases} \left(\dfrac{x-94}{118-94}\right)^2 \mathbb{1}_{\{x>94\}}, & \text{for } 0 \le y < 73.5 \\ -1.2\left(\dfrac{x-b_1(y)}{118-b_1(y)}\right)^2 \mathbb{1}_{\{x>b_1(y)\}}, & \text{for } 73.5 \le y < 85.5 \\ 3\left(\dfrac{x-b_2(y)}{118-b_2(y)}\right)^2 \mathbb{1}_{\{x>b_2(y)\}}, & \text{for } 85.5 \le y < 99.5 \\ 0, & \text{for } 99.5 \le y < 103.5 \\ -2\left(\dfrac{x-87}{118-87}\right)^2 \mathbb{1}_{\{x>87\}}, & \text{for } 103.5 \le y < 123.5 \\ 1, & \text{for } 123.5 \le y \le 181 \end{cases} \tag{7.2.1}$$

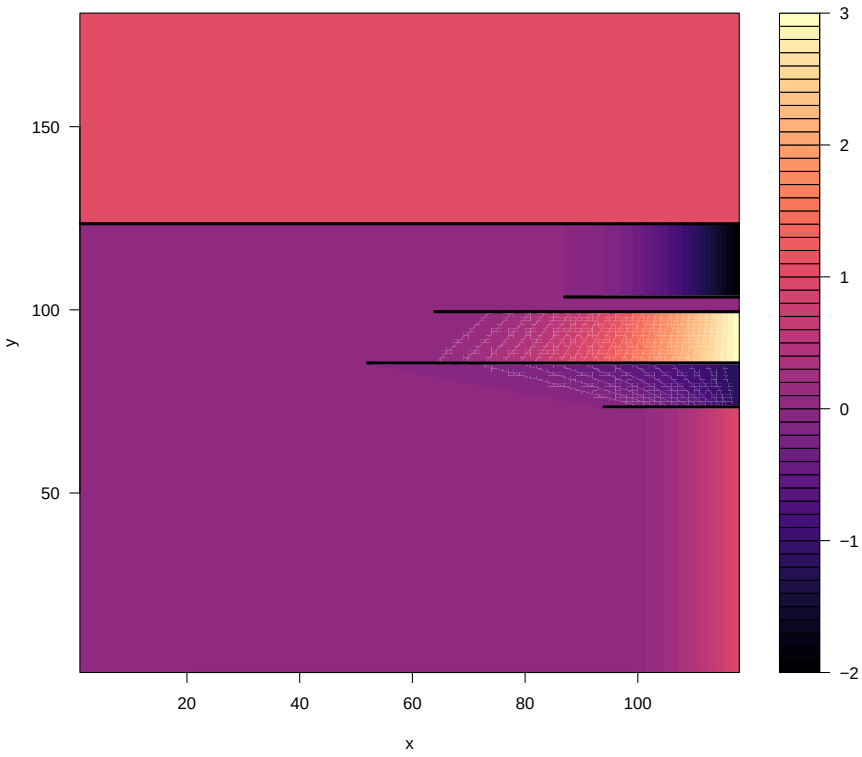
where

$$b_1(y) = 94 + \frac{y-73.5}{\left(\frac{85.5-73.5}{52-94}\right)} \quad \text{for } 73.5 \le y < 85.5$$

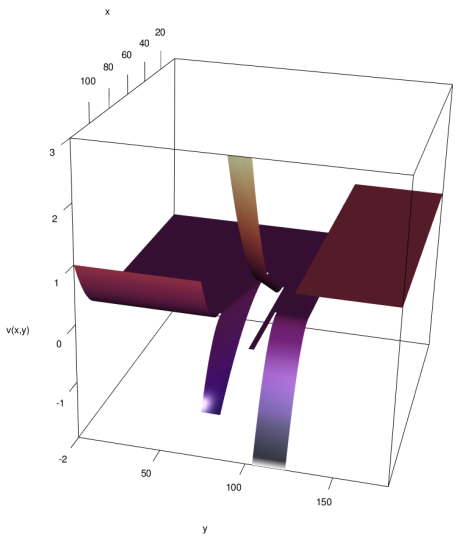
and

$$b_2(y) = 52 + \frac{y-85.5}{\left(\frac{99.5-85.5}{64-52}\right)} \quad \text{for } 85.5 \le y < 99.5$$

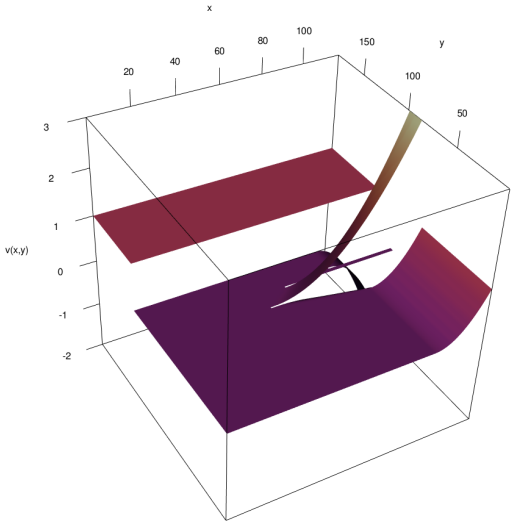
The fault locations are used to define six regions bordered by either two neighbouring faults, or a fault and the y-parameter lower or upper limit, yielding a piecewise function, as seen in Figure 7.1a. Between partial faults (lines 1 to 5), the function is of a quadratic form and curves towards a pre-specified height on the right-boundary with an alternating sign. The nature of the quadratic curvature depends on the lengths of two neighbouring faults and handled by the linear interpolation of the x-coordinate in $b_1(y)$ and $b_2(y)$. Consequently, $v(x,y)$ is of a complex form and is a twisted quadratic within these regions. To the left of the fault boundaries, $v(x,y)$ is flat with height 0. Note that the fourth region is defined to be of height 0 and hence is also flat. Above the sealing fault, $v(x,y)$ is fixed at a greater constant height of 1. This is to ensure that it is sufficiently distant from the rest of the embedding surface and thus induces the full discontinuity, as evidenced in Figure 7.1c.



(a) 2D projection of the embedding surface.



(b) Embedding surface 3D view 1.



(c) Embedding surface 3D view 2.

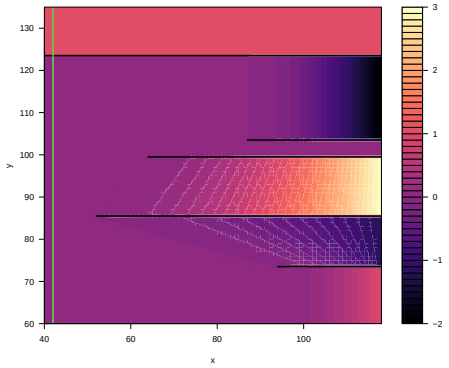
Figure 7.1: Plots of the embedding surface, $v(x, y)$, defined in Equation (7.2.1), and used in the TENSE emulator for the TNO OLYMPUS Well Placement Optimisation Challenge. The surface is torn along the location of each of the geological fault boundaries, depicted by the black lines in Figure 7.1a, where various regions of the surface are curled up or down to induce the partial discontinuities. Figure 7.1a shows a 2-dimensional projection onto the location parameter space, whilst Figures 7.1b and 7.1c provide a 3-dimensional visualisation.

7.3 Prior Covariance Structure

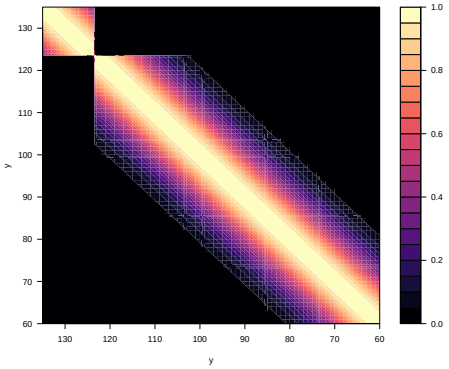
In this section we validate the proposed form of embedding surface from Section 7.2 by demonstrating that the emulator exhibits discontinuous jumps across each of the fault boundaries with minimal warping, as addressed in Sections 6.3.1 and 6.3.2, by examining the induced prior covariance structure. This is illustrated along multiple green vertical strips of points that cross (a subset of) the partial discontinuities in Figure 7.2. The left column shows a zoomed in section of the $v(x, y)$ surface projection onto 2-dimensions with the discontinuities represented by horizontal black lines (as in Figures 6.2 and 7.1a), whilst the right column provides a visualisation of the corresponding induced correlation matrix along the green strip on projection back to the original 2-dimensional parameter space. This is computed as $\text{Cov}[f(\mathbf{x}), f(\mathbf{x}')] \equiv \text{Cov}[f(\mathbf{v}(\mathbf{x})), f(\mathbf{v}(\mathbf{x}'))]$, using a non-stationary version of the squared exponential covariance function defined in Equations (6.3.66) and (6.3.67), also utilising Equation (6.3.37), which is used through the analysis of the OLYMPUS model. Scaling by a variance hyperparameter yields the covariance matrix.

First consider the induced correlation matrix along a vertical strip at $x = 42$ which is to the left of all the partial fault boundaries, as shown in Figures 7.2a and 7.2b. This crosses the sealing fault only at $y = 123.5$ dividing the parameter space into two regions with the output for $y > 123.5$ and $y < 123.5$ being uncorrelated as desired. Moreover, the squared exponential covariance function defines the correlation within each region, whilst the partial faults are yet to affect the prior correlation matrix.

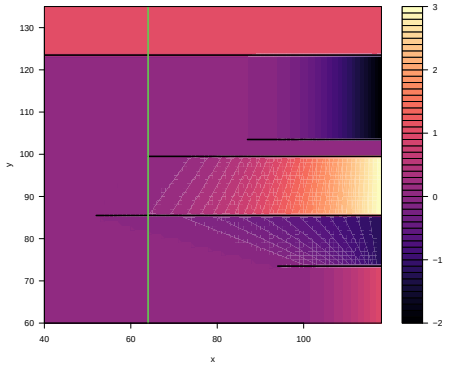
Figures 7.2c and 7.2d are defined as above for the line $x = 64$ which crosses the longest partial discontinuity (the 2^{nd} fault in Table 7.1) and is situated close to the interior endpoint. In the induced correlation matrix plot this manifests as a pinching at the fault location, $y = 85.5$, depicting a decrease in the correlation across the fault, but not a full separation of the regions, as is observed for the top sealing fault. This is enabled by the embedding framework and thus accurately portrays the diminished correlation around the fault interior endpoint achieved via the increased separation in the 3^{rd} dimension.



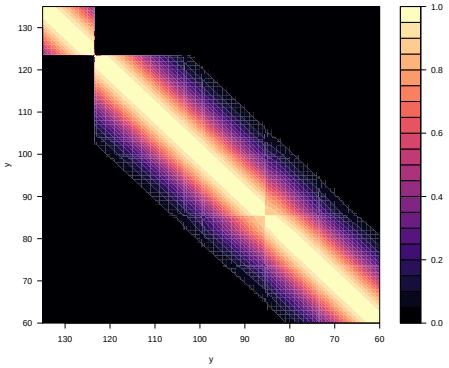
(a) $v(x, y)$ with vertical line at $x = 42$.



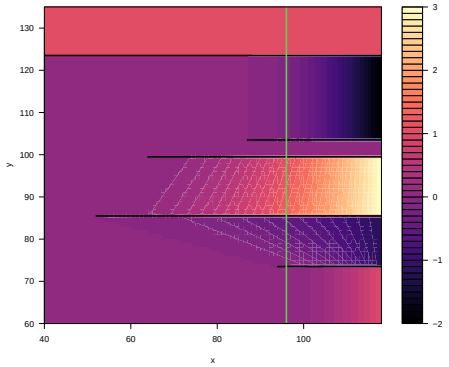
(b) Induced correlation matrix, $x = 42$.



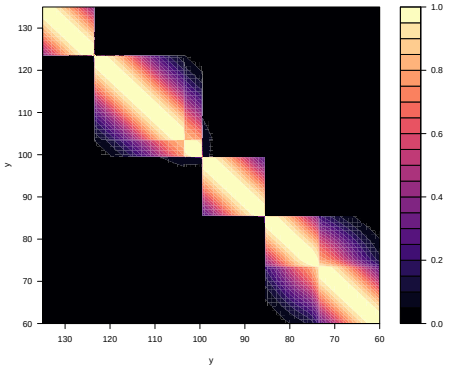
(c) $v(x, y)$ with vertical line at $x = 64$.



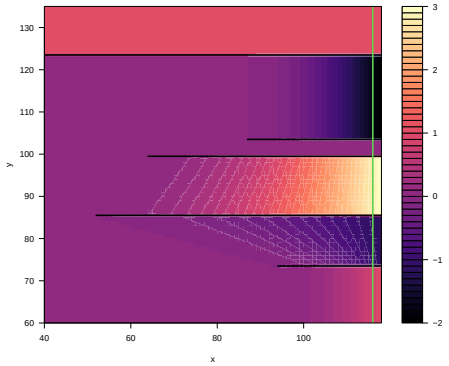
(d) Induced correlation matrix, $x = 64$.



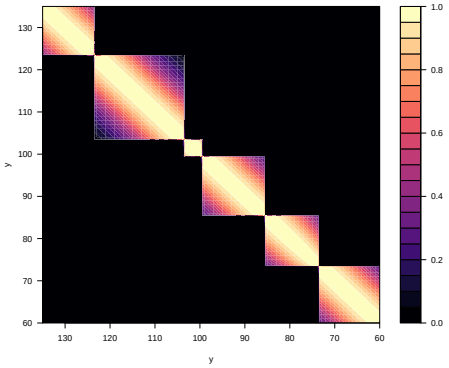
(e) $v(x, y)$ with vertical line at $x = 96$.



(f) Induced correlation matrix, $x = 96$.



(g) $v(x, y)$ with vertical line at $x = 116$.



(h) Induced correlation matrix, $x = 116$.

Figure 7.2: Prior correlation structure along vertical strips for the TNO OLYMPUS Well Placement Optimisation Challenge. The left plots show the embedding surface, $v(x, y)$, whilst the right plots show the corresponding induced emulator correlation matrix along the green line, at locations $x = 42$, 64, 96 and 116 respectively.

The plots in Figures 7.2e and 7.2f illustrates the correlation structure for $x = 96$ where the green line crosses all of the fault boundaries. Within the region bounded above and below by the two longest partial faults (the 2nd and 3rd faults in Table 7.1) there is negligible correlation with points in the neighbouring regions due to the long distance from the fault interior endpoints, as illustrated by the square sub-matrix between $y = 85.5$ and $y = 99.5$. For the two shortest partial faults (the 1st and 4th faults in Table 7.1) there exists a suppressed correlation across the fault boundaries due to the proximity to their interior endpoints, shown by a pinching, as described for the vertical strip at $x = 64$.

In Figures 7.2g and 7.2h the case of $x = 116$ is examined where the correlation matrix plot depicts six uncorrelated regions separated by the five faults. This arises due to the distance from the interior endpoints and thus portrays the desired behaviour. Moreover, the construction of the TENSE approach of embedding in higher dimensions guarantees that all of the visualised correlation matrices are valid.

7.4 Well 1 Wave 1 Analysis

We proceed to perform a wave 1 analysis for the first well location within the sequential drilling strategy using the embedding surface $v(x, y)$ defined in Section 7.2 and constructing an emulator for the expected NPV utility function, as represented by $f(\mathbf{x})$, in the presence of the discontinuities caused by the geological faults. An initial design is formulated in Section 7.4.1 with the results of emulation presented in Section 7.4.2.

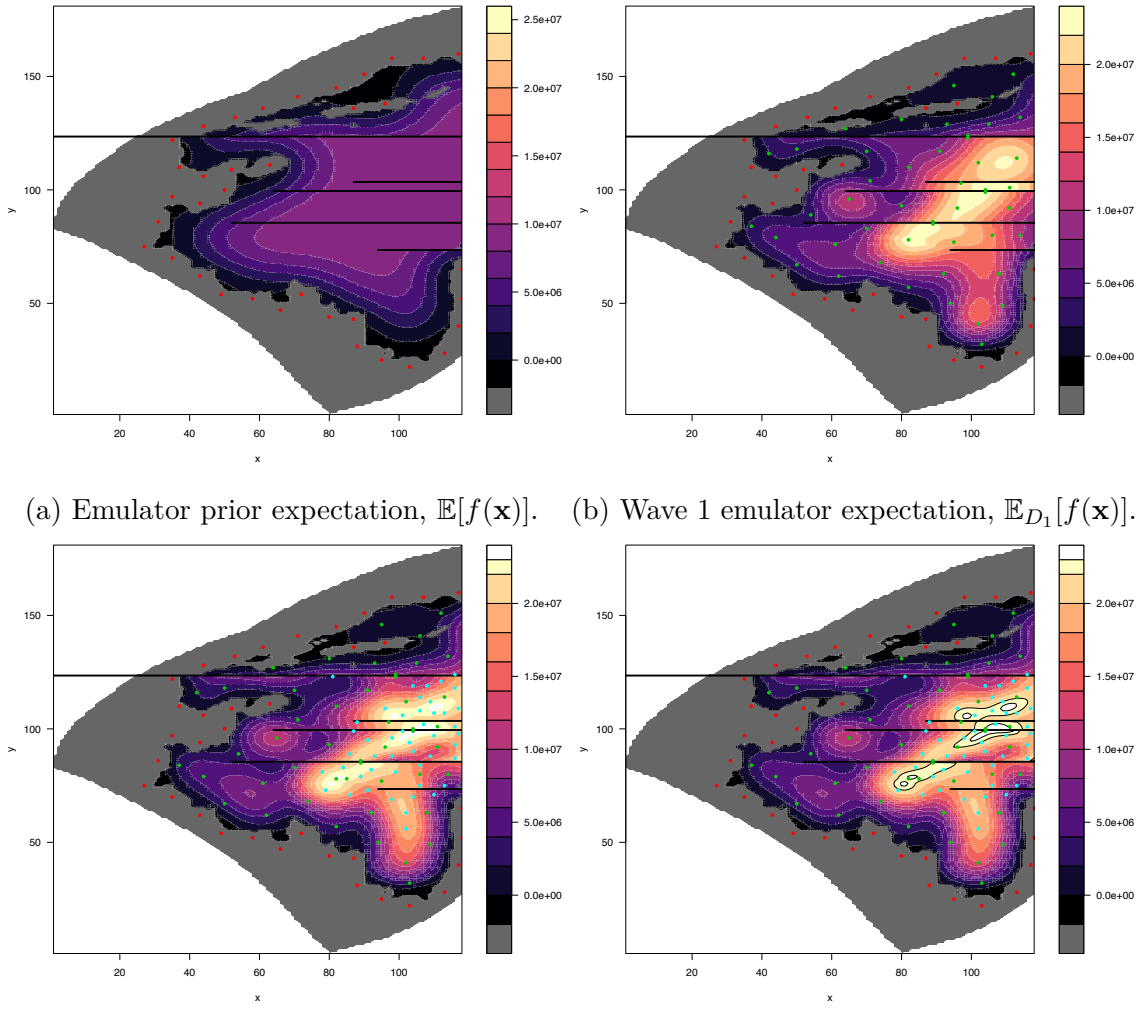
7.4.1 Well 1 Wave 1 Design for Simulations

A necessary first step is to determine the first well type prior to evaluating simulations where reservoir engineering insight suggests this should be a producer. This is to relieve the reservoir pressure and thus avoid an excessive build-up that would occur for a first injector well without any producers present, whilst also ensuring an income

at earlier times to mitigate the high discounting effects.

In Figure 6.2 it is observed that there are regions of the OLYMPUS field that contain no oil, as shown in blue, constituting additional prior information which should be exploited. It is known, without performing any model evaluations, that any producer well placed outside of the oil containing region of the reservoir will yield no oil production and hence contribute zero to the NPV. For the first well analysis this implies that the NPV will be zero. For information along straight boundaries in 2-dimensions, or more generally on hyperplanes in higher dimensions, such known model behaviour can be incorporated analytically within the emulator construction as described in [95, 173]. However, this methodology is inapplicable for the OLYMPUS model as the boundary surrounding the oil containing part of the reservoir is of a complex shape. Instead, a set of 36 manually chosen "ghost points" just outside the oil containing region are added, with the NPV of each set to $f(\mathbf{x}) = 0$. The effect of this prior information is illustrated in Figures 7.3a and 7.4a which displays the emulator prior expectation, $\mathbb{E}[f(\mathbf{x})]$, and standard deviation, $\sqrt{\text{Var}[f(\mathbf{x})]}$, respectively over $\mathcal{X}$, and where the ghost point locations are shown in red. It is evident that the ghost runs provide important information about the location of the oil containing section of the reservoir and thus regulate the expectation and uncertainty accordingly.

A wave 1 space filling design was generated respecting the following considerations. The OLYMPUS model is computationally intensive, and when planning the design of simulations, our collaborator informed us that there was uncertainty on the availability of computational resources, implying potential early termination of the design. Design strategies such as those discussed in Section 2.3, for example, Latin hypercube designs, are therefore unsuitable. Instead, a sequential design strategy was employed with each point chosen to minimise the mean TENSE emulator variance over $\mathcal{X}$. This calculation utilises the TENSE emulator correlation structure, thus it respects the discontinuities; specifically the uncorrelated regions as highlighted in Figure 7.2. Due to the sequential design construction early



(a) Emulator prior expectation, $\mathbb{E}[f(\mathbf{x})]$.

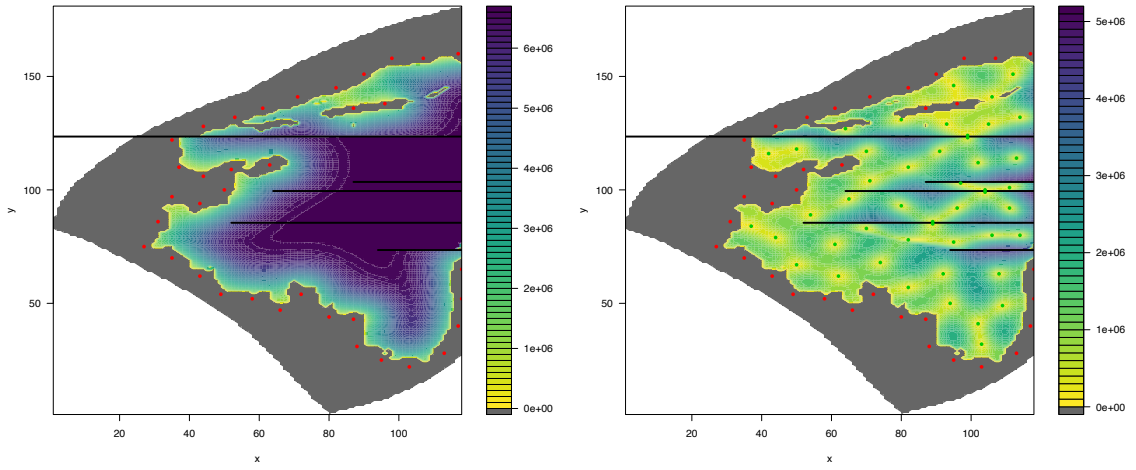
(b) Wave 1 emulator expectation, $\mathbb{E}_{D_1}[f(\mathbf{x})]$.

(c) Wave 2 emulator expectation, $\mathbb{E}_{D_1 \cup D_2}[f(\mathbf{x})]$.

(d) Wave 2 regions of high expected NPV.

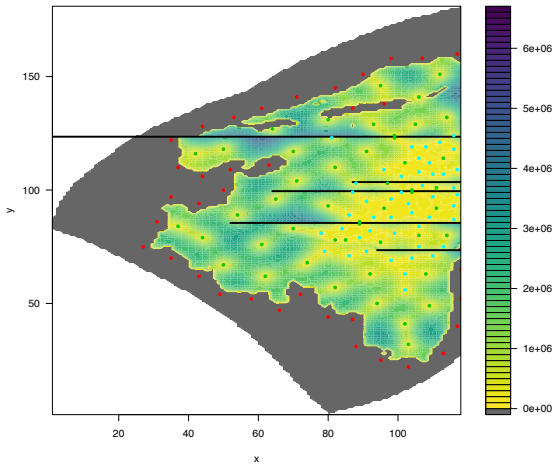
Figure 7.3: TENSE emulator adjusted expectation for the OLYMPUS expected NPV as a function of the location of well 1. Figure 7.3a shows the prior emulator expectation, $\mathbb{E}[f(\mathbf{x})]$, trained only on the ghost runs (red points) situated in the none oil producing region. Figure 7.3b shows the wave 1 TENSE emulator expectation, $\mathbb{E}_{D_1}[f(\mathbf{x})]$, which is trained on the 47 wave 1 design points (green), and the ghost runs. Figures 7.3c and 7.3d show the wave 2 TENSE emulator expectation, $\mathbb{E}_{D_1 \cup D_2}[f(\mathbf{x})]$, trained on an additional set of 48 wave 2 simulations (cyan), with the latter highlighting the regions corresponding to a high expected NPV.

termination would still yield an optimised collection of simulations. For practical application $\mathcal{X}$ is substituted for a sufficiently dense grid of points $\mathbf{S} \subset \mathcal{X}$ to aid computational tractability. Each design point, $\mathbf{x}^{(i)}$, $i = 1, \ldots, n$, is selected using



(a) Emulator prior standard deviation, $\sqrt{\text{Var}[f(\mathbf{x})]}$.

(b) Wave 1 Emulator standard deviation, $\sqrt{\text{Var}_{D_1}[f(\mathbf{x})]}$.



