Soft computing for tool life prediction a manufacturing application of neural - fuzzy systems

Emmanouilidis, Christos I.

SOFT COMPUTING FOR TOOL LIFE PREDICTION

A MANUFACTURING APPLICATION OF

NEURAL - FUZZY SYSTEMS

A thesis submitted to the

School of Engineering

University of Durham

for the degree of

Master of Science

by

Christos I. Emmanouilidis

September 1997

ABSTRACT

Tooling technology is recognised as an element of vital importance within the manufacturing industry. Critical tooling decisions related to tool selection, tool life management, optimal determination of cutting conditions and on-line machining process monitoring and control are based on the existence of reliable detailed process models. Among the decisive factors of process planning and control activities, tool wear and tool life considerations hold a dominant role. Yet, both off-line tool life prediction, as well as real time tool wear identification and prediction are still issues open to research. The main reason lies with the large number of factors, influencing tool wear, some of them being of stochastic nature. The inherent variability of workpiece materials, cutting tools and machine characteristics, further increases the uncertainty about the machining optimisation problem.

In machining practice, tool life prediction is based on the availability of data provided from tool manufacturers, machining data handbooks or from the shop floor. This thesis recognises the need for a data-driven, flexible and yet simple approach in predicting tool life. Model building from sample data depends on the availability of a sufficiently rich cutting data set. Flexibility requires a tool-life model with high adaptation capacity. Simplicity calls for a solution with low complexity and easily interpretable by the user.

A neural-fuzzy systems approach is adopted, which meets these targets and predicts tool life for a wide range of turning operations. A literature review has been carried out, covering areas such as tool wear and tool life, neural networks, fuzzy sets theory and neural-fuzzy systems integration. Various sources of tool life data have been examined. It is concluded that a combined use of simulated data from existing tool life models and real life data is the best policy to follow. The neurofuzzy tool life model developed is constructed by employing neural network-like learning algorithms. The trained model stores the learned knowledge in the form of fuzzy IF-THEN rules on its structure, thus featuring desired transparency. Low model complexity is ensured by employing an algorithm which constructs a rule base of reduced size from the available data. In addition,

the flexibility of the developed model is demonstrated by the ease, speed and efficiency of its adaptation on the basis of new tool life data.

The development of the neurofuzzy tool life model is based on the Fuzzy Logic Toolbox (v1.0) of MATLAB (v4.2c1), a dedicated tool which facilitates design and evaluation of fuzzy logic systems. Extensive results are presented, which demonstrate the neurofuzzy model predictive performance. The model can be directly employed within a process planning system, facilitating the optimisation of turning operations. Recommendations are made for further enhancements towards this direction.

To my parents

Man is soft and thirsty like grass,
insatiable like grass, his nerves roots that spread;
when the harvest comes
he would rather have the scythes whistle in some other field;
when the harvest comes
some call out to exorcise the demon
some become entangled in their riches, others deliver speeches.
But what good are exorcisms, riches, speeches
when the living are far away?
Is man ever anything else?
Isn't this that confers life?

Last Stop, George Seferis

from George Seferis, *Collected Poems,* Princeton UP, 1969,

Edited and Translated by Edmund Keeley and Philip Sherrard

ACKNOWLEDGMENTS

I wish to express my thanks to my supervisor Dr. Paul Maropoulos for offering me the opportunity to carry out this research project. I would also like to acknowledge Paul's keen interest which enabled me to work in an exciting area, towards applying computational intelligence ideas in manufacturing.

It is hard to overestimate the guidance and assistance I received from Dr. Jim Bumby. Moreover, the moral support and encouragement Jim has offered me are gratefully appreciated.

I am particularly thankful to Dr. Bubakar Alamin for communicating with me ideas about the tool life problem with great enthusiasm and generosity. Many thanks are also due to several people in the University of Durham and especially to Dr. Mary Kilitziraki, Dr. Ian Carpenter, Mr. Kelvin Revere and Mr. Rob Baker. I would also like to express my sincere thanks to the Principal of the Graduate Society, Dr. M. Richardson for his consideration and help.

I wish also to acknowledge the understanding and support of a number of people at University Sunderland, specifically Dr. John McIntyre, Professor Chris Cox and Professor Peter Smith, which enabled me to complete this thesis.

I feel deeply grateful to the Alexander S. Onassis Foundation in Greece for providing me with support for this research.

I am always in dept of gratitude to my parents in Greece, Ioannis and Antiope and also to my sister Vicky. Finally, I wish to thank Anna for her patience and constant support throughout the period of this study.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

1.1 Neurofuzzy and soft computing

Within the last few years there is a growing number of engineering applications of **artificial neural networks** and **fuzzy logic**, ranging from consumer products, industrial decision support and control systems, to financial trading and forecasting. Neural networks and fuzzy logic are, in fact, computational metaphors for the human brain architecture, learning capacity and ability to perform approximate reasoning based on imprecise or incomplete information. Terms such as **soft computing** [Zadeh 1994] or **computational intelligence** [Bezdek 1994] have been used in the past to mark the distinct features shared by neural networks, fuzzy logic systems and some advanced gradient-free probabilistic optimisation techniques, such as evolutionary strategies, genetic algorithms and simulated annealing. These terms also aim at defining a different computational approach than those that "hard computing" or"artificial intelligence" adopt. In traditional, hard computing, precision, certainty and rigour prevail, whereas in soft computing an allowance is made for imprecision and uncertainty. Within the context of computational intelligence, fuzzy logic is primarily concerned with imprecision, neural networks with learning and probabilistic reasoning with uncertainty (Figure 1.1) [Zadeh 1994].

Some of the most important characteristics of computational intelligence are [Jang et al. 1997]:

- **Model-free learning**: Neural networks and adaptive fuzzy inference systems posses the capability of building models based on sample data. A-priori knowledge relevant to the unknown system can ease the modelling procedure, without being a prerequisite.

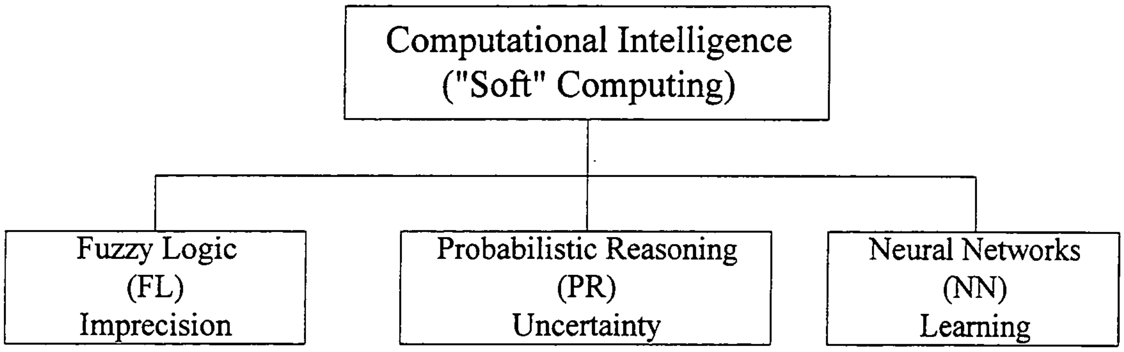

Figure 1.1 Attributes of computational intelligence approaches

- **Human expertise**: Human knowledge can be incorporated into computational intelligence systems in the form of fuzzy IF-THEN rules, as well as in conventional knowledge representations, to facilitate practical problem solving.

- **Numerical computation**: Soft computing allows for numerical processing of information, whereas traditional artificial intelligence (AI) techniques are limited on symbolic information processing .

- **New optimisation techniques**: Computational intelligence can employ innovative optimisation methods, such as genetic algorithms and evolutionary strategies (inspired by the evolution and selection processes), or simulated annealing (inspired by thermodynamics), random search methods etc. Since these methods do not require the gradient of a vector, they are more flexible in complex optimisation tasks.

- **Intensive computation**: Deriving rules or regularity in data sets usually involves intensive computation in most soft computing systems. However, no significant background knowledge concerning the problem to be solved is required.

- **New application domains**: Soft computing has found successful applications in computationally intensive areas, such as adaptive signal processing, nonlinear system identification and control, pattern recognition, nonlinear regression and others.

- **Fault tolerance**: Neurofuzzy systems encode information in a parallel, distributed or redundant manner. Therefore they are not particularly sensitive to faults such as the deletion of a neuron in a neural network or of a fuzzy rule in a fuzzy inference system.

- **Goal driven**: Neurofuzzy and soft computing systems are goal-driven, thus the path from a current state to the solution does not really matter, especially when derivative-free optimisation techniques are used. A-priori knowledge, when available, can be built into them to facilitate the search or optimisation task.

- **Real world applications**: Most real world applications are inherently complex, nonlinear and time varying with built-in uncertainties. Hence, detailed description of the problem is rarely available, precluding the use of conventional problem-solving approaches. Computational intelligence techniques are well suited to solving particularly these kind of problems, providing satisfactory engineering solutions to real-world problems.

Methodology	**Strength**
Neural Networks	Learning and adaptation
Fuzzy Sets Theory	Knowledge representation via fuzzy if-then rules
Probabilistic Reasoning	Systematic random search
Conventional AI	Symbolic manipulation

Table 1.1: Strengths of soft computing and traditional artificial intelligence [Jang et al. 1997]

The main strengths of soft computing and traditional artificial intelligence are summarised in table 1.1 [Jang et al. 1997]. Soft computing approaches have overlapping application domains and they should be considered as complementary rather than competitive methodologies. It can be argued, however, that neural networks are usually more suitable in processing low abstraction level information, such as sensorial data, while fuzzy logic offers a model of reasoning at a rather higher abstraction level. Probabilistic reasoning, on the other hand, is suitable for efficient search in large parameter spaces, when applied to complex optimisation problems. Neurofuzzy and soft computing aims at taking full advantage of individual merits of each one approach by integrating them into a single framework.

1.2 Computational intelligence applications in manufacturing

The thriving emergence of computational intelligence applications in various engineering domains has also had its impact on manufacturing. The range of the problems met in today's manufacturing systems is probably wider than in any other system of human activity. Flexibility has become recognised as the key to success in pursuing continuous competitiveness. There is no unique measure of flexibility for all manufacturing activities. What is common in every decision level is the need to respond promptly to a changing manufacturing environment. The information abstraction level of each manufacturing activity world model varies. Higher layers involve more abstract, less time critical information flows, whereas time considerations become crucial at lower organisational layers, together with a need for data-rich and more precise information. Yet, there is a common need for adopting a flexible world model at each level, equipped with some sort of adaptation capacity. At the very heart of that capacity is the concept of learning.

In manufacturing enterprises the need for learning tends to pertain to all production activities. This applies both to human workforce and production machines activities. As life-long learning is required more and more in the human factor, the same - up to a certain extent - applies to manufacturing equipment. The notion of learning is tightly coupled with the existence of intelligence. The nature of the intelligence needed varies accordingly with the hierarchical level that each manufacturing activity belongs to. Higher organisation layers require more conceptual learning with little concern for precision. Uncertainty handling becomes more crucial at lower, execution level activities. Traditional artificial intelligence exhibits high levels of conceptual, symbolic information processing capabilities, with good representation and explanation power. On the other hand, computational intelligence approaches, such as neural networks or neurofuzzy inference systems are well suited to handle more detailed information at lower abstraction levels.

An extensive survey of modern machine learning applications in manufacturing has been reported, as a result of the activities of the *CIRP Working Group on Applications of Artificial Intelligence in Manufacturing Engineering* [Monostori et al. 1996]. A summary of the key characteristics of various learning approaches, as compiled in that survey, is shown in Figure 1.2. Neural networks and neurofuzzy methods are compared against traditional artificial intelligence methods such as symbolic classification, explanation based reasoning and discovery/analogy methods. Statistical pattern recognition is also included, considering its rather overlapping application domain with some neural network-based of neurofuzzy methods.

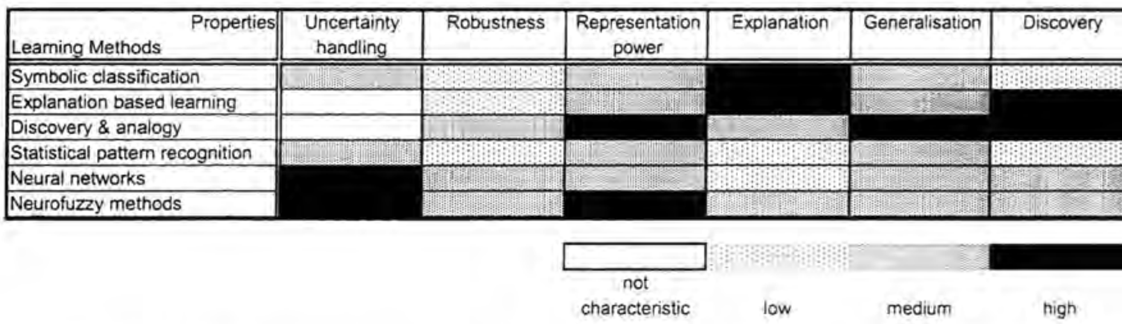

Figure 1.2: Main attributes of learning approaches [Monostori et. al 1996]

The requirements of various manufacturing activities in intelligence properties are illustrated in Figure 1.3.

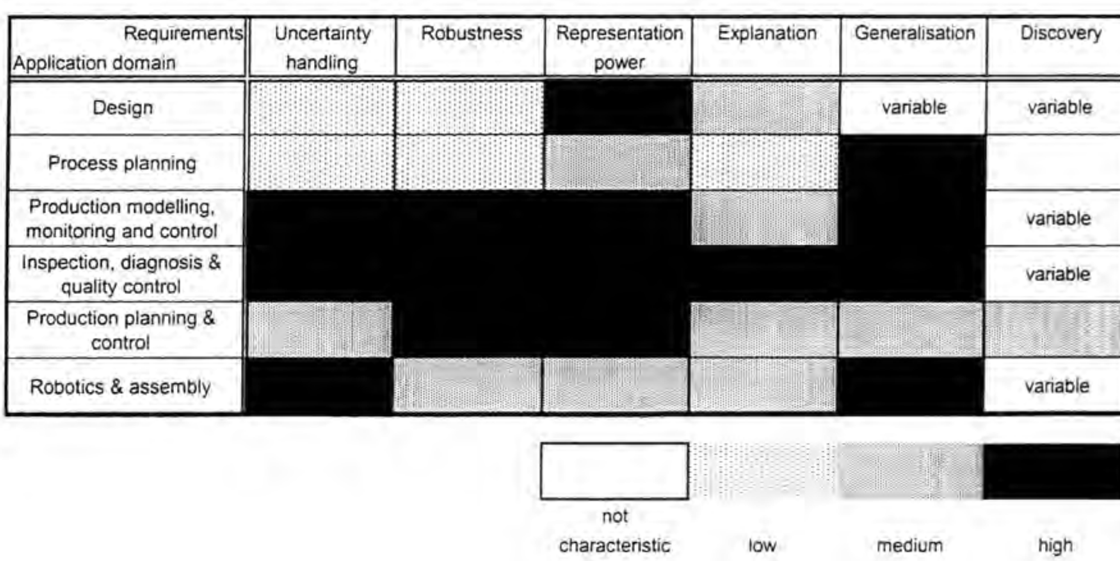

Figure 1.3: Requirements of manufacturing application domains [Monostori et. al 1996]

The information contained in these two figures is compiled into Figure 1.4, where the applicability of each learning approach to different application domains is illustrated. Evidently, soft computing techniques appear to be relevant to all manufacturing decision

