WEB-BASED STRATEGIES IN THE MANUFACTURING INDUSTRY

by

Luis Alexis Velásquez

University of Durham

School of Engineering

A thesis submitted to the University of Durham for the degree of Doctor of Philosophy (**Ph.D.**).

NOVEMBER 2000

To my mother

There is not distance for your unconditional love. I have always known, even without you telling me, how constant I am in your prayers. During these last years, with only the telephone to bridge the ocean between us, your "bless you son" has thrilled and warmed me. I have been able to picture the love in your eyes and feel your arms around your child again. Your image has been the best incentive during my harder days. Soon, I will be able to look at you while I remember whom I missed, and who loves me, so much.

Perenne manantial de amor que no sabe de distancias. Blanca sencillez, que cierras tus ojos y levantas tu mano en alto para bendecirme, cuando ves mi espalda alejarse y crees que no te observo. Siempre he sabido, aún sin decírmelo, cuán presente he estado en tus oraciones. No imaginas lo cálido y placentero que ha sido durante estos últimos años escuchar tu "Dios te bendiga hijo" a través del teléfono. Tus manos y brazos, ya maltratados por el trabajo y el paso del tiempo, nunca han dejado de regalarme su más cálido abrazo. Tu imagen ha sido un aliciente durante mis días mas difíciles. Gracias por hacerme sentir tan especial, gracias por ser tan especial.

To Clara Giselle and María Andreína

Wishing you a future better than mine, where you can be active protagonists, as part of a new and promising generation.

Deseándoles un futuro mejor que el mío, en el cual sean activas protagonistas, como parte de una nueva y prometedora generación.

DECLARATION

I hereby declare that this thesis is the result of work undertaken by myself. It has not been submitted for any other degree in this or any other University. All sources of referenced information have been duly acknowledged.

Luis Alexis Velásquez
Web-based Strategies in the Manufacturing Industry
Ph.D. 2000.

ABSTRACT

The explosive growth of Internet-based architectures is allowing an efficient access to information resources over geographically dispersed areas. This fact is exerting a major influence on current manufacturing practices. Business activities involving customers, partners, employees and suppliers are being rapidly and efficiently integrated through networked information management environments. Therefore, efforts are required to take advantage of distributed infrastructures that can satisfy information integration and collaborative work strategies in corporate environments. In this research, Internet-based distributed solutions focused on the manufacturing industry are proposed. Three different systems have been developed for the tooling sector, specifically for the company Seco Tools UK Ltd (industrial collaborator). They are summarised as follows.

SELTOOL is a Web-based open tool selection system involving the analysis of technical criteria to establish appropriate selection of inserts, toolholders and cutting data for turning, threading and grooving operations. It has been oriented to world-wide Seco customers. ***SELTOOL*** provides an interactive and crossed-way of searching for tooling parameters, rather than conventional representation schemes provided by catalogues. Mechanisms were developed to filter, convert and migrate data from different formats to the database (SQL-based) used by ***SELTOOL***.

TTS (*Tool Trials System*) is a Web-based system developed by the author and two other researchers to support Seco sales engineers and technical staff, who would perform tooling trials in geographically dispersed machining centres and benefit from sharing data and results generated by these tests. Through ***TTS*** tooling engineers (authorised users) can submit and retrieve highly specific technical tooling data for both milling and turning operations. Moreover, it is possible for tooling engineers to avoid the execution of new tool trials knowing the results of trials carried out in physically distant places, when another engineer had previously executed these trials. The system incorporates encrypted security features suitable for restricted use on the World Wide Web.

An urgent need exists for tools to make sense of raw data, extracting useful knowledge from increasingly large collections of data now being constructed and made available from networked information environments. This explosive growth in the availability of information is overwhelming the capabilities of traditional information management systems, to provide efficient ways of detecting anomalies and significant patterns in large sets of data. Inexorably, the tooling industry is generating valuable experimental data. It is a potential and unexplored sector regarding the application of knowledge capturing systems. Hence, to address this issue, a knowledge discovery system called ***DISKOVER*** was developed.

DISKOVER is an integrated Java-application consisting of five data mining modules, able to be operated through the Internet. *Kluster* and *Q-Fast* are two of these modules, entirely developed by the author. *Fuzzy-K* has been developed by the author in collaboration with another research student in the group at Durham. The final two modules (*R-Set* and *MQG*) have been developed by another member of the Durham group. To develop *Kluster,* a complete clustering methodology was proposed. *Kluster* is a clustering application able to combine the analysis of quantitative as well as categorical data (conceptual clustering) to establish data classification processes. This module incorporates two original contributions. Specifically, consistent indicators to measure the *quality of the final classification* and application of *optimisation methods* to the final groups obtained. *Kluster* provides the possibility, to users, of introducing case-studies to generate cutting parameters for particular input requirements. *Fuzzy-K* is an application having the advantages of hierarchical clustering, while applying fuzzy membership functions to support the generation of similarity measures. The implementation of fuzzy membership functions helped to optimise the grouping of categorical data containing missing or imprecise values. As the tooling database is accessed through the Internet, which is a relatively slow access platform, it was decided to rely on faster information retrieval mechanisms. *Q-fast* is an SQL-based exploratory data analysis (EDA) application, implemented for this purpose.

ACKNOWLEDGEMENTS

I wish to express my gratitude to my supervisor, Professor Phil Mars. I am so appreciative for his guidance, advice and patience during this research. Beyond his professional qualifications, he is an honourable man, an example that I will always remember.

I am deeply grateful to my wife. Thanks for your love. Thanks for always being there, in my sombre as well as my happier moments. As a colleague, her contribution to this research has been invaluable.

I want to express my sincere thanks to Robin Paterson and May. Their home was also my home while I was in Durham. It will be hard to say good bye to you when I have to leave England, your friendship will remain forever. Robin's help with the English grammar made the production of this thesis markedly more readable.

I would like to thank Dr. Kelvin Revere for his active participation and contributions to this research.

I would like to acknowledge the Secretarial and Technical staff of the School of Engineering, especially Alan Swann. Thanks also to the members of the Design and Manufacturing Research Group. Particularly, to John Darlington, Alima Laguda, and Neil Stutchbury. Their support is strongly appreciated.

I want to thank the financial support provided by the following Venezuelan institutions: Comisión Nacional de Investigaciones Científicas y Tecnológicas (**CONICIT**), Fundación Gran Mariscal de Ayacucho (**Fundayacucho**) and Universidad Nacional Experimental de Guayana (**UNEG**). In particular, thanks to Professor José Tarazona, his concern and support have been inestimable.

Finally, I wish to acknowledge the co-operation given by the company, Seco Tools (UK) Ltd, as the industrial collaborator. Especially, to the General Manager Mr. Hans Åhman and the Technical Manager, Mr. John Tindall.

TABLE OF CONTENTS

DECLARATION i

ABSTRACT..... ii

ACKNOWLEDGEMENTS iv

TABLE OF CONTENTS v

LIST OF FIGURES x

LIST OF TABLES xiii

LIST OF ABBREVIATIONS xiv

CHAPTER 1 : INTRODUCTION 1

1.1 MANUFACTURING AND TOOLING INDUSTRY 1

1.1.1 General Manufacturing 2

1.1.2 Tooling Industry 4

1.1.2.1 Tool Selection 4

1.1.2.2 Cutting Data 4

1.1.2.3 Tool Life Prediction 5

1.2 AGILE MANUFACTURING 6

1.3 GLOBAL AND DISTRIBUTED MANUFACTURING 8

1.4 VIRTUAL MANUFACTURING 11

1.5 FROM RAW DATA STORAGE TO A WEB-BASED KNOWLEDGE DISCOVERY APPROACH 15

1.5.1 KDD Technology in the Tooling Industry 18

1.6 SUMMARY 19

1.7 AIMS AND OBJECTIVES OF THIS RESEARCH 19

1.8 THESIS STRUCTURE 20

CHAPTER 2 : REMOTE INFORMATION EXCHANGE IN CORPORATE ENVIRONMENTS 22

2.1 INTRODUCTION 22

2.2 REQUIRED FUNCTIONALITY 22

2.3 THE ROLE OF THE INTERNET 26

2.4 DATABASE ACCESS AND CONNECTIVITY ISSUES 27

2.4.1 Connectivity and Application Programming Interfaces (APIs) 28

2.4.2 Connectivity, Database Access and Drivers 30

2.4.3 Common Gateway Interface versus Java-based Approach 31
2.5 AN APPROACH TO SUPPORT REMOTE INFORMATION EXCHANGE IN CORPORATE ENVIRONMENTS 33
2.5.1 *ARIEX* Architecture 35
2.6 SUMMARY 40
CHAPTER 3 : DEPLOYING INTERNET-BASED SOLUTIONS 41
3.1 INTRODUCTION 41
3.2 SYSTEM 1: TOOL TRIALS SYSTEM (TTS) 42
3.2.1 Considerations for Developing *TTS* 42
3.2.1.1 The Programming Language 43
3.2.1.2 Java-Applications versus Java-Applets 43
3.2.1.3 Data Replication Approach versus Real-Time Access 45
3.2.1.4 Defining a suitable configuration for TTS 46
3.2.2 Functionality of *TTS* 49
3.3 SYSTEM 2: *SELTOOL* 51
3.3.1 *SELTOOL*, Justification 52
3.3.2 *SELTOOL*, Functionality 52
3.3.3 Logic Associated with the Tool Selection Process 54
3.3.4 Database Considerations 58
3.4 COMPUTATIONAL SUPPORT 62
3.5 SUMMARY 62
CHAPTER 4 : KNOWLEDGE DISCOVERY IN DATABASES (KDD) 64
4.1 INTRODUCTION 64
4.2 DEFINITION 64
4.3 REASONS TO APPLY KNOWLEDGE DISCOVERY IN DATABASES 65
4.4 CONSIDERATIONS TO IMPLEMENT KDD-BASED SYSTEMS 67
4.5 ARCHITECTURE OF KNOWLEDGE DISCOVERY IN DATABASES 69
4.5.1 Building a Data Warehouse 70
4.5.2 Data Pre-processing 71
4.5.3 Data Mining 75
4.5.4 Post-processing 77
4.6 DATA MINING TECHNIQUES 77
4.6.1 Cluster Analysis 78
4.6.2 Neural Networks 78
4.6.3 Rough Sets 79
4.6.4 Genetic Algorithms 80
4.6.5 Fuzzy Sets 81

4.6.6 Discriminant Analysis 82
4.6.7 Machine Learning 83
4.6.8 Discussion 83
4.7 SUMMARY 85
CHAPTER 5 : CLUSTER ANALYSIS METHODOLOGY. PRE-PROCESSING STAGE 86
5.1 INTRODUCTION 86
5.2 CLUSTER ANALYSIS 87
5.3 CONVENTIONAL VERSUS CONCEPTUAL CLUSTERING 90
5.4 CLUSTERING METHODOLOGY 91
5.5 PRE-PROCESSING STAGE 92
5.5.1 Variable Selection 94
5.5.2 Entities Analysis 94
5.5.3 Variable Analysis 95
5.5.3.1 Analysis of Grades to obtain numerical similarity values 96
5.5.3.2 Analysis of Materials to obtain numerical similarity values 99
5.5.4 Data Standardisation 107
5.6 SUMMARY 108
CHAPTER 6 : CLUSTER ANALYSIS METHODOLOGY. PROCESSING AND POST-PROCESSING STAGES 110
6.1 INTRODUCTION 110
6.2 PROCESSING STAGE 110
6.2.1 Learning Modes 111
6.2.2 Generation of Similarity/Dissimilarity Matrix 113
6.2.2.1 Similarity 113
6.2.2.2 Dissimilarity 116
6.2.3 Clustering Method 116
6.2.4 Stop Criteria 119
6.3 POST-PROCESSING STAGE 120
6.3.1 Performance Evaluation 120
6.3.1.1 Intracluster Cohesion Index (ICI) 123
6.3.1.2 Intercluster Separation Level (ISL) 125
6.3.1.3 Quality of Final Classification (Q_c) 127
6.3.2 Optimisation Phase 128
6.3.3 Analysis of Results and Tests 130
6.4 SUMMARY 131
CHAPTER 7 : KNOWLEDGE DISCOVERY APPLICATIONS 133

7.1 INTRODUCTION 133
7.2 MOTIVATION FOR APPLYING KDD TECHNOLOGY TO THE TOOLING INDUSTRY 133
7.3 *DISKOVER*, AN INTEGRATED KDD-ORIENTED SYSTEM 134
7.3.1 *KLUSTER*, a Clustering Application 135
7.3.1.1 Application 137
7.3.1.2 Stopping Method 137
7.3.1.3 Learning Modes 137
7.3.1.4 Cluster Method 137
7.3.1.5 Approximation Method 138
7.3.2 A Fuzzy-Clustering Application 138
7.3.2.1 Analysis of Materials to obtain numerical similarity values 143
7.3.2.2 Analysis of Grades to obtain numerical similarity values 145
7.3.3 Q-FAST, an Exploratory Data Analysis (EDA) Application 148
7.3.3.1 Specific Type of Operation 148
7.3.3.2 Test Objectives 149
7.3.3.3 Materials 151
7.3.3.4 Benefits 151
7.3.4 *MQG*, a Multiple Query Generator Application 151
7.3.5 *R-Set*, a Rough Sets-based Application 151
7.4 SUMMARY 151
CHAPTER 8 : TESTING THE SYSTEMS AND ANALYSIS OF RESULTS..... 153
8.1 INTRODUCTION 153
8.1 REASONS TO DESIGN AND APPLY A SET OF TESTS 153
8.2 TESTS AND RESULTS FOR *SELTOOL* 154
8.2.1 A Case-Study 154
8.2.2 Completeness of the Database 156
8.3 TESTS AND RESULTS FOR *TTS* 160
8.3.1 Connectivity and Downloading Aspects 160
8.3.2 Confidence Scores Values 162
8.4 TESTS AND RESULTS FOR *DISKOVER* 164
8.5 SUMMARY 170
CHAPTER 9 : CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK 173
9.1 RESEARCH NOVELTY 173
9.2 CONCLUSIONS 174
9.3 RECOMMENDATIONS FOR FURTHER WORK 181

7.1 INTRODUCTION 133
7.2 MOTIVATION FOR APPLYING KDD TECHNOLOGY TO THE TOOLING INDUSTRY 133
7.3 *DISKOVER*, AN INTEGRATED KDD-ORIENTED SYSTEM 134
7.3.1 *KLUSTER*, a Clustering Application 135
7.3.1.1 Application *137*
7.3.1.2 Stopping Method *137*
7.3.1.3 Learning Modes *137*
7.3.1.4 Cluster Method *137*
7.3.1.5 Approximation Method *138*
7.3.2 A Fuzzy-Clustering Application 138
7.3.2.1 Analysis of Materials to obtain numerical similarity values *143*
7.3.2.2 Analysis of Grades to obtain numerical similarity values *145*
7.3.3 Q-FAST, an Exploratory Data Analysis (EDA) Application 148
7.3.3.1 Specific Type of Operation *148*
7.3.3.2 Test Objectives *149*
7.3.3.3 Materials *151*
7.3.3.4 Benefits *151*
7.3.4 *MQG*, a Multiple Query Generator Application 151
7.3.5 *R-Set*, a Rough Sets-based Application 151
7.4 SUMMARY 151
CHAPTER 8 : TESTING THE SYSTEMS AND ANALYSIS OF RESULTS..... 153
8.1 INTRODUCTION 153
8.1 REASONS TO DESIGN AND APPLY A SET OF TESTS 153
8.2 TESTS AND RESULTS FOR *SELTOOL* 154
8.2.1 A Case-Study 154
8.2.2 Completeness of the Database 156
8.3 TESTS AND RESULTS FOR *TTS* 160
8.3.1 Connectivity and Downloading Aspects 160
8.3.2 Confidence Scores Values 162
8.4 TESTS AND RESULTS FOR *DISKOVER* 164
8.5 SUMMARY 170
CHAPTER 9 : CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK 173
9.1 RESEARCH NOVELTY 173
9.2 CONCLUSIONS 174
9.3 RECOMMENDATIONS FOR FURTHER WORK 181

9.3.1 Recommendations for Further Academic Research....................................181

9.3.2 Recommendations for the Tooling Industry..183

APPENDIX A : HIERARCHICAL CLUSTERING METHODS - A NUMERICAL APPLICATION... 185

APPENDIX B : APPLYING ROUGH SETS TO DETECT DATA INCONSISTENCY.. 192

APPENDIX C : SELECTED DATA MINING APPLICATIONS 195

APPENDIX D : REGRESSION AND CORRELATION ANALYSIS 202

APPENDIX E : TEST DATA FOR TURNING OPERATIONS (TTS) 207

APPENDIX F : PUBLICATIONS 208

REFERENCES 209

LIST OF FIGURES

Figure 1.1 – A responsive manufacturing model. 7
Figure 1.2 - Sample of the Cyber Factory System Configuration (Noda, 1998). 13
Figure 1.3 - Proposed Architecture for an integrated CAPP/VR system 14
Figure 1.4 – System architecture of the described system 17
Figure 1.5 – VxInsight architecture 18

Figure 2.1 - Simple Java-based Architecture 32
Figure 2.2 - ARIEX Architecture. 35
Figure 2.3 - Complementary approach. Access of public and classified information sources. 39

Figure 3.1 - Java-application and Java-applet functionality 44
Figure 3.2 - Distributed solution and its different components 47
Figure 3.3 – Initial screen of TTS 49
Figure 3.4 – Data entering screen for tool trials 50
Figure 3.5 – A particular output screen provided by TTS 51
Figure 3.6 – SELTOOL, Functional Architecture 53
Figure 3.7 – Input screen of SELTOOL 54
Figure 3.8 - Matching Inserts with Toolholders 55
Figure 3.9 - (a) Profile OUT and approach angles – External Turning. (b) Profile IN and trailing angles – External Turning. (c) Profile OUT and approach angles – Internal Turning and (d) Profile IN and trailing angles, Internal Turning. 56
Figure 3.10 – Output screen of SELTOOL 57
Figure 3.11 - Data transformation process using DPS 58
Figure 3.12 - Input screen of the read-and-write Java application 59
Figure 3.13 - Data translation process 60

Figure 4.1 – KDD General Architecture 69
Figure 4.2 – Discretisation example 74
Figure 4.3 - Specialisation and generalisation examples 75
Figure 4.4 – Post-Processing application 77

Figure 5.1 - Four different formats to represent dendrograms 88
Figure 5.2 – Difference between conventional and conceptual clustering 90
Figure 5.3 - Clustering Methodology. 93
Figure 5.4 - Graphical representation of insert grade applications. 96
Figure 5.5 – Possible overlapping cases of insert grade applications. 98
Figure 5.6 – Regression Analysis for Steel material groups 106
Figure 5.7 – Regression Analysis for Stainless Steel material groups 106

Figure 5.8 – Regression Analysis for Cast Iron material groups *107*

Figure 6.1 – Visual recognition of a particular classification *119*
Figure 6.2 - Dissimilarity matrix and basic test data set *124*
Figure 6.3 - Dissimilarity matrix and considered distances *126*
Figure 6.4 - Measuring the quality of the final clusters *128*
Figure 6.5 – Optimisation phase after application of performance evaluation *129*

Figure 7.1 – Input screen for Kluster *136*
Figure 7.2 – Setting the run environment of Kluster *136*
Figure 7.3 – Typical output screen after running the optimisation phase *139*
Figure 7.4 – Examples of Gaussian and Trapezoidal membership functions *141*
Figure 7.5 – Membership grades for Material Groups *145*
Figure 7.6 – Overlapping areas between grades TP200 and TP15 .... *146*
Figure 7.7 – Input screen for Specific Type of Operation (Turning) *148*
Figure 7.8 – Typical output for Specific Type of Operation (Turning) *149*
Figure 7.9 – Input screen for Test Objectives option (Turning) *150*
Figure 7.10 - output screen for Test Objectives (Turning) *150*

Figure 8.1 – Input values for a selected case-study *155*
Figure 8.2 – Output Screen for a selected case-study *156*
Figure 8.3 - First and Second Choice Inserts for External Turning *157*
Figure 8.4 - First and Second Choice Inserts for Internal Turning *158*
Figure 8.5 - Inserts found using Profile Angles *159*
Figure 8.6 - Effectiveness of Confidence Scores (CS) *162*
Figure 8.7 – Average values for Confidence Scores *163*
Figure 8.8 – Obtaining cutting parameters from specific input requirements *166*
Figure 8.9 - Cutting parameters from specific input requirements running Fuzzy-K *167*
Figure 8.10 – Stratification of Grades and Materials regarding the best cluster found (running Kluster). *167*
Figure 8.11 - Stratification of Grades and Materials regarding the best cluster found (running Fuzzy-K). *168*
Figure 8.12 – Distribution of the three closest clusters according to Grades and Materials specifications (running Kluster) *169*
Figure 8.13 - Distribution of the three closest clusters according to Grades and Materials specifications (running Fuzzy-K) *169*

Figure A.1 - Initial data representation *185*
Figure A.2 - First data grouping *186*
Figure A.3 - Second data grouping *187*
Figure A.4 - Third data grouping *188*

Figure A.5 - Successive iterations. *189*
Figure A.6 - Successive iterations. *191*

Figure C.1 – Multiple Regression applied to Sales prediction. *196*
Figure C.2 - Decision Tree application. *199*
Figure C.3 – Discriminant Analysis to classify similar customer behaviours. *201*

Figure D.1 – Basic elements of Multiple Regression (Linear and Polynomial). *203*
Figure D.2 – Scatterplots for various values of correlation coefficients. *205*

LIST OF TABLES

Table 2.1 - Driver categories for database access using JDBC. 30

Table 3.1 - Cutting Speed data 57

Table 5.1 – Selected Variables 94
Table 5.2 – Workpiece materials classified into material groups 95
Table 5.3 – Numerical representation of insert grade applications 97
Table 5.4 – Workpiece material groups. 100
Table 5.5 – Mechanical Properties for Steel Material Groups. 101
Table 5.6 - Normalised Mechanical Property Values For Steel Material Groups. 103
Table 5.7 - Total scores of normalised mechanical properties 103
Table 5.8 - Individual Normalisation of Material Groups (Percentage). 104
Table 5.9 - Linearisation of the Machinability Index Scores For Regression Analysis. 105
Table 5.10 – Sample of the total data set considered 108
Table 5.11 - Additional standardisation measures 108

Table 6.1 – Medical data sample 114
Table 6.2 – Clustering methods and their parameters 118

Table 7.1 – Mechanical properties for Steel Material Group 143
Table 7.2 – Similarity generated for TX150 and TP grades family 146

Table 8.1 - Inserts Found for External Turning Operations. 156
Table 8.2 - Inserts Found for Internal Turning Operations 157
Table 8.3 - Number of inserts found. 159
Table 8.4 – Connectivity and Downloading Tests (TTS). 161
Table 8.5 – Performance of Clustering Methods (Kluster) 164
Table 8.6 – Quality indexes changing the stopping rule option 165

Table B.1 - Process condition and parameters. 192
Table B.2 - Process condition after transformation. 193

Table C.1 - Results of NN example. Reproduced with permission of Prof. Raymond Burke, Indiana University, USA. 197

Table E.1 - Test Data for Turning Operation (TTS) 207

LIST OF ABBREVIATIONS

API	Application Programming Interface
ARIEX	Architecture for Remote Information EXchange
ARPA	U.S. Defence Department's Advanced Research Projects Agency
ARPANET	ARPA experimental packet-switched network
AWT	Abstract Window Toolkit
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CAPP	Computer Aided Process Planning
CIM	Computer Integrated Manufacturing
CGI	Common Gateway Interface
CLI	Call Level Interface
CNC	Computer Numerically Controlled
CORBA	Common Object Request Broker Architecture
DBMS	Database Management System
DB2	Proprietary IBM Database
DMS	Database Migratory System
DPS	Database Populate System
DR	Data Replication
DW	Data Warehouse
EDA	Exploratory Data Analysis
FTP	File Transfer Protocol
GA	Genetic Algorithms
GUI	Graphical User Interface
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IBL	Instance-Based Learning
ICI	Intracluster Cohesion Index
IDL	Interface Definition Language
ISL	Intercluster Separation Level

ISP	Internet Service Provider
IT	Information Technology
JAVA VM	Java Virtual Machine
JDBC	Java Database Connectivity
JDK	Java Development Kit
JIT	Just In Time
KDD	Knowledge Discovery in Databases
NC	Network Computing
NN	Neural networks
ODBC	Open Database Connectivity
OMA	Object Management Architecture
OMG	Object Management Group
OOP	Object Oriented Programming
ORB	Object Request Broker
PP	Process Planning
PP&C	Process Planning and Control
RDBMS	Relational Database Management System
RI	Rule Induction
RMI	Remote Method Invocation
SPC	Statistical Process Control
SQL	Structured Query Language
TCP/IP	Transmission Control Protocol/Internet Protocol
TQC	Total Quality Control
URL	Uniform/Universal Resource Locator
VM	Virtual Manufacturing
VR	Virtual Reality
VRML	Virtual Reality Modelling Language
VRTP	Virtual Reality Transport Protocol
WWW	World Wide Web
W3C	WWW Consortium

CHAPTER 1

INTRODUCTION

This chapter introduces a general overview of the tooling sector within the manufacturing industry. However, the main contribution of this chapter is to provide a broad review of two important issues organisations are currently facing.

Firstly, the definition of efficient distributed strategies for supporting information integration and collaborative work schemes, particularly in corporate environments where it is necessary to share information of mutual interest but where their different members and branches are geographically dispersed. The motivation for adopting Web-based approaches is outlined, where concepts such as *Global Manufacturing*, *Agility* and *Virtual Manufacturing,* are reviewed.

Secondly, the existence of large amounts of raw data, occupying costly space and apparently idle, but potentially useful in terms of the hidden knowledge that can be discovered to support decision making processes. The idea of implementing knowledge discovery approaches exploiting the facilities provided by the Internet to access distributed information sources, is introduced.

A summary of this chapter and a statement of the research objectives follow. Finally, an outline of the thesis structure will be presented.

1.1 MANUFACTURING AND TOOLING INDUSTRY

According to Dagli (1994), *Cost, Markets, Quality* and *Flexibility* were the main issues addressed by the manufacturing industry in the last four decades. Competitive advantage in the 1960s was achieved by ***Cost*** minimisation and high product volume reducing set up time, to benefit from economies of scale.

In the 1970s the ability to access ***Markets*** was more important, and manufacturing information system design became an important issue implemented by using mainframe computers. At that time, the concepts of master production planning and control, hierarchical approach to planning and scheduling, functional flow control and computer numerical control (CNC), were introduced.

During the 1980s, it became more difficult to stay competitive in the market place because of exponential growth in communication systems around the world and the emergence of global markets. ***Quality*** was the key to competitive advantage during this time frame. The concepts of just-in-time (JIT), optimised production technology, statistical process control (SPC), total quality control (TQC), computer aided design (CAD), and computer aided manufacturing (CAM), became popular.

The availability of personal computers speeded up the previously described process, reducing the need for working in mainframe-based scenarios and providing extensive computational power at various locations within the factory. The first steps towards efficient networked environments and paperless organisations were made.

In the 1990s, manufacturing strategies were based on ***Flexibility*** as the key to competitive thrust. The manufacturing community, influenced by growing globalisation, was demanding more efficient methods to provide distributed business solutions.

In the early 2000s, the concept of ***Cyber-Factory*** is becoming an important paradigm supported by accelerated improvement in communications technology, virtual work schemes and information integration, where the Internet is playing a significant role.

1.1.1 General Manufacturing

Wu (1992), defines manufacturing as *the organised activity devoted to the transformation of raw materials into marketable goods.* Manufacturing industry is often called a *secondary* industry, because this is a sector of a nation's economy that is concerned with the processing of raw materials supplied by the *primary* industry (agriculture, forestry, fishing, mining, and so on) into the end products. A manufacturing system usually employs a series of value-adding manufacturing processes to convert the raw materials into more useful forms and eventually into finished products.

A manufacturing system is, therefore, a typical input-output system which produces outputs (economics goods) through activities of transformation from inputs (raw materials, parts or factors of manufacturing).

In a manufacturing company three main stages can be distinguished with regard to the fabrication of a product [(Hunt, 1989), (Rembold *et al.*, 1993), (Parsaei, 1995)]:

Design, where the bill of materials, quality control procedures, the dimensions and tolerances of the product will be described. *Process Planning*, where given the engineering design of an item which has to be manufactured, process planning is the act of generating an ordered sequence of the manufacturing operations necessary to produce that part within the available manufacturing facility.

Finally, *Production Planning & Control*, where once the process plan is generated, the following steps are typical at this stage:

- The operation sequencing starts to manufacture the product.
- Monitoring activities are implemented to follow a job and all parts through the plant.
- Keep track of the presence of employees and required skills.
- Monitor the assignment of the manufacturing resources.
- Observe the correct functions of the manufacturing equipment, control machine and defects.

The development, prosperity, welfare and living standards of a nation depend to a great extent upon the success of its manufacturing industries; to any industrialised country manufacturing is important internally as well as externally. Significant internal factors are; continued employment, quality of life and the creation and preservation of skills. External factors are national defence and the nation's position and strength in world affairs.

1.1.2 Tooling Industry

The manufacturing industry is highly dependent on the tooling, which is needed to machine the components used to make the range of products seen in today's society (Revere, 2000).

Tooling operations requiring more attention are those relating to *tool selection, tool life prediction, cutting data recommendation, machining operation selection and comparative analysis of cutting conditions.* A brief overview of some of these parameters is presented as follows.

1.1.2.1 Tool Selection

One of the more important functions in Process Planning (PP) is the appropriate selection of tools for the machining processes. It is necessary to apply certain criteria of selection in order to identify from a considerable number of available options, the best tool to be used. An efficient tool selection method will contribute to minimise the manufacturing time cycle and, hence, it will have a strong influence in the reduction of costs.

The method should be able to identify the tool requirements, once the design parameters of the workpiece, type of material, type of operation, type of cutting and other relevant information have been evaluated. Then, a match can be made between these specifications and existing tools.

Simple workpieces can be machined using only one tool. However, certain parameters can influence the selection of more than one tool, according to the need to satisfy additional machining requirements. In this case the selection method must consider a multiple tool selection criterion.

1.1.2.2 Cutting Data

When a cutting tool is applied to a workpiece, a layer of metal is removed in the form of a chip. The type of chip produced during metal cutting depends on the material being machined and the cutting conditions used (Boothroyd and Knight, 1989).

The information generated when defining the cutting conditions is known as cutting data. The cutting data usually include information about *Feed Rate, Cutting Speed* and *Depth of Cut*.

Procedures to select appropriate cutting data are needed in order to reduce the restrictions imposed by the cutting processes. Various constraints act upon the cutting process, the most critical being: the geometric tool suitability, tool life, cutting forces, machine power, tool or workpiece deflection, chip capacity of a cutter, chatter and surface finish (Carpenter, 1996).

1.1.2.3 Tool Life Prediction

The life of a determined tool can be reduced by its progressive wear as well as by premature failures.

Signs of wear can be recognised on areas of the face and flanks of the cutting tools, while premature failures can be produced by factors such as dropping the tool, incorrect selection and use, shop-floor conditions, incorrect coolant conditions and damages to accessories such as carbide inserts (Boothroyd and Knight, 1989).

Alamin (1996) designed an off-line tool life control and management system (TLC). The predictions were based on the optimisation of cutting data using three tool life criteria: user defined tool life, tool life for minimum production cost and tool life for maximum production rate. The functioning of TLC is based on tool life coefficients obtained from tool manufacturers' data in order to calculate the tool life value in a theoretical way.

Having provided a summarised description of the main issues addressed by the tooling industry, to conclude this section a review of selected applications in this area will follow.

Monostori and Viharos (1995) proposed a novel approach for generating multipurpose models of machining operations, combining machine learning and search techniques. These models are intended to be applicable at different engineering and management assignments. Simulated annealing search is used for finding the unknown parameters of the model in given situations. It is expected that the developed block-oriented framework will be a valuable tool for modelling, monitoring and optimisation of manufacturing processes and process chains.

A framework for machining operation planning systems was proposed by Kojima *et al.* (1999). A prototype system to advise the engineer of cutting conditions, including trouble shooting for side end milling was developed to demonstrate the concept.

Zhou and Harrison (1997) described a fuzzy neural hybrid model to compensate process errors in CNC machining by touch trigger probe systems. The proposed system reveals that it is feasible to achieve an improved machining performance by implementing fuzzy membership functions and generating linguistic control rules.

Wood *et al.* (1997) examined the application of Neural Networks (NN) in traditional machining processes. Also, they developed a NN-based multi-layer perceptron to produce a correlation between vibration measurements gathered during a machining operation and the condition of the tool tip. This investigation showed the potential of NN in condition monitoring applications.

The above reviewed applications constitute good examples of AI-based implementations to provide solutions in the tooling industry. It was noted how disciplines such as Neural Networks and Fuzzy Sets play a predominant role in the development of current applications.

1.2 AGILE MANUFACTURING

In general all definitions of Agile Manufacturing converge on a common goal, the ability to quickly respond to demanding and dynamic scenarios. For example, Yuan *et al.* (1999) affirm that the agility of an enterprise is determined by its ability in respect of opportunity and innovation management. Noaker (1994), states that Agile Manufacturing is lean manufacturing specifically tailored to deliver rapid response to continually changing situations.

The primary goal of the manufacturing enterprise of the future remains essentially on its agility or dynamic adaptability. It has to achieve rapid, flexible and integrated development and manufacture of innovative products at a price the customer is prepared to pay. To thrive in the emerging market conditions it has to be capable of rapidly responding to market trends and operating as an efficient member of an extended and increasingly global supply network (Gindy, 1999).

Furthermore, Gindy proposed a responsive manufacturing model for the UK aerospace industry, which is shown in Figure 1.1. *Adaptability* and *change proficiency* can be identified as the most significant parameters in achieving manufacturing responsiveness. A description of the main factors characterising these parameters is presented.

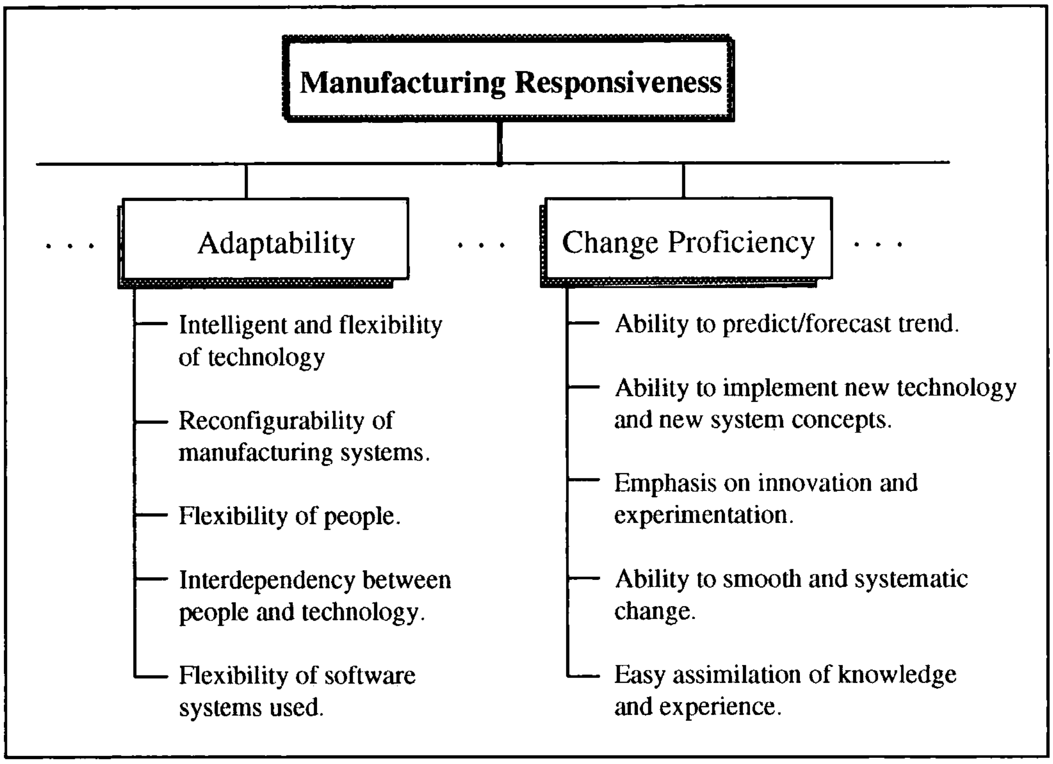

Manufacturing Responsiveness
. . .
Adaptability
. . .
Change Proficiency
. . .
Intelligent and flexibility of technology
Reconfigurability of manufacturing systems.
Flexibility of people.
Interdependency between people and technology.
Flexibility of software systems used.
Ability to predict/forecast trend.
Ability to implement new technology and new system concepts.
Emphasis on innovation and experimentation.
Ability to smooth and systematic change.
Easy assimilation of knowledge and experience.

Figure 1.1 – A responsive manufacturing model.

Park *et al.* (1993) proposed using the Internet, along with the standards, protocols and technologies developed and refined over the years by the academic community, to build what is called *Agile Manufacturing Systems* (Goldman and Nagel, 1993). The concept is built around the principle of client-server computing. The servers encapsulate manufacturing resources and applications, and make them available as network services.

Tsinopoulos and McCarthy (1999), introduced a biological classification technique called *"cladistics"* to achieve manufacturing agility. The main idea behind this concept is that, regardless of the industrial sector, managers could use the proposed *cladistic* approach as an evolutionary technique for determining "*where they have been and where they are now*".

Yuan *et al.* (1999) investigated the influence and importance of "Cellular Manufacturing" in agile manufacturing scenarios. They compared two kinds of manufacturing cells, namely *Grouping Manufacturing Cell* and *Virtual Manufacturing Cell.* A manufacturing cell was defined as any functional entity which holds certain kind of specific manufacturing capabilities and can accomplish all the manufacturing tasks related to a part or an entire product.

As an advanced manufacturing paradigm for the next century, the strategic objective of Agile Manufacturing is to produce products of the highest quality, the lowest cost and satisfying the customer from multiple aspects within the given time frame by the effective integration of people, technology and organisation. It makes synthetic use of concepts such as Just-In-Time, Concurrent Engineering and Resources Optimisation, and it is thoroughly customer-oriented (Nagel *et al.*, 1991).

1.3 GLOBAL AND DISTRIBUTED MANUFACTURING

Currently, the work strategies of manufacturing organisations are being influenced by a growing world-wide standardisation of processes and the emergence of automated infrastructures around global information networks. Therefore, the different business activities involving customers, partners, employees and suppliers must be efficiently integrated in collaborative environments, regardless of the physical locations where these activities are carried out.

In a manufacturing context, collaborative environment refers to a situation where a group of geographically alienated people work together on certain product development or manufacturing operations (Leung, 1995).

It has become widely accepted that the future of manufacturing organisations will be information-oriented, knowledge driven and much of the daily operations will be automated around a global information network that connects everyone together (Leung *et al.*, 1995).

The arrival of the Internet and its adoption as an international standard, has been a decisive factor to implement world-wide information exchange infrastructures, where the advantages of sharing valuable information and knowledge from remote locations are now considerably exploited.

In a very competitive world and increasing global markets, manufacturing companies are taking advantage of networking activities in which resources, learning, experience and knowledge can be shared (Casavant and Singhal, 1994).

In a sharper perspective, Regli (1997) affirms that *"Companies that can not handle the accelerated flow of information made possible by networks may be eclipsed by those that can"*.

In recent years, interesting research initiatives focused under distributed philosophies and web-based architectures have been undertaken, and are presented as follows.

In the area of industrial manufacturing engineering, Feldmann and Rottbauer (1999) proposed an electronically networked assembly framework for global manufacturing.

In their project about integrating structured databases into the Web, Eichmann *et al.* (1998) developed an architecture called *MORE* (*Multimedia Oriented Repository Environment*). *MORE* was designed as a set of application programs, specifically a set of CGI executables, that operate in conjunction with a stock HTTP server to provide access to a relational database of meta-data.

Following the same guidelines of the above research, Smith and Wright (1998) proposed to take advantage of WWW technologies to provide the first computer-aided design-to-manufacture web-site, through the development of a tool called *"CyberCut"*. It provides users with a CAD front-end to a computer aided process planner already proven within an end-to-end CAD/CAPP/CAM system.

The way the architecture of *CyberCut* has been structured requires further explanation. The external client enters the design interface over the Web, while the planner and manufacturing system reside in the company web server. The client then proceeds to design the required component, using an interface developed for this particular task. The user introduces the design feature by feature, and is given different options on how to enter the data. Each design decision is communicated automatically to the planner. The process plan, cost, feasibility and estimated time to manufacture the emerging component are communicated back to the user automatically.

When the design is successfully completed, the designer clicks on an order part button. The part is placed in a fabrication queue, and when previously ordered jobs are completed, manufacturing is initiated at the company, and the part is shipped to the designer through a mail carrier of his/her choice.

Tian *et al.* (1997) reviewed the impact of the WWW technology on manufacturing based information technology (IT), and addressed the lack of a general information infrastructure for supporting interactions between co-operating partners in virtual manufacturing enterprises. The authors discuss a prototype based on integration of the Internet and local manufacturing networks, which is the backbone of the global agile manufacturing strategies that will be widely implemented in the very near future.

To provide remote-manufacturing services, Küssel *et al.* (1999) examined the concept of "TeleService". This term was formulated in order to fulfil these criteria:

- *Geographical Distance.* The service has to be provided over a spatial distance. This means that the service has to be provided by a technician who is spatially separated from the customer.
- *Use of information Technology (IT).* The use of communication and IT is essential when carrying out a service.
- *Industrial Service.* The services performed, had to be in the field of industrial services. Industrial services are, for instance, maintenance, diagnosis and repairs.

In the same way, for distributed solutions, and in the context of industrial design and manufacturing, Zhao *et al.* (1999) proposed an object-oriented manufacturing data model for a global enterprise, where the information components of the model will be distributed across a global network.

Frequently, there are hardware differences, and heterogeneous design and computing environments existing in manufacturing companies. Pan *et al.* (1999) proposed an approach based on Java programming and Intranet/Internet technologies. The main aim of this proposal, is to provide a universal platform-independent environment for effectively accessing different hardware, computing, design and manufacturing systems at various levels within a manufacturing organisation.

1.4 VIRTUAL MANUFACTURING

The accelerated advances in Virtual Reality (VR) technology as well as the efforts to gain competitive advantage in current manufacturing scenarios, characterised by open markets, reductions in trade barriers and improvements in communications, have led to the emergence of Virtual Manufacturing (VM) environments.

VR interaction techniques offer a significant step forward in man-machine communication, beyond graphical WIMP (Windows, Icons, Menus and Pointer) based interactive systems (Pimentel and Texeira, 1994).

In VR systems, application data are represented as virtual geometrical objects that interact with each other and the user with semantically realistic behaviour. Furthermore, with VR, the application developer is able to construct a simulation environment in which the user is an essential participant rather than a passive observer, thereby providing the user with the illusion of "being there" (Sastry and Boyd, 1998).

The implementation of highly immersive VR techniques in the manufacturing industry is still the subject of much research (Taylor, 1995). However, some concepts derived from VR technology have recently been applied to the industrial arena, supporting Virtual Manufacturing (VM) activities.

The concept of virtual environment fits well into current manufacturing activities, strongly supported by network computing (NC). NC supports “virtual offices”, where any user can just plug into a NC port connected to the corporate network (whether onsite or through an Internet connection) and access his/her own workspace, documents, e-mail and other information resources. The NC approach allows scaling down a computing environment from large servers to personal digital assistants, a move that usually produces significant cost savings (Westphal and Blaxton, 1998).

According to Hamel and Wainwright (1999), a virtual corporation is a temporary partnership of independent organisations and/or individual suppliers of specific goods and services, who are linked through modern telecommunication to exploit and profit from rapidly changing business opportunities. This corporation is called "virtual" because it is composed of partners of core competence and has neither a central office nor hierarchical or vertical integration.

From the point of view of product development and according to Wahab and Bendiab (1997), “virtual organisations” is a strategic concept, which enables two or more organisations with complementary core competencies, to jointly develop new products irrespective of departmental or organisational boundaries and geographical location.

Companies eventually form consortiums in response to certain market sectors. The consortiums only exist as long as the market is there, and break up quickly and form another when necessary. This is sometimes called *virtual companies*, a concept where companies are considered as being autonomous units of an enterprise that manufactures a certain product (Leung *et at.*, 1995).

In the area of product realisation processes, Giachetti (1999) states that the current market conditions have led enterprises to focus on their core businesses and increasingly co-operate with suppliers and customers. This is referred to as virtual enterprise. The inclusion of suppliers in the product realisation process calls for greater collaborative work than has previously been the case. Activities must now be performed across organisational boundaries throughout the product realisation process. Information technology (IT) is regarded as a means for these geographically dispersed companies to collaborate on new product development, manufacture and delivery.

More broadly, Noda (1998) discussed the new role of manufacturing industries in the Internet era. The author has defined the word "*Internet Cyber-Factory*" as the ability to access all resources of a production system through a networks-based environment, and where the Internet technology plays a fundamental role. The goal of this project is to construct a tool called Cyber FA Kit to make access to the *Internet Cyber-Factories* as simple as possible.

Then, a new industry could be described by the following elements: **(i)** Cyber Mall, **(ii)** Cyber Manufacturing and **(iii)** Cyber FA (Factory) Kit. Figure 1.2 shows an experimental system configuration including the key elements.

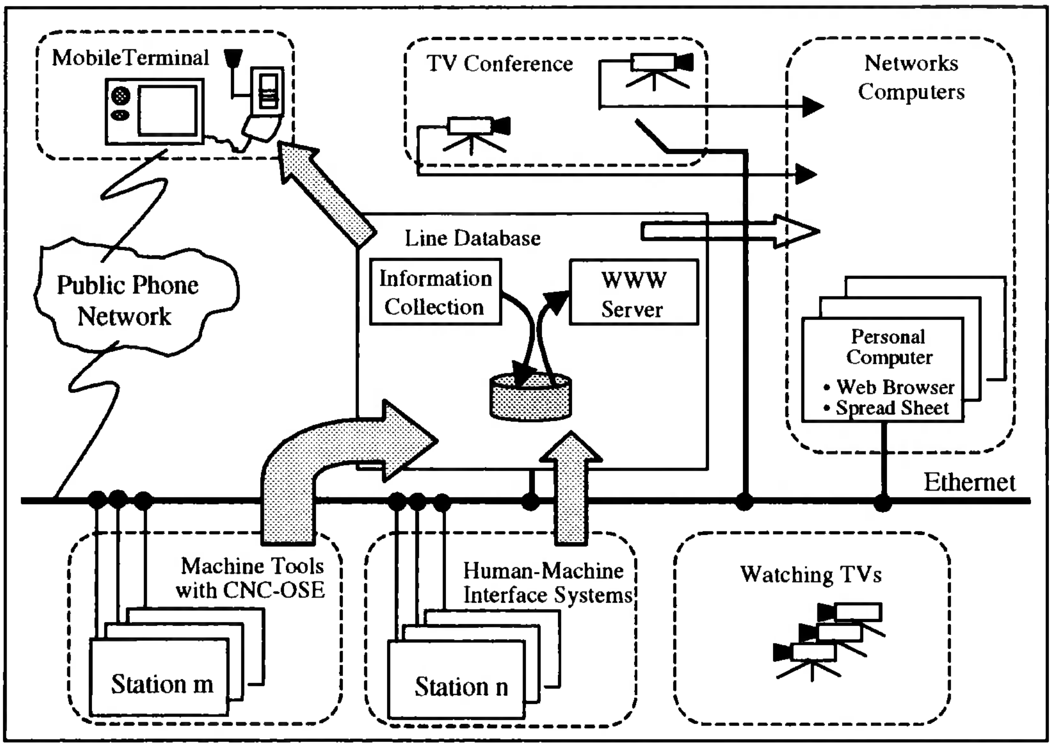


Figure 1.2 - Sample of the Cyber Factory System Configuration (Noda, 1998).

The above configuration shows the functionality of the proposed architecture to monitor for example, machine tools with CNC-OSE[1], from any WWW terminal. This application is useful in production line supervision tasks.

[1] The Open System Environment (OSE) Consortium in Japan, is one of the standardisation study group for the manufacturing industries, especially for an open controller architecture.

Some selected applications of VR in the area of manufacturing are described as follows.

In the area of cable harness design, Ng *et al.* (1999), described a novel software tool to assist users to perform cable routing in a virtual environment. This application provides a potentially useful solution to a traditionally difficult, costly and tail-end part of the overall product design process.

Carpenter *et al.*, (1997) affirm that one particularly novel area of investigation is the use of VR for automated manual assembly planning. In this area, VR could provide an excellent environment for the unobtrusive observation of assembly experts at work. As the expert works, the system can record the movements of the user and all the components. Thus, after the expert has assembled the given components in the virtual scenario, an assembly plan can be automatically generated from the stored usage data.

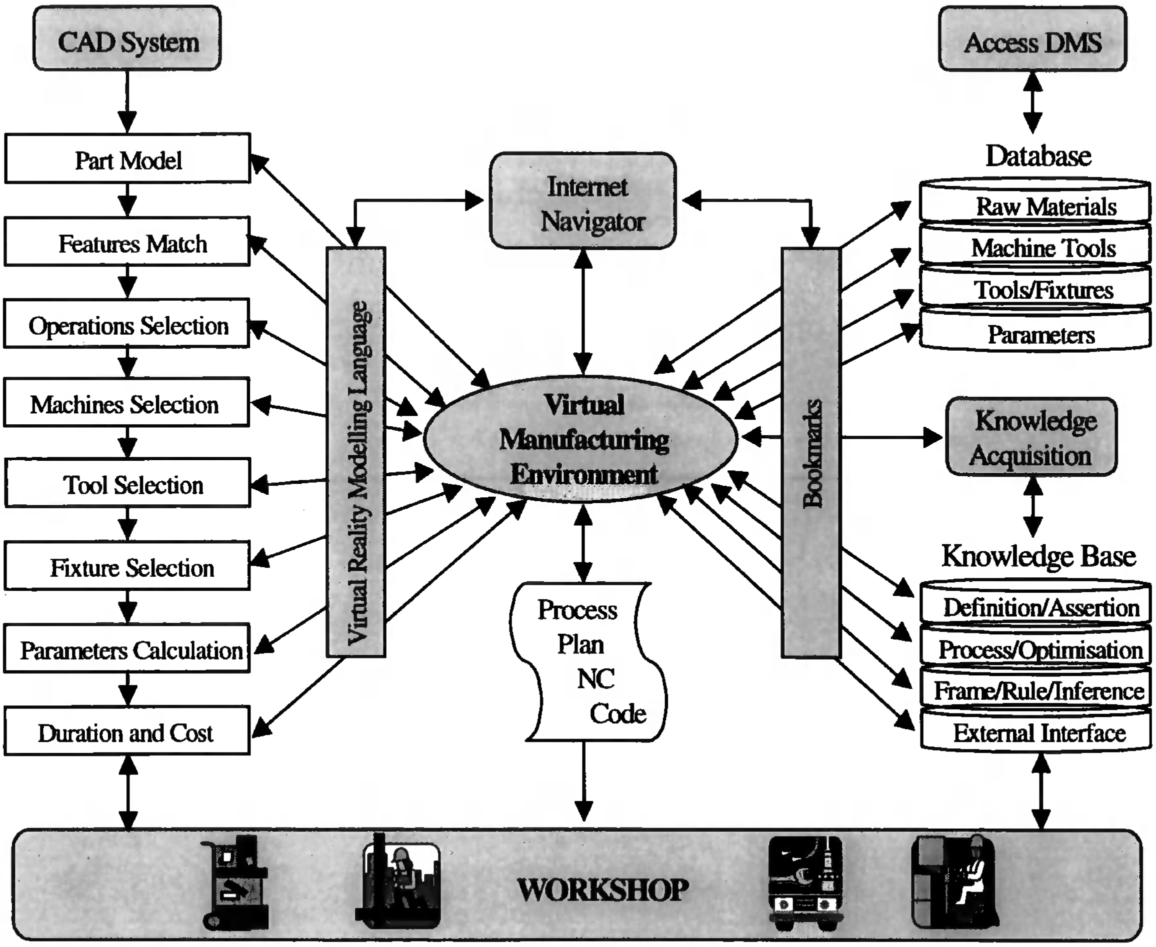


Figure 1.3 - Proposed Architecture for an integrated CAPP/VR system

Working on models for extracting geometric features, Ma and Chu (1999) developed a set of algorithms for constructing a high-level CAD model found in a virtual environment. The constructed CAD model can be transferred into a traditional CAD system for further design, model verification and product reengineering.

Based on a model of VM systems for an extended CIM enterprise, Plipovic *et al.* (1999) presented the concepts and architecture for a control-centred VM system, and a virtual environment for the modelling of virtual programmable logic controllers and assembly cells, consisting of pick-and-place robots.

Peng *et al.* (1999) proposed an integrated CAPP/VR system to improve current CAPP aplications. The architecture of this system is shown in Figure 1.3.

It should be noted how it is possible to directly access various resources from a database, a knowledge base and the Internet, during the process planning. The design process of CAPP can be simulated visually and can interact with the designer in real time.

1.5 FROM RAW DATA STORAGE TO A WEB-BASED KNOWLEDGE DISCOVERY APPROACH

An urgent need exists for tools to help manage, extract information and discover knowledge from increasingly large collections of data, now being constructed and made available from data warehouses.

Efforts to reveal meaningful patterns in large sets of raw data have led to the emergence of a relatively new research area, often called Knowledge Discovery in Databases (KDD). It has also been referred to as *Knowledge Discovery, Knowledge Mining from Databases (KMD), Knowledge Extraction, Data Analysis* or simply *Data Mining.*

KDD technology has usually been applied on stand-alone data repositories, where the benefits of accessing multiple and distributed data sources are not fully exploited. The vast interconnection of remotely dispersed computers, connected through a World Wide Web (WWW), opens a promising path to implement KDD-oriented techniques for analysing geographically dispersed databases.

Carpenter (2000) affirms that efforts to implement knowledge engineering in manufacturing are being supported mainly, by two factors:

- Relentless pressure for faster innovations.
- The increasing importance of the Internet, providing new communicational opportunities.

Some authors have highlighted the significance of the above idea. For example, (Chen *et al.*, 1996) state *"As a young and promising field, data mining still faces many challenges and unsolved problems ... for example ... and data mining in Internet information systems are important issues for further study."*

Despite copious literature about conventional KDD applications, not many works were found in which KDD approaches are implemented using a Web-based platform. However, some proposals and related ideas to address this issue have recently been suggested and they are discussed as follows.

(Wong *et al.*, 1998) developed a *"Framework for a World Wide Web-based Data Mining system"* that deals with stock-market data available on certain Web-sites. The system classifies these stocks as *buy* or *sell* using decision tree techniques. The user interaction is through a Web browser, which makes the Web-based nature of the data gathering and processing transparent to the user.

A view of the architecture of this system is shown in Figure 1.4. The different steps are summarised below.

- *Step 1: User interface.* Commercially available Web browsers support the interaction. The Web server is used for communication with the user, and also to interact with the WWW. CGI is a standard for communication between Web documents and CGI scripts. CGI scripting, or programming, is the act of creating a program that adheres to this standard of communication.

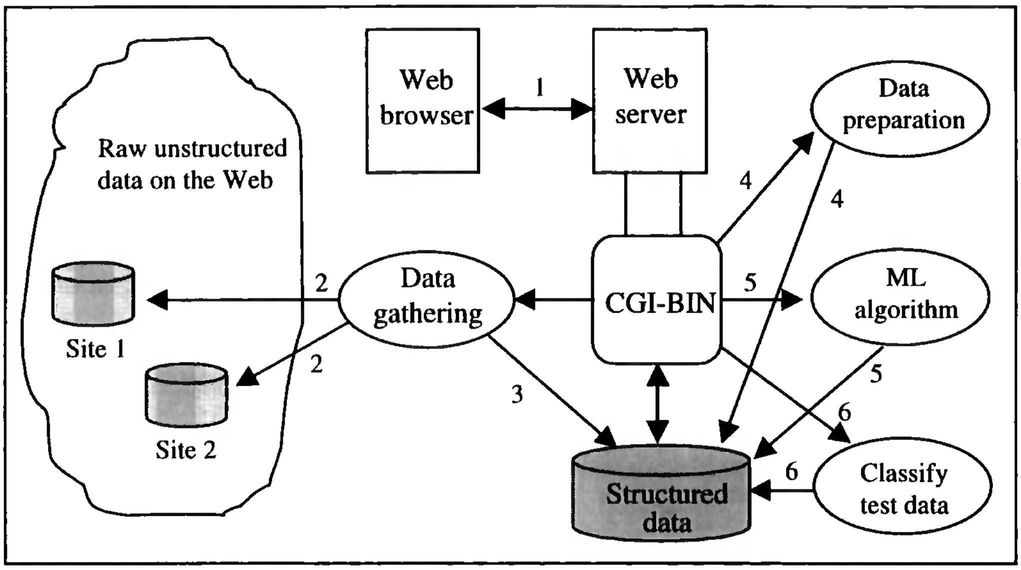


Figure 1.4 – System architecture of the described system

- *Step 2: Data gathering.* This activity is related to contacting the various Web-sites that contain the raw stock-market data.

- *Step 3: Storing the data.* Databases are not usually designed for discovery purposes and they contain spurious information, which need to be cleaned out before the data can be used. After this step, the data will be in a *structured* format, ready to be processed later.

- *Step 4: Data preparation.* This task converts the structured data into a format that is to be used by the machine learning (ML) algorithm.

- *Step 5: ML algorithm.* Using the training data that is prepared in the previous step, the test data is classified.

- *Step 6: Classify test data.* Finally, using the tree created in the previous step, the test data is classified.

A different application named VxInsight that does not use a Web-based interaction but a client-server approach was developed by (Davidson *et al.*, 1998). VxInsight is a visualisation tool built to find interesting patterns and information underlying large quantities of textual data.

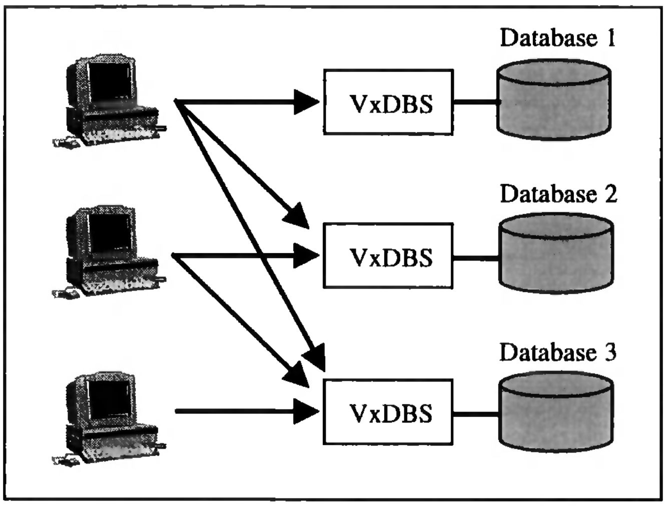


Figure 1.5 – VxInsight architecture

Figure 1.5 shows the connections between the databases and the clients using visualisation facilities. In general, the analyst may connect to several databases and each database may have multiple clients. The client-server connections are based in particular *sockets* that allow cross-platform communications.

Finally, an algorithm for the mining of association rules in distributed databases (*DMA*) has been developed (Cheung *et al.*, 1996). Although *DMA* was not built under a Web-based philosophy, its distributed features overcome the performance of sequential algorithms.

The distributed database in the above model is a horizontally partitioned database. The database scheme of all the partitions is the same, i.e., the records are transactions on the same set of items.

1.5.1 KDD Technology in the Tooling Industry

In spite of a considerable amount of tooling data that is continuously generated in the machining centres of manufacturing companies, to the best of the author's knowledge, there are no formal KDD-oriented systems implemented in this field.

The last decade has seen a considerable application of KDD technology in social, economic and scientific fields. However, tooling remains a largely unexplored sector.

Furthermore, the utilisation of the Internet by tooling corporate companies as an important resource to integrate information distributed across their geographically dispersed branches, and the use of this information for later KDD purposes, is not yet a fully exploited idea.

1.6 SUMMARY

This chapter has explored general issues in relation to manufacturing and the tooling industry. *Tool selection, tool life prediction, cutting data recommendation, machining operation selection and comparative analysis of cutting conditions* were identified as topics requiring special attention.

The needs for information integration and more co-ordinated distributed manufacturing strategies in corporate environments, have been examined. It was shown how *Agility, Global Manufacturing, Cyber Factories* and *Virtual Manufacturing,* are important concepts significantly influencing the fast changes taking place in the current industrial manufacturing scenarios.