(c) Wave 2 emulator standard deviation, $\sqrt{\text{Var}_{D_1 \cup D_2}[f(\mathbf{x})]}$.

Figure 7.4: TENSE emulator adjusted standard deviation for the OLYMPUS expected NPV as a function of the location of well 1. Figure 7.4a shows the prior emulator standard deviation, $\sqrt{\text{Var}[f(\mathbf{x})]}$, trained only on the ghost runs (red points). Figure 7.4b shows the wave 1 TENSE emulator standard deviation, $\sqrt{\text{Var}_{D_1}[f(\mathbf{x})]}$, which is trained on the 47 wave 1 design points (green), and the ghost runs, whilst Figure 7.4c shows the wave 2 TENSE emulator standard deviation, $\sqrt{\text{Var}_{D_1 \cup D_2}[f(\mathbf{x})]}$, trained on an additional set of 48 wave 2 simulations (cyan).

the criterion in Equation (7.4.1), where the adjusted variance is computed with respect to $D_1^{(i)} = \{f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(i-1)}), f(\mathbf{x})\}$, for $i > 1$, with $D_1^{(1)} = \{f(\mathbf{x})\}$. The notation $\text{mean}_A(g(\cdot))$ represents the arithmetic sample mean of the function $g(\cdot)$

over the set A.

$$\mathbf{x}^{(i)} \coloneqq \arg\min_{\mathbf{x}\in\mathbf{S}} \left\{ \operatorname*{mean}_{\mathbf{x}'\in\mathbf{S}} \left(\mathrm{Var}_{D_1^{(i)}}[f(\mathbf{v}(\mathbf{x}'))] \right) \right\} \tag{7.4.1}$$

Note that the emulator adjusted variance calculations using Equation (2.2.4) do not require the simulation evaluations; hence the full design can be constructed prior to launching any simulations.

The above sequential design strategy was applied to the first well within the TNO OLYMPUS Well Placement Optimisation Challenge to construct a 41 point design. An approximation to the full mean calculation uses $\mathbf{S} = \{(x, y) \mid x \in [30, 118] \cap \mathbb{Z},\ y \in [25, 158] \cap \mathbb{Z}\} \subset \mathcal{X}$, the smallest rectangle of integer grid points which encompass the oil containing region. For the TENSE emulator, a squared exponential covariance function over the 2-dimensional location parameter space was assumed, with a single fixed correlation length hyperparameter, $\theta = 12$. This choice is informed by the local correlation seen in the vertical oil content of the geological realisations in Figure 6.2, whilst also noting that both coordinate parameters are on the same scale. Moreover, it is unnecessary to stipulate a variance hyperparameter at this stage as it factors out of the design calculation. In addition, a nearest neighbour approximation to the emulator variance calculation was employed to improve computational efficiency. This was especially relevant when selecting later design points. An extra three pairs of points on either side of three of the major faults were manually added to yield direct information regarding the discontinuities in these regions. The resulting 47 point wave 1 design, denoted $\mathcal{D}_1$, is shown in Figure 7.3b as the green points, and given in Table 7.3. It displays good space filling properties, whilst adequately exploring each of the uncorrelated regions between the faults. At each design point, $\mathbf{x}^{(i)},\ i = 1, \ldots, 47$, all 50 of the geological realisations were evaluated with vertical well trajectories, with the ensemble mean calculated to obtain $f(\mathbf{x}^{(i)})$ and $D_1' = \{f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(47)})\}$. This is augmented to include the 36 ghost runs with $D_1 = \{D_1', 0 \ldots, 0\} \in \mathbb{R}^{(47+36)}$ for use in the emulator equations. Note that the full TNO OLYMPUS Well Placement Optimisation Challenge also

Simulation Index	x	y		Simulation Index	x	y
Run 1	61	76		Run 25	50	67
Run 2	92	63		Run 26	91	117
Run 3	82	110		Run 27	64	127
Run 4	106	141		Run 28	109	49
Run 5	102	41		Run 29	95	77
Run 6	65	96		Run 30	94	50
Run 7	44	79		Run 31	103	32
Run 8	102	112		Run 32	74	68
Run 9	82	78		Run 33	97	103
Run 10	80	131		Run 34	93	129
Run 11	50	118		Run 35	114	132
Run 12	108	63		Run 36	70	83
Run 13	96	92		Run 37	71	104
Run 14	62	62		Run 38	42	116
Run 15	106	80		Run 39	114	80
Run 16	111	92		Run 40	111	101
Run 17	105	129		Run 41	37	84
Run 18	82	57		Run 42	89	85
Run 19	113	114		Run 43	89	86
Run 20	80	93		Run 44	104	99
Run 21	70	117		Run 45	104	100
Run 22	95	146		Run 46	99	123
Run 23	112	151		Run 47	99	124
Run 24	54	89		Run 48	85	78

Table 7.3: Locations of the 47 design points for the TNO OLYMPUS Well Placement Optimisation Challenge well 1, wave 1 analysis. The additional run 48 was performed after the wave 1 analysis for the purpose of investigating well trajectories with the vertical well trajectory used in formulating the wave 2 design.

includes well trajectory; an extra problem that is not considered in this thesis.

7.4.2 Well 1 Wave 1 Emulation Results

The TENSE framework presented in Section 6.3 is applied to the expected NPV utility function employing the embedding surface $v(x, y)$ (see Section 7.2) using a constant mean function with prior $\boldsymbol{\mu}_\beta = \text{mean}_{\mathbf{x}\in\mathcal{D}_1}\left(f(\mathbf{x})\right) = 9.753 \times 10^6$. A squared exponential covariance function is assumed, as described in Equations (6.3.66) and (6.3.67) with the embedded input variance matrix defined in Equation (6.3.37). The hyperparameters are estimated via maximum likelihood under a Gaussian as-

sumption for D_1' resulting in $\sigma = 6.080 \times 10^6$ and $\theta = 12$. A nugget term is included with $\rho = 0.00001$.

A Bayes linear update is performed for the embedded emulator via Equations (2.2.3) to (2.2.5) given D_1 discussed in Section 7.4.1, yielding the adjusted expectation $\mathbb{E}_{D_1}[f(\mathbf{x})]$ shown in Figure 7.3b by the coloured contours. It is observed how the emulator incorporates jumps in $f(\mathbf{x})$ due to the discontinuities induced by the faults, whist remaining smooth in all other parts of $\mathcal{X}$, thus exhibiting the desired behaviour. The OLYMPUS map coloured by the emulator adjusted standard deviation shown in Figure 7.4b demonstrates similar desirable characteristics. Neither plot exhibits any warping effect that was previously observed for the naïve embedding approach, thus demonstrating the success of the correction in Sections 6.3.2 and 6.3.3. Implementation of the TENSE framework has successfully achieved the first aim of this analysis to provide a clear visualisation of the expected NPV surface across the oil reservoir along with a quantification of the uncertainty, highlighting regions of suspected high NPV for further investigation.

The second aim is to identify and examine regions corresponding to high expected NPV for consideration by the relevant expert(s) or decision maker(s). This is achieved following the iterative decision support procedure presented in Section 3.6 to identify candidate regions of the parameter space instead of a single maximum [135]. A modified form of the decision support implausibility measure defined in Equation (7.4.2) is employed, where $\mathbb{E}_D[f(\mathbf{x})]$ and $\mathrm{Var}_D[f(\mathbf{x})]$ are the (TENSE) emulator adjusted expectation and variance, $f^+ = \max_{\mathbf{x}\in\mathcal{D}} f(\mathbf{x})$, the current highest simulated expected NPV, σ^2_{MD} is structural model discrepancy variance (for further details, see Section 2.7.4), and δ is a tolerance parameter enabling the user to specify the exploration distance below the maximum. The tolerance parameter represents the main difference versus the decision support implausibility measure in Equation (3.6.3).

$$I'_{\mathrm{DS}}(\mathbf{x}) = \frac{|(f^+ - \delta) - \mathbb{E}_D[f(\mathbf{x})]|}{\sqrt{\mathrm{Var}_D[f(\mathbf{x})] + \sigma^2_{\mathrm{MD}}}} \cdot \mathbb{1}_{\{\mathbb{E}_D[f(\mathbf{x})] < f^+ - \delta\}} \tag{7.4.2}$$

As in Section 3.6.3, the modified implausibility measure, $I'_{\text{DS}}(\mathbf{x})$, is used to exclude regions of $\mathcal{X}$ that are highly unlikely to yield values of $f(\mathbf{x})$ greater than $f^+ - \delta$, by imposing the following constraint:

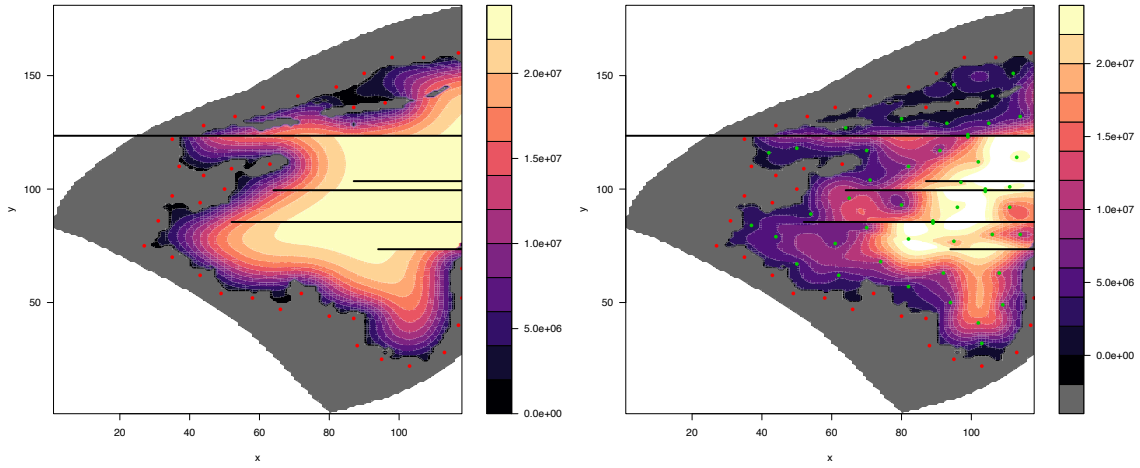
$$I'_{\text{DS}}(\mathbf{x}) \leq c \tag{7.4.3}$$

The threshold is typically chosen to be $c = 3$ based on Pukelsheim's 3-sigma rule for arbitrary continuous unimodal distributions [142]. A new non-implausible region, $\mathcal{X}_1$, of well locations that will potentially yield a high expected NPV is defined in Equation (7.4.4) for further exploration.

$$\mathcal{X}_1 \equiv \{\mathbf{x} \in \mathcal{X} \mid \mathbb{E}_D[f(\mathbf{x})] + c\sqrt{\text{Var}_D[f(\mathbf{x})] + \sigma^2_{\text{MD}}} \geq f^+ - \delta\} \tag{7.4.4}$$

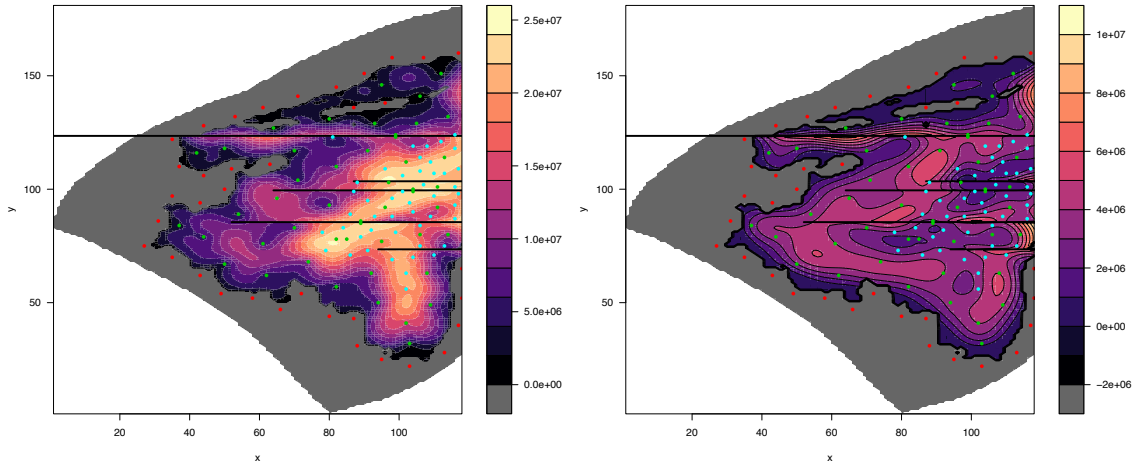
Figures 7.5a and 7.5b show the credible interval upper bound: $\mathbb{E}[f(\mathbf{x})] + c\sqrt{\text{Var}[f(\mathbf{x})]}$ for the prior; and $\mathbb{E}_{D_1}[f(\mathbf{x})] + c\sqrt{\text{Var}_{D_1}[f(\mathbf{x})]}$ after wave 1 where the emulator has been adjusted by D_1, respectively, for $c = 2$. The latter is used in the first wave of iterative decision support compared to $f^+ - \delta$ with the brightest yellow contours identified as well 1 locations most likely to yield a high expected NPV. The wave 1 non-implausible region, $\mathcal{X}_1$, is defined using the standard choice of $c = 3$, along with $f^+ - \delta \equiv 0.95 \cdot f^+ = 2.0 \times 10^7$, representing a 5% reduction from the maximum simulated expected NPV. This constitutes a volume of 24.14% of the oil containing region, or equivalently, 7.97% of the full OLYMPUS location parameter space, and defines the search region for wave 2. Note that a 48^{th} simulation was performed at $(x, y) = (85, 78)$ for the purpose of investigating well trajectory (not addressed in this thesis). The wave 1 design is augmented to include this 48^{th} simulation with vertical well trajectory for the wave 2 analysis.

During the wave 1 design phase in Section 7.4.1, a collection of 36 ghost runs were introduced to pin down the boundary of the oil containing region of the OLYMPUS reservoir. These are of vital importance for updating the emulator, as seen in Figure 7.6 where only the 47 point wave 1 design is used with simulation output in D'_1. Comparing Figures 7.3b and 7.6a, it is evident that omission of the ghost



(a) Prior upper CI, $\mathbb{E}[f(\mathbf{x})] + 2\sqrt{\text{Var}[f(\mathbf{x})]}$.

(b) Wave 1 upper CI, $\mathbb{E}_{D_1}[f(\mathbf{x})] + 2\sqrt{\text{Var}_{D_1}[f(\mathbf{x})]}$.

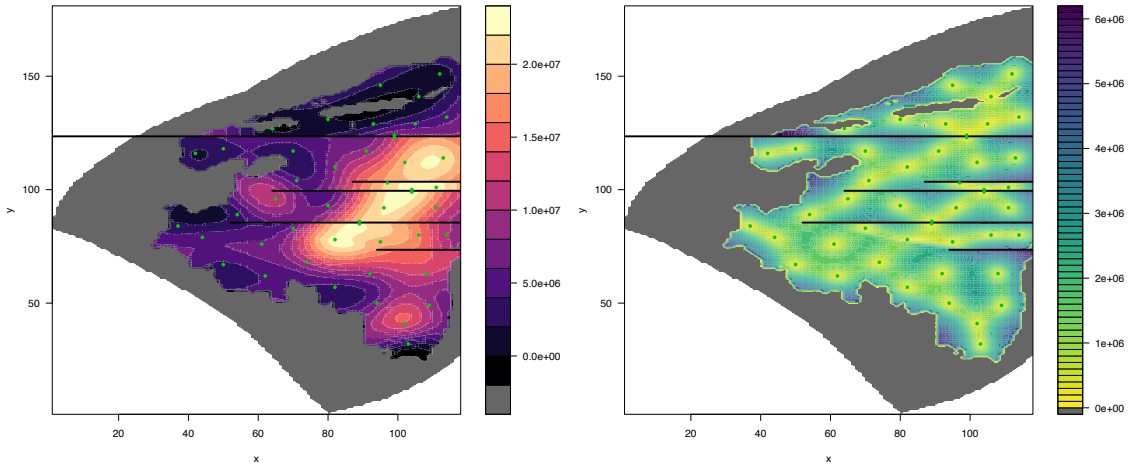


(c) Wave 2 upper CI, $\mathbb{E}_{D_1 \cup D_2}[f(\mathbf{x})] + 2\sqrt{\text{Var}_{D_1 \cup D_2}[f(\mathbf{x})]}$.

(d) Emulator expectation for the NPV standard deviation over the OLYMPUS ensemble.

Figure 7.5: TENSE emulator credible interval (CI) upper bound for the OLYMPUS expected NPV as a function of the location of well 1. Figure 7.5a shows the prior emulator upper CI, $\mathbb{E}[f(\mathbf{x})] + 2\sqrt{\text{Var}[f(\mathbf{x})]}$, trained only on the ghost runs (red points). Figure 7.5b shows the wave 1 TENSE emulator upper CI, $\mathbb{E}_{D_1}[f(\mathbf{x})] + 2\sqrt{\text{Var}_{D_1}[f(\mathbf{x})]}$, which is trained on the 47 wave 1 design points (green), and the ghost runs. Figure 7.5c show the wave 2 TENSE emulator upper CI, $\mathbb{E}_{D_1 \cup D_2}[f(\mathbf{x})] + 2\sqrt{\text{Var}_{D_1 \cup D_2}[f(\mathbf{x})]}$, trained on an additional set of 48 wave 2 simulations (cyan). The TENSE emulator expectation applied to the NPV standard deviation over the 50 OLYMPUS ensemble geological realisations, depicted in Figure 7.5d. This displays lower variation in most of the regions of interest highlighted by the contours in Figure 7.3d.

runs yields an excessively high emulator adjusted expectation close to the edge of the oil containing region where the mean oil concentration is much lower, as seen in Figure 6.2. This is reflected by the larger emulator adjusted standard deviation in Figure 7.6b near the edge, in contrast with Figure 7.4b where the uncertainty tends towards zero on the boundary.



(a) Wave 1 emulator expectation, $\mathbb{E}_{D_1'}[f(\mathbf{x})]$. (b) Wave 1 emulator standard deviation, $\sqrt{\mathrm{Var}_{D_1'}[f(\mathbf{x})]}$.

Figure 7.6: TENSE emulator adjusted expectation (left) and standard deviation (right) trained without the ghost runs for the OLYMPUS expected NPV as a function of the location of well 1 and corresponds to an emulator update with respect to the simulated expected NPV in D_1' only. These should be compared to Figures 7.3b and 7.4b respectively. Note that the ghost runs are situated in none oil producing regions and so provide zero contribution to the NPV, hence they can be evaluated at no extra computational cost.

7.5 Well 1 Wave 2 Analysis

Following the iterative decision support procedure, a second wave analysis is performed for the location of the first well in the TNO OLYMPUS Well Placement Optimisation Challenge in this section. Firstly, a wave 2 design over the wave 1 non-implausible region is formulated in Section 7.5.1. This is used in conjunction with the wave 1 simulations to update a TENSE emulator for the expected NPV

utility function in Section 7.5.2, whilst risk preference considerations are addressed via emulation of the quantiles from the OLYMPUS ensemble in Section 7.5.3.

7.5.1 Well 1 Wave 2 Design for Simulations

In this section a second wave of space-filling simulations is designed over the $\mathcal{X}_1$ region identified in Section 7.4.2. There remains uncertainty regarding the available computing resources implying potential early termination of the design, as was the case in wave 1. A sequential design procedure was thus employed to minimise the mean TENSE emulator variance over $\mathcal{X}_1$ through an adaptation of the criterion in Equation (7.4.1) to amalgamate any previously selected wave 2 design points with the entire wave 1 design with the dense sample of points $\mathbf{S} \subset \mathcal{X}_1$. In addition, a split strategy is exercised for the specification of c in defining $\mathcal{X}_1$ via Equation (7.4.4), before augmenting to include four points of interest.

First, a 36 point sequential design is formulated over $\mathcal{X}_{1,A} \equiv \mathcal{X}_1$ defined by $c = 3$ and $f^+ - \delta \equiv 0.95 \cdot f^+ = 2.0 \times 10^7$, and corresponds to the non-implausible region discussed in Section 7.4.2. This is with the aim of further shrinking the non-implausible region as $\mathcal{X}_1$ contains both locations which are believed to yield a high expected NPV, as well as parameter settings for which the uncertainty is sufficiently high that it is not possible to conclude either way after wave 1. These are denoted in Table 7.4 by the first 36 runs.

A further 7 point sequential design is generated, given the combined wave 1 and the above 36 point designs, over the region $\mathcal{X}_{1,B} \equiv \{\mathbf{x} \mid \mathbb{E}_{D_1}[f(\mathbf{x})] \geq f^+ - \delta = 2.0 \times 10^7\}$, corresponding to $c = 0$. Within $\mathcal{X}_{1,B}$ the TENSE emulator adjusted expectation exceeds the current highest expected NPV, minus a tolerance, with the purpose being to better assess the maximum possible NPV and the corresponding well locations. Note that $\mathcal{X}_{1,B} \subset \mathcal{X}_{1,A}$, therefore designs are formulated over $\mathcal{X}_{1,A}$ and then $\mathcal{X}_{1,B}$. For iterative decision support a greater emphasis is placed on contracting $\mathcal{X}_1$, hence a larger sample is drawn from $\mathcal{X}_{1,A}$. A 7 point design over $\mathcal{X}_{1,B}$ represents a sufficient number of additional runs given the five fault boundaries which divide

the location space into 6 regions, as in $v(x, y)$, with the aim of placing at least one point within each region. Without the discontinuities, a smaller design of 2 or 3 points would suffice. These correspond to runs 37 to 43 in Table 7.4.

After sampling from $\mathcal{X}_{1,A}$ and $\mathcal{X}_{1,B}$, the design is augmented with an extra four points. Firstly, runs 44 and 45 in Table 7.4 are selected as $\mathbf{x}$ which maximise $\mathbb{E}_{D_1}[f(\mathbf{x})]$ within the two main targeted regions of the OLYMPUS map after wave 1 situated between the 2^{nd} & 3^{rd}, and 4^{th} & 5^{th} faults respectively. In addition, runs 46 and 47 are added for the purpose of obtaining more detailed information on the right and left boundary of $\mathcal{X}_1$ between the 2^{nd} and 3^{rd} faults. Note that run 46 is also situated near to the x-coordinate upper limit. It is necessary to manually add these final two points as the design criterion initially focuses on parameter settings located in central regions of the space as these have a greater impact on minimising the mean emulator adjusted variance.

These are coalesced to form the full 47 point wave 2 design in Table 7.4 which is depicted by the cyan points in Figure 7.3c. Evaluation of this design using the OLYMPUS model yields a second output vector, D_2, with the TENSE emulator adjustment now performed with respect to $D_1 \cup D_2$.

7.5.2 Well 1 Wave 2 Emulation Results

The expected NPV utility function is emulated at wave 2 using the TENSE framework employing $v(x, y)$, as in Section 7.2, and a constant mean function with prior $\boldsymbol{\mu}_\beta = \text{mean}_{\mathbf{x}\in\mathcal{D}_2}(f(\mathbf{x})) = 1.795 \times 10^7$; note that this is much greater than in wave 1 where $\boldsymbol{\mu}_\beta = 9.753 \times 10^6$. A squared exponential covariance function is assumed where the hyperparameters are estimated via maximum likelihood estimation under a Gaussian assumption for D_2 resulting in $\sigma = 6.197 \times 10^6$ and $\theta = 12$. A nugget term is included with $\rho = 0.0001$.

The Bayes linear emulator adjusted expectation, $\mathbb{E}_{D_1 \cup D_2}[f(\mathbf{x})]$, for the expected NPV if a producer well was located at $\mathbf{x}$ is shown in Figure 7.3c by the coloured contours, providing a detailed representation of the areas of high expected NPV in

Simulation Index	x	y		Simulation Index	x	y
Run 1	104	70		Run 25	98	88
Run 2	113	107		Run 26	102	63
Run 3	116	93		Run 27	104	119
Run 4	108	119		Run 28	112	121
Run 5	87	99		Run 29	91	73
Run 6	104	90		Run 30	116	87
Run 7	116	118		Run 31	113	75
Run 8	102	56		Run 32	116	101
Run 9	86	71		Run 33	101	106
Run 10	106	108		Run 34	109	88
Run 11	88	103		Run 35	109	112
Run 12	99	81		Run 36	106	84
Run 13	110	71		Run 37	84	83
Run 14	98	69		Run 38	92	82
Run 15	117	109		Run 39	93	88
Run 16	96	109		Run 40	103	94
Run 17	91	95		Run 41	110	107
Run 18	78	81		Run 42	89	79
Run 19	106	75		Run 43	106	114
Run 20	110	97		Run 44	101	99
Run 21	79	73		Run 45	107	102
Run 22	81	123		Run 46	117	98
Run 23	98	96		Run 47	86	89
Run 24	116	124				

Table 7.4: Locations of the 47 wave 2 design points used in the TNO OLYMPUS Well Placement Optimisation Challenge well 1, wave 2 analysis. The first 36 points are in $\mathcal{X}_{1,A}$, with points 37 to 43 in $\mathcal{X}_{1,B}$. This is augmented to include runs 44 and 45 which maximise $\mathbb{E}_{D_1}[f(\mathbf{x})]$ within the two main targeted regions after wave 1 between the 2^{nd} & 3^{rd}, and 4^{th} & 5^{th} faults respectively. In addition, runs 46 and 47 are added for the purpose of obtaining more detailed information on the right and left boundary of $\mathcal{X}_1$ between the 2^{nd} and 3^{rd} faults. Note that run 46 is also situated near to the x-coordinate upper limit. Fault indices are given in Table 7.1.