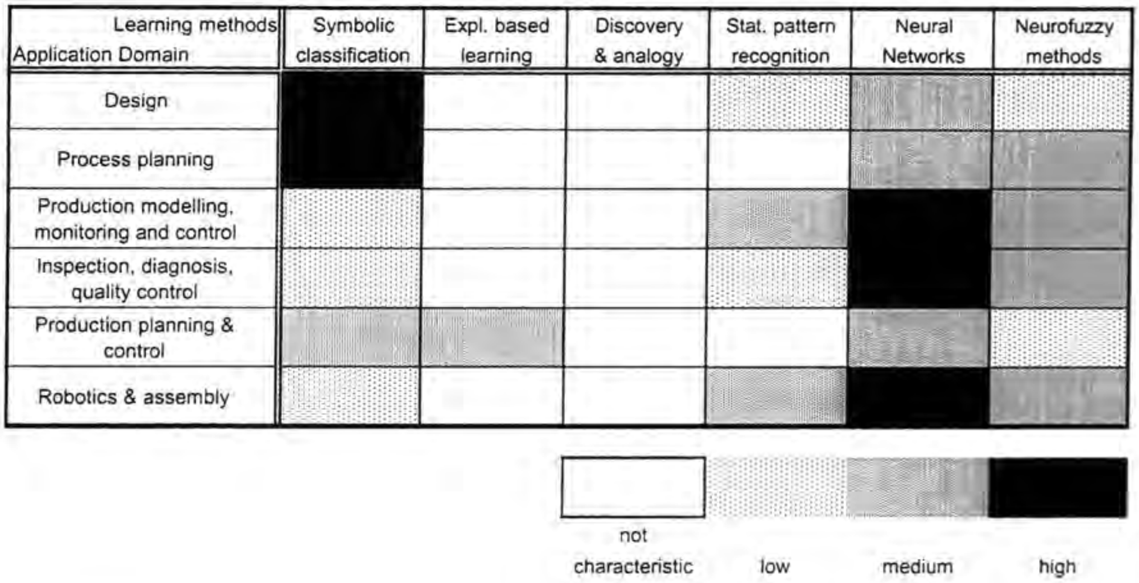

Figure 1.4: Applicability of machine learning approaches [Monostori et. al 1996]

Although a detailed treatment of the applicability of computational intelligence methods to all aspects of manufacturing decision making is beyond the scope of the present work, it should be noted that there have already been reported numerous such applications. These include [Monostori and Barschdorf 1992, Udo and Gupta 1994, Huang and Zhang 1994, Dagli 1994, Shin et al. 1992, Huang and Zhang 1995, Monostori et al. 1996]:

- design (feature extraction and recognition, design retrieval)
- group technology (part grouping aiming at cell configuration design, part feature recognition) and capacity constraints problems, (resource allocation and constraints satisfaction)
- information management (database management, content-based information retrieval)
- process planning (operations selection and sequencing, process modelling, tool selection, machining process parameter selection, machining feature recognition and processing, attribute selection and coding)
- shop floor scheduling
- real-time process modelling, monitoring and control

- process planning (operations selection and sequencing, process modelling, tool selection, machining process parameter selection, machining feature recognition and processing, attribute selection and coding,)
- shop floor scheduling
- real-time process modelling, monitoring and control
- production activity simulation
- quality assurance
- robotics and assembly (robot world model acquisition, planning and control)

It should be noted that not all the above applications have reached a maturity level to be considered of industrial relevance. Indeed, lower hierarchical level applications such as process modelling, monitoring and control, quality assurance, robot planning and control are more well developed, while the upper level applications are still very much specialised and substantial effort is still needed for the development of more generic computational intelligence methods.

1.3 Tool wear and tool life

One of the areas wherein soft computing is considered as a promising problem solving approach is detailed process modelling. In contrast with high level models which are relevant to product development phases, detailed process modelling is essential for planning and execution of specific operations. It can be distinguished into predictive (off-line) modelling for process planning and optimisation and real-time (on-line) for process monitoring and control [Maropoulos 1995]. This work examines the application of neurofuzzy methods for an important machining process modelling problem, namely that of tool life prediction for turning operations. Tool wear and tool life considerations are of vital importance for a range of critical tooling decision making activities, such as tool selection [Maropoulos and Hinduja 1990, Maropoulos and Hinduja 1991, Maropoulos 1992, Maropoulos and Gill 1995], tool life management [Maropoulos and Alamin 1996, Alamin 1996], optimal determination of cutting conditions and process monitoring and control [Ulsoy and Koren 1993].

Reliable off-line tool life prediction is a very formidable task, due to the high degree of uncertainty about how wear progresses at different tool faces and for various combinations of the parameters, which give rise to variations in the way that tool wear evolves. On the other hand, in-process or real-time tool life prediction, i.e. prediction based on information about the actual wear status, usually suffers from the increased cost, the complexity of the required signal processing and the low reliability of the instrumentation involved. The tool life prediction approaches reported in the literature are:

- Use of tool life data provided by tool manufacturers and machining data handbooks
- Deterministic empirical modelling, based on adopting some simple tool life formulae
- Analytical tool life modelling based on cutting process theory
- Probabilistic modelling of (usually) the first one or two statistical moments of tool life
- Probabilistic interpretation of empirical or analytical tool life models
- Tool wear evolution modelling based on the theory of random processes, such as reliability based tool life modelling, or other Markovian process modelling
- Chaotic models of tool wear process
- Alternative approaches to tool life modelling such as neural networks or models based on fuzzy sets theory.

Among the above approaches to tool life prediction only those based on tabled machining data from tool manufacturers and handbooks, empirical tool life modelling and to a lesser extent probabilistic modelling or probabilistic interpretation of empirical or analytical tool life models are currently reported to be of industrial relevance. All the above modelling methods heavily depend on data availability in order to define values for their free parameters for a range of operations. Machining data can be derived from tool manufacturers' catalogues, machining data handbooks, laboratory experiments and shop floor operations. Cutting data found in catalogues and handbooks usually tends to be very conservative to ensure stable operation, while laboratory tests often significantly deviate from the machining results obtained at the shop floor.

Shop floor data contains rich information about the performance of the cutting process for the specific combinations of operation, machine tool, workpiece material and cutting tool. Yet such data are often discarded in machining practice and no advantage is taken from the information they contain. Provision for utilising such approved operations data is a key feature in several prototype systems recently developed at Durham University, including Computer Aided Process Planning systems for turning [Maropoulos 1992, Maropoulos and Gill 1995, Maropoulos and Alamin 1995], Tool Life Prediction and Management system for turning [Alamin 1996, Maropoulos and Alamin 1996] and Machinability Assessment and Tool Selection for milling [Carpenter 1996].

Due to the significant tool life variability observed in machining practice, deterministic interpretation of tool life formulae is inadequate to handle the existing uncertainty about the cutting process. Probabilistic methods work well as long as their statistical assumptions are valid. However tool life variability can not always be described by simple tool life distributions, such as the normal distribution. Adopting more complex and flexible distributions inevitably increases the complexity of the computation involved for the determination of the distribution free parameters from available data. Reliability approaches and those methods based on the theory of random or chaotic processes are even more complicated and they can only achieve improved precision at the cost of significantly increased complexity.

Admittedly, there is still a need for improved and yet simple and data-driven tool life models which overcome some of the shortcomings that existing models present. Neurofuzzy methods offer an attractive alternative of handling uncertainty with reduced complexity. Such a neurofuzzy model is developed in this thesis for tool life prediction in turning operations. The model is non-parametric in the statistical sense, i.e. it does not rely on any statistical assumption about the underlying probability distribution function of tool life. It is also flexible enough and can be easily adapted to capture modelling discrepancies, based on available tool life cutting data.

1.4 Research objectives

The main objectives of the research work described in the present thesis are:

- To study the main requirements for the specification and development of a reliable neurofuzzy tool life model.
- To obtain a neurofuzzy tool life model of reduced complexity.
- To provide a single generic model for a wide range of combinations of cutting conditions, type of cut, cutting tool and workpiece material, which can facilitate machining optimisation.
- To build a complete data-driven model, i.e. to derive a neurofuzzy tool life model whose structure and parameters can be completely defined based on available cutting data.
- To incorporate a priori knowledge about the cutting process by accommodating results of tool life modelling research recently carried out at Durham University [Alamin 1996].
- To achieve flexible tool life modelling, which is easily adaptable on the basis of new tool life data.

The computational platform on which the present research work was carried out was a 166MHz Pentium computer with 32MB of RAM, equipped with the Fuzzy Logic Toolbox (v 1.0) of the MATLAB (v. 4.2c1) software package for numerical computation and visualisation (The Mathworks Inc.).

1.5 Thesis structure

The thesis comprises eight chapters whose content, apart from the introduction, can be summarised as follows:

Chapter 2 introduces some basic concepts of artificial neural networks with a focus on two specific types of feedforward layered networks, namely the multi-layer perceptron and the radial basis function network, whose learning algorithms are relevant to the training of the neurofuzzy model developed.

Fundamental definitions of fuzzy sets theory are provided in Chapter 3. A brief description of the key constituents of a basic fuzzy logic system is given. This introduction to fuzzy logic is the necessary background required for the description of neurofuzzy inference systems presented in Chapter 4. The motivation and the benefits for obtaining neurofuzzy representations are briefly discussed, followed by a more detailed description of the functionality of the specific neurofuzzy structure employed in the present work, i.e. that of adaptive, network-based fuzzy inference system (ANFIS) [Jang 1993]. This model possess the capacity of learning from examples via neural network-like learning algorithms.

Chapter 5 provides with a basic background on tool wear and tool life. The importance of tool wear considerations for machining process optimisation is noted and the main parameters influencing tool wear are mentioned. The tool life modelling problem is then discussed. In recognition of the fact that little commercial interest has so far been shown to real-time tool wear identification and tool life prediction, this work focuses on off-line tool life prediction methods, and a review of the main adopted approaches is presented.

In Chapter 6 the main requirements for the development of a neurofuzzy tool life model are studied. The reasons for selecting the ANFIS model are explained and the data availability issue is discussed. The ANFIS structure identification procedure is then described. The representation power of neurofuzzy systems is also demonstrated to facilitate the model building for the tool life prediction problem.

Chapter 7 describes the ANFIS training procedure and explains problems met and how they were overcome. A series of indicative results are then presented which validate the developed model and confirm its adequacy for tackling the off-line tool life prediction problem. Finally, the flexibility of the model is demonstrated, which is of particular importance for its successful application on the shop floor.

The thesis concludes in Chapter 8 with a synopsis of the main results obtained, and conclusions drawn from the work carried out, including suggestions for further research.

NEURAL NETWORKS

2.1 Introduction

The development of artificial neural networks has been motivated in recognition of the powerful information processing capabilities of the human brain. The speed of response of human brain cells to stimuli is five to six orders of magnitude slower than of hard silicon implementations of logic gates [Jain et al. 1996]. However, the efficiency of the brain in performing tasks such as perception and pattern recognition is unreachable by even the fastest available digital computer. The main characteristics of human brain cells activity, which are up to a certain extent present in artificial neural networks, are [Hertz et al. 1991, Haykin 1994, Jain et al. 1996]:

- **Nonlinearity**: Neurons are nonlinear elements, highly interconnected. The whole neural network exhibits complex nonlinear behaviour as a result of the interaction between the neurons.
- **Massive parallelism**: Computational power is dramatically increased due to the massively parallel neurons interconnections.
- **Learning ability**: The adaptive nature of neural networks is a particularly important characteristic. Neural networks perform learning by examples instead of traditional "programming".
- **Generalisation ability**: Neural networks posses the capacity of generalising a solution, given sufficiently rich learning examples. Therefore, they extract knowledge from examples instead of performing simple pattern matching.

- **Inherent contextual information processing**: Knowledge is globally represented in the whole structure and neuronal activity of the network. Therefore, artificial neural networks exhibit a natural contextual information processing capability.
- **Robustness and fault tolerance**: Neural networks can be robust in dealing with corrupted, missing or inconsistent information. The deletion or damage of a neuron has a minimum effect to the overall network performance, since knowledge representation is distributed over the whole network structure.
- **Energy efficiency (VLSI implementability)**: Due to their massively parallel structure, neural networks are convenient structures for hardware VLSI implementation, which can achieve high computational performance with very low energy consumption.
- **Data fusion ability**: Artificial neural networks can receive and process multiple inputs and establish complex interrelationships between them in a distributed way throughout their structure.

Artificial neural networks have found numerous engineering applications. They can be effectively employed in tasks such as pattern recognition, clustering, function approximation, prediction, optimisation, control and they can serve as content-addressable or associative memories. The following paragraphs briefly review the basic concepts of neural networks, including architectural and learning issues, with a focus on multilayered feed forward networks. In particular, some emphasis will be given to multilayer perceptrons and radial basis functions networks whose learning capabilities and structure are relevant to the neurofuzzy modelling approaches that are employed in this work.

2.2 General concepts

Artificial neural networks are biologically inspired structures of computational elements, called **neurons**, nodes, network units or simply units. Each one of these nodes is a simplified model of the human brain cells. Neurons are interconnected with **synapses** and each synapse is assigned a strength or weighting factor, called **synaptic weight** or simply weight. Each neuron receives at its input signals from other neurons. The functional behaviour of the neuron depends on the neuronal model employed. Generally, if **X** is the input vector and **W** is the corresponding synaptic weight matrix, the neuron output is:

$$V = g\,[\,\mathbf{W} \circledcirc \mathbf{X}\,] \tag{2-1}$$

where © is a confluence operator, providing a measure of similarity between **W** and **X**, and g(•) the activation function of the neuron. There are two basic categories of confluence operations, the inner product of **W** and **X** and the Euclidean distance measure between **W** and **X**. The most common activation functions employed are linear, piecewise linear, hard limiter, sigmoid (unipolar, bipolar, multimode) and Gaussian. A basic taxonomy of the existing artificial neural network architectures distinguishes three broad categories, namely **feedforward, recurrent** and **modular** neural networks. Feedforward neural networks are hierarchically ordered structures without any connections directed from a higher in hierarchy node towards a node of a lower layer. Recurrent networks are those that allow for at least on feedback loop to exist between nodes of different layers. Complex architectures consisting of several different neural structures acting in synergy are referred to as modular neural networks. Hybrid architectures which amalgamate neural networks with other intelligent information processing approaches such as traditional artificial intelligence (AI) and fuzzy logic have also been developed. A more detailed architectural taxonomy of neural networks can be found in [Gupta and Rao 1994, Hassoun 1995, Jain et al. 1996, Mehrota et al. 1997].

The learning paradigms involved in neural network training are **error-based learning (supervised, reinforcement)** and **output-based learning (unsupervised)**. Within this broad categorisation, a plethora of learning rules for neural network training has been suggested. A thorough compilation of such rules can be found in [Hassoun 1995]. Of particular interest for the present work are the multi-layered feed forward structures and the learning algorithms which are applicable to them. These algorithms can be properly modified to apply also to the neurofuzzy models which are examined in later chapters.

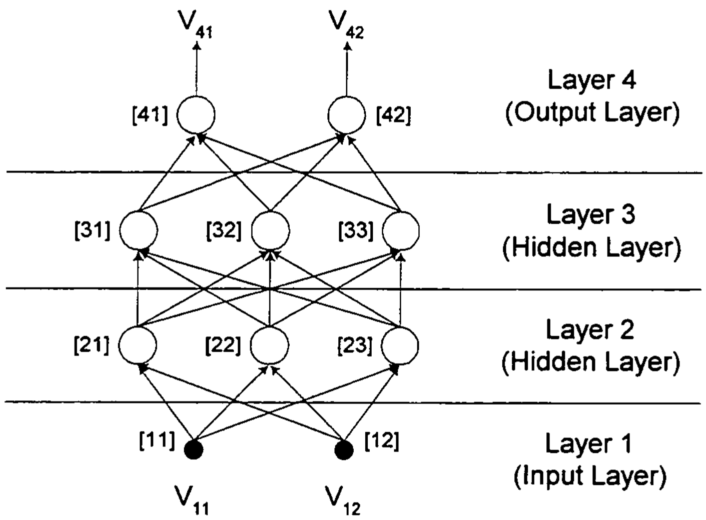

Figure 2.1: Feedforward neural network with two hidden layers

2.3 Feed forward neural networks

Feed forward neural networks have multilayer hierarchical structure. The first layer is the input layer, while the last one is the output layer. Any other layer is called hidden layer. The static feedforward neural networks (Fig. 1) either in the form of multilayer perceptrons with sigmoid activations, or of radial basis functions networks are particularly popular architectures. Any arbitrary continuous function on a compact set may be approximated to any desired degree of accuracy by a multilayer perceptron with sigmoid activations or by a radial basis functions network, i.e. these networks behave as universal approximators. Detailed analysis of the approximation capabilities of neural networks can be found in [Jin et al. 1996, Chen and Chen 1995]. Any lack of success in the application of such a neural network for function approximation, must arise from inadequate learning, insufficient number of hidden units (units that have no direct connections with network inputs or outputs), or lack of deterministic relationship between the input and the target. Different neural network architectures result in different learning difficulties. Therefore, the choice of an appropriate approximation structure will ultimately determine the success of an application. From this point of view, a successful neural approximation procedure may be divided into the following steps [Jin et al. 1996] :

- Determination of the universal approximation structure of the neural network, that is to ensure the inherent approximation capabilities of the neural network through adjusting the number of hidden units and layers.
- Choice of an adequate weight learning algorithm.
- Selection of learning signals that contain sufficient information.