The motivation of corporate companies for implementing collaborative and shared-information platforms, where Web-based strategies are playing a fundamental role to integrate distributed information sources, was evidenced.

This chapter has also shown how Knowledge Discovery in Databases (KDD) has become a potentially useful discipline to analyse data-rich scenarios, disclosing evidences that can contribute to provide a better understanding of the domain under study and support decision-making processes. Finally, It was found that the manufacturing tooling industry is a potentially unexplored sector requiring the application of KDD technology.

1.7 AIMS AND OBJECTIVES OF THIS RESEARCH

This research focuses on two main aims. First, the proposal and implementation of Web-based strategies to support information integration and collaborative work in corporate environments. Secondly, the development and implementation of KDD-oriented approaches, under an Internet-based platform, to identify meaningful patterns and useful relationships in raw data sets.

It is expected that these solutions can be implemented in the manufacturing tooling area, specifically, in the company Seco Tools (UK) Ltd, and later, be adopted by the remaining group of Seco companies located in other countries.

To achieve the general aims described above, the following specific objectives are proposed:

- To investigate the main issues in relation to the implementation of Web-based strategies to support information integration in corporate environments.
- To define a Web-based architecture to support remote information exchange in a manufacturing tooling company.
- To create a well-structured database of tooling data, for supporting a later implementation of systems oriented to solve industrial tooling problems.
- To provide an effective Web-based framework to support tooling engineers in the task of solving a particular tooling problem, whilst also avoiding the need for executing new tool trials.
- To provide a Web-based framework where world-wide users will be able to access tooling information, benefiting from the implementation of an open-access architecture.
- To analyse the motivation, benefits and main issues to be addressed when KDD-oriented systems are implemented.
- To propose a formal methodology to support the development of KDD-oriented systems.
- To analyse manufacturing tooling data, in order to discover potentially useful hidden knowledge, under an Internet-based platform.

1.8 THESIS STRUCTURE

This thesis has been organised within nine chapters, which are outlined as follows.

Chapter 1 has presented a review of the areas related to this research. Some examples in which Web-based strategies can be applied to support the implementation of distributed manufacturing solutions, were provided. Also, the motivation for implementing knowledge discovery approaches was evidenced.

Chapter 2 examines important issues to be considered when implementing Web-based strategies in corporate work environments. An *Architecture for* ***R****emote* ***I****nformation* ***Ex****change* (***ARIEX***) is proposed.

Chapter 3 provides details about ***TTS*** and ***SELTOOL***, two Internet-based systems developed by the author and his colleagues, to take advantage of sharing tooling information from geographically distant places.

Chapter 4 concentrates on the definition of a well-structured architecture to support the development of KDD-based systems. Also, selected data mining techniques are discussed.

Chapter 5 proposes a formal methodology to support the development of clustering applications. The whole methodology has been structured in three main stages. This chapter will provide details about the *Pre-Processing* stage.

Chapter 6 then continues examining the remaining structure of the clustering methodology proposed in the previous chapter. This chapter goes on to explain the *Processing* as well as the *Post-Processing* stages.

Chapter 7 looks in detail at the design and implementation of ***DISKOVER***, an Internet-based and integrated KDD system, developed by the author and another colleague. ***DISKOVER*** incorporates a clustering application, a hybrid application combining cluster analysis and fuzzy sets, an SQL-based Exploratory Data Analysis (EDA) utility, a multiple query generator utility and a rough sets-based application.

Chapter 8 presents the test structure applied to all the systems developed in this research, as well as a comparative analysis of the results obtained.

Chapter 9 summarises this thesis, drawing overall conclusions and identifying opportunities for undertaking further research.

CHAPTER 2

REMOTE INFORMATION EXCHANGE IN CORPORATE ENVIRONMENTS

2.1 INTRODUCTION

The tooling sector constitutes the industrial area where the applications in this research have been focused. Hence, in the previous chapter an investigation about the main technical issues, particularly in the arena of process planning, was carried out. Also, the need for more co-ordinated distributed manufacturing strategies was evidenced, where related concepts such as *Agility*, *Virtual Environments* and *Global Manufacturing* were examined. Furthermore, the suitability for implementing a knowledge discovery approach, taking benefit from distributed data sources, was presented. Finally, a critical review of some proposed solutions within the literature was simultaneously provided.

Here, the challenge is to propose a Web-based distributed framework to support the access and updating of geographically dispersed information. In the first section, some important considerations in relation to implementing distributed work strategies are examined. An analysis of the significant role of the Internet in the current global scenarios follows. The next section explores the issues to be addressed when working on distributed platforms, particularly in relation to connectivity and database access. The final section presents *ARIEX*, an ***A**rchitecture for **R**emote **I**nformation **Ex**change*, which constitutes the core of the framework proposed.

2.2 REQUIRED FUNCTIONALITY

The potential of Web-based strategies to support information integration and collaborative work in the manufacturing industry has been outlined in the previous chapter. However, some topics need particular attention and a more detailed analysis. In this section, these issues are addressed as follows.

i) *Information Security.*

The proposal of Web-based strategies embraces an apparent paradox. Firstly, the primary concept of WWW naturally encourages information openness. In an ideal scenario, all users would access the information resources in a transparent way, regardless of their software, hardware or installed communication platforms. On the other hand, it is not surprising that information security is a critical issue when industrial applications are considered.

Hence, there is an imperative need for implementing efficient mechanisms to serve as a barrier between *private* and *free-access* information sources. Measures to keep the confidentiality of private information sources include the creation of security levels (hierarchical access), encrypted passwords and firewall programs, at lower levels.

ii) *Database Technology.*

Relational Database Management Systems (RDBMS) have been adopted in the last decade, as a standard database model for providing a relatively successful platform for industrial data storage. However, the current relational data management technology requires improvement in relation to two main aspects.

First, suitable tools and techniques to extract, transform, replicate and update data from multiple and heterogeneous sources. Secondly, the efficient storage and retrieval of large objects, such as heavy images, animated objects and elements that can represent simulation of industrial processes (Bezos, 1994). For instance, the current SQL-based databases do not support an efficient management of queries in relation to multimedia data.

The object-oriented model appears to be the logical alternative to overcome the relational model limitations. Nevertheless, an object-oriented database model faces the following challenges:

- Its development represents a complex task.
- The object-oriented model lacks a standardised language like SQL for the international community using relational models.

- Also, as with current RDBMSs, an object-oriented database needs to address the problem of storing and retrieving multimedia data entities.

iii) *Ability to Manage Knowledge.*

There are two main factors motivating the application of knowledge engineering approaches to data-rich scenarios. First, as a consequence of the multiple processes and operations that take place in the manufacturing industry, large amounts of data are continuously generated and stored for later, varied purposes. Secondly, the decision-making processes are better supported through available knowledge than raw data.

Hence, it is appropriate to rely on mechanisms that allow extracting potentially useful knowledge from large data storage sources. *Knowledge Discovery in Databases* (*KDD*) has emerged as an important and fast-growing discipline for satisfying this important need (see Chapter 4).

iv) *Portable Solutions.*

Due to the distributed nature of Web-based solutions, it is desirable to rely on modular and easily portable systems, able to be installed across multiple and heterogeneous computer platforms. The code should be designed so that it is easy to modify, extend and transfer, to enable customisation for various clients, inter-organisational divisions and corporate branches.

v) *Standards.*

Geographically dispersed users accessing information through the WWW are often using different operating systems, communication protocols and multiple data formats. Manufacturing is one of the main areas where the convergence of these multiple resources is clearly evident, particularly, due to its extensive use of automation and technology.

Therefore, there is an urgent need for creating standards that can support the dynamic and global information exchange processes, currently taking place in the manufacturing industry.

vi) *Computing Support.*

The software that is being currently used in networked environments, keeps relatively rigid structures so that simple tasks and manufacturing processes can be difficult to describe or simulate. Computing animations and graphical representations will need improved software that can support the dynamic work schemes of manufacturing environments.

One important aspect is the weak support provided by the current browsers to database operations on the Internet.

Specifically, the improvements would be focused on:

- Data storage, retrieval and updating technology.
- Emulation of flows and industrial processes (animation).
- Quick and safe transfer of applications among remote workstations and easy modification of these applications for interdisciplinary work groups.
- More friendly and powerful interfaces supporting advanced Multimedia/Hypermedia.
- Multi-parallel processing.
- Strong compatibility between software, hardware and communication tools.

vii) *Multimedia/Hypermedia as an Integration Tool.*

The concepts of Multimedia and Hypermedia are often used interchangeably. In its most basic form, multimedia can be defined as a computer process that can handle text, images, video, sound and animated graphics (Scott, 1990). Hypermedia originated from hypertext, defined as text in electronic form that takes advantage of the interactive capabilities (Conklin, 1987).

Leung *et al.* (1995) have carried out an exhaustive analysis into the potential benefits of advanced multimedia/hypermedia. They confirm that these concepts are quite useful in addressing the problems of information integration in the manufacturing industry. One reason is because manufacturing is a unique environment in the way that there exist a large number of software systems, user manuals, databases and standard documents.

The concept of hypermedia can provide a higher level of integration in two respects. First, to logically associate information residing in heterogeneous systems, to save time in hunting for precise information (Gertley and Magee, 1991), and secondly, the advantages of exchanging dynamic information with these systems [(Isakowitz, 1993), (Bieber, 1993)].

2.3 THE ROLE OF THE INTERNET

Requests for information are increasing, forcing researchers to look for better computing facilities that can satisfy their requirements. The arrival of the Internet and its adoption as an international standard, has been a decisive factor in this development. Internet technologies allow open and user friendly communication; the Internet is a world-wide network that helps people communicate more effectively. Communication can take place irrespective of languages, cultures, distances and at a relatively low cost.

It is becoming obvious that the Internet can be used as a computational platform, to integrate multidisciplinary work groups, taking decisions and developing applications in different and distant places. However, these advantages are still not exploited to their full extent.

Through the Internet, new tools for global collaboration and data sharing in a global marketplace can be downloaded. Successful companies will get the right information and tools to the right person at the right time, regardless of where the person is located. In fact, through the Internet interactions that previously never took place, are now possible.

The Internet supports clients with on-line information 24 hours a day. Companies working within a collaborative engineering environment on the Internet, can reach prospective clients around the world by describing their organisation's capabilities and providing a way whereby investors can easily show their interest or place orders. The time zone differences are not a problem because web sites would always be available for potential clients. Furthermore, companies can rapidly update their clients on the changes that are taking place in their spheres, introduce new products and services and announce price changes and special offers quickly at relatively low cost. Also, winning orders at web sites can reduce sale costs.

Recent times have seen a huge growth not only in the number of users found on the World Wide Web, but also the number of companies providing Internet services and establishing sites via the purchase of dedicated Web-servers, (Yang and Keiser, 1996), (Berners-Lee and Cailliau, 1990).

The Internet has significantly enhanced the interaction between companies and their respective customers, suppliers and partners, acting as an important platform to deploy open and distributed manufacturing solutions (Park *et al.*, 1993).

This exceptional growth (Baentsch *et al,* 1996) has opened exciting opportunities to businesses by providing another way of reaching potential customers. Pant and Hsu (1996) affirm that the use of the Internet as a business tool may have the same effect on businesses as the rapid spread of personal computers during the 1980's.

The popularity of Internet-based applications, the functionality of which is usually supported by database operations, is growing considerably, increasing the applicability of Java-based development environments (Tian *et al,* 1997). The different applications are programmed using Java language and published in a Web-Server, where they are accessed from remote locations.

In the next section, issues of connectivity and database access through the Internet are addressed.

2.4 DATABASE ACCESS AND CONNECTIVITY ISSUES

The development of distributed manufacturing solutions using the Internet, as was indicated in section 2.2, involves the successful resolution of technical issues as connectivity and database management. In this section, these important topics are examined.

2.4.1 Connectivity and Application Programming Interfaces (APIs)

Client-server computing has traditionally been undertaken using *sockets*. A Socket provides a two-way connection between programs running on different systems on the Internet.

In addition to the previous option (sockets), Java has several Application Programming Interfaces (APIs) such as RMI (Remote Method Invocation), CORBA (Common Object Request Broker Architecture) and JDBC (Java Database Connectivity), that provide better facilities for communicating with distributed stand-alone applications and SQL compatible relational Databases. These APIs are examined as follows.

a) *Remote Method Invocation (RMI)*

RMI is an API that can be used to access the methods (or functions) of remote objects. Usually, three programs must be created in order to build applications using *RMI*:

- An interface application that defines the operations/services that will be available from the object.
- A Java application, running on the server computer, that registers one or more Java objects.
- A Java application, running on the client computer, that accesses the methods of the objects of the application running on the server.

b) *Common Object Request Broker Architecture (CORBA)*

CORBA has been created to define interfaces for interoperable software. When CORBA is used, communication between two computers is carried out by implementing an Object Request Broker (ORB). An object in a program running on a computer (the client) can use the ORB to access the public attributes of another object in some other program, perhaps on a different computer (the server) that is also using the ORB.

Cornelius (1998) established a comparative analysis between *RMI* and *CORBA*, according to the kind of interfaces and programs that must be created. With

RMI, the programmer has to provide the interface in Java, whereas with CORBA it has to be provided in an Interface Definition Language (IDL). IDL is the language used in CORBA to describe the interface associated with an object. Also, with *RMI*, both the client and the server have to be Java applications/applets running in some Java environment. However, with *CORBA*, the client and server programs may be in different programming languages.

As it was described for *RMI*, at least three programs must be created to build applications using JavaIDL:

- An interface application that defines the operations/services that will be available from the object.
- A Java application running on the server computer that allows creating a Java object and the needed relationships, in order to link both objects.
- A Java application running on the client computer that accesses the methods of the object of the application running on the server.

c) Java Database Connectivity (JDBC)

JDBC is a Java API for executing SQL statements. It consists of a set of classes and interfaces written in Java. JDBC provides a standard API for database developers and makes it possible to write database applications using a pure Java API.

Using JDBC it is possible to write a program to access different relational databases without having to re-write a different program for each one of them. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database.

JDBC extends what can be done in Java. For example, with Java and the JDBC API, it is possible to publish a web page containing an applet that uses information obtained from a remote database. Alternatively, an enterprise can use JDBC to connect all its employees (even if they are using a conglomeration of Windows, Macintosh, and Unix machines) to one or more internal databases via an Intranet (Hamilton *et al.*, 1997).

JDBC makes it possible to do three things:

- Establish a connection with a Database.
- Send SQL statements.
- Process the results.

2.4.2 Connectivity, Database Access and Drivers

In 1990 the SQL Access Group defined the Call Level Interface (CLI) as a standard for accessing databases. To implement CLI, one needs a connector (commonly named driver) that can translate a CLI call into the language used to access a particular database. For example, Open Database Connectivity (ODBC) is an API for Microsoft Windows that implements an extended version of CLI.

One important characteristic of the Internet is that the information is distributed in different world servers. There are different types of drivers to access databases using the Internet. Currently, there exist four driver categories (Hamilton *et al.*, 1997) shown in Table 2.1.

Table 2.1 - Driver categories for database access using JDBC.

Driver Category	Pure Java	Needs to load code on user machine
1.- JDBC-ODBC Bridge.	No	Yes
2.- Native API as basis.	No	Yes
3.- JDBC-Net.	Yes	No
4.- Native protocol as basis.	Yes	No

1. - JDBC-ODBC Bridge driver: This provides JDBC access via ODBC driver. The driver requires prior installation of client software on each user's computer.

2. - Native-API partly-Java driver: This kind of driver converts JDBC calls into calls on the client API for a range of DBMSs. As the category 1, the driver also requires software installation on user's computer.

3. - JDBC-Net pure Java driver: This driver translates JDBC calls into a DBMS-independent net protocol, which is then translated to a DBMS protocol by a server. The specific protocol used depends on the vendor. The driver does not need software installation on the user's machine.

4. - Native-protocol pure Java driver. This kind of driver converts JDBC calls directly into the network protocol used by DBMSs. This allows a direct call from the client machine to the DBMS server and is an excellent solution for Intranet access.

2.4.3 Common Gateway Interface versus Java-based Approach

In relation to database access, Revere (2000) carried out a comparative study between traditional CGI (Common Gateway Interface) and Java-based architectures.

CGI was mainly developed to provide a generic interface between an HTTP server and server applications being run by a Web user. Historically, CGI programs used to be the only option available to provide database access on the Web, hence, it became the adopted standard for establishing a link between HTTP servers and external applications (Duan, 1996).

A typical database application on the Web consists of three main components. These are the Web browser, commonly referred to as a Web client, an HTTP server with a CGI program and a database server.

The user generating a request initiates the whole process, being the information provided by the user normally contained within HTML forms. Once the request is initiated, the query will be sent to the HTTP server thereby invoking the CGI program, which is resident on that HTTP server. The CGI program converts the information contained within the HTML form to a specific database query and submits this query to the database for processing. Once the database server has processed the query the results will be returned to the CGI program, and finally passed to the Web client through the HTTP server.

Although CGI applications are widespread and the concept of CGI is relatively simple, the overall architecture of CGI suffers from a number of drawbacks that can be especially significant in an era when Web traffic is showing no signs of decreasing.

The first problem is that the use of CGI on an HTTP server means the possibility of direct SQL submission to the database is removed. This is to say that communication between the Web client and the database server must always go through the HTTP server. In times of busy traffic it is possible to have a significant bottleneck in the overall process, as the HTTP server has to convert every user request to an SQL query via the CGI script and then, convert the data back to HTML format before transmitting it to the user.

Furthermore, in his analysis of CGI programs, Duan (1996) identifies a second difficulty also created during times of busy traffic, but originating from a lack of efficiency in a CGI based database access script. Implementations of logon and logoff procedures take up system resources and in times of heavy traffic, the resources used could be significant.

Now let us consider the architecture associated with the use of Java to provide remote database access to a Web client. Figure 2.1 shows the same three components that were discussed for CGI based transactions, but the way in which Java links these components is fundamentally different, as the HTTP server no longer acts as a stepping stone between the Web client and the Database server.

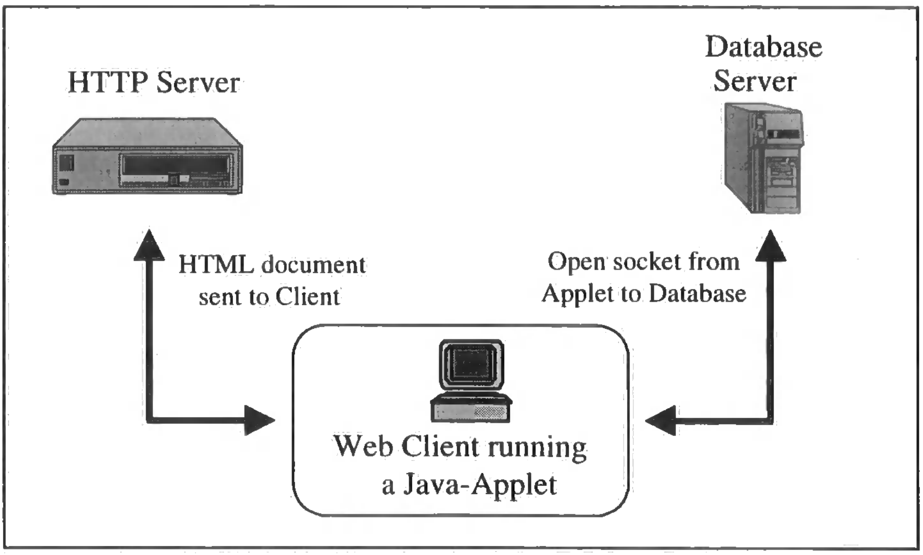

HTTP Server
Database Server
HTML document sent to Client
Open socket from Applet to Database
Web Client running a Java-Applet

Figure 2.1 - Simple Java-based Architecture

Once the Java applet has been initiated on the Web-client, it has the ability to make its own connection to the database server via the use of sockets (Cornelius, 1997). It is possible for a Java applet to communicate directly with the database, thus eliminating the bottleneck that CGI imposes on the HTTP server. In addition, an applet has the ability to provide session-oriented communications with the database. This is to say that once an applet connects to a database, the connection can be kept open as long as the applet is alive and the user is in session. As a result, interactive queries and multiple database transactions can be supported (an advance on CGI), which closes a connection as soon as an individual query has been processed irrespective of whether another query is queuing to be sent by the same user.

Not only does an applet have the ability to communicate directly with a database server, but it also has a full set of drawing functions made available through the AWT (Abstract Window Toolkit). Hence Java can handle sophisticated graphics and provide a comprehensive distributed computing arrangement integrated into an object-oriented environment. Having overcome problems encountered with CGI, Java offers impressive scope for industries wishing to deploy interactive Web applications.

2.5 AN APPROACH TO SUPPORT REMOTE INFORMATION EXCHANGE IN CORPORATE ENVIRONMENTS

Computer network platforms constitute one of the most used information resources to support distributed business solutions, the Internet being presently, the most popular and available open platform.

The concept of deploying distributed solutions does not necessarily apply in areas too remote or geographically alienated. However, where companies like large corporations are considered, with widespread branches and world-wide client-portfolio, it is eminently suitable to rely on a distributed infrastructure.

Smith and Wright (1998), argue that two of the fundamental changes occurring in the Web is the increasing importance of **i)** *Distributed Computing* and **ii)** *Client-Side Processing.*

i) Distributed Computing means that instead of having all of the computation tasks taking place on the user's desk, the user's computer sends data to one or more remote machines, which return information that is then displayed on the local machine. With more of this remote processing, it is becoming possible to run software normally reserved for workstation level platforms, on relatively low-end systems that have a good link to the Internet.

ii) Client-Side or Browser-Side Processing represents an important contribution in distributed manufacturing environments, because it allows the server to give the client's browser some of the responsibility for processing data. However, Client-side processing is not common on the Web at present.

Some important benefits can be obtained when processes are carried out using a distributed approach:

- Efficient access to resources over a geographically dispersed area.
- Cheaper information exchange processes.
- Closer interaction between Clients and Companies.
- Major support to assimilate the company growths.
- Improved distribution of software and hardware resources.

In the last decade, many papers have been written about the increasing impact of distributed solutions and global enterprise, focusing mainly on *models* (Zhao *et al.*, 1999; Hamel and Wainwright, 1999) and *organisational* problems (Giachetti, 1999). Although these papers provide significant theoretical contributions, there is an urgent need to address technical issues to overcome the problems imposed by real applications. In addition, studies in matters of standardisation, collaborative work and efficient remote information exchange lack the required level of integration and industrial applicability to be considered significant advances, and the results often have been overestimated.

Despite the limited number of current Web-based industrial applications, some work has provided important theoretical and practical contributions. Two good examples are the systems developed by Brown & Wright (1999) and Chang & Lu (1999).

An architecture to support remote information exchange in corporate environments is proposed as follows.

2.5.1 *ARIEX* Architecture

In order to provide an efficient infrastructure to manage information in distributed environments, an ***Architecture for Remote Information Exchange*** (*ARIEX*) is proposed (Velásquez and Velásquez (b), 2000) and shown in Figure 2.2. It is evident how the definition of connectivity functions as well as database technology considerations, play a crucial role for performing manufacture support operations using the Internet.

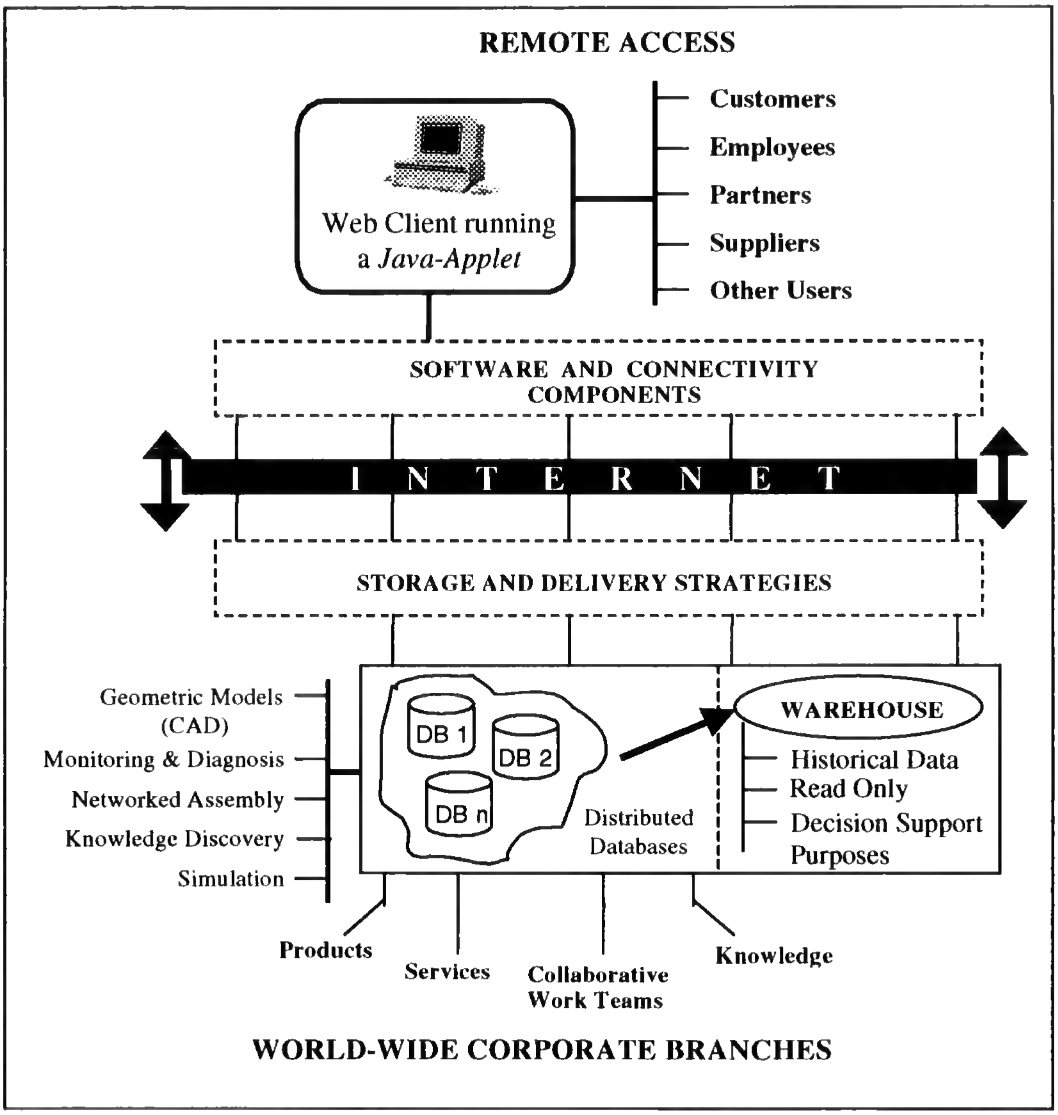


Figure 2.2 - ARIEX Architecture.

The architecture relies on integration of corporate information, distributed on databases having the same internal structure but different data, along geographically dispersed branches. The convenience of sharing information of mutual interest to internal users (employees and partners) as well as external agents (suppliers and customers) working in a platform-independent architecture and controlling data security aspects, constitutes its main advantage.

ARIEX considers distributed access and centralised data management for those industries having a common platform or a low number of interconnected branches, providing a flexible work scheme.

The last approach (centralised data management) would eliminate the problems associated with fragmented databases requiring regular updating whilst also allowing distributed access for effective remote updating processes.

However, for Internet-based applications, a centralised approach imposes some problems and becomes a bottleneck, particularly in relation to the heavy traffic of transactions generated when a unique centralised information repository is accessed. For instance, when the communication with the central database cannot be established, the users are unable to take benefit of the data.

The connectivity aspect is covered by the utilisation of 100% pure Java-drivers, hence, the problem of asking the users to download and configure the driver is eliminated.

All users have the option to establish a link to the required sites and databases of interest, accessing the data from remote locations using conventional browsers in the case of *Java-applets*, or executing *Java-applications*, otherwise.

Although the data and Web services can co-exist in the same server machine, the best results are reached when there are different servers working to deploy Web and database services separately. In this way, the data are stored in a database server and all the net services are the responsibility of an exclusive Web server, establishing a load balance between the data and Web access management.

2.5.1.1 Functionality of ARIEX

The architecture proposed here considers a corporation having branches in widely dispersed places. In this context, *ARIEX* focuses on three main functions, which are discussed as follows.

a) Information sharing on free-access platforms.

This philosophy is ideal when companies wish to share information about products, services and operations with their customers or when employees of the same company in remote branch locations need to exchange information. It is an open solution, because there are no restrictions about the information that is going to be accessed.

Remote users can interact with the system entering input requirements and obtaining answers to their queries. The users only need any available commercial browser to download Java-applets. Also, all Java classes would be stored in the local user machines, to speed up the download operations.

The implementation of Internet-based strategies for delivering and exchanging information about product and services to widely distributed customers, will overcome the well known problems in relation to the use of conventional representation (paper-based) and distribution (regular mail) mechanisms.

The financial implications are clearly favourable. For example, the savings in costs would be considerable, particularly when massive amounts of technical catalogues and information about products and services, have to be sent to remote customers. Further benefits include: instantaneous updates, better visualisation facilities (3-D and multimedia) and a higher participation of market.

b) Collaborative work strategies.

Other benefits arise when this infrastructure is utilised for allowing distributed collaborative work, especially in concurrent engineering environments where CAD/CAM activities are carried out.

Geometric modelling, monitoring and diagnosis and networked assembly are some useful functions that would be implemented. In a product-manufacturing scenario, once the geometric model is designed, it will be sent to a geometric file translation server to perform file conversion. A collaborative module would then provide all participants a virtual collaborative environment to view the converted VRML[2] created, and to perform communication via the Internet.

Finally, after performing collaboration, the designer sends the new design part to a manufacturing module for process planning (Chang & Lu, 1999).

In this kind of collaborative work environment, the emergence of Internet-enabled CAD browsers is expected, to improve the current capabilities of conventional browsers, particularly in relation to efficient 3-dimensional object manipulation and representation.

c) Knowledge Discovery.

Furthermore, within the business goals of the company, *ARIEX* considers the implementation of knowledge capturing systems in an attempt to discover previously unseen relationships within the data, an expanding and relatively promising new area known as *Knowledge Discovery in Databases* (see chapter 4).

2.5.1.2 Managing Different Security Levels

In relation to corporations wishing to manage public (free-access) and private (restricted access) information simultaneously, Figure 2.3 shows an example of a complementary approach based on a tooling company model (Revere, 2000). It can be noted how is possible to combine strategies to allow the access of classified and public information sources.

For explanation purposes, let us assume that a tool manufacturer company (company 'A') has a repository of tooling data spread over different storage sources. In this particular case, two different databases would be created with access being made possible by the development of interactive utilities that could be executed via World Wide Web browsers.

[2] VRML, Virtual Reality Modelling Language, is the industrial standard of non-proprietary file format for displaying scenes consisting of three-dimensional objects on the World Wide Web (Ames, Nadeau & Moreland, 1997).

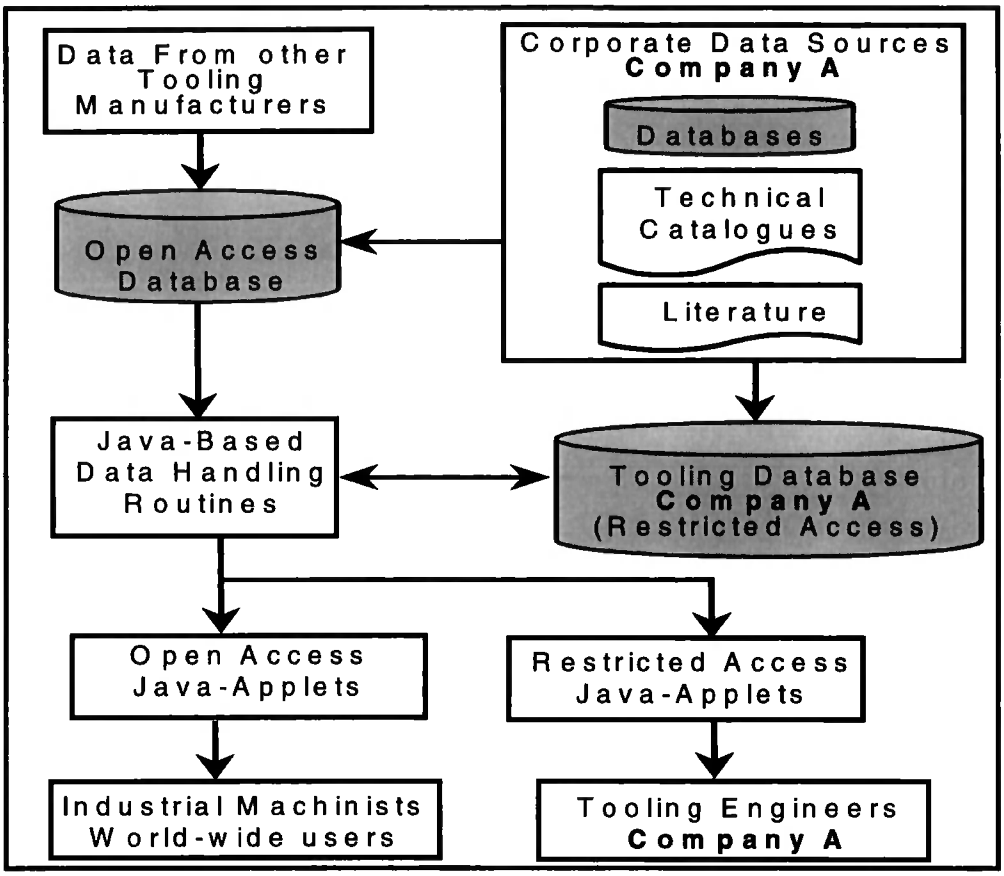

Data From other Tooling Manufacturers
Corporate Data Sources
Company A
Databases
Technical Catalogues
Literature
Open Access Database
Java-Based Data Handling Routines
Tooling Database Company A (Restricted Access)
Open Access Java-Applets
Restricted Access Java-Applets
Industrial Machinists World-wide users
Tooling Engineers Company A

Figure 2.3 - Complementary approach. Access of public and classified information sources.

One of these databases would contain public information about the products and services offered by the company, and the other one, tooling technical trials to be accessed only by authorised tooling engineers of the company. Generic technical support in the form of tool and cutting data selection is another function that can be provided by an open-access area. Considering the incorporation of security features suitable for restricting the access on the World Wide Web, tooling engineers (*authorised* users) would take advantage of this infrastructure to submit and retrieve highly specific technical tooling information. Once users introduce their ID and encrypted passwords, a level of permission is retrieved. If the users are registered, they can access the information corresponding to these access levels. When the user is not registered, only non restricted modules should be activated (free-access).

ARIEX has been successfully implemented on two Internet-based systems, which will be described in the next chapter.

2.6 SUMMARY

This chapter has mainly concentrated on the investigation of the main issues in relation to the implementation of Web-based strategies to support information integration in corporate environments.

In order to implement effective Web-based strategies to support distributed manufacturing solutions important considerations were addressed. *Information security, database technology and ability to manage knowledge* were identified as topics requiring special attention.

It has been shown how the WWW has the potential to provide a global communication environment, where geographical distances, cultures, languages and time zones are no longer a significant factor for consideration in commercial operations.

Furthermore, technical issues of connectivity and database access have been addressed, and, within these considerations, an Internet-based ***A**rchitecture for **R**emote **I**nformation **Ex**change* (*ARIEX*) has been proposed.

ARIEX is a flexible architecture, considering distributed access and centralised data management for those industries having a common platform or a low number of interconnected branches.

The developments in the area of Web-based distributed strategies have been and will continue to be fast-paced and exciting. Due to the Internet environments represent, in essence, heterogeneity, advanced capabilities as CAD-enabled browsers and a major implementation of parallel processing, are two important issues that will be strongly addressed by the next generation of Web-based tools.

In the next chapter, it will be shown how *ARIEX* has been successfully implemented in two different systems, to provide distributed solutions to the manufacturing tooling industry.

CHAPTER 3

DEPLOYING INTERNET-BASED SOLUTIONS

3.1 INTRODUCTION

In the previous chapter, Web-based strategies to support collaborative work in distributed environments were examined. In this chapter, two different Internet-based systems, ***TTS***[3] and ***SELTOOL***[4]***,*** to provide solutions in the tooling area for the company Seco Tools (UK) Ltd (Seco, henceforward), are presented.

TTS was developed in collaboration by two another colleagues. It will show how it is possible to use a shared-information platform to access a nation-wide source of tooling knowledge, whilst keeping a restricted access policy. On the other hand, ***SELTOOL*** will be primarily focused to provided distributed solutions in the area of tool selection, considering the implementation of a free-access architecture.

This chapter presents the technical aspects and distributed functionality of both systems.

A third Internet-based system named ***DISKOVER***[5], which implements an integrated set of knowledge discovery applications has also been developed in collaboration with another colleague. However, a full explanation of this last system is left to Chapter 7, after the theoretical foundations of the knowledge discovery area are discussed.

[3] Developed by the author and two other researchers in the group at Durham.

[4] Developed by the author.

[5] Developed by the author and another research student in the group at Durham.

3.2 SYSTEM 1: TOOL TRIALS SYSTEM (TTS)

A system named ***Tool Trials System (TTS)***, which is capable of collating and disseminating information relating to tool trials amongst a variety of user groups, has been developed (Velásquez and Velásquez (d), 2000). ***TTS*** has provided a World Wide Web platform from which tooling engineers (*authorised* users) can submit and retrieve highly specific technical tooling data for both milling and turning operations.

This work demonstrated not only the suitability of the Internet as a distributed computing resource, but more importantly, it was possible to look at the ways in which approved data could be analysed and then applied to cutting data selection within the Process Planning arena. ***TTS*** has been developed under a distributed philosophy and it can be downloaded by remote users in the form of *Java-applets*, through any computer with Internet connection and using conventional Java enabled browsers.

Because of the dimensions and the corporate nature of ***TTS***, three researchers were assigned to develop this project. The main contribution of the author was to provide an Internet-based framework to support the distributed nature of the proposed solutions. Two different tasks were carried out. Firstly, the selection of an appropriate strategy for sharing information in a distributed environment using the Internet and secondly, the definition and implementation of suitable methods to allow the access to authorised users only (restricted access policy).

The two formerly mentioned tasks were analysed taking into consideration the access of geographically dispersed databases and the interest of Seco for relying in a tooling data repository, accessible from world-wide authorised users. These tasks are described as follows.

3.2.1 Considerations for Developing *TTS*

This section examines important factors considered when developing and implementing ***TTS***. At the end of the section, a discussion on why certain methods and strategies were selected, is presented.

3.2.1.1 The Programming Language

The utilisation of a robust programming language constitutes a key factor in the development of a distributed system. Because of its features, Java has become a well-known programming language in recent years, in support of the development of Internet-based solutions. The reason being that Java programs are compiled into a platform-independent format called *Java byte code*. This *byte code* is designated to be executed by a *virtual machine* called the Java VM. Therefore, any platform that has an implementation of the Java VM can interpret byte code and run Java programs. A Java VM is often called a *Java Interpreter*, since its main task is to interpret Java code.

Because of its independent-platform quality, many Java development platforms have been created to aid in the programming of systems that have to run on distributed environments.

3.2.1.2 Java-Applications versus Java-Applets

Currently, there are two kinds of Java programs, called *Applications* and *Applets*, that can be created to develop systems able to run on Internet environment, allowing retrieval and submission of database operations. Figure 3.1 shows the differences between both approaches running on Internet platforms.

Java-applications are programs oriented to provide stand-alone solutions running in a local client-computer, as well as distributed solutions using a suitable API to establish communication and database access through the Internet. As shown in Figure 3.1-a, when the application is developed to run on an Internet environment, all the classes, images and Java files need to be stored in the client-computer. Further to this, the operating system used by every remote client-computer must be the same.

In contrast to *Java-applications*, an *Applet* is a Java program that can be executed through a Web browser. When the *applet* runs within a Web page (embedded within an HTML file), it displays a form where a remote user can access and enter any desired information, subject to some security restrictions.

In order to deal with the security aspects that the Web imposes, *Java-applets* have been designed after considering the following key restrictions:

- *Applets* cannot execute any other application on the user's system.

- They cannot have access to any storage on the client-computers; for example, they can not 'peek' at their files or delete the contents of their hard disks.

- *Applets* cannot make network connections to any machine other than the system containing the original Web page (the Web page that invoked the *applet*).

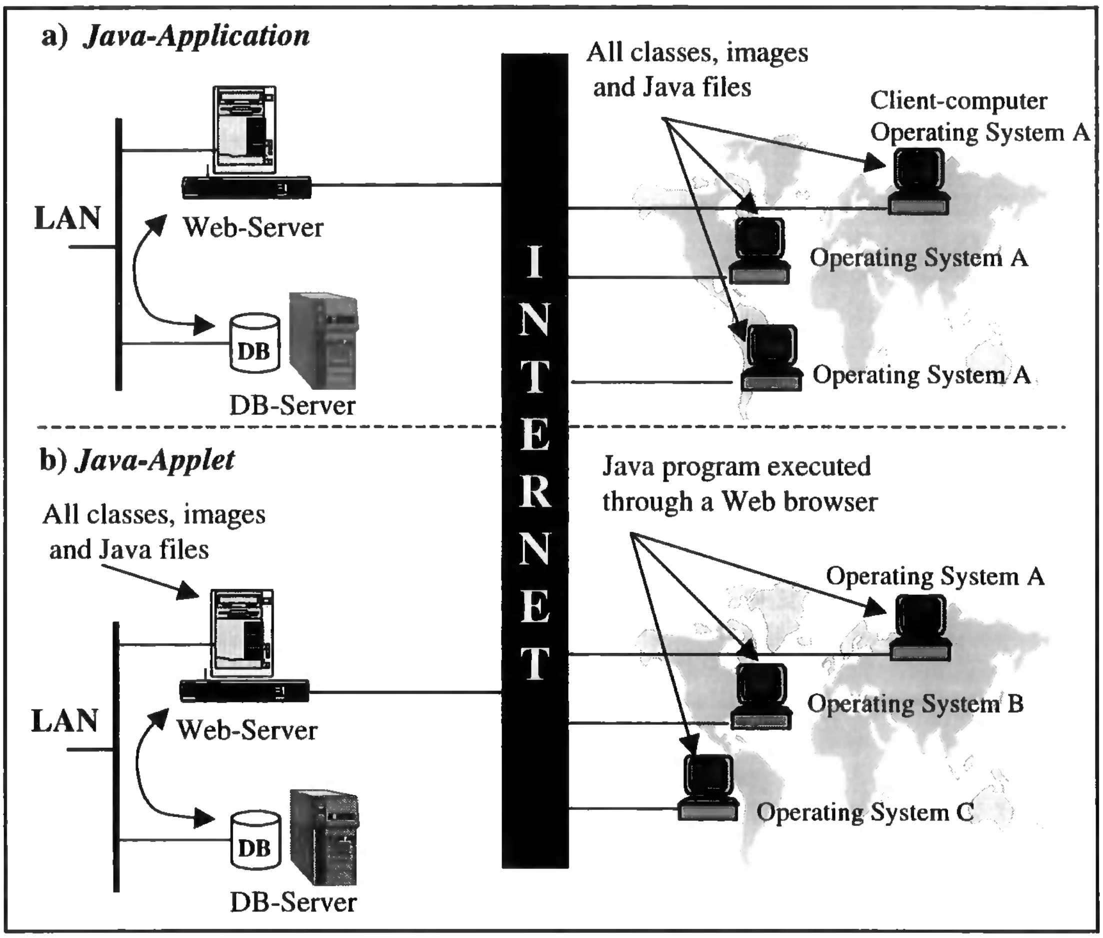


Figure 3.1 - Java-application and Java-applet functionality

All the classes, images and Java files are stored in the Web-server and due to the independent-platform feature of *applets*, the operating system of remote client-computers, do not need to be the same.

In both cases (implementing *applications* or *applets*) the database can reside in the Web-server, but the ideal situation is when the database resides in a DB-server, thus allowing the Web-server to take charge of the usually heavy information traffic in the WWW.

3.2.1.3 Data Replication Approach versus Real-Time Access

Two different approaches for accessing and updating data from physically remote locations are examined.

a) *Data Replication*

Data Replication (DR) consists in sharing data among physically remote databases, but having the same structure and using the same DBMS. Changes made to share data at any one database are replicated to the other databases when the user connects to a central database that is updated from the different locations each time the connection is established.

Some benefits of applying DR (Sybase, 1997) are explained as follows.

- *Data availability at any time.* One of the key benefits of a data replication system is that data is made available locally, rather than through potentially expensive, less reliable, and slow connections to a single central database. Data is accessible locally even in the absence of any connection to a central server, so that the program is not cut off from data in the event of a failure of a long-distance network connection.

- *Good performance.* Replication improves response times for data requests because they are processed on a local server without accessing some wide area network. This makes the transfer rates faster.

- *Integrity of the data.* One of the challenges of any replication system is to ensure that each database retains data integrity at all times. Transactions are replicated atomically: either a whole transaction is replicated, or none of it is replicated.

- *Consistency of the data.* All changes are replicated to each site over time in a consistent manner, but because of the time lag, different sites may have different copies of data at any moment.

b) *Real-time access*

Using a real-time access approach, remote users can download a Java-applet in order to access a central database through the Internet. In this case, the updating process is carried out directly into the database at the same time as the user is making these changes, because they occur on-line.

The advantages of this approach are summarised as follows.

- *Platform-independent architecture.* Users could access the database regardless of the differences between the operating system running in their local computers and the operating system where the database resides (Web server or DB-server), because of the use of *Java-applets.*
- *Higher data security.* The data is centralised in a DB-server and not locally on the client-side. This means, monitoring operations can be implemented to register what kind of database operation is performed and which user was involved in this transaction.
- *Saving storage resources.* Large databases can occupy costly space in the local client-computers. Implementing a real-time access approach, the database does not need to be stored locally.
- *On-line updating.* All the changes carried out by remote users connected to the database are instantly updated.

*3.2.1.4 Defining a suitable configuration for **TTS***

Before discussing the reasons for choosing some of the alternatives formerly presented, instead of others, let us first examine the framework where these alternatives were considered.

Consideration of the scheme displayed in Figure 3.2 reveals the corporate and distributed nature of the solution suggested. Under this approach, Seco sales engineers using their respective laptops would access ***TTS*** through the Internet, regardless of their physical location.

The benefits to the tooling manufacturer are, that the tooling engineers, working in a collaborative information environment, would be able to share a nation-wide database of knowledge, created from their previous work.

Moreover, it should be possible for tooling engineers to avoid the execution of new tool trials, when another engineer has previously carried out these trials. In this case the engineer would execute a query, entering his/her parameters of interest and the system would provide result sets, introduced in the past, matching the same input parameters specified.

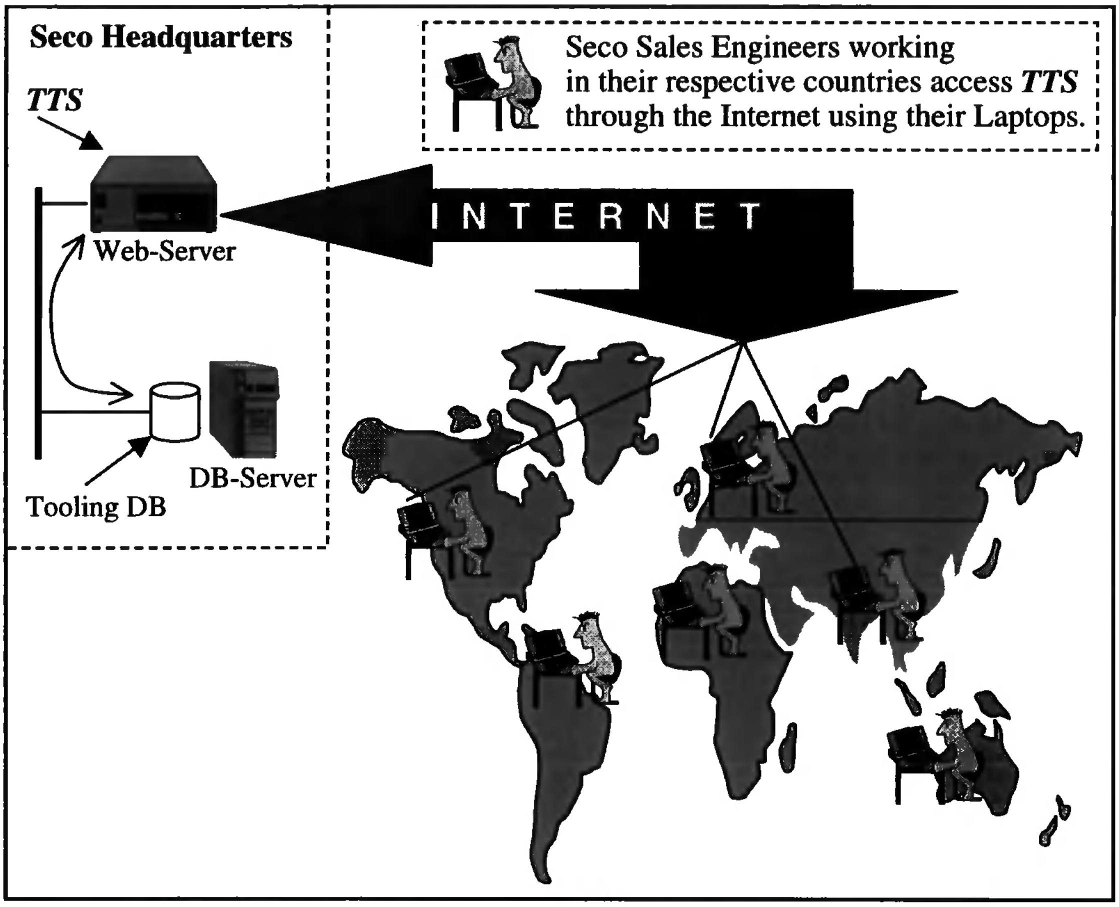

Seco Headquarters
TTS
Web-Server
DB-Server
Tooling DB
Seco Sales Engineers working in their respective countries access TTS through the Internet using their Laptops.
INTERNET

Figure 3.2 - Distributed solution and its different components

Following a decision on the adoption of a global architecture, it was necessary to decide upon the most suitable methods and options that would be implemented.

Returning to the explanation in sub-section *3.2.1.2*, it should be noted that *applets* can also be configured to run as stand-alone applications, within certain limitations, as the fact that they still need to be downloaded through HTML files. When using *applets*, the operating system of remote client-computers do not necessarily need to be the same, because the concept of independent computational platform associated with Java language is fully exploited. This advantage would support the information update processes from remote locations, where heterogeneous computational platforms are utilised. It was therefore decided that *applets* would be implemented instead *applications.*

After due consideration and discussion it was decided that Java Database Connectivity (JDBC) would be an Application Programming Interface (API) suitable for submitting SQL statements to a DB-server containing a central tooling database. JDBC was chosen because it allows rising above the socket-level aspects of distributed computer solutions. Also, JDBC supports the utilisation of 100% pure Java drivers, which eliminates the problem of asking the users to download and configure the driver.

In this research, the use of Data Replication (DR) technology was considered, mainly because remote users can access their databases locally and only when they need to send new information to the central database (and receive the latest updates), a Web connection needs to be established.

However, due to the high level of confidentiality of the information generated by the tool trials, it was decided to keep a central database accessed remotely through *Java-applets*, adopting a real-time access approach instead of keeping a copy of the database stored in each client-computer, as required by a DR approach.

To deal with the security aspects in relation to the access of the system, several procedures were implemented, including the creation of encrypted passwords and monitoring functions to register the database transactions.

Whilst this concludes the reasons for implementing a real-time access approach supported by *Java-applets*, instead of DR techniques, this is not to say that there are not other disadvantages when using real-time updating operations. For example, when the Internet connection cannot be established, it is not possible for the users to access the database, which does not occur using DR, because in this case, the database is also stored locally.

Another problem faced by users when downloading applets, is their relatively slow speed because they are downloaded through HTML files. This problem becomes critical during hours of heavy traffic on the Internet.

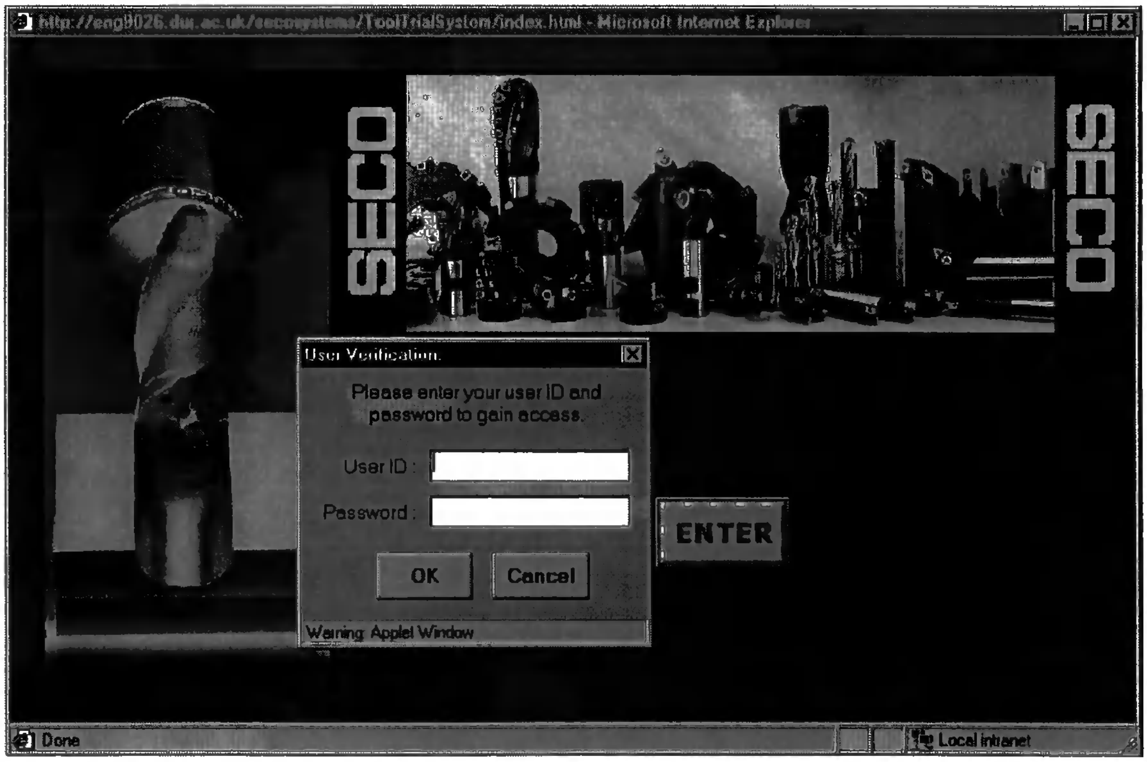
3.2.2 Functionality of *TTS*

This section presents a summary of functional aspects of ***TTS***. A detailed analysis of all its operational issues will overwhelm the scope of this chapter, so, for practical reasons only some selected input and output functions are examined.

Based upon the easily expandable feature of ***TTS*** and the number of possible applications that would be developed using the tooling database, four levels of permission were defined, which will provide the users with the authorisation levels for using these applications.

These four levels go from the less restrictive to the most confidential and were named as *open user*, *medium user*, *high level user* and *super user*.

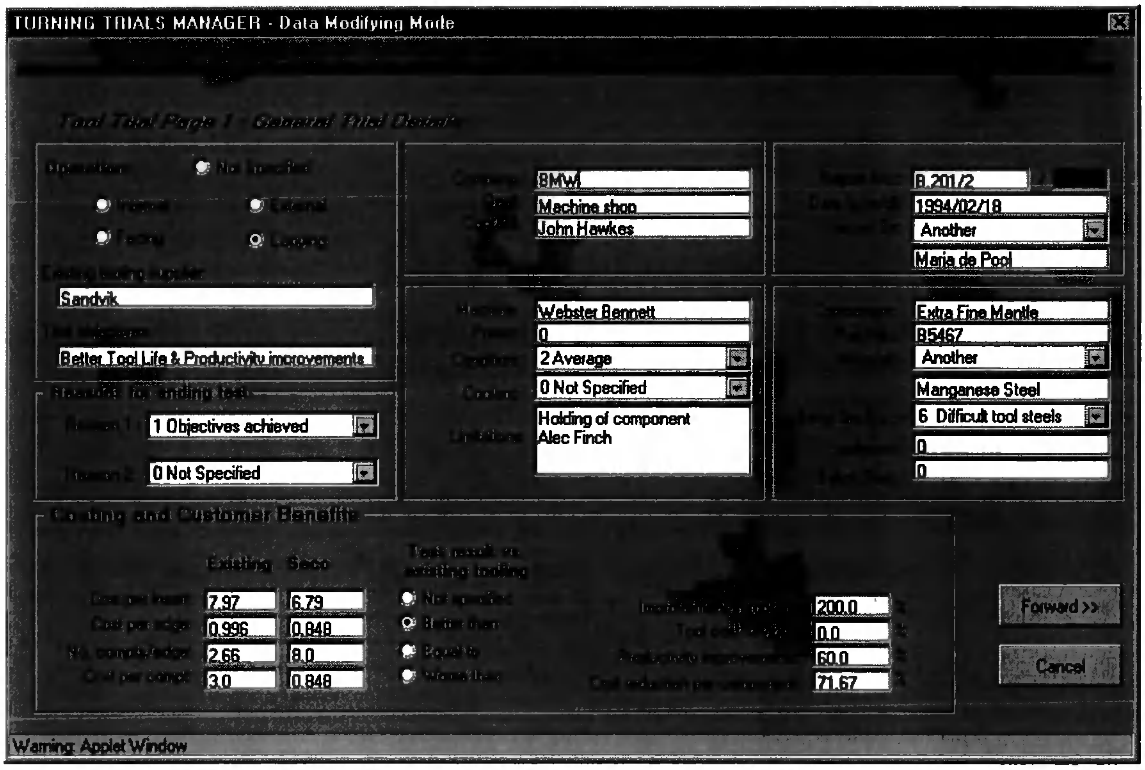
When ***TTS*** is accessed, an initial security screen becomes visible, as shown in Figure 3.3. Two fields, *User ID* and *Password*, must be entered to allow the system to check the authenticity of the user by searching for the corresponding record in the database. When the record is found, the verification of the level of authorisation of the user is carried out. Once this access level is recognised, the general functions of the system are shown.



Please enter your user ID and password to gain access.

Figure 3.3 – Initial screen of TTS

In order to incorporate a new report of tool trails to the database, ***TTS*** provides a versatile input interface having the same appearance, as does the original paper-based form. Figure 3.4 shows this screen.



TURNING TRIALS MANAGER - Data Modifying Mode

Tool Trial Page 1 - General Trial Details

Existing tooling supplier: Sandvik
Test objectives: Better Tool Life & Productivity improvements

Reasons for ending test
Reason 1: 1 Objectives achieved
Reason 2: 0 Not Specified

BMW
Machine shop
John Hawkes

Webster Bennett
0
Condition: 2 Average
Coolant: 0 Not Specified
Limitations: Holding of component
Alec Finch

B.201/2
1994/02/18
Another
Maria de Pool

Extra Fine Mantle
B5467
Another
Manganese Steel
6 Difficult tool steels
0
0

Costing and Customer Benefits

	Existing	Seco
Cost per insert	7.97	6.79
Cost per edge	0.996	0.848
No. compts/edge	2.66	8.0
Cost per compt	3.0	0.848

Test result vs. existing tooling
○ Not specified
● Better than
○ Equal to
○ Worse than

200.0 %
Tool cost saving: 0.0 %
Productivity improvement: 60.0 %
Cost reduction per component: 71.67 %

Warning: Applet Window

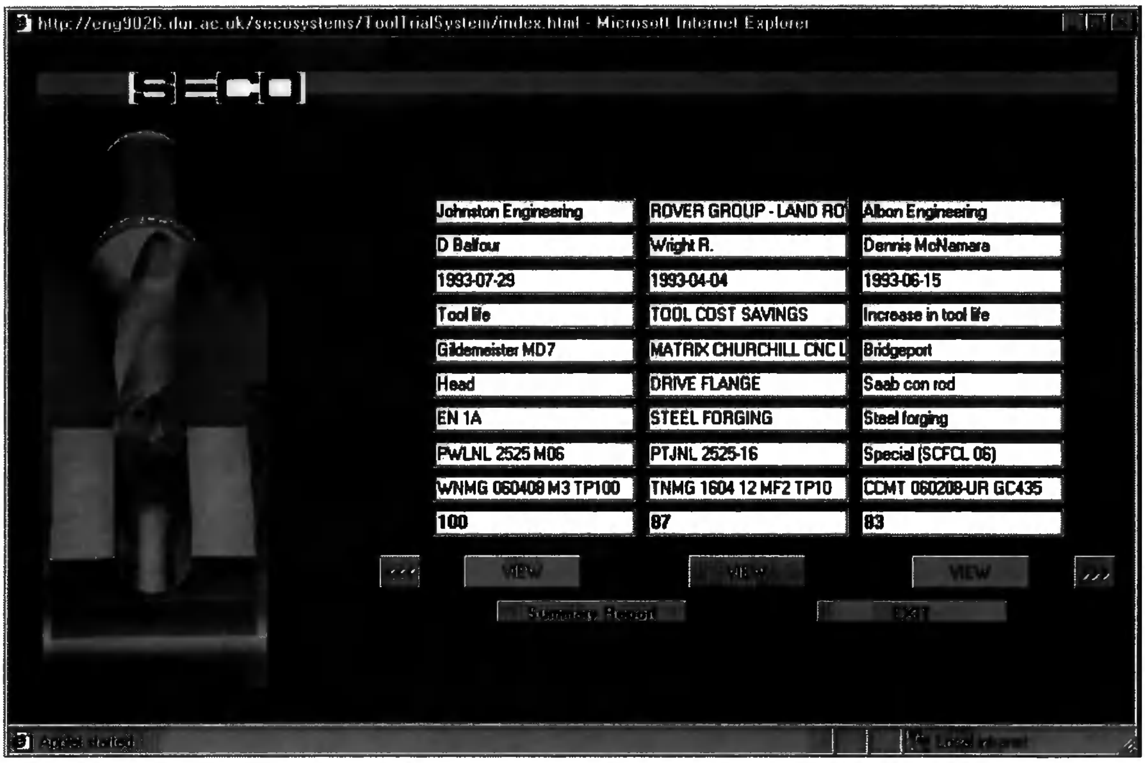
Figure 3.4 – Data entering screen for tool trials

Once the engineer has performed an initial test according to the tooling parameters required, it is registered as an existing test. Next, the engineer performs alternative tests and obtains comparative results, which must also be registered in the system.