$\mathcal{X}$, whilst naturally incorporating the fault discontinuities. Moreover, Figure 7.4c depicts $\sqrt{\text{Var}_{D_1 \cup D_2}[f(\mathbf{x})]}$ which emphasises that the emulator uncertainty is low throughout $\mathcal{X}_1$. Examination of the credible interval upper bound after the wave 2 runs, $\mathbb{E}_{D_1 \cup D_2}[f(\mathbf{x})] + 2\sqrt{\text{Var}_{D_1 \cup D_2}[f(\mathbf{x})]}$, in Figure 7.5c shows that there is little additional information to be obtained about this region by performing further sim-

ulations. The final region of candidate well locations is highlighted in Figure 7.3d where the solid contours corresponding to thresholds of $f^+ - \delta = 2.2 \times 10^7$ and 2.3×10^7 respectively, where $f^+ = \max_{\mathbf{d} \in \mathcal{D}_1 \cup \mathcal{D}_2} f(\mathbf{x})$.

Information gained from the emulator, including the quantile emulators discussed in Section 7.5.3 and illustrated in Figure 7.8 (page 385), is combined with reservoir engineering insight to select the optimal location for the first producer well. This acknowledges a tendency in the petroleum industry for risk averse decisions. Four candidate locations are proposed based on the emulator and explored using the tNavigator software [38] to highlight the advantages and disadvantages associated with each location. It was determined that three of the locations were unsuitable due to porosity, as well as fault location and transmissibility considerations. For example, high porosity may result in nearby water being drawn by the producer well at early times rendering them uneconomical. High fault transmissibility may permit fluids to pass through the fault with the potential for water on the opposite side to then be extracted by the producer well. In contrast, intransmissible faults can limit the potential oil reserves being accessed by a producer well. To confirm this insight, an additional simulation over the OLYMPUS ensemble was performed at each proposed producer well location with the results being comparable to the current f^+. The selected location is depicted by the yellow square in Figure 7.9 (page 387) with $(x, y) = (111, 109)$.

Full optimisation of the OLYMPUS model expected NPV with respect to multiple well configurations is not the focus of this work, although the following observations are made. As discussed in Section 3.1.2 and in [135], the notion of finding a single optimum decision (in this case well placement) is somewhat misleading due to the model imperfection. Instead, when providing decision support it is more informative to provide classes of good decisions, such as shown in Figure 7.5c, for further consideration by the decision maker who may, as is common in the oil industry, have a set of additional preferences unknown to the reservoir analyst. Examples of these may include unknown risk preferences, political, financial or environmental considerations,

or other corporate logistical issues. This is demonstrated above by working with our collaborator. Furthermore, anticipation of these issues by the analyst can partially inform the choice of the δ tolerance parameter used in the definition of $\mathcal{X}_1$.

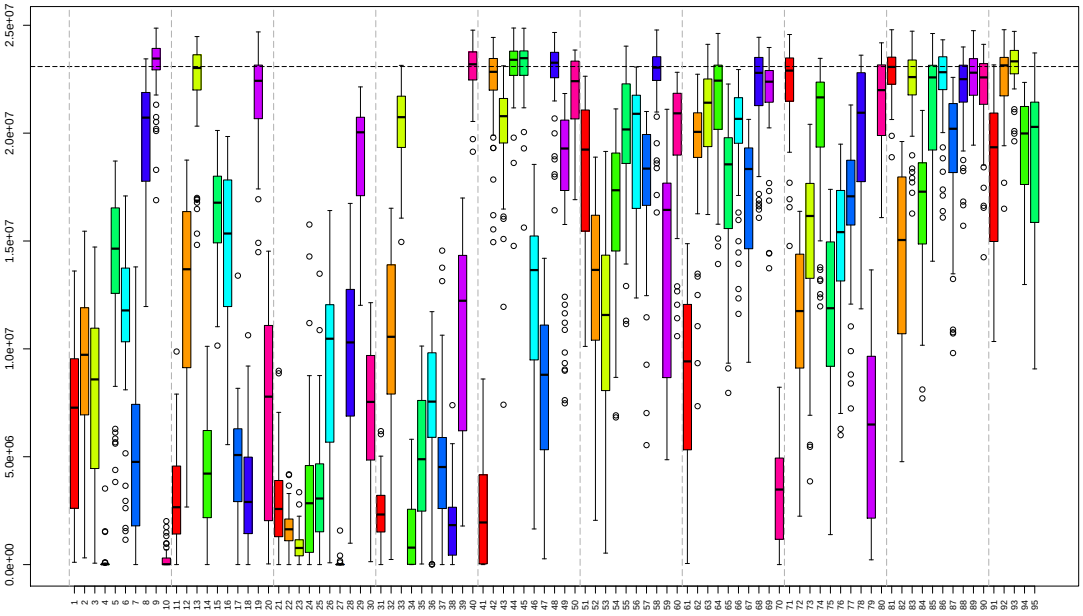
7.5.3 Well 1 Wave 2 Quantile Emulation Results

Risk preferences form an important consideration in any decision analysis and are addressed for the well 1 location. Figure 7.7 displays boxplots of the NPV over the 50 OLYMPUS geological realisations for the 95 simulations performed for the analysis. Ordering by the ensemble mean NPV in Figure 7.7b, it is evident that simulations with higher ensemble mean NPV exhibit a smaller spread over the 50 realisations compared to those with intermediate ensemble mean NPVs. This is further emphasised in Figure 7.5d which depicts the TENSE emulator adjusted expectation for the standard deviation of the NPV over the OLYMPUS ensemble, instead of the ensemble mean NPV, where the smallest uncertainty occurs for the regions of interest in Figure 7.3d.

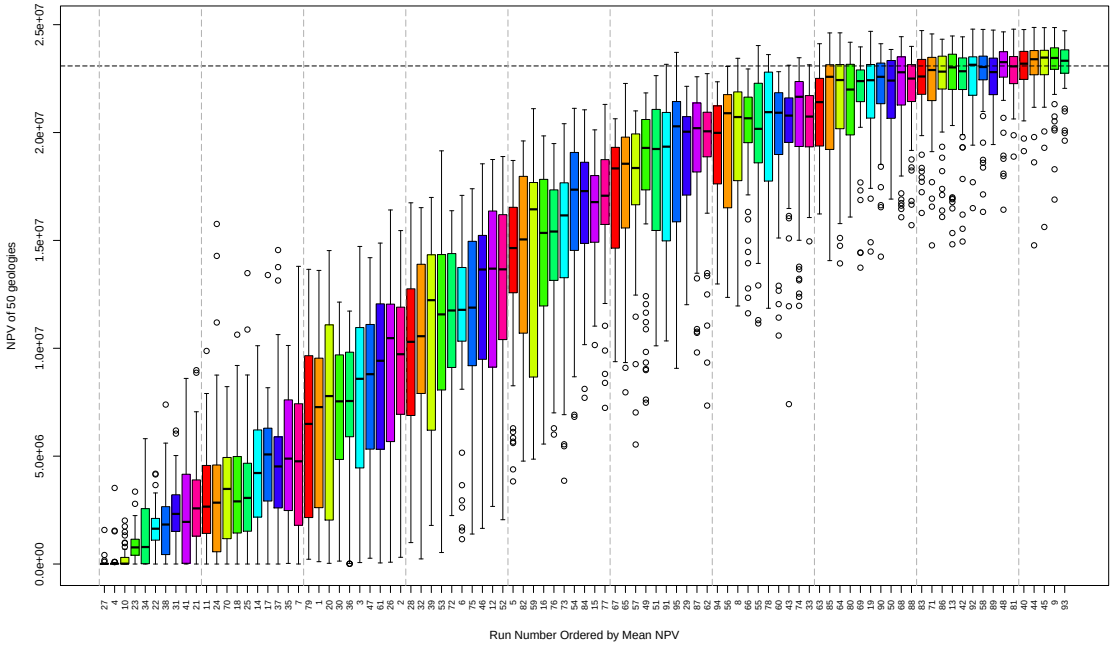
The TENSE framework is also applied to quantile emulation for the NPV over the OLYMPUS ensemble in order to investigate differing risk levels. Results are shown for the 10%, 25%, 50%, 75% and 90% quantiles in Figure 7.8 where updates are performed with respect to the wave 1 and 2 simulated quantiles. In all instances, the region identified in Section 7.5.2 also yields the highest emulator expectation for the respective NPV quantile, thus validating the conclusions for the well 1 location, with the choice being perceived to be robust to geological uncertainty, assuming that the OLYMPUS ensemble is representative.

7.6 Well 2 Analysis

The next stage of the analysis is to select the second well location, conditional on the first producer well location. In Section 7.6.1 discussion of the well type is presented before constructing a pair of well 2 designs. These are used in Section 7.6.2 within

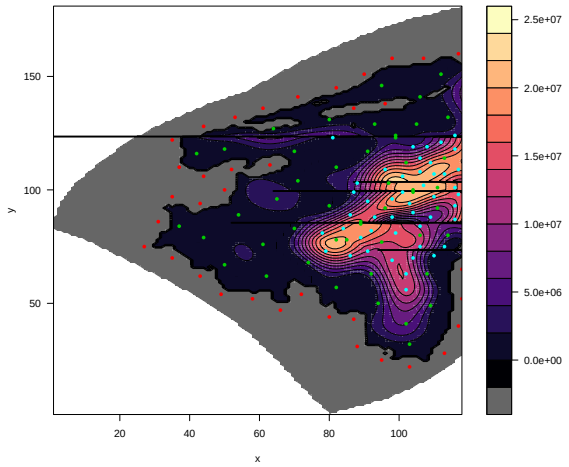


(a) Boxplot of the NPV in order of simulation evaluations.

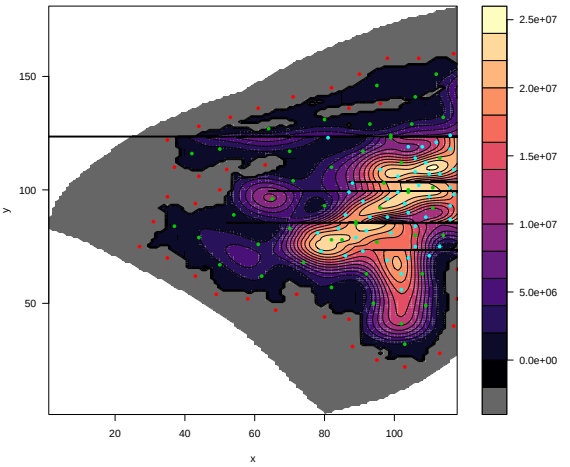


(b) Boxplot of the NPV ordered by increasing ensemble mean NPV.

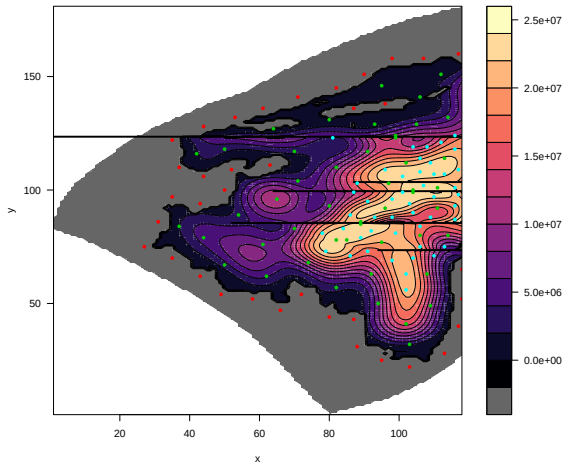
Figure 7.7: Boxplots of the NPV over the 50 OLYMPUS geological realisations for the 95 well 1, waves 1 and 2 simulations. Note that simulations with higher ensemble mean NPV exhibit smaller spread over the 50 realisations compared to simulations with intermediate ensemble mean NPV. This is in accordance with Figure 7.5d.



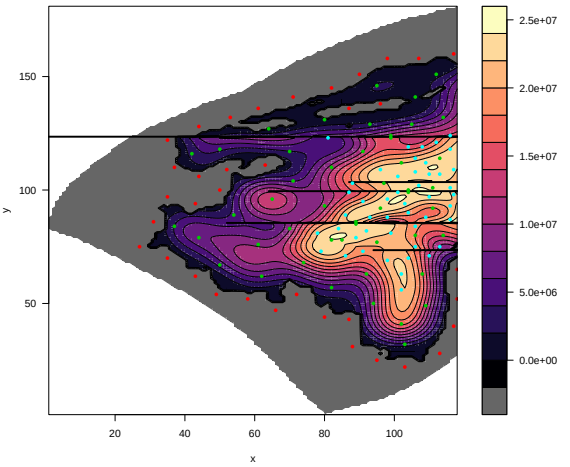
(a) TENSE expectation for 10% NPV quantile.



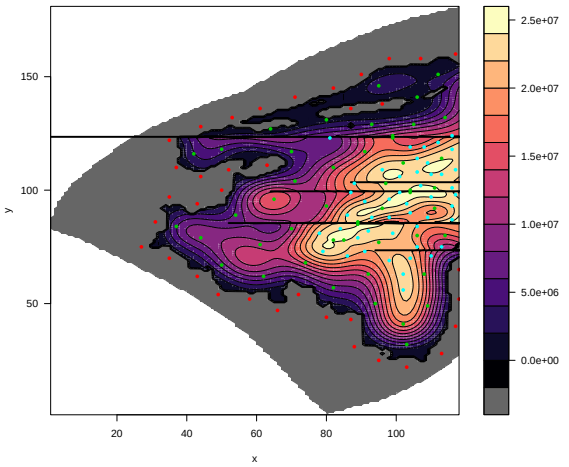
(b) TENSE expectation for 25% NPV quantile.



(c) TENSE expectation for 50% NPV quantile.



(d) TENSE expectation for 75% NPV quantile.



(e) TENSE expectation for 90% NPV quantile.

Figure 7.8: Quantile emulation using the TENSE framework applied to the NPV over the 50 OLYMPUS geological realisations for well 1 location.

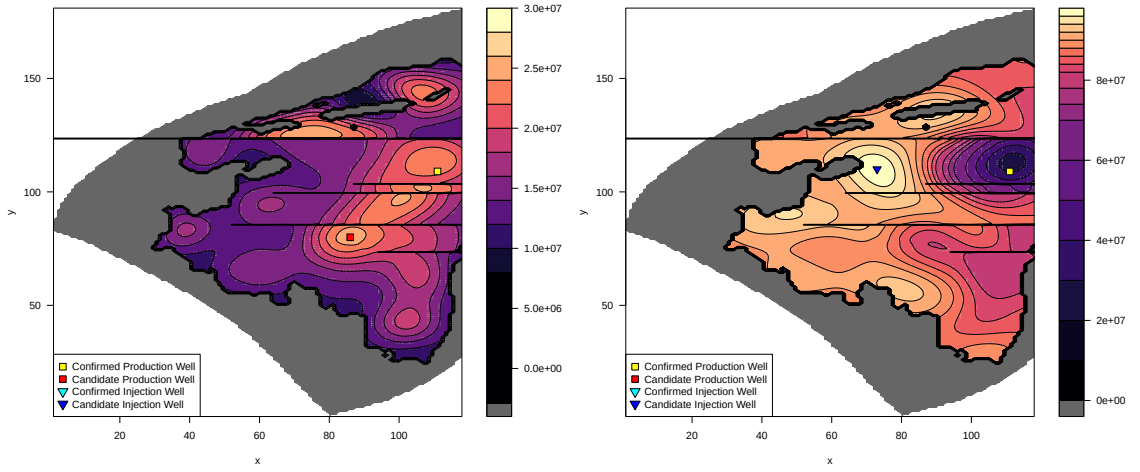
the TENSE framework to first determine the well type and then subsequently fix its location.

7.6.1 Well 2 Design for Simulations

Unlike for the first well where reservoir engineering knowledge unanimously favoured a producer, the choice of second well type is much more ambiguous with decisions dependent on the exact composition of the oil reservoir; information that is vastly uncertain due to the black-box nature of the model. It is therefore decided that two parallel analyses should be performed positioning both a producer and an injector well. In both cases, the same design as for the well 1, wave 1 analysis shown in Table 7.3 is used, although the ghost runs are included and excluded for the producer and injector respectively. This again enforces the prior constraint that a producer well should not be drilled into non-oil containing regions of the reservoir. In contrast, injector wells can be drilled anywhere in the field, including in regions containing only water, since they aim to sweep oil towards the producer wells. The re-use of the well 1, wave 1 design is justified for both cases because it was formulated as a sequentially optimal design with respect to minimising the mean emulator adjusted variance criterion over $\mathcal{X}$ without consideration for the oil-water boundary, that is omitting the ghost well locations. Simulations are performed over the full OLYMPUS ensemble, but also incorporating the first producer well.

7.6.2 Well 2 Emulation Results

Continuing the parallel analyses, the TENSE framework is applied to both the expected NPV as a function of the location of a producer and an injector well, with the formulation and prior specification of each as described for well 1 in Sections 7.4.2 and 7.5.2. Plots for a producer and an injector are shown in Figures 7.9a and 7.9b respectively, with contours coloured by the emulator adjusted expectation. Both respect the discontinuities induced by the faults. A red square and blue triangle are used respectively to denote the candidate well locations that maximise the expected



(a) Emulator adjusted expectation for well 2 as a producer.

(b) Emulator adjusted expectation for well 2 as an injector.

Figure 7.9: TENSE emulator adjusted expectation for the OLYMPUS expected NPV as a function of the location of well 2 providing a comparison between a producer (left) or an injector (right), given the confirmed location of the first producer well depicted by a yellow square. The candidate location for well 2 which maximises the emulator expectation is shown by the red square for a producer in 7.9a, and by a blue triangle in Figure 7.9b. Note the much higher NPVs are achieved by selecting an injector compared to a second producer well.

NPV. It is immediately evident that the choice of a second producer well results in substantially smaller emulator adjusted expectations than for an injector well, therefore determining that the second well is to be an injector positioned at the blue triangle. This is located at $(x, y) = (73, 110)$. Note that this does not violate reservoir engineering principles, whilst its location is consistent with expert insight. It is a suitable distance from the first producer well such that water flooding does not occur that would render the producer well uneconomical; the cause of the low purple region around the first producer well in Figure 7.9b, whilst also inducing a sufficiently high pressure gradient to sweep oil towards the producer well resulting in substantial oil production at earlier times, thus mitigating the high discounting effect, as depicted by the bright yellow region in the plot. The combination of drilling a producer followed by an injector well yields a substantial increase in the expected NPV, approximately 9.8×10^7 \$, versus a single producer only, approximately 2.3×10^7 \$.

7.7 Well 3 Analysis

This section presents an analysis for the third well location, given the first two well positions shown in Figure 7.9b. In Section 7.7.1 the well type and design are discussed, with the results of emulation presented in Section 7.7.2.

7.7.1 Well 3 Design for Simulations

Well 3 is suggested to be a second producer well. This is based on insight from oil reservoir engineers concerning pressure constraints having already positioned a producer and an injector. The design given in Table 7.3 is again used, plus the ghost wells, and is justified for the reasons outlined in Section 7.6.1. Simulations are evaluated for the 50 OLYMPUS geological realisations and include the two previously positioned wells. Note that if domain expert knowledge did not provide adequate certainty of the well type choice, a parallel analysis as in Section 7.6 could be performed, assuming sufficient computational resource availability. A parallel analysis is unnecessary here.

7.7.2 Well 3 Emulation Results

The TENSE framework is applied to the expected NPV as a function of the third well location, a producer, given the locations of the first two wells. The emulator formulation and prior specification is derived following the same approach as for the first well in Sections 7.4.2 and 7.5.2. Figure 7.10 shows the OLYMPUS map with contours coloured by the emulator adjusted expectation. The yellow square and cyan triangle depict the locations of the first producer and injector wells respectively, whilst the red square denotes the candidate second producer well location corresponding to $\mathbf{x}$ which maximises the expected NPV. This is located at $(x, y) = (85, 79)$, with the field development strategy for the first three wells described in Table 7.5. A comparison with Figures 7.3c and 7.3d for the well 1, wave 2 TENSE adjusted expectation illustrates broadly similar shaped contours, except for in the region between the

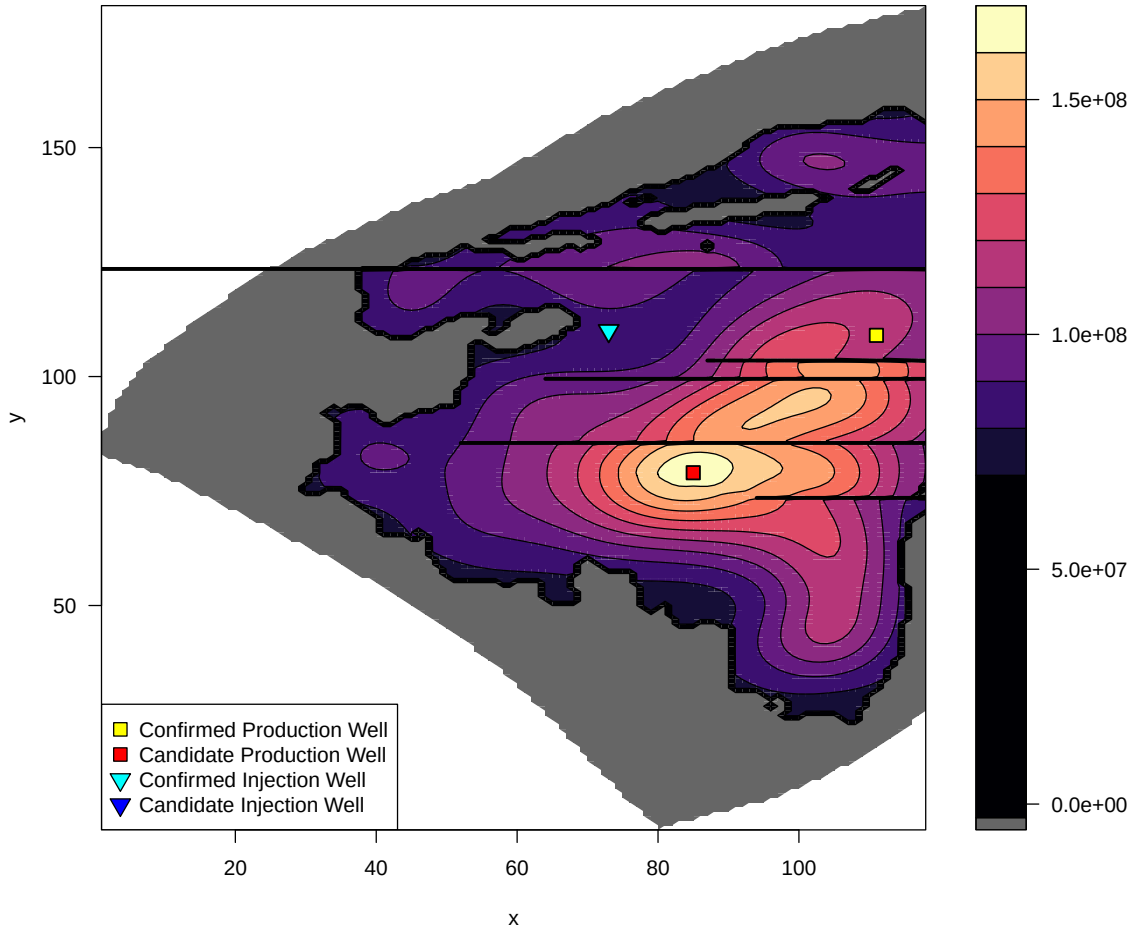


Figure 7.10: TENSE emulator adjusted expectation for the OLYMPUS expected NPV as a function of the location of well 3, a second producer well, given the confirmed locations of a first producer (yellow square) and injector wells (cyan triangle). The candidate location for well 3 as a producer which maximises the emulator expectation is shown by the red square.

faults 4 and 5 close to the already placed producer well where it would not be rational to position a second producer in adherence with oil reservoir engineering principles. Moreover, the relative position to the injector well is consistent maintaining a suitable distance to avoid water flooding, but with the potential for the water injection to sweep oil towards the well depending on the fault transmissibility. The result of adding a third well to the field development strategy is a further substantial increase in the expected NPV to approximately 1.6×10^8 \$, compared with 9.8×10^7 \$ for two wells.

Drilling Order	Well Type	x	y	Expected NPV
Well 1	Producer	111	109	2.3×10^7 \$
Well 2	Injector	73	110	9.8×10^7 \$
Well 3	Producer	85	79	1.6×10^8 \$

Table 7.5: TNO OLYMPUS Well Placement Optimisation Challenge field development strategy for the first three wells including drilling order, well type and placement, as well as the expected NPV after each well is positioned.

7.8 Well Placement Optimisation Conclusion

In this chapter we have presented an analysis of the TNO OLYMPUS Well Placement Optimisation Challenge demonstrating the novel TENSE methodology developed in Chapter 6 for the Bayesian emulation of computer models possessing general forms of structured discontinuities, motivated by this problem. Subsequently, this is used to provide decision support for the well placement strategy. First, an embedding surface is formulated encompassing the discontinuities induced by the partial geological fault boundaries, before discussing the resultant prior covariance structure. The TENSE framework is then applied to the expected NPV for the complex and computationally expensive OLYMPUS ensemble to sequentially select the location of the first three wells. This includes an application of the iterative decision support procedure developed in Section 3.6 to identify regions of good candidate well locations working with an experienced oil reservoir engineer to ultimately select the first well position. Moreover, the TENSE framework is employed to select the well 2 type. The TENSE methodology and derivation presented in Chapter 6, along with this application to the TNO OLYMPUS Well Placement Optimisation Challenge, is presented in [174] which is currently in submission to a journal.