2.3.1 Multilayer perceptrons (backpropagation networks)

These are by far the most popular neural architectures. Backpropagation networks neuronal activity involves inner product confluence operation followed by a sigmoid activation function. The main algorithm employed for their training is the backpropagation, which is a gradient descent stochastic approximation algorithm, but random optimisation techniques can also be employed. The backpropagation algorithm is characterised by relative computational simplicity and convenience for parallel implementation, but quite often is a long and tedious procedure, prone to stuck in local minima. Fortunately, there have already been proposed numerous variations of the basic backpropagation algorithm which improve its convergence. A brief description of the standard backpropagation algorithm follows.

The output of the i-th neuron of the m-th layer (i=1,2,..., n_m and m=1,2,...,M, where n_m is the number of nodes in the m-th layer and M is the number of layers) is:

$$V_{mi} = g\left(h_{mi}\right) = g\left(\sum_{j=1}^{n_m-1} W_{(m-1)j}^{mi} \cdot V_{(m-1)j}\right), \; m = 2,\ldots,M \tag{2-2}$$

where, V_{mi}, h_{mi} is the output and the net input of the i-th neuron of the m-th layer respectively, $W_{(m-1)j}^{mi}$ is the synaptic weight connecting the j-th neuron of the (m-1)-th layer with the i-th neuron of the m-th layer. When a pattern ξ^{μ} is applied to the input layer, i.e. $V_{1r} = \xi_r^{\mu}$, r=1..n_1, the signal is propagated forward towards the output layer until all the network outputs have been calculated. The aggregated error of the network over the whole set of patterns (i.e., the cost function to be minimised) is:

$$E = \frac{1}{2} \sum_{\mu} \sum_{i=1}^{n_M} \left(\zeta_i^{\mu} - V_{Mi} \right)^2 \tag{2-3}$$

where ζ_i^{μ} is the desired output of the i-th output neuron corresponding to the μ-th pattern and the notation $\sum_{\mu}$ denotes aggregation over the whole set of patterns. This is a continuous differentiable function of every weight, so by applying gradient descent the weight update rule can be derived. For the (M-1)-th to M-th layer connections, the weight updates $\Delta W_{(M-1)j}^{Mi}$, which decrease the cost function, are on the negative direction of the cost function gradient with respect to the weight $W_{(M-1)j}^{Mi}$:

$$\Delta W_{(M-1)j}^{Mi} = -\eta \cdot \frac{\partial E}{\partial W_{(M-1)j}^{Mi}} = -\eta \frac{\partial}{\partial W_{(M-1)j}^{Mi}} \left\{ \frac{1}{2} \sum_{\mu} \sum_{i=1}^{n_M} \left[\zeta_i^{\mu} - g\left(\sum_{j=1}^{n_{M-1}} W_{(M-1)j}^{Mi} \cdot V_{(M-1)j} \right) \right] \right\} \Rightarrow \text{(2-4)}$$

$$\Delta W_{(M-1)j}^{Mi} = \eta \cdot \sum_{\mu} (\zeta_i^{\mu} - V_{Mi}) \cdot g'(h_{Mi}) \cdot V_{(M-1)j} = \eta \cdot \sum_{\mu} \delta_{Mi}^{\mu} \cdot V_{(M-1)j} \tag{2-5}$$

where the "delta" at each output node is defined as the product of the derivative of the activation function by the output error, i.e.:

$$\delta_{Mi}^{\mu} = g'(h_{Mi}) \cdot \left[\zeta_i^{\mu} - V_{Mi} \right] \tag{2-6}$$

For the lower layer's synapses differentiation should be performed with respect to weights which are more deeply embedded in (2-3). Thus, :

$$\Delta W_{(k-1)r}^{kj} = \eta \sum_{\mu} \delta_{kj} \cdot V_{(k-1)r} \tag{2-7}$$

where the hidden layers deltas are defined as:

$$\delta_{kj} = g'\left(h_{kj}\right) \cdot \sum_{i=1}^{n_{k+1}} W_{kj}^{(k+1)i} \cdot \delta_{(k+1)i} \,,\; r = 1,..,n_{k-1} \,,\; j = 1,..,n_{k+1} \tag{2-8}$$

Equation (2-7) enables the determination the δ for a given hidden unit in terms of the δ's of the units that it feeds, i.e. by propagating the δ's backwards; hence the name error-back propagation or simply backropagation.

Several modifications of the basic backpropagation algorithm have been proposed that can improve its convergence, including the presence of a momentum term in (2-8), the delta-bar-delta rule and learning-rate adaptation through training. The latter can be achieved by the adoption of some simple heuristics to control the learning rate. In fact these heuristics can easily be coded into a fuzzy representation, resulting in a fuzzy control approach for the backpropagation algorithm [Haykin 1994]. Other techniques are aiming at simultaneously optimising both the parameters and the structure of the neural network. These are network unit-allocating or network prunning techniques and include the cascade correlation learning architecture (CASCOR) and weight decay or elimination algorithms. Other supervised learning algorithms for multilayer perceptrons include the quickprop algorithm, the extended Kalman type of backpropagation learning, the conjugate-gradient method, Newton's method etc. [Haykin 1994, Hassoun 1995, Mehrotra et al. 1997].

2.3.3 Radial basis function networks

In some problems it is useful to combine unsupervised and supervised learning. The most common idea is to have one layer that learns in an unsupervised way, followed by one or more layers trained under supervision. The problem could be dealt within a purely supervised way. However, backpropagation is often extremely slow and it is often preferable to speed up learning considerably by training some layers in an unsupervised way. This works well in problems where similar input vectors produce similar outputs. Then it would be sensible first to categorise the inputs into clusters using competitive learning and use only the category information for supervised learning.

The radial basis function networks (RBF) are hybrid neural architectures, consisting of an input layer, a hidden layer of nonlinear nodes and an output layer of linear nodes. The output of a hidden unit is of the following form:

$$g(\xi) = g\left(\left\|\xi - \mu_j\right\|\right), j=1..n \tag{2-9}$$

where n is the number of hidden units. Equation (2-9) is a radially symmetric scalar function with μ_j at its center. The most common choice for the $\left\|\xi - \mu_j\right\|$ norm is the Euclidean distance. The input/output mapping performed by a radial basis function network is:

$$V_i = \sum_{j=1}^{n} W_j^i \cdot g\left(\left\|\xi - \mu_j\right\|\right) \tag{2-10}$$

where V_i is the *i*-th output node and W_j^i is the weight of the synaptic connection linking the *j*-th node of the hidden layer to the *i*-th output node and *n* is the number of hidden nodes. Possible choices of radial basis functions include multi-quadratic, inverse multi quadratic, Cauchy and spline functions. However, the most commonly employed radial basis function is the Gaussian kernel. The input/output mapping of the jth RBF hidden unit with Gaussian activation is:

$$g_j(\xi) = \exp\left[\frac{\left(\xi - \mu_j\right)^2}{2\sigma_j^2}\right], \tag{2-11}$$

Each hidden unit has its own receptive field in the input space, a region centred on μ_j with size proportional to σ_j. The output layer applies a linear transformation of the hidden output signal. There is a variety of training algorithms for RBF networks [Haykin 1994]. The basic one employs a two-step learning strategy, or hybrid learning. It estimates kernel positions and kernel widths using an unsupervised clustering algorithm, followed by a supervised least mean square (LMS) algorithm to determine the synaptic weights between the hidden and the output layer. After initial training is performed, a supervised gradient-based algorithm can be used to refine the network parameters. This hybrid learning algorithm converges much faster than the backpropagation algorithm for multilayer perceptrons at the expense of increased number of necessary hidden units to achieve the desired mapping. A particularly interesting case of RBF networks are those employing normalisation of the hidden units:

$$g\ (\xi)=\frac{g\left(\left\|\xi-\mu_j\right\|\right)}{\sum_j g\left(\left\|\xi-\mu_j\right\|\right)},\ \mathrm{j}=1,\ldots,n \tag{2-10}$$

The above class of neural networks have proved to be of some equivalence to certain neurofuzzy representations, as it will be discussed in chapter 4. In particular, it has been proven that RBF networks can be considered as a special case of a major class of neurofuzzy inference systems. The importance of this equivalence lies with the potential of employing the powerful algorithms, which are available for neural networks training, to adapt the parameters of neurofuzzy systems. The next chapter summarises general concepts of fuzzy logic systems, in order to provide the necessary background for introducing neurofuzzy approaches in chapter 4.

FUZZY SETS THEORY AND FUZZY SYSTEMS

3.1 Introduction

In most engineering problems both numerical and linguistic information has to be taken into account. Artificial neural networks are powerful computational intelligence tools, particularly suitable for numerical data processing. However, numerical quantities evidently suffer from lack of representation power. Fuzzy logic systems can simultaneously handle numerical data as well as linguistic information. Fuzzy sets theory was first introduced by Zadeh as a mathematical framework for handling uncertainty and imprecision, inherently present in the way natural language describes objects [Zadeh 1965]. The rationale for fuzzy sets theory is that precision and certainty in computation carry a cost and therefore allowance should be made for exploiting the tolerance for imprecision and uncertainty, wherever possible [Zadeh 1994]. In what is referred to as the **principle of incompatibility**, Zadeh argues that "... *as the complexity of a system increases, our ability to make precise and yet significant statements about its behaviour diminishes until a threshold is reached beyond which precision and significance (or relevance) become almost mutually exclusive characteristics*" [Zadeh 1973].

Three types of uncertainty can be distinguished, namely **nonspecificity** (**imprecision**), which deals with sizes (cardinalities) of sets of alternatives, **fuzziness** (or vagueness), which results from unsharp boundaries between fuzzy classes of objects and **strife** (or discord), which expresses conflicts between different sets of alternatives [Klir and Yuan 1995]. Strife and nonspecificity are both related to **ambiguity** as per the choice of an object amongst different alternatives. It arises from the lack of specific distinctions characterising an object or by conflicts between existing distinctions. On the other hand, fuzziness arises from the lack of sharp distinctions between objects. Among the various mathematical theories dealing with information uncertainty, **classical set theory** deals with nonspecificity, **probability** with strife, **possibility** and **evidence theory** with both nonspecificity and strife, while fuzziness is dealt with **fuzzy sets theory** or an extension of evidence theory, referred to **as fuzzified evidence theory** [Klir and Yuan 1995].

Fuzzy sets theory and fuzzy logic had been a controversial issue and had received fierce criticism for a long time before becoming accepted by a significant proportion of the scientific and, indeed, of the industrial community. The main point of departure for questioning the scientific grounds behind fuzzy sets theory is that statistics may sufficiently describe uncertainty and that, in fact, non-statistical uncertainty does not exist. In one of the most assertive statements about the invalidity of fuzzy sets theory in handling uncertainty, Lindley states that "*... probability is the only sensible description of uncertainty and is adequate for all problems involving uncertainty. All other methods are inadequate.*" [Lindley 1987]. However, the probability monopoly has been questioned by several scientists and recently a compilation of papers on the probability versus fuzziness dilemma appeared in a special issue of the *IEEE Transactions on Fuzzy Systems* (February 1994 issue). The opposite extreme viewpoint has also been adopted by some authors, that in fact it is probability that is not a theoretical primitive of mathematics [Kosko 1992]. The *Technometrics* journal (August 1995 issue) has also hosted a relevant debate.

From an engineering perspective, all the above approaches for handling uncertainty offer valid problem solving frameworks as long as they remain applicable to real world problems [Mendel 1995]. The recent emergence of a significant number of commercial products with increased "*Machine Intelligence Quotient* (MIQ)" [Zadeh 1994] have constituted a breakthrough in the industrial acceptance of fuzzy engineering. Industrial applications of fuzzy logic are now so diverse that comprise areas such as [Yen et al. 1995, Hirota and Sugeno 1995, Marks 1994]:

- consumer products (e.g., cameras, photocopiers, tv sets, washing machines, refrigerators, vacuum cleaners, air conditioners, cookers, microwave ovens, kerosene fan heaters, NiCd battery chargers, voice recognisers etc.)

- motion control, transport and power systems applications (e.g., power transmission control, automatic train operation control, helicopter control, autonomous vehicle motion planning, crane control, automotive engine and transmission control, spacecraft control, space camera tracking systems etc.)

- industrial process control (refining, distillation, cement kiln incineration plants etc.)

- robotics and manufacturing (e.g., electrical discharge machine, robot motion planning and control etc.)

- dedicated fuzzy software (development of decision making tools) and hardware (fuzzy semiconductor devices, processors, controllers etc.)

The thriving present number of fuzzy logic applications results in a growing tendency to extend the theoretical framework of fuzzy sets theory which already encompasses areas such as fuzzy arithmetic, fuzzy mathematical programming, fuzzy topology, fuzzy graph theory and fuzzy data analysis. It has been suggested that, in fact, any crisp theory can be "fuzzified" by extending the definition of a set within that theory to the concept of a fuzzy set [Zadeh 1994].

The following sections briefly review the basic concepts and definitions of fuzzy sets and systems [Wang 1994, Jang et al. 1997, Mendel 1995, Lin and Lee 1996], which will facilitate the description of neurofuzzy systems and the neurofuzzy tool life model developed in later chapters.

3.2 Basic concepts of fuzzy logic systems

A **fuzzy set** F defined on a **universe of discourse** U is characterised by a **membership function** which takes on values in the interval [0,1]. A fuzzy set is an extension of an ordinary subset whose membership value takes only two values, zero or unity. A membership function provides a measure of the degree of similarity of an element in U to the fuzzy subset. A fuzzy set may be represented as a set of ordered pairs of a generic element x and its membership function: $F = \{(x, \mu_F(x)) \mid x \in U\}$. Alternatively the fuzzy set F can be represented as $F = \left\{ \left[\mu_F(x) / x \right] \mid x \in U \right\}$, $F = \left\{ \sum_{x \in U} \left[\mu_F(x) / x \right] \right\}$ or $F = \int_U \mu_F(x) / x$.

The **support** of a fuzzy set F is the crisp set of all points $x \in U$ such that $\mu_F(x) > 0$. The **kernel** or **core** of a fuzzy set is the point(s) $x \in U$ at which $\mu_F(x)$ achieves its maximum value. If the support of a fuzzy set F is a single point in U at which $\mu_F(x) = 1$, the set is called a fuzzy singleton. Linguistic variables are those whose values are not numbers but words or sentences in a natural or artificial language. Let u denote the name of a linguistic variable. Numerical values of a linguistic variable u are denoted x, where $x \in U$. The concept of a linguistic variable is central in fuzzy sets theory. Describing quantities with linguistic variables instead of precise numerical values is essentially a data compression method, often referred to as **granulation** [Zadeh 1994] more powerful than quantisation. The main difference between granulation and standard quantisation is that in the former case the values are overlapping fuzzy sets and not intervals with crisp boundaries. The transition from one linguistic variable to another is gradual and not abrupt as it is in the case of quantisation, providing continuity and increased robustness.

A fuzzy logic system (Fig. 3.1) contains four main components; a **fuzzy rule base**, a **fuzzifier**, an **inference engine** and a **defuzzifier**. It can be viewed as a mapping from crisp inputs to crisp outputs, *y=f(x)*, where ***x*** is the input vector and *y* the output. The fuzzy rules may be provided by experts or can be derived from numerical data. The fuzzifier converts crisp numbers into fuzzy sets. This is necessary in order to activate fuzzy rules, expressed in terms of linguistic variables and having fuzzy sets associated with them. The inference engine maps fuzzy sets into fuzzy sets, by combining the fuzzy rules. Finally, the defuzzifier maps output fuzzy sets into crisp outputs.