When all the values are added to the forms, the user will finally submit the report to the database.

Regarding output functionality, Figure 3.5 shows the screen displayed, where the reports and their respective confidence scores are presented.

The user can either view a specific report or select the "Summary Report" option. The last alternative displays some statistical values calculated with the data obtained from the query. Cutting parameters constitute the variables considered for generating this report.


http://eng9026.dur.ac.uk/secosystems/ToolTrialSystem/index.html - Microsoft Internet Explorer
<<<
VIEW
VIEW
VIEW
>>>
Summary Report
EXIT
Applet started
Local intranet

Johnston Engineering	ROVER GROUP - LAND RO	Albon Engineering
D Balfour	Wright R.	Dennis McNamara
1993-07-29	1993-04-04	1993-06-15
Tool life	TOOL COST SAVINGS	Increase in tool life
Gildemeister MD7	MATRIX CHURCHILL CNC L	Bridgeport
Head	DRIVE FLANGE	Saab con rod
EN 1A	STEEL FORGING	Steel forging
PWLNL 2525 M06	PTJNL 2525-16	Special (SCFCL 06)
WNMG 060408 M3 TP100	TNMG 1604 12 MF2 TP10	CCMT 060208-UR GC435
100	87	83

Figure 3.5 – A particular output screen provided by TTS

3.3 SYSTEM 2: *SELTOOL*

SELTOOL is an Internet-based tool selection system for turning operations (Velásquez *et al.* (c), 2000). It is able to deliver numerical and graphical information about suitable selections of inserts and toolholders for specific machining operations, workpiece material group and cutting type, and recommend the respective cutting data. ***SELTOOL*** covers three types of operations, namely, Turning, Grooving and Threading, each of them for external and internal cuts.

SELTOOL was implemented using a Java development environment. The information used to create the database was obtained from technical catalogues and proprietary tooling databases. Developed under an open and distributed philosophy, ***SELTOOL*** can be downloaded by remote users through any computer with Internet connection and using conventional Java enabled browsers.

The benefits arising for using this system are based in its world-wide access capability, which means an updated and cheap information delivery to users. ***SELTOOL*** has been tested with encouraging results.

The following sections present the justification and functionality of ***SELTOOL*** and chapter 8 discusses the execution of a test phase.

3.3.1 *SELTOOL*, Justification

The main reasons for developing the system described hereby were:

a) *Solution for the tooling industry.* Because of its extensive tool selection functionality and the numerical and graphical capabilities this system is useful for the tool machining industry.

b) *Evaluate the suitability for new implementations.* The system was developed for Seco as a pilot project in order to test the feasibility of future Internet-based tool selection methods.

c) *Availability of up-to-date tooling data and knowledge*: the technical tooling data and certain grouping criteria for the selection of specific tools is available from three sources: paper-based catalogues, PDF-based format (Compact Disc) and a database Access-based format.

d) *World-wide access*: ***SELTOOL*** allows remote and distributed access, cheaper information exchange processes and direct interaction between users and company products.

In the next section the appearance and functionality of ***SELTOOL*** will be presented.

3.3.2 *SELTOOL*, Functionality

The main functional features include:

a) Tool selection for external and internal Turning, Grooving and Threading operations.

b) The following range of insert information: 10 grades, 7 types of chipbreakers for external and internal Turning and common standard profiles for Threading.

c) Toolholder information for suitable inserts and approach angles.

d) 19 material groups.

e) 3 cutting types: finishing, medium roughing and roughing.

f) An easily expandable system, because of its modular structure.

g) The accessing and processing of distributed data (i.e., they are performed on the local computers from remote locations), while the storage of the information in the database is centralised.

h) The interface with the user being made through URL address in the way of an HTML document, which invokes the applet containing the operational structure. The system has been downloaded successfully using current versions of proprietary browsers.

i) An independent computational platform, because of using Java language and a 100% pure Java driver for database access. That means it is possible to download the system from any computer connected to the Internet, using Java-enabled browsers.

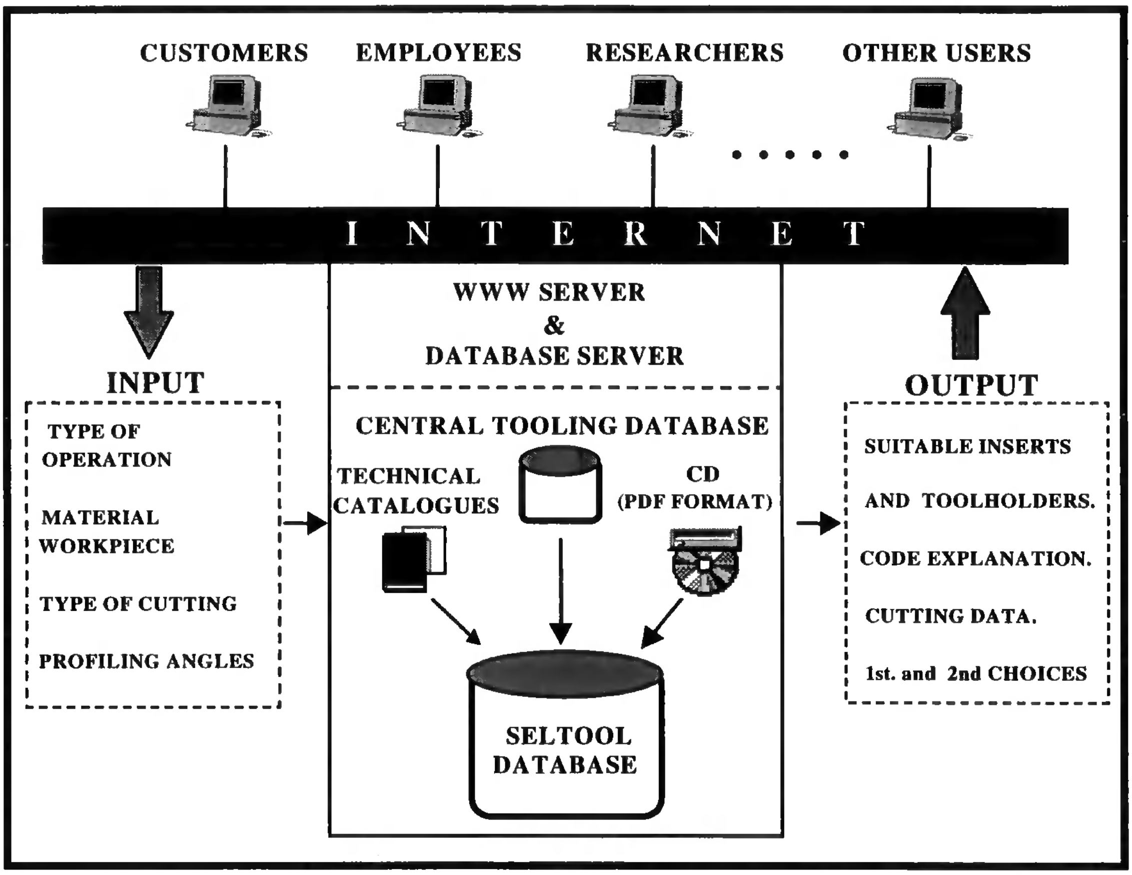


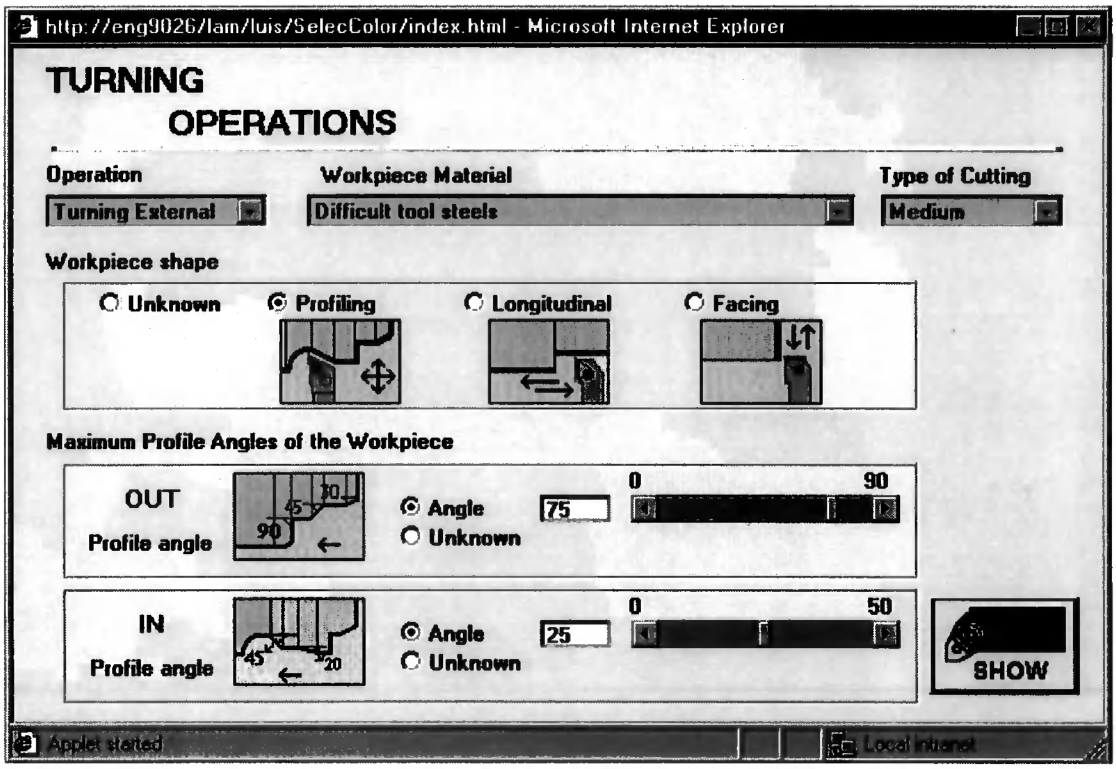
Figure 3.6 – SELTOOL, Functional Architecture

Figure 3.6 shows the functional architecture of ***SELTOOL***, where remote users can access the system through the Internet and obtain the tooling information of their interest, according to given input specifications. The database of ***SELTOOL*** is a repository of information embracing data from technical catalogues (paper and CD-based formats) and a proprietary company database (Access format).

3.3.3 Logic Associated with the Tool Selection Process

This section describes the procedure for matching Inserts with Toolholders considering turning operations, which provides an efficient criterion to select tools.

SELTOOL requires at least 3 input parameters for internal and external turning operations: *Type of Operation, Workpiece Material* and *Type of Cutting*, as shown in Figure 3.7.



http://eng9026/lam/luis/SelecColor/index.html - Microsoft Internet Explorer

TURNING
OPERATIONS

Operation: Turning External

Workpiece Material: Difficult tool steels

Type of Cutting: Medium

Workpiece shape

○ Unknown ● Profiling ○ Longitudinal ○ Facing

Maximum Profile Angles of the Workpiece

OUT Profile angle: ● Angle 75 ○ Unknown

IN Profile angle: ● Angle 25 ○ Unknown

Applet started

Figure 3.7 – Input screen of SELTOOL

The process used to match inserts and toolholders is summarised as follows.

i) ***SELTOOL*** explores the database for those suitable inserts, matching the input parameters specified by the user.

ii) For each insert found the program searches for the corresponding toolholders matching the insert selection, as shown in Figure 3.8. The insert and toolholder codes have been obtained from technical catalogues and, as can be seen in Figure 3.8, three tooling parameters are used in the matching process, namely, *Shape*, *Insert Side Clearance Angle* and *Cutting Edge Length*.

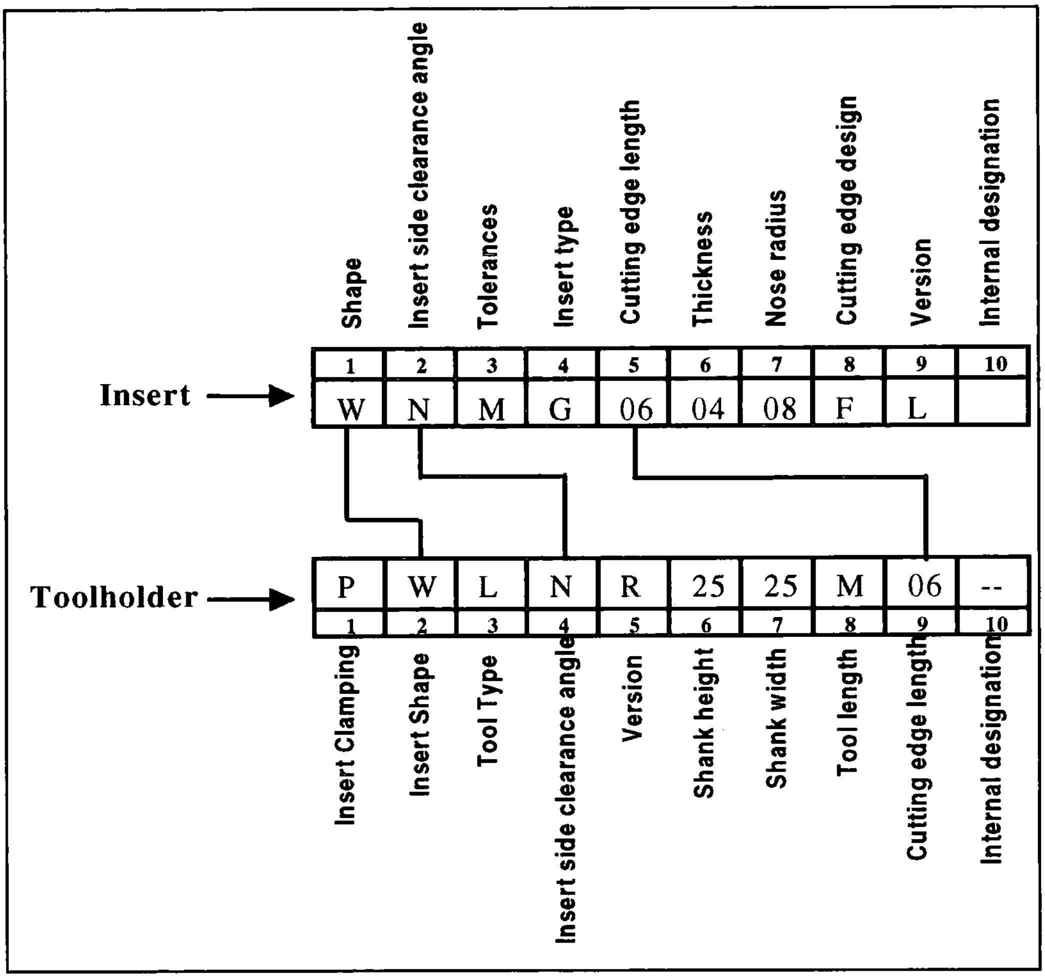


Figure 3.8 - Matching Inserts with Toolholders

iii) If the user knows the workpiece shape to be machined, optional parameters such as the *Profile Out* and *In Angles* can also be introduced. Figure 3.9 shows these angles and their relationship with the approach (κ) and trailing angles (ψ) of the tool.

Depending on the profile angles submitted the system searches for compatible toolholders with approach angle and trailing angle higher than the maximum workpiece angles given by the user.

The "*Profile-OUT*" angle is the profile angle generated by the cutting edge in the direction of the feed rate. Hence, this angle is compared with the approach angle (κ) of the tool as shown in Figure 3.9(a) and 3.9(c).

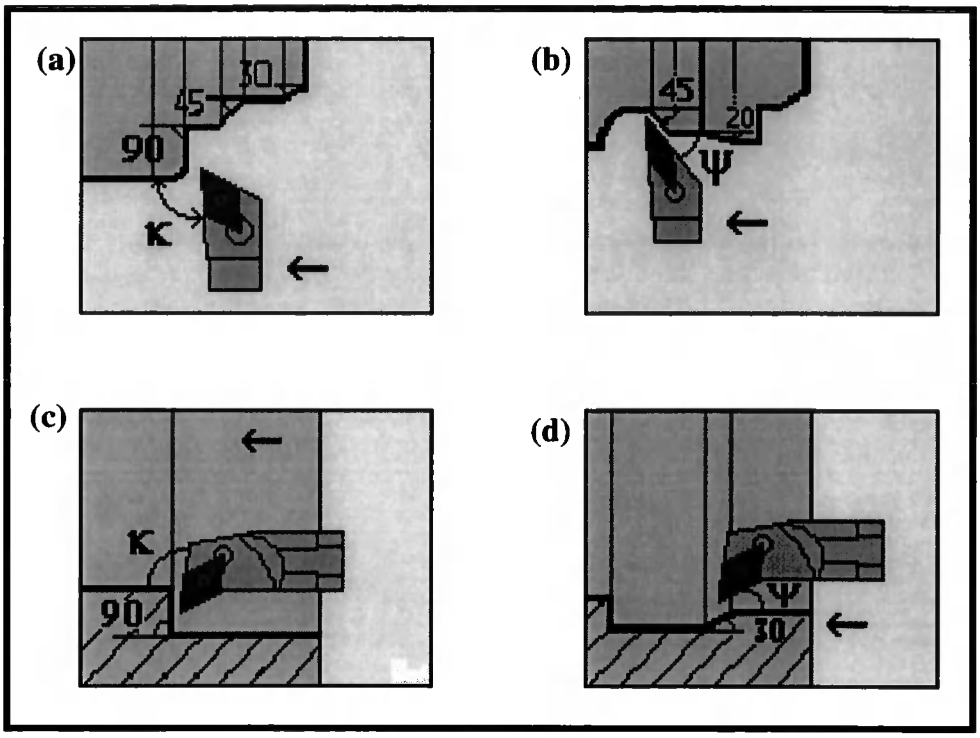


Figure 3.9 - (a) Profile OUT and approach angles – External Turning. (b) Profile IN and trailing angles – External Turning. (c) Profile OUT and approach angles – Internal Turning and (d) Profile IN and trailing angles, Internal Turning.

The "*Profile-IN*" angle is compared with the trailing angle of the tool (ψ) during the generation of recesses. The user can input the maximum values of profile angle values and this acts as an additional constraint during the selection of toolholders.

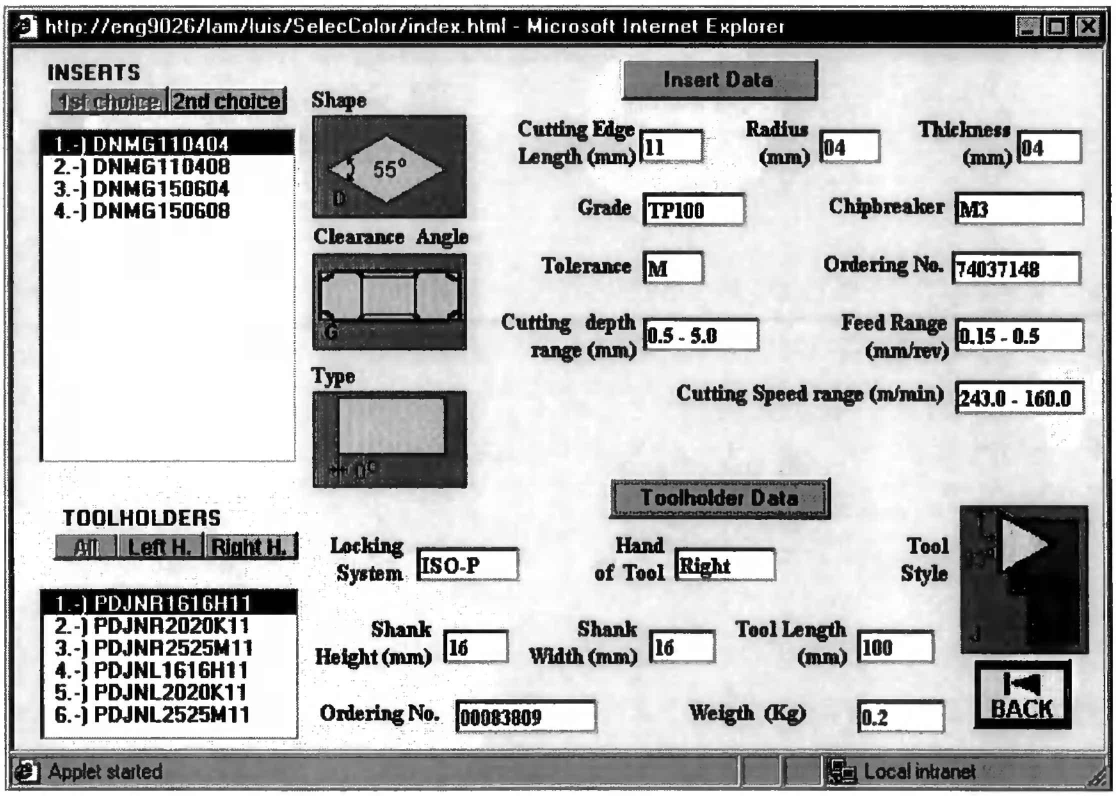
iv) In order to recommend cutting speed and generate ranges matching the corresponding feed rate values, it was necessary to apply a mathematical interpolation procedure.

Table 3.1 shows the recommended cutting speed and feed rates given for specific insert grades and material groups. Let us suppose that once the system processes the input parameters, the feed rate interval associated to the suitable insert chosen is "0.05 - 0.25". As can be seen in Table 3.1, there is not a similar cutting speed range matching this specific feed rate interval. That means, the system must be able to interpolate the feed rate values "0.05 - 0.25" with those shown in Table 3.1, to find a new cutting speed range.

Table 3.1 - Cutting Speed data

SECO material Group No.	Recommended cutting speed, Vc (m/min)								
	TP100			TP200			TP300		
	Feed Rate, f (mm/rev)			Feed Rate, f (mm/rev)			Feed Rate, f (mm/rev)		
	0,2	0,4	0,6	0,2	0,4	0,6	***0,2***	***0,4***	0,6
1	500	385	320	445	340	285	***365***	***275***	230
2	425	325	270	380	290	240	305	235	195
3	365	275	230	325	245	205	260	200	165
⋮		⋮			⋮			⋮	

SELTOOL provides the following information for inserts: *first and second choices together with, shape, clearance angle, type, cutting edge length, thickness, radius, tolerance, grade, chipbreaker, cutting depth, cutting speed, feed rate and ordering number*. In the case of toolholders, ***SELTOOL*** provides information about these parameters: *hand of tool, tool length, locking system, weight, tool style, shank height, shank width* and *ordering number*. Figure 3.10 shows a typical output screen displaying this information.



1.-) DNMG110404
2.-) DNMG110408
3.-) DNMG150604
4.-) DNMG150608

Cutting Edge Length (mm) 11
Radius (mm) 04
Thickness (mm) 04
Grade TP100
Chipbreaker M3
Tolerance M
Ordering No. 74037148
Cutting depth range (mm) 0.5 - 5.0
Feed Range (mm/rev) 0.15 - 0.5
Cutting Speed range (m/min) 243.0 - 160.0

1.-) PDJNR1616H11
2.-) PDJNR2020K11
3.-) PDJNR2525M11
4.-) PDJNL1616H11
5.-) PDJNL2020K11
6.-) PDJNL2525M11

Locking System ISO-P
Hand of Tool Right
Shank Height (mm) 16
Shank Width (mm) 16
Tool Length (mm) 100
Ordering No. 00083809
Weigth (Kg) 0.2

Figure 3.10 – Output screen of SELTOOL

As was shown, the graphical interface permits the user to submit the input parameters and obtain the results through an user-friendly interface. All the specifications of inserts, toolholders and cutting data are presented in one screen and additional options for second choice inserts are provided. These characteristics allow a better visualisation, and a faster and interactive way of searching suitable tools than conventional representation schemes provided by technical catalogues.

3.3.4 Database Considerations

Thirteen tables linked in a relational model constitute the database of ***SELTOOL***. An SQL-based Database Management System (DBMS) was used for creating the database.

Initially, the database was populated manually, which represented very time consuming work, without taking into consideration the difficulties of subsequent modifications and updating. In order to minimise human errors whilst the data is inserted, optimise the update times, provide a better maintenance of the database and take advantage of existing data in Access format, two programs were developed, the Database Populate (*DPS*) and the Database Migratory (*DMS*) Systems.

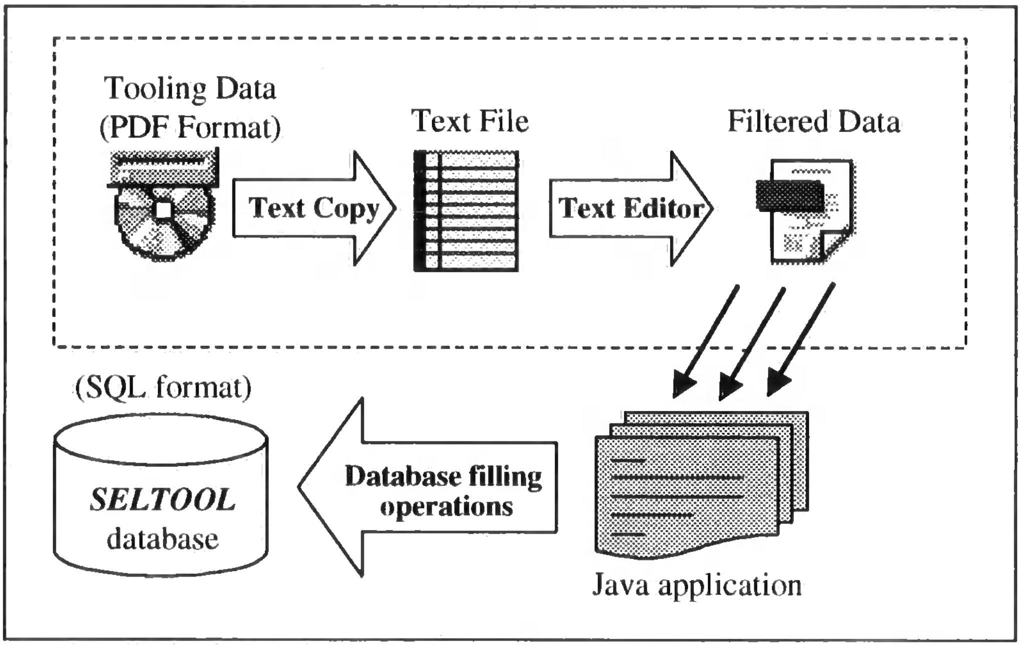


Figure 3.11 - Data transformation process using DPS

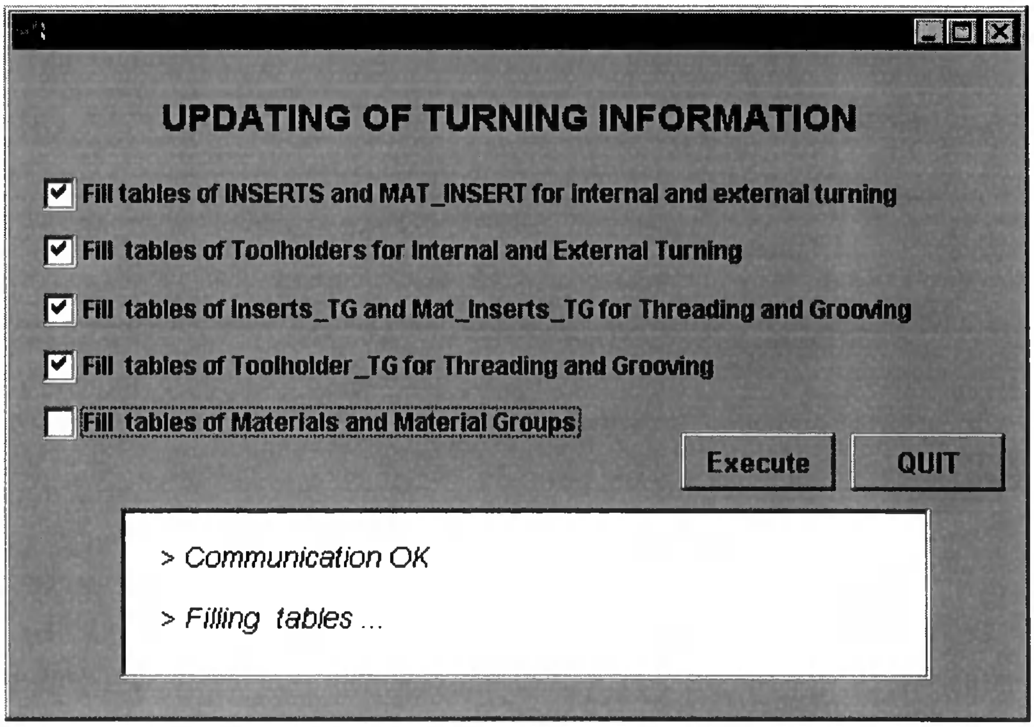
a) *Database Populate System (DPS)*

DPS was developed to carry out a data transformation process, which is illustrated in Figure 3.11.

To create a temporal text file, the tooling data stored in a Compact Disc (PDF format) is used. This file contains the data required but also has additional data and special characters not needed. In this case, a filtering process must be performed to extract this unsuitable data.

DPS reads the text file and after performing information filtering, the data is automatically written in the database. To support the read-and-write operations carried out by *DPS*, a Java program was developed.

Figure 3.12 shows how flexible the Java program developed is, to fill the different tables used by ***SELTOOL***, allowing a particular selection of options, depending on the type of data that the user wants to update. The bottom window is used to display information to the operator about the status of the program.



UPDATING OF TURNING INFORMATION

☑ Fill tables of INSERTS and MAT_INSERT for internal and external turning

☑ Fill tables of Toolholders for Internal and External Turning

☑ Fill tables of Inserts_TG and Mat_Inserts_TG for Threading and Grooving

☑ Fill tables of Toolholder_TG for Threading and Grooving

□ Fill tables of Materials and Material Groups

> Communication OK

> Filling tables ...

Figure 3.12 - Input screen of the read-and-write Java application

b) *Database Migratory System (DMS)*

In the searching of useful data for ***SELTOOL***, a barrier to using the available information was found. The information required was stored in DB2 and Access DBMS formats, as the rest of SECO databases, but the available DBMS for building the system was SQL Anywhere. To solve this situation *DMS* was developed.

DMS converts data from an existing tooling database (Access format) to the SQL database (SQL-Anywhere format) used by ***SELTOOL***. Figure 3.13 shows the data translation process carried out by *DMS*.

The first phase involved the duplication of an empty database. This is achieved through a method that manages the files as byte arrays and generates from them, new identical archives.

As the databases cannot be created through a program, an empty SQL-Anywhere DB, to be taken as reference for future translations, was created. This SQL DB consists of two files: one is where the data is going to be stored (.DB file) and the second is a .LOG file which contains the lists of DB indexes. These two files are part of the application and must exist to perform the translation process.

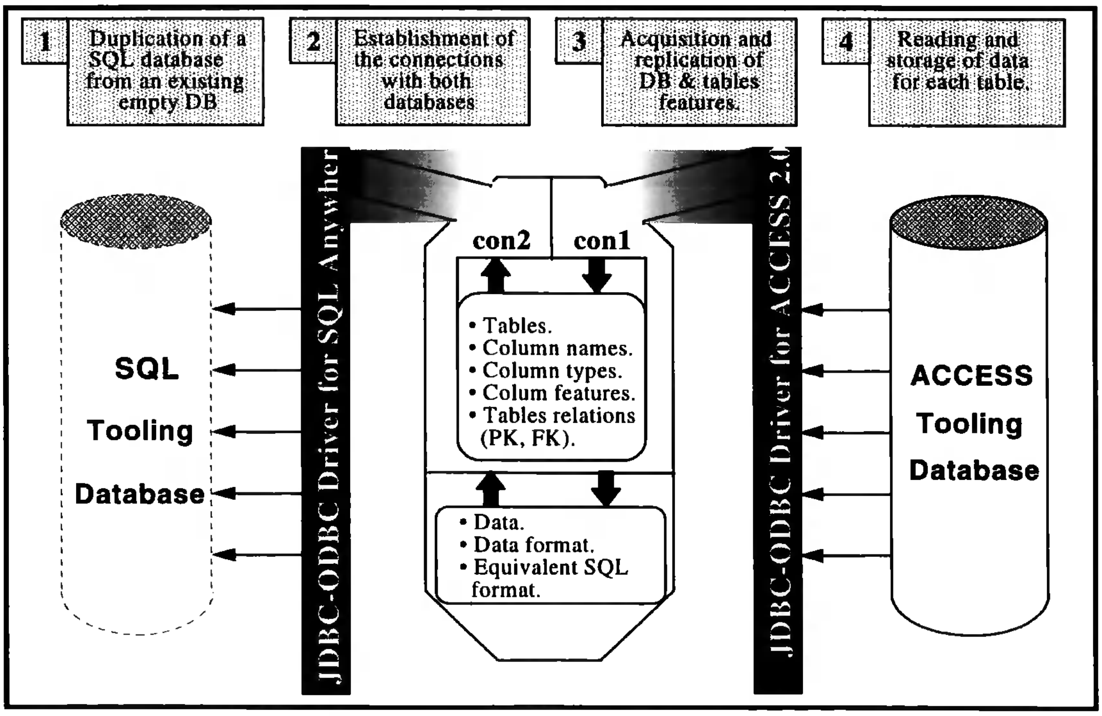

1
Duplication of a SQL database from an existing empty DB
2
Establishment of the connections with both databases
3
Acquisition and replication of DB & tables features.
4
Reading and storage of data for each table.
SQL Tooling Database
JDBC-ODBC Driver for SQL Anywher
con2
con1
• Tables.
• Column names.
• Column types.
• Colum features.
• Tables relations (PK, FK).
• Data.
• Data format.
• Equivalent SQL format.
JDBC-ODBC Driver for ACCESS 2.0
ACCESS Tooling Database

Figure 3.13 - Data translation process

The second stage is the connection with the databases. This application establishes the connection with both databases through the bridge Java Database Connectivity – Open Database Connectivity (JDBC-ODBC), and transforms the data from one DBMS to the other by using JDBC Applications Programming Interfaces (APIs). In this case, the bridge JDBC-ODBC is the convenient one because ODBC provides an extensive amount of drivers for a diversity of SQL compatible databases.

When the connection is established, the structure of the original database and their respective tables is read to obtain its properties.

The third phase is to procure the name of the tables, primary keys (PK), foreign keys (FK), relationship with others tables and the number of columns that the tables have. With this information the type and the size of the fields of each table must be obtained. Once this goal is achieved, it is necessary to create the features in the new database.

The last phase consists of reading each record from the Access database, the search of equivalent features in SQL-Anywhere and the writing of records in the new table using SQL format.

One of the problems found in the conversion process was the utilisation of SQL key words as names in tables and fields of the Access database, generating errors in the creation of the new SQL database. To solve this problem, a word parser function was developed that compares the names of tables and fields against a list of SQL key words. If a similar word is found, a new character is added to the name of the table (or field) to avoid errors in the creation of the new database structure.

In order to minimise the downloading times to access ***SELTOOL*** through the Internet, the API Java classes utilised by the system can be locally stored on the client machine. The new path where these Java classes are stored must be addressed setting the option *Classpath* of the file *Regedit* in the folder Windows or Windows NT.

The particular Java classes of the system, originated in the development environment, can, otherwise, remain stored in the server machine.

3.4 COMPUTATIONAL SUPPORT

This section describes the computer resources used for developing ***TTS*** as well as ***SELTOOL***.

TTS and ***SELTOOL*** were developed using *PowerJ* 2.0, which is a programming tool with graphical facilities and able to speed up the creation of Java projects.

Although *PowerJ* manages Java versions 1.1 and 1.02, the version used to develop the system was 1.02, due to the problems that still remain in current browsers with version 1.1, when applets containing database operations must be downloaded.

In order to establish the connection with the Internet, ***TTS*** and ***SELTOOL*** use *jConnect*, a 100% pure Java driver.

Nowadays it is very popular among Java programmers to use the bridge ODBC-JDBC to solve the connectivity barriers when databases have to be accessed through the Internet, because of the benefits of accessing Microsoft products, such as Access and SQL Server. However, ***TTS*** as well as ***SELTOOL*** do not use the ODBC-JDBC bridge since the DBMS used was Sybase SQL Anywhere, which supports the database operations through an Open Server Gateway included as part of the basic *PowerJ* tool package.

For developing and testing the systems, an internal net provided the facilities to transfer files and programs efficiently between deploy and development environments. A computer configured as a Web-server (Windows NT) was used to store the HTML files, images and all the programs and Java classes needed to download and run the system from remote locations.

3.5 SUMMARY

The preliminary aim of this chapter was to analyse the suitability of deploying Internet-based systems, examining the multiple factors involved when working in a collaborative information environment, whilst also providing solutions oriented to satisfy restricted as well as free-access information approaches.

In order to meet the needs of these fundamentally different approaches, two internet-based systems were deployed.

The first of them, ***TTS***, was developed under a private philosophy, where Seco tooling engineers working in a shared information environment, would be able to access a nation-wide database of knowledge, created from their previous work. Moreover, it was possible for tooling engineers to avoid the execution of new tool trials and know the results of trials carried out in physically distant places, when another engineer had previously executed these trials.

In order to keep high confidentiality levels, multiple security methods to access the system, were implemented.

The second system, ***SELTOOL***, was developed adopting a free-access philosophy, where world-wide Seco customers would remotely access tooling information in the field of tool selection. The information contained in the database of ***SELTOOL*** was obtained from tooling technical catalogues, available for Seco customers on paper-based and CD-based formats, so, it was not needed to implement restricted access procedures.

In the next chapter, the theoretical foundations of the Knowledge Discovery in Databases (KDD) area are examined, preparing the way for a subsequent application of data mining methods, in order to reveal common patterns and interesting associations in a database containing tooling trials data.

CHAPTER 4

KNOWLEDGE DISCOVERY IN DATABASES (KDD)

4.1 INTRODUCTION

In previous chapters, Web-based strategies to support distributed manufacturing solutions, as well as two Internet-based systems, have been examined. A general discussion of Knowledge Discovery in Databases (KDD) follows. The aim is to analyse the multiple factors involved when KDD-based systems are implemented, presenting their motivation, benefits, architecture and important issues to be considered.

4.2 DEFINITION

We live in an information age. Information has become a very important commodity. Every second hundreds of thousands of new records of information are generated. This information needs to be summarised and synthesised in order to support effective decision-making. In short, there is an urgent need to make sense of large amounts of data (Cios *et al.*, 1998). As a consequence, the expanding area of knowledge discovery in databases has emerged.

Knowledge Discovery in Databases (KDD) is a process oriented to discover potentially useful knowledge, analysing vast amounts of raw data. This process usually involves the development of efficient data warehouses, application of data pre-processing techniques, data mining methods and mechanisms to improve the comprehensibility of the discovered knowledge.

The main objective of KDD-based systems consists of making sense of data and using the discovered knowledge for decision-support purposes. KDD-based systems apply efficient data-organisation techniques and scientific methods to reveal deviations, dependencies, regularities and interesting patterns in raw data sets.

The diversity of data considered by KDD-based systems varies from text to sound and images, depending on the domain under study. Commonly studied areas are related to banking transactions, medical data, monitoring of industrial processes, production and marketing data, human social habits, analysis of data collected from astronomical observations and pattern recognition.

The existing KDD-based commercial systems require adjustment according to the needs of client organisations; so, the current KDD-based commercial tools are highly context-dependent.

4.3 REASONS TO APPLY KNOWLEDGE DISCOVERY IN DATABASES

The main benefits obtained from using a KDD-based system are related to the discovery of new knowledge as a consequence of analysing raw data, and assisting the manager or planning staff in decision-making processes. Newly discovered evidence contributes to providing a better understanding of the domain and therefore, it may change established practices. A discussion of these and other benefits follows.

- Direct influence on decision-making processes. Only the most significant and meaningful patterns are identified, which have been obtained taking into consideration the organisation's goals.

- Better understanding of areas under study. One of the most important resources of an organisation is its information, which is primarily stored in the form of raw data. So, any process oriented to analyse this data will automatically contribute to increase the understanding of the domain under study.

- Dealing with unsuitable data. Real-world databases usually contain noisy data. The concept of "noisy" can be interpreted as imprecise, contradictory, redundant or incomplete nature of the data. KDD technology can provide appropriate mechanisms to handle these unsuitable situations.

- Help to establish new directions. KDD is a regular time-basis process, so, each additional evidence obtained in subsequent analyses might indicate exploration along new, promising paths, which could provide new effective procedures. The results obtained due to application of knowledge discovery methods allow for example, the analysis of the feasibility of improving the performance of some processes or modify technical specifications to obtain higher quality of products.

For example, in a test process, the evidence from unsuccessful tests can reveal the influence of a particular factor causing the problem. After proper analysis, it will be possible to identify if the corrections have to be made to the production process, the product itself or about equipment manipulation methods by humans.

In a different context, when the sale expectations of a particular product are low, the results provided by applying KDD methods could shed light on the true origin of the problem, whether the causes are because of technical specifications, a poor distribution policy or due to wrong marketing practices. Hence, it is possible to isolate the factor causing the problem and carry out rectification actions.

- Minimum preconceived assumptions. Unlike situations in which it is necessary to employ standard mathematical or statistical analyses to test predefined hypotheses, KDD-oriented systems prove their power in exploratory analysis scenarios in which there are not predetermined conceptions about what will constitute an "interesting" pattern (Westphal and Blaxton, 1998).

- Automated processing. The analysis of huge amounts of raw data becomes extremely complex, tedious and inefficient when it is made manually. One of the significant benefits of a KDD-based system, is based on its capability to take advantage of computational power of current computers to carry out this analysis. An efficient KDD-based system must be able to deliver knowledge already interpreted as much as possible, although usually the participation of human analysts is expected to validate the final results and implement the respective decisions.

- Searching for knowledge out of business hours. Though KDD-based systems can be applied during normal business hours, they can continue working during hours and days considered non-operatives or when the members of the organisation are out of their regular business hours.

4.4 CONSIDERATIONS TO IMPLEMENT KDD-BASED SYSTEMS

Knowledge discovery systems face challenging problems from the real-world databases, which tend to be very large, noisy and dynamic (Hu, 1995). Also, KDD is not just a single task, on the contrary, it represents an integrated process usually requiring collaborative and multidisciplinary work teams. Hence, it is highly suitable to analyse the role, interrelation and availability of the multiple elements that participate in a KDD-based project, before deciding to carry out its development.

The following considerations involving domains, protagonists and strategic aspects are examined:

i) *Domain Knowledge.* It is very important to rely on useful knowledge sources about the domain under study. This knowledge allows the correct definition of the different functions that the system will be able to manage as well as provide ways to verify the truthfulness of the conclusions reached. Some reliable knowledge sources can be obtained from human domain experts, technical catalogues, similar knowledge-based systems, and nowadays, consulting related Web-based reference sites.

ii) *Data Sources Periodically Updated.* Successful KDD-based systems are oriented to improve the quality of their answers, as the amount of new data is supplied and processed in subsequent analyses. So, it does not make much sense to apply knowledge discovery techniques on static data repositories. It is highly suitable to rely on processes that can generate "fresh" data to be periodically incorporated to the discovery knowledge mechanisms.

Some sources of continuous or periodical data are obtained, for instance, from banking transactions, measurements taken as part of the monitoring of medical signals, industrial production processes, marketing, financial operations and any other process of interest that can generate information capable of being registered on a regular time-basis.

iii) *Suitable Computing Support.* A KDD process involves the application of automated methods to process large quantities of raw data, therefore, computer skills are needed to support all development stages. Some of the main computer tasks include:

- An appropriate management of huge amounts of data, developing a suitable database structure and efficient mechanisms to facilitate the operations of accessing, retrieval and modification of these data sets.
- Pre-processing functions to remove data inconsistencies, perform scaling processes and data conversions.
- Development of data mining algorithms.
- Implementation of mechanisms to improve the comprehensibility of the discovered knowledge and,
- Creation of a robust and friendly user-interface.

iv) *Multidisciplinary and Complementary Work Teams.* KDD is definitely a multidisciplinary approach. It heavily relies on a number of existing techniques and algorithms. In order to face challenging problems in relation to noisy, very large and dynamic real-world databases, KDD-based systems will be developed not only integrating a major number of techniques but also promoting the increase of more complete hybrid solutions.

It will allow exploiting the benefits of applying complementary learning approaches, increasing the spectrum and quality of the findings.

v) *Security.* Data security is another issue concerning to KDD technology. Many organisations are very sensitive on this matter, so, it is imperative that KDD implementations cannot compromise the confidentiality of private data. This situation becomes critical when facilities to access widely dispersed data, such as the WWW, are utilised. Therefore, authentication, authorisation and encryption procedures may be required.

vi) *Distributed Data.* Mining information from different and widely dispersed sources of data poses new challenges to KDD technology. In this distributed information scenario, local and wide-area computer networks connect many sources of data, providing easier access to remote locations. In the same manner, the remarkable growth of the WWW has effectively created a global-market, overcoming the barriers of cultures and languages and where geographical distances are no longer a factor for consideration in commerce and sharing information (Wong *et al.*, 1998). Hence, the use of computer networks facilities, particularly the Internet, should be highly considered.

4.5 ARCHITECTURE OF KNOWLEDGE DISCOVERY IN DATABASES

The previous section has sought to examine some issues and challenges to be addressed by KDD technology. In this section, a formal methodology to assist in the development of KDD-oriented systems is proposed.

KDD is a process that usually involves four main complementary stages: **i)** building a Data Warehouse, **ii)** data Pre-processing techniques, **iii)** Data Mining algorithms and **iv)** Post-processing methods.

Each main stage includes the definition and implementation of several operations grouped in sub-phases. The main stages are shown in Figure 4.1.

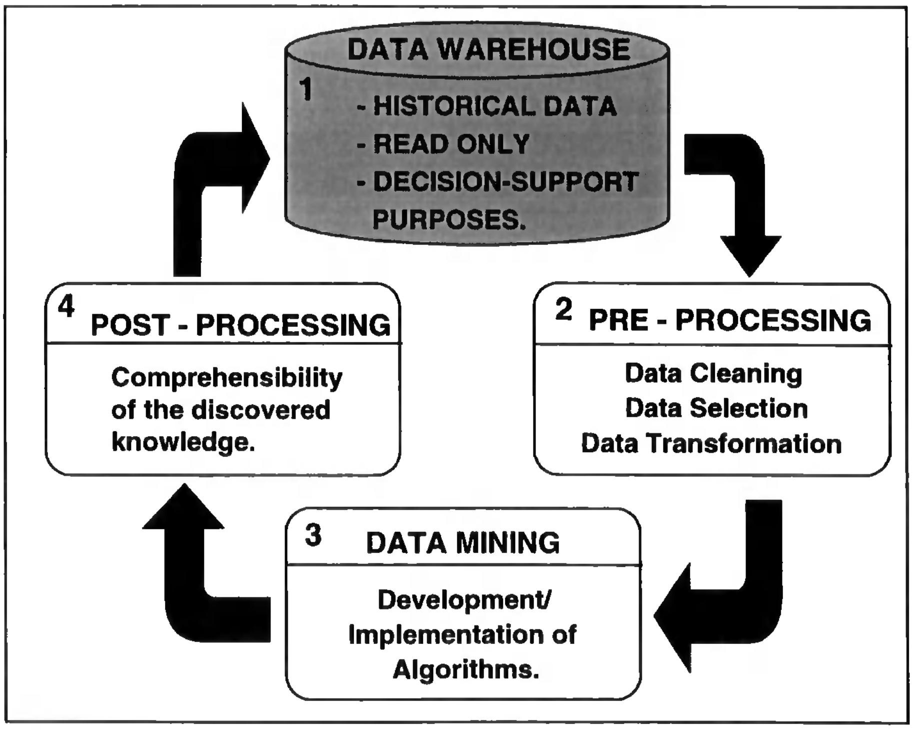


Figure 4.1 – KDD General Architecture

A detailed explanation about each of the main stages follows.

4.5.1 Building a Data Warehouse

The initial phase of a KDD process consists of building a Data Warehouse (DW), which is a repository of integrated, historical and read-only data, focused on storing and accessing information useful for high-level decision support systems, rather than for low-level operational (production) purposes.

The main reasons for building a DW are related to the interest of the analyst in efficiently managing the diversity of data stored (multiple patterns) and its magnitude (order of terabytes), which are two factors notably affecting the operations of access, retrieval, storage and maintenance of large databases. When such quantity and variety of information is stored, some of the following undesirable situations can occur:

⇒ The database segmentation is not the best suitable, for subsequently applying discovering methods to find significant deviations or meaningful patterns.

⇒ The access to the information is slow.

⇒ Redundant or inconsistent data exist, occupying costly space in the database.

Kimball *et al.* (1998) consider the following points as fundamental goals for DW:

- **Makes an organisation's information accessible**. The contents of the DW must be understandable, navigable and its access characterised by fast performance. *Understandable* means correctly labelled and obvious. *Navigable* means recognition about any user-destination on the screen and getting there easily in one click.

- **Makes the organisation's information consistent**. Information from one part of the organisation can be matched with information from another part of the organisation. If two measures of an organisation have the same name, then they must mean the same thing. Conversely, if two measures do not mean the same thing, then they are labelled differently.

- **DW is an adaptive and resilient source of information.** The DW is designed for continuous change. When new DW requirements are asked and new data is added to the DW, the existing data and the technologies are not changed or disrupted. The design of a DW must consider a distributed and incremental structure.

- **DW is a secure bastion that protects our information asset**. The DW not only controls access to the data effectively, but gives its owners great visibility into the uses and abuses of that data, even after it has left the DW.

- **DW is the foundation for decision making**. The DW has the data in it to support decision-making. Whilst more efficient is the organisation of the DW, higher will be its contribution to the successful application of subsequent knowledge discovery methods.

DW operations require special attention when performed on distributed environments, because they usually must be implemented between remote and different computer platforms, which demands certain connectivity and database considerations, just as discussed in Chapter 2.

4.5.2 Data Pre-processing

The second phase of a KDD process can be divided in three sub-phases:

Data Cleaning, in order to check possible redundancy and inconsistency in the data analysed. Redundancy exists when two records having the same values (or the same meaning according to the domain analysed) in all their attributes, the effect on the decision attributes (conclusion) is the same. That means one of the two records can be removed without affecting the potential of the data to provide useful knowledge.

Inconsistency occurs due to contradictory relationships between two records, given by conflicts between condition and decision attribute values.

The identification of inconsistent relationships in the data is not a trivial task, particularly when there are a considerable number of attributes to be analysed. For example, in a particular chemical process the presence of the same temperature for two records can result in apparent conflicting decisions (*normal* and *unstable*), when these decision values should also have been influenced by the presence of further condition attributes (pressure and humidity).

Also, the presence of records having apparent contradictory decisions can lead to discovering interesting deviations in the data and, therefore, their removal must be done taking the respective precautions.

Data Selection, because of the considerable variety of data contained in a Data Warehouse, only some key attributes will be relevant, according to the goals and interests of the analyst. The data selection process is usually done as a joint task between the data mining methods developer and the owner of the data, because it allows selecting those variables closely related to the business goals. About the decision of which particular variables to select, as well as their appropriate number, there is, in general, no solid theoretical basis for supporting these solutions and the problem must therefore be approached empirically.

Data Transformation, which includes the application of mechanisms oriented to convert all the variables to the same scale (standardisation) and complementary tasks, such as weighting of variables and discretisation operations. Each of these operations are explained below:

(a) Data Standardisation: real applications contain a variety of types of data having completely different measurement units, so, all the variables need to be standardised to some common numerical properties. The main reason to apply standardisation is primarily to convert all the values of the variables, which usually are not expressed in the same units, to the same scale. The conversion process usually requires dividing all the values for each variable, by a suitable equalising factor (Anderberg, 1973). Different equalising factors are discussed as follows.

⇒ **Range of the variable:** the range of a data set is the difference between its two most extreme values, so, dividing all values by their range, smooth the complete data set in relation to the maximum possible score found for each variable.

⇒ **Mean of the variable:** dividing all values of each variable by its mean, will produce a smoothing effect around the more frequent values present in the complete data set.

⇒ **Standard deviation of the variable:** dividing all values of each variable by its standard deviation, will contribute to minimise the range of variation, not only among values belonging to the same variable, but also the variations produced by significant differences between the units of measurement for any two variables.

Some authors (Duda and Hart, 1973) question the utilisation of equalising factors, because the standardisation process is applied on each variable separately, which could ignore possible correlation between the variables. A further disadvantage is related to the discriminatory potential of some group of variables, which can be diluted for the sake of implementing the standardisation processes.

⇒ **Considering variables of mixed type:** the three former standardisation methods can be directly applied when all the data is quantitative. When the data set contains quantitative variables, as well as categorical ones (describing qualitative attributes), it is highly suitable to rely on some mechanisms to convert all the values to common numerical scores.

Gower (1971) suggested a similarity coefficient to be used in clustering analysis, which contributes to solve the situation of having mixed variable types. The treatment given to categorical data to calculate a similarity factor is the key element, in which Gower assigns the value one, when two variables have the same value and zero otherwise.

In Chapter 6, an extended explanation about Gower's coefficient is presented and an improved version is implemented in a clustering context.

(b) Weighting of Variables: to weight a variable consists in giving greater or lesser importance to this variable, than others considered in the analysis. The validity of this procedure has been questioned by some authors, who argue that the weights can only be based on intuitive judgements of what is important (Everitt,1993), and that these may simply reflect existing classifications of the data when, on the contrary, data mining methods are applied to sets of data in the hope that previously unnoticed and potentially useful knowledge can emerge.

If there are not clear and available evidences about the relevance of the different variables, the assignment of equal weighting would seem appropriate (Gordon, 1981).

(c) Discretisation operations: are oriented to split the values of a continuous (integer-valued or real-valued) attribute into a small list of intervals. An example of discretisation in the tooling sector is shown in Figure 4.2, where twenty five continuous values of tool life were transformed into five discrete intervals (I_j).

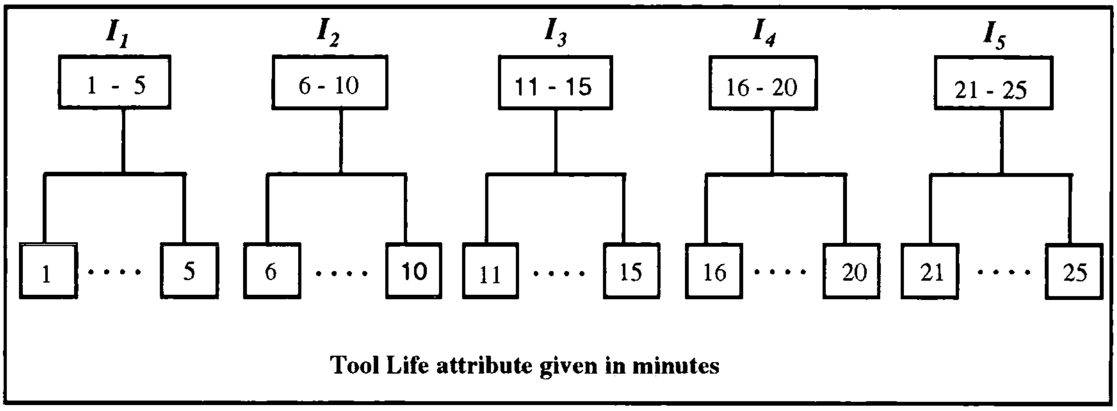


Figure 4.2 – Discretisation example

For instance, those tools presenting tool life values in the range 16 to 20 minutes can be identified through a unique variable (I_4), grouping all the possible individual values into this interval.

When generating decision rules in a context of Rule Induction, specialising and generalisation operations to transform a candidate rule into another, can be applied. Examples of specialisation and generalisation operations in rule induction in the tooling area are shown in Figure 4.3.

Figure 4.3-a) shows how adding a new condition (standard insert profile) to the antecedent of the original rule, the criterion to obtain the resulting conclusion (grade insert = “cp50”) is now more restricted. In contrast, Figure 4.3-b) shows how changing a particular kind of steel material subgroup (very soft and low carbon steels) into a more generic group (steels), it is now possible to relax a condition in the antecedent of the original rule, so, the conclusion can include now a higher interval of values.

Original Rule: If (Operation = ‘Ext. Threading”)
Then (Grade Insert = “CP50 or CP60”)

Specialised Rule: If (Operation = ‘Ext. Threading” and Standard Insert Profile = “UN”)
Then (Grade Insert = “CP50”)

a) Specialising a rule by adding a conjunction to its antecedent

---

Original Rule: If (Operation = ‘Ext. Threading” and
Workpiece material = “very soft, low carbon steels”)
Then (Cutting speed = “160”)

Generalised Rule: If (Operation = ‘Ext. Threading” and Workpiece material = “steel”)
Then (150 <= Cutting speed <= 180”)

b) Generalising a rule by relaxing a condition in its antecedent

Figure 4.3 - Specialisation and generalisation examples

The main reason for applying conversion mechanisms is to prepare the data for a later and easier application of data mining algorithms. Practical examples of data standardisation and weighting variables can be seen in Chapters 5 and 6 respectively, and an exhaustive analysis about scale conversions and type of variables can be found in (Anderberg, chapter 3, 1973) and (Gordon, chapter 2, 1981).

4.5.3 Data Mining

Data mining methods rely on the application of algorithms to automatically process large amounts of pre-processed data, in order to identify the most significant and meaningful patterns. Data mining methods demonstrate their usefulness especially in data-rich and knowledge-poor processing scenarios.

Although data mining algorithms can be applied to small repositories of data, a better performance is reached if there is a whole warehouse of data for the mining algorithms to work on. When more data is available, the greatest are the chance and the opportunities to discover hidden knowledge.

Part of the strength of data mining applications is due to the fact that they use more than one type of algorithm. Data mining methods are highly complementary, they apply a combination of statistical analysis, machine learning techniques and mathematical support to search for interesting patterns in the data.

There are two difficult problems faced by data mining algorithms, *overfitting* and definition of correct *bias* (Freitas & Lavington, 1998). *Overfitting* is a first problem associated with noisy data and spurious relationships. Noisy data can occur due to unintentional errors like typing wrong values when measures are collected and, spurious relationships is a term associated with a description of facts whose predictive power is apparent rather than true. Both of them, noisy data and spurious relationships have undesirable effects on the performance of data mining solutions, because they may lead the algorithms to induce a model that overestimates the data.

Cios *et al.* (1998) consider overfitting as a problem of over-training when applying learning methods such as neural networks, machine learning and even statistics. It refers to the tendency of a learning method to favour weights in the case of neural networks, or the generated rules in case of inductive machine learning, to agree with the training data too closely, in order to correctly describe all of the training examples. This is done at the expense of generalisation to other data, on which the trained network or the generated rules are to be tested later.

The second problem is associated with a correct definition of a *bias*, which can be defined as any basis for favouring one hypothesis over another, other than strict consistency with the data being analysed. A proper definition of good bias plays a significant role in data mining applications, because given a potentially high number of hypothesis or concept descriptions, the decision to choose the best of them among many other consistent ones, is crucial.