A key part of the TENSE framework is the formulation of an embedding surface which captures the location, shape and magnitude of the discontinuities for the function of interest. In the TNO OLYMPUS Well Placement Optimisation Challenge these are induced by the geological fault boundaries. These possess a complex form with an endpoint that lies within the interior of the well location parameter

space; thus they cannot be handled by existing emulation methodology for computer models with discontinuities. The faults are of a known location and shape and are incorporated by the piecewise definition of the embedding surface, $v(x, y)$, for which the choice of a quadratic form is justified. A priori the magnitude of the discontinuities is approximated by the choice of the region boundary endpoint height combined with the quadratic structure of $v(x, y)$. This is subsequently validated by positioning pairs of simulations either side of several of the faults. The prior covariance structure is illustrated along vertical strips which emphasises how the choice of $v(x, y)$ within the TENSE framework is successfully able to handle the partial discontinuities.

For the well 1, wave 1 analysis a sequential design strategy was employed that successfully guards against potential early design termination resulting from uncertainty of the available computational resources to always deliver an optimal design. In addition, a collection of ghost well locations were chosen in the non-oil containing region of the OLYMPUS reservoir to exploit the prior knowledge that if a producer well is situated in a region of no oil, then its contribution to the expected NPV utility function will be zero. For this application it was also necessary to augment the design with an additional six points arranged in three pairs on either side of the faults. This was for the purpose of learning about the magnitude of the discontinuities.

Decision support is then performed sequentially for the first three well locations employing the TENSE approach. For each well and wave of the analysis, the fitted emulator is shown to respect the partial discontinuities whilst being continuous everywhere else, thus accurately and efficiently capturing the behaviour of the expected NPV utility function given only a limited collection of simulations. The iterative decision support procedure is employed for the well 1 location where extra simulations in the wave 1 non-implausible region, also designed via a sequential strategy, yields further emulator refinement with negligible uncertainty within multiple regions separated by partial discontinuities. Working with our collaborator, an expert in the petroleum industry, the information gained from the emulators for the expected

NPV and the quantiles is combined with insight and risk preferences to suggest a first producer well location. This provides a successful demonstration of how the iterative decision support framework incorporating (TENSE) emulators may be used in practice to guide real-world decision making processes.

Subsequent analysis utilising the TENSE methodology is performed for the second well location and is shown to be capable of efficiently assessing the optimal well type. This is achieved through a parallel analysis of optimally positioning a producer and an injector well, given the first producer well location, with results emphatically indicating that the second well should be an injector. The effectiveness of the TENSE approach is further corroborated in the well 3 analysis yielding a well configuration that is consistent with the prior beliefs of an experienced oil reservoir engineer. The field development strategy for the first three wells is summarised in Table 7.5.

The next stage of this analysis is to select the locations of subsequent wells, although the application presented in this chapter serves as a demonstration of the procedure, as well as the effectiveness of the TENSE framework for computer models with a finite collection of structured partial discontinuities. In addition, the full well placement strategy incorporates well trajectories; a problem not yet addressed with all wells using a vertical trajectory. An initial approach is to use emulators with a hemispherical parameter space describing the well endpoints. Other avenues to this research include consideration of the interaction between multiple wells. For example, after sequentially positioning wells there is the option to return and perturb the locations of all wells in order to further increase the expected NPV utility function. Alternatively, a more complex strategy that is anticipated to yield greater returns is to simultaneously position groups of k wells by extending the TENSE methodology for embedding a $2k$-dimensional location parameter space in $3k$-dimensions.

Chapter 8

Conclusion

The aim of this thesis was to develop Bayesian emulation, uncertainty quantification and robust decision support methodology for use with complex computer models in order to provide guidance for real world decision making problems. The major contributions from this research included:

1. Establishment of a hierarchical Bayesian emulation framework that exploits known constrained simulator behaviour.
2. Advancement of the uncertainty quantification methodology for performing decision support, specifically where there is a separation between the analysts and decision makers.
3. Formulation of an iterative decision support procedure.
4. Development of Bayesian emulators for computer models with structured partial discontinuities.

This was motivated by and demonstrated on the TNO OLYMPUS Field Development Optimisation Challenge from the petroleum industry, although the methodology is generalisable, and easily transferred and adapted to many other applications.

In Chapter 2 we reviewed existing methodology for the Bayesian uncertainty analysis of computer models with a focus on performing inferences about the corresponding real world physical systems. We began by comparing the full Bayesian and Bayes linear approaches to such analyses; our preference being the Bayes linear

framework for reasons of robustness and computational tractability. An extensive discussion was presented of Bayesian emulators as a stochastic belief specification for deterministic computer models. This included a comprehensive derivation of the Bayes linear update formulae for an emulator which were compared with the commonly used Gaussian process emulation methodology, whilst examining techniques for the prior specification of the mean function and residual process hyperparameters. Multilevel and multivariate emulators were also introduced, although not explicitly used in this thesis. An uncertainty quantification is of critical importance for linking the computer model to reality and thus making inferences about the real world system through the incorporation of terms such as structural model discrepancy. Finally, we introduced the history matching process as a powerful tool for identifying a set of model parameters that yield an acceptable match between the simulator output and observed data with respect to an uncertainty analysis.

The principal aim of this thesis is to develop methodology for decision support under uncertainty which is discussed in Chapter 3. We introduce the TNO OLYMPUS Field Development Optimisation Challenge as motivation and application for the subsequently developed methodology. A critique highlights how this is a synthetic challenge which inadequately represents the real world problem of devising optimal field development strategies; it is designed to stimulate research of ensemble optimisers, and instead we reformulate as a decision support problem for the expected NPV utility function. Notably, the particular stipulations of the TNO OLYMPUS Challenge ruled out using a sequential decision theory approach.

Ensembles of models are frequently used to portray forms of uncertainty, for example, geological uncertainty, however, their use greatly increases the computing times of already expensive analyses. We devised an Efficient Geological Ensemble Subsampling (EGES) technique to identify a representative subset of models that provides an effective mechanism for predicting the ensemble mean of quantities of interest, although it is acknowledged that there is room for further development. Emulation plays a vital role in such decision analyses with a hierarchical Bayesian

emulation framework constructed for the expected NPV utility function. Of particular importance is the decomposition into NPV constituents by model for which an emulator structure is conceived that exploits this known constrained behaviour leading to superior accuracy. Another benefit is the ability to step inside the hierarchy and investigate the behaviour of individual aspects of the utility function providing insight into the computer model behaviour and the quantities of greatest importance to achieve a high NPV. Further improvements to the hierarchical emulation framework could be realised such as by revising the structured emulators methods of change point estimation, classification and truncation, as well as via the refinement of the hierarchy uncertainty propagation.

Decision problems are traditionally addressed via either optimisation or decision theoretic approaches. These are contrasted with our preferred decision support that explicitly incorporates a separation between the analysts and decision makers, and hence acknowledges the induced uncertainty on the decision makers' utility by delivering a class of decisions that is robust to all major sources of such uncertainties. Furthermore, our decision support approach encompasses uncertainties that are challenging to elicit or quantify. A central part of this framework is the formulation of an uncertainty analysis to perform decision support, thus linking the utility function with the uncertainties. It is remarked that linear transformations of the utility function do not change the ranking of decisions; hence the conclusions of decision support are unaffected, with a technique developed to assess and remove such effects for cost parameters to achieve meaningful decision support. Future research may extend this to non-parametric forms of uncertainty, as well as the interaction between multiple types. Decision implementation error is frequently overlooked. We address this by combining expert knowledge with a fast approximation using Bayesian emulators with a natural extension encompassing targetted simulations anticipated to improve the accuracy.

The main goal is achieved through the development of an iterative decision support procedure, analogous to the history matching process, using an implausibility

measure to identify a class of decisions with respect to the uncertainty analysis. This addresses the problem of the separation of the analysts from the end decision makers and the resulting utility function uncertainty. A further important consideration is a framework for delivering this class (or classes) of decisions to the (committee of) decision maker(s), along with communicating any assumptions, insights or judgements used. There exist many directions for future research on decision support including:

- Handling multiple utility functions such as where these cannot be amalgamated into a single utility function; a potentially natural extension for multivariate implausibility measures. This may also be applicable to situations of multiple competing but known utility functions for a committee of decision makers. The returned decisions should satisfy all concerns where decision support can distinguish if any solutions exist.
- Techniques for the analyst or statistician to assess reasonable decision maker utility preferences, as well as communicating the results for each instance.
- An extension of decision support to mixed parameter types, such as categorical and ordinal parameters. The current procedure is designed for continuous numerical parameters, although this is likely to also require advancements to the emulation and uncertainty quantification methodology.
- The more complex challenge of sequential decision support; where interventions at future times are adjusted in light of new observations that are conditional on the original decisions. These must be chosen to both optimise the utility function, but also provide useful but uncertain future observation data designed to enhance the later decisions. This is far more representative of real world decision making problems where all decisions are not made before the initial time point. Further challenges arise when the intervention times are unknown.

In Chapter 4 we apply the methodology developed in Chapter 3 to the TNO OLYMPUS Well Control Optimisation Challenge. Implementation of the EGES technique efficiently reduces total simulator evaluation time. A method of formulating a targetted Bayesian design is devised using prior insight of physically realistic con-

trol strategies to reduce the volume of the decision parameter space, with sampling distributions chosen to target regions of interest whilst ensuring samples across the entire space. This may be adapted for use in other applications. A comparison of the Bayes linear and developed hierarchical emulation frameworks demonstrates the superior accuracy for relatively little additional computational costs, particularly versus the cost of evaluating the OLYMPUS ensemble. In the uncertainty quantification we provide a more in depth treatment of geological uncertainty than in the TNO OLYMPUS Well Control Optimisation Challenge, extending from 50 to the potentially infinite number of compatible underlying geologies. The first wave of iterative decision support is performed including an efficient comparison of alternative uncertainty specifications. This highlights the importance of a comprehensive uncertainty analysis whilst emphasising the enormous implications of accounting for linear transformations due to the utility function cost parameter uncertainty. Moreover, consideration of the treatment of geological uncertainty shows that the TNO approach is insufficient. Visualisation techniques from the history matching literature have been adapted to provide a graphical representation of decisions, enabling easier interpretation by the end decision makers.

Future work, not performed in this thesis due to constraints on time and computing resources, is to execute additional waves of iterative decision support. Another direction is to provide a more rigorous treatment of geological uncertainty via the underlying stochastic OLYMPUS geology model, if it were accessible. This represents the natural generalisation of the decision support procedure for use with stochastic simulators, and would likely incorporate the emulation of stochastic computer models methodology such as discussed in [2, 3, 7, 106].

Chapter 5 forms the commercial client placement part of my iCASE PhD studentship with Emerson. The purpose was to demonstrate the benefits of this research to enhance Emerson's software capability. An analysis of the well control problem is performed for their Jade model beginning in a similar vein as for the TNO OLYMPUS Well Control Optimisation Challenge application with important extensions to

the uncertainty quantification to assess structural model discrepancy and decision implementation error, before conducting multiple waves of iterative decision support encompassing a revision of the uncertainty analysis. A validation of the final non-implausible decision support region is performed demonstrating: the accuracy and effectiveness of the EGES technique as a representation for the ensemble mean NPV; the successful emulation of the ensemble mean NPV; and the accurate link between the expected and ensemble mean NPV. In both this and the OLYMPUS analysis, conservative uniform estimates are used for each of the uncertainties. An area of future work is to construct more sophisticated uncertainty representations using variance emulation techniques to obtain parameter dependent uncertainties and attain superior decision support results. Moreover, a notable conclusion from both analyses is the dominance of geological uncertainty: further decreases to the non-implausible decision volume are only achievable using a larger initial ensemble than 50 models. This suggests that more serious care is required in the treatment of geological uncertainty in the petroleum industry, although the EGES technique can alleviate some of the additional computational burden.

We develop novel methodology for the Bayesian emulation of computer models with structured partial discontinuities in Chapter 6 , an extremely challenging problem, motivated by the TNO OLYMPUS Well Placement Optimisation Challenge. This is accomplished by constructing emulators for an embedding of the input parameter space in higher dimensions where the embedding surface is chosen to characterise the known locations and shape of the (partial) discontinuities, with the separation reflecting the magnitude of each discontinuity. Derivation of the embedded parameter variance matrix shows that this depends on the choice of embedding surface and the parameters, necessitating that non-stationary emulators are employed. Taken together, these steps correct for an induced warping effect whilst leading to global emulators that are correct up to first-order, which is sufficient for most reasonable choices of the embedding function. The complete framework is referred to as the Torn Embedding Non-Stationary Emulation (TENSE) approach.

Applications to known examples demonstrates its effectiveness, as well as flexibility to functions possessing complex non-linear forms of partial discontinuities. There remains many potential areas for future research including:

- A natural extension of the derivation to handle different embedding dimensions, for example, d in $d+1$, or d in $d+m$ to address multiple hypersurfaces of partial discontinuities.
- Methods to learn the optimal embedding surface, for example, via maximum likelihood estimation (MLE).
- Formulation of the embedding surface when the location and/or shape of the discontinuities are unknown, potentially also via MLE.
- Generalisation of the TENSE framework beyond univariate emulators to handle such facets as multivariate outputs or known boundary information.
- Further address the design problem in the presence of partial discontinuities where standard design methodology over parameter hypercubes is no longer applicable. This may take the form of a design criterion to optimise given a choice of embedding.

The TENSE methodology is applied to the TNO OLYMPUS Well Placement Optimisation Challenge in Chapter 7 where a sequential well location analysis is performed with the aim of devising a good solution, although it is acknowledged that this will be sub-optimal compared to full backwards induction. A quadratic form for the embedding surface is chosen and justified, with the prior covariance structure graphically illustrated to demonstrate how the embedding successfully induces the desired partial discontinuity structure. Consideration is also given to sequential design methodology to address uncertainty in the available computing resources, whilst also incorporating additional knowledge of model output at no extra cost through the use of ghost runs. The TENSE methodology is applied and successfully captures the structure of the expected NPV utility surface as a function of well location, whilst also suggesting classes of well placements that are consistent with oil reservoir engineering theory. Iterative decision support is performed demonstrating

how to synthesise this insight with expert knowledge and experience to obtain a well configuration. The next step is to select the locations of subsequent wells.

There exist numerous extensions and improvements to our analysis of the TNO OLYMPUS Well Placement Optimisation Challenge. Firstly, it is anticipated that accounting for the interaction between multiple wells will yield a superior solution. This may be addressed through the perturbation of already selected well locations to yield an increase in the expected NPV utility function. Alternatively, the placement of groups of k wells through the TENSE approach by embedding the $2k$-dimensional location parameter space in $3k$-dimensions. Another direction is to examine the full field development optimisation problem which also incorporates well trajectories. An initial approach is to combine the existing strategy with emulators possessing a hemispherical parameter space describing the well endpoints.

The TNO OLYMPUS Joint Field Development and Well Placement Optimisation Challenge has not been addressed in this thesis and remains a complex problem with limited progress to date. Current industry approaches generally use a two stage optimisation procedure to first design a field development plan, before formulating a well control strategy [12, 109]. This is likely to be suboptimal. An interesting area of future work is to combine the methodology of Chapters 3 and 6 to devise joint well placement and control strategies within the decision support framework.

Computer models are increasingly prevalent across science, industry and government, to represent complex real world systems, and inform our future actions through decision making processes. The research presented in this thesis represents a substantial contribution to the Bayesian emulation and uncertainty analysis of computer models for decision support literature. We have advanced uncertainty representation and quantification techniques in the context of decision support, which are encompassed within a formal iterative decision support framework to deliver a robust class of decisions, along with accompanying information, to a committee of decision makers, in a setup that accurately portrays real world decision making processes. The developed methodology thus forms an important toolkit for addressing

the substantial problems of providing robust decision support using computer simulators across a broad spectrum of applications, extending far beyond the petroleum industry.

Nomenclature

$(\mu + \lambda)$ ES $(\mu + \lambda)$ Evolutionary Strategy, page 96

α_{EGES} Regression intercept coefficient in EGES linear model, see Equation (3.2.1), page 102

α_i Standardised truncation lower bound for a truncated Gaussian process within a structured emulator for computer model output i, page 112

α_i^2 Eigenvalues of Σ_{3D} with respect to the w-basis, page 343

$\widetilde{\text{NPV}}_j(\cdot)$ Average discounting approximate NPV for model j, defined in Equation (3.3.6), page 118

$\bar{f}(\cdot)$ Ensemble mean of a computer model output, page 101

β_i Standardised truncation upper bound for a truncated Gaussian process within a structured emulator for computer model output i, page 112

$\beta_{0,\widetilde{\text{NPV}}_j}$ Intercept in simple linear model linking the exact and approximate NPV for model j, see Equation (3.3.11), page 121

$\beta_{1,\widetilde{\text{NPV}}_j}$ Slope coefficient in simple linear model linking the exact and approximate NPV for model j, see Equation (3.3.11), page 121

$\beta_{\text{C},0,i}$ Intercept in the uncertainty quantification linear model for the i^{th} sampled utility function cost parameter setting, see Equation (3.5.1), page 139

$\beta_{\text{C},0,i}$ Slope coefficient in the uncertainty quantification linear model for the i^{th} sampled utility function cost parameter setting, see Equation (3.5.1), page 139

β_{ij} Unknown scalar regression coefficient for the j^{th} deterministic regression function in an emulator for computer model output i, see Equation (2.4.1), page 23

$\beta_{k,\text{EGES}}$ Regression coefficient of the k^{th} selected individual model output in EGES linear model, page 102

$\hat{\boldsymbol{\beta}}_{\text{EGES}}$ Vector of estimates of EGES linear model coefficients, page 123

$\hat{\boldsymbol{\beta}}_{GLS}$ Generalised Least Squares estimate for regression parameters, $\boldsymbol{\beta}$, page 41

$\hat{\boldsymbol{\beta}}_{OLS}$ Ordinary Least Squares estimate for the regression parameters, $\boldsymbol{\beta}$, page 41

$\mathcal{O}(\cdot)$ Big "O" notation describing the limiting behaviour of computation time, page 37

$\mathbf{B}$ Collection of randon quantities in Section 2.2.2, page 13

$\boldsymbol{\beta}_i$ Vector of unknown regression coefficients for computer model output i in an emulator, see Equation (2.4.1), page 23

$\mathbf{C}$ Collection of randon quantities in Section 2.2.2, page 13

$\mathbf{c}$ Vector of utility function cost parameters, page 130

$\mathbf{c}_0$ Base utility function cost parameter scenario, page 139

$\mathbf{c}_i$ The i^{th} sampled utility function cost parameter vector, page 131

$\mathbf{D}$ Collection of randon quantities in Section 2.2.2, page 13

$\mathbf{D}$ Vector of observation data for full Bayesian posterior computation in Section 2.2.1, page 12

$\mathbf{d}$ Vector of decision parameters, page 10

$\mathbf{d}^*$ Vector of decision parameters which maximises the expected utility function, defined in Equation (3.4.6), page 132

$\mathbf{d}^*(\mathbf{c})$ Vector of decision parameters which maximises the expected utility function for a fixed vector of cost parameters, page 132

$\mathbf{d}_{\max,3\text{CIupr}}$ Decision parameter vector which maximises the 3 adjusted standard deviation credible interval upper bound, page 269

$\mathbf{d}_{\max,\exp}$ Decision parameter vector which maximises the utility function emulator adjusted expectation, page 269

$\mathbf{d}_{\max}$ Vector of decision parameters which maximises an objective function, defined in Equation (3.4.1), page 126

$\boldsymbol{\delta}_{\mathrm{I}}$ Vector of implemented decisions, see Equation (3.5.4), page 141

$\mathbf{e}$ Vector of observation errors, see in Definition 2.7.1, page 61

$\mathbf{e}_i$ Canonical basis vector in the i^{th} direction in $\mathbb{R}^D$, page 29

$\mathbf{e}_{\mathrm{I}}(\cdot)$ Random quantity representing the implementation uncertainty in the decisions, page 141

$\mathbf{F}$ Vector of computer model evaluations in Section 2.4.5, page 38

$\mathbf{F}_i$ Vector of computer model evaluations for output i, page 26

$\mathbf{F}_i'$ Subset of simulated output satisfying a constraint on the change point upper bound for computer model output i, page 111

$\mathbf{g}(\mathbf{x})$ Deterministic regression vector function evaluated at parameter vector $\mathbf{x}$, page 46

$\mathbf{g}(\cdot)$ Vector valued deterministic global mean function used in multivariate emulation, $\mathbf{g}\colon \mathcal{X} \to \mathbb{R}^p$, see Equation (2.5.1), page 53

$\mathbf{g}_i(\mathbf{x}_{A_i})$ Vector function of known deterministic regression or global mean functions of the active parameters, $g_{ij}(\mathbf{x}_{A_i})$, in an emulator for computer model output i, see Equation (2.4.1), page 23

$\mathbf{h}_i$ Rotated and scaled basis vector in $\mathbb{R}^2$, defined in Equation (6.3.46), page 349

$\mathbf{m}(\cdot)$ Multivariate emulator mean function, $\mathbf{m}\colon \mathcal{X} \to \mathbb{R}^q$, defined in Equation (2.5.3), page 54

$\mathbf{p}$ Parameter vector in the rotated t-basis, see Equation (6.3.43), page 348

$\mathbf{q}$ Parameter vector in the rotated and scaled h-basis, see Equations (6.3.48) and (6.3.49), page 349

$\mathbf{r}(\mathbf{x}, X)$ Column vector of correlations in $\mathbb{R}^n$ between $\mathbf{x}$ and each design point in X used in Section 2.4.5, $\mathbf{r}(\mathbf{x}, X) \in \mathbb{R}^n$, page 39

$\mathbf{t}_i$ Rotated basis vector in $\mathbb{R}^2$ by angle ϕ, defined in Equations (6.3.41) and (6.3.42), page 348

$\boldsymbol{\theta}$ Vector of random quantities or parameters in Section 2.2.1, page 12

$\boldsymbol{\theta}_i$ Vector of correlation length hyperparameters for computer model output i, page 24

$\mathbf{u}$ Vector in $\mathbb{R}^n$ of realisations of the emulator weakly stationary stochastic process at the design inputs in Section 2.4.5, page 39

$\mathbf{u}$ Vector of $\mathbf{q} \in \mathbb{R}^2$ in h-basis embedded in $\mathbb{R}^3$ in Chapter 6, page 350

$\mathbf{v}$ Vector of $\mathbf{x} \in \mathbb{R}^2$ embedded in $\mathbb{R}^3$ in Chapter 6, page 350

$\mathbf{v}(\cdot)$ Embedding function from a 2- to 3-dimensional parameter space, page 334

$\mathbf{v}^{(i)}$ Design point parameter vector in the embedded parameter space, page 336

$\mathbf{w}$ Vector in $\mathbb{R}^n$ of realisations of the emulator nugget term in Section 2.4.5, page 39

$\mathbf{w}_i$ Orthonormal basis vector in $\mathbb{R}^3$, defined with respect to the tangent plane of $v(x, y)$ at $\mathbf{x}_0$, page 342

$\mathbf{x}$ Vector of model parameters in Chapters 2 to 6, page 10

$\mathbf{x}$ Vector of well location parameters in the TNO OLYMPUS Well Placement Optimisation Challenge in Chapter 7, page 362

$\mathbf{x}^*$ "Best" input parameter vector to a computer model, meaning that $\mathbf{x}^*$ best represents the system properties that result in system behaviour $\mathbf{y}$, page 68

$\mathbf{x}^{(k)}$ Design point parameter vector, page 26

$\mathbf{x}_{\mathcal{I}}$ Sub-vector of linear model predictors in $\mathcal{I}$ in Appendix A, page 432

$\mathbf{x}_{A_i}$ Sub-vector of the parameters in $\mathbf{x}$ deemed to be active inputs for computer model output i, see Equation (2.4.1), page 23

$\mathbf{x}_0$ Reference parameter vector, page 340

$\mathbf{y}$ Vector of the true values of quantities of interest for a physical system, $\mathbf{y} \in \mathbb{R}^s$, page 61

$\mathbf{z}$ Scalar or vector of observation data, $\mathbf{z} \in \mathbb{R}^s$, page 61

$\mathbf{0}$ Zero vector, page 38

$\mathrm{Cov}[\cdot, \cdot]$ Covariance between two random quantities, as defined in the Bayes linear paradigm, page 13

$\mathrm{Cov}_D[\cdot, \cdot]$ Bayes linear adjusted covariance between two random quantities given data D, either a scalar or a matrix, see Equation (2.2.5), page 13

$\mathrm{Cov}_F[\cdot, \cdot]$ Bayes linear emulator adjusted covariance given computer model evaluations, $\mathbf{F}$, page 40

c^l_{jk,t_i} Change point lower bound for an NPV constituent, where j and k represent the well type and number, and t_i is a control interval start date, defined in Equation (3.3.2), page 108

c^u_{jk,t_i} Change point upper bound for an NPV constituent, where j and k represent the well type and number, and t_i is a control interval start date, defined in Equation (3.3.1), page 107

c_{jk,t_i} Change point parameter for an NPV constituent, where j and k represent the well type and number, and t_i is a control interval start date, page 107

d_{jk,t_i} Scalar decision parameter corresponding to an NPV constituent, where j and k represent the well type and number, and t_i is a control interval start date, page 107