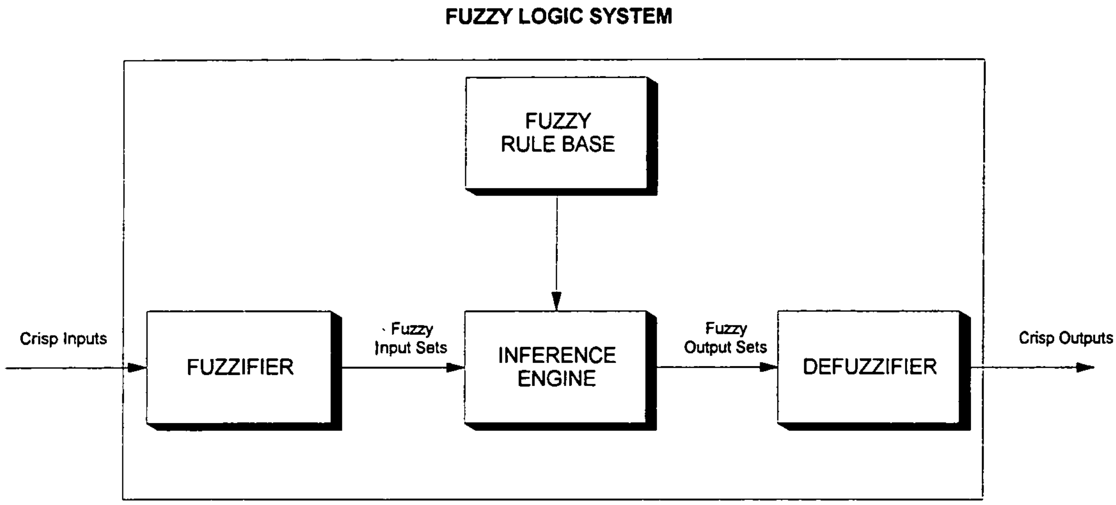

Figure 3.1: Block diagram of a fuzzy logic system

3.2.1 Triangular norms and conorms

Triangular norms (T-norms) and conorms (T-conorms) are employed for intersection and union operations respectively between fuzzy sets, instead of the classical union and intersection operations between crisp sets [Lin and Lee 1996, Jang et al. 1997]. A T-norm, denoted by ★, is a two-place function from [0,1] × [0,1] to [0,1], which includes fuzzy intersection, algebraic product, bounded product and drastic product, defined as:

$$x \star y = \begin{cases} \min\{x,y\} & \text{fuzzy intersection} \\ x \cdot y & \text{algebraic product} \\ \max\{0, x+y-1\} & \text{bounded product} \\ \\ x \ \text{if}\ y=1 & \\ y \ \text{if}\ x=1 & \text{drastic product} \\ 0 \ \text{if}\ x,y<1 & \end{cases} \tag{3-1}$$

where $x, y \in [0,1]$. A T-conorm, denoted by ⊕, is a two-place function from [0,1] × [0,1] to [0,1], which includes fuzzy union, algebraic sum, bounded sum and drastic sum, defined as:

$$
x \oplus y = \begin{cases} \max\{x,y\} & \text{fuzzy union} \\ x+y-xy & \text{algebraic sum} \\ \min\{1,x+y\} & \text{bounded sum} \\ \left.\begin{array}{l} x \ \text{if}\ y=0 \\ y \ \text{if}\ x=0 \\ 1 \ \text{if}\ x,y>1 \end{array}\right\} & \text{drastic sum} \end{cases} \tag{3-2}
$$

where $x, y \in [0,1]$.

3.2.2 Fuzzy relations

Fuzzy relations represent a degree of presence or absence of association, interaction, or interconnectedness between the elements of two or more fuzzy sets [Klir and Yuan 1995, Lin and Lee 1996]. Let U, V be two universes of discourse. A fuzzy relation, $R(U,V)$ is a fuzzy set in the product space $U \times V$, i.e. it is a fuzzy subset of $U \times V$ and is characterised by a membership function $\mu_R(x,y)$, where $x \in U, y \in V$, i.e.

$$
R(U,V) = \{(x,y), \mu_R(x,y) \mid (x,y) \in U \times V\} \tag{3-3}
$$

Since fuzzy relations are fuzzy sets, fuzzy set operations can be applied to them. Let $R(U,V)$ and $S(U,V)$ be two fuzzy relations in the same product space $U \times V$. The intersection and union of R, S, which are compositions of the two relations are then defined as:

$$
\mu_{R \cap S}(x,y) = \mu_R(x,y) \star \mu_S(x,y) \tag{3-4}
$$

$$
\mu_{R \cup S}(x,y) = \mu_R(x,y) \oplus \mu_S(x,y) \tag{3-5}
$$

where $\star$ is any t-norm and $\oplus$ any t-conorm.

***Fuzzy Relations and compositions on different product spaces*:**. Let $R(U,V)$, $S(V,W)$ be two fuzzy relations in the product spaces $U \times V$ and $V \times W$ respectively. The **fuzzy composition** of R and S denoted $R \circ S$ can be expressed by:

$$\mu_{R \circ S}(x,z) = \sup_{y \in V}[\mu_R(x,y) \star \mu_S(y,z)] \tag{3-6}$$

known as **sup-star composition** [Mendel 1995]. When U, V, W are discrete universes of discourse, then the supremum operation is the maximum. The most commonly used sup-star compositions are the sup-min and sup-product.

3.2.3 Fuzzy rule base

A fuzzy rule base consists of a collection of IF THEN rules which are expressed as:

$$R^{(l)}: \; IF \;\; u_1 \;\; is \;\; F_1^l \;\; and \;\; \ldots \;\; u_p \;\; is \;\; F_p^l \;\; THEN \;\; v \;\; is \;\; G^l \tag{3-7}$$

where $l = 1,2,\ldots, M$, M is the number of fuzzy rules in the rule base, F_i^l and G^l are fuzzy sets in $U_i \subset \Re$ and $V \subset \Re$ ($\Re$ denotes the set of real numbers), $\mathbf{u}=[u_1,u_2,\ldots,u_p]^T \in U_1 \times U_2 \times \ldots \times U_p$, and $v \in V$. $\mathbf{u}$ and v are linguistic variables. Their numerical values are $\mathbf{x} \in U$ *and* $y \in V$ respectively. Fuzzy rules can be extracted by numerical data either by letting the data establish the fuzzy sets that appear in the antecedents and consequents of the rules or prespecify fuzzy sets for the antecedents and consequents and then associate the data with these fuzzy sets. In the first approach the antecedent and consequent membership functions adapt to the locations of the data that are used to create the rules. In the second approach the fuzzy sets are established by determining domain intervals for all input and output variables. Then the degrees (i.e. the membership function values) of the elements of each numerical sample are determined. Each variable is assigned to the region with the maximum degree and consequently one rule can be derived from each input/output pair. The more data are available the more likely is to obtain conflicting rules. One way of resolving such conflicts is to assign a degree to each rule and accept only the rule with the maximum degree from a group of conflicting rules.

3.2.4 Fuzzy inference

In the fuzzy inference engine fuzzy logic principles are used to combine fuzzy IF-THEN rules from the fuzzy rule base into a mapping from fuzzy input sets $U_1 \times U_2 \times ... \times U_p$ to fuzzy output sets in V. In crisp logic a rule will be fired only if the first premise is exactly the same as the antecedent of the rule, and the result of such rule-firing will be the actual consequent. In fuzzy logic, on the other hand, a rule is fired so long as there is a non-zero degree of similarity between the first premise and the antecedent and the result of such rule-firing will be a consequent that has a non-zero degree of similarity to the rule's consequent. Each rule is interpreted as a **fuzzy implication**. A fuzzy implication, denoted by $A \rightarrow B$, is a special kind of fuzzy relation in $(U \ni A) \times (V \ni B)$ corresponding to an interpretation of the fuzzy IF-THEN rule based on intuitive criteria or generalisation of the classical logic, where $A \equiv F_1^l \times ... \times F_p^l$ and $B \equiv G^l$. The overall mapping is considered to be performed by means of a membership function $\mu_{A \rightarrow B}(\mathbf{x}, y)$, which measures the degree of truth of the implication between **x** and y. Generally speaking there are two basic interpretations of the fuzzy implication $A \rightarrow B$, either A coupled with B, or A entails B.

A coupled with B: $R = A \rightarrow B = A \times B = \int_{UXV} \mu_A(\mathbf{x}) \star \mu_B(y) / (x,y)$ which results in different fuzzy relations depending on the specific T-norm operation employed.

A entails B: The basic relations of this interpretation of the fuzzy implication are [Jang et al. 1997]:

- Material implication: $R = A \rightarrow B = \neg A \cup B$.
- Propositional calculus: $R = A \rightarrow B = \neg A \cup (A \cap B)$.
- Extended propositional calculus: $R = A \rightarrow B = (\neg A \cup \neg B) \cup B$.
- Generalised Modus Ponens: $\mu_R(x,y) = sup\{c \mid \mu_A(x) \star c \leq \mu_B(y)$ and $0 \leq c \leq 1\}$.

where $\neg A$ stands for the complement of the fuzzy set A and is defined as $\mu_{\neg A}(x) = 1 - \mu_A(x)$. All the above formulae reduce to the familiar identity $R \equiv A \rightarrow B \equiv \neg A \cup B$ when A and B are propositions in the sense of two-valued (crisp) logic. According to the particular choices made for T-norms, fuzzy conjunction and fuzzy disjunction, a number of different possible interpretations of the fuzzy implication can be derived. The final fuzzy set B which is determined by all the rules of the rule base is obtained by combining the fuzzy sets resulting by each firing rule. The rules are usually connected using a T-conorm (i.e. the fuzzy union) and this seems to give very good results when minimum or product implication operators are used [Mendel 1995]. However, there does not appear to be a unique or compelling theoretical reason for combining rules using a T-conorm.

Combining rules additively is an attractive approach [Kosko 1992, Kosko and Dickerson 1995, Dickerson and Kosko 1996]. An additive combiner which can be interpreted as an adaptive filter whose inputs are the output fuzzy sets is shown in Figure 3.2. The weights w_i can be thought as providing degrees of belief to each rule. If information concerning rules reliability is not known ahead of time, then either all w_i are set equal to unity or a training procedure is used to learn optimal values for the weights.

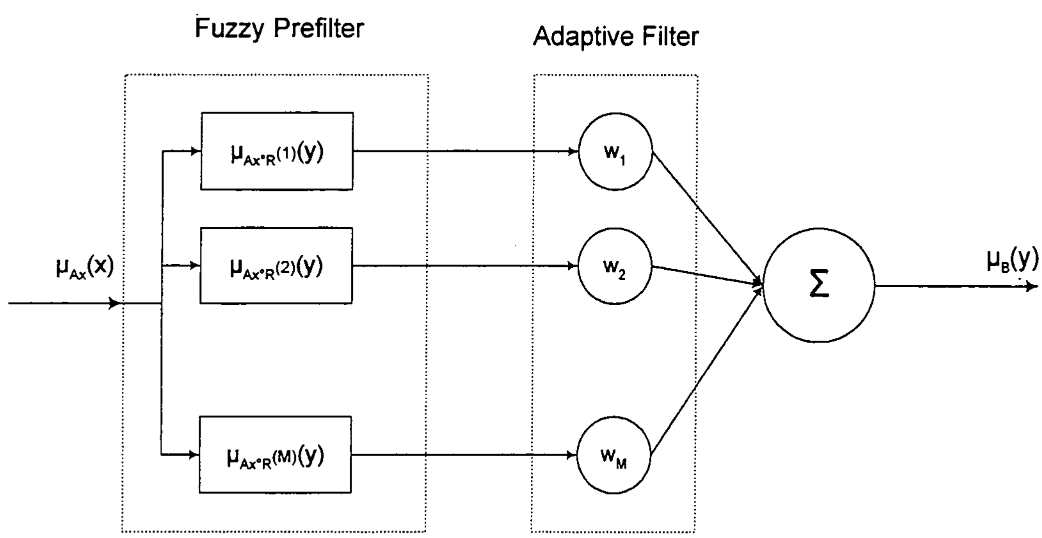

Figure 3.2: Kosko Additive Fuzzy Combiner

Generally speaking, the process of fuzzy or approximate reasoning can be summarised as follows [Jang et al. 97]:

- Determine the degree of compatibility of known facts with respect to each antecedent membership function of fuzzy rules.
- Calculate the degree to which the antecedent part of each rule is satisfied, i.e. the firing strength of each rule, by combining the degrees of compatibility with respect to each antecedent membership function in a rule, using fuzzy conjunction or disjunction operators.
- Calculate the qualified consequent of each fuzzy rule, by applying the firing strength of each rule to the corresponding consequent membership function.
- Aggregate all the qualified consequent membership functions to obtain an overall output membership function.

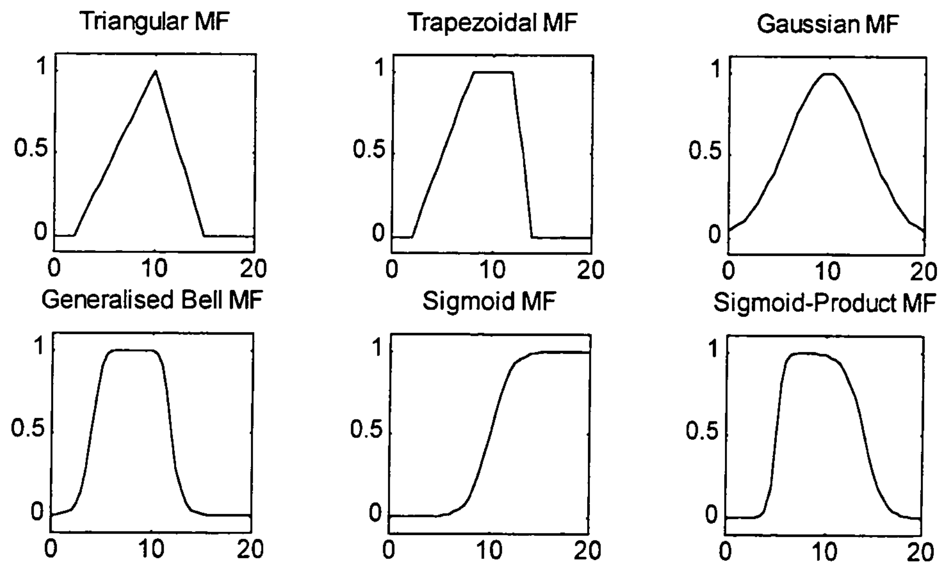

Figure 3.3: Membership functions

3.2.5 Fuzzification

The fuzzifier maps a crisp point $\mathbf{x} = [x_1, \ldots, x_n] \in U$ into a fuzzy set A^* in *U*. The most widely used fuzzifier is the singleton fuzzifier, which is nothing more than a fuzzy singleton. Singleton fuzzification may not always be adequate, especially when data is corrupted by measurement noise. Nonsingleton fuzzification provides a means for handling such uncertainties [Mouzouris and Mendel 1997]. A nonsingleton fuzzifier is one for which $\mu_{A^*}(\mathbf{x}')=1$ and $\mu_{A^*}(\mathbf{x})$ decreases from unity as $\mathbf{x}$ moves away from $\mathbf{x}'$. Examples of membership functions are the triangular, trapezoidal, Gaussian, generalised bell, sigmoid and product of two sigmoids (Figure 3.3). The broader these functions are, the greater is the uncertainty about $\mathbf{x}'$.