Some of the most important data mining techniques are examined in Section 4.6.

4.5.4 Post-processing

This last phase involves a refinement of the information generated once a particular data mining algorithm is applied.

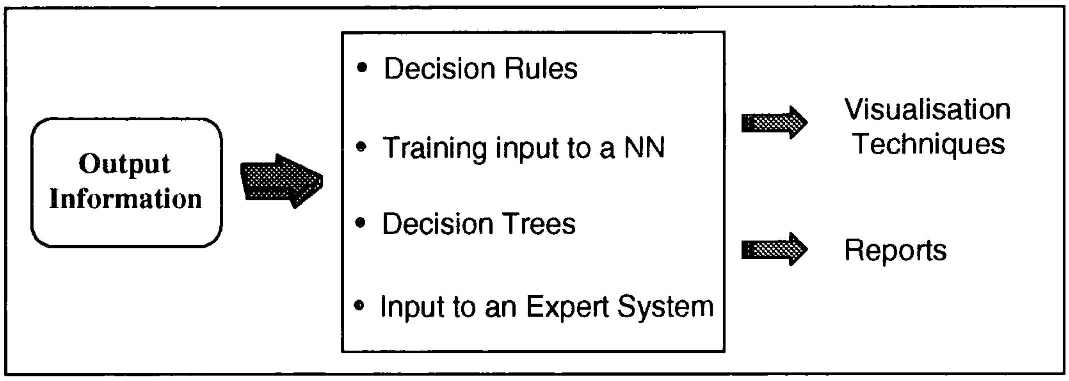


Figure 4.4 – Post-Processing application

Figure 4.4 shows two different tasks to be carried out by post-processing methods. Firstly, the patterns generated may be put to a multitude of uses, such as serving as the training input to a neural network or being encoded as a rule into an expert system or decision tree model (Westphal and Blaxton, 1998). Secondly, a data interpretation analysis is usually required to improve the comprehensibility of the knowledge discovered.

The more complex the data sources, the more likely that different perspectives of the data will be required in order to fully characterise patterns and trends of interest. Data visualisation techniques play an important role, providing multidimensional data views and different perspectives in order to show to the final users, clear and easy to understand results.

4.6 DATA MINING TECHNIQUES

Currently, the valuable contribution of knowledge discovery methods in making sense of large amounts of data has originated the rise of a huge variety of data mining tools. Next, seven of the most important data mining techniques implemented by these tools are examined, and Appendix C contains examples illustrating typical applications. A discussion and the reasons to apply some of these techniques to analyse tooling data, is finally presented.

4.6.1 Cluster Analysis

Clustering constitutes a fundamental technique to be used in exploratory data analysis, which considers the grouping of similar objects to produce a classification. The criteria of similarity to classify these objects are established in accordance with the objectives and goals of the researcher about the domain being studied.

Clustering methods represent a convenient method of partitioning huge amounts of data so that the resulting classification can reveal natural associations, logical structures and useful relationships. In the next chapter, an explanation about clustering methods, as well as their application in the analysis of tooling trials data is presented. Also, in Appendix A, hierarchical clustering methods are applied to solve a complete exercise involving two-dimensional data points.

4.6.2 Neural Networks

Neural Networks (NN) are systems constructed to make use of some of the organisational principles felt to be used by the human brain, especially those in relation to learning (Dagli, 1994). In the generic NN, also called *connectionist model*, there are three main components: the Neurones or processing elements, the network topology and the learning algorithm or strategy. Usually, there are three classical NN architectures commonly used: Perceptron, Backpropagation and Adaptive Resonance Theory.

Once weighted interconnections among the neurones are established, each one of them performs a very simple computation, such as calculating a weighted sum of its input connections, and computes an output signal that is sent to other neurones. The training (mining) phase of a NN consists of adjusting the weights of the interconnections, in order to produce the desired output. The adjustment of interconnection weights is usually performed taking into consideration that if two neurones are active simultaneously, the weight of their interconnection must be increased.

NN systems tend toward system robustness, resistance to noise, contradictions and incompleteness. However, the use of NN in KDD has two drawbacks.

Firstly, the distributed, low-level representation used by NN has the disadvantage of not being easily comprehensible to the user. That is, typically a NN returns the predicted class but it cannot provide a comprehensible explanation about why that class was chosen. In the context of KDD, it is sometimes desirable to convert the learned interconnection weights into a set of "If-Then" rules, to make the discovered knowledge comprehensible to the user. Unfortunately, this conversion is difficult, and often one of the prices to pay for this conversion is a reduction of the classification accuracy of the NN. In some application domains, such as finance, the accuracy of the discovered knowledge tends to be much more important than its comprehensibility. In these cases NN can be a promising approach to the mining of noisy, real-world databases. Secondly, training a NN can be a very time-consuming process.

NN have also been applied successfully in the areas of pattern recognition, forecasting and risk analysis.

4.6.3 Rough Sets

To provide a systematic framework for analysing incomplete and imprecise information, Zdzislaw Pawlak (1982) introduced Rough Sets. The concept of Rough Sets is based on equivalence relations which partition a data set into equivalence classes, and consists of the approximation of a set by a pair of sets, called *lower* and *upper* approximations.

The term "*Reduct*" is another important concept widely used in the rough sets community. *Reducts* allow establishing attribute relationships, which is important to reduce information (redundant attributes) in a data set. Readers interested in knowing more details about *Reducts* can find good examples consulting Shao (1996), Pawlak (1996), Ziarko (1995) and Hu (1995).

An important advantage of rough set theory is that it does not need any preliminary information about data, like probability in statistics and grade of membership or the value of possibility in fuzzy sets theory.

In order to show how rough sets theory can be used to identify inconsistent relationships in a data set, an example is presented in Appendix B.

4.6.4 Genetic Algorithms

In this paradigm, also called Evolutionary Computing, the KDD algorithm is an iterative procedure that maintains a population of "individuals" or "chromosomes", which are strings representing a candidate solution to a given problem. At each iteration (or generation) the individuals of the current population are evaluated by a fitness function, which measures the quality of the candidate solution represented by the individual. Then, genetic operators such as selection, crossover and mutation are applied to the individuals, modifying their corresponding strings and creating a new generation of individuals.

The key idea is that the generic operators evolve individuals according to the principle of the survival of the fittest, based on Darwin's principle of natural selection. Hence, the populations of individuals tend to converge on highly fit individuals that represent good solutions to the target problem.

In the context of KDD, individuals often represent candidate rules and the fitness function measures the quality of these rules. Genetic Algorithms carry out a global search in the solution space, in contrast with the local search used by most rule induction algorithms. It should be noted that one of the prices to pay for this global search is that Genetic Algorithms tend to be time consuming, in comparison with local search-based rule induction algorithms.

The advantages of Genetic Algorithms for the design of complex decision-making models could be summarised as follows:

- The algorithm itself does not deal with the actual input data but with binary representations. This allows virtually any kind of input data to be put into the control of the algorithms, as long as an *encoding/decoding* algorithm can be defined.

- The Genetic Algorithm is a domain-independent algorithm, which can be applied to any kind of problem domain. There are therefore, no serious restrictions on the type of model, which can be put under Genetic Algorithms control.

- Genetic Algorithms can deal simultaneously with a large number of input data and can create a large number of output results, allowing the analysis of complex, multi-dimensional problem matrices.

- The flexibility of Genetic Algorithms allows easy combination with other AI techniques to create *hybrid* models specifically suitable for a given task (like classifier systems or optimisation of neural network design).

- Genetic Algorithms allow implementation of constraints in various ways. This is important as the training of some models must observe strict risk management and thresholds applications.

- Genetic Algorithms are well-suited for parallel and distributed processing, as each population may consist of 100 or more individual models which can be evaluated simultaneously in parallel processes.

4.6.5 Fuzzy Sets

Decision-making involves perception, attitudes, subjection, conceptions and emotions and these nomenclatures can not be modelled or explained by randomness alone. Fuzzy logic is based on the mathematics of fuzzy sets introduced in the mid-sixties by Professor Lotfi Zadeh (1965) to model the features of uncertainty or vagueness that probability theory could not adequately handle.

A fuzzy set A in the universal set U is defined by a membership function $\mu_A(x)$ which associates a real number in the interval [0,1] to each element x in U. The fuzzy set is a class in which the transition from membership to non-membership is gradual rather than abrupt. A fuzzy set is characterised by its membership function $\mu_A(x)$, which takes on values between and including 0 and 1. If $\mu_A(x) = 0$ then the element is not in A, if $\mu_A(x) = 1$ the element is in A, and if $\mu_A(x)$ is, say, 0.7 then the element is more in A that not in A. As $\mu_A(x)$ approaches the value 1, the more x belongs to A.

Fuzzy logic and clustering techniques can be combined (Fuzzy-clustering) to produce a useful hybrid solution in which the grouping of particular objects can be supported by similarity relationships given by the implementation of membership functions.

Fuzzy-K, a hybrid tool developed combing fuzzy sets and clustering methods, is presented In Chapter 7.

4.6.6 Discriminant Analysis

Discriminant Analysis is a classification method that measures the importance of factors determining membership within a category. A typical example of applying the discriminant analysis technique consists of identifying common customer group behaviours.

There are some basic assumptions when using discriminant analysis. First of all, the data cases should be members of two or more mutually exclusive groups. Data cases are the basic units of analysis, the elemental things being studied. These may be people, animals, countries, etc.

Usually, the activities related to discriminant analysis can be divided into those used for *interpreting* the groups differences, and those employed to *classify* cases into groups.

A researcher performs *interpretation* when studying the ways in which groups differ, that is, when he/she is able to 'discriminate' between the groups on the basis of some sets of characteristics, how well do they discriminate, and which particular characteristics can be identified as the most powerful discriminators. The other application is to derive one or more mathematical equations for the purpose of *classification*. These equations, called 'discriminant functions', combine the group characteristics in a way that will allow one to identify the group, which classification it most closely resembles.

Another important assumption is that no discriminant variable may be a linear combination of other discriminant variables used in the study. A variable defined by a linear combination (perfect correlation) does not contain any new information beyond what is contained in the present combination of discriminant variables, so it is redundant.

A final assumption is that each group is drawn from a population that has a multivariate normal distribution of the discriminant variables. Such a distribution exists, when each variable has a normal distribution about fixed values on all the others. This permits the precise computation of tests of significance and probabilities of group membership.

The above assumptions constitute the mathematical model on which the most common approaches of discriminant analysis are based (Klecka, 1980).

4.6.7 Machine Learning

Traditional multivariate data analysis methods such as conventional clustering, discriminant analysis, multivariate analysis of variance, regression and factor analysis are mainly oriented towards a numerical characterisation of a data set.

On the other hand, machine learning methods are primarily oriented towards developing symbolic logic-style descriptions of data, which may characterise one or more sets of data qualitatively, differentiate between different classes, create a "conceptual" classification of data, select the most representative cases and qualitatively predict sequences. These techniques are particularly well suited for developing descriptions that involve categorical variables in data (Michalski, 1998).

The primary reason for applying machine learning methods is to carry out conceptual data exploration (see chapter 5, section 5.3). This exploration is mainly supported by two kinds of algorithms. Firstly, algorithms oriented towards the construction of hierarchies (conceptual clustering and decision trees) and secondly, algorithms oriented towards the inductive derivation of general rules characterising the relationship between designated output and corresponding input attributes.

4.6.8 Discussion

Whilst it is relatively easy to find good examples of AI-based implementations in the tooling industry, a vast amount of raw data that is constantly generated in the machining centres of tooling companies, is not considered for discovery knowledge purposes. There is a significant lack of KDD implementations and tools applied to this sector, which could reveal the presence of potentially useful hidden knowledge.

In order to meet these needs it was decided that a formal KDD-based system would be developed. In this research, from the seven data mining methods formerly examined, clustering methods (based on numerical and conceptual procedures) and a hybrid solution combining fuzzy sets and cluster analysis, were implemented. Also, an SQL-query based solution was developed as a complementary tool to support fast exploratory data analysis. Chapter 7 presents all of these methods operating in an integrated fashion.

Cluster Analysis was chosen because of its widely recognised power as a data classification tool. In the context of this research, where trials data obtained from different tooling machining centres was considered, one of the main issues consisted of grouping these data sets according to logic classification criteria. Likewise, fuzzy sets were implemented in order to deal with imprecise, and sometimes, incomplete information in the tooling reports analysed.

With regard to the rest of the data mining methods examined here, they also can, within some limits, be applied in the current context. For example the low-level representation of neural networks, makes them harder to interpret the resultant model, with their layers of weights and transformations. In the case of genetic algorithms, they tend to be very time consuming, because they carry out global searches in the solution space. Machine learning methods are suitable for analysing data sets involving variables where qualitative attributes prevail, and in this research the greater part of the variables are quantitative.

The implementation of the above mentioned methods, taking into consideration the referred limitations, could be considered in subsequent stages, after primary techniques had performed an initial classification of data. This idea seems appropriate due to the fact already discussed in section 4.5.3, i.e. that data mining algorithms are highly complementary and the conclusions obtained due to the application of a particular technique, can be strengthened, validated or even weakened by other methods.

Finally, because in this particular research the data mining methods are implemented under an Internet environment, which is a platform characterised by a relatively slow access, special attention must be paid in relation to the computer processing times.

4.7 SUMMARY

In this chapter, the fundamental aspects of KDD-based systems have been discussed. It has been shown that KDD is a fast growing technology that has emerged as an attempt to fulfil the interest of many companies to take advantage of their large repositories of raw data, stored as a consequence of their continuous business operations. The primary motivation to implement KDD-based systems, consists of making sense of data and using the discovered knowledge for decision-support purposes.

The work in this chapter has concentrated on the definition of a well-structured architecture to support the development of KDD-based systems. Four main stages were identified in this architecture: *Building a Data Warehouse, Pre-processing phase, application of Data Mining techniques and Post-Processing methods to improve the comprehensibility of the knowledge discovered.*

Finally, due to the application of data mining techniques constituting the key stage for discovering interesting patterns hidden in the data, seven of the most important data mining methods, currently used in the KDD domain, were discussed.

In the next chapter, a complete methodology to apply Cluster Analysis is proposed. This methodology will be implemented to develop ***Kluster***, a clustering tool oriented to discover useful patterns and relationships in a tooling database.

CHAPTER 5

CLUSTER ANALYSIS METHODOLOGY. PRE-PROCESSING STAGE

5.1 INTRODUCTION

In the previous chapter, important considerations when implementing KDD-based systems were discussed. Also, to support the development of KDD-based systems a four-step architecture was proposed and some selected data mining methods were examined. In this chapter, the fundamental notions of cluster analysis, an important area in the wide spectrum of the KDD discipline, are presented.

However, the main contribution of this Chapter, is the proposal of a formal methodology to support the development of clustering applications. This methodology has been structured according to three main stages, namely, *Pre-processing, Processing* and *Post-processing.* Each of these three stages includes in its turn several operations grouped in sub-phases, and their complete explanation will require a considerable analysis. It was therefore decided to split the presentation of the whole methodology into two Chapters, in order to provide manageable reading. The stage of *Pre-processing* will be presented here and the next Chapter will go on to examine the remaining two stages.

The first function in the *Pre-processing* stage consists in choosing those variables considered relevant to the domain being studied. An entity analysis follows, in order to remove or incorporate new individuals. Next, a variable analysis considers the implementation of conversion mechanisms to support the calculation of numerical similarities between mixed variables. Finally, standardisation procedures are applied when significant differences exist in the magnitude of the variables under analysis.

5.2 CLUSTER ANALYSIS

Clustering techniques have been playing an important role in exploratory data analysis, partitioning huge amounts of data, in order to reveal natural associations, logical structures and useful relationships. Cluster analysis can thus be classified, as a data mining method in the wide spectrum of knowledge discovery process.

Usually, clustering methods require the raw data matrix to be transformed into an *n* x *n* symmetric matrix of pairwise dissimilarities *D*, represented by d_{ij} values, where d_{ij} denotes the dissimilarity (or distance) between the *i*th and *j*th individuals. These dissimilarity measurements must satisfy the following minimum conditions:

(i) $d_{ij} \geq 0, \forall ij \in D;$

(ii) $d_{ij} = d_{ji}, \forall ij \in D;$

(iii) $d_{ii} = 0; \forall ij \in D.$

According to condition **(ii)** the maximum number of distances found in a dissimilarity matrix is given by *n(n-1)/2*, where *n* is the number of individuals to be clustered.

Gordon (1987), suggested that data recorded in the form of an asymmetric dissimilarity matrix, where the dissimilarity between the *i*th and *j*th individuals is now denoted by a_{ij}, can be analysed by transforming them to $d_{ij} = \frac{1}{2}(a_{ij} + a_{ji})$.

The generation of a consistent dissimilarity matrix, able to introduce reliable numerical distances between all the individuals considered, constitutes one of the key pieces in the whole clustering analysis. When the indicator to measure the association between the *i*th and *j*th individual is based in *similarity* levels, instead of dissimilarities, some transformations are needed. In chapter 6 some examples of these transformations are shown.

To represent the unions generated when a particular clustering algorithm is applied, a two-dimensional diagram known as *dendrogram* is used. Figure 5.1 shows four different formats to represent dendrograms.

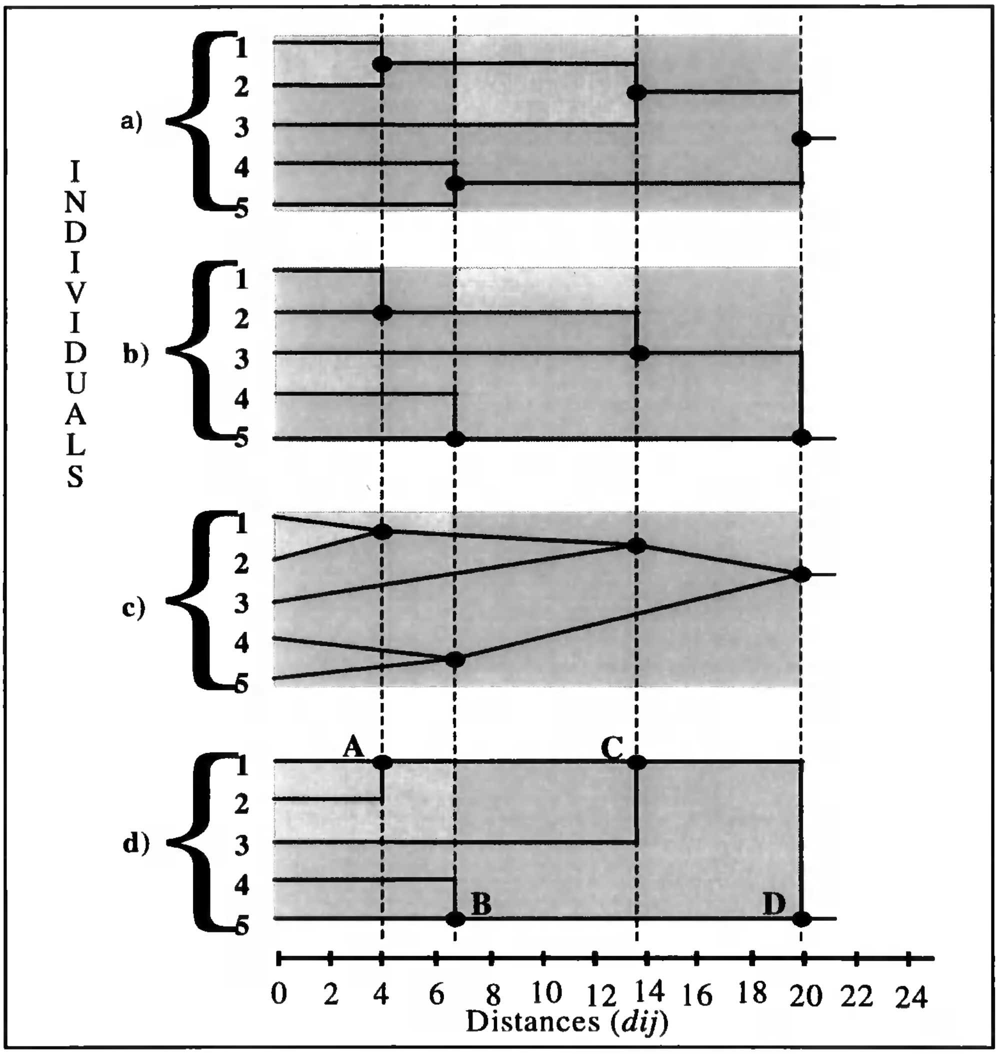


Figure 5.1 - Four different formats to represent dendrograms.

The points in the dendrograms represent the distances at which the successive unions take place. For instance, point *A* indicates a first fusion between the individuals 1 and 2, and point *D* represents the maximum distance to which the data set has been reduced to a final cluster containing all the individuals.

Cluster analysis has long been applied and there is a varied and vast literature on this matter. Xie and Beni (1991) combined fuzzy sets with clustering techniques (fuzzy-clustering) to solve problems in pattern recognition and image processing areas. They developed a validity function, applied to colour image segmentation, in a computer colour vision system for recognition of integrated circuit wafer defects, which are otherwise impossible to detect using grey-scale image processing.

In the field of supervised learning-algorithms, Djoko *et al.* (1997) implemented a method for guiding the discovery process using domain-specific knowledge. Results showed that incorporation of domain-specific knowledge improves the search for substructures that are useful to the domain and leads to greater compression of the data. This supervised learning-approach has been applied successfully in this research.

Bajcsy and Ahuja (1998) presented a new approach to hierarchical clustering of point patterns. This approach is applied to a two-step texture analysis, where points represent centroid and average colour of the regions in image segmentation. The authors compared their proposed hierarchical location- and density-based clustering algorithms, against four other methods, namely, *Simple*, *Complete*, *FORGY* and *CLUSTER* (Jain and Dubes, 1988). In many cases the results showed a better performance of their two algorithms than the above mentioned methods, considering the kind of process being analysed (image segmentation).

Deserving special mention is the research carried out by Michalski and Stepp (1983), which developed a method for automated construction of classifications called *conceptual clustering*, implemented in the program CLUSTER/2. This method of conjunctive conceptual clustering was analysed and compared to a number of clustering techniques used in numerical taxonomy.

The major difference between the above method and numerical taxonomy methods is that it performs clustering not on the basis of some mathematical measure of object similarity, but on the basis of "concept membership". From the viewpoint of traditional clustering methods, *conceptual* can be interpreted as an approach that also uses a certain measure of object similarity, but of a quite different kind. This new kind of similarity measure takes into consideration not only the distance between objects but also their relationship to some predetermined concepts, called by the authors, *conjunctive descriptions.*

The next section discusses the main differences between conventional and conceptual clustering approaches.

5.3 CONVENTIONAL VERSUS CONCEPTUAL CLUSTERING

Conventional clustering methods typically classify data on the basis of a similarity measure that is a function exclusive to the properties of the individuals being compared (attribute values), and not of any other parameters (Michalski *et al.*, 1998):

Similarity(X,Y) = *f*[(attributes(*X*), attributes(*Y*)];

Where *X* and *Y* are individuals being compared.

In contrast, a conceptual clustering program classifies data on the basis of a *conceptual cohesiveness*, which is a function not only of properties of the individuals, but also of two other parameters: the description language *L*, which the system uses for describing the classes of individuals, and the environment, *E*, which is the set of neighbouring examples.

Conceptual cohesiveness(X,Y) = *f*[(attributes(*X*), attributes(*Y*), *L*, *E*];

Hence, two individuals may be similar, i.e., close according to some numerical distance (or similarity) measure, while having a low conceptual cohesiveness, or vice versa. An example of the first situation is shown in Figure 5.2.

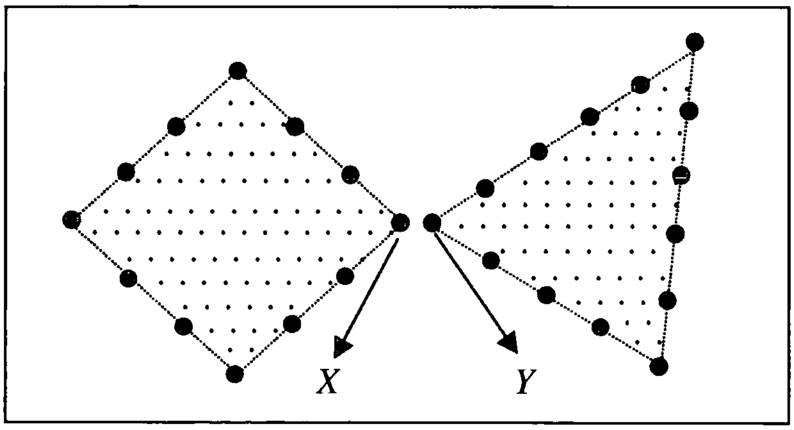


Figure 5.2 – Difference between conventional and conceptual clustering

As can be seen in figure 5.2, the points *X* and *Y* are close to each other, therefore, they would be placed into the same cluster by any method based exclusively upon the numerical distances between these points. Nevertheless, these points have small cohesiveness, because they belong to geometrical configurations representing different concepts. A conceptual clustering method, if provided with an appropriate description language, would cluster the points *X* and *Y* into a *rhombus* and a *triangle* respectively, as would be expected from a human interpretation.

In this particular research, in order to minimise incorrect data grouping processes, such as the situation illustrated in the previous example, the inclusion of important tooling concepts was considered to establish similarity between some attributes, before building the similarity matrix.

The Information to support the analysis of the tooling concepts was obtained from industrial technical catalogues, as well as engineering material books, particularly in the case of *Material* and *Grade* parameters. Hence, in this research, knowledge about how close are two records in the database, based in conceptual analyses of their attributes, has been implemented. This procedure is explained in detail in section 5.5.3.

The following section presents the clustering methodology developed in this research.

5.4 CLUSTERING METHODOLOGY

There is copious literature about data classification methods, however, few authors address the issue of exposing a formal methodology to carry out clustering analysis (Velásquez *et al.* (a), 2000). One reason for this lack of standard methodologies is probably because each classification problem requires a completely different analysis according to the domain under study and, due to the complexity and diversity of the topics involved.

Nevertheless, some general guidelines have been proposed on this matter. For instance, a procedure to carry out a step-by-step cluster analysis based on seven factors was suggested by Milligan (1996). These steps are summarised below:

Clustering Elements. A representative set of entities (individuals) to be clustered must be selected.

Clustering Variables. The variables to be used in the cluster analysis must be selected.

Variable Standardisation. If required, a standardisation process for each variable must be implemented.

Measure of Association. A similarity or dissimilarity measure must be selected.

Clustering Method. Selection and application of the clustering method, in order to apply a suitable classification algorithm.

Number of Clusters. Optimum determination of the number of clusters in the solution.

Interpretation, Testing and Replication. To interpret the results within the context of the domain under study, testing to determine if there is significant cluster structure in the data and finally, replication analysis is used to determine whether the obtained cluster structure can be replicated in a second sample.

The previous methodology summarises the significant steps in a cluster analysis, however, it does not consider some important factors such as *learning modes*, *variable analysis*, consistent indicators to measure the *quality of the final classification* and *optimisation methods*. An 11-step methodology that includes the last four mentioned factors is shown in Figure 5.3.

This section will examine the *Pre-processing* stage and the remaining stages will be presented in the next Chapter.

5.5 PRE-PROCESSING STAGE

The *Pre-processing* stage has been structured in four phases, oriented to prepare the data for a later generation of a consistent dissimilarity matrix and application of clustering algorithms. *Variable Selection, Entities Analysis, Variable Analysis* and *Data Standardisation* integrate this first *Pre-processing* stage.

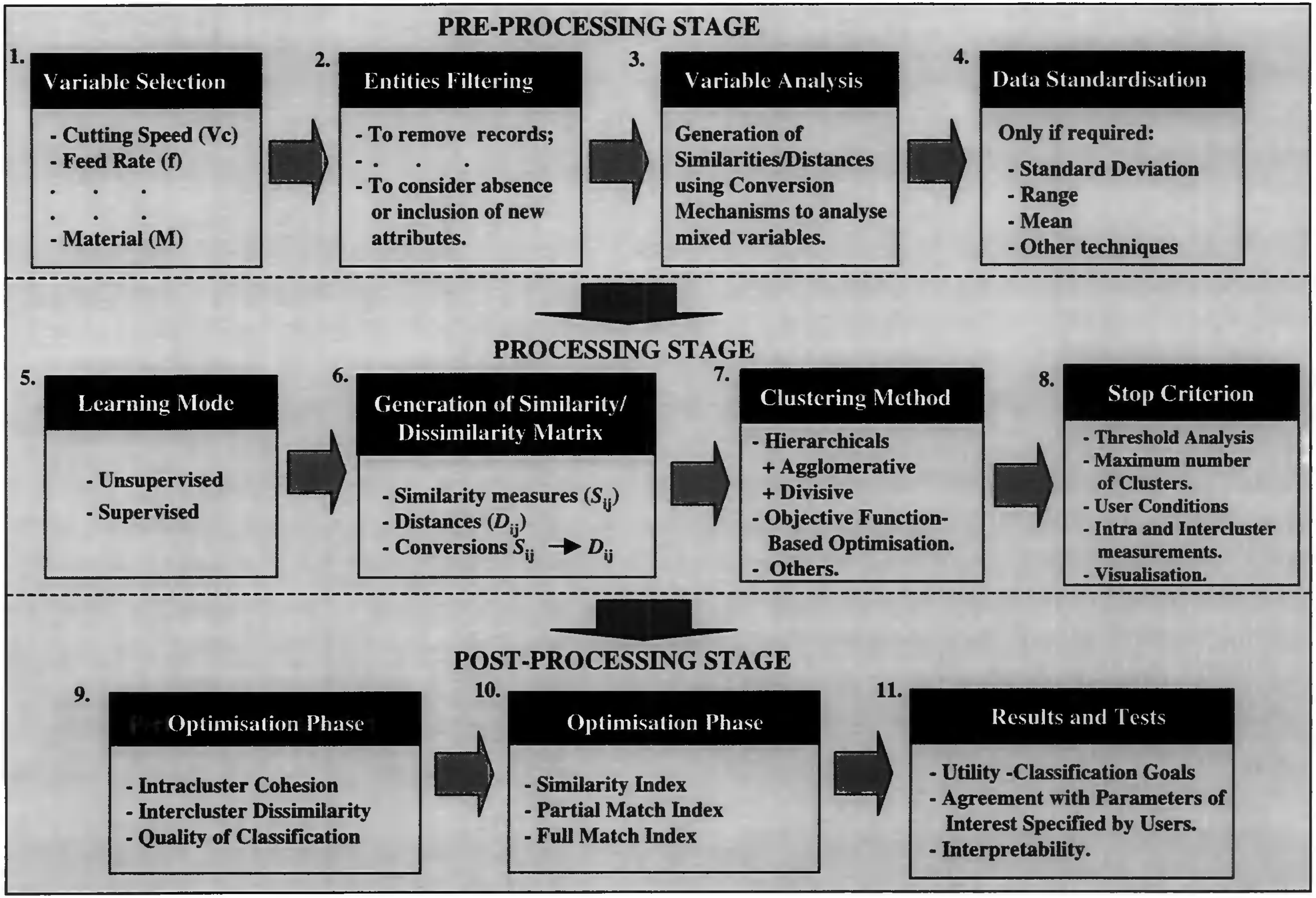


Figure 5.3 - Clustering Methodology.

5.5.1 Variable Selection

A suitable variable selection constitutes a crucial first stage in a cluster analysis. Data warehouses contain large repositories of raw data, but usually only some variables are relevant to support those areas of interest to the researcher. These areas are closely related to the user goals.

In this research, nine variables were selected to represent each record under analysis. These variables are shown in Table 5.1.

Table 5.1 – Selected Variables

Report #	Test #	Material Workpiece	Material Group	Nose Radius	Grade	Cutting Speed	Feed Rate	Depth of Cut
1	1.1	Stainless Steel	8	0.4	TP200	220	0.25	0.20
1	1.2	Stainless Steel	8	0.4	TP100	180	0.20	0.25
1	1.3	Stainless Steel	8	0.8	TP200	200	0.40	0.30
2	2.1	Cast Iron	11	0.8	CM	240	0.40	1.50
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
n	n.m	Nodular Cast Iron	14	1.2	TX150	280	0.35	4.0

Because each tooling report analysed can contain several tests, the first two columns in Table 5.1 correspond to control variables, needed to identify which and how many tests are associated with each report.

The remaining seven variables constitute important tooling parameters. In this research a number of 410 reports were analysed for a total of 1248 tooling tests.

5.5.2 Entities Analysis

To select and remove appropriate individuals from raw data sets to be clustered, and decide the incorporation of new attributes to the existing individuals, is a task mainly dependent on the goals and interests of the domain under study.

Empty fields, null entries, absence of some meaningful parameters to the user and presence of irrelevant attributes are examples of data that can spoil subsequent analyses. In this particular research, all the records, empty or zero-valued, corresponding to the variables *Grade, Nose Radius, Cutting Speed, Depth of Cut and Feed Rate,* were removed from the database.

Table 5.2 – Workpiece materials classified into material groups

Material GroupNo.	Material Description	Workpiece materials classified into material groups according to known standards						
		AIS	W-stoff	DI	BS	AFNOR	SS	UN
1	Very soft, low-carbon steels. Low-carbon and purely ferritic mild steels. Ultimate tensile strength: up to 450 N/mm^2.	1006 1010 - ⋮	1.0201 1.1121 1.1121	St 36 Ck 10 St 37-1	- 045 M10 436040A	Fd 5 Xc 10	1160 1265 1300	- C10 -
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
15	High-alloy cast iron which is difficult to machine.	A48-50B A48-60B A220-...	0.6035 0.6040 0.8170	GG 35 GG-40 GTS -	Grade350 Grade400 -	Fd 35D Ft 40D Mn700	0135 0140 0864	G35 - GMN 70

Likewise, in order to complement the information about materials contained in the database, additional workpiece material groups as shown in Table 5.2, were incorporated into a new file.

Table 5.2 shows an example of the structure of the different material groups, and a description of their constituent workpiece materials additionally included in the database. The values corresponding to the column *BS* (British Steel) will be used later in Chapter 6, as the basis to calculate the similarity between any two material groups according to important properties such as *tensile strength, yield strength, izod impact strength and hardness.*

5.5.3 Variable Analysis

In this section, an exhaustive analysis of the conversion mechanisms needed to support a later calculation of similarity/dissimilarity values has been conducted. Although this analysis is valid for any type of variable, it is particularly applicable when there are mixed variables, since the analysis becomes simpler when only quantitative variables are considered.

In this research a standardisation phase, fully explained in section 5.5.4, was the only consideration in relation to quantitative variables. With regard to categorical attributes, the variables *Grade* and *Materials* were considered. The procedure to carry out this analysis follows.

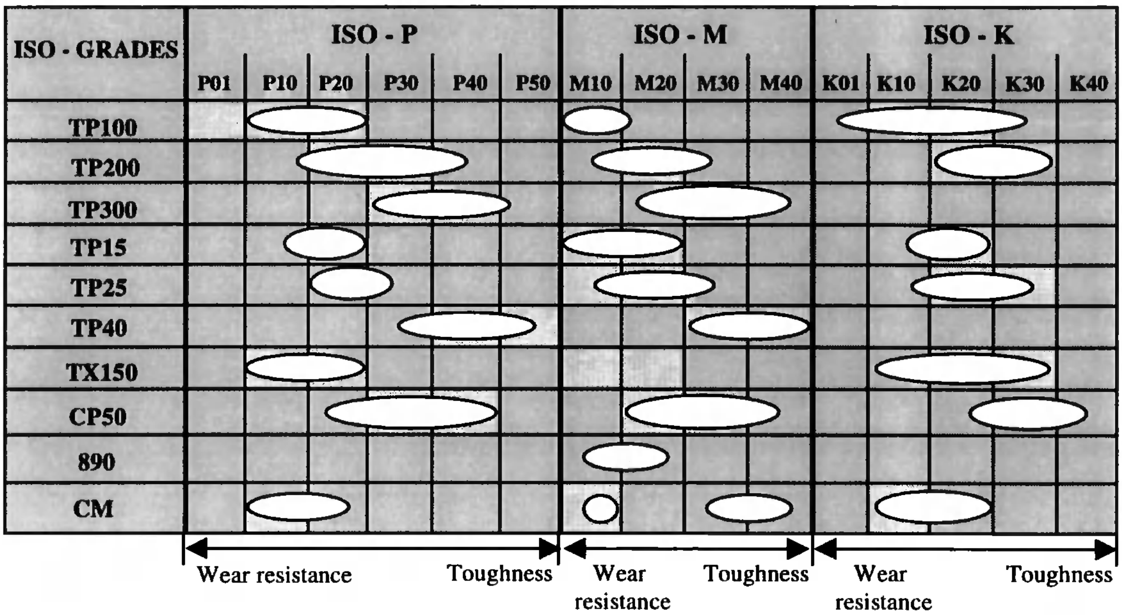


Figure 5.4 - Graphical representation of insert grade applications.

5.5.3.1 Analysis of Grades to obtain numerical similarity values

Insert grades are represented as categorical values and, therefore, the calculation of similarity measures requires the application of certain conversion mechanisms, in order to transform them to quantitative values. This conversion is based in the analysis of the workpiece materials which each insert grade has the ability to machine.

Figure 5.4 shows the overlapping areas in which a particular insert grade can be used alternatively to machine materials presenting different *toughness* and *wear resistance* properties.

Specifications shown in Figure 5.4 have been taken from tooling technical catalogues and they were used as the basis to obtain a numerical representation of minimum and maximum values of twenty insert grade applications according to three main groups of toughness/hardness indicators (*P,M,K*). This numerical representation is shown in Table 5.3.

Table 5.3 – Numerical representation of insert grade applications

	P		M		K	
	MIN	MAX	MIN	MAX	MIN	MAX
TP10	10	25	0	0	10	25
TP20	20	35	16	30	25	40
TP30	25	40	20	35	0	0
TP05	5	19	0	0	10	20
TP100	10	30	10	21	6	33
TP15	15	30	11	30	15	30
TP200	18	45	16	34	20	40
TP25	20	35	16	33	17	38
TP300	30	52	21	45	0	0
TP35	30	45	26	45	0	0
TP40	35	55	30	49	0	0
TX150	10	30	0	0	10	40
CP20	10	25	11	25	10	30
CP25	18	33	19	34	17	36
CP50	26	50	20	43	26	44
CM	10	25	15	20	0	0
CR	18	30	20	30	0	0
S25M	26	45	26	35	0	0
890	0	0	15	25	10	30
HX	0	0	20	30	15	33
883	0	0	20	31	20	36

According to Figure 5.4 there are six possible cases that can be identified when determining the extent of the overlapping areas. These cases are shown in Figure 5.5.

Revere (2000) used a similar procedure named *Parameter Relaxation Technique* to calculate confidence scores in order to assess the validity of one insert grade when compared to another.

These confidence scores were implemented in the *Tool Trials System* (***TTS***) described in Chapter 3, to show the way in which database records are scored based upon the number of parameters which exactly match those specified by the user.

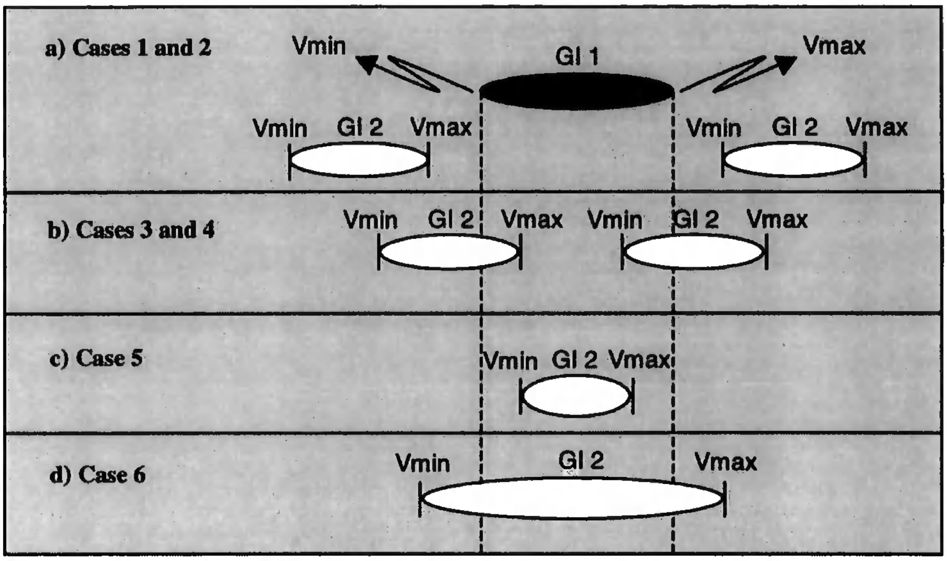


Figure 5.5 – Possible overlapping cases of insert grade applications.

The six cases shown in Figure 5.5 have been used as a reference to generate the respective numerical similarity values, calculated through the following rules:

Figure 5.5.a), cases 1 and 2,

If ((GI1.Vmax < GI2.Vmin) OR (GI1.Vmin > GI2.Vmax)) Then

$s_{ijk} = 0;$

Figure 5.5.b), cases 3 and 4,

If ((GI1.Vmax > GI2.Vmin) AND (GI1.Vmax < GI2.Vmax)) Then

$$s_{ijk} = \frac{\left(\left(\frac{GI1.V\max - GI2.V\min}{GI2.V\max - GI2.V\min}\right) + \left(\frac{GI1.V\max - GI2.V\min}{GI1.V\max - GI1.V\min}\right)\right)}{2};$$

If ((GI1.Vmin > GI2.Vmax) AND (GI2.Vmin < GI1.Vmin)) Then

$$s_{ijk} = \frac{\left(\left(\frac{GI2.V\max - GI1.V\min}{GI2.V\max - GI2.V\min}\right) + \left(\frac{GI2.V\max - GI1.V\min}{GI1.V\max - GI1.V\min}\right)\right)}{2};$$

Figure 5.5.c), case 5,

If ((GI1.Vmin <= GI2.Vmin) AND (GI1.Vmax >= GI2.Vmax)) Then

$$s_{ijk} = \frac{\left(1 + \left(\frac{GI2.V\max - GI2.V\min}{GI1.V\max - GI1.V\min}\right)\right)}{2};$$

Figure 5.5.d), case 6,

If ((GI1.Vmin >= GI2.Vmin) AND (GI2.Vmax >= GI1.Vmax)) Then

$$s_{ijk} = \frac{\left(1 + \left(\frac{GI1.V\max - GI1.V\min}{GI2.V\max - GI2.V\min}\right)\right)}{2};$$

Where,

s_{ijk} =The similarity component between the individuals *i* and *j* corresponding to the variable *k* (grade in this case) under analysis.

GI1,GI2 = Grades of individuals 1 and 2 respectively.

Vmax,Vmin = maximum and minimum values of grade applications for a particular individual respectively, taken from Table 5.3.

In this context, an individual is represented for a complete database record constituted by nine variables as shown in Table 5.1. A procedure to obtain the total similarity between two individuals including the partial similarity component calculated in this section is explained in Chapter 6.

5.5.3.2 Analysis of Materials to obtain numerical similarity values

The analysis of materials was carried out using the information of material groups already existent in technical tooling catalogues, as shown in Table 5.2. Therefore, to attach any workpiece material to a material group, an integer number between one (1) and fifteen (15) was utilised. When considering any two material groups, in order to obtain the similarity component, a further procedure was implemented.

Table 5.4 – Workpiece material groups.

Super Groups	Material GroupNo.	Specification
G1 = Steel	1	Very soft, low-carbon steels.
		⋮
	7	Difficult high-strength steels.
G2 = Stainless Steel	8	Easy -cutting austenitic stainless steels.
		⋮
	10	Austenitic and duplex stainless steels difficult to machine.
G3 = Cast Iron	11	Cast iron with medium hardness.
		⋮
	15	High-alloy cast iron difficult to machine.

Table 5.4 shows all material groups (1 to 15) and three main super groups (*G1, G2, G3*) obtained through technical tooling catalogues. In this section, comparing two material groups belonging to the same super group will be considered, all the remaining cases will be explained in Chapter 6. The analysis consists in assigning a similarity value to two material groups, on the basis of how close they are into a super group to which they belong.

To calculate the similarity component when comparing two material groups belonging to the same super group, a regression analysis based on four important mechanical properties was carried out. These properties are *Tensile Strength*, *Yield Strength*, *Izod Impact Strength* and *Hardness*, which are shown in Table 5.5.

Values in Table 5.5 were firstly obtained by Revere (2000), who applied a similar regression analysis to support a materials matching process when comparing materials specified by final users against those existing in a tool trials database. He called this procedure "*Data Relaxation Technique*".

The justification presented by Revere with respect to the fundamental rule governing the allocation of material groups is that materials within any particular group will exhibit similar properties, in terms of their machining characteristics. Were it that those materials within any single group did display correlation in terms of their mechanical properties, then it would be possible to look at the relationships between material groups.

Table 5.5 – Mechanical Properties for Steel Material Groups.

	Tensile Strength (MPa)	Yield Strength (MPa)	Izod Impact Strength (J)	Hardness (HB)
Material Group 1	Very soft, low-carbon steels. Low-carbon and purely ferritic			
Mean Values	452	280	23	142
Standard Deviation	29.8	4.76	0.67	13.78
Material Group 2	Free-cutting steels, excluding stainless steels.			
Mean Values	519	325	22	162
Standard Deviation	38.4	26.7	2.85	12.3
Material Group 3	Structural steels and carbon steels. Plain carbon steels with			
Mean Values	631	349	30	183
Standard Deviation	37.2	30.1	2.84	13.2
Material Group 4	High-carbon and ordinary low-alloy steels. Medium-hard			
Mean Values	646	403	31	209
Standard Deviation	49.7	38.2	4.28	18.9
Material Group 5	Normal tool steels. Harder hardening and tempering steels.			
Mean Values	732	428	30	237
Standard Deviation	62.1	39.0	3.88	22.9
Material Group 6	Difficult tool steels. High-alloy steels with high hardness.			
Mean Values	898	496	35	242
Standard Deviation	81.8	47.0	3.51	21.5
Material Group 7	Difficult high-strength steels with high hardness.			
Mean Values	928	545	38	272
Standard Deviation	82.2	53.7	3.26	25.1

Based upon Revere's work and information presented by other researchers (Choudhury and Baradie, 1996) it was decided that a number of widely used mechanical properties would form the basis of a material machinability assessment, so, the four properties shown in Table 5.5 were chosen.

For each material super group shown in Table 5.4 (*G1..G3*) three different similarity equations were generated:

Belonging to ***G1*** → $s_{ijk} = -0.0226x^2 - 8E - 16x + 0.8832;$ ***Equation* (5.1)**

Belonging to ***G2*** → $s_{ijk} = -0.0364x^2 + 2E - 16x + 0.8886;$	***Equation*** **(5.2)**
Belonging to ***G3*** → $s_{ijk} = -0.0833x^2 + 9E - 16x + 0.8762;$	***Equation*** **(5.3)**

Where,

s_{ijk} = Similarity between two individuals *i* and *j* corresponding to the *k*th variable (material group in this case).

x = The result of subtracting the material group of individual *i* from material group of individual *j*.

To obtain the regression coefficients of *Equations* 5.1, 5.2 and 5.3, a modified version of the procedure followed by Revere (2000) was implemented. An explanation is presented below.

i) Some basic statistics such as the *mean* and the *standard deviation* were obtained for each material group, according to each of the mechanical properties. This information was found by consulting some engineering material books such as Carvill (1993), Bolton (1989) and British Steel (1989). The values for material groups 1 to 7 (super group *G1*) are shown in Table 5.5. To obtain the statistics for super groups *G2* and *G3* the same method has been applied.

ii) It was necessary to carry out a normalisation procedure because of the substantial differences existing between the absolute magnitudes of some parameters such as *Tensile Strength* and *Izod Impact Strength*, as shown in Table 5.5. The *Tensile Strength* values were set as the reference basis and the normalisation procedure was carried out on the other parameters relative to this property. The equation used to obtain the normalised values is given by:

$$V_n = V_o \left(\frac{\sum\limits_{n=1}^{7} R_n}{\sum\limits_{n=1}^{7} P_n} \right); \qquad \textbf{\textit{Equation} (5.4)}$$

Where,

V_n = The normalised value.

V_o = The original average value of the parameter is being normalised.

R_n = The average value of the parameter taken as reference (*Tensile Strength* in this case).
P_n = The average value of the parameter is being normalised.
n = Material group considered.

The normalised mechanical property values are shown in Table 5.6.

An example of the application of equation 5.4, using information displayed in Table 5.5 to obtain the first normalised score (***477***) of the parameter Yield Strength shown in Table 5.6, is given below.

$$V_n = V_o\left(\frac{\sum_{n=1}^{7} R_n}{\sum_{n=1}^{7} P_n}\right) = 280\left(\frac{\sum 452 + \ldots + 928}{\sum 280 + \ldots + 545}\right) = 280\left(\frac{4806}{2826}\right) = 477;$$

Table 5.6 - Normalised Mechanical Property Values For Steel Material Groups.

Material	Tensile	Normalised	Normalised	Normalised
1	452	***477***	532	471
2	519	553	514	539
3	631	593	679	609
4	646	685	719	693
5	732	728	689	788
6	898	843	798	803
7	928	927	875	902

iii) For each material group the sum of the normalised material property values was taken. Table 5.7 shows these new results.

Table 5.7 - Total scores of normalised mechanical properties

Material Group Number	Sum of Material Properties
1	1932
2	2125
3	2512
4	2743
5	2937
6	3342
7	3632

The values obtained above will form the basis to make the similarity analysis between the different material groups.

iv) Based on the scores in Table 5.7 a new normalisation process is carried out assuming that the similarity between two records having the same material group must have a score of unity. Table 5.8 shows these new values.

The indexes in Table 5.8 constitute identifiers of each material group to associate the values of the other groups relative to the particular material group being analysed. For example, the column "**Index 1**" represents how far is each material group from material group *1* in terms of percentage value. Table 5.8 represents a symmetric matrix where the distance is the same between material groups *i* and *j* than between *j* and *i*, so, the upper-right values are redundant.

Table 5.8 - Individual Normalisation of Material Groups (Percentage).

Material Group	Normalised values of material properties relative to each material group						
	Index 1	**Index 2**	**Index 3**	**Index 4**	**Index 5**	**Index 6**	**Index 7**
1	100						
2	110	100					
3	130	118	100				
4	142	129	109	100			
5	152	138	117	107	100		
6	173	157	133	122	114	100	
7	188	171	145	132	124	109	100

v) Based on the relative distance values shown in Table 5.8, relative similarity values between the different material groups were generated. These values are shown in Table 5.9, where the first column shows the resulting values from subtracting any two material groups.

The middle row values of one hundred (100) in Table 5.9 indicates a perfect similarity when comparing two records having the same material group, while the value equal to zero (0) in the same row indicates the result of subtracting these two material groups.

Table 5.9 - Linearisation of the Machinability Index Scores For Regression Analysis.

	Index set as a pivot element for each material group						
Reference values between Material Groups	Index 1	Index 2	Index 3	Index 4	Index 5	Index 6	Index 7
-6							12
-5						27	29
-4					48	43	55
-3				58	62	67	68
-2			70	71	83	78	76
-1		90	82	91	93	86	91
0	100	100	100	100	100	100	100
1	90	82	91	93	86	91	
2	70	71	83	78	76		
3	58	62	67	68			
4	48	43	55				
5	27	29					
6	12						

Example, when material groups *1* and *7* are being compared, the resulting values correspond to the last row in Table 5.9 (numbers 6 and 12). The number six (6) is obtained simply by subtracting both material groups under comparison (7-1), when the material group *1* is assumed as the pivot element in the subtraction process, relative to the other material groups, otherwise the resulting remainder will be *-6*. The similarity value of twelve (12) is obtained considering the information shown in Table 5.8, column "*Index 1*" where material group *7* is eighty eight percent away from material group *1*, so, it implies a similarity of 12% (100-88). The remaining values in Table 5.9 were obtained applying the same criteria.

vi) A second order regression analysis was carried out to estimate the equation of the line formed by the values shown in Table 5.9. Figure 5.6 shows the results obtained for material groups *1* to *7* (super group *G1*). The values for the vertical axis were obtained calculating the average of those scores in each row from Table 5.9.

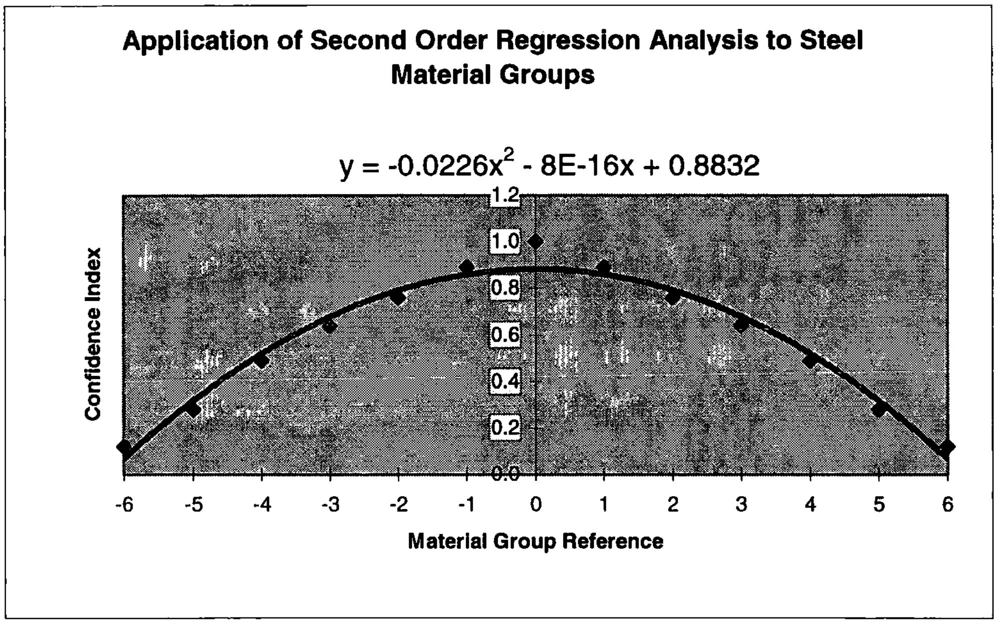


Figure 5.6 – Regression Analysis for Steel material groups

The six-step procedure already explained was equally applied to obtain the regression curves for the remainder material groups, the results being displayed in Figures 5.7 and 5.8.

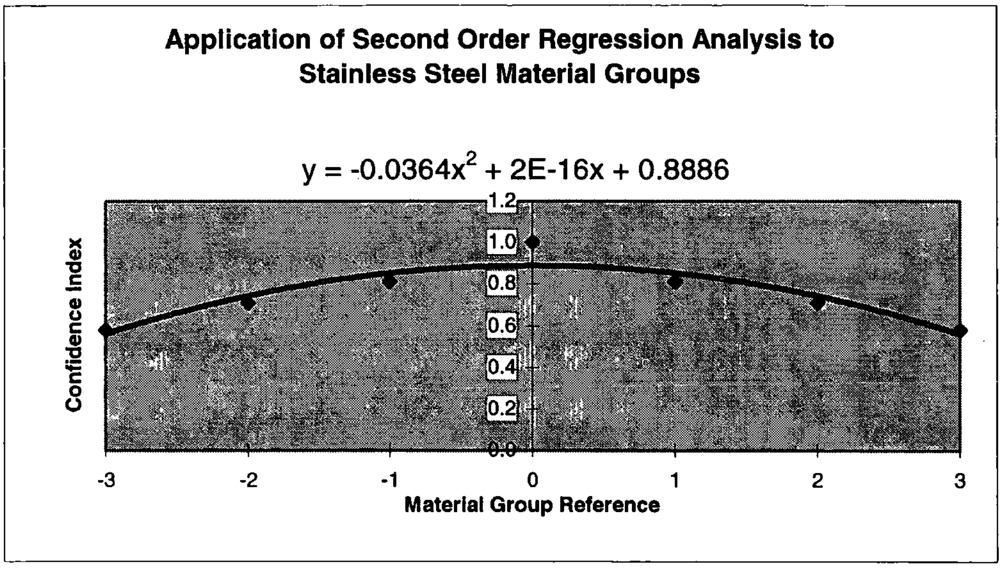


Figure 5.7 – Regression Analysis for Stainless Steel material groups

Once the *x* value is calculated by subtracting the material groups of any two records under analysis, this value can be used to obtain the similarity component between both material groups applying the equations 5.1 to 5.3.

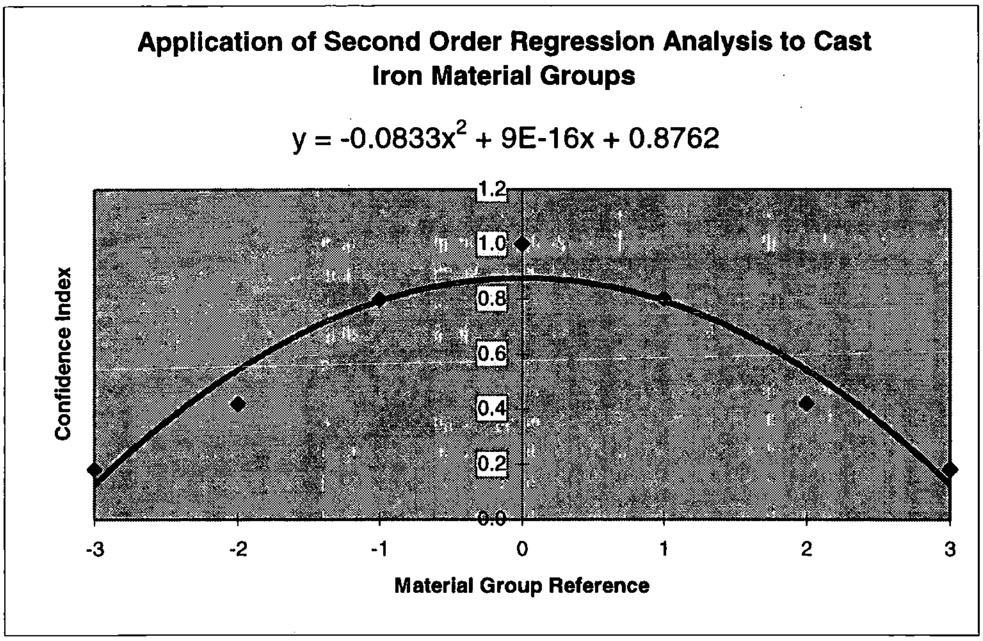


Figure 5.8 – Regression Analysis for Cast Iron material groups

A procedure to obtain the total similarity between two individuals, including the material similarity component calculated in this section is explained in Chapter 6, and regression analysis concepts are examined in Appendix D.

5.5.4 Data Standardisation

Standardisation constitutes a procedure applied when significant differences exist in the magnitude of the variables under analysis, or, when their variances differ to any great extent. These variables are usually described using different units such as *mm/rev*, *mm* and *m/min* to mention only a few.

In this particular research, a standardisation process was implemented due to substantial differences in the values of some variables, as shown in Table 5.10.

The new standardised values were obtained using the following expression:

$z_{ik} = \frac{x_{ik}}{S_k}$; Where,

z_{ik} = *i*th standardised value of the *k*th variable.

x_{ik} = *i*th original value of the *k*th variable.

S_k = Standard deviation for all values of the *k*th variable.

Table 5.10 – Sample of the total data set considered

Nose Radius (mm)	Cutting Speed (m/min)	Feed Rate (mm/rev)	Depth of Cut (mm)
0.4	180	0.15	0.5
0.8	240	0.65	4.0
.	.	.	.
.	.	.	.
.	.	.	.

The expression to represent a suitable standardisation measure is not unique, and there are some other approaches to variable standardisation. Milligan (1996) defined six additional standardisation measures, which are shown in Table 5.11.

Although Table 5.11 shows six different standardisation measures, proper criteria to their formal selection was not found in the literature consulted, particularly when generalisations based on studies in alternative data structures have not been yet established.

Table 5.11 - Additional standardisation measures

$z_1 = \frac{x - \bar{x}}{S};$	$z_2 = \frac{x}{Max(x)};$	$z_3 = \frac{x}{Max(x) - Min(x)};$
$z_4 = \frac{x - Min(x)}{Max(x) - Min(x)};$	$z_5 = \frac{x}{\sum x};$	$z_6 = Range\ (x);$
S = Standard deviation of variable *x*. *Range(x)* = Range of variable *x*.		

So, each analysis must be conducted taking into consideration the particular distribution existing in the data sets under study.

5.6 SUMMARY

In this chapter important concepts in the field of cluster analysis have been presented. It has been shown how cluster analysis can be classified as an important data mining technique in the wide spectrum of Knowledge Discovery process.