Δt_i Time interval in NPV calculations, page 87

δ Tolerance parameter specifying the exploration distance below the maximum in decision support in Chapter 7, page 375

Δ_i Difference constraint between time consecutive control parameters, page 166

δ_i Maximum absolute error in slope region within emulators exploiting known simulator behaviour for NPV constituents, page 112

δ_i Tolerance parameter in change point estimation for computer model output i, page 107

Δ'_i Rescaled difference constraint between time consecutive control parameters, page 166

$\delta_{i,l}$ Tolerance parameter in change point lower bound estimation for computer model output i, page 107

$\delta_{i,u}$ Tolerance parameter in change point upper bound estimation for computer model output i, page 107

$\pi(\mathbf{D} \mid \boldsymbol{\theta})$ Likelihood for data $\mathbf{D}$ given random quantities or parameters $\boldsymbol{\theta}$, page 12

$\text{diag}\{\cdot\}$ Diagonal matrix with the specified arguments, page 24

Ω Decision parameter space, page 10

Ω^* Class of decision theoretic optimal decisions for a finite set of cost parameter settings, defined in Equation (3.4.7), page 132

Ω_0 Initial decision parameter space in iterative decision support, page 146

Ω_k Non-implausible decision parameter space after wave k of iterative decision support, page 146

$\mathbb{E}[\cdot]$ Expectation of a random quantity, as defined in the Bayes linear paradigm, page 13

$\mathbb{E}_D[\cdot]$ Bayes linear adjusted expectation of a random quantity given data D, either a scalar or vector, see Equation (2.2.3), page 13

$\mathbb{E}_F[\cdot]$ Bayes linear emulator adjusted expectation given computer model evaluations, $\mathbf{F}$, page 40

$\ell_d(\cdot)$ Arbitrary positive length-scale function used in the Gibbs covariance function, see Equation (2.4.10), page 32

$\emptyset$ The empty set, page 73

$\mathbb{E}[\text{NPV}](\cdot)$ Expected NPV, defined in Equation (3.1.2), page 87

$\varepsilon_{\text{C}}(\cdot)$ Random quantity or residual representing the utility function cost parameter uncertainty, page 139

$\hat{\varepsilon}_{\text{C}}(\mathbf{d}, \mathbf{c}_i)$ Utility function cost parameter uncertainty linear model residual, page 139

$\varepsilon_{\text{DS}}(\cdot)$ Random quantity representing the collective uncertainty on the decision making process, defined in Equation (3.6.1), page 144

ε_{G} Random quantity representing the induced geological uncertainty, page 140

$\varepsilon_{\text{I}}(\cdot)$ Random quantity representing the induced decision implementation error, page 141

$\varepsilon_{\mathrm{MD}}(\cdot)$ Random quantity representing structural model discrepancy, with a potential parameter dependency, see Definition 2.7.2, page 62

$\varepsilon_{\mathrm{MD,ext}}(\cdot)$ Random quantity representing external structural model discrepancy, with a potential parameter dependency, see Equation (2.7.7), page 63

$\varepsilon_{\mathrm{MD},i}$ Structural model discrepancy for computer model output i, page 68

$\varepsilon_{\mathrm{MD,int}}(\cdot)$ Random quantity representing internal structural model discrepancy, with a potential parameter dependency, see Equation (2.7.7), page 63

b_{jk,t_i} Extrapolation cut-off for an NPV constituent, where j and k represent the well type and number, and t_i is a control interval start date, page 110

$\hat{f}_{\mathrm{LM}}(\mathbf{x})$ Linear regression model prediction at parameter vector $\mathbf{x}$, page 50

f^+ Current highest simulated expected NPV, page 375

Γ Standard deviation matrix in Section 2.4.6, page 51

$\Gamma(\cdot)$ Gamma function, see Table 2.1, page 28

$\mathcal{GP}(m_i(\cdot), \kappa_i(\cdot,\cdot))$ Gaussian process emulator with mean function $m_i(\cdot)$ and covariance function $\kappa_i(\cdot,\cdot)$, see Equation (2.4.13), page 35

$\mathcal{GP}_q(\mathbf{m}(\cdot), \kappa(\cdot,\cdot)\Sigma_{\mathrm{MV}})$ Multivariate Gaussian process for a q-dimensional random quantity with specified mean function and a separable covariance structure, defined in Equation (2.5.2), page 54

$\hat{\alpha}_{\mathrm{EGES}}$ Estimate of EGES linear model intercept coefficient, page 123

$\hat{\beta}_{k,\mathrm{EGES}}$ Estimate of EGES linear model k^{th} selected individual model coefficient, page 123

$\hat{\varepsilon}_{\mathrm{LM}}(\mathbf{x})$ Standardised linear regression residuals at parameter vector $\mathbf{x}$, page 50

I_n $n \times n$ identity matrix, page 41

$I_{\mathrm{DS}}(\cdot)$ Decision support implausibility measure, defined in Equation (3.6.3), page 145

$I'_{\mathrm{DS}}(\cdot)$ Alternative form of the decision support implausibility measure, defined in Equation (7.4.2), page 376

$\perp\!\!\!\perp$ Independent, page 58

$\mathbb{1}_{\{\cdot\}}$ Indicator function, see Equation (2.4.4), page 25

$\kappa_i(\mathbf{x}, \mathbf{x}')$ Prior covariance for a univariate Bayes linear emulator for computer model output i, defined in Equation (2.4.12), page 33

$\kappa(\cdot, \cdot)$ Covariance function or kernel, page 27

$\kappa_{\mathrm{NS}}(\cdot, \cdot)$ Non-stationary covariance function, defined in Equation (6.3.64), page 353

$\kappa_{\mathrm{S}}(\cdot, \cdot)$ Stationary isotropic covariance function, page 353

λ_i Weighted average discounting factor, defined in Equation (3.3.5), page 118

Λ Diagonal matrix in $\mathbb{R}^{3\times 3}$ where $\Lambda_{ii} = \alpha_i^2$, page 343

$|\cdot|$ Absolute value, page 51

$\tilde{\mathbf{x}}$ Extension of a parameter vector $\mathbf{x}$ to include a constant, $\tilde{\mathbf{x}} = (1, \mathbf{x})$, page 31

$\mathbf{a}_i$ Vector of coefficients for the linear combination of data which minimise the expected squared error for the i^{th} random quantity, page 13

$\mathbf{S}$ Collection of parameter vectors used as a substitute to the full parameter space in evaluating a minimax design, defined in Equation (2.3.2), page 21

$\mathcal{U}[\cdot, \cdot]$ Uniform distribution, page 168

$\mathcal{C}$ Cost parameter space, page 131

$\mathcal{C}'$ Finite set of cost parameters vectors, page 132

$\mathcal{D}$ Computer model design of experiments, page 20

$\mathcal{D}_i'$ Subset of design satisfying constraint on the change point upper bound for computer model output i, page 111

$\mathcal{D}_k$ Design for wave k of iterative decision support, page 146

$\mathcal{D}_{\mathrm{I}}$ Design to assess decision implementation error, page 142

$\mathcal{D}_{av}$ Minimum average distance design, defined in Equation (2.3.6), page 22

$\mathcal{D}_{Mm}$ Maximin design, defined in Equation (2.3.2), page 21

$\mathcal{D}_{mM}$ Minimax design, defined in Equation (2.3.4), page 21

$\mathcal{F}_{m,n-p}$ Fisher-Snedecor distribution with m and $n-p$ degrees of freedom, see Equation (2.4.49), page 51

$\mathcal{I}$ Set of internal structural model discrepancy types in Section 2.7.4, page 64

$\mathcal{I}$ Subset of linear model predictors in Appendix A, page 431

$\mathcal{TN}(\mu, \sigma^2, a, b)$ Truncated normal distribution with mean μ, variance σ^2, lower and upper truncation bounds a and b respectively, page 258

$\mathcal{X}$ Model parameter space in Chapters 2 to 6, page 10

$\mathcal{X}$ Well location parameter space in the TNO OLYMPUS Well Placement Optimisation Challenge in Chapter 7, page 362

$\mathcal{X}^*$ History matching non-implausible region, defined in Equation (2.8.9), page 72

$\mathcal{X}_0$ Initial parameter space in history matching, page 73

$\mathcal{X}_1 \times \mathcal{X}_2$ Tensor product of two parameter spaces, page 32

$\mathcal{X}_k$ Non-implausible parameter space after wave k of history matching in Section 2.8, page 73

$\mathcal{X}_k$ Non-implausible well location parameter space after wave k of iterative decision support in Chapter 7, page 376

$\mathcal{X}_{1,A}$ Well 1 sequential design region defined as the iterative decision support non-implausible region in Section 7.5, page 379

$\mathcal{X}_{1,B}$ Well 1 sequential design region defined using the emulator adjusted expectation in Section 7.5, page 379

VR Variance ratio history matching stopping rule for a univariate computer model output, defined in Equation (2.8.12), page 79

VR_{multi} Variance ratio history matching stopping rule for multivariate computer model output, defined in Equation (2.8.13), page 79

$\text{mean}_A(g(\cdot))$ Arithmetic sample mean of the function $g(\cdot)$ over the set A, page 372

$\overline{\text{NPV}}(\cdot)$ Ensemble mean NPV, defined in Equation (3.1.2), page 87

$\mu_i(\mathbf{x})$ Prior expectation for a univariate Bayes linear emulator for computer model output i, defined in Equation (2.4.11), page 33

$\boldsymbol{\mu}_\beta$ Prior mean for the regression coefficients in an emulator, $\boldsymbol{\mu}_\beta \in \mathbb{R}^p$, page 38

∇ Gradient operator in vector calculus, page 342

$\mathcal{N}_{n_1}$ Multivariate normal distribution for an n_1-dimensional random quantity, see Lemma 2.4.9, page 36

$\text{NPV}_j(\cdot)$ NPV for model j in an ensemble, defined in Equation (3.1.1), page 87

ω_j Ensemble mean NPV weight for model j, page 123

δ_i Scaled difference between normalised parameters in Section 4.3, page 166

$\boldsymbol{\delta}$ Vector of scaled differences between normalised parameters in Section 4.3, page 167

v_{q_1} Partial derivative of two-dimensional embedding surface with respect to q_1 and evaluated at $\mathbf{q}_0$, defined in Equation (6.3.58), page 351

v_{q_2} Partial derivative of two-dimensional embedding surface with respect to q_2 and evaluated at $\mathbf{q}_0$, defined in Equation (6.3.59), page 351

v_x Partial derivative of two-dimensional embedding surface with respect to x and evaluated at $\mathbf{x}_0$, page 341

v_y Partial derivative of two-dimensional embedding surface with respect to y and evaluated at $\mathbf{x}_0$, page 341

ϕ Rotation angle used to define the t-basis in $\mathbb{R}^2$, $\phi \in [0, \pi)$, page 348

$\Phi(\cdot)$ Standard normal cumulative distribution function, page 112

$\phi(\cdot)$ Standard normal distribution probability density function, page 112

$\pi_c(\mathbf{c})$ Prior distribution for the utility function cost parameters, page 131

$\pi_x(\mathbf{x})$ Prior distribution for the model parameters, page 131

$\pi_{\mathrm{I}}(\cdot \mid \cdot)$ Prior distribution for the uncertainty in the implemented decisions, page 141

ρ Proportion of the residual variance attributed to the emulator nugget term, $\rho \in [0, 1]$, page 33

$\rho_p(\mathbf{x}, \mathcal{D})$ Minimum p-norm distance between parameter vector $\mathbf{x}$ and its nearest neighbour in the design $\mathcal{D}$, defined in Equation (2.3.3), page 21

$\rho_p(\cdot, \cdot)$ p-norm between two vectors of random quantities, defined in Equation (2.3.1), page 20

$\mathrm{RSS}_{\mathcal{I}}$ Residual Sum of Squares for a linear model containing predictors in the set $\mathcal{I}$ in Appendix A, page 431

$R_{3\mathrm{D}}$ Rotation matrix between the standard and t-bases embedded in $\mathbb{R}^3$, defined in Equation (6.3.52), page 350

$R_{2\mathrm{D}}$ Rotation matrix between the standard and t-bases in $\mathbb{R}^2$, defined in Equation (6.3.50), page 349

$\mathrm{sd}[\cdot]$ Standard deviation of a random quantity, page 297

Σ Covariance matrix, page 27

σ^2 Variance hyperparameter, page 27

σ_e^2 Observation error variance, page 61

σ_i^2 Residual variability after accounting for the global mean function of an emulator for computer model output i, page 33

$\sigma_{\widetilde{\mathrm{NPV}}_j}$ Residual standard error of the linear model linking the exact and approximate NPV for model j, page 122

$\Sigma_{\mathrm{MD,int},j}(\cdot)$ Internal structural model discrepancy of type j variance matrix, with a potential parameter dependency, page 64

σ_{lm} Estimate of the residual standard error of a linear model, page 174

$\sigma_{u_i}^2$ Variance hyperparameter in the covariance function used in the emulator for computer model output i, page 24

$\sigma_{w_i}^2$ Variance hyperparameter for the nugget term in an emulator for computer model output i, see Equation (2.4.4), page 25

Σ_β Prior variance matrix of the regression coefficients in an emulator, $\Sigma_\beta \in \mathbb{R}^{p\times p}$, page 32

$\Sigma_{\beta,\mathrm{EGES}}$ Variance matrix of EGES linear model coefficient estimates, page 123

Σ_e Observation error variance matrix, page 62

$\sigma_{e_\mathrm{I},i}$ Standard deviation for sampling the i^{th} decision parameter for an assessment of decision implementation error, page 258

$\hat{\sigma}_{\mathrm{EGES}}$ Estimate of EGES linear model residual standard error, page 123

$\hat{\sigma}_{\mathrm{LM}}$ Estimated linear model standard error, page 50

$\Sigma_{\text{MD}}(\cdot)$ Structural model discrepancy variance matrix, with a potential parameter dependency, page 63

σ^2_{MD} Structural model discrepancy variance, page 63

$\Sigma_{\text{MD,ext}}(\cdot)$ External structural model discrepancy variance matrix, with a potential parameter dependency, page 66

Σ_{MV} Covariance matrix for multivariate computer model outputs, $\Sigma_{\text{MV}} \in \mathbb{R}^{q\times q}$, page 54

Ω Short notation for $\Sigma+\sigma_w^2 I_n$, defined in Equation (2.4.25), used in Section 2.4.5, page 40

Σ_{3D} Covariance matrix for the 3-dimensional embedded input parameters, page 335

$\Sigma^{\mathbf{h}}_{\text{3D}}$ Covariance matrix for the 3-dimensional embedded input parameters with respect to the h-basis, page 350

Σ_{2D} Covariance matrix for the 2-dimensional input parameters, page 335

$\Sigma^{\mathbf{h}}_{\text{2D}}$ Covariance matrix for the 2-dimensional input parameters with respect to the h-basis, page 349

$\Sigma^{\mathbf{t}}_{\text{2D}}$ Covariance matrix for the 2-dimensional input parameters with respect to the t-basis, page 348

$\text{sign}(\cdot)$ Sign function taking values 1 or -1 when the argument is positive or negative respectively, page 328

S_{3D} Scaling matrix between the t- and h-bases embedded in $\mathbb{R}^3$, defined in Equation (6.3.52), page 350

S_{2D} Scaling matrix between the t- and h-bases in $\mathbb{R}^2$, defined in Equation (6.3.50), page 349

τ Discounting time interval in NPV calculations, page 87

Θ Space of random quantities or parameters, page 12

θ_i^2 Correlation lengths in the directions of the t-basis vectors for $i = 1, 2$ in Chapter 6, page 348

θ_{ik} Correlation length hyperparameter for the k^{th} parameter and computer model output i, page 24

$\pi(\boldsymbol{\theta} \mid \mathbf{D})$ Posterior probability distribution for random quantities or parameters $\boldsymbol{\theta}$ given data $\mathbf{D}$, page 12

$\pi(\boldsymbol{\theta})$ Prior probability distribution for random quantities or parameters $\boldsymbol{\theta}$, page 12

$\tilde{D}$ Dimension of model parameter space in Section 2.1.3, $\tilde{D} \in \mathbb{N}$, page 10

$\text{Tr}(\cdot)$ Trace of a matrix, page 79

$U_{\max}$ Maximum utility achieved across all simulations within a design, page 145

$\text{Var}[\cdot]$ Variance of a random quantity, as defined in the Bayes linear paradigm, page 13

$\text{Var}_D[\cdot]$ Bayes linear adjusted variance of a random quantity given data D, either a scalar or a matrix, see Equation (2.2.4), page 13

$\text{Var}_F[\cdot]$ Bayes linear emulator adjusted variance given computer model evaluations, $\mathbf{F}$, page 40

ε_i Random quantity representing a form of uncertainty determined by subscript i, page 58

$\varepsilon_{\widetilde{\text{NPV}}_j}$ Residual in simple linear model linking the exact and approximate NPV for model j, see Equation (3.3.11), page 121

$\varepsilon_{\text{EGES}}(\cdot)$ Residual of EGES linear model, page 102

$\varepsilon_{\text{MD,int},j}(\cdot)$ Random quantity representing internal structural model discrepancy of type j, with a potential parameter dependency, page 64

$\mathcal{V}$ Embedded parameter space, page 334

$X_{\mathrm{EGES}}(\mathbf{d})$ Input vector to EGES linear model, page 123

$\zeta(\cdot)$ Stochastic process, defined in Definition 2.4.1, page 24

A Linear embedding operator for raising the 2-dimensional position vector $\mathbf{x}$ onto the corresponding location on the 3-dimensional tangent plane of the embedding surface at $\mathbf{x}_0$, see Equation (6.3.15), page 341

A_{BL} Set of active decision parameters for a Bayes linear emulator, page 174

a_i Truncation lower bound for a truncated Gaussian process for computer model output i, page 112

B Matrix of regression coefficients in multivariate emulation, $B \in \mathbb{R}^{p\times q}$, see Equation (2.5.1), page 53

$b(y)$ Function which linearly interpolates between the interior end points of two neighbouring parallel partial discontinuities, see Equation (6.3.9), page 337

B_i Scalar random quantity for index i in Section 2.2.2, page 13

b_i Truncation upper bound for a truncated Gaussian process for computer model output i, page 112

$b_i(y)$ Functions which linearly interpolates between the interior end points of two neighbouring parallel partial faults indexed by i, see Equations (6.3.9) and (7.2.1), page 365

c Implausibility cut-off used in history matching and iterative decision support, page 70

$c(\mathbf{x}-\mathbf{x}')$ Function of the difference between two parameter vectors, see Definition 2.4.2, page 27

C_i Scalar random quantity for index i in Section 2.2.2, page 13

c_i Normalising constant for basis vector $\mathbf{w}_i$, see Equations (6.3.21) to (6.3.23), page 343

$C_{\mathcal{I}}$ Mallows' C_p, defined in Equation (A.0.1), page 431

c_{stop} History matching stopping rule cut-off, page 79

D Dimension of (decision) parameter space, $D \in \mathbb{N}$, page 10

d Discount factor in NPV calculations, page 87

$D(\cdot)$ General emulator diagnostic function, page 50

$D_1^{(i)}$ In the sequential design of computer experiments, a vector of function evaluations including for the first $(i-1)$ selected design points, and a proposed i^{th} design point, page 372

D_i Scalar random quantity for index i in Section 2.2.2, page 13

d_i Scalar decision parameter determined by the subscript i, page 10

$D_j(\mathbf{d}, t_i)$ Drilling costs in time interval Δt_i under field development strategy $\mathbf{d}$ for model j, page 87

D'_k Wave k vector of simulated expected NPV over the design without the ghost runs in Chapter 7, page 373

$D_\Gamma(\cdot)$ Individual correlated prediction errors emulator diagnostic, defined in Equation (2.4.50), page 51

$D_{\text{MD}}(\cdot)$ Mahalanobis distance emulator diagnostic, defined in Equation (2.4.48), page 51

$D_{ij}(\cdot)$ The j^{th} univariate standardised prediction error for an emulator of computer model output i, defined in Equation (2.4.47), page 50

$f(\cdot)$ Computer model or simulator which takes inputs $\mathbf{x} \in \mathbb{R}^{\tilde{D}}$, $\mathbf{d} \in \mathbb{R}^{D}$, or $(\mathbf{x}, \mathbf{d}) \in \mathbb{R}^{(\tilde{D}+D)}$, and returns an output vector $f(\mathbf{x})$, $f(\mathbf{d})$ or $f(\mathbf{x}, \mathbf{d}) \in \mathbb{R}^{q}$ respectively, page 8

$f(X^*)$ Vector of simulated computer model output for emulator diagnostics in Section 2.4.6, page 49

$f^*(\cdot)$ Emulator representation for a computer model output $f(\cdot)$, used in Sections 2.4.6 and 2.7, page 50

$f^{(i_k)}(\cdot)$ Output for computer model i_k, page 101

$f_i(\cdot)$ Univariate computer model or simulator output corresponding to component i, page 8

$f_T(t)$ Sampling distribution for sum of normalised parameters in targetted Bayesian design, page 167

$f_{i,\text{diff}}(\mathbf{d})$ Difference between $f_{i,\max}(\mathbf{d}) = d_{jk,t_i} \cdot \Delta t_i$ and $f_i(\mathbf{d})$, page 107

$f_{i,\max}$ Maximum over a design for a computer model output i, page 107

G Design matrix obtained by evaluating the mean functions over the design in Section 2.4.5, $G \in \mathbb{R}^{n \times p}$, page 39

$g(x, y, z)$ Function used in the derivation of the w-basis to define a level set, page 342

$g_{ij}(\mathbf{x}_{A_i})$ Known deterministic regression or global mean function of the active parameters in an emulator for computer model output i, see Equation (2.4.1), page 23

$I(\cdot)$ General implausibility measure, see Definition 2.8.1, page 69

$I_i(\cdot)$ History matching univariate implausibility measure, defined in Equation (2.8.4), page 70

$I_M(\cdot)$ History matching maximum implausibility measure, defined in Equation (2.8.5), page 71

$I_{2M}(\cdot)$ History matching second maximum implausibility measure, defined in Equation (2.8.6), page 71

$I_{3M}(\cdot)$ History matching third maximum implausibility measure, defined in Equation (2.8.7), page 71

$I_{multi}(\cdot)$ History matching multivariate implausibility measure, defined in Equation (2.8.8), page 71

$K_\nu(\cdot)$ Modified Bessel function of the second kind, $\nu > 0$, see Table 2.1, page 28

L Transformation matrix used in targetted Bayesian design, see Section 4.3.1, page 166

$L_{\mathcal{I}}$ Log-likelihood of data for a linear model containing predictors in the set $\mathcal{I}$ in Appendix A, page 431

M Jacobian matrix in $\mathbb{R}^{2\times 2}$ of partial derivatives for the transformation between the standard and h-bases in Section 6.3, defined in Equation (6.3.61), page 352

M Variance matrix of the input parameters used within covariance functions in Chapter 2, page 24

M_i Maximum of the i^{th} parameter range, page 258

m_i Minimum of the i^{th} parameter range, page 258

$m_i(\mathbf{x})$ Emulator mean function for computer model output i, defined in Equation (2.4.14), page 35

$m_{(p,\lambda)}(\mathcal{D})$ Average distance between design points in $\mathcal{D}$, defined in Equation (2.3.5), page 21

$m_{\mathrm{C}}(\mathbf{d}_j)$ Sample mean of the residuals over the utility function cost parameter uncertainty linear models for each simulated cost parameter scenario, defined in Equation (3.5.2), page 139

$m_{\mathrm{I},i}(\mathbf{d}_i)$ Sample mean for decision implementation error about $\mathbf{d}_i$, page 142

N Number of models in an ensemble, page 87

n Number of design points, page 20

N_{EGES} Number of models identified via the EGES techniques, page 101

n_c Number of sampled cost parameter vectors, page 131

N_t Number of discounting time intervals in NPV calculations, page 87

n_x Number of sampled model parameter vectors, page 131

n_{gen} Number of candidate designs, page 269

n_{prop} Number of design proposal points, page 268

n_{ref} Number of points in a reference design, page 268

N_{t_i} Number of NPV discounting time intervals in control interval ending at t_i, page 117

p Number of parameters in the emulator mean function, see Equation (2.4.1), page 23

p Real number with $p \geq 1$ which defines the p-norm or L^p-norm in Section 2.3.5, page 20

$P_j(\mathbf{d}, t_i)$ Platform costs in time interval Δt_i under field development strategy $\mathbf{d}$ for model j, page 87

$p_{\mathcal{I}}$ Number of linear model predictors in set $\mathcal{I}$ in Appendix A, page 431

Q Set of univariate computer model outputs used in history matching, page 70

q Computer model output dimension in Section 2.5, $q \in \mathbb{N}$, page 53

q-EI q-points Expected Improvement, page 129

$Q(\cdot, \cdot)$ Quadratic form used in non-stationary emulation, defined in Equation (6.3.63), page 353

Q_0 Initial collection of computer model outputs in history matching, page 73

Q_k Collection of computer model outputs considered at wave k of history matching, page 73

$Q_{j,op}(\mathbf{d}, t_i)$ Total oil production in time interval Δt_i under controls $\mathbf{d}$ for model j, page 87

$Q_{j,wi}(\mathbf{d}, t_i)$ Total water injection in time interval Δt_i under controls $\mathbf{d}$ for model j, page 87

$Q_{j,wp}(\mathbf{d}, t_i)$ Total water production in time interval Δt_i under controls $\mathbf{d}$ for model j, page 87

$r(\cdot, \cdot)$ Correlation function or kernel, page 27

r^2 Sum of squared first partial derivatives of $v(x, y)$ evaluated at $\mathbf{x}_0$ in Section 6.3, page 344