3.2.6 Defuzzification

Defuzzification produces a crisp output from the fuzzy set that is the output of the inference block in Figure 3.1. Many defuzzifiers have been proposed in the literature [Mendel 1995, Lin and Lee 1996]; however, there are no scientific bases for any of them. Consequently defuzzification is an art rather than a science. From an engineering perspective of fuzzy logic, one criterion for the choice of a defuzzifier is computational simplicity. The most commonly used defuzzifiers are [Mendel 1995]:

1. **Maximum Defuzzifier**: The defuzzifier output is the value of y for which $\mu_B(y)$ attains a maximum value. This choice often leads to peculiar results.
2. **Mean of Maxima Defuzzifier**: This defuzzifier first determines the values of y for which $\mu_B(y)$ is a maximum. It then computes the mean of these values as its output. It can also lead to unacceptable results.
3. **Centroid Defuzzifier**: It is often referred to as weighted average defuzzifier. The defuzzifier determines the centre of gravity (centroid), $\bar{y}$ of B and uses this value as the output of the fuzzy logic system:

$$\bar{y}=\left[\int_S y\mu_B(y)dy\right]\Big/\left[\int_S \mu_B(y)dy\right] \tag{3-8}$$

where S denotes the support of $\mu_B(y)$. When S is discretised $\bar{y}$ is given by:

$$\bar{y}=\left[\sum_{i=1}^{I} y_i\mu_B(y_i)\right]\Big/\left[\sum_{i=1}^{I} \mu_B(y_i)\right] \tag{3-9}$$

The centroid defuzzifier, is often difficult to compute. It has been shown [Mendel 1995] that for product inference and additive combining rules, $\bar{y}$ can be computed using centroid information about the individual M rules.

4. **Height Defuzzification**: The defuzzifier first evaluates the centre of gravity, $\bar{y}^l$, of the fuzzy set B^l, which is associated with the activation of rule $R^{(l)}$. Then it evaluates $\mu_{B^l}(y)$ at $\bar{y}^l$ and the output of the FLS is:

$$y_h = \left[\sum_{l=1}^{M} \bar{y}^l \mu_{B^l}(\bar{y}^l)\right] \Bigg/ \left[\sum_{l=1}^{M} \mu_{B^l}(\bar{y}^l)\right] \tag{3-10}$$

It is very easy to use (3-10) because the centres of gravity of commonly used membership functions are known beforehand. Regardless of whether minimum or product inference is used, the centre of gravity of B^l for:

- A symmetric triangular consequent membership function is at the apex of the triangle.
- A Gaussian consequent membership function is at the centre value of the Gaussian function.
- A symmetric trapezoidal membership function is at the midpoint of its support.

Although (3-10) is easy to use it also has its own drawback. Regardless of whether or not the consequent membership function is very narrow (broad), i.e. there is a strong belief (disbelief) in that rule, or is very broad, the height defuzzifier gives the same result.

5. **Modified Height Defuzzifier**: The modified height defuzzifier first evaluates $\mu_{B^l}(y)$ at $\bar{y}^l$ and then computes the output of the FLS as:

$$y_h = \left[\sum_{l=1}^{M} \bar{y}^l \mu_{B^l}(\bar{y}^l)/(\delta^l)^2\right] \Bigg/ \left[\sum_{l=1}^{M} \mu_{B^l}(\bar{y}^l)/(\delta^l)^2\right] \tag{3-11}$$

where δ^l is a measure of the spread of the consequent for rule $R^{(l)}$. For triangular and trapezoidal membership functions, δ^l could be the support of the triangle or trapezoid, whereas for Gaussian membership functions, δ^l could be its standard deviation.

It should be noted that defuzzification is the final stage of a fuzzy inference system which yields a crisp output as a result of fuzzy reasoning. However, the output of such a system often is to be utilised for other complex decision making or optimisation activities. It is then desirable to obtain a degree of the quality of this output, by means of quantifying the uncertainty about it. In probabilistic terms this problem is dealt with by introducing confidence intervals. Such a treatment can not be suitable for fuzzy systems outputs, since that would involve converting a problem that is inherently non-probabilistic to a probabilistic one. An alternative approach proposes the introduction of a ranking index for the outputs of a fuzzy system and a mathematical formulation for doing so has been suggested [Saade 1996]. Nonetheless, it is generally worth examining whether the fuzzy outputs of a fuzzy inference engine can be utilised within the fuzzy set theory framework, i.e. by introducing the fuzzy outputs directly to a fuzzy logic based decision making or optimisation process.

3.3 Designing a fuzzy logic system

Designing a fuzzy logic system involves decisions upon the type of fuzzification (singleton or nonsingleton), functional forms for membership functions (triangular, trapezoidal, piecewise linear, Gaussian etc.), parameters of membership functions (fixed ahead of time, tuned during a training procedure), composition (max-min, max-product), inference (minimum, product) and defuzzification (centroid, height, modified height etc.). An important consideration in designing a fuzzy systems is the partitioning of the input space in order to form the antecedents of the fuzzy rules. The most common methods of input space partitioning are the grid, tree and scatter partitioning [Jang et al. 1997]. Generally speaking, the overall process of building a fuzzy system, termed fuzzy modelling, can be pursued in two stages. The first stage is the identification of the surface structure, which includes the following tasks [Jang et al. 1997]:

1. Select relevant input and output variables.
2. Chose a specific type of fuzzy inference system.
3. Determine the number of linguistic terms associated with each input and output variables.

4. Design a collection of fuzzy if-then rules.

The above can be accomplished by using background knowledge (common sense, simple physical laws e.t.c.) about the target system, information provided by human experts who are familiar with the target system, or simply by trial and error. After obtaining a rule base which is relatively descriptive of the target system behaviour, a detailed definition of the linguistic terms is needed. In other words, it is necessary to identify the deep structure which determines the membership function of each linguistic term. The deep structure identification involves the following steps:

1. Chose an appropriate family of parametrised membership functions.
2. Determine the membership function parameters by extracting knowledge from human experts.
3. Refine the membership function parameters using regression and optimisation techniques.

The first two tasks require the availability of human experts, while the third one relies upon the availability of an appropriate input-output data sets.

Among the different approaches to intelligent computation, fuzzy logic provides a strong framework for achieving robust and yet low cost solutions. The main principle of fuzzy set theory is to relax the requirements for certainty and rigor, recognising that there is no profound reason for pursing high cost solutions that would attempt to achieve precision, when this is either impractical or impossible. Thus fuzzy logic appears to offer a model for reasoning which, being approximate rather than exact, is closer to the model of thinking of the human mind. Fuzzy logic can be further solidified by the introduction of learning capabilities, such as those of artificial neural networks. The next chapter presents a brief review of some methods for neurofuzzy systems integration and a detailed description of the specific modelling approach, which has been employed in the present work for the development of a neurofuzzy tool life model.

Chapter 4

ADAPTIVE NEUROFUZZY SYSTEMS

4.1 Motivation

The design of a fuzzy inference system involves the selection of a number of fuzzy model parameters. When only human expert knowledge is taken into account, this selection is rather arbitrary. The key question is how to tune the parameters of a fuzzy logic system using available numerical sample data. Neural network models are computational structures well suited for such tasks. Therefore, considerable amount of research is now being carried out in order to incorporate neural network-like learning capabilities into fuzzy logic systems. As a result, a constantly increasing number of integrated neurofuzzy models are reported in the literature, which aim to amalgamate the benefits of both computational approaches, namely the learning capabilities of neural networks and the representation power of fuzzy logic systems. From the neural network viewpoint this integration aims at providing more insight and transparency on the way that a neural structure represents knowledge. On the other hand, this enables fuzzy inference systems to exhibit real adaptivity via learning. The complementary rather than competitive nature of the two computational approaches enabled some equivalence between certain classes of neural networks and fuzzy logic systems to be established. The benefit of obtaining neurofuzzy representations is that the resulting models can incorporate both expert linguistic knowledge, as well as sample data information, a task that can not be performed by utilising solely neural networks or traditional artificial intelligence techniques.

The main approaches in merging neural networks and fuzzy logic technologies can be categorised as [Lin and Lee 1996]:

- **Neural fuzzy** (or **neurofuzzy**) systems, where neural networks are used as tools in fuzzy models. The overall functionality is of a pure fuzzy nature with all the distinct elements of a fuzzy inference system, i.e. fuzzification, fuzzy rule base, fuzzy inference and defuzzification. The numerical processing is carried out via neural network-like computational structures. Neural-fuzzy systems aim at automating the process of designing a fuzzy inference system. Thus, neural network related learning techniques are also employed for the adaptation of the neurofuzzy system parameters, such as premise or consequent parameters of fuzzy rules. After learning, the acquired fuzzy rules are easily understood by the user. Therefore, neural-fuzzy systems can replace classical non-adaptive fuzzy systems in any application, whilst retaining the transparency of a pure fuzzy system. [Lin and Lee 1995, Lin 1996, Wang 1995, Jang 1993, Lin and Lu 1996, Lin and Lu 1995, Sun 1994]

- **Fuzzy neural networks**, i.e. neural networks with some "fuzzified elements". For instance a neuron can be replaced by a fuzzy neuron which provides a fuzzy output rather than a crisp signal as a response to input stimuli. Alternatively, fuzzy rules can facilitate the convergence of a neural network training algorithm. Another example is the substitution of the summation or product operators by the minimum and maximum operators often employed in fuzzy systems. Yet, the structure of the model is still of neural network type. What differs from a pure neural network is the enhancement in user flexibility provided by the embodiment of fuzzy principles in the network, as well as the improvement in the robustness of the model's behaviour. In contrast with neurofuzzy models, the overall functionality of a fuzzy neural network is not that of a pure fuzzy inference system. Fuzzy neural networks are mostly employed in pattern recognition tasks. [Zhag et al. 1996, Lee et al. 1996, Pal and Mitra 1992, Mitra and Pal 1995, Nie 1995, Dickerson and Kosko 1996, Carpenter and Grossberg 1994, Buckley and Hayashi 1994]

- **Hybrid fuzzy-neural** structures, where separate neural and fuzzy models are incorporated in order to carry out distinct tasks which complement each other in achieving a common goal. Such computational structures are quite often application-oriented and therefore their applicability can be quite diverse [Arabshahi et al. 1996].

For the purposes of the present work, a neurofuzzy approach is adopted. This can offer both the transparency of a fuzzy system and the neural learning capability for the design of a data-driven tool life model. Therefore a neurofuzzy structure, namely that of the

Adaptive Network Based Fuzzy Inference System (ANFIS) [Jang 1993] is adopted for tool life prediction. The reasons for this choice are:

- The ANFIS model is a typical case of a fuzzy inference system, comprising a fuzzification stage, a fuzzy rule base, a fuzzy inference engine and a defuzzifier.
- ANFIS are adaptive models trainable by efficient neural-network learning algorithms [Jang 1993, Jang and Mizutani 1996].
- The universal approximation capabilities of ANFIS models have been established [Jang and Sun 1993].
- ANFIS models have already found several engineering applications ranging from nonlinear system identification, chaotic time series prediction, automobile miles per gallon (MPG) prediction, adaptive noise cancellation, printed character recognition etc [Jang et al. 1997].
- The ANFIS architecture is directly supported by the Fuzzy Logic Toolbox of MATLAB [Jang and Gulley 1995]. MATLAB is probably the most widely used platform for engineering numerical computation and visualisation.

A brief discussion about some conceptual analogy and functional equivalence that exists between fuzzy systems and neural networks is given in the next section. Then a description of the ANFIS structure and functionality follows. The chapter concludes by reviewing some methods for constructing ANFIS models based on sample data.

4.2 Conceptual and functional equivalence of neural and fuzzy systems

At a high abstraction level, a fuzzy system processes known facts by applying a reasoning mechanism to utilise knowledge represented by fuzzy IF-THEN rules, acquired by some knowledge acquisition procedure. Neural networks process numerical information by performing a certain type of computation, determined by their prespecified structure. The computation is parametrised by a number of connection weights, or other parameters of the processing units, which are derived following a learning procedure.

A qualitative comparison can reveal a correspondence between [Nie et al. 1997]:

- The structure of a neural network and knowledge representation in a fuzzy system.
- The computation performed in a neural structure and the reasoning mechanism in a fuzzy inference system.
- The neural network learning and the fuzzy rules acquisition procedure.

A more detailed qualitative comparison of the characteristics of fuzzy and neural systems is shown in Table 4.1 [Nie at al. 1997].

Issue	**Fuzzy Systems**	**Neural networks**
Representation (Structure)	Cognitive level Qualitative Abstract Rule-based Universal approximator Localised Transparent	Biological Level Quantitative Detailed Unit (neuron) -based Universal approximator Distributed/localised Black-box
Reasoning (Computing)	Logic Parallel Numerical Membership function Interpolation	Algebraic Parallel Numerical Activation function Interpolation
Acquisition (learning)	Expert + programming Not goal-directed Without feedback	Sample + training Goal-directed Feedback

Table 4.1: Comparative characteristics of fuzzy systems and neural networks [Nie at al. 1997]

For a complete description of a fuzzy logic system (FLS) it is necessary to establish a mathematical formula that maps a crisp input $\mathbf{x}$ into a crisp output $y = f(\mathbf{x})$. In order to write such a formula, specific choices have to be made for the FLS elements. For singleton fuzzification, max-product composition, product inference and height or weighted average defuzzification, leaving the choice of membership functions open, the I/O relationship is:

$$y = f(\mathbf{x}) = \sum_{l=1}^{L} \bar{y}^l \phi_l(\mathbf{x}) \tag{4-1}$$

where L is the number of fuzzy rules, superscript l denotes the l-th fuzzy rule and $\phi_l(\mathbf{x})$ are called fuzzy basis functions (FBF) and are given by [Wang and Mendel 1992, Mendel 1995]:

$$\phi_l(\mathbf{x}) = \prod_{i=1}^{n} \mu_{F_i^l}(x_i) \Bigg/ \left[\sum_{l=1}^{L} \prod_{i=1}^{n} \mu_{F_i^l}(x_i)\right] \tag{4-2}$$

where $\mu_{F_i^l}(x_i)$ is the membership function corresponding to the i-th input fuzzy set of the l-th fuzzy rule and n is the dimension of the input vector. Although the index l on the FBF seems to be associated with a rule number, each FBF is affected by all the rules because of the denominator in $\phi_l(\mathbf{x})$; hence it is only partially correct to associate the j-th FBF to the j-th rule. However, when a fuzzy rule is added or removed, thereby increasing or decreasing M, then a FBF is added or removed from the FBF expansion.

It is important to note that by interpreting a fuzzy logic system as fuzzy basis function expansion, places it among the more global perspective of function approximation. In particular, it has been proved that FBFs of the form (4-1) are universal approximators, i.e. they can approximate any continuous real function into a compact set to arbitrary degree of accuracy [Wang and Mendel 1992]. Other proofs of universal approximation theorems for different classes of fuzzy logic systems have also been reported in the literature, including those comprising additive fuzzy systems with singleton fuzzification, product inference and implication and weighted average defuzzification [Kosko and Dickerson 1995] and for some classes of ANFIS models [Jang et al. 1997].

The relationships between FBFs and other basis functions have also been examined [Wang and Mendel 1992, Kim and Mendel 1995, Hunt et al. 1996] and FBFs are shown to be more general than radial basis functions, generalised radial basis functions and hyper basis functions. The denominator in (4-1) which result from height or weighted average defuzzification, serve to normalise the numerators of the FBFs. The numerators are also radially symmetric, so FBFs can also be referred to as normalised radial basis functions. It should be noted that FBFs are not normalised by abstraction but rather by design of the overall Fuzzy Logic System.

Neural networks with one hidden layer can also be expressed as a basis function expansion of the form of equation (4-1). In such a representation, $\phi_l(\mathbf{x})$ stands for the activation function employed, l indicates the l-th hidden node out of a total of L hidden nodes and $\bar{y}$ could be replaced by the relevant connection weight. As a result of such comparisons, the functional equivalence between certain classes of fuzzy inference systems and neural networks have been established [Wang and Mendel 1992, Hunt et al. 1996, Jang and Sun 1995]. This equivalence apply mainly to neural networks with local receptive fields, such those employing Gaussian or B-splines basis functions. Equivalent neurofuzzy models have therefore been derived with premise membership functions of the same type as the activation functions of the neural units. The benefit of interpreting these networks as neurofuzzy structures is not related to enhanced modelling performance compared to that of the pure neural network, but to the transparency that a fuzzy representation offers [Brown and Harris 1995]. This transparency allows for a much easier and more meaningful model initialisation procedure to be followed, thus reducing development time and effort. In addition, such neurofuzzy representations are easily evaluated and validated, as they can store structured knowledge in the form of fuzzy rules.