In addition, a formal methodology to support the development of clustering applications has been proposed. ***Kluster,*** a clustering tool developed in this research to analyse tooling trials data, has been successfully implemented using this methodology.

This Chapter has examined the fundamental functions included as part of the *Pre-processing* stage.

With these *Pre-processing* functions, the benefits of their application are clear, that is, they provide selection, cleaning and conversion mechanisms in order to prepare the data for a later generation of a robust dissimilarity matrix and the application of clustering algorithms.

Previous work on cluster analysis has primarily focused the problem of grouping data on the determination of a similarity measure, that is a function exclusively to the numerical values of the attributes being compared. It is unlikely that this analysis can be applied efficiently in domains where a significant amount of qualitative attributes are considered.

In this chapter, in order to minimise the lack of conceptual considerations when applying conventional clustering methods, conversion mechanisms to establish similarity relationships between some categorical attributes, were implemented.

The aim of the next chapter is to examine the *Processing* and *Post-processing* stages of the clustering methodology.

CHAPTER 6

CLUSTER ANALYSIS METHODOLOGY. PROCESSING AND POST-PROCESSING STAGES

6.1 INTRODUCTION

In the previous chapter, the first main stage of a formal clustering methodology has been presented. Here, the idea of examining the remaining structure of this methodology is extended. This chapter will go on to complete the analysis, describing the *Processing* as well as the *Post-processing* stages.

Four functions constitute the *Processing* stage. A first function evaluates the role of external knowledge to support the process of calculating similarity values. Secondly, a consistent dissimilarity matrix is obtained. Next, clustering algorithms provide an initial classification of the data. In order to establish comparative analyses, four different clustering algorithms were implemented. Finally, an analysis about the determination of suitable *stopping rules*, one of the hardest phases in the whole clustering methodology, is carried out.

In relation to the *Post-processing* stage, three functions are examined. The definition of mechanisms to measure the quality of a final classification constitutes a first function. An optimisation phase follows, in order to determine which cluster provides the most closely matching data, according to user's requirements. The last phase involves the analysis of the results, according to the domain being studied.

6.2 PROCESSING STAGE

The main aim in the *Processing* stage is to obtain an initial classification of the data, mainly through the application of clustering algorithms. However, additional issues must be considered and they are examined as follows.

6.2.1 Learning Modes

Learning modes in the context of clustering analysis, are those criteria whose incorporation in some key stages of the whole classification process can affect the final results. These learning strategies are mainly concerned with the level of participation of external agents to guide the discovering process.

Although clustering analysis is primarily supported by unsupervised learning techniques, where natural associations and interesting unpredictable patterns should be revealed without the participation of external agents to guide the discovering process, there is an alternative worthwhile case to discuss.

Unsupervised Mode. In this mode the level of participation of final users is scarce or nil. Unsupervised learning algorithms concentrate their action on finding patterns without the incorporation of external knowledge (labelled data). In case users provide any input information, this information is not taken into consideration to calculate similarity or distance values, and in general, to guide the clustering process. Examples of systems applying this strategy are those based on algorithms to analyse *DNA* structures or astronomical data, where final users are more interested in obtaining novel results about the unpredictable structure of the data sets than in incorporating prefixed conditions that should spoil the knowledge discovery process.

Supervised Mode. This mode allows involving the end users in the analysis, in order to make the knowledge discovery process more interactive and highly collaborative. This is the strategy commonly adopted for systems where the participation of end users is crucial and the obtained results are closely influenced by their inputs. For instance, determining the importance of some variables in respect of others, establishing certain input conditions or acceptance levels and defining threshold values to control the clustering process. All systems managing customer transactions, sales and market research, buying patterns, banking risks, production data and medical diagnosis, are good examples of systems applying supervision learning modes.

In a clustering process there are three additional stages which can be affected by external knowledge, **a)** when applying a particular clustering method, **b)** when implementing a stop criterion and **c)** when final users are interested in post-processing tasks to optimise a first classification.

Jain and Dubes (1988) provide a summary of strategies for cluster validation, closely related to the learning modes implemented: "*External criteria* measure performance by matching a clustering structure to a priori information ... *Internal criteria* assess the fit between the structure and the data, using only the data themselves ... *Relative criteria* decide which of two structures is better in some sense, such as being more stable or appropriate for the data".

In the field of unsupervised learning, Shen and Leng (1996) presented an interesting approach to combine a *"Metapattern"* (also known as metaquery) generator with an existing human-directed discovery loop in order to build an integrated data mining system that can automatically provide useful feedback and interaction for human users. The most significant contribution of their work, is the notion of mettapaterns and its role in automatically exploiting the interdependencies between induction, deduction and human guidance. The automated discovery loop also provides an algorithm that can learn relational patterns directly from databases without requiring humans to pre-label the data as positive or negative examples for some given target concepts.

In this research, unsupervised and supervised modes were considered. In unsupervised learning mode the clustering process is carried out taking a group of variables considered relevant to the domain under study, without the assignation of conditions, weights or other external contributions (labelled data). In supervised learning, acceptance levels introduced by the user guide the final process of calculating a similarity matrix. These acceptance levels are expressed in terms of percentage values for the variables *Grade, Material Group and Nose Radius*. Likewise, the users have the option to decide about the inclusion of three tooling parameters named *Feed Rate, Cutting Speed and Depth of Cut.*

In the next section, a complete explanation about the obtaining of a final matrix representing the mathematical distance between all individuals in the database is presented.

6.2.2 Generation of Similarity/Dissimilarity Matrix

This section presents two dominant elements in cluster analysis, *Similarity* and *Dissimilarity* (distance) between the individuals of a data set to be clustered.

6.2.2.1 Similarity

Similarity is a key factor in any classification study. It indicates the main purpose of grouping individuals based in analysing their most distinguishing characteristics.

The notion of similarity makes sense as the researcher can identify the dominant role of some variables that can contribute to determine a closeness degree between any two individuals in a data set. Therefore, similarity is a context-dependent measurement in which attributes incorporating some level of differentiation must be clearly defined.

In classification analysis, the term S_{ijk} is usually utilised to represent the similarity between two individuals *i* and *j* to be clustered, according to the analysis of the *k*th variable. So, in a similarity measurement S_{ij}, it is necessary to consider the partial contribution of all the variables under analysis.

A maximum similarity value $S_{ij} = 1$ indicates the strongest closeness degree between two individuals and a minimum value $S_{ij} = 0$ indicates a total lack of similarity.

Once the raw data is standardised, the next step is to obtain the similarity matrix. Here, this similarity matrix was generated in order to calculate the similarity measures for variables of mixed type, just as we have in our tooling database and taking into consideration the assignments of relative weights to certain variables of interest in the analysis. The similarity values will constitute the basis to obtain the distance matrix described later in this section.

In order to calculate the similarity measures an improved version of the Gower's coefficient was implemented. The similarity coefficient suggested by Gower (1966) is given by:

$$\delta_{ij} = \frac{\sum_{k=1}^{p} W_{ijk} S_{ijk}}{\sum_{k=1}^{p} W_{ijk}}; \qquad \textit{Equation}\ \mathbf{(6.1)}$$

Where W_{ijk} is a binary weigh taking values 1 or 0, depending if the comparison is considered valid for the k_{th} variable. S_{ijk} is the similarity between the i_{th} and j_{th} individuals as measured by the k_{th} variable and its expression is given by $S_{ijk} = 1 - \frac{|x_{ik} - x_{jk}|}{R_k}$; where x_{ik} and x_{jk} are the two individuals' values for variable *k*, and R_k is the range of variable *k*.

The following example illustrates the use of Gower's coefficient (Everitt,1993).

Table 6.1 – Medical data sample

	Weight (pounds)	Anxiety Level	Depression present ?	Hallucination present ?	Age Group
Patient 1	120	Mild	No	No	Young
Patient 2	150	Moderate	Yes	No	Middle
Patient 3	110	Severe	Yes	Yes	Old
Patient 4	145	Mild	No	Yes	Old
Patient 5	120	Mild	No	Yes	Young

Table 6.1 shows the data for five psychiatric ill patients. In this case it is supposed that the investigator has excluded negative matches on depression and hallucinations variables. The similarity between patients *1* and *2* is then calculated using the Gower's coefficient given by *Equation* 6.1:

$$\delta_{12} = \frac{1x\left(1 - \frac{30}{40}\right) + 1x0 + 1x0 + 0x1 + 1x0}{1 + 1 + 1 + 0 + 1} = 0.0625;$$

The similarity factor S_{ijk} is appropriate for quantitative variables, but for categorical variables Gower assigns the value one when the two individuals have the same value (*Hallucination*), and zero otherwise (*Anxiety level*, *Depression* and *Age group*), which gives the same treatment to several pair qualitative categories (Mild-Moderate and Mild-Severe). This criterion aggregates a certain degree of ambiguity to the measurement.

In this research an improved version was implemented, where categorical variables such as *Grade* and *Material Group* were numerically transformed and, therefore, it was possible to apply the coefficient as suggested for quantitative variables.

The calculation of the similarity between any two individuals is based on the sum of six (6) partial similarity components, corresponding to the six variables considered key in the analysis. For quantitative variables such as *Nose Radius, Cutting Speed, Depth of Cut and Feed,* the Gower's coefficient has been applied without alterations. The factor *Weight* (W_{ijk}) in *Equation* 6.1 takes values 1 or 0, depending if the variable is considered in the analysis for the k_{th} variable.

In the case of the variable *Grade*, when both grades under comparison have the same value, their similarity is one (1). If one of them does not appear in Table 5.3, their similarity is equal to zero (0). If both of them can be found, the conversion is based on the analysis of the workpiece material which each insert grade has the ability to machine, calculating the overlapping areas as explained in the previous Chapter.

In the case of the variable *Material Group*, when two groups belong to the same super group (*2* and *5,* for example), a regression analysis taking into consideration some important mechanical properties of the constituent materials for each material group, was conducted, as explained in the previous Chapter. If both groups belong to different super groups (2 and 9, for example) their similarity is equal to zero (0).

If one of the material groups is found in the database with a value equal to zero (0), a comparison using the field "*material*" is established. If both materials have the same value, the similarity is equal to one (1) and zero (0) otherwise.

Once a similarity analysis for all pairs of individuals in the database was carried out, a similarity matrix was generated.

6.2.2.2 Dissimilarity

Dissimilarity (distance) is the opposite concept to similarity. The lower the similarity between two individuals, the higher the level of their distance.

Dissimilarity or distance measures (***d**$_{ij}$*) can be transformed into similarity measures (δ_{ij}) or vice versa. Two main transformations have proved to be useful (Gordon, 1981):

i) $\delta_{ij} = c - d_{ij}$, where *c* is a constant usually equal to one.

ii) $\delta_{ij} = 1/(1 + d_{ij})$.

Also, for binary data Gower (1966) suggested that the expression $d_{ij} = \sqrt{2(1 - \delta_{ij})}$ could be used to perform a good transformation.

When a data set contains only quantitative variables, the Euclidean distance has been widely used, which belongs to the family of Minkowski's distances given by: $\|X - Y\| = \sqrt[p]{\sum_{i=1}^{n} | x_i - y_i |^p}$, so, if the index $p = 2$, we obtain the Euclidean distance given by: $\|X - Y\| = \sqrt{\sum_{i=1}^{n} | x_i - y_i |^2}$.

In this particular research, once the similarity matrix was obtained, the distance matrix necessary to apply a later clustering method was generated applying the expression given in **ii)**: $d_{ij} = [(1/\delta_{ij}) - 1]$.

Where $0 < \delta_{ij} \leq 1$ and, d_{ij} = distance between the individuals *i* and *j*.

6.2.3 Clustering Method

In relation to clustering methods, there are two dominant approaches based on *Hierarchical* and *Objective Function-Based* techniques. Hierarchical methods are best known and they have had a wider applicability. In this research only hierarchical methods will be considered, the reader interested in knowing more details about *Objective Function-Based Optimisation* techniques can find many applications consulting Bezdek (1981) and Cios et al. (1998).

Six different hierarchical methods are summarised as follows:

i) Single linkage method: It is also known as the *nearest neighbour algorithm* (Everitt, 1993). If *C1* and *C2* are two clusters under analysis, the similarity between them is calculated based on the minimal distance between the closest pair of patterns belonging to *C1* and *C2.* If *C1* and *C2* have been merged in a new cluster labelled *N*, the similarity S_{NT} between *N* and a third cluster *T* is given by:

$S_{NT} = \text{Min}\ (S_{C1\text{-}T}, S_{C2\text{-}T})$.

ii) Complete linkage method: This case is opposite to the Single linkage method in the sense that the basis for calculating the similarity S_{NT} between clusters *N* (obtained by merging *C1* and *C2*) and *T*, relies on the maximum distance between their patterns.

$S_{NT} = \text{Max}\ (S_{C1\text{-}T}, S_{C2\text{-}T})$.

iii) Average linkage method: Here the similarity S_{NT} between clusters *N* and *T* is calculated based upon the average of the distances between their patterns (Cios et al., 1998),

$$S_{NT} = \frac{1}{card(N)card(T)} \sum_{x\in N, y\in T} \|x - y\| .$$

iv) Centroid-based method: In this case the similarity S_{NT} between clusters *N* (obtained by merging *C1* and *C2*) and *T* is calculated based upon the mean values of the variables of *C1* and *C2* versus the values of the variables of *T*.

v) Median-based method: This case is similar to the former centroid-based method but using the median instead the mean. The use of the median eliminates the disadvantage of the main-based methods when joining groups of very different sizes, where the centroid of a newly generated group will be close to that of the larger group and the influence of the smaller group can be diluted significantly.

vi) Ward's method: Ward (1963) proposed a clustering procedure using the sum of squared errors to calculate the loss associated during each grouping process, joining those pairs of clusters whose fusion resulted in a minimum increase of information loss. An interesting equation that applies this concept to calculate the dissimilarity $d_{T(C1,C2)}$ between clusters *N* (obtained by merging *C1* and *C2*) and a new cluster *T*, was later presented by Anderson (1966). This equation is given by the expression:

$$d_{T(C1,C2)} = \frac{n_T + n_{C1}}{n_T + n_{C1} + n_{C2}} d_{TC1} + \frac{n_T + n_{C2}}{n_T + n_{C1} + n_{C2}} d_{TC2} - \frac{n_T}{n_T + n_{C1} + n_{C2}} d_{C1C2};$$

Where,

d_{ij} = distance between individuals *i* and *j*.

n_i = cardinality of cluster *i*.

A masterly summary of the former six clustering methods in terms of their parameter values has been made by Anderson (1987), which is shown in Table 6.2.

Table 6.2 – Clustering methods and their parameters

Clustering Method	α_j	β	λ
Single linkage	$\frac{1}{2}$	0	$-\frac{1}{2}$
Complete linkage	$\frac{1}{2}$	0	$\frac{1}{2}$
Average linkage	$\frac{w_i}{w_i + w_j}$	0	0
Centroid-based	$\frac{w_i}{w_i + w_j}$	$\frac{- w_i w_j}{(w_i + w_j)^2}$	0
Median-based	$\frac{1}{2}$	$-\frac{1}{4}$	0
Ward's method	$\frac{w_i + w_k}{w_+}$	$\frac{- w_k}{w_+}$	0

The parameters shown in Table 6.2 are useful to calculate the dissimilarity between a cluster obtained by merging C_i and C_j ($C_i \cup C_j$) and a new cluster C_k, applying an equation proposed by Lance and Williams (1966) given by:

$$d(C_i \cup C_j, C_k) = \alpha_i d(C_i, C_k) + \alpha_j d(C_j, C_k) + \beta d(C_i, C_j) + \lambda \left| d(C_i, C_k) - d(C_j, C_k) \right|$$

In this research, *Simple, Complete, Average and Ward's* clustering methods were implemented. For the sake of complementing the theoretical basis, a step-by-step development of a full clustering exercise applying these four methods, is presented in Appendix A.

6.2.4 Stop Criteria

One of the main bottlenecks in cluster analysis is the stop criterion (known also as cluster validation). It is necessary to define mechanisms to know when an optimum number of clusters have been got. The problem of deciding on the appropriate number of clusters for the data usually implies the definition of threshold values, measurements of association between elements of the same cluster (intracluster closeness) and establishment of independence criteria between clusters (intercluster separation).

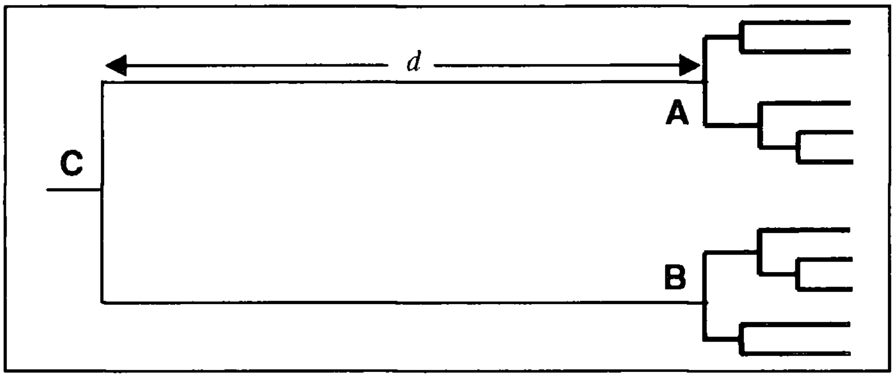


Figure 6.1 – Visual recognition of a particular classification.

Visualisation techniques are currently gaining popularity in identifying evidence that shows obvious or easy grouping decisions. Figure 6.1 displays a particular dendrogram presenting a long distance (*d*) before joining two groups, *A* and *B,* and a natural association between the elements within each group, which could be taken as a good indicator to leave groups *A* and *B* separated.

In this research, two different stop criteria were implemented, **i)** Mojena's stopping rule and, **ii)** Threshold defined by users.

i) *Mojena's stopping rule.* This procedure, proposed by Mojena (1977), is based upon the relative sizes of the different distance fusions. The idea is to stop the fusion process and therefore, select the number of groups already found, when the following condition is satisfied:

$Z_{i+1} > \bar{Z} + kS_Z$;   Where,

Z_0, Z_1,..., Z_{n-1} are the fusion levels corresponding to stages with n, n-1, ...1 clusters.

$\bar{Z}$= The mean of the Z values.

S_Z= The standard deviation of the Z values.

k = constant.

Values of k in the range 1.0 to 5.0 were used. The best performance was obtained for $4.0 < k < 4.5$. For values of $1.0 < k < 4.0$, too many groups were generated and, for $4.5 < k < 5.0$, the number of groups was too small to provide reliable conclusions.

The reader may note the fact that the good results fixing k in the range 4.0 to 4.5, are from this particular study. Mojena suggests that values of k in the range 2.75 to 3.50 give the best results. However, Milligan and Cooper (1985) got good results for values of k = 1.25.

ii) *Threshold defined by users.* Here, the user specifies the number of clusters. After initial investigation, an interval between 15 and 50 clusters was considered suitable for selection.

While the threshold is closest to 50, the classification is more restrictive, because the elements inside groups present a stronger similarity between them, which favours performing more accurate analyses. In contrast, fixing the number of clusters close to 15, it is possible to group more elements per cluster, which allows the establishment of more flexible criteria to match the parameters specified by users with those present in the data set.

6.3 POST-PROCESSING STAGE

The main aims in the *Post-Processing* stage are firstly, to evaluate the quality of the classification previously obtained by applying clustering algorithms; secondly, implement procedures to recognise optimum clusters and finally, interpret and test the final results. These three phases are examined as follows.

6.3.1 Performance Evaluation

To measure the final results obtained through the application of particular clustering methods is not an easy task, specially when there is not a widely accepted standard definition of the term "*Quality*" of the cluster, when the participation of users is key to assess the utility and interpretability of these final results, and further, when the clustering methods always produce a classification, even if this constitutes an inappropriate representation of the data.

Everitt (1993) identifies the following three distinct types of comparisons:

- The solutions to be compared arise from the use of different clustering methods on the same data set (this is the case to be analysed later in this section).

- The solutions have been obtained from the same clustering method applied to different similarity or distances matrices arising from the raw data.

- The different solutions have been obtained applying the same clustering method to the same proximity matrix derived from data sets taken from different sources.

Some attempts have been made in the past to measure the quality of the clusters, assigning a certain numerical value to quantify in a certain manner the quality of the obtained final results, which in its turn, allow the comparison of the efficiency of different clustering methods implemented. Also, some parameters have been identified as fundamental to measure the term *Quality* of a cluster, which offer mathematical support to compare the solutions obtained through the application of different clustering methods. As already mentioned, the contribution of the users to validate the final results is relevant, and it constitutes an essential complement to the numerical analysis.

Some authors (e.g. Williams, 1965) have considered the assessment of classification processes through the generation of hypotheses, which have to be tested on the basis of utilising new data, but this approach requires a high level of knowledge about the domain under study.

A formal procedure for comparing classifications has been suggested by Rand (1971), which expression may be written as:

$$R_k = \frac{\left[T_k - \frac{1}{2}P_k - \frac{1}{2}Q_k + \binom{n}{2}\right]}{\binom{n}{2}};$$ ***Equation* (6.2)**

Where:

$$T_k = \sum_{i=1}^{k}\sum_{i=1}^{k} m_{ij}^2 - n; \qquad P_k = \sum_{i=1}^{k} m_{i.}^2 - n; \qquad Q_k = \sum_{i=1}^{k} m_{.j}^2 - n;$$

The term *n* is the total number of individuals to be clustered. The quantity m_{ij} is the number of individuals in common between the *i*th cluster of the first solution, and the *j*th cluster of the second. The terms $m_{i.}$ and $m_{.j}$ are appropriate marginal totals of the matrix of m_{ij} values. For a given number of clusters, *k*, the index R_k can be interpreted as the probability that two individuals are treated alike in both solutions. R_k lies in the interval (0,1) and takes its upper limit when there is complete agreement between the two classifications.

In this research, a different procedure to measure the quality of a final classification was implemented. This method is based on the calculation of two important parameters called *Intracluster Cohesion Index* (***ICI***) and *Intercluster Separation Level* (***ISL***). The *Intracluster Cohesion Index* can be used to measure the cohesion within clusters, that is to say, it is an indicator of how closely grouped are the individuals that are part of the same cluster. On the other hand, the *Intercluster Separation Level* is a measure of how distant are the individuals of one cluster in respect of the individuals of a different cluster found for the same data set.

The quality of a final classification based on numerical analyses, denoted $\boldsymbol{Q_c}$, applying a particular clustering method *C*, can be expressed in terms of ***ICI*** and ***ISL*** parameters, where the criteria to choose the best value of $\boldsymbol{Q_c}$ are based on high intracluster cohesion and high intercluster separation levels.

The following part of this section explains the method developed in this investigation to calculate the parameters ***ICI*** and ***ISL*** as well as the quality of each classification obtained through the application of the four clustering methods implemented.

6.3.1.1 *Intracluster Cohesion Index (ICI)*

The method implemented to calculate the *Intracluster Cohesion Index* (***ICI***) is based on a relative distance criterion given by the expression:

$$ICI = \frac{\sum_{w=1}^{k} D_w}{DT}; \qquad \textbf{\textit{Equation}} \ \textbf{(6.3)}$$

Where:

$$D_w = \sum_{i=1}^{s-1} \sum_{j=i+1}^{s} d_{ij}; w = 1...k; \qquad \textbf{\textit{Equation}} \ \textbf{(6.4)}$$

$$DT = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij}; \qquad \textbf{\textit{Equation}} \ \textbf{(6.5)}$$

Being:

D = The sum of all distances in a particular cluster.

DT = The sum of all distances in a whole data set for the *n* individuals.

d_{ij} = distance between the individuals *i* and *j*.

n = total number of individuals to be clustered.

s = number of individuals in a particular cluster.

k = number of clusters in the final classification.

The parameter ***ICI*** lies in the interval (0,1), where a result closest to zero indicates a better cohesion within the cluster. To illustrate the application of this method let us suppose we have an initial data set of seven individuals and let us also suppose that two clusters, *C1* and *C2*, were obtained in a final classification, applying a clustering method *A*.

The relative distances and the resulting classification are shown in Figure 6.2.

Applying *Equation (6.4)* the following results are obtained:

$\boldsymbol{D_1} = d_{13} + d_{15} + d_{16} + d_{35} + d_{36} + d_{56} = 2.10;$

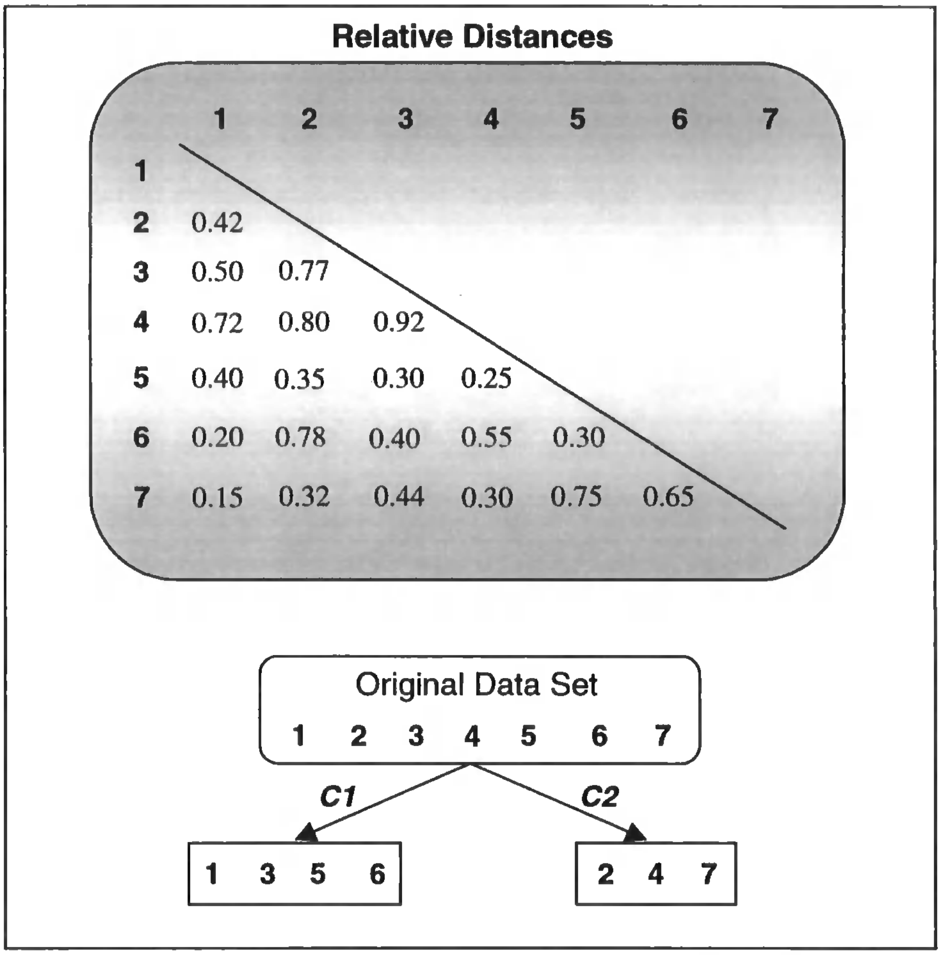
$$\boldsymbol{D_2} = d_{24} + d_{27} + d_{47} = 1.42;$$

And applying *Equation (6.5)*, ***DT*** is given by:

$$\boldsymbol{DT} = \sum_{i=1}^{6} \sum_{j=i+1}^{7} d_{ij} = 10.27;$$

Then, applying *Equation (6.3)* the *Intracluster Cohesion Index* is obtained:

$$\boldsymbol{ICI} = \frac{2.10 + 1.42}{10.27} = \mathbf{0.34};$$



Relative Distances

	1	2	3	4	5	6	7
1							
2	0.42						
3	0.50	0.77					
4	0.72	0.80	0.92				
5	0.40	0.35	0.30	0.25			
6	0.20	0.78	0.40	0.55	0.30		
7	0.15	0.32	0.44	0.30	0.75	0.65	

Figure 6.2 - Dissimilarity matrix and basic test data set

The value of the parameter ***ICI*** obtained according to the previous procedure can be used as a reference to compare different clustering methods applied to the same data set, in terms of internal cluster cohesion. That method whose ***ICI*** results closest to zero, will be the best in terms of intracluster cohesion. In section 6.3.1.3 a method to calculate the *quality* of a final classification, which include the parameter ***ICI***, is proposed.

The main advantages of applying this method are:

- The analysis considers the simultaneous effect of all the variables that describe the characteristics of each individual.

- It is computationally easy to implement.

- It is possible to compare the internal cohesion between two different clusters. Clusters highly cohesive are hoped to contain individuals sharing similar properties or common patterns, which constitutes encouraging results to the researcher.

The presence of mixed variables (as categorical and quantitative) do not incorporate an additional complexity in the calculations, because all the distances have been previously obtained taking into consideration the conversion mechanisms explained in Chapter 5.

However, some disadvantages have been identified:

- The processing time to calculate the factor *DT* increases considerably, as the number of individuals to be clustered is increased in a high proportion. For example, when the number of individuals *n = 500*, there are 124.750 distances to be added (*n(n-1)/2*), but when *n = 10.000*, the number of distances is equal to 49.995.000.

- It is necessary to keep a register of the relative distances of all individuals to be clustered.

*6.3.1.2 Intercluster Separation Level (**ISL**)*

The method implemented to calculate the *Intercluster Separation Level* (***ISL***) takes into consideration the minimum and maximum distances between all the clusters in the final classification. Its expression is given by:

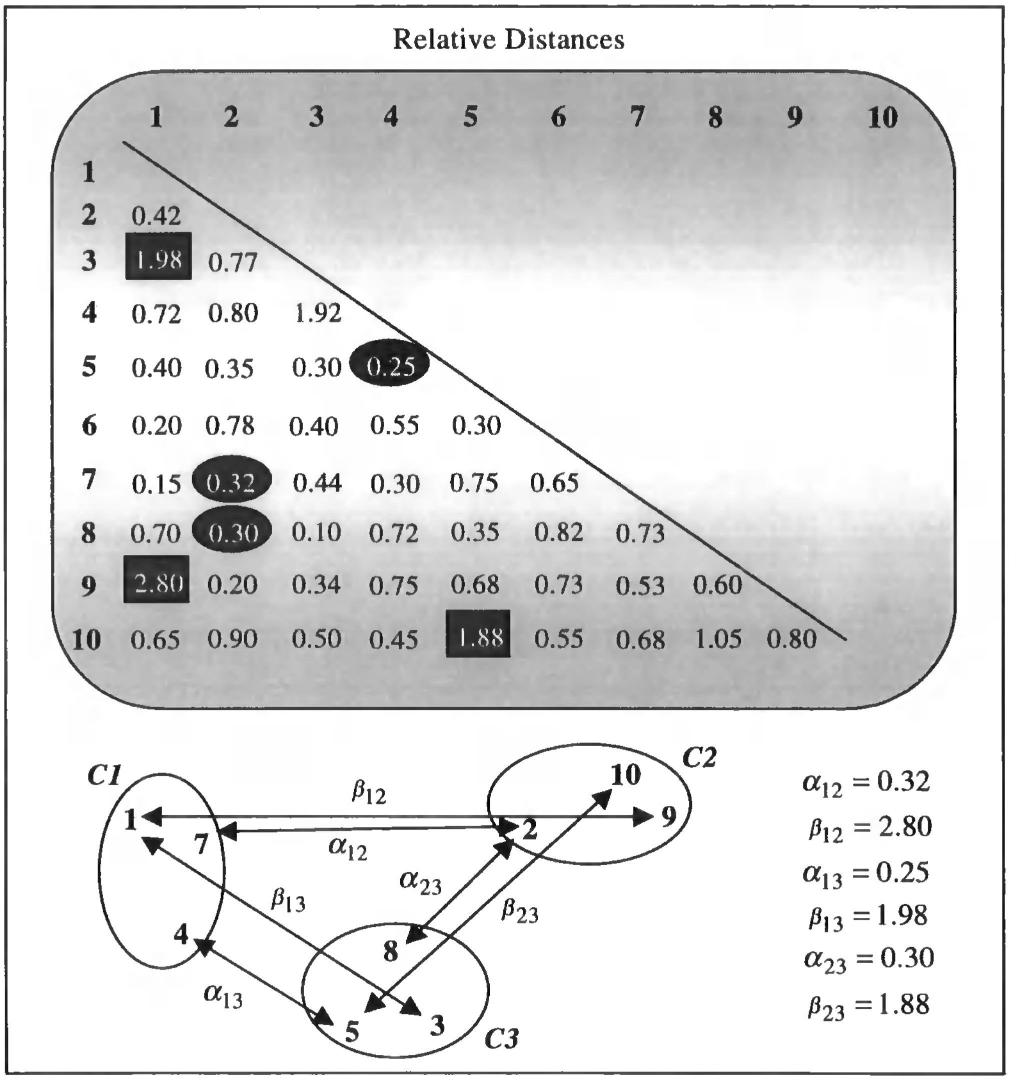
$$ISL = \frac{\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \left(\alpha_{ij} + \beta_{ij}\right)}{k(k-1)}; \qquad \textbf{\textit{Equation (6.6)}}$$

Where:

α_{ij} =Smallest distance between clusters *i* and *j*.

β_{ij} =Biggest distance between clusters *i* and *j*.

k = Number of clusters in the final classification.



Relative Distances

	1	2	3	4	5	6	7	8	9	10
1										
2	0.42									
3	1.98	0.77								
4	0.72	0.80	1.92							
5	0.40	0.35	0.30	0.25						
6	0.20	0.78	0.40	0.55	0.30					
7	0.15	0.32	0.44	0.30	0.75	0.65				
8	0.70	0.30	0.10	0.72	0.35	0.82	0.73			
9	2.80	0.20	0.34	0.75	0.68	0.73	0.53	0.60		
10	0.65	0.90	0.50	0.45	1.88	0.55	0.68	1.05	0.80	

$\alpha_{12} = 0.32$
$\beta_{12} = 2.80$
$\alpha_{13} = 0.25$
$\beta_{13} = 1.98$
$\alpha_{23} = 0.30$
$\beta_{23} = 1.88$

Figure 6.3 - Dissimilarity matrix and considered distances.

To illustrate the application of this method let us suppose we have an initial data set of ten individuals, and let us also suppose that three clusters, *C1*, *C2* and *C3*, were obtained in a final classification applying a clustering method *A*.

The relative distances and the resulting classification are shown in Figure 6.3.

Applying Equation (6.6) the *Intercluster Separation Level* is given by,

$$\boldsymbol{ISL} = \frac{(0.32+2.80)+(0.25+1.98)+(0.30+1.88)}{3(3-1)} = \mathbf{1.26};$$

In the next section, a method to calculate the *quality* of a final classification, which include the parameters ***ICI*** and ***ISL***, is proposed.

6.3.1.3 Quality of Final Classification (Q_c)

Once the parameters ***ICI*** and ***ISL*** have been calculated, as shown in the two previous sections, the quality of a final classification $\boldsymbol{Q_c}$, given the application of a *k*th clustering method, can be obtained through this expression:

$$Q_C = \frac{\left[\frac{m_C}{ICI_C} + \frac{ISL_C}{M_C}\right]}{2}; \qquad \textbf{\textit{Equation}}\ \mathbf{(6.7)}$$

Where $\boldsymbol{ICI_C}$ and $\boldsymbol{ISL_C}$ are the Intracluster Cohesion Index and the Interclustering Separation Level respectively, corresponding to the *c*th clustering method implemented; m_C is the ***ICI*** lowest value chosen between all the clustering methods implemented. Finally, M_C is the ***ISL*** highest value chosen between all the clustering methods implemented.

The numerical constant *2* has been included in order to obtain a value of Q_C lying in the interval (0,1). That clustering method whose value of Q_C is closest to 1, will be the best in terms of the intracluster proximity and intercluster separation distance criteria.

To illustrate the procedure utilised to calculate Q_C, let us assume that three different clustering methods have been applied to the same data set, and the parameters ***ICI*** and ***ISL*** have been calculated according to the *equations (6.3)* and *(6.6)* respectively. Figure 6.4 shows the quality of each final classification and the selection of the best clustering method, on the basis of hypothetical ***ICI*** and ***ISL*** values found.

Clustering Method *A*	$ICI_A = 0.45$ $ISL_A = 4.00$	$\longrightarrow Q_A = \left[\frac{0.20}{0.45} + \frac{4.00}{4.30}\right] / 2 = \mathbf{0.687}$
Clustering Method *B*	$ICI_B = 0.30$ $ISL_B = \mathbf{\mathit{4.30}}$	$\longrightarrow Q_B = \left[\frac{0.20}{0.30} + \frac{4.30}{4.30}\right] / 2 = \mathbf{0.833}$
Clustering Method *C*	$ICI_C = \mathbf{\mathit{0.20}}$ $ISL_C = 4.00$	$\longrightarrow Q_C = \left[\frac{0.20}{0.20} + \frac{4.00}{4.30}\right] / 2 = \mathbf{0.965}$

Figure 6.4 - Measuring the quality of the final clusters

The best classification corresponds to the application of the clustering method ***C***, with a quality $Q_c = 0.965$.

A further equation to calculate the quality Q_C was considered. It is given by the expression:

$Q_C = 1 - \frac{ICI}{ISL}$. However, the results in some test cases did not reflect a substantial differentiation when comparing two methods having marked performance differences.

6.3.2 Optimisation Phase

In order to determine which cluster provides the most closely matching data, according to the user's requirements, an optimisation procedure would be carried out. Figure 6.5 shows how after a final data classification is obtained, according to the quality criterion proposed in the previous section, an optimisation phase would be applied to produce better clustering results.

To carry out this optimisation phase, three different procedures were implemented, and are described as follows.

(a) *Similarity Index.* Given a number of *n* clusters, a similarity index determines how closely certain parameters specified by the user match the same parameters in the clusters under consideration. The highest value of this index identifies the best cluster, in terms of its closest matching with the specifications introduced by the user.

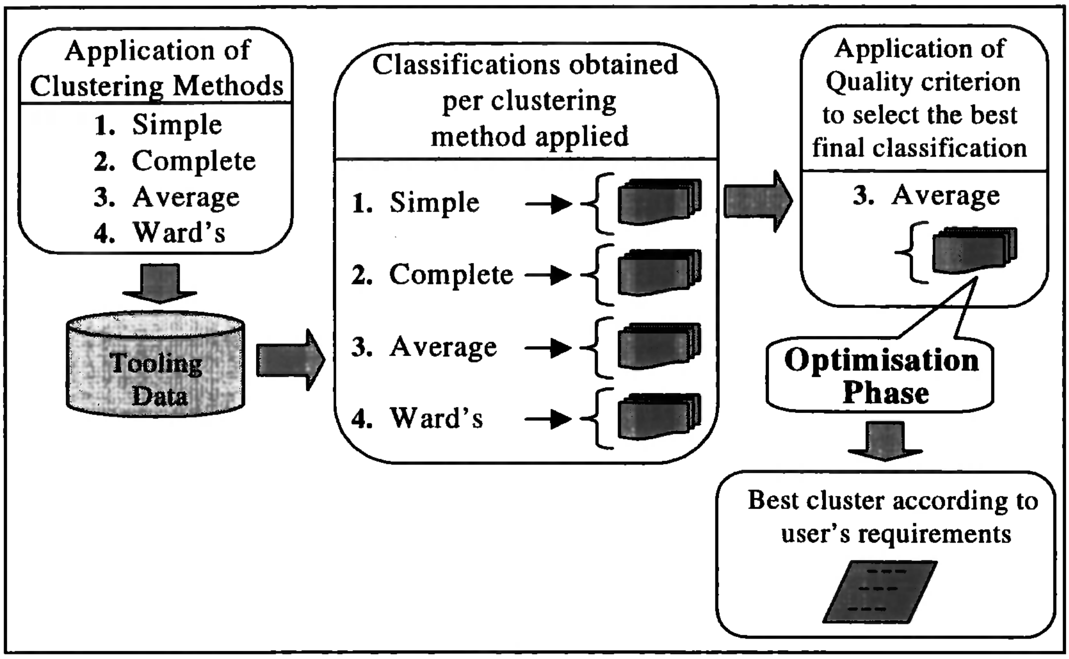

Application of Clustering Methods
1. Simple
2. Complete
3. Average
4. Ward's
Tooling Data
Classifications obtained per clustering method applied
1. Simple
2. Complete
3. Average
4. Ward's
Application of Quality criterion to select the best final classification
3. Average
Optimisation Phase
Best cluster according to user's requirements

Figure 6.5 – Optimisation phase after application of performance evaluation

The idea is to calculate a similarity index per cluster, based on comparisons between each variable introduced by the user and the same variable in the cluster under consideration. This procedure is applied to all elements belonging to each cluster. The expression to calculate the similarity index (SI) per each cluster is given by:

$$SI_z = \frac{\sum_{i=1}^{k}\left(\sum_{j=1}^{t} sim(Val_{ij} \rightarrow U_j)/t\right)}{k};$$ Where,

k = number of elements in a cluster.

t = number of variables considered, 1 = *Material group*, 2 = *Grade*, 3 = *Nose radius*.

z = cluster under analysis, z = 1, ..., n.

$sim(Val_{ij} \rightarrow U_j)/t$ = similarity between the variable *j* corresponding to the element *i* of the cluster under consideration, and the variable *j* introduced by the user.

(b) *Partial Match Index.* In this case the criterion to select the best cluster, is based on an approximation factor (*AF*) given by:

$AF_z = \frac{FM_z}{PM_z}$; Where,

FM = number of full matches found considering all the variables introduced by the user.

PM = number of partial matches found, including those cases where at least one of the variables introduced by the user, matches the same variable in the cluster under analysis.

z = cluster under analysis, z = 1, ..., n.

This method allows more flexibility to find result sets, when no exact matches are found between the information provided by users and the information contained in the clusters.

(c) *Full Match Index.* In this case the criterion to select the best cluster is similar to the above case, but now the approximation factor (*AF*) only considers full matches between the variables selected by the user and the same variables in the cluster under analysis.

$AF_z = FM_z$;

This frequency-based criterion is more restrictive than the two previous methods, and would be used when the user is interested in precise information searches.

6.3.3 Analysis of Results and Tests

This last phase involves the interpretation of the results according to the domain of the applied problem. The usefulness of any clustering method is incomplete if the final data classification can not be appropriately interpreted. Moreover, a set of tests should be carried out to determine the scope and the validity of the obtained results.

In order to facilitate the interpretation of clustering results, many researchers implement graphical representations, such as dendrograms and decision trees structures. Currently, because of the advantages of high power computer graphics, data visualisation techniques seem to be very effective presentation and analysis tools.

Visualisation tools take advantage of human perception as a method for analysis and data interpretation. What numbers can not show to users, suitable pictures often can. For instance, a linear trend in data might not be evident from a table of data. However, a scatterplot which shows a series of points lined up on a straight line, provides clearer insight into data relations.

In this research, graphical methods were used to support the interpretation of the clustering results. These methods, as well as a complete testing structure, are presented in chapter 8.

6.4 SUMMARY

A formal clustering methodology was introduced in the previous Chapter, and a first *Pre-processing* stage was examined. In this Chapter, the remaining two main stages of the whole methodology, based on *Processing and Post-processing* functions, have been explored. Based upon this methodology, not only has a formal cluster analysis been presented, but its implementation has supported the development of ***Kluster***, a clustering-based data mining tool that will be described in the next chapter.

The *Processing* stage presented here incorporates two important contributions, namely, unsupervised and supervised *learning modes* and the generation of a *dissimilarity matrix* from the analysis of qualitative variables. Also, two original contributions have been incorporated in the *Post-processing* stage. Specifically, consistent indicators to measure the *quality of the final classification* and application of *optimisation methods* to the final data groups obtained.

Four different hierarchical clustering methods were applied, *Simple*, *Complete*, *Average* and *Ward's* linkages. In this context, the *Average linkage* was the optimum method in terms of a ***Quality*** (***Q_c***) parameter obtained according to the calculation of two important factors, the Intracluster Cohesion Index (***ICI***) and Intercluster Separation Level (***ISL***).

In the next chapter, ***DISKOVER***, an integrated KDD-based system developed under a distributed philosophy and which incorporates clustering, fuzzy-clustering and SQL-based techniques, is examined.

CHAPTER 7

KNOWLEDGE DISCOVERY APPLICATIONS

7.1 INTRODUCTION

In the previous two chapters the fundamental issues of Cluster Analysis have been examined. It has been shown how clustering constitutes one of the most important techniques currently applied for knowledge discovery purposes. Furthermore, a formal clustering methodology was proposed, showing significant improvements over traditional procedures, which do not explicitly consider, for example, *quality* measurements and *optimisation* methods applied to the final data classification obtained.

Here, ***DISKOVER***, a KDD-oriented system developed by the author and another colleague for analysing tooling data will be described. This system integrates clustering, fuzzy-clustering, rough sets programs and SQL-based exploratory data analysis methods, under a unified Internet-based architecture and graphical interface.

7.2 MOTIVATION FOR APPLYING KDD TECHNOLOGY TO THE TOOLING INDUSTRY

Tooling data is continuously generated using the machining centres of manufacturing companies. To improve operational methods, explore new machining procedures and establish tool selection criteria, different tests are regularly conducted. This information is usually registered manually and later stored in tooling databases.

The manual analysis of this information presents several problems:

i) The reports are analysed in an isolated way, without establishing useful relationships among them.

ii) As the amount of data increases, it is difficult to manually discover common patterns, deviations or significant regularities useful to the analyst.

iii) It is not easy to know if another tooling engineer has previously carried out tool trials. Hence, the execution of some trials could be redundant.

iv) It is not easy to obtain information from a set of previously performed tests, when the user is interested only in a few relevant parameters.

In order to address the above situations, a system called ***DISKOVER*** has been developed by the author and another colleague.

7.3 *DISKOVER*, AN INTEGRATED KDD-ORIENTED SYSTEM

From the discussion in chapter 4, it should be clear that there is no universally best KDD-oriented system across all application domains. An approach to increasing the robustness of KDD-oriented applications is to use an integrated architecture, applying different kinds of algorithms and/or hybrid algorithms to a given data set, to maximise the efficacy of the discovery process. Examples of these hybrid architectures can be found in (Brachman and Anand, 1994).

Therefore, current developments in the area of KDD technology must be able to provide flexible schemes, at the initial stage of the problem, allowing the incorporation of complementary and multidisciplinary solutions.

DISKOVER is a context-oriented KDD-based system that has been developed considering the implementation of multiple data mining techniques. In contrast to the two Internet-based systems described in chapter 3, which were built using *Java-applets*, ***DISKOVER*** is also an Internet-based system, but built using a *Java-application*.

One of the main advantages of using *Java-applications* is that they do not need to be embedded in HTML files, and, consequently, downloaded through conventional browsers, which is a distinguishing characteristic of *Java-applets*.

DISKOVER integrates five main modules:

- *Kluster*[6], a clustering application.
- *Q-Fast*[6], an SQL-based Exploratory Data Analysis (EDA) application.
- *Fuzzy-K*[7], an application combining fuzzy sets with clustering analysis.
- *MQG*[8], a multiple query generator application.
- *R-Set* [8], a rough sets-based application.

The first three modules are described as follows. A summary of the last two modules is given at the end of this section.

As indicated of the five modules two have been developed entirely by the author. Of the remaining three one was developed jointly by the author and another research student in the group at Durham. The further two modules are the sole work of the other PhD. student.

7.3.1 *KLUSTER*, a Clustering Application

Kluster is an application developed according to the clustering methodology explained in chapters 5 and 6. This section presents its functionality and in the next chapter results are discussed. Figure 7.1 displays a first input screen.

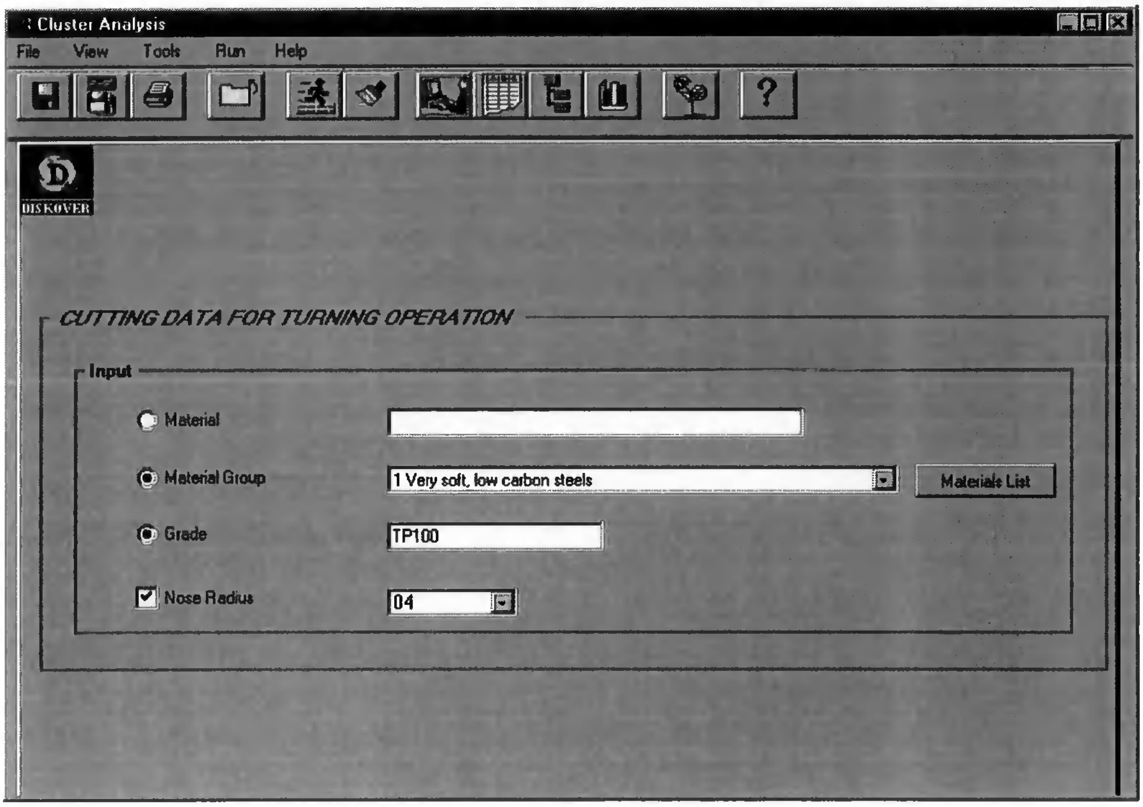
After due consideration it was decided to allow users to introduce information about four tooling parameters: *Material, Material Group, Grade and Nose Radius*. The analysis of *cutting condition* parameters and relationships between *grades* and *material workpiece* was considered as of major concern, mainly due to their impact on tool selection criteria.

After submitting the input parameters, the user is required to fill five options, in order to define the conditions under which the system will be run. These options are shown in Figure 7.2 and explained as follows.

[6] Author's contribution

[7] Joint contribution

[8] Colleague's contribution



CUTTING DATA FOR TURNING OPERATION

Input

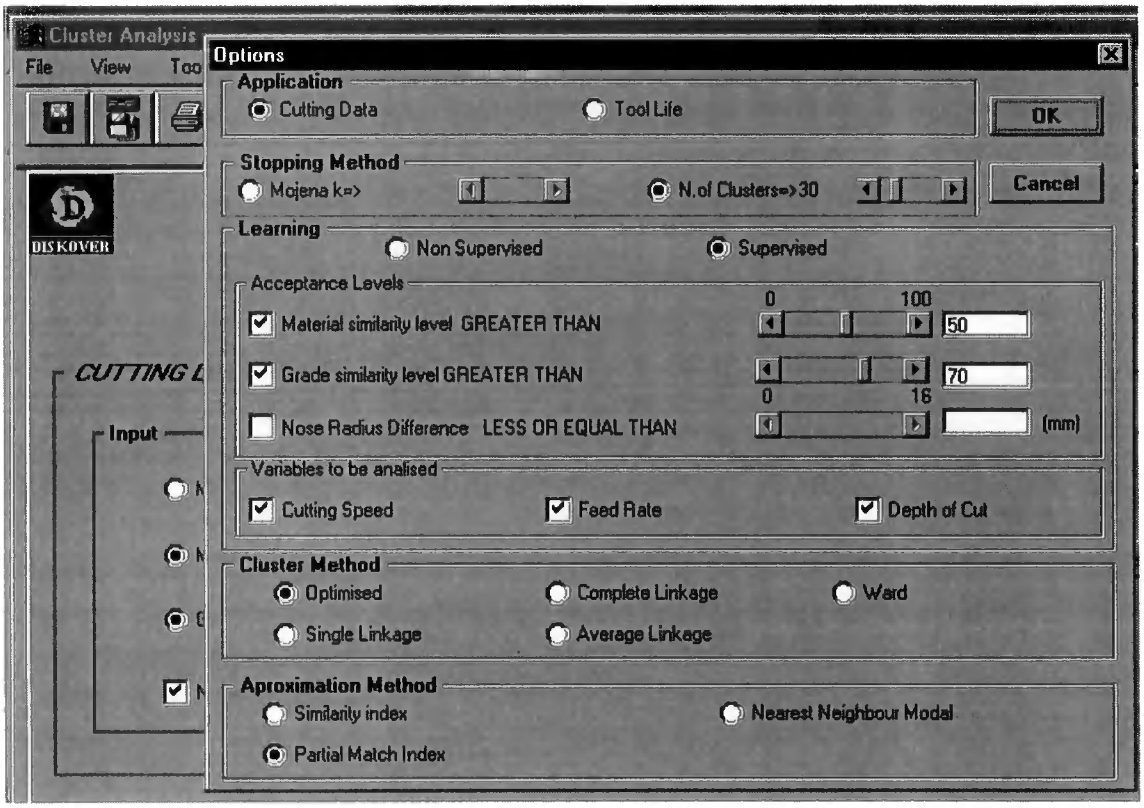
○ Material

● Material Group: 1 Very soft, low carbon steels — Materials List

● Grade: TP100

☑ Nose Radius: 04

Figure 7.1 – Input screen for Kluster.



Application

● Cutting Data ○ Tool Life

Stopping Method

○ Mojena k=> ● N.of Clusters=>30

Learning

○ Non Supervised ● Supervised

Acceptance Levels

☑ Material similarity level GREATER THAN 0 100 50

☑ Grade similarity level GREATER THAN 70 0 16

□ Nose Radius Difference LESS OR EQUAL THAN (mm)

Variables to be analised

☑ Cutting Speed ☑ Feed Rate ☑ Depth of Cut

Cluster Method

● Optimised ○ Complete Linkage ○ Ward

○ Single Linkage ○ Average Linkage

Aproximation Method

○ Similarity index ○ Nearest Neighbour Modal

● Partial Match Index

Figure 7.2 – Setting the run environment of Kluster.

7.3.1.1 Application

This option focuses on the analysis and grouping of *cutting data* variables, specifically, *cutting speed, feed rate* and *depth of cut* values, while identifying the values of alternative parameters considered during the execution of the tooling trials, such as *material workpiece, insert grade* and *nose radius.*

Tool life constitutes an alternative option proposed to support tool life prediction and cutting data optimisation. Its development is left as a future implementation. The decision to include this module will depend on the companies' specific considerations and the incorporation of new data, as tool life coefficients.

7.3.1.2 Stopping Method

Two options are considered. The Mojena's stopping rule, based upon the relative sizes of the different distance fusions, and an empirical and flexible approach based on a numerical threshold defined by users.

In relation to Mojena's criterion, the best performance was found for values of constant *k* in the range 4.0 to 4.5, as explained in chapter 6. By default, the system fixes 30 clusters as the stopping rule. This value has been established after conducting experiments for several threshold values, which are discussed in the next chapter.

7.3.1.3 Learning Modes

This option allows the participation of users to guide the classification process. In the case of *unsupervised* mode, the system runs without assigning prefixed weights to tooling parameters. When *supervised* mode is activated, the user is allowed to introduce the relative importance of certain tooling parameters, which affect the calculation of similarity values, and hence, the final classification obtained.

7.3.1.4 Cluster Method

Kluster is based in the application of four hierarchical clustering methods known as *Simple*, *Complete*, *Average* and *Ward's.* The system is able to run applying any of these methods separately. Also, the system provides the option

Optimised, which allows running the four clustering methods and selecting the optimum method in relation to a quality criterion based on a *performance evaluation* analysis already explained in chapter 6.

A comparative analysis of the performance of the four clustering methods is carried out in the next chapter.

7.3.1.5 Approximation Method

After a particular clustering method is applied, a finite number of clusters are generated. An optimisation phase follows, to choose the best cluster considering the importance (weight) given by users to certain tooling parameters. The system provides three different methods to carry out this analysis, namely, *similarity index, partial match index* and *full match index.* An explanation of these methods is given in chapter 6 (optimisation phase).

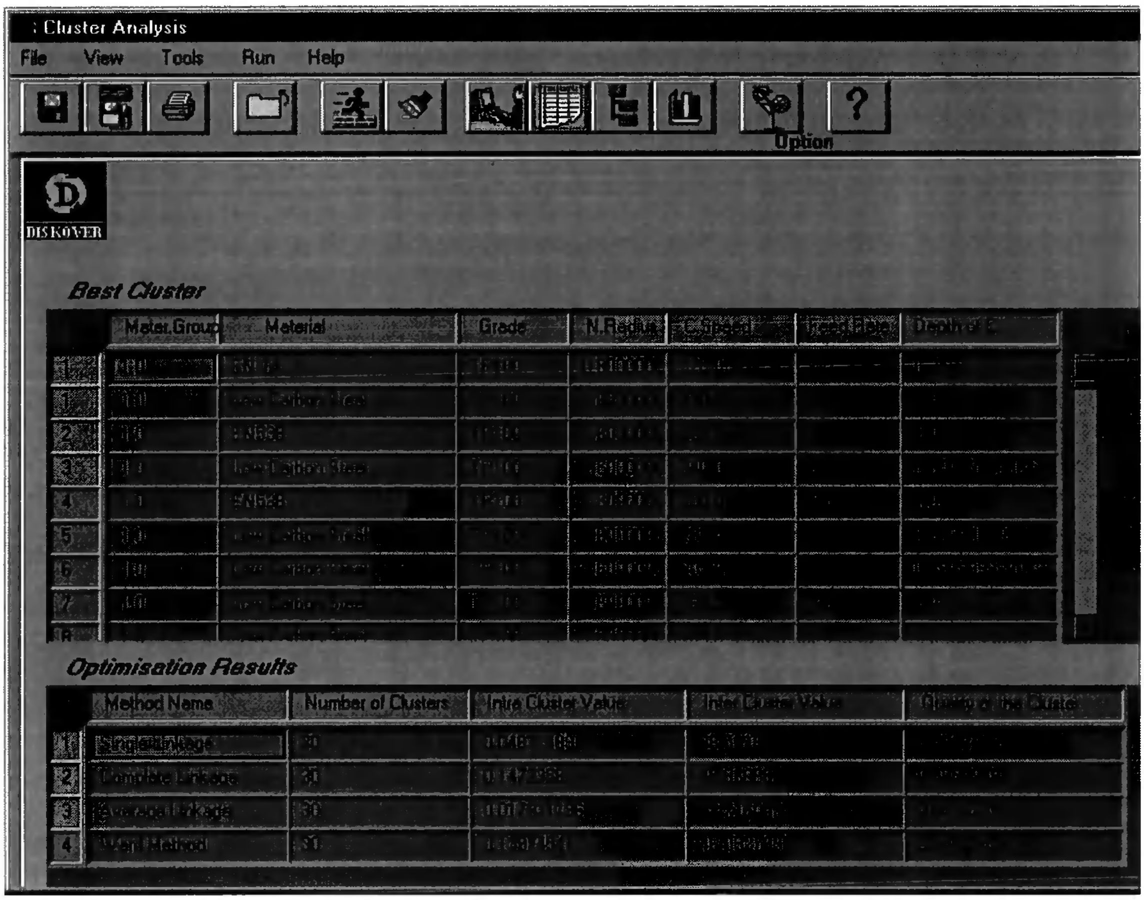
By default this option is disabled and only when the user selects a supervised learning mode, the option is enabled. Figure 7.3 illustrates a typical output once the *partial match index* optimisation option is selected.

In all cases, the cluster method applied, as well as the stopping rule and performance evaluation parameters are shown.

7.3.2 A Fuzzy-Clustering Application

Most of the techniques found in the literature in fuzzy-clustering are based on objective function-based methods (Huang & Ng, 1999) and (El-Sonbaty & Ismail, 1998). However, attempts to exploit the advantages of hierarchical clustering while maintaining fuzzy clustering rules, have recently been undertaken (Geva, 1999).