$R_j(\mathbf{d}, t_i)$ Function of the difference of all revenue and expenditure in the time interval Δt_i under controls $\mathbf{d}$ for model j, page 87

r_{op} Oil revenue per barrel, page 87

r_{wi} Water injection cost per barrel, page 87

r_{wp} Water production cost per barrel, page 87

s Dimension of observation data, $s \in \mathbb{N}$, page 61

s Sample standard deviation of a function, page 141

$s^2_{\mathrm{C}}(\mathbf{d}_j)$ Sample variance of the residuals over the utility function cost parameter uncertainty linear models for each simulated cost parameter scenario, defined in Equation (3.5.3), page 140

$s_{\mathrm{I},i}(\mathbf{d}_i)$ Sample standard deviation for decision implementation error about $\mathbf{d}_i$, page 142

t Sum of normalised parameters in Section 4.3.1, page 166

$U(\cdot)$ Utility function, page 130

$u_i(\mathbf{x}_{A_i})$ Weakly stationary stochastic process term as a function of the active parameters in an emulator for computer model output i, see Equation (2.4.1), page 23

$v(\cdot)$ Function for the embedding surface, page 334

$V_{\mathcal{I}}$ Arbitrary model selection criterion for a linear model containing predictors in the set $\mathcal{I}$ in Appendix A, page 432

W Matrix in $\mathbb{R}^{3\times 3}$ such that column i is $\mathbf{w}_i$, page 343

$w_i(\mathbf{x}_{A_i})$ Nugget term in an emulator for computer model output i, see Equation (2.4.1), page 23

X Design matrix of parameter vectors as rows in Section 2.4.5, $X \in \mathbb{R}^{n\times D}$, page 39

X^* New parameter design for emulator diagnostics in Section 2.4.6, $X^* \in \mathbb{R}^{m\times D}$, page 49

x_i Scalar model parameter determined by the subscript i in Chapters 2 to 6, page 10

$x_{A_i,k}$ The k^{th} active parameter for computer model output i, page 25

y Scalar or vector of the true values of quantities of interest for a physical system, page 58

ABC Approximate Bayesian Computation, page 80

AIC Akaike Information Criterion, defined in Equation (A.0.2), page 432

ARMA AutoRegressive-Moving-Average model, page 98

BFGS Broyden–Fletcher–Goldfarb–Shanno algorithm, page 99

BHP Bottom Hole Pressure, page 105

BHPSO Black Hole Particle Swarm Optimisation, page 97

BIC Bayesian Information Criterion, defined in Equation (A.0.3), page 432

CEL Cumulative Expected Loss, page 134

CMA Covariance-Matrix-Adaptation, page 94

CMA-EnOpt Covariance-Matrix-Adaptation Ensemble Optimiser, page 94

CVaR Conditional Value at Risk, page 96

CWG Controlled Wells Group, used in TNO OLYMPUS Well Control Optimisation Challenge in Chapter 4, page 153

DACE Design and Analysis of Computer Experiments, page 9

DIRECT Dividing Rectangles optimisation algorithm, page 127

EAGE European Association of Geoscientists and Engineers, page 84

ECMOR European Conference on the Mathematics of Oil Recovery, page 151

EGES Efficient Geological Ensemble Subsampling, see Section 3.2, page 100

EGO Efficient Global Optimisation algorithm, page 128

EI Expected Improvement Bayesian optimal design criterion, page 22

EM (Bayesian) Expectation Maximisation, page 129

EMC Evolutionary Monte Carlo, page 77

EnKF Ensemble Kalman Filter, page 93

EnKF-MDA Ensemble Kalman Filter with Multiple Data Assimilations, page 93

EnOpt Ensemble-based Optimisation scheme, page 94

EnRML Ensemble Randomised Maximum Likelihood, page 93

EQI Expected Quantile Improvement, page 129

ES Ensemble Smoother, page 93

ES-MDA Ensemble Smoother with Multiple Data Assimilations, page 93

EVPI Expected Value of Perfect Information, page 133

EVSI Expected Value of Sample Information, page 133

FFBS Forward Filtering, Backward Smoothing algorithm, page 129

FOIP Field Oil In Place, see Table 4.2, page 155

FOPT Field Oil Production Total, page 100

FWCT Field Water Cut, see Table 4.2, page 155

FWIT Field Water Injection Total, page 100

FWPT Field Water Production Total, page 100

GA Genetic Algorithm, page 95

GLS Generalised Least Squares, page 41

GP Gaussian Process, page 35

GPGO Gaussian Process Global Optimisation procedure, page 127

GPS Generalised Pattern Search, page 96

HEI Hierarchical Expected Improvement, page 128

HJDS Hooke-Jeeves Direct Search, page 96

HOPSPACK Hybrid Optimization Parallel Search Package, page 96

IAGO Informational Approach to Global Optimisation, page 127

IDEMC Implausibility Driven Evolutionary Monte Carlo, page 77

IF Implicit Filtering, page 129

IMSE Integrated Mean Squared Error, also known as Mean Squared Prediction Error (MSPE), page 22

INJ# or INJ.# Injector well number #, see Table 4.2, page 155

inj_x_yyyy_mm Target injection rate for well number "x" starting on the 1st of month "mm" in year "yyyy", page 154

IS-MOE Importance Sampled Mixture of Experts for Gaussian processes, page 38

ISAPP Integrated Systems Approach for Petroleum Production, page 85

Jade Emerson's oil reservoir model for a fictitious oil field, page 219

KL Kullback-Leibler Divergence, page 134

laGP Local approximate Gaussian Process, page 37

LCB Lower Confidence Bound Bayesian optimal design criterion, page 22

LSDF Line Search Derivative-Free, page 97

MCMC Markov Chain Monte Carlo, page 15

MLE Maximum Likelihood Estimation, page 33

MMSPE Maximum Mean Squared Prediction Error, page 22

MO Mean Optimisation, page 98

MOO Multi-Objective Optimisation, page 99

MRST MATLAB Reservoir Simulation Toolbox, page 96

MSPE Mean Squared Prediction Error, also known as Integrated Mean Squared Error (IMSE), page 22

MVO Mean-Variance Optimisation, page 98

NEI Noisy Expected Improvement, page 129

NPV Net Presented Value, page 86

OLS Ordinary Least Squares, page 41

OLYMPUS TNO developed oil reservoir model for a fictitious oil field, page 88

OPM Open Porous Media Flow simulator, page 155

OSS On-Site Surrogate Gaussian processes, page 37

PCA Principal Component Analysis, page 55

PES Predictive Entropy Search, page 129

PESC Predictive Entropy Search with Constraints, page 129

POI or PI Probability Of Improvement Bayesian optimal design criterion, page 22

PROD# or PROD.# Producer well number #, see Table 4.2, page 155

prod_x_yyyy_mm Target production rate for well number "x" starting on the 1st of month "mm" in year "yyyy", page 154

PSO Particle Swarm Optimisation, page 95

RBF Radial Basis Function method, page 127

REML Restricted Maximum Likelihood Estimation, page 33

RJMCMC Reversible-Jump Markov Chain Monte Carlo, page 80

SGV Sparse General Vecchia, page 330

SHELF Sheffield Elicitation Framework [72, 130], page 34

SNOBFIT Stable Noisy Optimisation by Branch and Fit algorithm, page 129

SNOPT Sparse Nonlinear OPTimiser, page 96

SPSA Simultaneous Perturbation Stochastic Approximation, page 94

SQP Sequential Quadratic Programming, page 95

ss-cc-StoSAG Singly-Smoothed Cross-Covariance Stochastic Simplex Approximate Gradient, page 96

StoSAG Stochastic Simplex Approximate Gradient, page 94

SUR Stepwise Uncertainty Reduction strategy, page 127

TENSE Torn Embedding Non-Stationary Emulation, page 324

TNO Netherlands Organisation for Applied Scientific Research, page 85

TSP Temporal Sure Preference principle, page 16

UCB Upper Confidence Bound Bayesian optimal design criterion, page 22

VaR Value at Risk, page 134

VFSA Very Fast Simulated Annealing, page 95

WCO Worst-Case Optimisation, page 98

WOPT Well Oil Production Total, page 101

WWIT Well Water Injection Total, page 101

WWPT Well Water Production Total, page 101

Appendix A

Linear Regression Stepwise Selection

In the construction of a Bayesian emulator, linear regression modelling along with stepwise selection methods and model selection criteria can be used as tools to aid the choice of active inputs and the known deterministic functions, $g_{ij}(\cdot)$, as discussed in Section 2.4.1. This appendix details the Mallows' C_p, AIC, BIC and adjusted R^2 model selection criteria, as well as both directions (also known as mixed), forwards and backwards linear model stepwise selection algorithms, as presented in [98, 183]. The notation $y_k,\ k = 1, \ldots, n$, is used to denote the (simulation) data sample to which the linear model is fitted with sample mean $\bar{y} = \frac{1}{n}\sum_{k=1}^{n} y_k$.

Given a full model with all p considered terms, often designated the full model; the model selection criteria Mallows' C_p, AIC, BIC and the adjusted R^2 for a candidate (or fitted) linear model containing the $p_{\mathcal{I}}$ terms in the subset $\mathcal{I}$ are defined in Equations (A.0.1) to (A.0.4) respectively. In each of the equations: n is the number of data points; s^2 is an estimate for the response variance for the full model; $\mathrm{RSS}_{\mathcal{I}} = \frac{1}{n-p_{\mathcal{I}}}\sum_{k=1}^{n}(y_k - \hat{y}_{\mathcal{I},k})^2$ denotes the residual sum of squares for the fitted model with $\hat{y}_{\mathcal{I},k}$ being the fitted linear model prediction for each $y_k,\ k = 1, \ldots, n$; and $L_{\mathcal{I}}$ is the log-likelihood for the fitted model.

$$C_{\mathcal{I}} = \frac{\mathrm{RSS}_{\mathcal{I}}}{s^2} + 2p_{\mathcal{I}} - n \tag{A.0.1}$$

$$AIC_{\mathcal{I}} = -2L_{\mathcal{I}} + 2p_{\mathcal{I}} \tag{A.0.2}$$

$$BIC_{\mathcal{I}} = -2L_{\mathcal{I}} + p_{\mathcal{I}}\log(n) \tag{A.0.3}$$

$$\left(\text{Adjusted } R^2\right)_{\mathcal{I}} = 1 - \frac{\text{RSS}_{\mathcal{I}}/(n - p_{\mathcal{I}} - 1)}{\sum_{k=1}^{n}(y_k - \bar{y})^2/(n-1)} \tag{A.0.4}$$

Each model selection criterion includes a trade-off between the goodness-of-fit of the fitted linear model to the data, versus a penalisation for large numbers of linear model predictors. In the case of the Mallows' C_p and adjusted R^2 criteria, the goodness-of-fit is captured by the residual sum of squares for which smaller values indicate a better model fit. For the AIC and BIC criteria this is encompassed by the log-likelihood which in turn also contains the residual sum of squares, in addition to an estimate for the linear model residual standard error. Similarly, a small value of the negative log-likelihood is preferred. The over-fitting penalisation comes via the inclusion of $p_{\mathcal{I}}$ in each of the expressions. Note that BIC is a variant on AIC obtained by substituting 2 for $\log(n)$ and thus constitutes a heavier over-fitting penalty for $n \geq 8$. This trade-off is achieved for small values of Mallows' C_p, AIC and BIC, whilst larger values are preferred for the adjusted R^2.

Let $V_{\mathcal{I}}$ denote a model selection criterion evaluated for the candidate linear model. Stepwise selection routines aim to minimise $V_{\mathcal{I}}$. We introduce an alternative adjusted R^2 criterion in Equation (A.0.5) for which minimisation is equivalent to maximisation of the adjusted R^2 as stated in Equation (A.0.4).

$$\left(\widetilde{\text{Adjusted } R^2}\right)_{\mathcal{I}} = \frac{\text{RSS}_{\mathcal{I}}}{(n - p_{\mathcal{I}} - 1)} \tag{A.0.5}$$

Stepwise selection methods consider candidate linear models in the scope between the model containing an intercept only, through to a full model. Let $\mathbf{x}$ denote the vector of all linear model predictors whilst $\mathbf{x}_{\mathcal{I}}$ be a sub-vector of predictors contained in $\mathcal{I}$. In the context of emulation, these are candidates for the deterministic function $g_{ij}(\mathbf{x}_{A_i})$ described in Section 2.4.1. Note that each sub-model must contain an intercept term, whilst before any interaction terms can be included, each of the corresponding main effect linear predictors must be included within the candidate linear model. The

both directions, forwards and backwards stepwise selection algorithms are presented in Algorithms 2 to 4 respectively where "compute" is used to implicitly mean fit the linear model and evaluate the model selection criterion. The notation $\mathcal{I} \cup \{j\}$ denotes the addition of a new predictor $j \notin \mathcal{I}$, and $\mathcal{I} \setminus \{j\}$ denotes the removal of a predictor j from $\mathcal{I}$.

Model selection is a challenging and potentially computationally demanding task. In order to fully evaluate all possible models using every combination of predictors from a collection would require the fitting of $2^{|\mathbf{x}|}$ models, a time consuming task, even for a modest number of predictors. Stepwise selection methods alleviate these difficulties by only considering a subset of these potential models based on the current state of the identified set of predictors at each iteration of the algorithm. Whilst this is computationally advantageous, this has the drawback that it is possible to fail to identify the best model with respect to the model selection criterion, although experience suggests that an adequately good model close to the optimal is often returned.

Algorithm 2: Both Directions Stepwise Selection

Result: Stepwise selected linear model containing predictors in $\mathcal{I}$ which approximately minimises $V_{\mathcal{I}}$.

Select an initial subset $\mathcal{I}_0$ of predictors from $\mathbf{x}$, including an intercept;
Fit a linear model and compute $V_0 = V_{\mathcal{I}_0}$;
Let $\mathcal{I} = \mathcal{I}_0$ and $V_{\mathcal{I}} = V_0$;
repeat
  Compute $V^f = \min_{j \notin \mathcal{I}} V_{\mathcal{I} \cup \{j\}}$ and set $j^f = \arg\min_{j \notin \mathcal{I}} V_{\mathcal{I} \cup \{j\}}$;
  Compute $V^b = \min_{j \in \mathcal{I}} V_{\mathcal{I} \setminus \{j\}}$ and set $j^b = \arg\min_{j \in \mathcal{I}} V_{\mathcal{I} \setminus \{j\}}$;
  Let $V' = \min\{V^f, V^b\}$;
  if $V' = V^f$ **then**
    Let $\mathcal{I}' = \mathcal{I} \cup \{j^f\}$;
  else
    Let $\mathcal{I}' = \mathcal{I} \setminus \{j^b\}$;
  end
  if $V' \leq V_{\mathcal{I}}$ **then**
    Update $V_{\mathcal{I}} = V'$;
    Update $\mathcal{I} = \mathcal{I}'$;
  end
until $V' > V_{\mathcal{I}}$;
return $\mathcal{I}$

Algorithm 3: Forward Selection

Result: Forward selected linear model containing predictors in $\mathcal{I}$ which approximately minimises $V_{\mathcal{I}}$.

Let the initial subset of predictors $\mathcal{I}_0$ contain an intercept only;
Fit a linear model and compute $V_0 = V_{\mathcal{I}_0}$;
Let $\mathcal{I} = \mathcal{I}_0$ and $V_{\mathcal{I}} = V_0$;
repeat
  Compute $V' = \min_{j \notin \mathcal{I}} V_{\mathcal{I} \cup \{j\}}$;
  Set $j' = \arg\min_{j \notin \mathcal{I}} V_{\mathcal{I} \cup \{j\}}$;
  Let $\mathcal{I}' = \mathcal{I} \cup \{j'\}$;
  if $V' \leq V_{\mathcal{I}}$ **then**
    Update $V_{\mathcal{I}} = V'$;
    Update $\mathcal{I} = \mathcal{I}'$;
  end
until $V' > V_{\mathcal{I}}$;
return $\mathcal{I}$

Algorithm 4: Backward Elimination

Result: Backward selected linear model containing predictors in $\mathcal{I}$ which approximately minimises $V_{\mathcal{I}}$.

Let the initial subset $\mathcal{I}_0$ contain all predictors in $\mathbf{x}$;
Fit a linear model and compute $V_0 = V_{\mathcal{I}_0}$;
Let $\mathcal{I} = \mathcal{I}_0$ and $V_{\mathcal{I}} = V_0$;
repeat
  Compute $V' = \min_{j \in \mathcal{I}} V_{\mathcal{I} \setminus \{j\}}$;
  Set $j' = \arg\min_{j \in \mathcal{I}} V_{\mathcal{I} \setminus \{j\}}$;
  Let $\mathcal{I}' = \mathcal{I} \setminus \{j'\}$;
  if $V' \leq V_{\mathcal{I}}$ **then**
    Update $V_{\mathcal{I}} = V'$;
    Update $\mathcal{I} = \mathcal{I}'$;
  end
until $V' > V_{\mathcal{I}}$;
return $\mathcal{I}$

Appendix B

Matrix Identities

In the derivation of the Bayes linear emulator adjustment formulae in Section 2.4.5, two matrix identities for the inverse of sums of matrices are utilised. These are the Sherman-Morrison-Woodbury formula in Lemma B.0.1 [118], and an alternative expression in Corollary B.0.3. In addition, a further identity is introduced in Lemma B.0.2 which is used in the proof of Corollary B.0.3.

Lemma B.0.1 (**Sherman-Morrison-Woodbury Formula**). *Let $A \in \mathbb{R}^{p\times p}$, $B \in \mathbb{R}^{p\times n}$, $C \in \mathbb{R}^{n\times n}$ and $D \in \mathbb{R}^{n\times p}$ be matrices for which A^{-1} and C^{-1} exist. Then the following identity holds:*

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \tag{B.0.1}$$

Proof. Equation (B.0.1) holds if the right-hand side is both a left- and a right-inverse of $A + BCD$. Define I_k to be the $k \times k$ identity matrix.

It is first shown that it is a right-inverse.

$$\begin{aligned}
&(A + BCD)(A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}) \\
&= I_p - B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \\
&\quad + BCDA^{-1} - BCDA^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \\
&= I_p - B\{(I_n + CDA^{-1}B)(C^{-1} + DA^{-1}B)^{-1} - C\}DA^{-1} \\
&= I_p - B\{C(C^{-1} + DA^{-1}B)(C^{-1} + DA^{-1}B)^{-1} - C\}DA^{-1}
\end{aligned}$$

$$= I_p - B\{C - C\}DA^{-1}$$
$$= I_p$$

Next, it is shown to be a left-inverse.

$$\begin{aligned}
&(A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1})(A + BCD) \\
&= I_p + A^{-1}BCD - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}D \\
&\quad - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}BCD \\
&= I_p - A^{-1}B\{(C^{-1} + DA^{-1}B)^{-1}(I_n + DA^{-1}BC) - C\}D \\
&= I_p - A^{-1}B\{C - C\}D \\
&= I_p
\end{aligned}$$

Hence the Sherman-Morrison-Woodbury formula holds. □

Lemma B.0.2. *Let $U \in \mathbb{R}^{p\times n}$ and $V \in \mathbb{R}^{n\times p}$. Assuming that $(I_p + UV)^{-1}$ and $(I_n + VU)^{-1}$ exist, then the following identity holds:*

$$(I_p + UV)^{-1}U = U(I_n + VU)^{-1} \tag{B.0.2}$$

Proof. Start from the equality:

$$U + UVU = U(I_n + VU) = (I_p + UV)^{-1}U$$

Applying $(I_n + VU)^{-1}$ on the right and $(I_p + UV)^{-1}$ on the left:

$$\iff (I_p + UV)^{-1}U = U(I_n + VU)^{-1}$$

This proves the identity in Equation (B.0.2). □

Corollary B.0.3. *Let $A \in \mathbb{R}^{p\times p}$, $B \in \mathbb{R}^{p\times n}$, $C \in \mathbb{R}^{n\times n}$ and $D \in \mathbb{R}^{n\times p}$ be matrices for which A^{-1} and C^{-1} exist. Then the following identity holds:*

$$A^{-1}B(DA^{-1}B + C^{-1})^{-1} = (BCD + A)^{-1}BC \tag{B.0.3}$$

Proof.

$$A^{-1}B(DA^{-1}B + C^{-1})^{-1} = A^{-1}B(CDA^{-1}B + I_n)^{-1}C$$

Applying Lemma B.0.2 with $U = B$ and $V = CDA^{-1}$.

$$\begin{aligned} &= A^{-1}(BCDA^{-1} + I_p)^{-1}BC \\ &= (BCDA^{-1} + I_p)BC \\ &= (BCD + A)^{-1}BC \end{aligned}$$

Hence this proves Corollary B.0.3. □

Bibliography

[1] Ioannis Andrianakis and Peter G. Challenor. 'The effect of the nugget on Gaussian process emulators of computer models'. In: *Computational Statistics & Data Analysis* 56 (12 2012), pp. 4215–4228. doi:10.1016/j.csda.2012.04.020.

[2] Ioannis Andrianakis, Ian R. Vernon, Nicky McCreesh, Trevelyan J. McKinley, Jeremy E. Oakley, Rebecca N. Nsubuga, Michael Goldstein and Richard G. White. 'Bayesian History Matching of Complex Infectious Disease Models Using Emulation: A Tutorial and a Case Study on HIV in Uganda'. In: *PLOS Computational Biology* 11.1 (2015). ISSN: 1553-7358. doi:10.1371/journal.pcbi.1003968.

[3] Ioannis Andrianakis, Ian R. Vernon, Nicky McCreesh, Trevelyan J. McKinley, Jeremy E. Oakley, Rebecca N. Nsubuga, Michael Goldstein and Richard G. White. 'History matching of a complex epidemiological model of human immunodeficiency virus transmission by using variance emulation'. In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 66.4 (2017), pp. 717–740. ISSN: 0035-9254. doi:10.1111/rssc.12198.

[4] Netherlands Organisation for Applied Scientific Research (TNO). *OLYMPUS Oil Reservoir Model Input Decks*. OLYMPUS Oil Reservoir Model Input Decks and documentation for use in the ISAPP TNO

OLYMPUS Field Development Optimisation Challenge.
https://www.isapp2.com/downloads/olympus-reservoir-model.pdf.

[5] Joseph Aylett-Bullock, Carolina Cuesta-Lazaro, Arnau Quera-Bofarull, Miguel Icaza-Lizaola, Aidan Sedgewick, Henry Truong, Aoife Curran, Edward Elliott, Tristan Caulfield, Kevin Fong, Ian Vernon, Julian Williams, Richard Bower and Frank Krauss. 'JUNE: open-source individual-based epidemiology simulation'. In: *Royal Society Open Science* 8.210506 (7 2021). ISSN: 2054-5703.
doi:10.1098/rsos.210506.

[6] François Bachoc. 'Cross Validation and Maximum Likelihood estimations of hyper-parameters of Gaussian processes with model misspecification'. In: *Computational Statistics and Data Analysis* 66 (2013), pp. 55–69. ISSN: 0167-9473.
doi:10.1016/j.csda.2013.03.016.

[7] Evan Baker, Pierre Barbillon, Arindam Fadikar, Robert B. Gramacy, Radu Herbei, David Higdon, Jiangeng Huang, Leah R. Johnson, Pulong Ma, Anirban Mondal, Bianica Pires, Jerome Sacks and Vadim Sokolov. 'Analyzing Stochastic Computer Models: A Review with Opportunities'. In: *Statistical Science* 37.1 (2022), pp. 64–89.
doi:10.1214/21-STS822.

[8] Leonardo S. Bastos and Anthony O'Hagan. 'Diagnostics for Gaussian Process Emulators'. In: *Technometrics* 51.4 (2009): *Special Issue on Computer Modeling*, pp. 425–438.
doi:10.1198/TECH.2009.08019.

[9] Maria J Bayarri, James O Berger, Rui Paulo, Jerry Sacks, John A Cafeo, James Cavendish, Chin-Hsu Lin and Jian Tu. 'A Framework for Validation of Computer Models'. In: *Technometrics* 49.2 (2007), pp. 138–154.
doi:10.1198/004017007000000092.

[10] Maria J. Bayarri, James O. Berger, German Molina, Nagui M. Rouphail and Jerome Sacks. 'Assessing Uncertainties in Traffic Simulation: A Key Component in Model Calibration and Validation'. In: *Transportation Research Record* 1876.1 (2004), pp. 32–40.
doi:10.3141/1876-04.

[11] Maria J. Bayarri, Eliza S. Calder, Simon Lunagomez, E. Bruce Pitman, James O. Berger, Keith Dalbey, Abani K. Patra, Elaine T. Spiller and Robert L. Wolpert. 'Using Statistical and Computer Models to Quantify Volcanic Hazards'. In: *Technometrics* 51.4 (2009), pp. 402–413. ISSN: 1537-2723.
doi:10.1198/TECH.2009.08018.

[12] Mathias C. Bellout, David Echeverría Ciaurri, Louis J. Durlofsky, Bjarne Foss and Jon Kleppe. 'Joint optimization of oil well placement and controls'. In: *Computational Geosciences* 16.4 (2012), pp. 1061–1079. ISSN: 1573-1499.
doi:10.1007/s10596-012-9303-5.

[13] James Berger. 'The Case for Objective Bayesian Analysis'. In: *Bayesian Analysis* 1.3 (2006), pp. 385–402.
doi:10.1214/06-BA115.