Fuzzy basis functions can include both linguistic and numerical information. When rich sample data are available, model building can be data-driven. In the absence of an adequate sample data set, the modelling task can be greatly facilitated by incorporating available expert knowledge. An inherent problem in designing fuzzy logic systems is that as the number of input variables and the number of overlapping regions defined in the universe of discourse of each one of the variables increases, the fuzzy rule base becomes very complex. In practice, however, one never needs the complete set of fuzzy rules, since there are large regions of the input space never seen in actual applications. Therefore, it is important to balance the need for high resolution in covering the input space with low complexity. Several different approaches for designing a fuzzy rule base from available input/output data have been suggested [Wang 1994, Jang et al. 1997, Lin and Lee 1994, Nie 1995, Sun 1994, Abe and Lan 1995, Lin and Lee 1996, Lin 1996, Dickerson and Kosko 1996].

Next, the discussion is focused on the analysis of the ANFIS model, which is employed in the present work for tool life prediction.

4.2 Adaptive Network-Based Fuzzy Inference Systems (ANFIS)

The general ANFIS architecture is shown in Fig. 4.1, where nodes belonging to the same layer have similar functionality. ANFIS modelling is equivalent with the so called **Reduced Direct Fuzzy Reasoning** method [Yager and Filev 1994a].

The input/output mapping of each node is explained below. The notation O_{ij} denotes the output of the *i*-th node in the *j*-th layer.

- **Inputs**: An input vector $x = [x_1, \ldots, x_i, \ldots, x_n]^T$ is applied to the first ANFIS layer.
- **Layer 1**: Each node A_{ij}, i=1,..,n, j=1,..,L, where n is the dimension of the input vector and L is the number of the fuzzy rules, is an adaptive node which receives x_i as input and provides $\mu_{A_{ij}}(x_i)$ as output, i.e. the degree (membership value) to which the input satisfies the quantifier A_{ij}. The membership function can be any parametrised membership function. The parameters of this function are referred to as premise parameters.
- **Layer 2**: Every node at this layer is a fixed node, labelled Π whose output is the product (fuzzy intersection) of the incoming signals:

$$O_{j2} = w_j = \prod_{i=1}^{n} \mu_{A_{ij}}(x_i), j=1,..L \tag{4-3}$$

Therefore, the *j*-th node of this layer represents the firing strength of the *j*-th fuzzy rule.

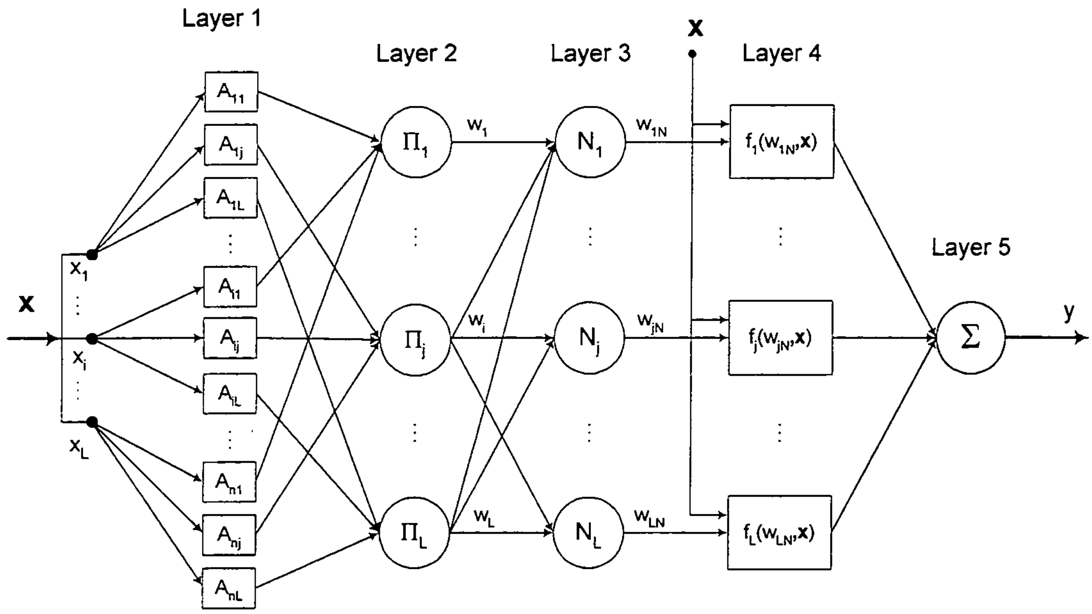

Figure 4.1: ANFIS architecture.

- **Layer 3**: Every node in this layer is a fixed node, labelled N. The *i*-th node calculates the ratio of the *i*-th rule's firing strength to the sum of all rules' firing strength, i.e. a normalised value for the firing strength of the *i*-th rule:

$$O_{j3} = w_{jN} = \frac{w_j}{\sum_{i=1}^{L} w_i}, j=1,..,L \tag{4-4}$$

For convenience, outputs of this layer are called normalised firing strengths.

- **Layer 4**: Every node in this layer is an adaptive node with the following node function:

$$O_{j4} = w_{jN} \cdot f_j(\mathbf{x}), \quad j=1,..,L \tag{4-5}$$

where the consequent function $f_j(\mathbf{x})$ is a crisp function of the input vector. This choice for consequent parts was earlier suggested by Sugeno [Takagi and Sugeno 1985] and is usually referred to as Sugeno reasoning or Sugeno fuzzy model. A common choice for $f_i(\mathbf{x})$ is a polynomial function. The order of the polynomial characterises the order of the Sugeno fuzzy model. For a first order Sugeno model

$f_j(\mathbf{x}) = \sum_{i=0}^{n} a_{ij} \cdot x_i$, with $x_0 = 1$ and j=1,..,L. The parameters involved in this layer (a_{ij}, i=0,1,..,n, j=1,...,L, for a first-order Sugeno fuzzy model) are referred as consequent parameters.

- **Layer 5**: The single node in this layer is a fixed node, labelled Σ, which calculates the overall output as the summation of all the incoming signals:

$$O_{15} = \sum_{j=1}^{L} w_{jN} \cdot f_j(\mathbf{x}) = \frac{\sum_{j=1}^{L} w_j \cdot f_j(\mathbf{x})}{\sum_{j=1}^{L} w_j} \tag{4-6}$$

In essence, equation (4-6) is the result of centroid defuzzification of the following output fuzzy set:

$$Y = \left\{ \frac{w_1}{f_1(\mathbf{x})} + \frac{w_2}{f_2(\mathbf{x})} + \cdots + \frac{w_L}{f_L(\mathbf{x})} \right\}$$

where the notation introduced in paragraph (3-2) is followed for the definition of a fuzzy set. It is worth noting that the output of each individual fuzzy rule of the ANFIS model is a fuzzy singleton.

The functional equivalence between ANFIS models and RBFs has been shown in [Jang 1993] for some restricted ANFIS class with Gaussian membership functions having constant widths and later was extended to encompass ANFIS models with Gaussian functions of arbitrary widths and arbitrary Sugeno type consequent parts [Hunt et al. 1996].

In order to build a complete neurofuzzy inference model, two distinct identification procedures have to be followed. The first one is the **structure identification**, which involves the determination of the number of fuzzy rules and the partition pattern. The second phase is the **parameter identification**, which is relevant to the selection and optimisation of the fuzzy inference system parameters. Within the general framework of system identification, structure identification applies a priori knowledge in order to determine a class of models within which the search for the most suitable model is to be conducted. Therefore, structure identification relies heavily on the experience and intuition of the designer. When the available a priori knowledge concerning the system to be modelled is very limited, structure identification becomes a very difficult problem, usually solved by following a trial and error procedure. The more knowledge is available about the target system, the easier it is to tackle the structure identification problem.

Parameter identification of ANFIS models can be performed by employing both back-propagation-type gradient descent to update premise parameters, which determines the shape and positions of the membership functions and the least squares method to identify the consequent parameters. In the case of a first-order Sugeno-type ANFIS model, a hybrid learning rule has been suggested [Jang 1993]. Specifically the rule involves two stages; first, in the "forward pass", functional signals go forward till layer 4 and the consequent parameters are identified using a Least Squares Estimator (LSE). while during the backward pass the error signal is propagated backward and the premise parameters are updated by gradient descent. Generally speaking, the specification of the exact learning procedure depends on the problem to be solved and consequently on the prespecified structure of the identifier. Several methods for rule base structure identification have also been proposed in the literature [Sun 1994, Abe and Lan 1995, Lin 1996, Chiu 1994, Jang 1994]. Before proceeding into a more detailed analysis of the ANFIS structure and parameter identification procedures, some general remarks on the approximation capabilities of neurofuzzy models as well as on their equivalence to certain classes of neural networks will be mentioned in the next section.

4.4 ANFIS structure identification

The determination of the initial ANFIS structure involves the selection of:

1. Relevant input variables.
2. Initial ANFIS architecture, including:
    - Input space partitioning.
    - Number of membership functions for each input.
    - Number of fuzzy IF-THEN rules.

- Premise parts of fuzzy rules.
- Consequent parts of fuzzy rules.

3. Initial membership function parameters.

When training adaptive fuzzy networks, there is no need to find the optimal partition at the structure identification stage, since the goal is to find a reasonably good initial state for the network. Several different approaches to structure identification of ANFIS models can be found in the literature [Jang et al. 1997, Sun 1994]. They usually employ objective functions that represent either a **density measure** which accounts for the density of the distribution of the sample data points near the cluster centre, or a **typicality measure**, which is a measure of the quality of the potential cluster centre in terms of how tight is the adherence of the data points to the cluster centre. In linguistic terms, the density measure is closely related to the **support** of a fuzzy set, while the typicality measure is related to its **core.**

Among the different approaches for fuzzy model structure identification, subtractive clustering [Chiu 1994] is a simple and efficient algorithm, which is supported by the Fuzzy Logic Toolbox of Matlab [Jang and Gulley 1995]. Subtractive clustering is a variation of the mountain clustering method [Yager and Filev 1994b]. Instead of considering certain grid points as potential cluster centre candidates, which results in an exponential growth in the required computation as the dimension of the problem increases, subtractive clustering treats each individual data point as a potential cluster centre. Thus, the computation is simply proportional to the number of data points and independent of the dimension of the problem. Once the number of the fuzzy rules and the premise parameters corresponding to the cluster centres have been identified, the initial consequent parameters can be derived by linear least squares estimation (LSE). The description of the algorithm follows:

1. Let $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a set of sample data in an M-dimensional space. It is assumed, without loss of generality, that these points are normalised within a unit hypercube.
2. A density measure, D_i is attributed to each one of the data points, defined as:

$$D_j = \sum_{i=1}^{n} \exp\left(-\frac{\left\| \mathbf{x}_j - \mathbf{x}_i \right\|^2}{\left(r_a/2 \right)^2} \right) \tag{4-7}$$

where the radius r_a defines the neighbourhood of influence of the cluster centre. Data points that fall outside of this area only slightly contribute to the density measure.

3. Select the data point with the highest D_j as the first cluster centre, $\mathbf{x}_{c_j}$.
4. If D_{c_j} is the density measure of the last chosen cluster centre $\mathbf{x}_{c_j}$, then the density measure for each sample data is updated according to:

$$D_i = D_i - D_{c_j} \exp\left(-\frac{\left\| \mathbf{x}_i - \mathbf{x}_{c_j} \right\|^2}{\left(r_b/2 \right)^2} \right) \tag{4-8}$$

where the radius r_b defines an inhibition area in which the presence of a new cluster centre is discouraged. r_b is usually larger than r_a to prevent closely spaced clusters.

5. If the stopping criteria are not satisfied then go back to 3 else stop.

The stopping criteria are:

If D_1 is the potential of the first cluster centre then:

⇒ IF $D_i > \bar{\varepsilon} D_1$ accept $\mathbf{x}_i$ as a cluster centre and continue.

⇒ ELSE IF $D_i < \underline{\varepsilon} D_1$ reject $\mathbf{x}_i$ and end the clustering process.

ELSE

⇒ Let d_{min} the shortest of the distances between $\mathbf{x}_i$ and the previously found cluster centres.

⇒ IF $\frac{d_{min}}{r_a} + \frac{D_i}{D_1} \geq 1$ THEN accept $\mathbf{x}_i$ as a cluster centre and continue.

⇒ ELSE Reject $\mathbf{x}_i$, set the potential at $\mathbf{x}_i$ to 0 and CONTINUE.

CONTINUE: Select the data point with the next highest potential as the new $\mathbf{x}_i$ and re-check.

where $\bar{\varepsilon}$ is a threshold for the potential above which a data point is definitely accepted as a cluster centre, $\underline{\varepsilon}$ defines a threshold below which a data point is definitely rejected as a cluster centre. When the potential falls within the grey area between these two values the data point is accepted as a cluster centre if it provides a good trade-off between having a relatively large potential and being sufficiently far from the existing cluster centres.

The described above algorithm is by no means optimal. Other efficient algorithms like the CART (classification and regression tree) algorithm for tree induction, fuzzy C-means clustering, fuzzy k-d tree partitioning, fuzzy binary boxtree rule structure identification and focus-set-based rule combination can also be employed for ANFIS structure identification [Jang 1997].

4.5 ANFIS parameter identification

Once the ANFIS structure and initial parameters have been determined, the next step is to optimise these parameters based on available sample data. The set of ANFIS parameters can be decomposed into two distinct subsets, namely the subset of nonlinear premise parameters and the subset of linear consequent parameters. When the values of the premise parameters are fixed, the output is a linear combination of the consequent parameters. Thus, the linear parameters can be identified by employing a simple linear least-squares estimation (LSE) method, while a gradient descent method can be used for the identification of the nonlinear parameters. This hybrid learning is summarised in table 4.2. [Jang 1997]:

	Forward Pass	Backward Pass
Premise Parameters	Fixed	Gradient Descent
Consequent Parameters	LSE	Fixed
Signals	Node Outputs	Error Signals

Table 4.2: Hybrid learning for ANFIS parameter identification.

The consequent parameters derived are optimal under the assumption that premise parameters are fixed. The hybrid learning rule has the merit of reducing both the search space for the gradient descent learning as well as the time required for convergence.