The *k*-means [(Ball and Hall, 1967), (MacQueen, 1967)] as well as *c*-means (Bezdek, 1973) algorithms are well known for their efficiency in clustering large data sets. Fuzzy versions of the *k*-means algorithm (fuzzy *k*-means) have been reported in (Ruspini, 1969) and later in (Bezdek, 1980). Likewise, Fuzzy versions of the *c*-means algorithm (fuzzy *c*-means) have been reported in (Gustafson and Kessel, 1979) and (Bezdek, 1981).



Best Cluster

	Mater.Group	Material	Grade	N.Radius	C.Speed	[illegible]	Depth of C
1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
1	[illegible]	Low Carbon Steel	[illegible]	[illegible]	[illegible]		[illegible]
2	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]		[illegible]
3	[illegible]	Low Carbon Steel	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
4	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
5	[illegible]	Low Carbon Steel	[illegible]	[illegible]	[illegible]		[illegible]
6	[illegible]	Low Carbon Steel	[illegible]	[illegible]	[illegible]		[illegible]
7	[illegible]	Low Carbon Steel	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
8	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Optimisation Results

	Method Name	Number of Clusters	Intra Cluster Value	Inter Cluster Value	Quality of the Cluster
1	Single Linkage	30	[illegible]	[illegible]	[illegible]
2	Complete Linkage	30	[illegible]	[illegible]	[illegible]
3	Average Linkage	30	[illegible]	[illegible]	[illegible]
4	Ward Method	30	[illegible]	[illegible]	[illegible]

Figure 7.3 – Typical output screen after running the optimisation phase.

In this work, a fuzzy-clustering application named *Fuzzy-K* was developed. It has the advantages of hierarchical clustering, while applying fuzzy membership functions to support the generation of similarity measures. In this particular context, the implementation of fuzzy membership functions helps to optimise the grouping of categorical data containing missing or imprecise values.

The main approaches to deal with missing values fall into the following categories (Liu *et al.*, 1997):

a) Data having missing values are deleted from the data set, or, if missing values occur in some attributes very often, these attributes are deleted from the entire data set.

b) Missing values are imputed iteratively during the execution of the fuzzy clustering algorithm.

c) To choose a method in the data analysis process that tolerates missing values.

Currently the most common way to deal with missing values in fuzzy cluster analysis is data pre-processing. Data with missing values is deleted, corresponding attributes are removed, or these missing values are imputed, before or during the execution of a fuzzy clustering algorithm. The consequence is that we have to choose between two unsatisfactory solutions. If a data set contains a high percentage of missing values, it is not possible to impute these missing values with a high reliability. If data with missing values are removed from the data sets, the classification we obtain becomes less reliable the higher the percentage of data having missing values (Timm and Kruse, 1998).

Another simple and quick method of dealing with missing data is completing the missing values using the arithmetic mean of the existing field values (Howard and Rayward-Smith, 1999). This approach is favoured when the data analysed have been taken over a long period of time (meteorological data, for example). In this case, the values obtained using the arithmetic mean are expected to be relatively reliable. A clear disadvantage of this method is its difficulty in dealing with categorical data.

To overcome the problem of missing data, this work focuses on the third category (**c**)) already indicated. In this context, *Material* and *Grade* were two attributes identified as requiring particular attention in relation to missing or invalid values. Other attributes presenting imprecise values were identified, such as *Machine Condition* (having values such as *good, average* and *poor*) and *Reasons for Ending Test* (having values as *insufficient power*), but these parameters were considered of low relevance to be included in the analysis.

The selection of a suitable membership function has a critical impact on the whole process of assigning elements to groups or categories. Klir and Folger (1988) affirm that the usefulness of a fuzzy set for modelling a conceptual class or a linguistic label depends on the appropriateness of its membership function. Therefore, the practical determination of an accurate and justifiable function for any particular situation is of major concern. The methods proposed for accomplishing this have been largely empirical and usually involve the design of experiments on a test population to measure subjective perceptions of membership degrees for some particular conceptual class.

Although they can take similar values, fuzzy membership grades are not probabilities. For example, the summation of probabilities on a finite universal set *must* be equal to 1, whilst this is not necessarily true for membership grades.

Figure 7.4.a shows a membership function for the fuzzy set of real numbers close to zero. It is possible to generalise this function in a family of functions representing the set of real numbers close to any given number *a* as follows (Klir and Folger, 1988):

$$\mu_A(x) = \frac{1}{1+10(x-a)^2}$$

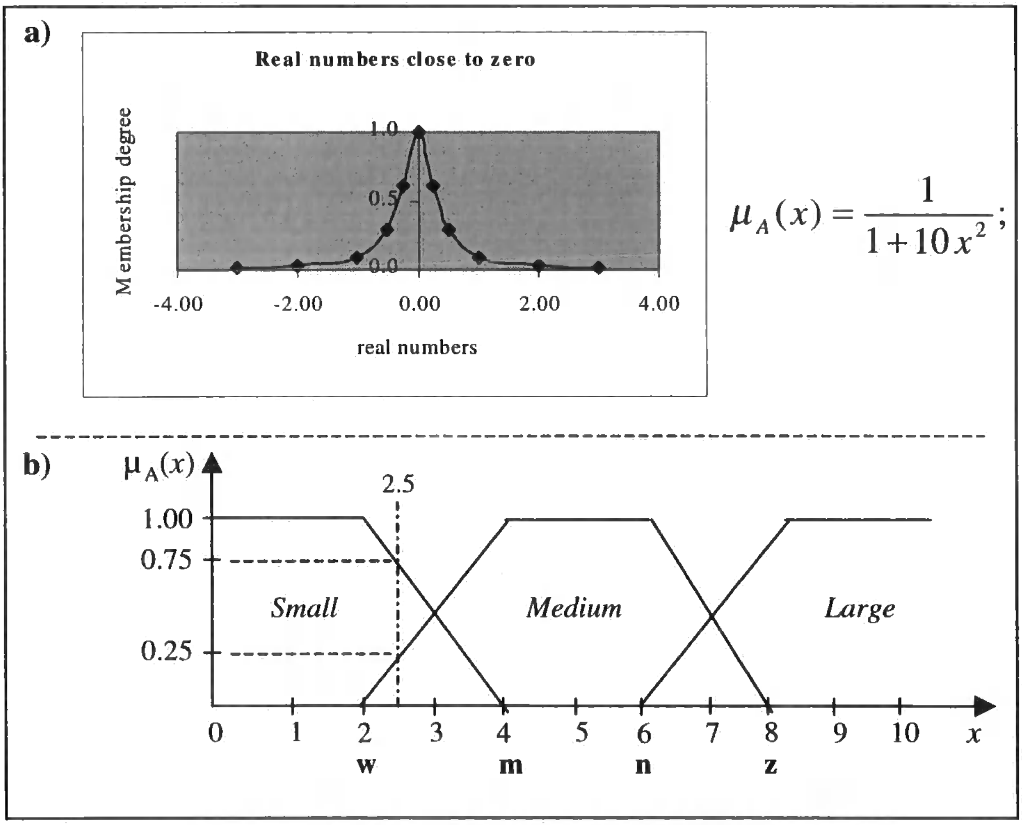


Figure 7.4 – Examples of Gaussian and Trapezoidal membership functions.

Another choice for a fuzzy membership function is a linear trapezoidal function (Welstead, 1994). Figure 7.4.b displays some examples for the fuzzy sets *Small, Medium* and *Large*. The membership function for the fuzzy set *Medium* is given by:

$$\mu_A(x) = \begin{cases} 0 & \text{If } x < \mathrm{w} \\ \dfrac{x-w}{m-w} & \text{If } x \in [\mathrm{w}, \mathrm{m}] \\ 1 & \text{If } x \in [\mathrm{m}, \mathrm{n}] \\ \dfrac{z-x}{z-n} & \text{If } x \in [\mathrm{n}, \mathrm{z}] \\ 0 & \text{If } x > \mathrm{z} \end{cases}$$

The membership functions for fuzzy sets *Small* and *Large* can be derived as particular cases of the former expression.

In Figure 7.4.b the point 2.5 is a member of *Small* to degree 0.75 and a member of *Medium* to degree 0.25. Likewise, the point 5 is a member of *Medium* to degree 1 and a member of *Large* to degree 0.

Fuzzy sets concepts have already been introduced in chapter 4, therefore, the rest of this section will focus mainly on their applicability in this context, rather than on additional theoretical considerations.

Instead of the procedures adopted by *Kluster* and explained in the previous chapter, two aspects have been optimised in this research by implementing fuzzy membership functions to calculate new similarity values:

- With the previous method, the calculation of the similarity between two materials representing the same group (2 versus 2, for example) was smaller than one (1), and not exactly one, as would be expected from this comparison. The equation implemented to calculate the partial similarity between two materials belonging to the *super group 1* is shown below.

$s_{ijk} = -0.0226x^2 - 8E - 16x + 0.8832$; If these materials have the same value, the value of the variable *x* in the above equation is equal to zero (0) and therefore, the partial similarity for the parameter "material group" will be equal to 0.8832, and not one (1), as expected in this case.

With the incorporation of fuzzy membership functions, it will always be assigned a membership degree equal to one (1), if two elements have the same values.

- The partial similarity component obtained for two materials having the same numerical difference interval, was the same. For example, when comparing material groups 1 and 4, 3 and 6, or 4 and 7, their interval difference is the same (3). Hence, this value is assigned to the variable *x* in the similarity equation, producing a partial similarity component equally, for all of the three cases mentioned.

In contrast, applying fuzzy membership functions, the partial similarity components for the three above instances were obtained considering a direct influence of the mechanical properties belonging to the material groups analysed and, hence, they do not necessarily converge to the same value.

The procedure of obtaining new partial similarity components for the variable *Material Group,* applying fuzzy membership functions, follows.

7.3.2.1 Analysis of Materials to obtain numerical similarity values

The procedure to calculate new similarity values is described as follows.

i) Based upon the information of four mechanical properties for steel material groups, shown in Table 5.5, Chapter 5, a similarity analysis considering the degree material groups are related to each other, was undertaken. Table 7.1 shows the mean values corresponding to these properties.

Table 7.1 – Mechanical properties for Steel Material Group.

	Tensile Strength (MPa)	Yield Strength (MPa)	Izod Impact Strength (J)	Hardness (HB)
Material Group 1	452	280	23	142
Material Group 2	519	325	22	162
Material Group 3	631	349	30	183
Material Group 4	646	403	31	209
Material Group 5	732	428	30	237
Material Group 6	898	496	35	242
Material Group 7	928	545	38	272

ii) For each material group, a fuzzy set was defined. A Membership function was then formulated to determine the membership grades of each fuzzy set. The membership function (discrete values) is given by the expression:

$$\mu_A = 2 - \left[\frac{\left(TS_j / TS_i\right) + \left(YS_j / YS_i\right) + \left(IIS_j / IIS_i\right) + \left(H_j / H_i\right)}{4} \right]$$; Where,

μ_A = Membership function by which a fuzzy set A is defined.

A = {MG1, MG2, ...MG7}, MG = Material Group.

TS_j, TS_i = *Tensile Strength* values corresponding to the *j*th and *i*th materials under comparison.

YS_j, YS_i = *Yield Strength* values corresponding to the *j*th and *i*th materials under comparison.

IIS_j, IIS_i = *Izod Impact Strength* values corresponding to the *j*th and *i*th materials under comparison.

H_j, H_i = *Hardness* values corresponding to the *j*th and *i*th materials under comparison.

iii) Based on the membership grades found for each fuzzy set, a similarity matrix for comparing material groups was obtained. The fuzzy sets and their membership grades are shown as follows.

MG 1 = [1.00, 0.90, 0.73, 0.58, 0.49, 0.31, 0.13];
MG 2 = [0.90, 1.00, 0.84, 0.71, 0.63, 0.47, 0.31];
MG 3 = [0.73, 0.84, 1.00, 0.88, 0.81, 0.67, 0.53];
MG 4 = [0.58, 0.71, 0.88, 1.00, 0.94, 0.81, 0.69];
MG 5 = [0.49, 0.63, 0.81, 0.94, 1.00, 0.87, 0.76];
MG 6 = [0.31, 0.47, 0.67, 0.81, 0.87, 1.00, 0.90];
MG 7 = [0.13, 0.31, 0.53, 0.69, 0.76, 0.90, 1.00];

For example, the value *0.73* corresponding to the fuzzy set **MG 1**, indicates that this set (material group 1) has a membership grade of 0.73 with regard to material group 3. A graphical representation of the seven fuzzy sets is shown in Figure 7.5.

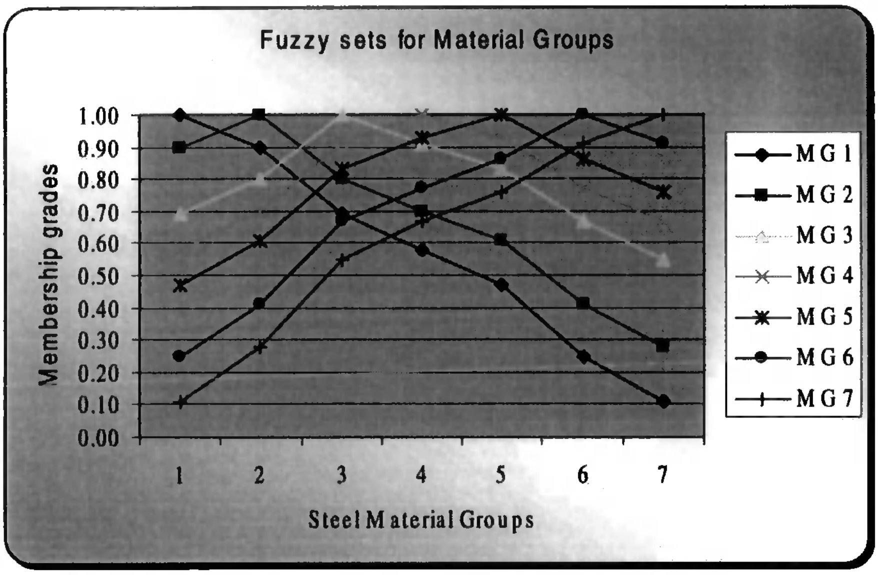


Figure 7.5 – Membership grades for Material Groups

It should be noted how two material groups (1 and 7, for example) within the same super group (Steel) have notable differences as indicated by their membership degree (0.13). Likewise, the reason why two material groups under comparison have a higher similarity since these materials belong to closer groups, is because the membership function was defined considering their mechanical properties.

7.3.2.2 Analysis of Grades to obtain numerical similarity values

The procedure to calculate similarity values described in chapter 5 was based on the analysis of the workpiece materials, which each insert grade has the ability to machine, according to three main groups of, toughness/hardness indicators (*P, M, K*). In this case similarity between two grades could take different values depending on the particular material groups considered.

Here, the definition of new similarity values is also dependent on the toughness/hardness indicators, but, a unique similarity relationship for all the twenty one grades analysed was established. Table 7.2 shows the fuzzy sets defined for *TX150* and *TP* grades family.

The values in Table 7.2 were obtained applying the same rules given in chapter 5 and derived from the analysis of all possible overlapping cases shown in Figure 5.5. In this case, the calculation of similarity values is obtained by calculating the average of the resulting similarity partial values derived for the three material properties identified as *P, M* and *K* (Table 5.3).

Table 7.2 – Similarity generated for TX150 and TP grades family.

	TP10	TP20	TP30	TP05	TP100	TP15	TP200	TP25	TP300	TP35	TP40	TX150
TP10	1.00	0.33	0.00	0.73	0.83	0.67	0.33	0.40	0.00	0.00	0.00	0.81
TP20	0.33	1.00	0.68	0.00	0.47	0.62	0.85	0.88	0.39	0.29	0.00	0.67
TP30	0.00	0.68	1.00	0.00	0.19	0.46	0.82	0.74	0.66	0.60	0.29	0.29
TP05	0.73	0.00	0.00	1.00	0.62	0.35	0.05	0.22	0.00	0.00	0.00	0.61
TP100	0.83	0.47	0.19	0.62	1.00	0.79	0.48	0.54	0.00	0.00	0.00	0.90
TP15	0.67	0.62	0.46	0.35	0.79	1.00	0.65	0.73	0.42	0.21	0.00	0.81
TP200	0.33	0.85	0.82	0.05	0.48	0.65	1.00	0.88	0.63	0.61	0.33	0.68
TP25	0.40	0.88	0.74	0.22	0.54	0.73	0.88	1.00	0.44	0.36	0.17	0.72
TP300	0.00	0.39	0.66	0.00	0.00	0.42	0.63	0.44	1.00	0.87	0.76	0.00
TP35	0.00	0.29	0.60	0.00	0.00	0.21	0.61	0.36	0.87	1.00	0.69	0.00
TP40	0.00	0.00	0.29	0.00	0.00	0.00	0.33	0.17	0.76	0.69	1.00	0.00
TX150	0.81	0.67	0.29	0.61	0.90	0.81	0.68	0.72	0.00	0.00	0.00	1.00

For example, the procedure to obtain the value of *0.65*, corresponding to the similarity between grades *TP200* and *TP15* is presented as follows.

i) The first step is the identification of the overlapping areas between both grades under comparison. Figure 7.6 illustrates this situation.

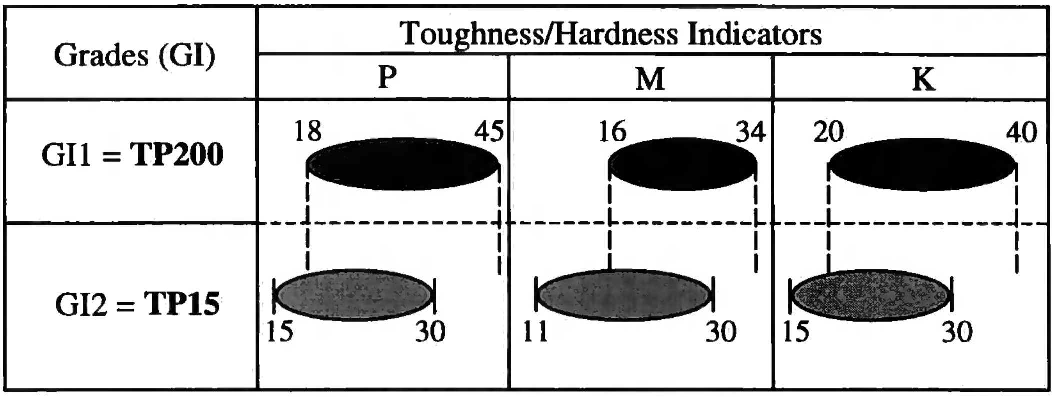


Figure 7.6 – Overlapping areas between grades TP200 and TP15.

ii) The selection of the appropriate similarity equation, for each of the three overlapping areas shown in Figure 7.6, follows. It is clearly evident how the three overlapping areas correspond to the case 3, as explained in chapter 5.

Therefore, the expression used to calculate the similarity for each indicator (P, M, K) is given by:

$$s_{ijk} = \frac{\left(\left(\frac{GI2.V\max - GI1.V\min}{GI2.V\max - GI2.V\min}\right) + \left(\frac{GI2.V\max - GI1.V\min}{GI1.V\max - GI1.V\min}\right)\right)}{2};$$

Similarity for ***P*** indicator:

$$s_{ijk} = \frac{\left(\left(\frac{30-18}{30-15}\right) + \left(\frac{30-18}{45-18}\right)\right)}{2} = 0.6222;$$

Similarity for ***M*** indicator:

$$s_{ijk} = \frac{\left(\left(\frac{30-16}{30-11}\right) + \left(\frac{30-16}{34-16}\right)\right)}{2} = 0.7572;$$

Similarity for ***K*** indicator:

$$s_{ijk} = \frac{\left(\left(\frac{30-20}{30-15}\right) + \left(\frac{30-20}{40-20}\right)\right)}{2} = 0.5833;$$

iii) Finally, the average of the above three partial results will provide the similarity between the grades under comparison:

$$s_{ijk} = \frac{0.6222 + 0.7572 + 0.5833}{3} = \mathbf{0.65}$$

s_{ijk} =The similarity component between the individuals *i* and *j* corresponding to the variable *k* (grade in this case) under analysis.

GI1,GI2 = Grades of individuals 1 and 2 respectively.

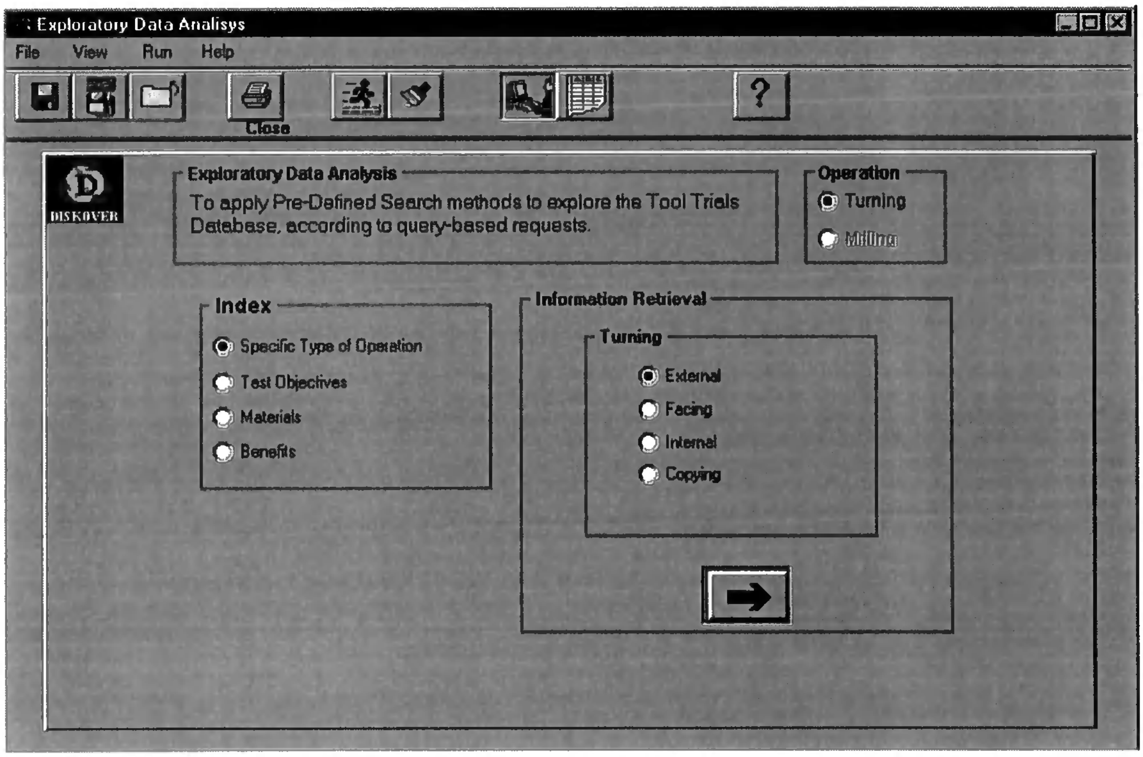
Vmax,Vmin = maximum and minimum values of grade applications for a particular individual respectively, taken from Table 5.3.

Another interesting case shown in Table 7.2, is given between grades TP300 and TX150 (similarity = 0), which can be rapidly corroborated observing how there are not overlapping areas between them, as indicated in Figure 5.4.

7.3.3 Q-FAST, an Exploratory Data Analysis (EDA) Application

Taking into consideration that ***DISKOVER*** is going to operate under an Internet environment, which is a platform characterised by a relatively slow access (especially when database operations are considered), *Q-Fast*, an Exploratory Data Analysis (EDA) application has been developed.

Q-Fast is an SQL-based application allowing a fast and exhaustive exploration of the tooling database for turning and milling operations. Four input options were considered, as shown in Figure 7.7. These options are explained as follows.



Exploratory Data Analysis
To apply Pre-Defined Search methods to explore the Tool Trials Database, according to query-based requests.

Operation
● Turning
○ Milling

Index
● Specific Type of Operation
○ Test Objectives
○ Materials
○ Benefits

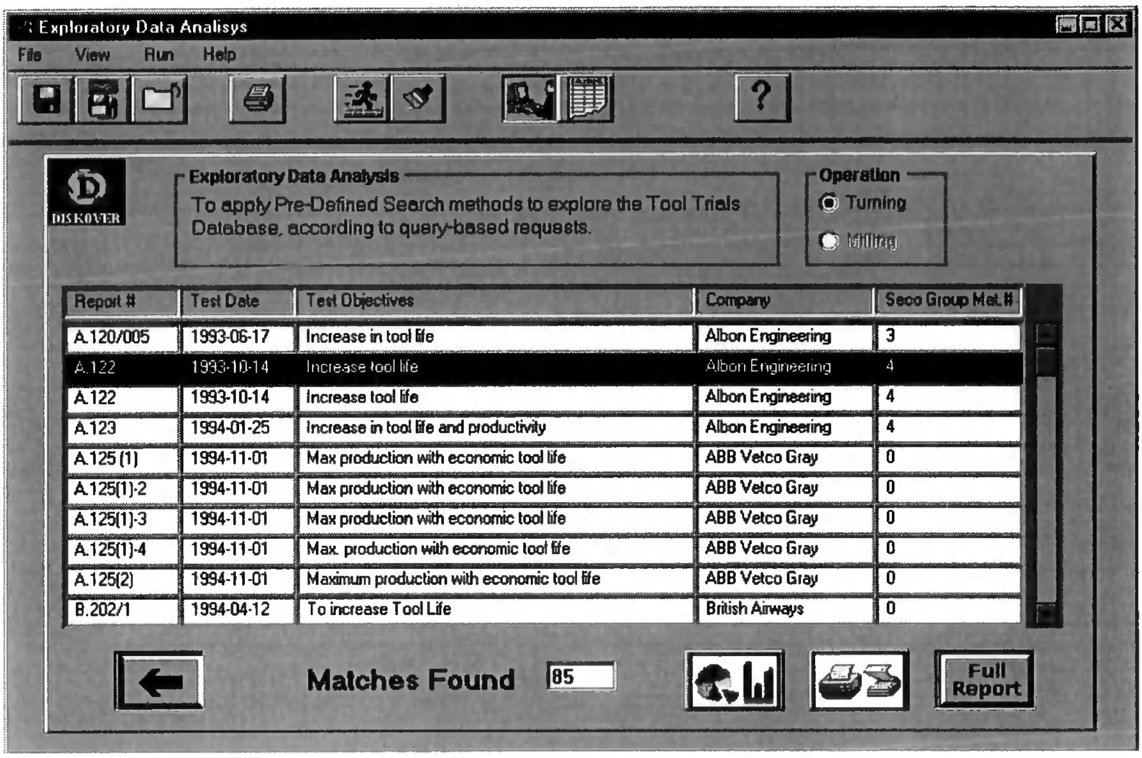
Information Retrieval
Turning
● External
○ Facing
○ Internal
○ Copying

Figure 7.7 – Input screen for Specific Type of Operation (Turning).

7.3.3.1 Specific Type of Operation

This option provides details about five parameters, as displayed in Figure 7.8. In the case of milling operations, this module offers the following options: *Face*

Milling, Slot End, Copy Milling, Square Shoulder, Disc Milling, Chamfering and *Profiling.*



Exploratory Data Analysis
To apply Pre-Defined Search methods to explore the Tool Trials Database, according to query-based requests.

Operation
● Turning
○ Milling

Report #	Test Date	Test Objectives	Company	Seco Group Mat.#
A.120/005	1993-06-17	Increase in tool life	Albon Engineering	3
A.122	1993-10-14	Increase tool life	Albon Engineering	4
A.122	1993-10-14	Increase tool life	Albon Engineering	4
A.123	1994-01-25	Increase in tool life and productivity	Albon Engineering	4
A.125 (1)	1994-11-01	Max production with economic tool life	ABB Vetco Gray	0
A.125(1)-2	1994-11-01	Max production with economic tool life	ABB Vetco Gray	0
A.125(1)-3	1994-11-01	Max production with economic tool life	ABB Vetco Gray	0
A.125(1)-4	1994-11-01	Max. production with economic tool life	ABB Vetco Gray	0
A.125(2)	1994-11-01	Maximum production with economic tool life	ABB Vetco Gray	0
B.202/1	1994-04-12	To increase Tool Life	British Airways	0

Matches Found 85

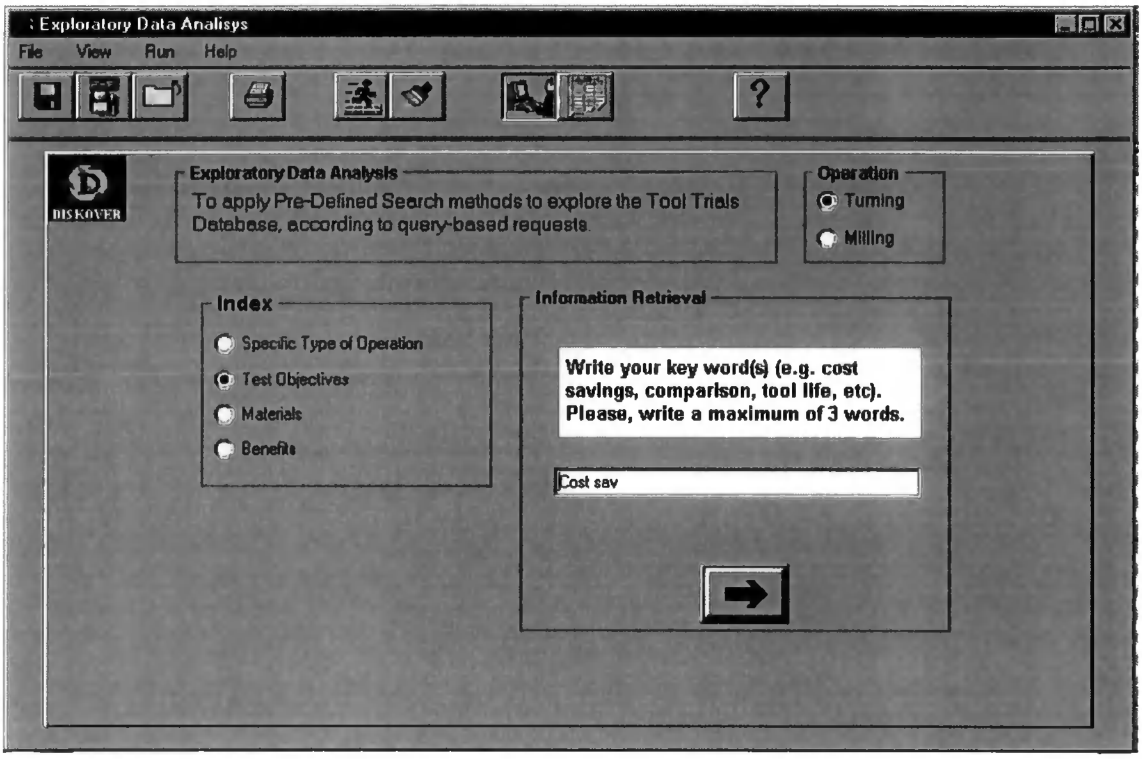
Figure 7.8 – Typical output for Specific Type of Operation (Turning)

Also, as shown in the bottom-right corner of Figure 7.8, *Q-Fast* provides access to the entire parameters of a particular report, through the option "Full Report".

7.3.3.2 Test Objectives

This option implements a search engine by key words, allowing the user to retrieve information about tooling parameters, according to the objectives of the tooling trials. Figure 7.9 shows a typical input screen where the interest of the user is centred on information about *Cost Savings.* Figure 7.10 displays the results of this request.

This application carries out a flexible syntactic analysis, allowing the submission of non pre-labelled data, incomplete phrases, unrestricted blank spaces, combined small and capital letters, etc.



Exploratory Data Analysis

To apply Pre-Defined Search methods to explore the Tool Trials Database, according to query-based requests.

Operation

● Turning

○ Milling

Index

○ Specific Type of Operation

● Test Objectives

○ Materials

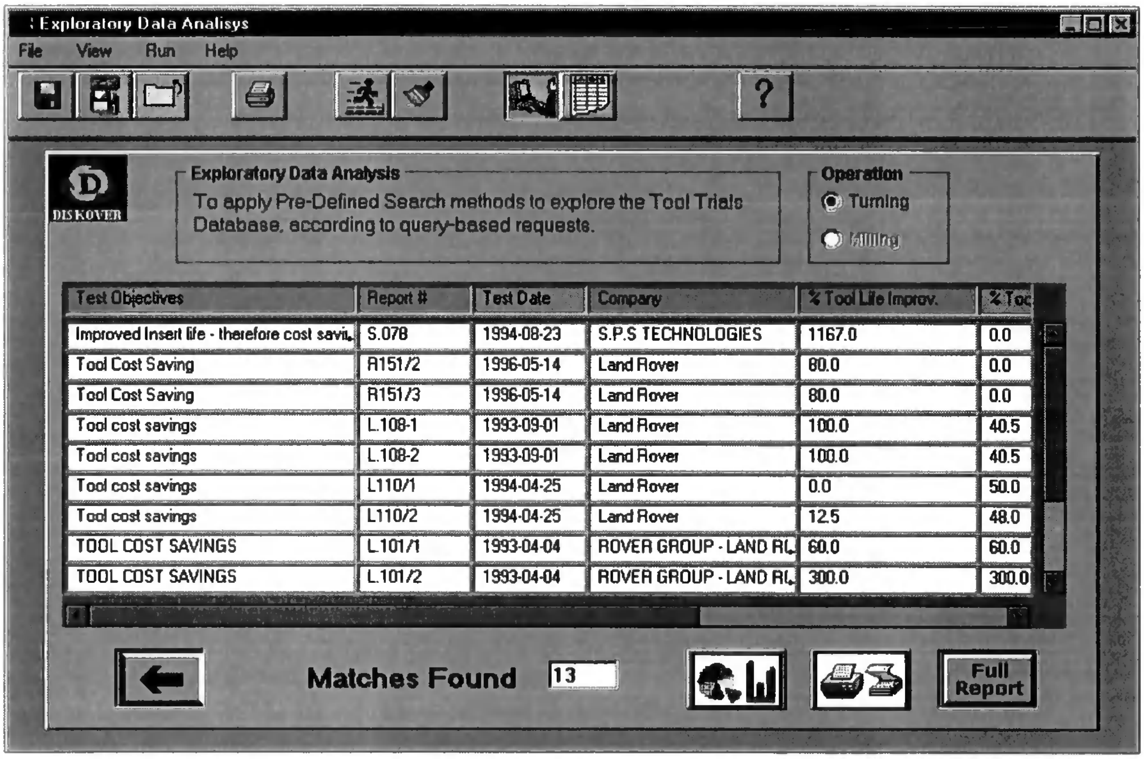
○ Benefits

Information Retrieval

Write your key word(s) (e.g. cost savings, comparison, tool life, etc). Please, write a maximum of 3 words.

Cost sav

Figure 7.9 – Input screen for Test Objectives option (Turning).



Exploratory Data Analysis

To apply Pre-Defined Search methods to explore the Tool Trials Database, according to query-based requests.

Operation

● Turning

○ Milling

Test Objectives	Report #	Test Date	Company	% Tool Life Improv.	% Toc
Improved Insert life - therefore cost savi	S.078	1994-08-23	S.P.S TECHNOLOGIES	1167.0	0.0
Tool Cost Saving	R151/2	1996-05-14	Land Rover	80.0	0.0
Tool Cost Saving	R151/3	1996-05-14	Land Rover	80.0	0.0
Tool cost savings	L.108-1	1993-09-01	Land Rover	100.0	40.5
Tool cost savings	L.108-2	1993-09-01	Land Rover	100.0	40.5
Tool cost savings	L110/1	1994-04-25	Land Rover	0.0	50.0
Tool cost savings	L110/2	1994-04-25	Land Rover	12.5	48.0
TOOL COST SAVINGS	L.101/1	1993-04-04	ROVER GROUP - LAND RL	60.0	60.0
TOOL COST SAVINGS	L.101/2	1993-04-04	ROVER GROUP - LAND RL	300.0	300.0

Matches Found 13

Figure 7.10 - output screen for Test Objectives (Turning).

The output screen shown in Figure 7.10 is able to provide information about eight parameters, namely, *Test Objectives, Report #, Test Date, Company, %*

Tool Life Improvement, % Tool Cost Savings, % Productivity Improvement and *% Cost Reduction.*

7.3.3.3 Materials

Due to the importance of the variable *Materials*, it was decided to include a third option allowing a fast exploration of the tooling database in relation to this parameter. Fifteen material groups were considered and seven output parameters are provided, namely, *Report #, Test Date, Material Name, Seco Group #, Component* and *Part Number.*

7.3.3.4 Benefits

Finally, the option *Benefits* provides information about improvements when executing trials using *Seco* tools versus those coming from different suppliers. The information is based on four aspects, namely, *Tool Life, Tool Cost Savings, Productivity Improvements* and *Cost Reduction per Component.*

7.3.4 *MQG*, a Multiple Query Generator Application

MQG is an SQL-based application developed[9] to provide cross-information retrieval in relation to turning and milling operations. The user is able to fix input conditions chosen among 30 tooling parameters, weigh the importance of these parameters and select output variables of his/her interest among 32 possible tooling attributes.

7.3.5 *R-Set*, a Rough Sets-based Application

R-Set is a rough sets-based application developed[9] to identify data inconsistency and provide ways to analyse the influence of tooling parameters (condition variables), on pre-fixed output attributes (decision variables).

7.4 SUMMARY

In this chapter, the motivation for applying KDD technology to the tooling industry was analysed. *Data increase, isolated analyses* and *lack of strategies*

[9] Developed by another research student in the group at Durham.

to carry out automated search of patterns were identified as factors of main concern.

The development of ***DISKOVER***, a system that integrates multiple data mining techniques, demonstrated the significant potential of combining complementary applications to implement knowledge discovery solutions. *Kluster, Fuzzy-K* and *Q-Fast* constitute applications based in clustering, fuzzy-clustering and SQL methods, which were implemented to analyse tooling data.

Six tooling parameters were considered as relevant during the pattern search processes, namely, *Workpiece Material, Insert Grade, Nose Radius, Cutting Speed, Feed Rate* and *Depth of Cut*.

As ***DISKOVER*** accesses the tooling database through the Internet, which is a relatively slow access platform, exploratory data applications were implemented, to satisfy faster information retrieval operations.

In the next chapter, a discussion of results and structured tests in relation to all the systems developed in this research, are presented.

CHAPTER 8

TESTING THE SYSTEMS AND ANALYSIS OF RESULTS

8.1 INTRODUCTION

The previous chapter examined the functionality of *Kluster, Q-fast and Fuzzy-K,* three data mining applications, which are part of an integrated knowledge discovery system, called ***DISKOVER***. This chapter presents a structured set of tests, applied on ***TTS, SELTOOL*** and ***DISKOVER***, the three Internet-based systems developed during this research by the author and his colleagues.

In the case of ***SELTOOL***, a complete performance evaluation including a case-study and an analysis of the completeness of the database was conducted. Connectivity and remote downloading considerations are the topics to be tested for ***TTS***.

In relation to ***DISKOVER***, *Kluster* and *Fuzzy-K* are the two modules to be considered. The criteria to choose the appropriate kind of tests are based on the establishment of meaningful relationships between relevant-context variables, and on functionality aspects of the modules. These relationships involve the analysis of cutting parameters specified by users, under certain input conditions, such as *Material Workpiece*, *Grade* and *Nose Radius.* Functionality features include the stopping rule of the algorithms and a performance comparison of the clustering methods implemented.

8.1 REASONS TO DESIGN AND APPLY A SET OF TESTS

After finishing the development phase, it is important to apply functional tests to any computational system. These tests will demonstrate that any weaknesses, which occurred in the design phase can be detected, leading to respective modifications, to correct unsuitable situations before delivering the system to its final operational environment.

Some important aspects to be considered in a test phase include:

- The robustness of the user-interface, particularly when the system has to support moderate or intense user interactions.
- The reliability of the answers given by the system. This analysis involves verifying the answers provided by the system against different knowledge sources, such as experts in the respective domain, technical catalogues and successful, similar systems.
- The completeness of the database to satisfy user information requirements.
- The operational spectrum of the system. This involves the introduction of different input combinations to check specified output values.
- The connectivity and remote access efficiency, particularly when networked or distributed solutions are implemented.

The next three sections present all the tests conducted and results obtained.

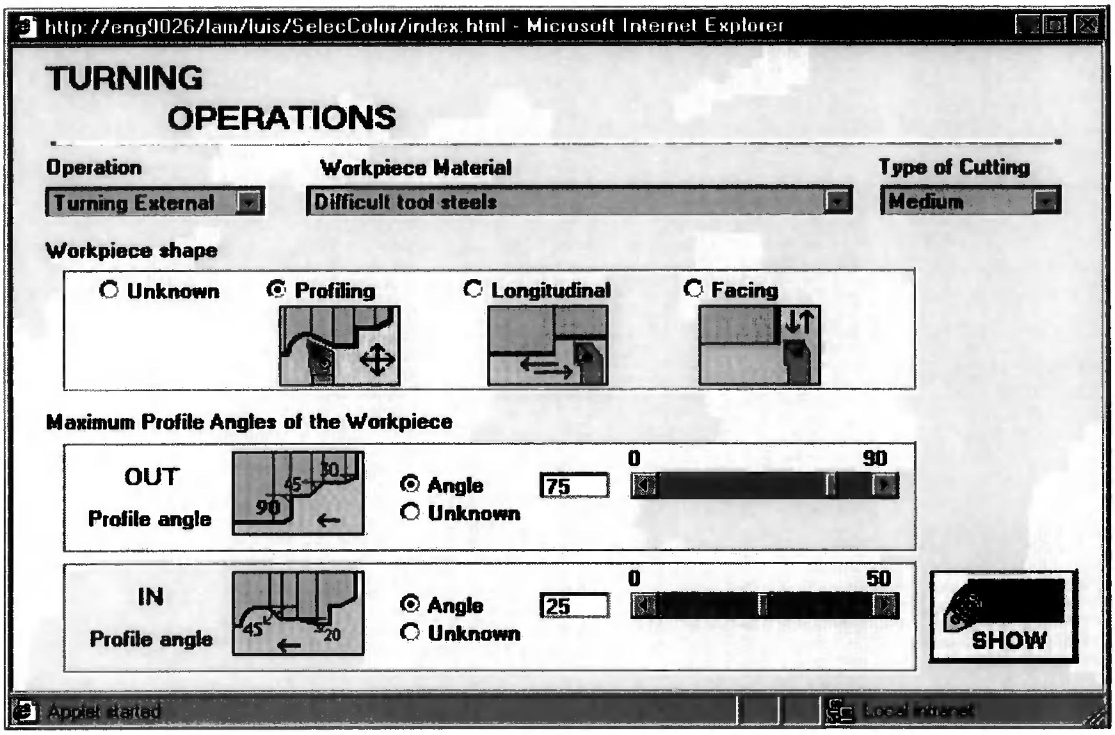
8.2 TESTS AND RESULTS FOR *SELTOOL*

As explained in chapter 3, ***SELTOOL*** is an Internet-based system developed adopting a free-access philosophy, where world-wide Seco customers would remotely access tooling information in the field of tool selection. In order to evaluate its functionality and identify the completeness of the database to satisfy user-requests, a testing phase was conducted.

8.2.1 A Case-Study

The first stage of the test phase was the creation of case-studies to validate the results provided by the system against existing information in catalogues. To demonstrate the functionality of ***SELTOOL***, one of these cases will be discussed.

Let us assume that the material of the workpiece to be machined is a "*Difficult tool steel*", the type of operation is "*External Turning*" and the type of cutting "*medium roughing*". It is also known that the shape of the workpiece is profiling with a "*maximum profiling OUT angle = 75°*" and *a "maximum profile IN angle = 28°*". The initial step is to introduce these input values to the system.



http://eng9026/lam/luis/SelecColor/index.html - Microsoft Internet Explorer

TURNING
OPERATIONS

Operation: Turning External

Workpiece Material: Difficult tool steels

Type of Cutting: Medium

Workpiece shape

○ Unknown ● Profiling ○ Longitudinal ○ Facing

Maximum Profile Angles of the Workpiece

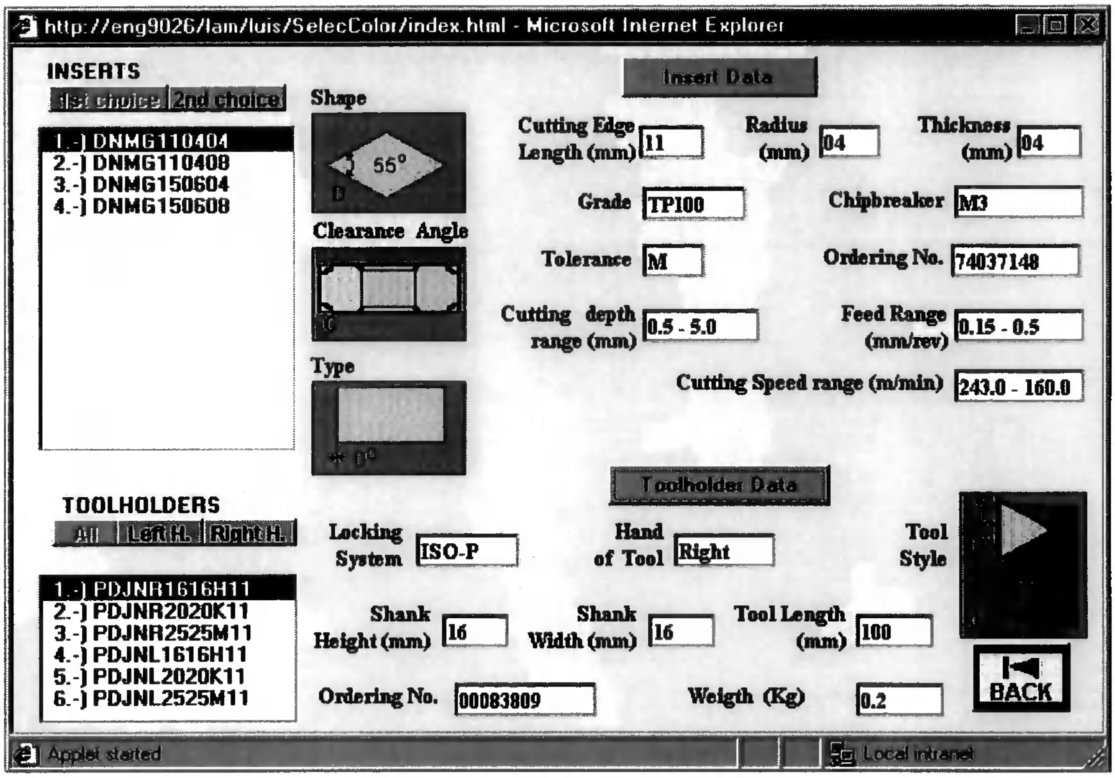
OUT Profile angle: ● Angle ○ Unknown 75 (0–90)

IN Profile angle: ● Angle ○ Unknown 25 (0–50)

Figure 8.1 – Input values for a selected case-study.

Figure 8.1 shows the input values and the corresponding results are displayed in Figure 8.2. The system creates two lists of suitable inserts, the "first choice" and the "second choice" inserts respectively. "First choice" inserts correspond to the usually recommended solution by the tool manufacturer, whilst the "second choice" list includes several other alternative inserts. Four "first choice" Inserts were obtained. The specification of the insert code and suitable cutting data range for these inserts are shown in the top-right part of the output screen.

When any insert is selected from the Inserts list (Figure 8.2, top-left) the suitable toolholders list for the specified insert is shown (Figure 8.2, Bottom-left). For the first insert chosen, there are six possible toolholders with different types of hand (left and right) and sizes in the *Shank Height* and *Width* fields. If only left hand toolholders are needed, the option "**Left H.**" can be activated and just left hand toolholders will be displyed in the list. In the same way, the second choice Inserts list is shown when the "**second choice**" option is activated. The second choice inserts are those that can be considered as alternative options but without providing the best performance as the first choice inserts. The second option is very important when first choice results cannot be obtained. In the case-study presented, five inserts were found as second choice.



http://eng9026/lam/luis/SelecColor/index.html - Microsoft Internet Explorer

INSERTS

1.-) DNMG110404
2.-) DNMG110408
3.-) DNMG150604
4.-) DNMG150608

Cutting Edge Length (mm) 11
Radius (mm) 04
Thickness (mm) 04
Grade TP100
Chipbreaker M3
Tolerance M
Ordering No. 74037148
Cutting depth range (mm) 0.5 - 5.0
Feed Range (mm/rev) 0.15 - 0.5
Cutting Speed range (m/min) 243.0 - 160.0

TOOLHOLDERS

1.-) PDJNR1616H11
2.-) PDJNR2020K11
3.-) PDJNR2525M11
4.-) PDJNL1616H11
5.-) PDJNL2020K11
6.-) PDJNL2525M11

Locking System ISO-P
Hand of Tool Right
Tool Style
Shank Height (mm) 16
Shank Width (mm) 16
Tool Length (mm) 100
Ordering No. 00083809
Weigth (Kg) 0.2

Figure 8.2 – Output Screen for a selected case-study.

The graphical interface allows a better visualisation and an interactive and crossed-way of searching suitable tools, than conventional representation schemes provided by catalogues.

8.2.2 Completeness of the Database

The second stage of the test phase was the execution of experiments with different input parameters to find out the levels of completeness of the database.

Table 8.1 - Inserts Found for External Turning Operations.

Material Groups	External Turning					
	Frist Choice			Second Choice		
	Finis.	Medium	Rough	Finis.	Medium	Rough
Steel < 90 fg/mm^2	35	18	10	44	25	41
Steel > 90 fg/mm^2	26	14	10	30	18	25
Easy-cut. & moder. austenitic steels	31	15	12	66	42	24
Austenitic and duplex stainless steels	26	24	0	31	15	16
Cast iron	26	14	10	43	18	25
Aluminium & other non-ferrous alloys	22	0	0	9	0	0

Table 8.2 - Inserts Found for Internal Turning Operations.

Material Groups	**Internal Turning**					
	Frist Choice			**Second Choice**		
	Finis.	Medium	Rough	Finis.	Medium	Rough
Steel < 90 fg/mm^2	26	11	04	32	17	14
Steel > 90 fg/mm^2	17	08	04	20	11	08
Easy-cut. & moder. austenitic steels	22	09	04	45	25	08
Austenitic and duplex stainless steels	17	14	0	22	09	06
Cast iron	17	08	04	30	11	08
Aluminium & other non-ferrous alloys	13	0	0	9	0	0

Because external and internal turning operations are quite common, the experiment focused on searching for all possible solutions considering these two operations. The workpiece material groups were evaluated using *finishing, medium* and *rough* type of cutting, and both lists of inserts, first and second choices, were taken into consideration in the analysis. Table 8.1 and Table 8.2 show the results for the combination of these input parameters.

Figure 8.3 and Figure 8.4 display a graphical representation of these results. From considerations of these results, it can be seen that the system is able to provide suitable first and second choice inserts for all specified material groups.

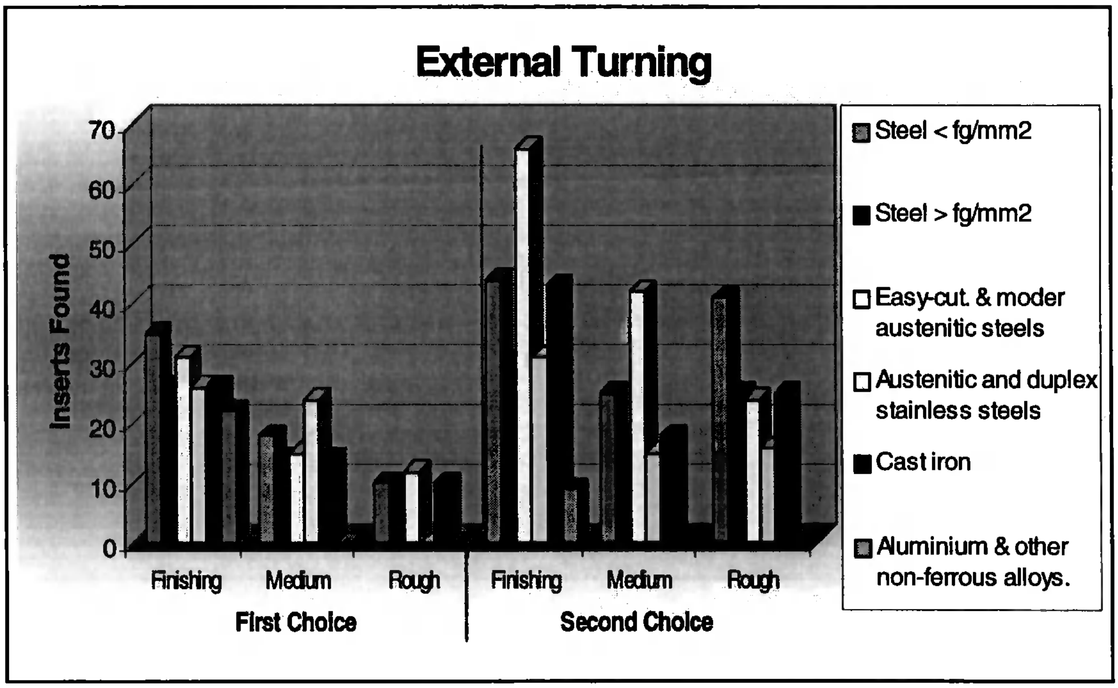


Figure 8.3 - First and Second Choice Inserts for External Turning.

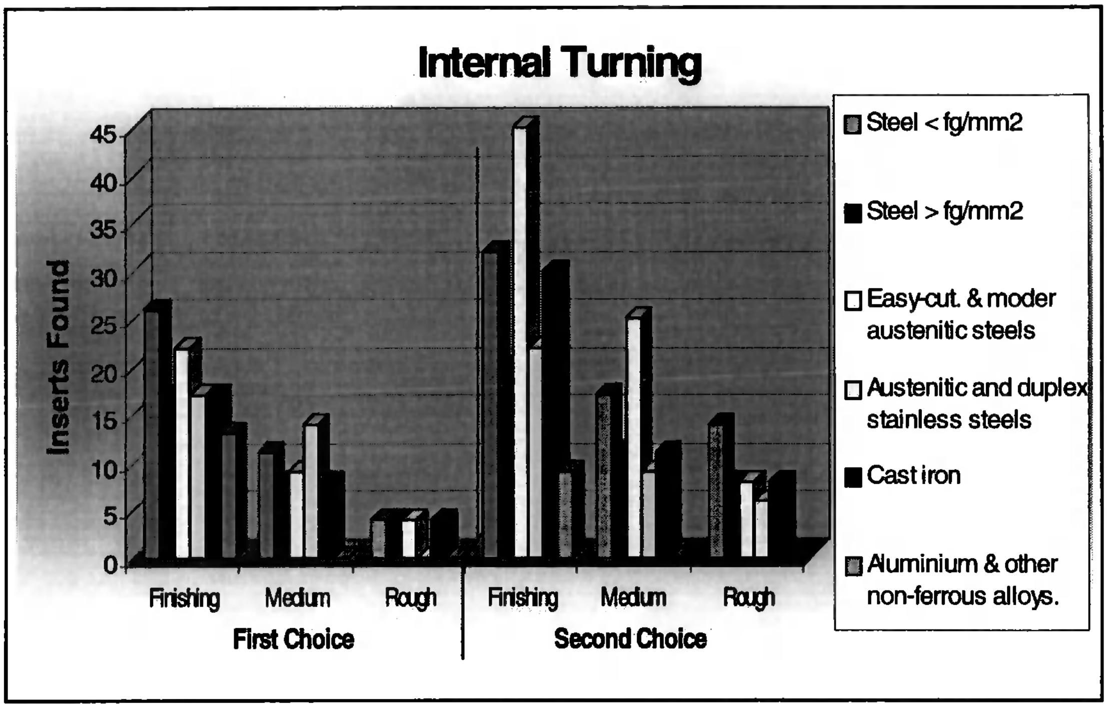


Figure 8.4 - First and Second Choice Inserts for Internal Turning.

However, for some types of material and types of cut there are no suitable inserts. For example, in the case of medium and rough types of cutting, the search for inserts for aluminium and non-ferrous alloys, results in no inserts. Similarly, the system is not able to provide suitable first choice inserts for cast iron and rough types of cutting, but when the second choice option is used, the system suggests a list of alternative inserts.

The values observed in Figure 8.4, for Internal Operations, indicate the same trend observed in Figure 8.3, but the number of inserts found for each type of cutting is smaller than those found for external operations.

To verify the effectiveness of using the profile angles features to obtain more accurate results in the search for tools, a particular test was carried out. This test allows the comparison of unconstrained results, which do not have to satisfy particular profile shapes, with constrained results that meet the profile shape requirements in terms of profile angles.

For ten cases, inserts were selected, initially assuming that the geometry of the profile was unknown, as shown in Table 8.3. Once the number of inserts for each case was obtained, the next step involved changing the value of the profile angles to find new sets of tools.

Table 8.3 - Number of inserts found.

No	Workpiece Shape *unknown*	Prof. OUT 40°	Prof. OUT 90°	Prof. IN 18°	Prof. IN 23°
1	35	31	25	15	13
2	24	24	22	12	9
3	26	22	17	9	7
4	4	4	4	2	0
5	8	8	8	4	2
6	31	27	26	15	13
7	10	10	8	4	2
8	15	15	13	7	4
9	22	22	21	12	11
10	11	11	11	5	3

The data from Table 8.3 is represented graphically in Figure 8.5.

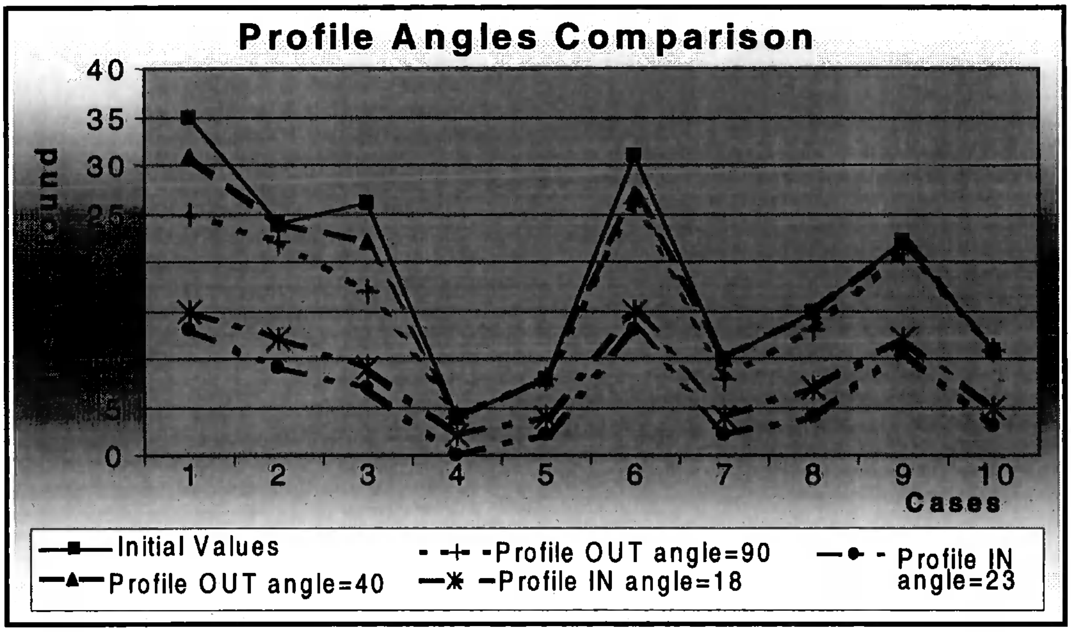


Figure 8.5 - Inserts found using Profile Angles.

It can be seen that the number of inserts found diminishes when the profile angles values are increased. A particular rapid reduction in the number of proper tools can be found by imposing a “profile-IN” constraint, which limits the selection of a large number of tools due to trailing angle unsuitability. For example, cases 1, 2, 3, 6 and 9 show a considerable reduction in the number of inserts found.

It is important to underline that in this last test only one profile angle (OUT or IN) was used at the time. When both angles are used at the same time, the number of matches found is even less, making easier the final selection by the user. Details of "Profile-OUT" and "Profile-IN" angles are given in chapter 3.

8.3 TESTS AND RESULTS FOR *TTS*

TTS is an Internet-based system developed under a private philosophy, where tooling engineers working in a shared information environment, would be able to access a nation-wide database of knowledge, created from their previous work. Moreover, it is possible for tooling engineers to avoid the execution of new tool trials knowing the results of trials already carried out in physically distant places, when another engineer has previously executed these trials.

The participation of the author in the development of this system was centred on connectivity, information security and some operational aspects. This section presents a set of tests in relation to connectivity and how sensible the system is to change the number of matching results found, when particular input parameters are introduced.