[14] Jose M. Bernardo and Adrian F. M. Smith. *Bayesian Theory*. Wiley Series in Probability and Statistics. Wiley, 1994. ISBN: 978-0-471-92416-6.
doi:10.1002/9780470316870.

[15] Eric Brochu, Vlad M. Cora and Nando de Freitas. 'A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning'. In: *Computational Research Repository* abs/1012.2599 (2010).
arXiv:1012.2599 [cs.LG].

[16] Steve Brooks, Andrew Gelman, Galin Jones and Xiao-Li Meng, eds. *Handbook of Markov Chain Monte Carlo*. 1st ed. CRC Press, 2011. ISBN: 9781420079418.
doi:10.1201/b10905.

[17] Jenný Brynjarsdóttir and Anthony O'Hagan. 'Learning about physical parameters: the importance of model discrepancy'. In: *Inverse Problems* 30.11 (2014).
doi:10.1088/0266-5611/30/11/114007.

[18] Adam D. Bull. 'Convergence rates of efficient global optimization algorithms'. In: *Journal of Machine Learning Research* 12.88 (2011), pp. 2879–2904.
http://dblp.uni-trier.de/db/journals/jmlr/jmlr12.html#Bull11.

[19] Camila C. S. Caiado and Michael Goldstein. 'Bayesian uncertainty analysis for complex physical systems modelled by computer simulators with applications to tipping points'. In: *Communications in Nonlinear Science and Numerical Simulation* 26.1 (2015), pp. 123–136. ISSN: 1007-5704.
doi:10.1016/j.cnsns.2015.02.006.

[20] Eugene T. Y. Chang, Mark Strong and Richard H. Clayton. 'Bayesian Sensitivity Analysis of a Cardiac Cell Model Using a Gaussian Process Emulator'. In: *PLOS ONE* 10.8 (2015).
doi:10.1371/journal.pone.0137004.

[21] Yan Chen, Dean S. Oliver and Dongxiao Zhang. 'Efficient Ensemble-Based Closed-Loop Production Optimisation'. In: *SPE Journal* 14.4 (2009), pp. 634–645. ISSN: 1086-055X.
doi:10.2118/112873-PA.

[22] Zhehui Chen, Simon Mak and Chi-Fang Jeff Wu. 'A hierarchical expected improvement method for Bayesian optimization'. 2021.
arXiv:1911.07285v2 [stat.ME]. In preparation.

[23] Stefano Conti and Anthony O'Hagan. 'Bayesian emulation of complex multi-output and dynamic computer models'. In: *Journal of Statistical Planning and Inference* 140 (3 2010), pp. 640–651. doi:0.1016/j.jspi.2009.08.006.

[24] Roger M. Cooke. *Experts in Uncertainty: Opinion and Subjective Probability in Science*. Oxford University Press, 1991. ISBN: 9780195064650. doi:10.1086/293541.

[25] Sam Coveney and Richard H. Clayton. 'Sensitivity and Uncertainty Analysis of Two Human Atrial Cardiac Cell Models Using Gaussian Process Emulators'. In: *Frontiers in Physiology* 11.364 (2020). ISSN: 1664-042X. doi:10.3389/fphys.2020.00364.

[26] Peter S. Craig, Michael Goldstein, Jonathan C Rougier and Allan H. Seheult. 'Bayesian Forecasting for Complex Systems Using Computer Simulators'. In: *Journal of the American Statistical Association* 96.454 (2001), pp. 717–729. ISSN: 1537-274X. doi:10.1198/016214501753168370.

[27] Peter S. Craig, Michael Goldstein, Allan H. Seheult and James A. Smith. 'Bayes linear strategies for matching hydrocarbon reservoir history'. In: *Bayesian Statistics 5*. Ed. by J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith. Proceedings of the Valencia International Meeting 5. Clarendon Press, 1996, pp. 69–95. ISBN: 978-0198523567.

[28] Peter S. Craig, Michael Goldstein, Allan H. Seheult and James A. Smith. 'Constructing partial prior specifications for models of complex physical systems'. In: *Journal of the Royal Statistical Society: Series D (The Statistician)* 47 (1 1998), pp. 37–53. ISSN: 1467-9884. doi:10.1111/1467-9884.00115.

[29] Peter S. Craig, Michael Goldstein, Allan H. Seheult and James A. Smith. 'Pressure matching for hydrocarbon reservoirs: a case study in the use of

Bayes linear strategies for large computer experiments'. In: *Case Studies in Bayesian Statistics. Volume III*. Ed. by Constatine Gatsonis, James S. Hodges, Robert E. Klass, Robert E. McCulloch, Peter Rossi and Nozer D. Singpurwalla. 1st ed. Vol. 121. Lecture Notes in Statistics. New York: Springer-Verlag, 1997, pp. 37–93. ISBN: 978-0-387-94990-1. doi:10.1007/978-1-4612-2290-3.

[30] Jonathan Cumming and Michael Goldstein. 'Bayes Linear Uncertainty Analysis for Oil Reservoirs Based on Multisale Computer Experiments'. In: *The Oxford Handbook of Applied Bayesian Analysis*. Ed. by Anthony O'Hagan and Mike West. Oxford University Press, 2010. Chap. 10, pp. 241–270. ISBN: 9780199548903.

[31] Jonathan A. Cumming and Michael Goldstein. 'Bayesian decision analysis for complex physical systems via embedded computer models'. 2019. In preparation.

[32] Jonathan A. Cumming and Michael Goldstein. 'Small Sample Bayesian Designs for Complex High-Dimensional Models Based on Information Gained Using Fast Approximations'. In: *Technometrics* 51.4 (2009), pp. 377–388. ISSN: 1537-2723. doi:10.1198/TECH.2009.08015.

[33] Jonathan A. Cumming and David A. Wooff. 'Dimension reduction via principal variables'. In: *Computational Statistics & Data Analysis* 52 (1 2007), pp. 550–565. ISSN: 0167-9473. doi:10.1016/j.csda.2007.02.012.

[34] Carla Currin, Toby Mitchell, Max Morris and Don Ylvisaker. 'Bayesian Prediction of Deterministic Functions, with Applications to the Design and Analysis of Computer Experiments'. In: *Journal of the American Statistical Association* 86.416 (1991), pp. 953–963. doi:10.1080/01621459.1991.10475138.

[35] Morris H. DeGroot. *Optimal Statistical Decisions*. McGraw-Hill, 1970. ISBN: 978-0070162426.

[36] Sy T. Do and Albert C. Reynolds. 'Theoretical connections between optimization algorithms based on an approximate gradient'. In: *Computational Geosciences* 17.6 (2013), pp. 959–973. doi:10.1007/s10596-013-9368-9.

[37] Matthew M. Dunlop, Mark A. Girolami, Andrew M. Stewart and Aretha L. Teckentrup. 'How Deep Are Deep Gaussian Processes?' In: *Journal of Machine Learning Research* 19.54 (2018), pp. 1–46.

[38] Rock Flow Dynamics. *tNavigator*. Version 21.4. 2021. https://rfdyn.com.

[39] *EAGE/TNO Workshop on OLYMPUS Field Development Optimization* (Barcelona, Spain). European Association of Geoscientists, Engineers (EAGE) and the Netherlands Organisation for Applied Scientific Research (TNO). EAGE Publications, 2018. ISBN: 978-94-6282-261-0.

[40] Tamsin L. Edwards, Mark A. Brandon, Gael Durand, Neil R. Edwards, Nicholas R. Golledge, Philip B. Holden, Osabel J. Nias, Antony J. Payne, Catherine Ritz and Andreas Wernecke. 'Revisiting Antarctic ice loss due to marine ice-cliff instability'. In: *Nature* 566 (2019), pp. 58–64. doi:10.1038/s41586-019-0901-4.

[41] Alexandre A. Emerick. 'Analysis of the performance of ensemble-based assimilation of production and seismic data'. In: *Journal of Petroleum Science and Engineering* 139 (2016), pp. 219–239. doi:10.1016/j.petrol.2016.01.029.

[42] Alexandre A. Emerick and Albert C. Reynolds. 'Ensemble smoother with multiple data assimilation'. In: *Computers & Geosciences* 55 (2013), pp. 3–15. doi:10.1016/j.cageo.2012.03.011.

[43] Alexandre A. Emerick and Albert C. Reynolds. 'History Matching Time-lapse Seismic Data Using the Ensemble Kalman Filter with Multiple Data Assimilations'. In: *Computational Geosciences* 16 (3 2012), pp. 639–659. ISSN: 1573-1499. doi:10.1007/s10596-012-9275-5.

[44] Emerson. *Tempest ENABLE*. Comp. software. Version 2021.2. Emerson, 2021. https://www.pdgm.com/products/tempest/tempest-enable/#.

[45] Malcolm Farrow and Michael Goldstein. 'Almost-Pareto decision sets in imprecise utility hierarchies'. In: *Journal of Statistical Theory and Practice* 3 (1 2009), pp. 137–155. doi:10.1080/15598608.2009.10411916.

[46] Malcolm Farrow and Michael Goldstein. 'Sensitivity of decisions with imprecise utility trade-off parameters using boundary linear utility'. In: *International Journal of Approximate Reasoning* 51 (9 2010), pp. 1100–1113. doi:10.1016/j.ijar.2010.08.002.

[47] Malcolm Farrow and Michael Goldstein. 'Trade-off sensitive experimental design: a multicriterion, decision theoretic, Bayes linear approach'. In: *Journal of Statistical Planning and Inference* 136 (2 2006), pp. 498–526. doi:10.1016/j.jspi.2004.07.008.

[48] Bruno De Finetti. *Theory of Probability*. Vol. 1. John Wiley & Sons Ltd, 1974. ISBN: 978-0471201410.

[49] Bruno De Finetti. *Theory of Probability*. Vol. 2. John Wiley & Sons Ltd, 1975. ISBN: 978-0471201427.

[50] Rahul Rahul-Mark Fonseca, Bailian Chen, Jan Dirk Jansen and Albert Reynolds. 'A Stochastic Simplex Approximate Gradient (StoSAG) for optimization under uncertainty'. In: *International Journal for Numerical*

Methods in Engineering 109 (13 2017), pp. 1756–1776. ISSN: 0029-5981. doi:10.1002/nme.5342.

[51] Rahul Rahul-Mark Fonseca, Olwijn Leeuwenburgh, Paul M. J. Van den Hof and Jan Dirk Jansen. 'Improving the Ensemble-Optimization Method Through Covariance-Matrix Adaptation'. In: *SPE Journal* 20.01 (2014), pp. 155–168. ISSN: 1086-055X. https://onepetro.org/SJ/article-pdf/20/01/155/2245263/spe-163657-pa.pdf doi:10.2118/163657-PA.

[52] Rahul Rahul-Mark Fonseca, Siavash S. Kahrobaei, L.J.T. van Gastel, Olwijn Leeuwenburgh and Jan Dirk Jansen. 'Quantification of the Impact of Ensemble Size on the Quality of an Ensemble Gradient Using Principles of Hypothesis Testing'. In: SPE Reservoir Simulation Conference (23rd Feb. 2015–25th Feb. 2015). D031S011R003. 2015. https://onepetro.org/spersc/proceedings-pdf/15RSS/3-15RSS/D031S011R003/1464858/spe-173236-ms.pdf doi:SPE-173236-MS.

[53] Thomas E. Fricker, Jeremy E. Oakley and Nathan M. Urban. 'Multivariate Gaussian Process Emulators With Nonseparable Covariance Structures'. In: *Technometrics* 55.1 (2013), pp. 47–56. doi:0.1080/00401706.2012.715835.

[54] Donald Geman and Bruno Jedynak. *An active testing model for tracking roads in satellite images*. Research Report RR-2757. Project SYNTIM. Institut National de Recherches en Informatique et Automatique (INRIA), 1995. https://hal.inria.fr/inria-00073935.

[55] Charles Geyer. 'Introduction to Markov Chain Monte Carlo'. In: *Handbook of Markov Chain Monte Carlo*. Ed. by Steve Brooks, Andrew Gelman,

Galin Jones and Xiao-Li Meng. 1st ed. CRC Press, 2011. Chap. 1, pp. 3–48. ISBN: 9781420079418.
doi:10.1201/b10905.

[56] David Ginsbourger, Rodolphe Le Riche and Laurent Carraro. 'A Multi-points Criterion for Deterministic Parallel Global Optimization based on Gaussian Processes'. Version 1. In: (2008).
https://hal.archives-ouvertes.fr/hal-00260579/document.

[57] Michael Goldstein. 'Bayes Linear Analysis'. In: *Encyclopedia of Statistical Sciences*. Ed. by Samuel Kotz, Campbell B. Read, Narayanaswamy Balakrishnan, Brani Vidakovic and Norman L. Johnson. 2nd ed. Vol. 3. Wiley, 2006, pp. 29–34. ISBN: 9780471150442.
doi:10.1002/0471667196.ess0986.pub2.

[58] Michael Goldstein. 'Bayes Linear Analysis for Complex Physical Systems Modeled by Computer Simulators'. In: *IFIP Advances in Information and Communication Technology* 377 (2011), pp. 78–94.
https://hal.inria.fr/hal-01518669
doi:10.1007/978-3-642-32677-6_6.

[59] Michael Goldstein. 'External Bayesian analysis for computer simulators'. In: *Bayesian Statistics 9*. Ed. by J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith and M. West. Proceedings of the Valencia International Meeting 9. Oxford University Press, 2011, pp. 201–228. ISBN: 978-0199694587.
doi:10.1093/acprof:oso/9780199694587.001.0001.

[60] Michael Goldstein. 'Observables and models: exchangeability and the inductive argument'. In: *Bayesian Theory and Applications*. Ed. by Paul Damien, Petros Dellaportas, Nicholas G. Polson and David A. Stephens. Oxford University Press, 2013. Chap. 1. ISBN: 978-0199695607.

[61] Michael Goldstein. 'Prior Inferences for Posterior Judgements'. In: *Structures and Norms in Science Volume Two of the Tenth International Congress of Logic, Methodology and Philosophy of Science*. Ed. by Maria Luisa Dalla Chiara, Kees Doets, Daniele Mundici and Johan Van Benthem. Vol. 260. Dordrecht: Springer, 1997, pp. 55–71. ISBN: 978-90-481-4787-8. doi:10.1007/978-94-017-0538-7_4.

[62] Michael Goldstein. 'Revising Exchangeable Beliefs: Subjectivist foundations for the inductive argument'. In: *Aspects of uncertainty: a tribute to D.V. Lindley*. Ed. by Peter R. Freeman and Adrian F. M. Smith. Wiley, 1994, pp. 201–222. ISBN: 0471943479.

[63] Michael Goldstein. 'Subjective Bayesian Analysis: Principles and Practice'. In: *Bayesian Analysis* 1.3 (2006), pp. 403–420. ISSN: 1936-0975. doi:10.1214/06-BA116.

[64] Michael Goldstein. *The Bayes linear approach to uncertainty analysis for complex computer models*. Unpublished. Presentation. The University of Warwick: Durham University, 2nd Feb. 2015. https://www2.warwick.ac.uk/fac/sci/maths/research/events/2014-15/nonsymposium/uccm/equipmgslides.pdf.

[65] Michael Goldstein and Nathan Huntley. 'Bayes linear emulation, history matching and forecasting for complex computer simulators'. In: *The Handbook of Uncertainty Quantification*. Ed. by Roger Ghanem, David Higdon and Houman Owhadi. 1st ed. Springer, 2017. Chap. 2, pp. 9–32. ISBN: 978-3-319-12384-4.

[66] Michael Goldstein and Jonathan Rougier. 'Bayes Linear Calibrated Prediction for Complex Systems'. In: *Journal of the American Statistical Association* 101.475 (2006), pp. 1132–1143. doi:10.1198/016214506000000203.

[67] Michael Goldstein and Jonathan Rougier. 'Probabilistic Formulations for Transferring Inferences from Mathematical Models to Physical Systems'. In: *SIAM Journal on Scientific Computing* 26.2 (2004), pp. 467–487. doi:10.1137/S106482750342670X.

[68] Michael Goldstein and Jonathan Rougier. 'Reified Bayesian modelling and inference for physical systems'. In: *Journal of Statistical Planning and Inference* 139 (3 2009), pp. 1221–1239. ISSN: 0378-3758. doi:10.1016/j.jspi.2008.07.019.

[69] Michael Goldstein, Allan Seheult and Ian Vernon. 'Assessing Model Adequacy'. In: *Environmental Modelling: Finding Simplicity in Complexity*. Ed. by John Wainwright and Mark Mulligan. 2nd ed. Wiley-Blackwell, 2013. Chap. 26, pp. 435–449. ISBN: 978-0470749111.

[70] Michael Goldstein and David Wooff. *Bayes Linear Statistics: Theory and Methods*. Wiley Series in Probability and Statistics. Wiley, 2007. ISBN: 978-0-470-01562-9. doi:10.1002/9780470065662.

[71] Michael Goldstein and David A. Wooff. 'Adjusting Exchangeable Beliefs'. In: *Biometrika* 85 (1 1998), pp. 39–54.

[72] John Paul Gosling. 'SHELF: The Sheffield Elicitation Framework'. In: *Elicitation: The Science and Art of Structuring Judgement*. Ed. by Luis C. Dias, Alec Morton and John Quigley. Vol. 261. International Series in Operations Research & Management Science. Cham: Springer International Publishing, 2018. Chap. 4, pp. 61–93. ISBN: 978-3-319-65051-7. doi:10.1007/978-3-319-65052-4_4.

[73] Robert B. Gramacy and Daniel W. Apley. 'Local Gaussian Process Approximation for Large Computer Experiments'. In: *Journal of Computational and Graphical Statistics* 24.2 (2015), pp. 561–578. doi:10.1080/10618600.2014.914442.

[74] Robert B. Gramacy and Herbet K. H. Lee. 'Bayesian Treed Gaussian Process Models with an Application to Computer Modeling'. In: *Journal of the American Statistical Association* 103.483 (2008), pp. 1119–1130. ISSN: 01621459.
doi:10.2307/27640148.

[75] Robert B. Gramacy and Herbet K. H. Lee. 'Cases for the nugget in modeling computer experiments'. In: *Statistics and Computing* 22.3 (2012), pp. 713–722.
doi:10.1007/s11222-010-9224-x.

[76] Robert B. Gramacy, Genetha A. Gray, Sébastien Le Digabel, Herbet K. H. Lee, Pritam Ranjan, Garth Wells and Stefan M. Wild. 'Modeling an Augmented Lagrangian for Blackbox Constrained Optimization'. In: *Technometrics* 58.1 (2016), pp. 1–11.
doi:10.1080/00401706.2015.1014065.

[77] Peter J. Green. 'Reversible jump Markov chain Monte Carlo computation and Bayesian model determination'. In: *Biometrika* 82.4 (1995), pp. 711–732. ISSN: 0006-3444.
doi:10.1093/biomet/82.4.711.

[78] Mengyang Gu and James O. Berger. 'Parallel Partial Gaussian Process Emulation for Computer Models with Massive Output'. In: *The Annals of Applied Statistics* 10.3 (2016), pp. 1317–1347. ISSN: 1932-6157.
doi:10.1214/16-AOAS934.

[79] Hans-Martin Gutmann. 'A Radial Basis Function Method for Global Optimization'. In: *Journal of Global Optimisation* 19.3 (2001), pp. 201–227.
doi:10.1023/A:1011255519438.

[80] David Hartley and Simon French. 'A Bayesian method for calibration and aggregation of expert judgement'. In: *International Journal of Approximate*

Reasoning 130 (2021), pp. 192–225. ISSN: 0888-613X.
doi:10.1016/j.ijar.2020.12.007.

[81] Matthew J. Heaton, Abhirup Datta, Andrew O. Finley, Reinhard Furrer, Joseph Guinness, Rajarshi Guhaniyogi, Florian Gerber, Robert B. Gramacy, Dorit Hammerling, Matthias Katzfuss, Finn Lindgren, Douglas W. Nychka, Furong Sun and Andrew Zammit-Mangion. 'A Case Study Competition Among Methods for Analyzing Large Spatial Data'. In: *Journal of Agricultural, Biological and Environmental Statistics* 24.3 (2019), pp. 398–425.
doi:10.1007/s13253-018-00348-w.

[82] Katrin Heitmann, David Higdon, Martin White, Salman Habib, Brian J. Williams, Earl Lawrence and Christian Wagner. 'The Coyote Universe II: Cosmological Models and Precision Emulation of the Nonlinear Matter Power Spectrum'. In: *The Astrophysical Journal* 705.1 (2009), pp. 156–174.
doi:10.1088/0004-637x/705/1/156.

[83] Philipp Hennig and Christian J. Schuler. 'Entropy Search for Information-Efficient Global Optimization'. In: *Journal of Machine Learning Research* 13 (2012), pp. 1809–1937.
https://jmlr.csail.mit.edu/papers/volume13/hennig12a/.

[84] José Miguel Hernández-Lobato, Matthew W. Hoffman and Zoubin Ghahramani. 'Predictive Entropy Search for Efficient Global Optimization of Black-Box Functions'. In: Advances in Neural Information Processing Systems 27 (8th Dec. 2014–13th Dec. 2014). Ed. by editor.
arXiv:1406.2541 [stat.ML].

[85] José Miguel Hernández-Lobato, James Requeima, Edward O. Pyzer-Knapp and Alán Aspuru-Guzik. 'Parallel and Distributed Thompson Sampling for Large-Scale Accelerated Exploration of Chemical Space'. In: 34th

International Conference on Machine Learning, ICML (6th Aug. 2017–11th Aug. 2017). Vol. 70, pp. 1470–1479. arXiv:1706.01825 [stat.ML].

[86] José Miguel Hernández-Lobato, Michael A. Gelbart, Matthew W. Hoffman, Ryan P. Adams and Zoubin Ghahramani. 'Predictive Entropy Search for Bayesian Optimization with Unknown Constraints'. In: 32nd International Conference on Machine Learning, ICML (6th July 2015–11th July 2015). Vol. 37, pp. 1699–1707. arXiv:1502.05312 [stat.ML].

[87] Dave Higdon, James Gattiker, Biran Williams and Maria Rightley. 'Computer Model Calibration Using High-Dimensional Output'. In: *Journal of the American Statistical Association* 103.482 (2008), pp. 570–583. doi:10.1198/016214507000000888.

[88] Philip B. Holden, Neil R. Edwards, James Hensman and Richard D. Wilkinson. 'ABC for Climate: Dealing with Expensive Simulators'. In: *Handbook of Approximate Bayesian Computation*. Ed. by Scott A. Sisson, Yanan Fan and Mark A. Beaumont. 1st ed. CRC Press, 2018. Chap. 19, pp. 569–596. ISBN: 9781315117195. doi:10.1201/9781315117195.

[89] Leanna House, Michael Goldstein and Ian Vernon. 'Second-Order Exchangeable Functions: An Approach to Quantify Uncertainty from System Conditions of Computer Models'. Durham University, 2018. Unpublished.

[90] Jiangeng Huang, Robert B. Gramacy, Michaël Binois and Mirko Libraschi. 'On-site surrogates for large-scale calibration'. In: *Applied Stochastic Models in Business and Industry* 36.5 (2020). doi:10.1002/asmb.2523.

[91] Waltraud Huyer and Arnold Neumaier. 'SNOBFIT – Stable Noisy Optimization by Branch and Fit'. In: *ACM Transactions on Mathematical*

Software 35.9 (2 2008), pp. 1–25.
doi:10.1145/1377612.1377613.

[92] Marco A. Iglesias, Kody J.H. Law and Andrew M. Stuart. 'Ensemble Methods for Inverse Problems'. In: *Inverse Problems* 29.4, 045001 (2013).
doi:10.1088/0266-5611/29/4/045001.

[93] *Integrated Systems Approach for Petroleum Production (ISAPP)*. ISAPP research program website. Netherlands Organisation for Applied Scientific Research (TNO).
https://www.isapp2.com/home.html.

[94] Kaoru Irie and Mike West. 'Bayesian Emulation for Multi Step Optimization in Decision Problems'. In: *Bayesian Analysis* 14.1 (2019), pp. 137–160. ISSN: 1931-6690.
doi:10.1214/18-BA1105.

[95] Samuel E. Jackson and Ian Vernon. 'Efficient Emulation of Computer Models Utilising Multiple Known Boundaries of Differing Dimensions'. In: *Bayesian Analysis* 18.1 (2023), pp. 165–191.
doi:10.1214/22-BA1304.

[96] Samuel E. Jackson and David C. Woods. 'Bayes Linear Emulation of Simulator Networks'. 2021.
arXiv:1910.08003v2 [stat.ME]. In preparation.

[97] A. Jahandideh and B. Jafarpour. 'Stochastic Oilfield Optimization For Hedging Against Uncertain Future Development Plans'. In: ECMOR XVI - 16th European Conference on the Mathematics of Oil Recovery (3rd Sept. 2018–6th Sept. 2018). EAGE. 2018.
www.earthdoc.org/publication/publicationdetails/?publication=93971
doi:10.3997/2214-4609.201802225.

[98] Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani. *An Introduction to Statistical Learning with Applications in R*. Springer Series in Statistics. Springer, 2013. ISBN: 978-1-4614-7137-0. doi:10.1007/978-1-4614-7138-7.

[99] Norman L. Johnson, Samuel Kotz and N. Balakrishnan. *Continuous Univariate Distributions*. 2nd ed. Vol. 1. Wiley Series in Probability and Statistics. New York: Wiley, 1994. ISBN: 9780471584957.