4.6 ANFIS modelling with the Fuzzy Logic Toolbox

The Fuzzy Logic Toolbox offers two alternative methods for obtaining fuzzy rules directly from numerical data. The first one is based on a grid partitioning of the input space. The main weakness of this approach is related to the "curse of dimensionality" problem, i.e. the fast growth in the number of fuzzy rules as the complexity of the problem increases. The second approach involves the subtractive clustering algorithm described in the previous section. Thus, ANFIS structures with Gaussian-shaped membership functions for the premise parts of the fuzzy rules and first order Sugeno type consequent parts can be derived directly from numerical data. For each rule $j=1,...,L$, where L is the number of the fuzzy rules, the membership function corresponding to the i-th input is defined as:

$$\mu_{A_{ij}}(x_i) = e^{-\frac{1}{2}\left(\frac{x_i - c_{ij}}{\sigma_{ij}}\right)^2} \tag{4-9}$$

where c_{ij}, σ_{ij} are the premise parameters corresponding to the mean value and the standard deviation of the Gaussian function. The vectors $\mathbf{c}_j = [c_{1j}, c_{2j}, ..., c_{nL}]$, $j=1,...,L$, are the cluster centres D_j identified by the subtractive clustering algorithm, whereas the vectors $\sigma_j = [\sigma_{1j}, \sigma_{2j}, \ldots, \sigma_{nL}]$ are related to the parameters r_a that appear in (4-7). The functional mapping performed by the particular ANFIS model is:

$$f(\mathbf{x}) = \sum_{j=1}^{L}\sum_{i=0}^{n} a_{ij} x_i \frac{\prod_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x_i - c_{ij}}{\sigma_{ij}}\right)^2}}{\sum_{j=1}^{L}\prod_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x_i - c_{ij}}{\sigma_{ij}}\right)^2}} = \sum_{j=1}^{L}\sum_{i=0}^{n} x_i w_{jN} a_{ij} \qquad (4\text{-}10)$$

When the premise parameters are fixed the overall output is linear with respect to the consequent parameters. Therefore, once the initial premise parameters have been identified by the subtractive clustering algorithm, the consequent parameters can be derived using a simple Least Squares Estimation (LSE) algorithm. Let M be the number of available patterns. If these patterns are fed into the equation (4-10), the following set of linear equations is obtained, expressed in matrix form:

$$\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_2 & . & . & . & \mathbf{A}_M\end{bmatrix}^T \cdot \begin{bmatrix}\mathbf{q}_1 \\ \mathbf{q}_2 \\ . \\ . \\ . \\ \mathbf{q}_L\end{bmatrix} = \mathbf{y} \qquad (4\text{-}11)$$

or simply $\mathbf{A}\cdot\mathbf{q} = \mathbf{y}$, where $\mathbf{A}$ is a $M \times (n+1)L$ design matrix, $\mathbf{q}$ is a $(n+1)L \times 1$ parameter column vector and y is a $M \times 1$ output column vector. The rest of the notation employed in (4-11) is:

$$\mathbf{A}_k = \begin{bmatrix}\mathbf{A}_{1k} & \mathbf{A}_{2k} & . & . & . & \mathbf{A}_{jk} & . & . & . & \mathbf{A}_{Lk}\end{bmatrix},\ k = 1,..,M$$

with $\mathbf{A}_{jk} = \begin{bmatrix}w_{jN}x_{0k} & w_{jN}x_{1k} & . & . & . & w_{jN}x_{ik} & . & . & . & w_{jN}x_{nk}\end{bmatrix}$

where k stands for the k-th pattern and

$$\mathbf{q}_j = \begin{bmatrix}a_{0j} & a_{1j} & . & . & . & a_{ij} & . & . & . & a_{nj}\end{bmatrix}.$$

If $\left(\mathbf{A}^T\mathbf{A}\right)$ is nonsingular the optimal estimation of $\mathbf{q}$ on the basis of the existing M patterns is:

$$\overset{\wedge}{\mathbf{q}} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{y} \tag{4-12}$$

It is worth noting that the above equation requires the inversion of an $(n+1)L \times (n+1)L$ matrix. Evidently this involves relatively heavy computation for large numbers of fuzzy rules and high dimensional input spaces. However, instead of the LSE one could employ a Recursive Least Squares Estimation procedure. Following such an approach the consequent parameters can be identified with the following set of formulas:

$$\overset{\wedge}{\mathbf{q}}(k+1) = \overset{\wedge}{\mathbf{q}}(k) + \mathbf{P}(k+1)\mathbf{A}_{k+1}[y_{k+1} - \mathbf{A}_{k+1}\overset{\wedge}{\mathbf{q}}(k)] \tag{4-13}$$

$$\mathbf{P}(k+1) = \mathbf{P}(k) - \frac{\mathbf{P}(k)\mathbf{A}_{i+1}^T\mathbf{A}_{i+1}\mathbf{P}(k)}{1+\mathbf{A}_{i+1}\mathbf{P}(k)\mathbf{A}_{i+1}^T}, k = 0,1,\ldots,M-1 \tag{4-14}$$

with initial conditions $\overset{\wedge}{\mathbf{q}}(0) = \mathbf{0}$, and $\mathbf{P}(0) = \gamma\mathbf{I}$, where $\mathbf{I}$ is the identity matrix and γ is a large positive value.

The next stage in the ANFIS modelling procedure is that of parameter identification. The initially identified premise and consequent parameters should be adjusted in order to achieve a more accurate I/O mapping. When a pattern k is presented into the input ANFIS layer the network provides at its output an estimate $O_{15,k}$ for the desired output y_k. The cost function to be minimised is:

$$E = \frac{1}{2}\sum_{k=1}^{M}\left(y_k - O_{15,k}\right)^2 = \frac{1}{2}\sum_{k=1}^{M}e_k^2 = \sum_{k=1}^{M}E_k \tag{4-15}$$

where e_k is the modelling error for the k-th pattern and E_k is the k-th pattern component of the cost function. In order to minimise the above function by employing gradient descent, each one of the network premise parameters has to be modified towards the negative direction of the function's gradient over this parameter. This gradient can be expressed as a function of the gradient of the cost function over the nodes whose output depend on the value of this parameter. For example, to update the parameters c_{ij} :the relevant gradient is:

$$\frac{\partial E_k}{\partial c_{ij}} = \sum_{O^* \in S} \frac{\partial E_k}{\partial O^*} \cdot \frac{\partial O^*}{\partial c_{ij}} \tag{4-16}$$

where S is the set of nodes whose output depends on c_{ij}, while $\frac{\partial E_k}{\partial O^*}$ can be expressed as:

$$\frac{\partial E_k}{\partial O_{jl,k}} = \sum_{m=1}^{\#(l+1)} \frac{\partial E_k}{\partial O_{m(l+1),k}} \cdot \frac{\partial O_{m(l+1),k}}{\partial O_{jl,k}} \tag{4-17}$$

where #(l) is the number of the l-th layer nodes and $O_{jl,k}$ is the output of the jth node of the l-th layer when the k-th pattern is inserted into the network. Evidently, the gradient of the kth pattern cost function component over the output of the last layer node is:

$$\frac{\partial E_k}{\partial O_{15,k}} = -\left(y_k - O_{15,k}\right) \tag{4-18}$$

From equations (4-15)-(4-18) the update rule for c_{ij} is:

$$\Delta c_{ij} = -\eta \frac{\partial E}{\partial c_{ij}} \tag{4-19}$$

where η is the learning rate:

$$\eta = \frac{\eta^*}{\sqrt{\sum_H \left(\frac{\partial E}{\partial c_{ij}}\right)^2}} \tag{4-20}$$

In the above equation η^* is the step size [Jang 1993] of each gradient transition into the parameter space and H is the set of premise parameters. The above normalisation ensures that the effective step size, i.e. the learning rate is sensitive to the magnitude of the gradients over the total set of adjustable premise parameters. The step size can also be adjusted according to some heuristics similar to those usually employed in the back propagation training of feed forward neural networks, in order to assist convergence. After the antecedent parts parameters are updated the linear consequent parameters are again modified according to the LSE method described earlier.

4.7 Discussion

This chapter has examined the motivation for implementing neurofuzzy modelling as well as methods for obtaining neurofuzzy inference systems from numerical data. Neural fuzzy representations are capable of simultaneously handling both linguistic and numerical information. They are also capable of employing neural network-like learning algorithms. Their universality as system approximators together with the above mentioned characteristics compose a strong framework, useful for a wide range of modelling problems. A specific case of neurofuzzy modelling approach, namely adaptive network-based fuzzy inference system (ANFIS) has been described in detail. This model exhibits all the basic characteristics of a fuzzy inference system and is equipped with some powerful algorithms for structure and parameter identification. The ANFIS structure has been employed in the present work as a means for obtaining a simple and reliable tool life model for turning operations. The model development procedure is described in chapters 6-7. Before that, the tool life modelling problem is discussed in the following chapter.

TOOL WEAR AND TOOL LIFE

5.1 Introduction

Tooling technology has long been recognised as an element of vital importance within the manufacturing industry. Critical tooling decisions such as those related to tool selection [Maropoulos and Hinduja 1990, Maropoulos and Hinduja 1991, Maropoulos 1992, Maropoulos 1995, Maropoulos and Gill 1995], tool life management [Maropoulos 1995, Alamin 1996, Maropoulos and Alamin 1996], optimal determination of cutting conditions, as well as the on-line machining process monitoring and control [Ulsoy and Koren 1993] are based on the existence of reliable detailed machining process models. Among the decisive factors of process planning and control activities, tool wear and tool life considerations hold a dominant role. In fact, most of the process optimisation decisions take place off-line and are related to the determination of optimal cutting parameters. These are only meaningful when based on accurate tool life modelling. This modelling should provide reliable prediction of tool life for all the combinations of workpiece materials, cutting conditions and cutting tools employed. On the other hand, on-line process optimisation, either in the form of real time tool replacement strategy determination, or of adaptive machining process control relies on timely available accurate tool wear identification information.

Due to the importance of obtaining robust and reliable cutting process models, much research effort has been devoted to the investigation of the complex interrelationships between the factors influencing the machining process. Despite the significant research that has been carried out, both off-line tool life prediction, as well as real time tool wear identification and prediction are still problems open to research. The main reason for that is that tool wear is influenced by a wide variety of factors and some of them are of stochastic nature and extremely difficult to model. Clearly, even though cutting processes are among the first material processing technologies, introduced during the early industrialisation years, its nature is so complex and exposed to so many disturbances that make accurate tool life prediction and in-process tool wear identification very formidable tasks. In addition, the inherent variability of workpiece materials, cutting tools and machine characteristics, even among those belonging to the same type, increases the uncertainty about the machining optimisation problem.

5.2 Tool wear, tool life and machining optimisation

A manufacturing enterprise should always pursue the minimisation of production costs in order to achieve competitiveness. The costs related to a manufacturing facility can be categorised as follows [Chryssolouris 1992]:

- Equipment and facility costs: They encompass the costs of the equipment required for performing the manufacturing operation, as well as the costs of the facilities, buildings and infrastructure that hosts the equipment operation.
- Material costs: They refer to the raw material costs and the auxiliary material costs such as those of lubricants and coolants.
- Labour: The direct labour that is necessary for operating the facility.
- Energy: The level of significance of energy costs considerations depends on the nature of the particular industry.
- Maintenance and Training: The total facility maintenance costs including labour and spare parts, as well as the cost of training the labour to new equipment/technology.
- Overhead: The portion of the cost that is attributed to infrastructure support but is not directly relevant to the facility operation.
- Capital cost: The cost of the borrowed capital.

The cost of a particular machining operation is roughly determined by the labour, overhead and machine cost rates, the operating time per workpiece, the tool changing time and the costs related to the cutting tool itself . In the case of single point cutting tool operations, the production cost per workpiece (C_p) for single pass operations can be expressed as [Agapiou 1992, Chryssolouris 1992, Tan and Creese 1995]:

$$C_p = C_m\left[t_m + t_h + \frac{t_m}{T}\left(t_c + \frac{C_t}{C_m}\right)\right] = C_m\left(t_m + t_h + \frac{t_m t_c}{T}\right) + C_t\frac{t_m}{T} \tag{5.1}$$

where C_m is the total machine, direct labour and overhead rate (£/min), t_m is the effective cutting time (min), t_h is the workpiece handling time (min), T is the total effective cutting time of the cutting edge before it is replaced (min), which equals to the tool life when no preventive tool replacement policy is employed, t_c is the tool changing time (min) and C_t is the tool cost per cutting edge (£), which is equal either to the cost of the disposable insert, or the cost of regrinding the cutting edge. In cases of multi-pass operations, the cost per workpiece is given by :

$$C_p = C_m\left(t_h + \sum_{i=1}^{m} t_{mi} + t_c\sum_{i=1}^{m}\frac{t_{mi}}{T_i}\right) + C_t\sum_{i=1}^{m}\frac{t_{mi}}{T_i} \tag{5-2}$$

where m is the number of passes, t_{mi} is the cutting time at the ith pass and T_i is the total effective cutting time of the cutting tool used at the ith pass (i.e. either the total cutting time of the disposable insert or the total cutting time before regrinding the tool cutting edge). Evidently, in both cases of single or multi-pass operations, the cost per workpiece is directly related to the total effective cutting time of the cutting edge. This time is equivalent to the tool life, when no preventive tool replacement policy is applied. When such a policy is employed, the optimal value for T or T_i is determined on the basis of the expected tool life value, in order to make maximum use of the cutting capacity of the tool, while preventing the occurrence of a high number of cutting process stoppages for tool replacements, by appropriate selection of the cutting conditions [Agapiou 1992, Tan and Creese 1995, Sheikh et al. 1980, Iakovou et al. 1996, Billatos and Kendall 1990].

On the other hand, when either on-line tool replacement policy [Zhou et al. 1990, Wang et al. 1996] or in-process control [Altintas et al. 1996a, Altintas et al. 1996b, Ulsoy and

Koren 1989, Ulsoy et al. 1983, Lundholm et al. 1992] are to be employed, they can only be successfully implemented on the basis of accurate tool wear status identification information. In the former case an objective function for the optimal tool replacement strategy determination can be formulated, based on the estimated actual tool wear. The more accurate is the tool wear state information obtained, the more likely it is that the replacement policy will be close to optimal. The need for timely available reliable tool wear estimation is also evident in machining process control. In such cases, it is usually desirable to optimise the cutting process by means of adequately adjusting process parameters such as the feed rate, while securing at the same time the stable operation of the machine tool. The optimisation is based on information directly or indirectly related to the actual tool wear. However, appropriate control action can only be implemented on the basis of reliable tool wear modelling.

Clearly, a wide range of machining process optimisation problems depend on tool wear and tool life modelling. Before, addressing in more detail the tool wear/tool life modelling problem for single point cutting tools, some basic definitions and theoretical issues related to tool wear will be briefly mentioned in the following paragraphs.

5.3 Tool wear: definitions and theory

Any machining process, involving single-point or multi-point cutting tools, apart from producing the desired workpiece shape and surface finish, inevitably results in some change on the shape of the cutting tool itself. This change varies with different combinations of cutting conditions, workpiece material and cutting tool. The shape of the cutting edge may be altered either as a result of plastic deformation or of wear. What distinguishes them is that a wear process always involves some material removal, whereas no such removal is observed in cases of plastic deformation. In both cases, when the cutting edge is so severely worn that the desired operation can not be successfully carried out anymore, the cutting edge is considered to have reached the end of its useful life and the total effective cutting time of the edge is referred to as tool life. A cutting tool may reach the end of its useful life either as a result of a gradual wear procedure or of a sudden - catastrophic - failure.

5.3.1 Mechanisms of wear

Tool wear can be caused by several mechanisms. The exact way in which these mechanisms act independently or in synergy to attack the tool shape is still a subject of scientific research.[Mari and Gonseth 1995, El Wardany and Elbestawi 1997]. The main wear mechanisms according to Trent [Trent 1991] are plastic deformation under compressive stress or by shear at high temperatures, diffusion wear, attrition wear, abrasive wear, thermal fatigue and wear under sliding conditions. Other classifications can also be found in the literature, categorising tool wear into adhesive, abrasive, diffusion wear and fatigue [Shaw 1984]. The following summary of wear mechanisms is mainly based on Trent's analysis [Trent 1991], where tool wear is examined from a metallurgy and materials engineering point of view, supported by extensive, up to date, experimental results.

5.3.1.1 Plastic deformation

The shape of the cutting tool can be altered due to high temperatures or high compressive stresses on the rake face. The temperature at the rake face of the tool can reach at very high levels which may result at the formation of a hollow on the tool rake surface in some distance from the cutting edge. This short of deformation may appear when cutting at high speeds with tool materials such as high speed steels. Carbide tools are more resistant to this type of deformation but they are still susceptible to plastic deformation under high compressive stresses at temperatures higher than 800°C [Mari and Gonseth 1993]. Plastic deformation is more common when cutting difficult to machine materials at high speeds. The deformation usually starts close at the tool nose, so tools with very small nose radius are more susceptible to it.

5.3.1.2 Diffusion wear

Diffusion wear refers to the exchange of atoms that may take place at the workpiece/tool interface when cutting at high temperatures. In particular, tool metal or carbon atoms may be carried away with the workpiece material stream removed or workpiece material atoms can be diffused into or react with the tool material, altering its surface properties.

Diffusion wear is more significant when machining with carbide tools, whereas in high speed steel tools is usually masked by the more sizeable plastic deformation that takes place at high temperatures. It can lead to accelerated crater wear.

5.3.1.3 Attrition wear

At low cutting speeds the temperature does not rise high enough to bring about diffusion wear or plastic deformation. The work material flow over the cutting edge is more intermittent and whole parts of the removed material may adhere to the tool rake forming strong bonds with the tool material and leading to the formation of a built-up edge. The shape and size of the built-up edge constantly changes during cutting and the tool/workpiece interface may become discontinuous, which in turn may result in fragments of the tool surface being sheared away from the tool, carrying along with them tool material particles. However, in certain cases, such as when cutting cast iron, the presence of an adhered layer of workpiece material particles may provide protective action for the tool material. On the other hand, when machining steels, large grains of tool material can be removed from the tool surface and bring the cutting edge to a premature failure. Attrition wear is more frequent in carbide tools, whereas high speed tools can be tougher and less vulnerable to it.