8.3.1 Connectivity and Downloading Aspects

Although Java-code embedded in HTML files can be easily downloaded by Java-enabled browsers, remote access of databases is not a trivial task. One reason is the inability of some Web browsers to deal with recent versions of Java language and the drivers that manage the connection to the databases. Furthermore, ***TTS*** is an Internet-based system containing Java classes, images, compressed files, HTML and other support files. Therefore, a set of tests was conducted to check the transparency in the connectivity, and the efficiency regarding the downloading of database and system files.

In relation to connectivity, access to the database was tested using the *JConnect* (100% pure Java) driver for Java 1.02. Two widely commercial Web browsers were used, Internet Explorer 5.0 and Netscape Communicator 4.5. In both cases the access was successful. However, for a new version of ***TTS,*** compiled using Java 1.1, both browsers were unable to support remote database operations. Table 8.4 shows results of downloading the system from different locations and using compression and local file-storage procedures.

Table 8.4 – Connectivity and Downloading Tests (TTS).

User Location with Internet connection	**DB and System Location**	**Normal Downloading** **(1)**	**Downloading Using JAR Files** **(2)**	**Downloading Storing Java Classes in Client Computer**
Local Stand-Alone User Durham City	DB and Web-Server Durham University	*OK*	*Successful and A bit Faster than (1)*	*Successful and Faster than (2)*
Local Area Network Durham University	DB and Web-Server Durham University	*OK*		
Stand-Alone Seco - Birmingham	DB and Web-Server Durham University	*OK*		
	DB and Web-Server Seco - Birmingham	*OK*		
University LAN Manchester (UMIST)	DB and Web-Server Durham University	*OK*		

From Table 8.4 it can be noted that regardless of the user location, the operations of downloading the database and ***TTS*** files were successful. The same test was conducted using JAR files and storing the Java classes locally in the user's machine.

JAR stands for *Java ARchive*. A JAR file is a single file containing a collection of Java class files. In this way, JAR files are similar to ZIP files and other file formats used to package a collection of files in a single file. JAR files compress the data they contain, so they are faster to download to a user system. One advantage of storing class files in a JAR file is that the user's system can download all the necessary files with a single file transfer request, rather than individual file transfer requests for each file that the Java-applet needs. One big file transfer or downloading, typically takes less time in total (in theory) than many little ones.

The access of the system was tested using JAR files. This reduced the downloading times, but not substantially. In order to access ***TTS*** through the Internet and minimise the downloading time, an additional test was conducted. The API Java classes utilised by the system were locally stored on the client machine, while the database was stored remotely in a Web-server. This produced access and downloading operations faster than the implementation of JAR files. One clear disadvantage of this procedure is the necessary installation of the files and the setting tasks in the local operating system of each client machine, particularly due to the distributed nature of the proposed solution. However, these disadvantages can be minimised if the distributed users share an efficient networked environment.

8.3.2 Confidence Scores Values

The implementation of Confidence Scores (CS) by ***TTS*** allows a more flexible criterion to relax the number of matching results that the system is able to find.

It was decided that in addition to searching for perfectly matching reports, it would be made possible for users to assign CS values to relax the number of final matches found.

Once the user has generated the search criteria and instigated the search, the first step is to retrieve all the report numbers of those trials, which match at least one of the search criterion. The number of reports returned can then be used to display the size of the result set to the user. If the number of reports generated by the search is small, it is possible for the user to modify the query manually and perform a further search. Alternatively, the user can decide either to view the few reports which have been identified, or opt for parameter relaxation techniques to be applied, the details of which without the scope of this chapter.

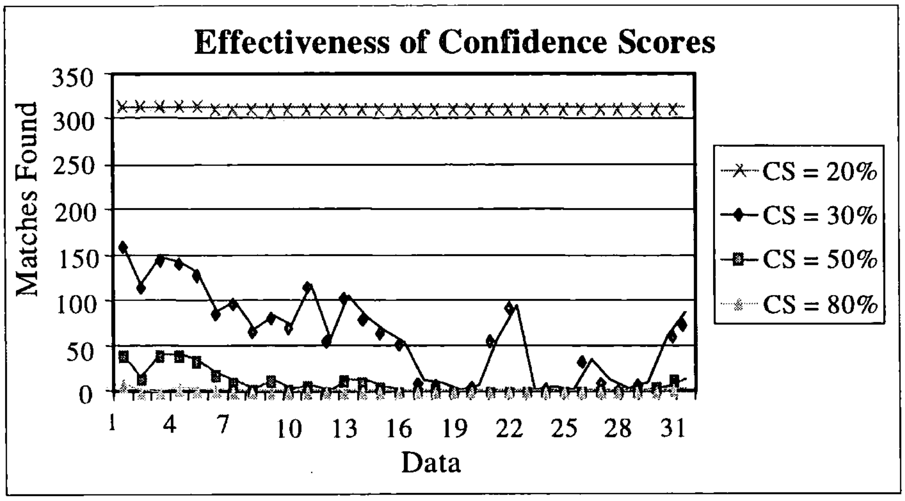


Figure 8.6 - Effectiveness of Confidence Scores (CS)

The confidence score for any single tool trial can assume any value between 0% and 100%. A confidence score of 100% indicates a tool trial report where all the parameters specified by the user have been matched exactly. Conversely, a confidence score of 0% indicates that none of the parameters match either exactly or partially.

Figure 8.6 shows how flexible is the system to find close matches when low values of CS are introduced, and the opposite situation where the system applies a strict filtering criterion, limiting the output to few matches. The data used for this test can be seen in Appendix E.

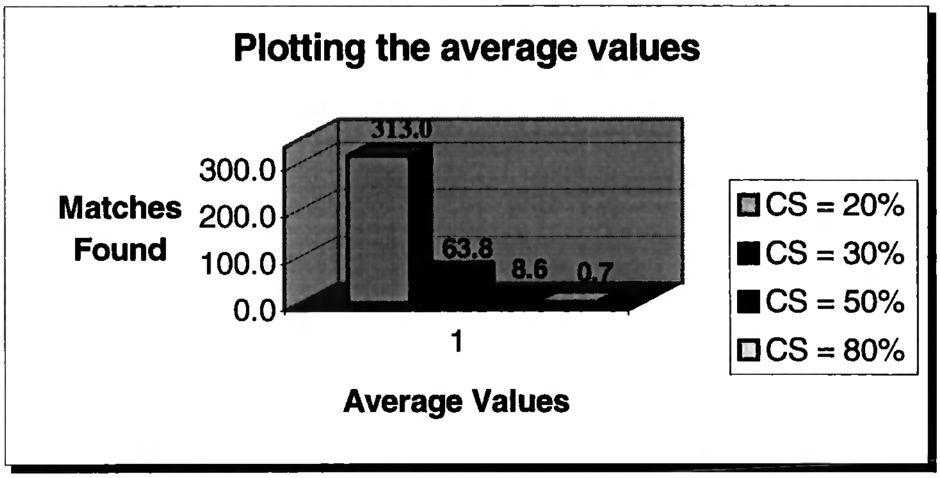


Figure 8.7 – Average values for Confidence Scores.

Figure 8.7 shows the average values, indicating an output of 313 reports (all reports initially contained in the database for Turning) with a CS of 20% and values of almost zero reports, to a CS of 80%. This reflects the capabilities of the system to "fluctuate" in a wide functional band, fitting its number of successful matches, according to the confidence levels specified by the users.

Finally, it is important to establish a comparison between the nature of the data used in the tests and the nature of the data contained in the database. The test data is taken from catalogues in their 1998 edition, and the data used to populate the database is taken from industrial trials performed between 1992 and 1998. Therefore, it is possible that the obtained results can be slightly slanted, and so, useful data for establishing full matches could not be considered.

8.4 TESTS AND RESULTS FOR *DISKOVER*

DISKOVER is a context-oriented KDD-based system that has been developed considering the implementation of multiple data mining techniques. It has been oriented to analyse tooling data collected from multiple machining centres. In contrast to the two Internet-based systems previously mentioned, which were built using *Java-applets*, ***DISKOVER*** is also an Internet-based system, but built using a *Java-application*. This eliminates the use of Web browsers to access the system. This section will concentrate in a set of tests applied on *Kluster* and *Fuzzy-K*, two important modules of ***DISKOVER***.

Kluster and *Fuzzy-K* are able to provide numerical quality indexes to compare the performance of the four clustering methods implemented. The final quality index is based on intracluster and intercluster measures as was described in chapter 6. Both systems calculate these quality parameters for each clustering method and show only the results matching the method having the best performance. Table 8.5 shows these performance indexes for *Kluster*.

As can be seen in Table 8.5 all the quality values of the final classification are higher than 0.5 and the *Average Linkage* resulted in the best clustering method. In general, the *average* method produced the best classification according to the quality criterion adopted.

Table 8.5 – Performance of Clustering Methods (Kluster).

Clustering Method	**Intracluster (*ICI*)**	**Intercluster (*ISL*)**	**Quality (Q_c)**
Simple	0.0487	39.1820	0.5777
Complete	0.1464	49.6089	0.5608
Average	0.0178	32.2350	**0.8249**
Ward's	0.1392	47.7573	0.5452

These results were corroborated considering different input parameters but keeping the same run options. However, a change in the stopping rule options

(Mojena's method versus # of clusters = 30) produced different results, as shown in Table 8.6.

Now, the best performance was obtained applying the *Simple* method. The changes in the new quality indexes look logical considering the influence of the Mojena's constant (*k*) on the number of clusters obtained in the final classification process. As was stated in chapter 6, the best performance was obtained for 4.0 < *k* < 4.5. Values of *k* in this range produced a number of clusters in the range 8 to 15, which influenced the calculation of new quality indexes, hence, the resulting new values, as shown in Table 8.6.

Table 8.6 – Quality indexes changing the stopping rule option

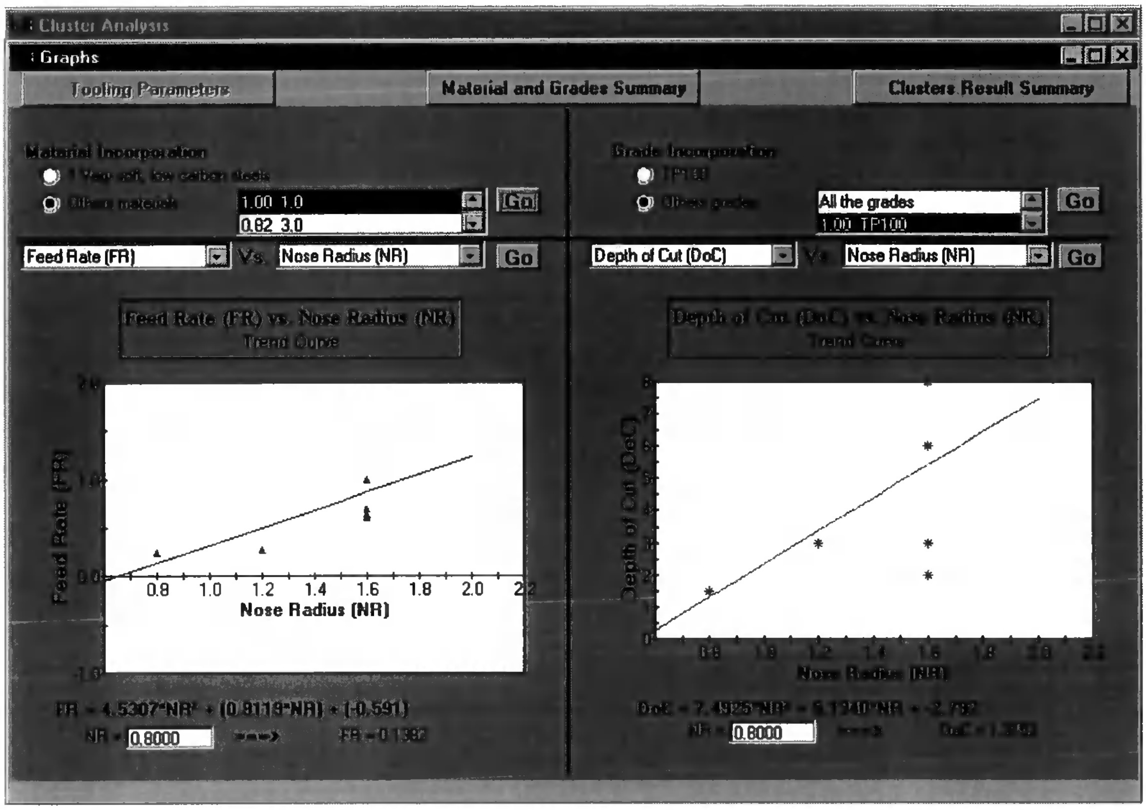
Clustering Method	**Intracluster (*ICI*)**	**Intercluster (*ISL*)**	**Quality (*Qc*)**
Simple	0.0230	36.2074	0.9383
Complete	0.1276	41.3104	0.5902
Average	0.0287	35.8100	0.8342
Ward's	0.1392	47.7573	0.5642

The quality indexes shown in Table 8.5 and Table 8.6 were obtained setting *Material Group* =1 and *Grade* = TP100 as the basic input parameters. *Kluster* obtains the closest set of elements (clusters) considering these input values, providing an optimal solution. This initial solution can be used to obtain information about cutting parameters, according to particular input specifications.

For example, Figure 8.8 displays the results obtained through the implementation of a second order regression analysis. In this case the user selected *Material Group* = 2 and *Grade* = TP10 as complementary input information.

It can be noted from the graphs (left side), how it was possible to determine the value of the *Feed Rate* (0.1382), regarding *Nose Radius* specifications (0.8 mm), considering the information contained in the clusters previously grouped.

The same case is shown (right side) for the cutting parameter *Depth of Cut* (1.3152). The user has the option to obtain a third cutting parameter known as *Cutting Speed*. The *Nose Radius* is an important parameter introduced by users depending the type of cutting (finishing, medium or rough) of the workpiece to be machined.


Cluster Analysis

Graphs

Tooling Parameters | Material and Grades Summary | Clusters Result Summary

Material Incorporation
○ 1 Very soft, low carbon steels
● Other materials
1.00 1.0
0.82 3.0
Go

Grade Incorporation
○ TP100
● Other grades
All the grades
1.00 TP100
Go

Feed Rate (FR) Vs. Nose Radius (NR) Go

Depth of Cut (DoC) Vs. Nose Radius (NR) Go

Feed Rate (FR) vs. Nose Radius (NR)
Trend Curve

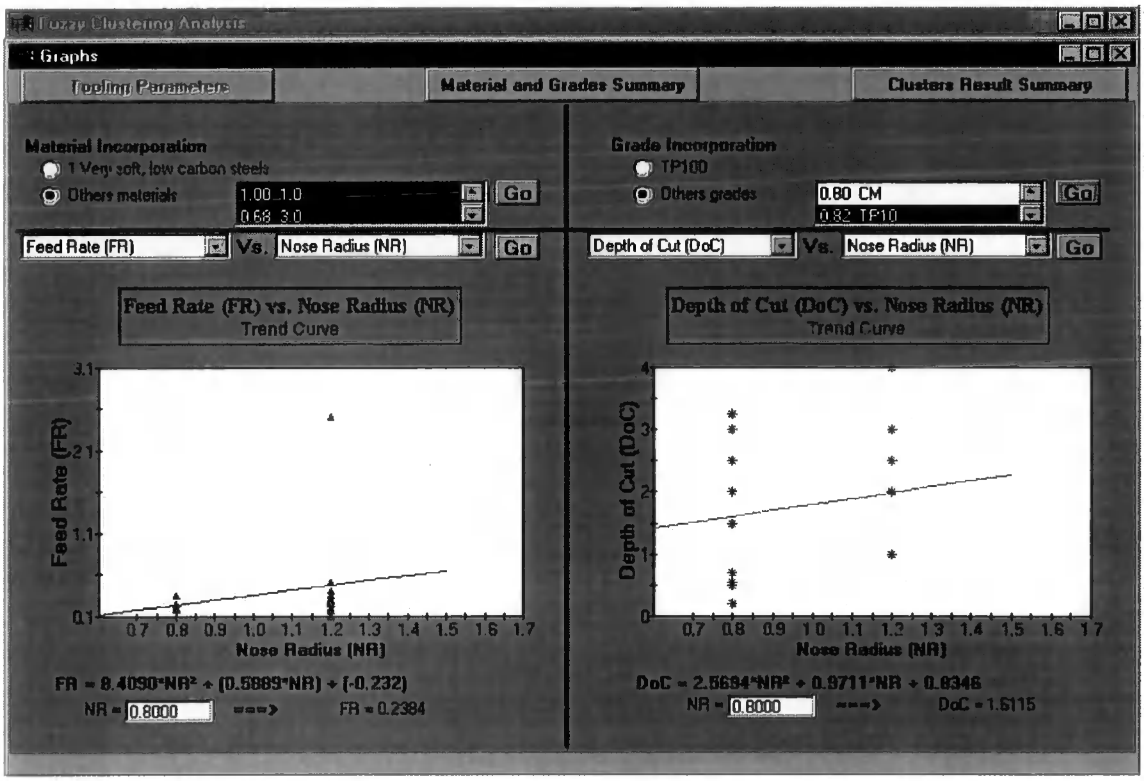
Depth of Cut (DoC) vs. Nose Radius (NR)
Trend Curve

FR = 4.5307*NR² + (0.8118*NR) + (-0.591)
NR = 0.8000 ===> FR = 0.1382

DoC = 7.4925*NR² - 5.1340*NR + -2.792
NR = 0.8000 ===> DoC = 1.3152

Figure 8.8 – Obtaining cutting parameters from specific input requirements.

The option of selecting cutting parameters can also be activated applying fuzzy-clustering analysis (*Fuzzy-K*), as can be seen in Figure 8.9. The distinguishing difference observed in the graphs shown in Figure 8.9 (applying *Fuzzy-K*) and that shown in Figure 8.8 (applying *Kluster*), is that *Fuzzy-K* performs a more consistent classification process. It groups the clusters considering only *Nose Radius* values of 0.8 and 1.2, rather than *Kluster* that also includes clusters having *Nose Radius* = 1.6. To select the new *Material Group* (*MG*) and *Grade* input values the system provides similarity values, showing the closer parameters in relation to the input initially chosen by the user. For example, the closest *MGs* in relation to *MG* = 1 (initially chosen), were *MG* = 2 and *MG* = 3, with a similarity of 0.89 and 0.82 respectively. Likewise, the closest Grade in relation to TP100 (initially chosen) was TP10, with a similarity of 0.875.



Material Incorporation
○ 1 Very soft, low carbon steels
● Others materials: 1.00 1.0 / 0.68 3.0

Grade Incorporation
○ TP100
● Others grades: 0.80 CM / 0.82 TP10

Feed Rate (FR) Vs. Nose Radius (NR)

Depth of Cut (DoC) Vs. Nose Radius (NR)

FR = 8.4090*NR² + (0.5889*NR) + (-0.232)
NR = 0.8000 ===> FR = 0.2384

DoC = 2.5694*NR² + 0.9711*NR + 0.8346
NR = 0.8000 ===> DoC = 1.6115

Figure 8.9 - Cutting parameters from specific input requirements running Fuzzy-K.

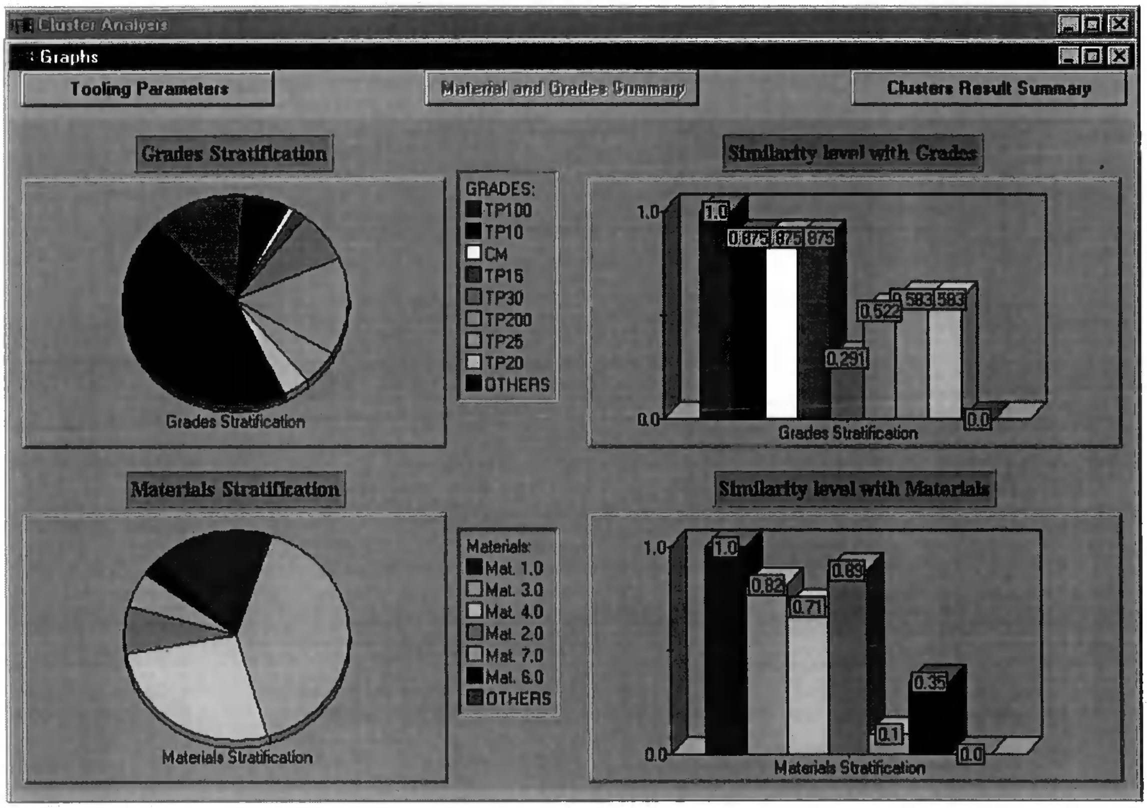


Figure 8.10 – Stratification of Grades and Materials regarding the best cluster found (running Kluster).

These similarity values can also be displayed selecting the option "Material and Grades Summary", as shown in Figure 8.10. Here, the stratification corresponding to the closest set of elements in relation to *Grades and Materials* parameters is illustrated. *Grade* = TP100 and *Material Group* = 1, have a similarity equal to unity, because they were the input values introduced by the user.

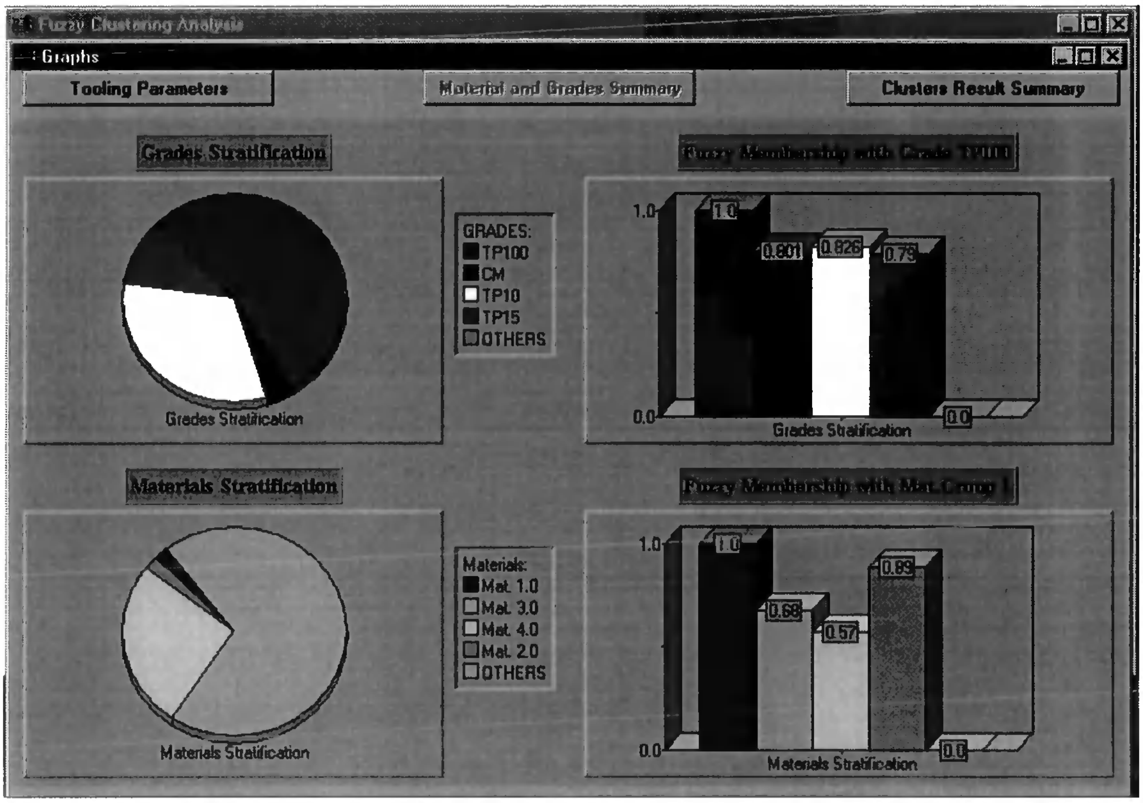


Figure 8.11 - Stratification of Grades and Materials regarding the best cluster found (running Fuzzy-K).

The results in Figure 8.10 indicate that parameters having higher similarity values (*Material Groups* 1, 2, 3 and 4) exhibit the higher appearance percentages in the stratification results (18, 7, 40 and 26%) respectively. This trend was also observed for *Grades.* Therefore, the clustering process was able to group those elements presenting the closer similarity levels, according to a total of seven tooling variables considered. A similar classification trend was obtained applying fuzzy-clustering methods, as shown in Figure 8.11. However, the results observed in Figure 8.11 (running *Fuzzy-K*) show again a more consistent classification than those shown in Figure 8.10 (running *Kluster*). In the case of implementing fuzzy-clustering methods, a smaller number of *Grades* and *Materials* were generated in the final classification.

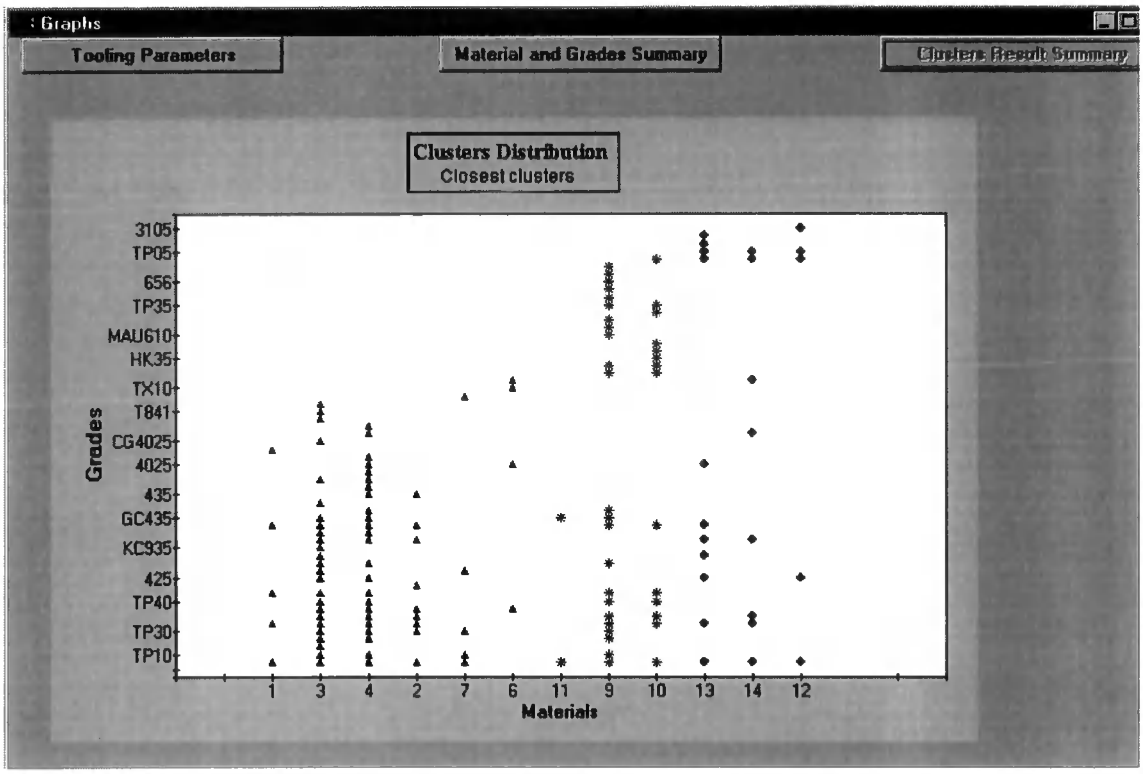


Figure 8.12 – Distribution of the three closest clusters according to Grades and Materials specifications (running Kluster).

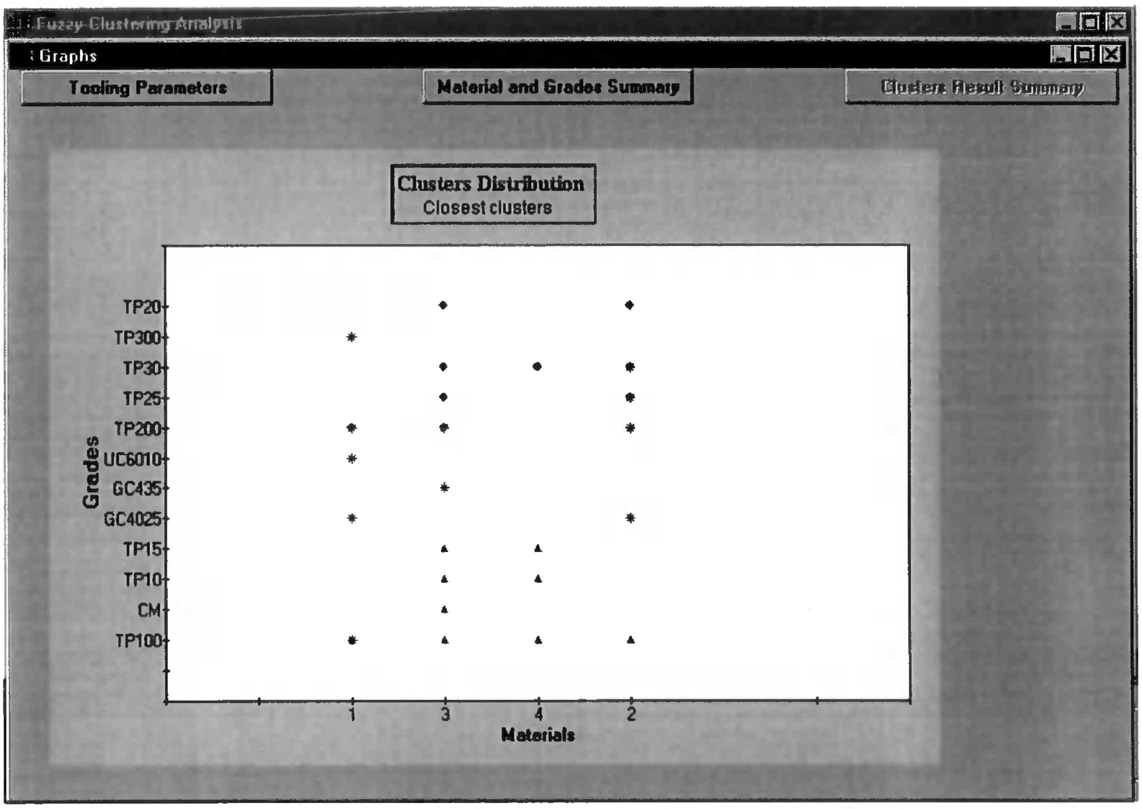


Figure 8.13 - Distribution of the three closest clusters according to Grades and Materials specifications (running Fuzzy-K).

Figure 8.12 displays a graphical representation of the three best clusters obtained, in relation to *Grades* and *Materials* parameters.

The influence (weight) of the variable *Material Group* in the classification process can be noted from Figure 8.12. The clustering algorithm takes into consideration the mechanical properties of the materials to obtain similarity values, as was analysed in chapter 5.

The incorporation of fuzzy membership functions (running *Fuzzy-K* instead *Kluster*) to support the generation of similarity measures, contributed to reduce still more the solution space, as shown in Figure 8.13. It can be noted how the materials and grades included in this classification are now closer to those input parameters specified as relevant by the user. These parameters were *Grade* = TP100 and *Material Group* = 1 (Steels). Stainless Steels and Cast Iron groups were not considered, providing a closer classification according to the user's input.

8.5 SUMMARY

In this chapter, a structured set of tests was applied to the three main systems developed in this research by the author and his colleagues. ***SELTOOL***, a free-access tool selection system for turning operations, provides an interactive and crossed-way of searching for tooling parameters, rather than conventional representation schemes provided by catalogues.

To test the functionality of ***SELTOOL***, a case-study was conducted. ***SELTOOL*** was able to recommend a suitable selection of inserts and toolholders for a specific operation, workpiece material group and cutting type, and show their code explanation, together with the respective cutting data. The system was also able to recommend a second choice, providing in this way, alternate solutions. ***SELTOOL*** found only a few matches for some types of materials such as austenitic & duplex stainless steels and aluminium & other non-ferrous alloys. This fact is only reflecting the need to increase, even more, the current size of the tooling database. The inclusion of new records will improve the capabilities of the system to satisfy a wider range of input requests. If the workpiece shape is known, the specification of profile angles provides a substantial reduction in the number of suggested inserts, which would help users to make a better final decision.

In the case of ***TTS*** it was for example, possible to look at the ways in which tooling data collected from distributed machining centres could be analysed for cutting data selection purposes. ***TTS*** has a database with a number of trial reports far less than would be expected for a system operating fully in an industrial environment. The inclusion of new reports will increase the potential of the system to satisfy full user requirements. ***TTS*** only provides information in relation to cutting parameters when the complete report is downloaded. Sometimes, however, reports satisfying particular input requirements cannot be found. For this reason, it has been given to the user the possibility running the classification modules of ***DISKOVER***, to introduce case-studies generating cutting parameters for particular input requirements.

The testing of ***DISKOVER*** evidenced the usefulness of the data mining methods employed to deal with situations when no perfectly matching results were found. The implementation of these methods seems appropriate taking into consideration that the reports contained in the database are insufficient and, in some cases, they present imprecise information. In relation to connectivity, the tests resulted successfully for a version of the system developed using Java 1.02. However, for a new version of ***TTS,*** compiled using Java 1.1, the browsers were unable to support remote database access and updating operations. Tests were conducted implementing file compression facilities (JAR files) and the storage of the Java classes locally in the user's machine. The implementation of JAR files produced better, but not significantly faster downloading times. In the case of storing the Java classes in client machines, the access performance and the downloading operations improved substantially. Finally, functional tests were conducted, reflecting the capabilities of ***TTS*** to "fluctuate" in a wide functional band, fitting its number of successful matches, according to confidence levels specified by the users.

All graphs generated during the development of the entire set of tests were programmed and plotted using a Java programming environment, without using external calls to applications providing graphical support capabilities (Excel, for example). In the next chapter, overall conclusions are drawn and opportunities for undertaking further research are identified.

CHAPTER 9

CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK

9.1 RESEARCH NOVELTY

The implementation of Internet-based knowledge discovery (KDD) approaches on the tooling industry and the proposal of a Web-based and multi-functional distributed architecture for information integration in corporate environments, constitute the main novel ideas produced by this investigation.

Much has been written about the increasing impact of Web-based distributed solutions and global enterprise, focusing mainly on *models* and *organisational* problems. Some works have surpassed expectations, but too narrowly. Although they have provided significant theoretical contributions, they still lack the required level of integration, multi-functionality and industrial applicability to be considered significant applications. Therefore, there is an imperative need to propose distributed and multi-strategic solutions to corporate environments, which can address the technical issues needed to overcome the problems imposed by real applications, running on industrial environments.

Furthermore, in this research the motivation for applying KDD technology to the tooling industry was analysed. It was found that in spite of the tooling industry around the world constantly generating a considerable amount of data, as a consequence of their daily machining operations, the implementation of formal KDD-oriented systems has not been encountered in this area during the development of this research. The last decade has seen a considerable application of KDD technology in social, economical and scientific fields. However, the tooling industry remains an unexplored sector of potential.

As a result of the consolidation of these ideas, other significant contributions were achieved, which are broadly examined in the next section.

9.2 CONCLUSIONS

In this research, significant attention has been paid to the definition of Web-based distributed strategies for supporting information integration and collaborative work schemes, in the context of corporate environments. The effect the Web has on businesses employing these distributed strategies, has been discussed. It was shown how *Agility, Global Manufacturing, Cyber Factories* and *Virtual Manufacturing* are important concepts impacting on operational practices in the current manufacturing scenarios. Technical issues regarding Internet-based connectivity and database access were addressed. Within these considerations, an *Architecture for Remote Information Exchange* (*ARIEX*) was proposed. *ARIEX* relies on integration of corporate information, distributed on databases having the same internal structure but different data, along geographically dispersed branches. The convenience of sharing information of mutual interest to internal users (employees and partners) as well as external agents (suppliers and customers) working in a platform-independent environment and considering data security aspects, constitutes its main advantage. In the context of corporations having branches in widely dispersed locations, *ARIEX* focuses on three main applications. These applications include information sharing on free-access platforms, collaborative work strategies and knowledge capturing systems. *Information security, database technology* and *ability to manage knowledge* were identified as topics requiring special attention.

TTS[10] and ***SELTOOL***[11], two different Internet-based systems to provide distributed solutions in the tooling area for the company Seco Tools Ltd (UK), were developed. ***TTS*** demonstrated how it is possible to use a shared-information platform to access a nation-wide source of tooling knowledge, whilst keeping a restricted access policy. ***TTS*** has provided a WWW platform from which tooling engineers (authorised users) can submit and retrieve highly specific technical tooling data, for both milling and turning operations. Moreover, it was possible for tooling engineers to avoid the execution of new tool trials,

[10] Developed by the author in collaboration with two other researchers in the group at Durham.

[11] Developed by the author.

because of access to the results of trials carried out in physically distant places, where another engineer had previously executed similar trials.

Because of the dimensions and the corporate nature of ***TTS***, three researchers were assigned to develop this project. The main contribution of the author was to provide an Internet-based framework to support the distributed nature of the proposed solutions. Two different tasks were carried out. Firstly, the selection of an appropriate strategy for sharing information in a distributed environment using the Internet and secondly, the definition and implementation of suitable methods to allow the access to authorised users only (restricted access policy).

These two tasks were analysed taking into consideration the access of geographically dispersed databases and the interest of Seco for relying on a tooling data repository, initially accessible by nation-wide (and subsequently world-wide) authorised users. Data Replication (DR) and real-time access approaches were examined. In this research, the use of DR technology was considered, mainly because remote users access their databases locally and only when they need to send new information to the central database (and receive the latest updates), does a Web connection need to be established. However, due to the high level of confidentiality of the information generated by the tool trials, it was decided to keep a central database accessed remotely through *Java-applets*, adopting a real-time access approach instead of keeping a copy of the database stored in each client-computer, as required by a DR approach. To deal with the security aspects in relation to the access of ***TTS***, several procedures were implemented, which included the creation of encrypted passwords and monitoring functions to register the database transactions.

On the other hand, ***SELTOOL*** was primarily focused to provide distributed solutions in the area of tool selection, but considering the implementation of a free-access architecture. The development of this system involved the analysis of technical criteria to establish an appropriate selection of inserts, toolholders and cutting data for turning, threading and grooving operations. The information used to create the database was obtained from technical catalogues and proprietary tooling databases. Initially, the database was populated manually, which represented very time consuming work, without taking into consideration the difficulties of subsequent modifications and updating. In order to minimise human errors whilst the data is inserted, optimise the update times, provide a better maintenance of the database and take advantage of existing data in MS-Access format, two programs were developed. The Database Populate (*DPS*) and the Database Migratory (*DMS*) Systems. *DPS* reads a text file previously created from tooling data stored in a CD (PDF format), and after performing information filtering, the data is automatically written into the database (SQL format). In the search for useful data for ***SELTOOL***, another barrier to using the available information was found. The information required was stored in DB2 (IBM) and Access (Microsoft) DBMS formats, as the rest of SECO databases, but the available DBMS for building the system was SQL Anywhere (Sybase). To solve this situation *DMS* was developed. *DMS* converts data from an existing tooling database (Access format) to the SQL database (SQL-Anywhere format) used by ***SELTOOL.***

TTS and ***SELTOOL*** were developed using the Java programming language, which provides cross-platform portability. *PowerJ* 2.0, which is a programming tool with graphical facilities and able to speed up the creation of Java projects was used. At the time this research was conducted, commercially available browsers were able to download Java-applets managing databases, using only the Java 1.02 version. It is important to upgrade the compatibility of browsers with later editions of Java in order to access database applications efficiently. To establish connection with the Internet, ***TTS*** and ***SELTOOL*** use *jConnect*, a 100% pure Java driver, which eliminates the problem of asking the users to download and configure the driver. The DBMS used was Sybase SQL Anywhere, which supports the database operations through an Open Server Gateway included as part of the basic *PowerJ* tool package. For developing and testing the systems, an internal net provided the facilities to transfer files and programs efficiently between development environments. A computer configured as a Web-server (Windows NT) and DB-server was used to store the HTML files, images, database and all the programs and Java classes needed to download and run the system from remote locations.

Nowadays, information sources are increasing in size, complexity and number, and current information retrieval techniques are insufficient for very large networked information sources. Hence, organisations are increasingly realising the importance of using knowledge and information residing in their networks for competitive advantage. This scenario has motivated the interest of many researchers to develop automated systems able to apply efficient data-organisation techniques and scientific methods to reveal deviations, dependencies, regularities and interesting patterns in these data sets. In this research, the motivation, benefits and technical considerations for undertaking KDD tasks were discussed. Further, a well-structured KDD architecture was examined.

It was decided following discussions with Seco-UK, to develop a knowledge discovery application for the tooling sector. Therefore, ***DISKOVER***, an KDD-oriented system for analysing tooling data was developed by the author and another research student in the group at Durham. This system integrates five modules including clustering, fuzzy-clustering, rough sets programs and SQL-based exploratory data analysis methods, under a unified Internet-based architecture and graphical interface. A functional summary of these modules follows.

Kluster[12] is a clustering application based on the 11-steps Cluster Analysis methodology proposed in chapter 5. In order to minimise the lack of conceptual considerations when applying conventional clustering methods, conversion mechanisms to establish similarity relationships between some categorical attributes, were implemented. Also, two original contributions have been incorporated in the *Post-processing* stage. Specifically, consistent indicators to measure the *quality of the final classification* and application of *optimisation methods* to the final data groups obtained. The analysis of *cutting condition* parameters and relationships between *grades* and *material workpiece* was considered as of major concern, mainly due to their impact on tool selection criteria.

Q-fast[12] is an SQL-based application allowing a fast and exhaustive exploration of the tooling database for turning and milling operations. *Specific Type of Operations, Test Objectives, Materials* and *Benefits,* were the four main options considered.

Fuzzy-K[13] is an application having the advantages of hierarchical clustering, while applying fuzzy membership functions to support the generation of similarity measures. The implementation of fuzzy membership functions helped to optimise the grouping of categorical data containing missing or imprecise values.

MQG[14] is an SQL-based *Multiple Query Generator* application developed to provide cross-information retrieval in relation to turning and milling operations. The user is able to fix input conditions chosen among 30 tooling parameters, weigh the importance of these parameters and select output variables of his/her interest among 32 possible tooling attributes. *R-Set*[14] is a rough sets-based application developed to identify data inconsistency and provide ways to analyse the influence of tooling parameters (condition variables), on pre-fixed output attributes (decision variables).

In relation to the deployment of the systems developed in this research, ***TTS*** has been successfully and initially installed on six laptops (Seco-Birmingham, UK) and now the system is being fully operated from the machining centres where the tooling trials are conducted. Subsequent installations would involve users in other UK-based branches and other international offices (like Sweden headquarters), provided that the format of tooling reports can be properly standardised and organisational details appropriately arranged.

In the case of ***SELTOOL***, due to its Web-based orientation and free-access philosophy, it has been, since its development, available to Seco customers and world-wide users. ***SELTOOL*** provides an interactive and more efficient way of searching for tooling parameters (crossed-searches concentrated in only one screen, for example), than conventional representation schemes provided by catalogues.

DISKOVER has been fully tested in the development environment. Its final presentation and delivery to the industrial environment (Seco-UK) will, it is anticipated, be carried out shortly. The development of this system

[12] Developed by the author.

[13] Developed jointly by the author and another research student in the group at Durham.

demonstrated the significant potential of the described multi-strategy methodology (different and complementary data mining techniques) in solving problems of knowledge discovery. In contrast to the two Internet-based systems previously mentioned, which were built using *Java-applets*, ***DISKOVER*** is also an Internet-based system, but built using a *Java-application*. *Java-applications* do not need to be embedded in HTML files and downloaded using Web-browsers, which is a distinguishing characteristic of *Java-applets*. This constitutes an important operational advantage, considering the computational time and memory resources demanded by data mining algorithms, and the execution of database operations through the Internet, which is a relatively slow access platform. To satisfy faster information retrieval operations, SQL-based exploratory data applications were additionally implemented.

In the author's opinion, Web-based manufacturing and information integration technologies would favour current manufacturing practices if:

- There was a standardised vision of what distributed manufacturing is and an international consensus on the underlying concepts for the benefit of the whole business community.
- There was a unified interface based on the previous consensus, able to provide standard data exchange formats, CAD-enabled browsers, remote collaborative work and a vendor-independent information technology environment.
- There was a networked environment to support knowledge sharing activities in which people, enabling technologies and organisational elements, can be closely integrated.

These are some of the challenges to be faced, in the future, by the manufacturing community. Although the Internet is an early example of the information networks of the future and is increasingly being commercially exploited, it is not automatically an obvious panacea for business success. A company being incorrectly managed and erroneously oriented through conventional operational methods, will continue being a badly managed

[14] Developed by a research student in the group at Durham.

company, even while operating under ever evolving Internet-based policies. These considerations indicate the need for serious decision making prior to adopting Internet-based strategies. There are some specific and some strategic company activities, more suitable than others for support by Internet-based operations.

The results of this investigation have implications for both research on information integration technologies and practices in companies. From a research perspective, the implications are for the development of systems supporting dynamic networked environments. From the enterprise practices point of view, current manufacturing tasks are being oriented towards agile responsiveness scenarios within a distributed environment, offering high flexibility during complex product design activities. This research proposes Web-based collaborative work strategies and knowledge capturing systems, implemented across company borders. There are clear benefits to be obtained when processes are carried out using an Internet-based distributed approach:

- Efficient access to resources over a geographically dispersed area.
- Cheaper information exchange processes.
- Closer interaction between Clients and Companies.
- Major support to assimilate the company growth.
- Improved distribution of software and hardware resources.

As Internet-oriented 3-D technologies become more commonplace in organisations, new potentialities are certain to arise. In particular, information representation and information sharing. Virtual social interaction will also increase rapidly. All these improvements will impact on the way in which remote work groups are collaboratively joining efforts to design, manufacture and deliver new products. It is the author's expectation that this thesis will arouse more attention from the manufacturing community, in relation to a better use of Internet-based technologies, in order to deal with a growing world-wide standardisation of processes and the emergence of automated infrastructures around global information networks.

9.3 RECOMMENDATIONS FOR FURTHER WORK

Because this research has been closely linked to the tooling industry, considerations for further work in this sector are given, as well as for the academic environment.

9.3.1 Recommendations for Further Academic Research

As the WWW continues to expand and become more prevalent in the life of the average consumer, a logical consequence is the diversification of the type of services available on the Web and a continued increase in the number of these services. The WWW has already become far more than a distribution of digital resumes and static home pages. It has shifted from the original concept of serving formatted text and simple graphics, to providing a wide range of applications. Examples of important industrial applications include, monitoring and diagnosis, networked assembly, knowledge discovery and simulation of production processes. The implementation of these applications using Web-based approaches, constitutes potential areas of investigation that still remain issues for research and development.

The participation of remote collaborative work teams, for integrating geometric modelling, product design and product manufacturing stages (concurrent engineering) is a matter of intensive research. There are few hard results available for such systems, however, and further investigation is required, taking care to demonstrate the benefits that a *distributed architecture*, based in the WWW, could bring to the problems which traditional techniques do not.

To date, virtual reality (VR) techniques have been mainly focused towards applications in relation to sensing and manipulating objects in virtual environments, with important implementations in the field of simulation. However, significant investment and effort are being dedicated to improve VR technology and, in the near future, using VR methods, it should be possible to explore databases exploiting the benefits of manipulating records of data supported by virtual immersive environments. Another important issue is the ability of certain systems to support multiple representations of the same data sets. The more complex the data sources, the more likely that different perspectives on the data will be required in order to fully characterise patterns and trends of interest.

Hence, fully immersive virtual reality techniques operating in distributed and shared scenarios and the development of faster CAD-enabled browsers managing multidimensional views of the model, are promising and still open fields where to conduct further research. Future Web-based tools will be developed taking into consideration the presentation of findings from different perspectives and multidimensional schemes, in order to provide more transparent and clear frames where end users can easily interpret the results.

During the process of applying KDD methods, it should be noted that the more data is available, the higher the potential to discover hidden knowledge. Therefore, improving access to complementary and widely dispersed information sources will contribute to provide a dominant position from which KDD tools would be applied. The amount of remotely dispersed computers, interconnected through a World Wide Web (WWW), opens a promising path to implement KDD-oriented techniques for analysing geographically dispersed databases. However, in WWW technology the bandwidth is often a bottleneck, and the flow of information accessed through the Internet is not as efficient as expected. Likewise, Cache-Web is the simplest cost-effective way to achieve a high-speed memory hierarchy. This provides research potentialities in the fields of parallel processing, bandwidth and Cache-Web.

Obviously, all the approaches already discussed demonstrate their usefulness in distributed scenarios, rather than stand-alone work philosophies. An important factor allowing a better access to geographically alienated information sources, are the expanding powerful features of the Internet. Such features include the emergence of "Mobile Internet", oriented to access the Web through devices without physical connection capabilities (satellite-based networks). Although this technology is not entirely new, it remains a potential area for further investigation.

The notion of Multi-Agent Systems (MAS) is another important concept that is gaining increasing application in different areas of management and control. They have been proposed as a new tool to integrate distributed objects. MAS allow integrated enterprise environments at the level of information access, monitoring, automation, co-operative work and system integration. *Multi-agent* technology provides an implementing framework for co-ordinating behaviour among a collection of autonomous agents. Although the term *agent* is used frequently, its universal meaning, definition and structure still remain vague, especially when used in different contexts. Ming *et al.* (1998) suggested that an *agent* is an entity that can perform some tasks and achieve a predetermined goal autonomously. According to this definition, human experts, intelligent CAD systems and intelligent machining cells are all agents of an intelligent manufacturing system. The study of Multi-agent systems is an area suitable for further investigation and may lead to better management of distributed manufacturing activities.

9.3.2 Recommendations for the Tooling Industry

The proposals suggested are particularly oriented to the company Seco Tools Ltd (UK). It constitutes the primary target where the applications in this research were focused.

In chapter 3, two Web-based systems to support tooling operations from different perspectives were considered. A natural extension of these works would be the incorporation of additional capabilities to the existing systems (or develop new ones) including functions such as *Technical Assistance, Marketing and Customer Servicing Order and Collaborative Distributed Design and Manufacturing,* all of them implemented under Web-based architectures.

In this research, it was needed to develop Java-based conversion programs to translate data stored in varied formats (PDF and Access) to an SQL-based DBMS. Other databases are built using DB2 (IBM proprietary). To overcome this and another technical difficulties when implementing the ideas of including new functions (previous paragraph), it would be beneficial if Seco could rely on a unified DBMS. This will allow a major flexibility, wider range of services made available to the customers and all the benefits that a standardised database platform could provide.

The set of tests conducted in chapter 8 reflected an insufficient amount of data in the current tooling databases. Important materials, such as aluminium and lighter materials known as super alloys were not found. It is important to update the current database population with these materials, given their increasing role in manufacturing sectors such as aerospace, construction and automotive industries.

TTS, ***SELTOOL*** and ***DISKOVER*** were developed under Windows operating system. Nevertheless, the backbone of the communicational structure of Seco is supported by a tool known as *Lotus Notes*. Basic operations include E-mail, Usenet Newsgroups, purchase order transactions and a variety of support systems. It would be interesting to evaluate the technical feasibility of managing ***TTS***, ***SELTOOL*** and ***DISKOVER*** directly from the Lotus Notes environment, to increase the transparency and easier accessibility of these systems.

A final and more important recommendation is oriented towards a wider application of knowledge discovery technology in other sectors of the company. Until now, ***DISKOVER*** has demonstrated its usefulness in classifying experimental tooling data, particularly oriented in relation to cutting data recommendations. However, KDD technology has proved to be a remarkable tool for analysing Financial, Marketing, Forecasting, Sales and Production data. KDD technology could be implemented by the company, in order to make sense of the data found in the above mentioned areas, and using the discovered knowledge for decision-support purposes.

APPENDIX A

Hierarchical Clustering Methods – A Numerical Application

In order to illustrate the procedure followed by hierarchical clustering methods to carry out data grouping processes, a basic numerical exercise applying *Single, Complete* and *Average Linkage* methods will be presented.

The Euclidean distance function will be assumed and the following two-dimensional data points are considered:

T = [**1.0 0.9**], S = [**-2.0 2.1**], U = [**-0.5 1.9**], V = [**1.2 0.7**], W = [**-1.8 1.5**];

Before solving the problem, it is convenient to define a *threshold* value (β) that will be used as a stopping rule. When the distance between the two closest clusters is greater than this *threshold* value, the data clustering process ends.

a) Single Linkage Method.

The initial plotting of all data points is shown in Figure A.1.

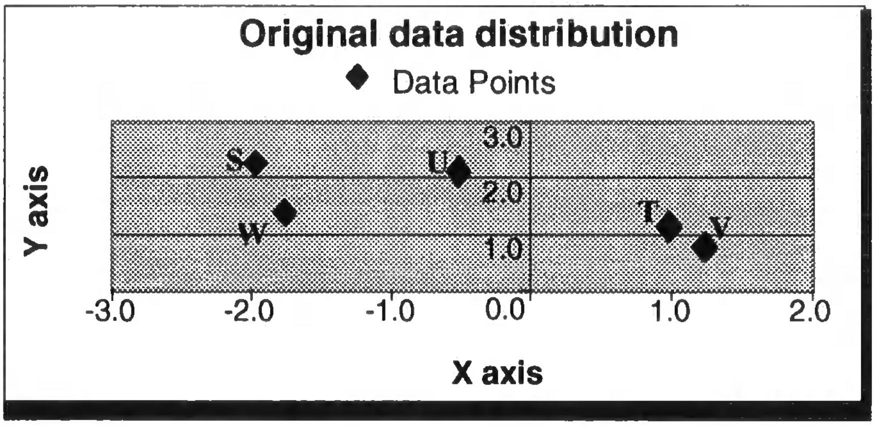


Figure A.1 - Initial data representation.

i) N = 5 clusters; β = 1.55; n = 2;

The Euclidean distance is defined by:

$$\left\| X - Y \right\| = \sqrt{\sum_{i=1}^{n} | Xi - Yi |^2}$$

Applying the former distance function,

$$\|T - S\| = \sqrt{|[(1.0) - (-2.0)]|^2 + |[(0.9) - (2.1)]|^2} =$$

$$= \sqrt{(3.0)^2 + (1.2)^2} = 3.23;$$

$$\|T - U\| = 1.80 ; \|T - W\| = 2.86 ; \|V - W\| = 3.10 ;$$
$$\|U - W\| = 1.74 ; \|U - V\| = 2.08 ; \|S - W\| = 0.63 ;$$
$$\|S - V\| = 3.49 ; \|S - U\| = 1.51 ; \|T - V\| = \boldsymbol{0.28} ;$$

A matricial representation of these distances is given below:

	T	S	U	V	W
T	-				
S	3.23	-			
U	1.80	1.51	-		
V	0.28	3.49	2.08	-	
W	2.86	0.63	1.74	3.10	-

As clusters ***T*** and ***V*** are the closest, and the distance between them is lower than *β*, they are merged into a new cluster named ***TV.***

ii) The new four clusters are:

TV = **{[1.0 0.9] [1.2 0.7]}**, S = [**-2.0 2.1**], U = [**-0.5 1.9**], W = [**-1.8 1.5**];

The resulting merge is shown in Figure A.2.

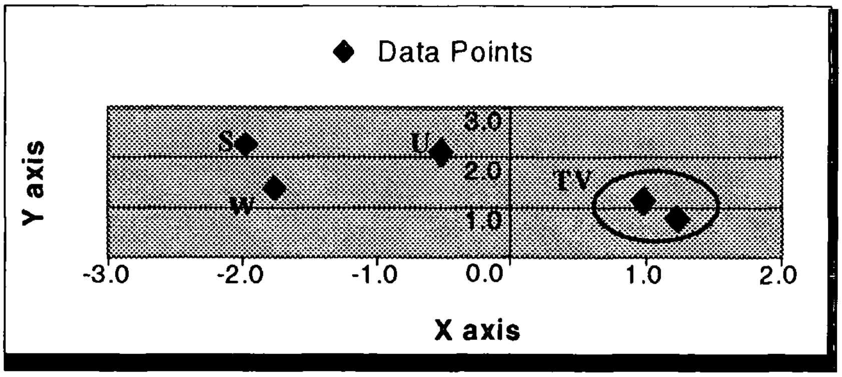


Figure A.2 - First data grouping.

Applying the Euclidean distance function on the new clusters,

$\|S - U\| = 1.51$; $\|U - W\| = 1.74$; $\|S - W\| = \mathbf{0.63}$;

- $\|S - TV\|$:

$\|S - T\| = 3.23$; $\|S - V\| = 3.49$; $\|S - TV\| = 3.23$;

- $\|U - TV\|$:

$\|U - T\| = 1.80$; $\|U - V\| = 2.08$; $\|U - TV\| = 1.80$;

- $\|W - TV\|$:

$\|W - T\| = 2.86$; $\|W - V\| = 3.10$; $\|W - TV\| = 2.86$;

A matricial representation is given below:

	TV	S	U	W
TV	-			
S	3.23	-		
U	1.80	1.51	-	
W	2.86	0.63	1.74	-

As clusters ***S*** and ***W*** are the closest, and the distance between them is lower than *β*, they are merged into a new cluster named ***SW.***

iii) The new three clusters are:

TV = {[**1.0 0.9**] [**1.2 0.7**]}, SW = {[**-2.0 2.1**] [**-1.8 1.5**]}; U = [**-0.5 1.9**],

The resulting merge is shown in Figure A.3.

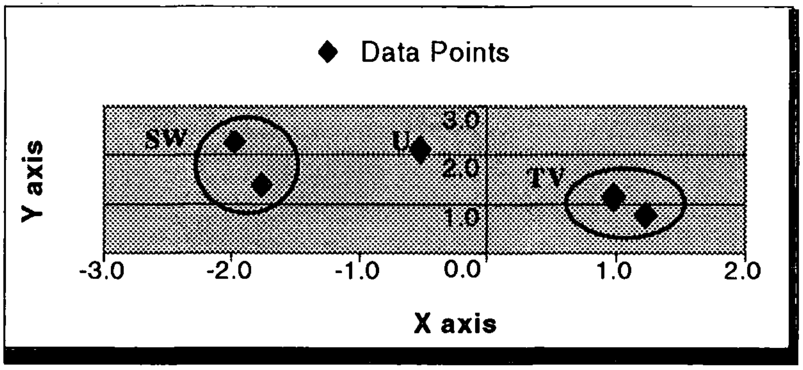


Figure A.3 - Second data grouping

Applying the Euclidean distance function on the new clusters,

- $\|U - TV\|$:

$\|U - T\| = 1.80$; $\|U - V\| = 2.08$; $\rightarrow \|U - TV\| = 1.80$;

- $\|U - SW\|$:

$\|U - S\| = 1.51$; $\|U - W\| = 1.74$; $\rightarrow \|U - SW\| = \boldsymbol{1.51}$;

- $\|TV - SW\|$:

$\|T - S\| = 3.23$; $\|V - W\| = 3.10$; $\|T - W\| = 2.86$;
$\|V - S\| = 3.49$; $\|TV - SW\| = 2.86$;

A matricial representation is given below:

	TV	SW	U
TV	-		
SW	2.86	-	
U	1.80	1.51	-

As clusters ***SW*** and ***U*** are the closest, and the distance between them is lower than *β*, they are merged into a new cluster named ***SWU.***

iv) The new three clusters are:

TV = **{[1.0 0.9] [1.2 0.7]}**, SWU = **{[-2.0 2.1] [-1.8 1.5] [-0.5 1.9]}**;

The resulting merge is shown in Figure A.4.