[100] Ross H. Johnstone, Eugene T.Y. Chang, Rémi Bardenet, Teun P. de Boer, David J. Gavaghan, Pras Pathmanathan, Richard H. Clayton and Gary R. Mirams. 'Uncertainty and variability in models of the cardiac action potential: Can we build trustworthy models?' In: *Journal of Molecular and Cellular Cardiology* 96 (2016), pp. 49–62. doi:10.1016/j.yjmcc.2015.11.018.

[101] Donald R. Jones, Cary D. Pertunen and Bruce E. Stuckman. 'Lipschitzian optimization without the Lipschitz constant'. In: *Journal of Optimization Theory and Applications* 79.1 (1993), pp. 157–181. doi:10.1007/BF00941892.

[102] Donald R. Jones, Matthias Schonlau and William J. Welch. 'Efficient Global Optimization of Expensive Black-Box Functions'. In: *Journal of Global Optimisation* 13.4 (1998), pp. 455–492. doi:10.1023/A:1008306431147.

[103] Matthias Katzfuss, Jonathan R. Stroud and Christopher K. Wikle. 'Understanding the Ensemble Kalman Filter'. In: *The American Statistician* 70.4 (2016), pp. 350–357. doi:10.1080/00031305.2016.1141709.

[104] Cari G. Kaufman, Derek Bingham, Salman Habib, Katrin Heitmann and Joshua A. Frieman. 'Efficient emulators of computer experiments using compactly supported correlation functions, with an application to

cosmology'. In: *The Annals of Applied Statistics* 5.4 (2011), pp. 2470–2492. doi:10.1214/11-AOAS489.

[105] C. T. Kelly. *Implicit Filtering*. Society for Industrial and Applied Mathematics, 2011. ISBN: 978-1-61197-189-7. doi:10.1137/1.9781611971903.

[106] Jack C. Kennedy, Daniel A. Henderson and Kevin J. Wilson. 'Multilevel Emulation for Stochastic Computer Models with Application to Large Offshore Wind Farms'. 2021. arXiv:2003.08921v5 [stat.ME]. In preparation.

[107] Marc C. Kennedy and Anthony O'Hagan. 'Bayesian calibration of computer models'. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.3 (2001), pp. 425–464. doi:10.1111/1467-9868.00294.

[108] Marc C. Kennedy and Anthony O'Hagan. 'Predicting the Output from a Complex Computer Code When Fast Approximations Are Available'. In: *Biometrika* 87.1 (2000), pp. 1–13. ISSN: 00063444. doi:10.1093/biomet/87.1.1.

[109] Cheng Seong Khor, Ali Elkamel and Nilay Shah. 'Optimization methods for petroleum fields development and production systems: a review'. In: *Optimization and Engineering* 18.4 (2017), pp. 907–941. ISSN: 1573-2924. doi:10.1007/s11081-017-9365-2.

[110] Louise Kimpton, Peter Challenor and Daniel B. Williamson. 'Classification of Computer Models with Labelled Outputs'. In Submission. 2020. arXiv:2002.00010v1 [stat.ME].

[111] Ksenia N. Kyzyurova, James O. Berger and Robert L. Wolpert. 'Coupling Computer Models through Linking Their Statistical Emulators'. In: *SIAM/ASA Journal on Uncertainty Quantification* 6.3 (2018),

pp. 1151–1171.
doi:10.1137/17M1157702.

[112] Peter M. Lee. *Bayesian Statistics: An Introduction*. 4th ed. Wiley, 2012. 496 pp. ISBN: 978-1-118-33257-3.

[113] Benjamin Letham, Brian Karrer, Guilherme Ottoni and Eytan Bakshy. 'Constrained Bayesian Optimization with Noisy Experiments'. In: *Bayesian Analysis* 14.2 (2019), pp. 495–519.
doi:doi:10.1214/18-BA1110.

[114] Faming Liang and Wing Hung Wong. 'Real-Parameter Evolutionary Monte Carlo With Applications to Bayesian Mixture Models'. In: *Journal of the American Statistical Association* 96.454 (2001), pp. 653–666.
doi:10.1198/016214501753168325.

[115] Haitao Liu, Yew-Soon Ong, Xiaobo Shen and Jianfei Cai. 'When Gaussian Process Meets Big Data: A Review of Scalable GPs'. In: *IEEE Transactions on Neural Networks and Learning Systems* 31.11 (2020), pp. 4405–4423.
https://ieeexplore.ieee.org/document/8951257
doi:10.1109/TNNLS.2019.2957109.

[116] Jason L. Loeppky, Jerome Sacks and William J. Welch. 'Choosing the Sample Size of a Computer Experiment: A Practical Guide'. In: *Technometrics* 51.4 (2009), pp. 366–376.
doi:10.1198/TECH.2009.08040.

[117] Jan Mandel. *A Brief Tutorial on the Ensemble Kalman Filter*. Provided by the SAO/NASA Astrophysics Data System. 2009.
arXiv:0901.3725 [physics.ao-ph].
http://adsabs.harvard.edu/abs/2009arXiv0901.3725M.

[118] Kanti V. Mardia, John T. Kent and John M. Bibby. *Multivariate Analysis*. Ed. by Zygmunt Wilhelm Birnbaum and Eugene Lukacs. Probability and

Mathematical Statistics. Academic Press Limited, 1979. 536 pp. ISBN: 9780124712522.

[119] Bertil Matérn. *Spatial Variation*. 2nd ed. Vol. 36. Lecture Notes in Statistics. Originally published (1960) in Meddelanden från Statens Skogsforskningsinstitut, 49, No.5., Almänna Förlaget, Stockholm. Berlin: Springer-Verlag, 1986. ISBN: 978-1-4615-7892-5. doi:10.1007/978-1-4615-7892-5.

[120] George P. McCabe. 'Principal Variables'. In: *Technometrics* 126.2 (1984), pp. 137–144. doi:10.1080/00401706.1984.10487939.

[121] Michael D. McKay, Richard J. Beckman and William J. Conover. 'A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code'. In: *Technometrics* 21.2 (1979), pp. 239–245. ISSN: 00401706. doi:10.2307/1268522.

[122] Gary R. Mirams, Pras Pathmanathan, Richard A. Gray, Peter Challenor and Richard H. Clayton. 'Uncertainty and variability in computational and mathematical models of cardiac physiology'. In: *The Journal of Physiology* 594.23 (2016), pp. 6833–6847. doi:10.1113/JP271671.

[123] Hossein Mohammadi, Peter Challenor, Marc Goodfellow and Daniel Williamson. *Emulating computer models with step-discontinuous outputs using Gaussian processes*. 2020. arXiv:1903.02071v4 [stat.ME]. In preparation.

[124] German Molina, Maria J. Bayarri and James O. Berger. 'Statistical Inverse Analysis for a Network Microsimulator'. In: *Technometrics* 47.4 (2005), pp. 388–398. ISSN: 0040-1706. doi:10.1198/004017005000000409.

[125] Max D. Morris, Toby J. Mitchell and Donald Ylvisaker. 'Bayesian Design and Analysis of Computer Experiments: Use of Derivatives in Surface Prediction'. In: *Technometrics* 35.3 (1993), pp. 243–255. doi:10.1080/00401706.1993.10485320.

[126] Jeremy Oakley. 'Eliciting Gaussian process priors for complex computer codes'. In: *Journal of the Royal Statistical Society: Series D (The Statistician)* 51 (1 2002), pp. 81–97. doi:10.1111/1467-9884.00300.

[127] Jeremy E. Oakley. 'Decision-Theoretic Sensitivity Analysis for Complex Computer Models'. In: *Technometrics* 51.2 (2009), pp. 121–129. ISSN: 1537-2723. doi:10.1198/TECH.2009.0014.

[128] Jeremy E. Oakley and Anthony O'Hagan. 'Probabilistic sensitivity analysis of complex models: a Bayesian approach'. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 66.3 (2004), pp. 751–769. doi:10.1111/j.1467-9868.2004.05304.x.

[129] Anthony O'Hagan. 'Bayesian analysis of computer code outputs: A tutorial'. In: *Reliability Engineering and System Safety* 91.10 (2006), pp. 1290–1300. ISSN: 0951-8320. doi:10.1016/j.ress.2005.11.025.

[130] Anthony O'Hagan. *SHELF: The Sheffield Elicitation Framework*. Version 4.0. University of Sheffield. 2019. http://www.tonyohagan.co.uk/shelf/SHELF4.html.

[131] Anthony O'Hagan. 'Some Bayesian Numerical Analysis'. In: *Bayesian Statistics 4*. Ed. by J. M. Bernardo, J. O. Berger, Morris H. DeGroot, A. P. Dawid and A. F. M. Smith. Proceedings of the Valencia International Meeting 4. Clarendon Press, 1992, pp. 345–363. ISBN: 978-0198522669.

[132] Anthony O'Hagan, Marc C. Kennedy and Jeremy E. Oakley. 'Uncertainty Analysis and other Inference Tools for Complex Computer Codes (with discussion)'. In: *Bayesian Statistics 6*. Ed. by J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith. Proceedings of the Valencia International Meeting 6. Clarendon Press, 1998-08-13, pp. 503–524. ISBN: 978-0198523567.

[133] Anthony O'Hagan and Jeremy E. Oakley. 'Probability is perfect, but we can't elicit it perfectly'. In: *Reliability Engineering and System Safety* 85.1 (2004), pp. 239–248. ISSN: 0951-8320.
doi:10.1016/j.ress.2004.03.014.

[134] Michael A. Osborne, Roman Garnett and Stephen J. Roberts. *Gaussian Processes for Global Optimisation*. Department of Engineering Science, University of Oxford. 2008.
http://www.robots.ox.ac.uk/~mosb/public/pdf/115/Osborne%20et%20al.%20-%202009%20-%20Gaussian%20processes%20for%20global%20optimization.pdf.

[135] Jonathan Owen, Ian Vernon and Richard Hammersley. 'A Bayesian statistical approach to decision support for TNO OLYMPUS well control optimisation under uncertainty'. In: ECMOR XVII – 17th European Conference on the Mathematics of Oil Recovery (14th Sept. 2020–17th Sept. 2020). Vol. 2020. 1. European Association of Geoscientists & Engineers (EAGE). 2020.
doi:10.3997/2214-4609.202035109.

[136] Christopher J. Paciorek and Mark J. Schervish. 'Spatial modelling using a new class of nonstationary covariance functions'. In: *Environmetrics* 17.5 (2006), pp. 483–506. ISSN: 1180-4009.
doi:10.1002/env.785.

[137] Roger Penrose. 'A generalized inverse for matrices'. In: *Mathematical Proceedings of the Cambridge Philosophical Society* 51 (3 1955), pp. 406–413. doi:10.1017/S0305004100030401.

[138] Victor Picheny, David Ginsbourger and Yann Richet. 'Noisy Expected Improvement and On-Line Computation Time Allocation for the Optimization of Simulators with Tunable Fidelity'. In: 2nd International Conference on Engineering Optimization, EngOpt 2010 (6th Sept. 2010–9th Sept. 2010). Ed. by editor. https://hal.archives-ouvertes.fr/hal-00489321.

[139] Victor Picheny, David Ginsbourger, Yann Richet and Gregory Caplin. 'Quantile-Based Optimization of Noisy Computer Experiments With Tunable Precision'. In: *Technometrics* 55.1 (2013), pp. 2–13. ISSN: 1537-2723. doi:10.1080/00401706.2012.707580.

[140] Victor Picheny, David Ginsbourger, Yann Richet and Gregory Caplin. 'Rejoinder'. In: *Technometrics* 55.1 (2013), pp. 31–36. ISSN: 1537-2723. doi:10.1080/00401706.2012.759155.

[141] Christopher A. Pope, John Paul Gosling, Stuart Barber, Jill S. Johnson, Takanobu Yamaguchi, Graham Feingold and Paul G. Blackwell. 'Gaussian Process Modeling of Heterogeneity and Discontinuities Using Voronoi Tessellations'. In: *Technometrics* 63.1 (2021), pp. 53–63. doi:10.1080/00401706.2019.1692696.

[142] Friedrich Pukelsheim. 'The Three Sigma Rule'. In: *The American Statistician* 48.2 (1994), pp. 88–91. doi:10.2307/2684253.

[143] Zhiguang Qian and Chi-Fang Jeff Wu. 'Bayesian Hierarchical Modeling for Integrating Low-Accuracy and High-Accuracy Experiments'. In: *Technometrics* 50.2 (2008), pp. 192–204. ISSN: 00401706. doi:10.1198/004017008000000082.

[144] Zhiguang Qian, Carolyn Connor Seepersad, V. Roshan Joseph, Janet K. Allen and Chi-Fang Jeff Wu. 'Building Surrogate Models Based on Detailed and Approximate Simulations'. In: *ASME Journal of Mechanical Design* 128.4 (2006), pp. 668–677. doi:10.1115/1.2179459.

[145] Chao Qin, Diego Klabjan and Daniel Russo. 'Improving the expected improvement algorithm'. In: Advances in Neural Information Processing Systems 30 (4th Dec. 2017–9th Dec. 2017). Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna Wallach, Rob Fergus, S. Vishwanathan and Roman Garnett, pp. 5381–5391. arXiv:1705.10033 [cs.LG].

[146] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006. ISBN: 0-262-18253-X. https://gaussianprocess.org/gpml/chapters/RW.pdf.

[147] Matthew Revie and Tim J. Bedford and. 'Evaluation of elicitation methods to quantify Bayes linear models'. In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 224.4 (2010), pp. 322–332. ISSN: 1748-006X. doi:10.1243/1748006XJRR304.

[148] Mark D. Risser and Daniel Turek. 'Bayesian inference for high-dimensional nonstationary Gaussian processes'. In: *Journal of Statistical Computation and Simulation* 90.16 (2020), pp. 2902–2928. doi:10.1080/00949655.2020.1792472.

[149] Jonathan Rougier. 'Efficient Emulators for Multivariate Deterministic Functions'. In: *Journal of Computational and Graphical Statistics* 17.4 (2008), pp. 827–843. doi:10.1198/106186008X384032.

[150] Jonathan Rougier, Michael Goldstein and Leanna House. 'Second-Order Exchangeability Analysis for Multimodel Ensembles'. In: *Journal of the American Statistical Association* 108.503 (2013), pp. 852–863. doi:10.1080/01621459.2013.802963.

[151] Jonathan Rougier, Serge Guillas, Astrid Maute and Arthur D. Richmond. 'Expert Knowledge and Multivariate Emulation: The Thermosphere–Ionosphere Electrodynamics General Circulation Model (TIE-GCM)'. In: *Technometrics* 51.4 (2009), pp. 414–424. doi:10.1198/TECH.2009.07123.

[152] Jerome Sacks, William J. Welch, Toby J. Mitchell and Henry P. Wynn. 'Design and Analysis of Computer Experiments'. In: *Statistical Science* 4.4 (1989). doi:doi:10.1214/ss/1177012413.

[153] Thomas J. Santner, Brian J. Williams and William I. Notz. *The Design and Analysis of Computer Experiments*. 1st ed. Springer Series in Statistics. New York: Springer-Verlag, 2003. XII, 84. ISBN: 978-1-4419-2992-1. doi:10.1007/978-1-4757-3799-8.

[154] Mark J. Schervish. *Theory of Statistics*. 1st ed. Springer Series in Statistics. Springer-Verlag New York, 1995. XVI, 716. ISBN: 978-1-4612-8708-7. doi:10.1007/978-1-4612-4250-5.

[155] Claudia Schillings and Andrew M. Stuart. 'Analysis of the Ensemble Kalman Filter for Inverse Problems'. In: *SIAM Journal on Numerical Analysis* 55.3 (2016), pp. 1264–1290. doi:10.1137/16M105959X.

[156] Warren Scott, Peter Fraizer and Warren Powell. 'The Correlated Knowledge Gradient for Simulation Optimization of Continuous Parameters using Gaussian Process Regression'. In: *SIAM Journal of Optimization* 21 (3

2011), pp. 996–1026.
doi:10.1137/100801275.

[157] M. Mohsin Siraj, Paul M.J. Van den Hof and Jan Dirk Jansen. 'Handling Geological and Economic Uncertainties in Balancing Short-Term and Long-Term Objectives in Waterflooding Optimization'. In: *SPE Journal* 22 (4 2017), pp. 1313–1325.
doi:10.2118/185954-PA.

[158] M. Mohsin Siraj, Paul M.J. Van den Hof and Jan Dirk Jansen. 'Risk management in oil reservoir water-flooding under economic uncertainty'. In: *2015 54th IEEE Conference on Decision and Control (CDC)* (15th Dec. 2015–18th Dec. 2015). 2015, pp. 7542–7547.
doi:10.1109/CDC.2015.7403410.

[159] Edward Snelson and Zoubin Ghahramani. 'Sparse Gaussian Processes using Pseudo-inputs'. In: Advances in Neural Information Processing Systems 18 (5th Dec. 2005–8th Dec. 2005). 2005, pp. 1257–1264.
http://www.gatsby.ucl.ac.uk/~snelson/SPGP_up.pdf.

[160] Jasper Snoek, Hugo Larochelle and Ryan P. Adams. 'Practical Bayesian Optimization of Machine Learning Algorithms'. In: Advances in Neural Information Processing Systems 25 (3rd Dec. 2012–8th Dec. 2012). Vol. 2, pp. 2951–2959.
arXiv:1206.2944v2 [stat.ML].

[161] Statoil, SINTEF, IRIS, Ceetron Solutions and HPC-Simulation-Software. *The Open Porous Media Initiative*. Version 2018.04. 2018.
https://opm-project.org/.

[162] Gemma Stephenson and Peter Challenor. *Adjoints without Adjoints: emulating model derivatives*. Technical Report. MUCM, 2010.
www.mucm.ac.uk/Pages/Dissemination/TechnicalReports.html.

[163] Mark Strong, Jeremy E. Oakley and Jim Chilcott. 'Managing structural uncertainty in health economic decision models: a discrepancy approach'. In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 61.1 (2012), pp. 25–45.
doi:10.1111/j.1467-9876.2011.01014.x.

[164] Taha Taha, Paul Ward, Gavin Peacock, John Heritage, Rafel Bordas, Usman Aslam, Steve Walsh, Richard Hammersley and Emmanuel Gringarten. 'History Matching Using 4D Seismic in an Integrated Multi-Disciplinary Automated Workflow'. In: SPE Reservoir Characterisation and Simulation Conference and Exhibition (17th Sept. 2019–19th Sept. 2019). Society of Petroleum Engineers. 2019. ISBN: 978-1-61399-669-0.
https://onepetro.org/SPERCSC/proceedings-pdf/19RCSC/3-19RCSC/D031S022R001/1177266/spe-196680-ms.pdf
doi:10.2118/196680-MS.

[165] Shuesi Tanaka, Zhenzhen Wang, Kaveh Dehghani, Jincong He, Baskar Velusamy and Xian-Huan Wen. 'Large Scale Field Development Optimization Using High Performance Parallel Simulation and Cloud Computing Technology'. In: Annual Technical Conference and Exhibition (Dallas, Texas, USA, 24th Sept. 2018–26th Sept. 2018). Chevron Energy Technology Company. Society of Petroleum Engineers, 2018. ISBN: 978-1-61399-572-3.
doi:10.2118/191728-MS.

[166] Michalis K. Titsias. 'Variational Learning of Inducing Variables in Sparse Gaussian Processes'. In: Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics (16th Apr. 2009–18th Apr. 2009). Ed. by David van Dyk and Max Welling. Vol. 5. Proceedings of Machine Learning Research. PMLR, 2009, pp. 567–574.
http://proceedings.mlr.press/v5/titsias09a.html.

[167] Robin Tokmakian and Peter Challenor. 'Near surface ocean temperature uncertainty related to initial condition uncertainty'. In: *Climate Dynamics* 53.7 (2019), pp. 4683–4700. ISSN: 1432-0894. doi:10.1007/s00382-019-04872-4.

[168] Giang T. Tran, Kevin I. C. Oliver, András Sóbester, David J. J. Toal, Philip B. Holden, Robert Marsh, Peter Challenor and Neil R. Edwards. 'Building a traceable climate model hierarchy with multi-level emulators'. In: *Advances in Statistical Climatology, Meteorology and Oceanography* 2.1 (2016), pp. 17–37. ISSN: 2364-3587. https://ascmo.copernicus.org/articles/2/17/2016/ doi:10.5194/ascmo-2-17-2016.

[169] Giang T. Tran, Kevin I. C. Oliver, Philip B. Holden, Neil R. Edwards, András Sóbester and Peter Challenor. 'Multi-level emulation of complex climate model responses to boundary forcing data'. In: *Climate Dynamics* 52.3 (2019), pp. 1505–1531. ISSN: 1432-0894. doi:10.1007/s00382-018-4205-4.

[170] Ian Vernon, Michael Goldstein and Richard Bower. 'Galaxy Formation: Bayesian History Matching for the Observable Universe'. In: *Statistical Science* 29.1 (2014), pp. 81–90.

[171] Ian Vernon, Michael Goldstein and Richard G. Bower. 'Galaxy Formation: a Bayesian Uncertainty Analysis'. In: *Bayesian Analysis* 5.4 (2010), pp. 619–670.

[172] Ian Vernon and John Paul Gosling. 'A Bayesian Computer Model Analysis of Robust Bayesian Analyses'. In: *Bayesian Analysis* (2022). doi:10.1214/22-BA1340. Accepted.

[173] Ian Vernon, Samuel E. Jackson and Jonathan A. Cumming. 'Known Boundary Emulation of Complex Computer Models'. In: *SIAM/ASA Journal on Uncertainty Quantification* 7.3 (2019), pp. 838–876. ISSN:

2166-2525.

doi:10.1137/18M1164457.

[174] Ian Vernon, Jonathan Owen and Jonathan Carter. 'Bayesian Emulation for Computer Models with Multiple Partial Discontinuities'. 2022. arXiv:2210.10468v1 [stat.ME]. In Submission.

[175] Ian Vernon and Allan Seheult. *Roxar Statistical Support: Report 14 Making Decisions Under Uncertainty*. Consultancy Report. Durham University, 2018. Unpublished.

[176] Ian Vernon and Allan Seheult. *Roxar Statistical Support: Report 16 Bayesian Optimisation using Emulators/Proxies*. Consultancy Report. Durham University, 2019. Unpublished.

[177] Ian Vernon, Jonathan Owen, Joseph Aylett-Bullock, Carolina Cuesta-Lazaro, Jonathan Frawley, Arnau Quera-Bofarull, Aidan Sedgewick, Difu Shi, Henry Truong, Mark Turner, Joseph Walker, Tristan Caulfield, Kevin Fong and Frank Krauss. 'Bayesian Emulation and History Matching of JUNE'. In: *Philosphical Transactions of the Royal Society A* 380 (2233 2022). ISSN: 1471-2962.

doi:10.1098/rsta.2022.0039.

[178] Ian Vernon, Junli Liu, Michael Goldstein, James Rowe, Jen Topping and Keith Lindsey. 'Bayesian uncertainty analysis for complex systems biology models: emulation, global parameter searches and evaluation of gene functions'. In: *BMC systems biology* 12 (1 2018). ISSN: 1752-0509.

doi:10.1186/s12918-017-0484-3.

[179] Julien Villemonteix, Emmanuel Vazquez and Eric Walter. 'An informational approach to the global optimization of expensive-to-evaluate functions'. In: *Journal of Global Optimization* 44.509 (2009), pp. 373–389.

doi:10.1007/s10898-008-9354-2.

[180] Victoria Volodina and Daniel Williamson. 'Diagnostics-Driven Nonstationary Emulators Using Kernel Mixtures'. In: *SIAM/ASA Journal on Uncertainty Quantification* 8.1 (2020), pp. 1–26. ISSN: 2166-2525. doi:10.1137/19M124438X.

[181] D. F. Vysochanskij and Y. I. Petunin. 'Justification of the 3-σ Rule for Unimodal Distribution'. In: *Theory of Probability and Mathematical Statistics* 21 (1980), pp. 25–36.

[182] James Watson and Chris Holmes. 'Approximate Models and Robust Decisions'. In: *Statistical Science* 31.4 (2016), pp. 465–489. doi:10.1214/16-STS592.

[183] Sanford Weisberg. *Applied Linear Regression*. 3rd ed. Wiley Series in Probability and Statistics. Wiley, 2005. ISBN: 978-0471663799. doi:10.1002/0471704091.

[184] Richard D. Wilkinson. 'Approximate Bayesian computation (ABC) gives exact results under the assumption of model error'. In: *Statistical Applications in Genetics and Molecular Biology* 12.2 (2013), pp. 129–141. doi:10.1515/sagmb-2013-0010.

[185] Daniel Williamson. 'Policy making using computer simulators for complex physical systems; Bayesian decision support for the development of adaptive strategies'. Durham theses. Durham, 2010. http://etheses.dur.ac.uk/348/.

[186] Daniel Williamson and Ian Vernon. *Efficient uniform designs for multi-wave computer experiments*. 2013. arXiv:309.3520v1 [stat.ME]. In preparation.

[187] Daniel Williamson, Michael Goldstein, Lesley Allison, Adam Blaker, Peter Challenor, Laura Jackson and Kuniko Yamazaki. 'History matching for exploring and reducing climate model parameter space using observations and a large perturbed physics ensemble'. In: *Climate Dynamics*

41.7 (2013), pp. 1703–1729. ISSN: 1432-0894.
doi:10.1007/s00382-013-1896-4.

[188] Michael Minyi Zhang and Sinead A. Williamson. 'Embarrassingly Parallel Inference for Gaussian Processes'. In: *Journal of Machine Learning Research* 20.169 (2019), pp. 1–26.
http://jmlr.org/papers/v20/18-374.html.