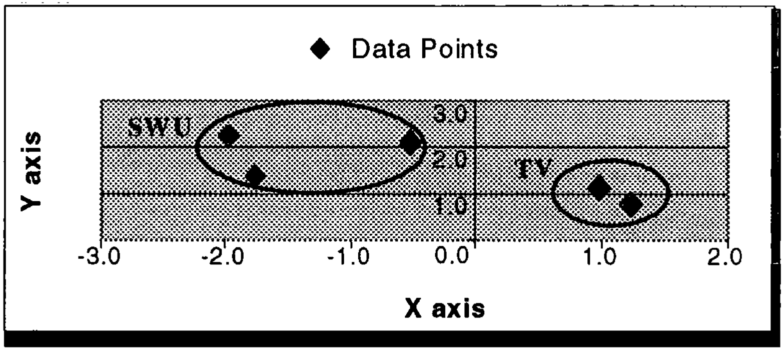


Figure A.4 - Third data grouping.

Applying the Euclidean distance function on the new clusters,

- $\left\| TV \ - SWU \ \right\|$:

$\left\| T - U \right\| = 1.80$; $\left\| T - W \right\| = 2.86$; $\left\| T - S \right\| = 3.23$;

$\left\| V - U \right\| = 2.08$; $\left\| V - W \right\| = 3.10$; $\left\| V - S \right\| = 3.49$;

$\left\| TV \ - SWU \ \right\| = 1.80$;

As the distance between clusters ***SWU*** and ***TV*** (1.80) is greater than β (1.50), their merge is not suitable and the clustering process ends with these two final clusters.

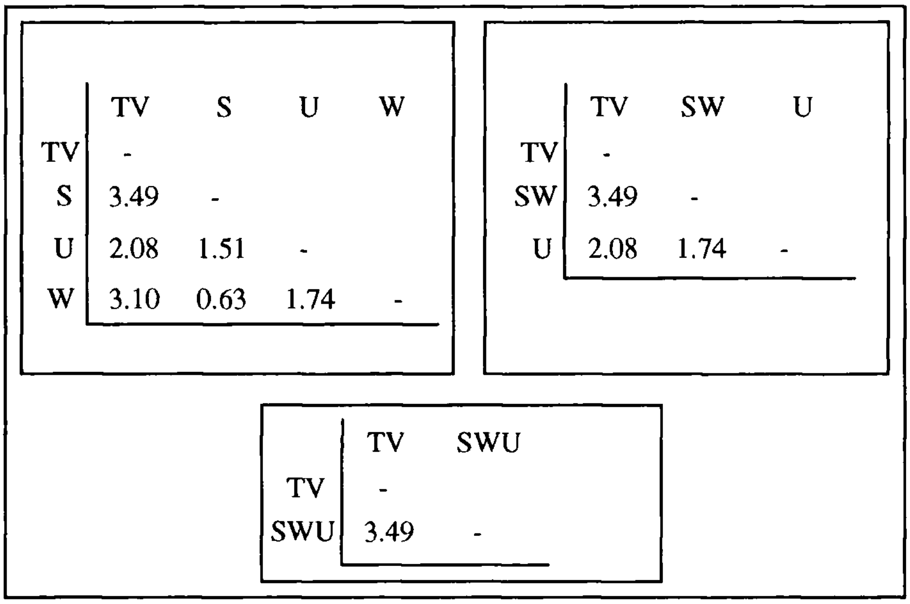
b) <u>Complete linkage method</u>:

N = 5 clusters; β = 3.0; n = 2;

i) The first step is identical to the procedure applied to the single linkage method. The clusters ***T*** and ***V*** are the closest, and the distance between them is lower than β, so, they are merged into a new cluster named ***TV***:

TV = **{[1.0 0.9] [1.2 0.7]}**, S = [**-2.0 2.1**], U = [**-0.5 1.9**], W = [**-1.8 1.5**];

ii) Applying the Euclidean distance function on the new clusters, the generated dissimilarity matrices are shown in Figure A.5.



	TV	S	U	W
TV	-			
S	3.49	-		
U	2.08	1.51	-	
W	3.10	0.63	1.74	-

	TV	SW	U
TV	-		
SW	3.49	-	
U	2.08	1.74	-

	TV	SWU
TV	-	
SWU	3.49	-

Figure A.5 - Successive iterations.

As the distance between clusters ***SWU*** and ***TV*** (3.49) is greater than *β* (3.0), their merge is not recommendable and the clustering process ends with the same two final clusters, as found in the single method.

c) Average linkage method:

N = 5 clusters; *β* = 2.25; n = 2;

i) The first step is identical to the procedure applied to the single linkage method. The clusters ***T*** and ***V*** are the closest, and the distance between them is lower than *β*, so, they are merged into a new cluster named ***TV***:

TV = **{[1.0 0.9] [1.2 0.7]}**, S = [**-2.0 2.1**], U = [**-0.5 1.9**], W = [**-1.8 1.5**]

ii) Applying the Euclidean distance function on the new clusters,

$$\|S - U\| = 1.51 \; ; \; \|U - W\| = 1.74 \; ; \; \|S - W\| = \mathbf{0.6}3;$$

- $\|S - TV\|$:

$$\|S - V\| = 3.49 \; ; \; \|S - T\| = 3.23 \; ; \; \|S - TV\| = 3.36 \; ;$$

The former value (3.36) was obtained applying the equation:

$$\|S - TV\| = \frac{1}{Card(S)Card(TV)} * \sum_{x \in S, y \in TV} \|X - Y\|;$$

$$\|S - TV\| = \frac{3.23 + 3.49}{(1) * (2)} = 3.36 \; ;$$

The remaining values shown below were obtained in the same way.

- $\|U - TV\|$:

$$\|U - T\| = 1.80 \; ; \; \|U - V\| = 2.08 \; ; \; \|U - TV\| = 1.94 \; ;$$

- $\|W - TV\|$:

$$\|W - V\| = 3.10 \; ; \; \|W - T\| = 2.86 \; ; \; \|W - TV\| = 2.98 \; ;$$

As clusters ***S*** and ***W*** are again the closest, and the distance between them is lower than *β*, they are merged into a new cluster named ***SW.***

iii) The new three clusters are:

TV = **{[1.0 0.9] [1.2 0.7]}**, SW = **{[-2.0 2.1] [-1.8 1.5]}**; U = **[-0.5 1.9]**

Applying the Euclidean distance function on the new clusters, the generated dissimilarity matrices are shown in Figure A.6.

	TV	S	U	W
TV	-			
S	3.36	-		
U	1.94	1.51	-	
W	2.98	0.63	1.74	-

	TV	SW	U
TV	-		
SW	3.17	-	
U	1.94	1.63	-

	TV	SWU
TV	-	
SWU	2.76	-

Figure A.6 - Successive iterations.

As the distance between clusters ***SWU*** and ***TV*** (2.76) is greater than *β* (2.55), their merge is not suitable and the clustering process ends with the same two final clusters, as found in the single and complete method.

APPENDIX B

Applying Rough Sets to Detect Data Inconsistency

In order to show how rough sets can be used to identify inconsistent relationships in a data set, an example is presented as follows.

The results of a series of measurements are summarised in Table B.1.

Table B.1 - Process condition and parameters.

Measure-ment	Pressure (P)	CO_2 Level (CO)	State
1	Normal	Optimum	Stable
2	Normal	Acceptable	Stable
3	High	Optimum	Stable
4	High	Alarming	Unstable
5	High	Acceptable	Unstable
6	Normal	Alarming	Stable
7	High	Optimum	Stable
8	Normal	Optimum	Stable
9	Normal	Acceptable	Unstable
10	Normal	Alarming	Unstable

It is assumed that the condition of a chemical process is monitored. Preliminary observations show that the condition of the process is related to the *Pressure (P)* and *CO_2 Level (CO)*. Two states, namely *Stable* and *Unstable*, are used to describe the condition of the process.

Due to the qualitative description of the variables, they have been transformed into real values, as can be seen in Table B.2. The following conversion scheme was applied:

Normal = 0; High = 1 ; Optimum = 0; Acceptable = 1; Alarming = 2;

Stable = 0; Unstable = 1 ;

From Table B.2, two possible concepts can be defined:

Concept 1: $\boldsymbol{C_1} = \{x_1, x_2, x_3, x_6, x_7, x_8\}$ → Class = 0 (Process status = *Stable*);
Concept 2: $\boldsymbol{C_2} = \{x_4, x_5, x_9, x_{10}\}$ → Class = 1 (Process status = *Unstable*);

Table B.2 - Process condition after transformation.

Measurement	Pressure (P)	CO$_2$Level (CO)	State
1	0	0	0
2	0	1	0
3	1	0	0
4	1	2	1
5	1	1	1
6	0	2	0
7	1	0	0
8	0	0	0
9	0	1	1
10	0	2	1

According to the set of attributes A = {P, CO}, the following indiscernible relationships can be identified:

$X_1 = \{x_1, x_8\}$; $X_2 = \{x_2, x_9\}$; $X_3 = \{x_3, x_7\}$; $X_4 = \{x_4\}$; $X_5 = \{x_5\}$; $X_6 = \{x_6, x_{10}\}$;

The above sets are commonly called elementary sets or equivalent classes. Using the notations previously described, the approximation can be obtained as follows.

Analysing Concept 1:

Lower Approximation → $\underline{A}C_1 = \{x_1, x_3, x_7, x_8\}$;
Upper Approximation → $\overline{A}C_1 = \{x_1, x_2, x_3, x_6, x_7, x_8, x_9, x_{10}\}$;
Boundary region → $BN_A(C_1) = \overline{A}C_1 - \underline{A}C_1 = \{x_2, x_6, x_9, x_{10}\}$;

Analysing Concept 2:

Lower Approximation → $\underline{A}C_2 = \{x_4, x_5\}$;
Upper Approximation → $\overline{A}C_2 = \{x_2, x_4, x_5, x_6, x_9, x_{10}\}$;
Boundary region → $BN_A(C_2) = \overline{A}C_2 - \underline{A}C_2 = \{x_2, x_6, x_9, x_{10}\}$;

Clearly, the boundary (doubtful) region for both concepts (C_1 and C_2) indicates that the results of measurements 2 and 9 contradict one another, as well as measurements 6 and 10. These cases have been highlighted in Table B.2.

This example has shown how using rough sets theory, some data inconsistencies have been identified.

APPENDIX C

Selected Data Mining Applications

This Appendix presents some selected applications where data mining techniques have been applied.

C.1 Regression Analysis

Alamin (1996) implemented an interesting application of multiple regression techniques in the area of tooling, developing a *Tool Life Prediction* (TLP) module. A database of cutting conditions and tool life data for a wide range of carbide tools and different workpiece materials was analysed. He applied multiple regression to calculate the theoretical tool life coefficients $(\ln C, \frac{1}{\alpha}, \frac{1}{\beta})$, of the extended form of Taylor's equation:

$T = \frac{C}{v^{1/\alpha} s^{1/\beta}}$; Where ***T*** is the tool life, ***v*** is the cutting velocity (m/min) and ***s*** is the feed rate (mm/rev).

The essential information required for TLP is in relation to the cutting conditions (cutting velocity, feed rate and depth of cut), tool code, insert grade, type of cutting fluid, machine tool and the material class and sub-class.

TLP predictions are based on the optimisation of cutting data using three tool life criteria namely, user defined tool life, tool life for minimum production cost or tool life for maximum production rate.

Sales forecasting is a different and more common application of regression techniques. The analysis is supported by historical data where only some significant variables are selected to predict the sales for a particular future period of time, given their influence on actual sales. Figure C.1 shows an example of sales forecasting for the next three months, based on analysis of historical data of previous months.

Forecasting can help an organisation to plan appropriate strategies for long-term growth.

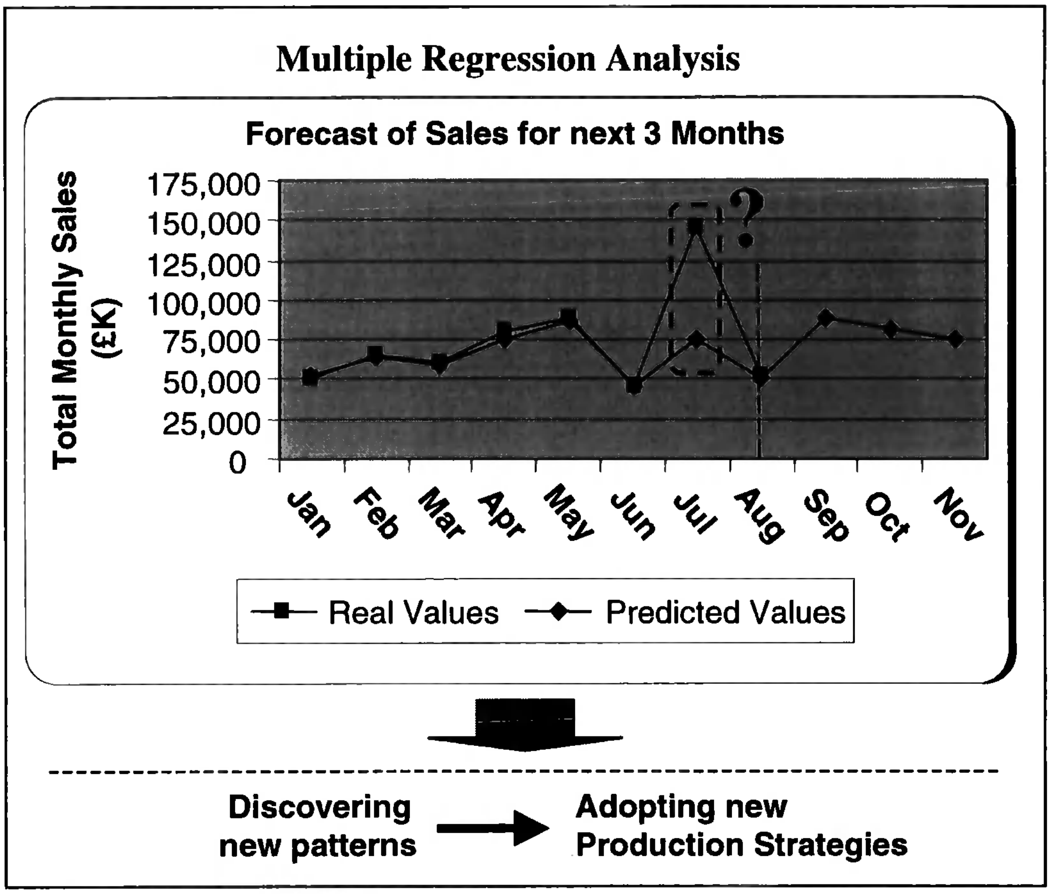


Figure C.1 – Multiple Regression applied to Sales prediction.

Carrying out a comparative analysis of projected versus real sales can result in discovering not only long-term linear trends, but also short-term cyclical fluctuations (question mark in Figure C.1), which once identified their causes, they can be analysed to obtain better sales predictions and adopt therefore, more real production strategies.

C.2 Neural Networks (NN)

A successful application of NN, the publication of which has been authorised by Prof. Raymond Burke, is described below.

An insurance company has identified in its database, one million customers without additional insurance. Assume the company prepared a direct mail shot offering a special promotion on additional insurance coverage, to send to all one million customers and subsequently receive ten thousand responders (one percent response rate). However, due to promotion budgets, companies of this size would typically not mail to all prospects on a list, but instead would chose randomly or by predictive methods a subset of this list towards which to direct the campaign.

Assume there is now a budget that restricts this subset to one hundred thousand customers (ten percent of the target set of customers). To select this subset three different methods were employed. Firstly, using a random selection only 1000 customers were caught, secondly, implementing *RFM* (*R*ecency/*F*requency/*M*onetary) which is a predictive technique that sorts a list of prospective customers according to their recent purchases, how often they purchase, and how much they purchase, they caught 2000 responders. Thirdly, 4000 answers were received applying Neural Networks.

Assume each direct mail piece costs the company $1.0 (including mailing costs) and each responder represented $100 in annual profit. Table C.1 shows the gross profit for each alternative discussed above.

Table C.1 - Results of NN example. Reproduced with permission of Prof. Raymond Burke, Indiana University, USA.

	Pieces	Cost per piece	Marketing Costs	Respon-ders	Response Rate	Annual Revenue per Responder	Revenue	Gross Profit
Random	100.000	$ 1	$ 100.000	1.000	1 %	$ 100	$ 100.000	$ 0
RFM	100.000	$ 1	$ 100.000	2.000	2 %	$ 100	$ 200.000	$ 100.000
Neural Network	100.000	$ 1	$ 100.000	4.000	4 %	$ 100	$ 400.000	$ 300.000

It is evident from the above example that there are significant financial gains to be generated by employing a neural network system in terms of gross profit benefits.

C.3 Genetic Algorithms

Darwin is a data mining commercial tool that implements Genetic Algorithms through one of its modules called StarGene. The main function of StarGene is to optimise the parameters used by data mining algorithms of other modules. For instance, StarGene can be used to optimise the interconnection weights of neural networks in the StarNet module (Freitas & Lavington, 1998).

Another important application of Genetic Algorithms in the financial area, has been developed by the company Rabatin Investment Technology Ltd through its project, Adaptive Portfolio Trading (APT), which involves the development of self-learning, self-adapting intelligent trading models, for portfolios of financial instruments.

The design of trading models developed within the APT system is based on the following requirements with the aim being not to maximise predictability of market prices, but to maximise consistency and predictability of trading performance. Namely,

- Integrating all aspects of the trading/investment decision into one complete decision-making model, which incorporates:

- Market Selection Decision.
- Portfolio Allocation Decision.
- Buy/Sell Decision.
- Market Price Risk Analysis.
- Portfolio Risk Analysis and Portfolio Risk Management Decision.

- Application of real-time constraints, such as user-defined risk thresholds, allocation restrictions, defined by the trading manager, throughout the entire training process.
- Designing trading models, as distributable objects, that can be executed across a network and allowing for performance to be replicated on several locations (but performing the training process centrally).

- Creating adaptive models that can learn and adapt without human interference.

Another interesting application of Genetic Algorithms has been implemented in the Operations Research area, for solving the classic Travelling Salesman Problem. In this problem the goal is to find the shortest distance between N different cities.

C.4 Decision Trees

Figure C.2 illustrates an example of Decision Trees using an algorithm called CHAID (Chi-square Automatic Interaction Detector) oriented to find the characteristics of a person likely to respond to a direct mail piece. These characteristics can then be translated into a set of rules.

It is shown in Figure C.2 that 8% of all people who received a direct mail piece responded to the offer. However, if we split the group into those who own their home versus those who do not, we can see that 17% of renters responded to the piece whereas only 6% of owners responded. It is possible to continue separating the group into segments to find a segment most likely to respond.

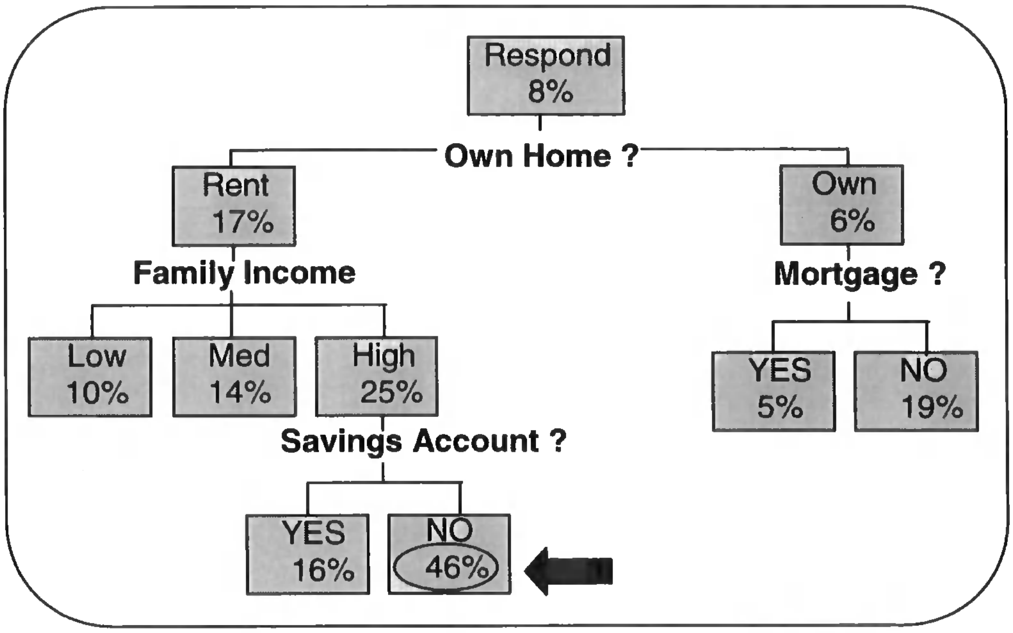


Figure C.2 - Decision Tree application.

This segment can then be expressed as a rule such as "If the recipient rents, and if the recipient has high family income, and finally if the recipient does not have a savings account, then that recipient is likely to respond with probability of 46%. Or more simply, 46% of the segment with those characteristics is likely to respond to a direct mail piece.

Another commercial application has been implemented by the company PMSI through one of its products called *Galvano Decision Trees Data Mining*. This tool establishes a link between descriptions of people (selected attributes) to the financial risk which they represent. The data comes from files of loans like those filed in the banks. The borrowers are described by a whole series of attributes: age, family circumstances, number of children, incomes, assembled saving, amount and duration of the authorised loan, etc, in order to determine a description of a good or bad course of the loan. The goal consists in finding the relationships between the various descriptive variables of the borrowers and the Good one or Bad risk.

C.5 Discriminant Analysis

A typical example of applying a discriminant analysis technique consists of identifying common customer group behaviours. Figure C.3 displays the distribution of a group of customers who show similar patterns when renewing certain products of commercial and extensive use (videos, computer games, CDs, car hiring, bicycles, etc.).

The analysis consists of testing the factors leading to the non-renewal of a determined product. If it is possible to identify the right factors, the model should be able to use these factors to *discriminate* between those likely to renew and those likely to not renew. Observing the probabilistic distribution of customers in Figure C.3, it is possible to say that this outcome constitutes a 'successful' discriminant analysis. The fitted model was able to find factors to separate those who renewed, from those who did not renew. Inevitably, there will be some variability in the scores within each of the groups, as is shown by the distribution of probabilities. A 'successful' discriminant analysis will be able to minimise the amount of overlap between these two distributions.

Another main goal of the above application is to make a prediction or tentative allocation for an unclassified entity (new customer).

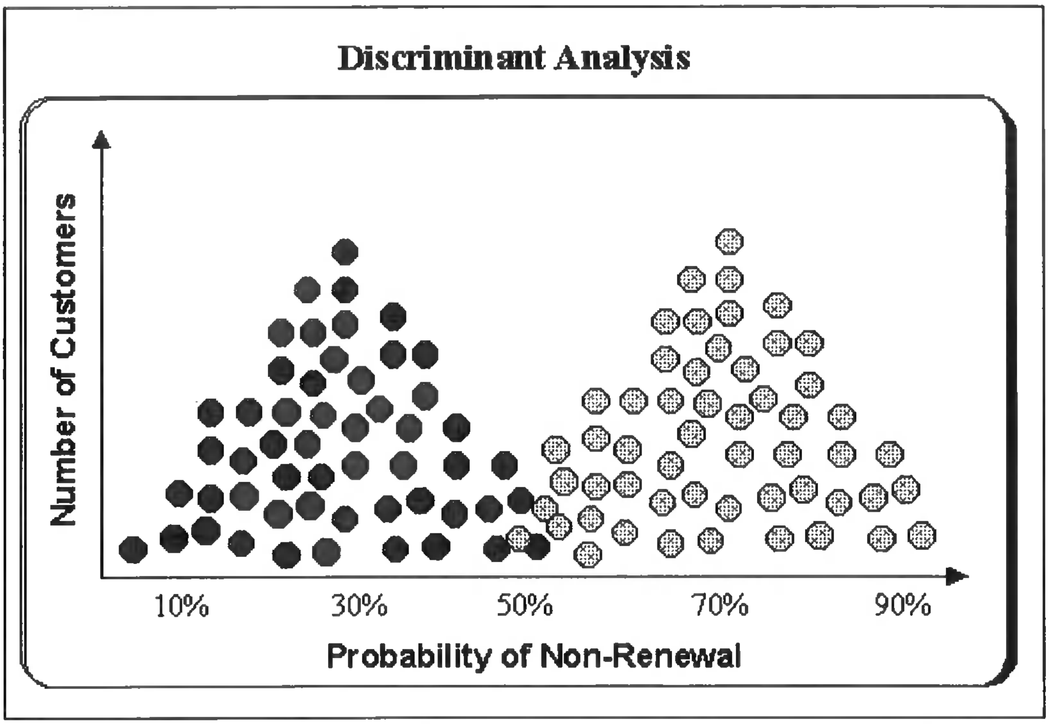


Figure C.3 – Discriminant Analysis to classify similar customer behaviours.

In another example concerning prediction, an economist may wish to forecast, on the basis of his/her most recent accounting information, those members of the corporate sector that might be expected to suffer financial losses leading to failure. For this purpose, a discriminant rule may be formed from accounting data collected on failed and surviving companies over many past years (McLachlan, 1992).

Examples where prediction or tentative allocation have to be made for an unclassified entity, occur frequently in medical diagnosis, when the definitive classification of a patient often can be made only after exhaustive physical and clinical assessments or perhaps even surgery. In some instances, the true classification can be made only on evidences that emerge after the passage of time, for instance, an autopsy. Hence, frequent diagnostic tests are performed, based on clinical and laboratory-type observations without too much inconvenience to the patient.

APPENDIX D

Regression and Correlation Analysis

Multiple Linear Regression and Correlation Analysis are two statistical techniques widely applied in the area of KDD. This Appendix addresses their analyses and applicability.

D.1 Multiple Linear Regression Analysis

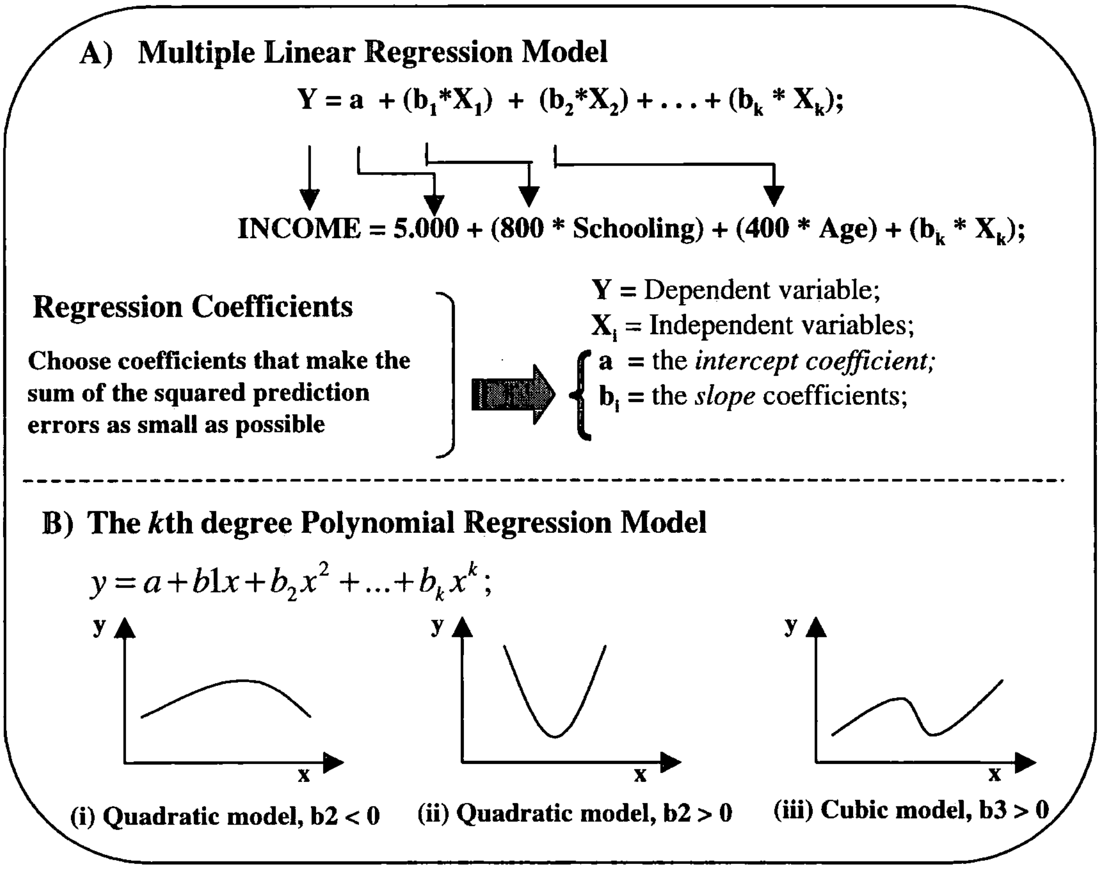
Multiple Linear Regression is a statistical method for studying the relationship between a single *dependent* variable and one or more *independent* variables, (Allison, 1999).

The term *Multiple* involves the existence of two or more independent variables. *Linear* means that the relationships between variables can be represented on a linear equation; in turn, a linear equation gets its name from the fact that if we graph the equation we get a straight line. The term *Regression* is harder to explain and it is associated with the early works undertaken by Sir Francis Galton (1822-1911) who used a linear equation to describe the relationship between heights of fathers and sons. He noticed that fathers tended to have sons who were taller than they were. He called this phenomenon “regression to the mean”, and somehow that name was associated to the entire method.

There are two major uses of multiple regression: prediction and causal analysis. In a prediction study, the goal is to develop a formula for making predictions about the dependent variable, based on the observed values of the independent variables. In a causal analysis, the independent variables are regarded as causes of the dependent variable, being the aim of the study to determine whether a particular independent variable *really* affects the dependent variable, and to estimate the magnitude of that effect, if any.

For prediction studies, multiple regression *combines* many variables to produce optimal predictions of the dependent variable. For causal analysis, it *separates* the effects of independent variables on the dependent variable, trying to examine the unique contribution of each variable.

The main objective of regression analysis is to find appropriate *regression coefficients* to determine an optimum value of a dependent variable given any values assigned to the independent variables. Figure D.1-A shows the basic elements of multiple linear regression with *k* independent variables, using *Schooling* and *Age* as predictor variables of a person's *Income*.



A) Multiple Linear Regression Model

$$Y = a + (b_1 * X_1) + (b_2 * X_2) + \ldots + (b_k * X_k);$$

$$INCOME = 5.000 + (800 * Schooling) + (400 * Age) + (b_k * X_k);$$

Regression Coefficients

Choose coefficients that make the sum of the squared prediction errors as small as possible

Y = Dependent variable;
X_i = Independent variables;
a = the *intercept coefficient;*
b_i = the *slope* coefficients;

B) The *k*th degree Polynomial Regression Model

$$y = a + b1x + b_2 x^2 + \ldots + b_k x^k;$$

(i) Quadratic model, b2 < 0 **(ii) Quadratic model, b2 > 0** **(iii) Cubic model, b3 > 0**

Figure D.1 – Basic elements of Multiple Regression (Linear and Polynomial).

When simple linear regression is applied, the methods used to calculate the coefficients *a* and b_i are too simple, as can be seen for the *equations (D.1)* and (*D.2)*; however, the computations become quite complicated as the number of independent variables increases (multiple regression).

$$b = \frac{\sum_i (y_i - \bar{y})(x_i - \bar{x})}{\sum_i (x_i - \bar{x})^2};$$

Equation (D.1)

$$a = \bar{y} - b\bar{x};$$

Equation (D.2)

The Ordinary Least Squares is the method most often used to get optimum values for the regression coefficients. The goal is to compute those coefficients that make the sum of the squared prediction errors as small as possible. The sum of the squared errors is given by *Equation (D.3).*

$$\sum_i [y_i - (a + bx_i)]^2;$$

Equation (D.3)

There are some cases when linear regression models are not appropriate to represent the data being analysed, so, different models must be considered. Figure D.1-B shows an example of a polynomial regression model, plotting some common polynomial regression functions.

There are non-linear relationships that can be appropriately transformed before applying a linear analysis, however, complicated methods must be used and there are also relationships that can not be linearised by transformations. A discussion of non-linear regression methods is beyond the scope of this section.

The main drawback of multiple linear regression analysis is that all variables are required to be continuous (and thus, the attribute to be predicted). The use of categorical values (low/high, present/absent, etc.) results inappropriate and requires conversion mechanisms that sometimes do not guaranty a good determination of the magnitude of the difference between two values, which is crucial in regression analysis.

D.2 Correlation Analysis

Another way of describing the relationship between two variables is using the *Correlation Coefficient*, more precisely called the *Pearson Product-Moment Correlation Coefficien*. Figure D.2 shows scatterplots for some correlation coefficient values.

The correlation coefficient can have any value between -1 and +1. If the correlation between *x* and *y* is +1, then there is a perfect linear relationship between the two variables. That means that if we draw a scatterplot for data on *x* and *y*, the points will all lie exactly on a straight line, as in Figure D.2-a.

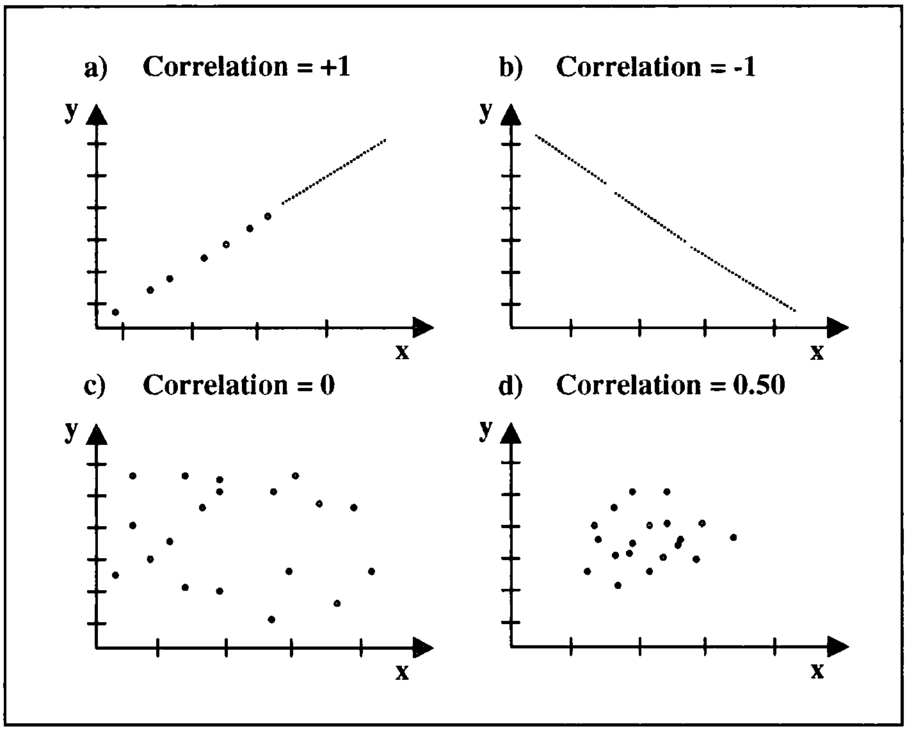


Figure D.2 – Scatterplots for various values of correlation coefficients.

In Figure D.2-b a scatterplot for a correlation of -1 is showed. Again there is perfect linear relationship between *x* and *y*, but now it is an *inverse* relationship: as *x* increases, *y* decreases.

In Figure D.2-c, we see a scatterplot for two variables that have a correlation of 0. In this case there is not linear relationship between *x* and *y*. If we were to draw a least squares regression line for Figure D.2-c, it would be a horizontal line passing through the mean of *y*. In other words, the slope would be 0.

Finally, Figure D.2-d shows a scatterplot corresponding to a correlation of 0.50. In this graph, there is a relative tendency for y to go up as x goes up, so, if we drew a least square regression line, there would be a good deal of scatter around the line. In general, it is possible to affirm that the correlation coefficient measures the *degree of scatter* around a regression line (Allison, 1999).

Regression and correlation are closely related. One way is when the slope coefficient is 0, then the correlation coefficient must also be 0, and vice versa. The correlation coefficient is often denoted by the letter *r* and can be written as:

$r_{xy} = \frac{S_{xy}}{S_x S_y}$; This expression says that the correlation is equal to the covariance of *x* and *y* divided by the standard deviation of *x* multiplied by the standard deviation of *y*. One point to observe here is that both *r* and *b* (the regression slope coefficient) have the same numerator (the covariance). That means that if one of them is equal to zero, the other must also be zero. It also has a major practical implication: testing the hypothesis that *r* = 0 is equivalent to testing whether *b* = 0, so, there is no need for two distinct tests.

APPENDIX E

Table E.1 - Test Data for Turning Operation (TTS).

Test #	Seco Material Group	Grade	Chipbreaker	Insert Type
1	1	TP100	M3	WNMG
2	8	TP300	M3	WNMG
3	14	TP05	M4	CNMA
4	6	TP100	M3	SNMG
5	2	TP200	M5	SNMG
6	12	TP100	MF2	WNMA
7	3	TP200	R4	DNMM
8	12	TP05	M4	CCMT
9	10	TP40	MF3	CNMG
10	9	TP15	F1	VBMT
11	1	TP300	F1	VBMT
12	5	TP15	MF2	WNMG
13	3	TP200	M5	TPMR
14	4	TP25	M3	TPMR
15	10	CP50	MF1	WNMG
16	9	TP300	M3	WNMG
17	13	TP10	MF2	CNMG
18	7	TP20	M5	TNMG
19	15	TP100	M3	CNMG
20	12	TP10	M4	CNMG
21	3	TP100	MF2	DNMG
22	1	TP200	MR7	SNMG
23	8	TP40	M3	TPMR
24	6	890	MF1	SNMG
25	11	TX150	M5	SNMN
26	4	TP200	M3	CNMG
27	13	TX10	M5	TPUN
28	14	TX150	M5	SNMG
29	7	TP100	MF2	DNMG
30	2	TP200	M5	DNMM
31	10	TP35	MR7	CCMT

APPENDIX F

Publications

[1] *'Cluster Analysis Applied to Manufacturing Tooling Data',* (With M. Velásquez and P. Maropoulos). Proceedings of the Second International Symposium on Engineering of Intelligent Systems, EIS'2000, University of Paisley, Scotland, UK, June/2000.

[2] *'Web-Based Strategies to Support Collaborative Work in the Manufacturing Industry',* (With M. Velásquez). Proceedings of the Second International Conference on Management and Control of Production and Logistics, Grenoble, France, July/2000.

[3] *'An Internet-Based Tool Selection System for Turning Operations',* (With M. Velásquez and P. Maropoulos). Proceedings of the 16th International Conference on Computer Aided Production Engineering, CAPE 2000, Edinburgh University, Scotland, UK, August/2000.

[4] *'An Internet-Based for the Technical Support of Tooling Operations',* (With M. Velásquez). Proceedings of the ASME 20th Computers and Information in Engineering (CIE) International Conference, Baltimore, USA, September/2000.

[5] *'Web-based Strategies to Support Remote Information Exchange in Corporate Environments'* (With M. Velásquez, J. Aguilar and C. Navarro). Submitted to *Automatica,* a Journal of the *IFAC*, June/2000.

[6] *'The Combinatorial Ant System'* (With J. Aguilar and M. Velásquez). Submitted to Journal of Artificial Intelligence, Kluwer Academic, October/2000.

[7] *'Applying Data Mining on Tooling Data Using Rough Sets Concepts on an Internet Platform'* (With M. Velásquez and J. Aguilar). Submitted to the Journal of Artificial Intelligence Review, September/2000.

REFERENCES

Achen, C. '*Interpreting and Using Regression*', Sage Publications, pp 12-37, 1983.

Alamin, B. 'Tool Life prediction and Management for an Integrated Tool Selection System', PhD. Thesis, University of Durham, England, pp 56-87, 1996.

Ames, A., Nadeau, D., and Moreland, J. *'VMRL 2.0 sourcebook'.* John Wiley & Sons, Inc., 1997.

Allison, P. '*Multiple Regression*', Pine Forge Press Inc., pp 1-46, 97-108, 175-180, 1999.

Anderberg, M. '*Cluster Analysis for Applications*', Academic Press, 1973.

Anderson, A. '*A review of some recent developments in numerical taxonomy*', M.Sc. thesis, University of Aberdeen, 1966.

Arabie, P., Hubert, L. and De Soete, G. '*Clustering and Classification*', World Scientific Publishing Co., pp 5-27, 341-371, 1996.

Ashby, M and Jones, D '*Engineering Materials 1*', Pergamon Press Plc., pp 71-86, 1988.

Baentsch, M., and Molter, G., and Sturm, P., *'Introducing application-level replication and naming into today's web'*, Computer Networks and ISDN Systems, vol. 28, pp. 921-929, 1996.

Bajcsy, P. and Ahuja, N. '*Location- and Density-Based Hierarchical Clustering Using Similarity Analysis*', IEEE Transactions on Pattern

Analysis and Machine Intelligence, Vol. 20, No. 9, pp 1011-1015, September 1998.

Ball, G. and Hall, D. '*A Clustering Technique for summarizing Multivariate Data*', Behavioral Sciences, Vol. 12, pp 153-155, 1967.

Berners-Lee, T., and Cailliau, R., '*World Wide Web proposal for a hypertext project*', CERN European Laboratory for Particle Physics, Geneva CH, 1990.

Berry, W. and Feldman, S. '*Multiple Regression in Practice*', Sage Publications, pp 9-33, 1986.

Bezdek, J. '*Fuzzy Mathematics in Pattern Classification*', Ph.D. Thesis, Applied Mathematical Center, Cornell University, Ithaca, 1973.

Bezdek, J. '*A Convergence Theorem for the Fuzzy ISODATA Clustering Algorithms*', IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. PAMI-2, pp 1-8, 1980.

Bezdek, J. 'Pattern Recognition with Fuzzy Objective Function Algorithms', Plenum Press, New York, 1981.

Bezos, A. '*STEP Technology for Product-Data Representation*', Proceedings of the 14th International CODATA Conference, France, pp 18-22, 1994.

Bieber, M. '*Providing Information Systems with Full Hypermedia Functionality*', Proceedings of the 26th Hawaii International Conference on System Sciences, pp 390-400, 1993.

Bolton, W. '*Engineering Materials Pocket Book*', BH Newnes, pp 22-82, 1989.

Boothroyd, G. and Knight, W. '*Fundamentals of Machining and Machine Tools*', Marcel Dekker Inc., pp 353-363, 1989.

Brachman R. and Anand T., *'The Process of Knowledge Discovery in Databases: A First Sketch'.* Proc. AAAI Workshop Knowledge Discovery in Databases, 1994.

British Steel '*Iron and Steel Specifications*', British Steel Plc., pp 71-188, 1999.

Carpenter, I *'Machinability Assessment and Tool Selection for Milling'*, PhD Thesis, University of Durham, Durham, UK, 1996.

Carpenter, I. '*Knowledge Engineering in Manufacturing*', Seminar in the University of Durham, UK, February-2000.

Carpenter, I., Ritchie, J., Simmons E. and Dewar, R. *'Virtual Manufacturing'*, Manufacturing Engineering, pp 113-116, June 1997.

Carvill, J. '*Mechanical Engineer's Data Handbook*', Butterworth-Heinemann, pp 218-263, 1993.

Casavant T. and Singhal M., *'Distributed Computing Systems'*, IEEE Computer Society Press, pp 6-30,116-132, 1994.

Chang, H., and Lu, F. *'WWW-based collaborative system for integrated design and manufacturing'.* Concurrent Engineering: Research and Applications, Vol. 7, Number 4, pp. 319-334, 1999.

Chan, M., Park, J. and Yu, P. '*Efficient Data Mining for Path Traversal Patterns*', IEEE Transactions on Knowledge and Data Engineering, Vol. 10, No. 2, March/April 1998.

Chen, M., Han, J. and Yu, P. '*Data Mining: An Overview from a Database Perspective*', IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No. 6, December 1996.

Cheung, D., Vincent, T., Fu, A. and Fu, Y. '*Efficient Mining of Association Rules in Distributed Databases*', IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No. 6, December 1996.

Cios, K., Pedrycz, W. and Swiniarski, R. '*Data Mining Methods for Knowledge Discovery*', Kluwer Academic Publishers, pp 1-125, 375-428, 1998.

Conklin, J. '*Hypertext – A Survey and Introduction*', IEEE Computer, Vol 20, No. 9, pp 17-39, 1987.

Cornelius, B. '*Developing Distributed Systems*', IT Service, University of Durham, 1997.

Cornelius, B. '*Using CORBA and JDBC to produce Three Tier Systems*', IT Service, University of Durham, 1998.

Dagli, C. 'Artificial Neural Networks for Intelligent Manufacturing', Chapman & Hall, pp 3-87, 1994.

Davidson, G., Hendrickson, B., Johnson D., Meyers Ch. and Wylie, B. '*Knowledge Mining with VxInsight: Discovery Through Interaction*', Journal of Intelligent Information Systems, No. 11, pp 259-285, 1998.

Denning, D. 'Internet Besieged: Countering Cyberspace Scofflaws', Addison Wesley, pp 1-27, 1998.

Devore, J. and Peck, R. '*Statistics*', West publishing Company, pp 495-536, 1986.

Djoko, S., Cook, D. and Holder, L. '*An Empirical Study of Domain Knowledge and Its Benefits to Substructure Discovery*', IEEE Transactions on Knowledge and Data Engineering, Vol. 9, No. 4, July/August 1997.

Duan, N., *'Distributed database access in a corporate environment using Java'*, Computer Networks and ISDN Systems, vol. 28, pp. 1149-1156, 1996.

Duda, R. and Hart, P. *'Pattern Classification and Scene Analysis'*, Wiley, New York, 1973.

Eichmann, D., McGregor, T. and Danley, D. *'Integrating Structured Databases into the Web: the MORE System'*, Research Institute for Computing and Information Systems, University of Houston – Clear Lake, USA.

URL: http://rbse.jsc.nasa.gov/eichmann/MORE_abstract.html.

El-Sonbaty, Y. and Ismail, M. *'Fuzzy Clustering for Symbolic Data'*. IEEE Transactions on Fuzzy Systems, Vol. 6, No. 2, pp 195-204, 1998.

Everitt, B. *'Cluster Analysis'*, John Wiley & Sons Inc., pp 2-9, 37-50, 55-87, 1993.

Freitas, A. and Lavington, S. *'Mining Very Large Databases with Parallel Processing'*, Kluwer Academic Publishers, pp 7-58, 1998.

Gertley, G. and Magee, B. *'Hypermedia Applied to Manufacturing Environment'*, Proceeding Hypertext, pp 419-424, 1991.

Geva, A. *'Hierarchical Unsupervised Fuzzy Clustering'*. IEEE Transactions on Fuzzy Systems, Vol. 7, No. 6, pp 723-733, 1999.

Giachetti R. 'A Standard Manufacturing information Model to Support Design for Manufacturing in Virtual Enterprises'. Journal of Intelligent Manufacturing, Vol. 10, pp 49-60, 1999.

Gindy N. *'Responsive Manufacturing in UK Aerospace Industry'*. 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Goldman, S. and Nagel, R. *'Management, Technology and Agility: the Emergence of a New Era in Manufacturing'*, Int. Journal Technology Management, Vol. 8, No. 1/2, pp 18-38, 1993.

Gordon, A. *'A Review of Hierarchical Classification'*, J.R. Statist. Soc. A, Part 2, pp 119-137, 1987.

Gordon, A. 'Classification, Methods for the Exploratory Analysis of Multivariate Data', Chapman and Hall, pp 13-49, 121-139, 1981.

Gower, A. 'A General Coefficient of Simmilarity and some of its Properties', Biometrics, Vol. 27, pp 857-874, 1971.

Gustafson, D. and Kessel, W. '*Fuzzy Clustering with a Fuzzy Covariance Matrix*', Proceedings of the IEEE CDC, San Diego, CA, pp 761-766, 1979.

Hamel J. and C. Wainwright, *'Virtual Integration: A Strategy for Agility'*. 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Hamilton G., Cattell R. and Fisher M. *'JDBC Database Access with Java'*, Addison Wesley, pp 33-89, 1997.

Howard C. and Rayward-Smith V. *'Discovering Knowledge from Quality Meteorological Databases'*, in Knowledge Discovery and Data Mining, edited by M.A. Bramer, published by IEE, pp 180-201, 1999.

Hu, X. 'Knowledge Discovery in Databases: An Attribute-Oriented Rough Set Approach', PhD. Thesis, University of Regina, Canada, 1995.

Huang, Z. and Ng, M. *'A Fuzzy k-Modes Algorithm for Clustering Categorical Data'*. IEEE Transactions on Fuzzy Systems, Vol. 7, No. 4, pp 446-452, 1999.

Hunt, D. '*Computer-Integrated Manufacturing Handbook*', Chapman & Hall, pp 130-148, 1989.

Isakowitz, T. '*Hypermedia, Information Systems and Organisations: A Research Agenda*', Proceedings of the 26th Hawaii International Conference on System Sciences, pp 361-369, 1993.

Jain, K. and Dubes, R. *'Algorithms for Clustering Data'*, Englewood Cliffs, Prentice Hall, 1988.

Khoo, L., Tor, S. and Zhai, L. '*A Rough Set-Based Approach for Classification and Rule Induction*', Advanced Manufacturing Technology, Vol. 15, pp 438-444, 1999.

Kimball, R., Reeves, L., Ross, M. and Thornthwaite, W. *'The Data Warehouse, Lifecycle Toolkit'*, John Wiley & Sons Inc., pp 1-91, 1998.

Klecka, W., *'Discriminant Analysis'*, Sage Publications, 1980.

Klir, G. and Folger, T. *'Fuzzy Sets, Uncertainty, and Information'*, Prentice-Hall International Inc., pp 1-32, 239-270, 1988.

Kojima T., H. Sekiguchi, H. Kobayashi, S. Nakahara and S. Ohtani. *'An Expert System of Machining Operation Planning in Internet Environment'.* Proceedings of the 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Küssel R., Liestmann V., Spiess M. and Stich V., *'TeleService, Gadget or Customer-Oriented and Efficient Service'.* Proceedings of the 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Lance, G. and Williams, W. *'A General Theory of Classificatory Sorting Strategies. Hierarchical Systems'*, Computer J., Vol. 9, pp 373-380, 1967.

Lemay, L. and Perkins Ch. *'Java 1.1 in 21 Days'*, Sams.net Publishing, 1997.

Leung, H. *'A Collaborative Manufacturing Environment with the use of Hypermedia'*, Proc. ASME Symposium on Computers in Engineering, Houston, TX, 1995.

Leung, R., Leung, H. and Hill, J. *'Multimedia/Hypermedia in CIM: state-of-the-art Review and Research Implications (part I: state-of-the-art Review)'*, Computer Integrated manufacturing Systems, Vol. 8 No. 4, pp 255-260, 1995.

Leung, R., Leung, H. and Hill, J. *'Multimedia/Hypermedia in CIM: state-of-the-art Review and Research Implications (part II: Research Implications)'*, Computer Integrated manufacturing Systems, Vol. 8 No. 4, pp 261-268, 1995.

Liu, W., White, A., Thompson, S. and Bramer, M. *'Techniques for Dealing with Missing Values in Classification'*, Advances in Intelligent Data Analysis, Springer, Berlin, 1997.

Ma, W. and Chu, K. *'Extracting Geometric Features from a Virtual Environment'*, Proceedings of the 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Mac Queen, J. '*Some Methods for Classification and Analysis of Multivariate Observations*', 5th Symposium on Mathematical Statistics and Probability, Berkeley, CA, Vol. 1, No. AD-669871, pp 281-297, 1967.

McLachlan, G. '*Discriminant Analysis and Statistical Pattern Recognition*', Wiley, 1992.

Michalski, R. and Stepp, R. *'Automated Construction of Classifications: Conceptual Clustering versus Numerical Taxonomy'*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-5, No. 4, pp 396-410, July 1983.

Michalski, R., Bratko I. and Kubat, M. *'Machine Learning and Data Mining'*, John Wiley & Sons Ltd., pp 71-112, 1998.

Milligan, G. *'Clustering Validation: Results and Implications for Applied Analysis'*, World Scientific Publishers, pp 341-371, 1996.

Milligan, G. and Cooper M. *'An Examination of Procedures for Determining the Number of Clusters in a Data Set'*, Psychometrika, pp 159-179, 1985.

Ming, L., Xiaohong Y., Tseng M. and Shuzi, Y. *'A CORBA-Based Agent-Driven Design for Distributed Intelligent Manufacturing Systems'*, Journal of Intelligent Manufacturing, Vol. 9, pp 457-465, 1998.

Mojena, R. *'Hierarchical Grouping Methods and Stopping Rules: an Evaluation'*, Computer J., vol. 20, pp 359-363, 1977.

Monostori, L. and Viharos Z. *'Multipurpose Modelling and Optimisation of Production Processes and Process Chains by Combining Machine Learning and Search Techniques'*, Computer and Automation Research Institute, Budapest, Hungary, 1995.

Nagel R, and Dove A. *'21st Century Manufacturing Enterprise Strategy'*, Volumes 1 and 2, Iacocca Institute, Lehigh University, Bethlehem, PA, 1991.

Ng, F., Ritchie, J., Simmons E. and Dewar, R. *'Tools for Cable Harness Design in Virtual Environments'*, Proceedings of the 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Noaker, M. *'The search for Agile Manufacturing'*, Manufacturing Engineering, 113, (5), 1994.

Noda, A. *'Manufacturing Industries in the Internet Era'*, Proceedings of 14th International Conference on Computer Aided Production Engineering, Tokyo, Japan, September/1998.

Pan, P., Cheng, K. and Harrison, D. *'A Neural-Fuzzy Approach to the Selection of Journal Bearings'*, Proceedings of the Advances in Manufacturing Technology XI, Glasgow, Scotland, 1997.

Pan, P., Cheng, K. and Harrison, D. *'JAVA-Based Systems: An Engineering Approach to the Implementation of Design Agility and Manufacturing Responsiveness'*, Proceedings of the 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Pant, S., and Hsu, C., *'Business on the Web: Strategies and Economics'*, Computer Networks and ISDN systems, vol. 28, pp. 1481-1490, 1996.

Park, H., Tenenbaum, J. and Dove, R. *'Agile Infrastructure for Manufacturing Systems (AIMS) A Vision for Transforming the US Manufacturing Base'*, Technical Report, 1993.

Parsaei, H., and Jamshidi, M., *'Design and implementation of Intelligent Manufacturing Systems'*, Prentice Hall, pp. 81-98, 1995.

Pawlak, Z. *'Rough Sets'*, International Journal of Computer and Information Sciences, 11:341-356, 1982.

Pawlak, Z. *'Why Rough Sets?'*, IEEE International Conference on Fuzzy Systems, Vol. 2, pp 738-743, 1996.

Peng, Q., Hall, F. and Lister, P. *'Using Virtual Reality as a tool to enhance Computer Aided Process Planning'*, Proceedings of the 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Pimentel, K. and Teixeira, K. *'Virtual Reality: Through the New Looking Glass'*, New York: McGraw-Hill, 1994.

Plipovic, M., Stojadinovich, A. and Spasic, Z. *'Virtual Environment for Advanced Manufacturing Automation'*, Proceedings of the 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Regli, W. *'Internet-enabled Computer-Aided Design'*, IEEE Internet Computing, pp 39-50, January-February, 1997.

Rembold, U., Nnaji, B. and Storr, A. *'Computer Integrated Manufacturing and Engineering'*, Addison Wesley, pp 1-25, 371-401, 1993.

Revere, K. *'A Distributed Decision Support System for Turning and Milling using the Internet'*, PhD. Thesis, University of Durham, UK, 2000.

Ruspini, E. *'A New Approach to Clustering'*, Information Contr., Vol. 19, pp 22-32, 1969.

Sastry, L., and Boyd, D. *'Virtual Environments for Engineering Applications'*, Springer-Verlag London Ltd, Virtual Reality, pp 235-244, 1998.

Scott, M. *'Multimedia Road Show Gets Set to Achieve Stardom'*, PC Week Magazine, p 12, 1990.

Shao, J. *'Using Rough Sets for Rough Classification'*, IEEE International Conference on Fuzzy Systems, pp 268-273, 1996.

Shen, W. and Leng, B. *'A Metapattern-Based Automated Discovery Loop for Integrated Data Mining-Unsupervised Learning of Relational Patterns'*, IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No. 6, pp 898-910, December 1996.

Smith, Ch., Wright, P. *'A World Wide Based Design to Fabrication Tool'*, University of California, Berkeley, 1998.

URL: http://kingkong.me.berkeley.edu/~smythe/school/cybercut.html.

Sybase Inc., *'Sybase SQL Anhywhere'*, Sybase Inc., 1997.

Taylor, P. *'Tactile and Kinaesthetic Feedback in Virtual Environments'*, Transactions of the Institute of Measurement and Control, 17 (5), pp 225-233, 1995.

Tian, G., Zhao, Z. and Baines, R. *'Agile Manufacturing Information Based on Computer Network of the World-Wide Web (WWW)'*, Proceedings of the 13th National Conference on Manufacturing, Glasgow Caledonian University, Scotland, UK, September/1997.

Timm, H. and Kruse, R. *'Fuzzy Cluster Analysis with Missing Values'*. IEEE Transactions on Fuzzy Systems, pp 242-246, 1998.

Timings, R. *'Engineering Materials Volume 1'*, Longman Group UK, pp 147-180, 1992.

Velásquez, L., Velásquez, M. and Maropoulos, P. (a) *'Cluster Analysis Applied to Manufacturing Tooling Data'*, Proceedings of the Second International Symposium on Engineering of Intelligent Systems, EIS'2000, University of Paisley, Scotland, UK, June/2000.

Velásquez, L. and Velásquez, M. (b) *'Web-Based Strategies to Support Collaborative Work in the Manufacturing Industry'*, Proceedings of the Second International Conference on Management and Control of Production and Logistics, Grenoble, France, July/2000.

Velásquez, L., Velásquez M., and Maropoulos, P. (c) *'An Internet-Based Tool Selection System for Turning Operations'*, Proceedings of the 16th International Conference on Computer Aided Production Engineering, CAPE 2000, Edinburgh University, Scotland, UK, August/2000.

Velásquez, L. and Velásquez, M. (d) *'An Internet-Based for the Technical Support of Tooling Operations'*, Proceedings of the 20th Computers and

Information in Engineering (CIE) International Conference, University of Maryland, USA, September/2000.

Wahab, D. and Bendiab, A. *'A Framework for Planning, Selecting and Deploying Enabling Technologies to Support Virtual Manufacturing Organisations'*, Proceedings of the 13th National Conference on Manufacturing, Glasgow Caledonian University, Scotland, UK, September/1997.

Welstead, S. *'Neural Network and Fuzzy Logic Applications in C/C++'*, John Wiley & Sons, pp 5-7, 395-408, 1994.

Westphal, Ch. and Blaxton, T. *'Data Mining Solutions'*, John Wiley & Sons, pp 5-24, 377-441, 1998.

Wong, J., Nayar R. and Mikler, A. *'A Framework for a World Wide Web-based Data Mining System'*, Journal of Network and Computer Applications, Vol. 21, pp 163-185, 1998.

Wood, C., Sihra, T. and Harrison, D. *'The Application of Neural Networks to Identify On-line Tool Wear in a Turning Process'*, Proceedings of the Advances in Manufacturing Technology XI, Glasgow, Scotland, 1997.

Wu, B. *'Manufacturing Systems Design and Analysis'*, Chapman & Hall, pp 5-34, 1992.

Xie, X. and Beni, G. *'A Validity measure for Fuzzy-Clustering'*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, pp 841-846, 1991.

Yang, J., and Kaiser, G., *'An architecture for integrating OODB's with WWW'*, Computer Networks and ISDN systems, vol. 28, pp. 1243-1254, 1996.

Yuan H., Cheung H. and Li X., *'The Cellular Manufacturing in the Agile Manufacturing Era'*. Proceedings of the 15th International Conference on

Computer-Aided Production Engineering, University of Durham, UK, 1999.

Zadeh, L. *'Fuzzy Sets'*, Information and Control, No. 8, pp 338-353, 1965.

Zhao J., Cheung W., Young R. and Bell R., *'An Object Oriented Manufacturing Data Model for a Global Enterprise'*. Proceedings of the 15th International Conference on Computer-Aided Production Engineering, University of Durham, UK, 1999.

Zhou, E. and Harrison, D. *'Application of a Fuzzy Neural Hybrid Model to a Machining Process'*, Proceedings of the Advances in Manufacturing Technology XI, Glasgow, Scotland, 1997.

Ziarko, W. and Shan, N. *'Discovering Attribute Relationships, Dependencies and Rules by using Rough Sets'*, Proceedings of the 28th Annual Hawaii International Conference on System Sciences, 1995.