5.3.1.4 Abrasive wear

Abrasive wear is caused by the presence of hard material particles along the tool/workpiece interface, which may be contained in the workpiece material or formed by chemical reaction (e.g. oxidation). It is widely considered as a major cause of wear, though its contribution to tool deformation under normal cutting conditions is sometimes questioned [Trent 1991]. However, it is generally agreed that wear by abrasion can play an important role under sliding conditions.

5.3.1.5 Thermal fatigue

This sort of wear appears mostly in cases of discontinuous cutting, like in milling, where the tool material undergoes a series of thermal shocks. The successive expansion and contraction of the tool surface layer that is close to the cutting edge can cause small cracks which may start from the rake face and gradually expand towards the cutting edge. When the number of cracks becomes significant, relatively large fragments of tool material can be detached from the tool surface. Thermal fatigue generally acts additively to other wear causes and reduces the tool resistance to fracture.

5.3.1.6 Wear under sliding conditions

This is the type of wear that occurs at those areas of the tool surface where sliding occurs, i.e. they are not continuously engaged at the cutting process. The wear mechanisms involved can be the same with those described so far, but they can be greatly accelerated by electrochemical interaction between the cutting tool, the workpiece material and the environment. The latter refers both to the cutting coolants and lubricants, which are used to assist cutting by reducing the temperature and the cutting forced developed, as well as to the presence of air. This electrochemical interaction can be very complex and is not yet fully understood [Trent 1991]. It can result to significant alteration of the tool sliding surface properties by allowing the formation of a weak layer at the face of the tool. This result in reduced shear strength and increased susceptibility to other mechanisms of wear.

The above mentioned wear mechanisms are rarely independently activated. Plastic deformation and diffusion wear are generally thermally initiated, when cutting at high speeds, in contrast with attrition wear, which is not temperature dependent and holds a dominant role at low cutting speeds. Depending on the type of operation, sliding wear processes and fatigue wear can become significant wear causes, whereas abrasive wear is less important when cutting with high strength tool materials.

5.3.2 Types of wear

As a tool is used in cutting operations, its shape gradually changes due to wear. When the main cutting process states, such as cutting forces, power consumption and vibration reach higher levels, the cutting performance deteriorates. This may result in noticeable loss of dimensional accuracy and poorer surface finish of the machined part. The cutting process state increasingly deviates from the desired set point and the process stability may be put into jeopardy. Tool wear develops at different faces of the cutting tool and is classified into several types, depending on its form (Fig. 5.1-5.2).

5.3.2.1 Flank wear

Flank wear occurs at the clearance or relief face of the cutting tool along the tool surface that is engaged on the cutting process. It may be formed as the result of diffusion, attrition and abrasive wear, as well as wear under sliding conditions.

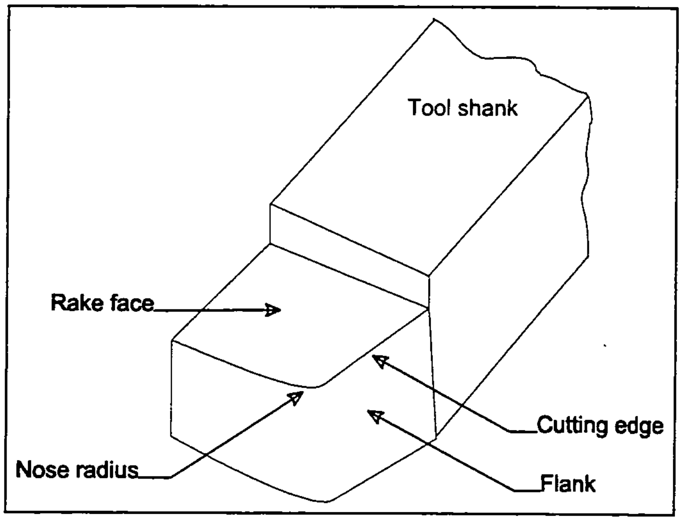

Figure 5-1: Cutting Tool Geometry

5.3.2.2 Crater wear

When cutting at high temperatures, diffusion wear can cause the formation of a depression or crater at the rake face of the cutting tool. This crater is deeper at the position that the highest temperatures appear at some distance from the cutting edge, whereas the tool may stay unworn at low temperature positions. Severe cratering results in weakened cutting edge and increased susceptibility to fracture.

5.3.2.3 Plastic deformation

Plastic deformation as a result of high compressive stresses or high temperatures can significantly alter the shape of the cutting edge. Cutting performance deteriorates as the cutting edge geometry gradually changes and increasingly high forces and temperatures are developed. Significant plastic deformation of the cutting edge can reduce its resistance to fracture.

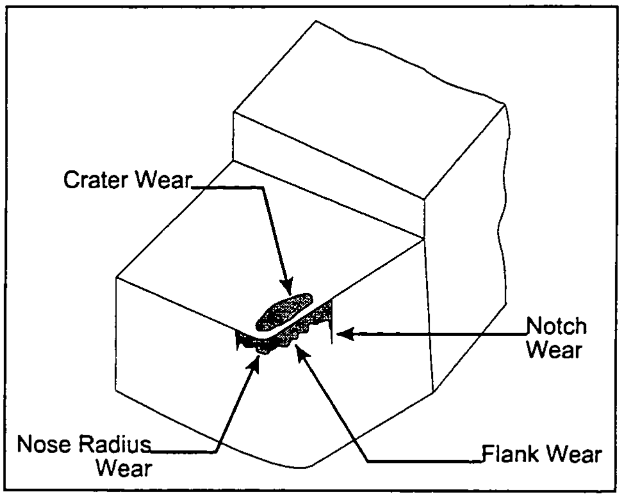

Figure 5-2: Principal types of wear

5.3.2.4 Notch wear

It is also known as groove wear. Notch wear usually appears at the end of the flank wear land, where the chip edge moves over the cutting tool. It is generally considered to result from chemical interaction at the workpiece/tool interface and is associated with the presence of oxygen at a position where seizure is discontinuous. The freshly machined surface is chemically very active especially under the presence of oxygen that penetrates the workpiece/tool interface up to a limited depth after which seizure is continuous. In the small intermediate area, where sliding occurs, it is likely that the surface of the tool presents some local anomalies either because it is contaminated by oxides or by work hardening from previous cuts. As the workpiece material flows over the cutting edge the cutting forces and temperatures developed may reach at high levels at the positions where such anomalies exist and whole fragments of the tool contaminated surface may be detached and carried away with the chip. Thus, deep grooves or notches may be formed on the tool surface which can eventually lead to tool failure.

5.3.2.5 Built-up edge

When cutting at low or intermediate cutting speeds, work hardened workpiece material fragments may adhere on the rake face of the tool and towards the cutting edge, forming strong bonds with the tool material. This leads to the formation of a built-up layer on the rake face of the tool. When the size of this layer becomes significant is referred to as built-up edge (BUE). The BUE acts as a natural extension to the cutting tool and as long as it is just a thin layer, it can play a protective role, preventing tool wear at the rake face. However, when it grows significantly in size, its structure becomes increasingly unstable. The locally increased shear strength can cause micro-cracks which may further develop and cause fracture of the BUE. Since the adhered to the tool surface work material has formed strong bonds with the tool material, when detached from the rake face can shear away with it small fragments of tool material, altering the shape of the tool surface. This can lead to poorer surface finish of the machined surface, edge chipping and finally fracture of the cutting edge.

5.3.2.6 Edge chipping

Edge chipping is the break-off of small fragments from the cutting edge which may take place in interrupted cutting with brittle materials or when an unstable BUE is periodically cracked away. The result of edge chipping is poorer surface finish of the machined workpiece and increased susceptibility to cutting edge fracture.

5.3.2.7 Edge cracking

This is usually the result of thermal fatigue and it appears as a series of parallel or perpendicular to the cutting edge small cracks. Weakened cutting edge and increased risk of tool failure can be the consequence of edge cracking.

5.3.2.8 Edge rounding

Cutting edge rounding or nose wear occurs as a result of attrition or abrasion wear and reduces the cutting efficiency of the tool. When nose wear is excessive, the energy is not spent on actual cutting action but instead on tool plastic or elastic tool deformation. The grinding of a double rake on a high hardness cutting tool prevents nose rounding by allowing the formation of a stable BUE.

5.3.2.9 Tool breakage (Catastrophic failure)

Catastrophic failure may occur as the result of accumulated tool wear of all the above types, or of the selection of inadequate cutting conditions. Brittle materials like ceramics or cemented carbides are more exposed to fracture hazards.

5.4 Tool wear measurement

Tool wear measurement methods are broadly classified into **off-line** and **on-line** ones. The former are only applicable after the cutting tool is disengaged from the cutting process, whereas the latter are designated to allow for in-process identification of tool wear state. Sometimes the distinction is between continuous and intermittent measurement methods [Byrne et al. 1995]. Continuous measurements are those on-line measurement techniques that enable the capture of all the changes that the measured variable undergoes, including cases of sudden and unexpected disturbances, without any significant information loss. In contrast, intermittent measurements are methods that involve either interruptions of the machining process or specific measurement intervals, which inevitably results in some loss of information. By definition, all the off-line methods are intermittent measurement methods, while the latter ones include also some on-line methods that are not capable of continuously monitoring the tool wear process.

Tool wear measurements methods are also categorised as **direct** or **indirect**, depending on whether tool wear is directly measured or estimated indirectly by inference from other measured variables which are correlated with tool wear, such as cutting forces, acoustic emission, vibration etc [Park and Ulsoy 1993]. Since it is usually very complicated to implement on-line direct methods for tool wear measurements, the term direct measurement is often confounded with off-line measurement. However, there should be a distinction between them, since an off-line method, can be employed for the indirect inference of the actual tool wear status, i.e. it can be an indirect measurement method, without being an on-line one. For instance, tool wear status inference by observing the machined workpiece surface quality is an off-line indirect measurement method.

Evidently, a reliable direct wear measurement sensor should be able to posses higher measurement accuracy in contrast with indirect methods, which rely on inferential estimation of the actual wear status on the basis of auxiliary variables sensorial feedback. The ideal tool wear sensor in terms of both accuracy and time response would be one that is simultaneously direct and on-line. However, the nature and complexity of the cutting process is such, that direct on-line sensors are rarely applicable in practice. A review of wear measurement techniques was carried out and can be found in Appendix A.

5.5 Tool life modelling

The complexity of tool wear processes and tool wear measurement methods is inevitably reflected into tool wear modelling and tool life prediction problems. Admittedly, there is a plethora of parameters influencing they way tool wear evolves during cutting. Tool wear modelling can be either predictive (off-line) or real-time (on-line) [Maropoulos and Alamin 1996]. Predictive tool wear modelling is relevant to off-line process optimisation activities, such as optimal selection of cutting conditions and preventive tool replacement strategy determination. On the other hand, on-line tool wear modelling applies to tasks such as machining process monitoring and control or to real-time tool replacement policy decision making, based on actual tool wear status.

Depending on the viewpoint adopted, tool wear and tool life modelling problems present various difficulties. The main difficulty in the case of off-line modelling is the high degree of uncertainty that surrounds the tool wear evolution for each individual cutting operation and results to significant tool life variability. When tool wear status information is available, more accurate on-line tool wear identification and real-time tool life prediction can be achieved without the need for the availability of analytical theoretical or empirical tool life models. However, real time tool wear status estimation usually suffers from the high complexity of the instrumentation and signal processing involved. Further difficulties arise from the poor reliability of the on-line sensors and the lack of knowledge about the exact way the measured signals correlate with the actual tool wear. Because of the above reasons, little commercial interest has been shown to on-line process optimisation, based on real time tool-wear estimation [Maropoulos 1995]. Instead, machining process optimisation decisions are usually based on off-line tool life prediction.

5.5.1 Tool failure criteria

According to ISO recommendations [ISO 3685, 1993, 2nd edition, "Tool life testing with single-point turning tools"], a turning tool is considered to have reached the end of its useful life when the following criteria are met:

- High speed or ceramic tools

1. Catastrophic failure.
2. V_B=0.3mm, if the flank wear is regularly worn.
3. V_{Bmax} = 0.6 mm, for unevenly worn flank, scratched, chipped or badly grooved.

- Sintered carbide tools

1. V_B=0.3mm.
2. V_{Bmax} = 0.6mm if the flank is irregularly worn.
3. K_T=0.06+0.3f, where s is the feed rate.

Where V_B and V_{Bmax} are the mean width and the maximum width of the flank wear respectively, K_T is the crater wear and f is the feed rate. The following section briefly reviews current methods for off-line tool life modelling.

5.5.2 Off-line tool life modelling

Off-line tool life modelling is commonly based on cutting process theory, tool life empirical formulae and tool-life data provided by tool manufacturers and machining handbooks as well as laboratory experiments and shop floor cutting operations. A wide range of mathematical modelling and data handling methods have been employed for tool-life modelling. These methods provide with different frameworks for utilising the available knowledge about the performance of the cutting process.

5.5.2.1 Taylor formula

Most of the models employed for tool-life prediction are either extensions of or consistent with the well-known Taylor empirical formula which correlates tool life with cutting speed. Taylor's work [Taylor 1906] is considered as the first systematic treatment of the tool wear and tool life problem with a view to the machining economics optimisation problem. The basic formula is:

$$v^{-1/a}T = C \tag{5-3}$$

where v is the cutting speed (m/min) and T is the tool life (min), while a and C are constants for a particular tool-workpiece combination. This formula was later extended to accommodate some more parameters influencing tool life:

$$T = \frac{C}{v^{\frac{1}{\alpha}} f^{\frac{1}{\beta}} d^{\frac{1}{\gamma}}} \tag{5-4}$$

where f denotes the feed rate (mm/rev), d the depth of cut and α, β, γ are constants depending on the tool/workpiece/type of cut combination. Empirical formulae of the above form have been very popular, since they provide with a simple means of correlating tool life with machining conditions. Even though tool life modelling have moved a lot further from Taylor's formulation, many of the suggestions in literature for tool life models appear to be - up to a certain degree - consistent with the above formulae.

5.5.2.2 Models based on cutting theory

Cutting process theory may provide the means for obtaining either direct analytical models for tool life prediction or - most commonly - for switching over between different tool life models, by providing a physically meaningful way of determining the applicability range of the models. However, it is often difficult to assert the validity of the theoretical tool life models for a wide range of combination of tools, workpiece and cutting conditions, since tool wear evolves at different tool faces and it is rarely the case that this is due to a single wear mechanism. It has been suggested that the mechanical behaviour of tungsten carbide tools can be classified into three main domains, depending on the cutting temperatures developed [Mari and Gonseth, 1993]. In particular, WC-Co is brittle below 500°C, tough between 500°C and 800°C and susceptible to plastic deformation above 800°C. Based on this remark and the assumption that there is a direct relationship between the mechanical energy related to the deterioration of the cutting tool geometry and the tool wear, the following formula has been suggested for tool life when cutting at temperatures higher than 800°C, which is usually the case with carbide tools [Mari and Gonseth, 1993]:

$$T = \frac{E_c}{\alpha V + \beta e^{-Q/k\vartheta}} \tag{5-5}$$

where E_c (J) represents the critical energy required to obtain a given level of flank wear, k is the Boltzmann constant, Q the activation energy (eV/atom), θ the cutting temperature, while α, β are constants related to the friction force and the stress respectively. This above formulation assumes that the level of the critical energy E_c for a certain amount of deformation is constant. The above equation was then modified in order to model also the effect of oxidation at lower temperatures:

$$T = \frac{E_c}{\alpha V + \beta_1 e^{-Q_1/k\vartheta} + \beta_2 e^{-Q_2/k\vartheta}} \tag{5-6}$$

where the notation is similar with equation (5-5) apart from the fact that two distinct thermally activated mechanisms are now defined, acting at different temperature levels and characterised by different activation energies. The models of equations (5-5, 5-6) were found [Mari and Gonseth, 1993] to give a good description of the wear process when plastic deformation or oxidation (wear under sliding conditions) are the main forms of wear. However, they are highly parametrised and their applicability in a practical environment is restricted by the need to conduct extensive experimentation and statistical data processing in order to define appropriate values for the parameters involved.

The fact that certain wear mechanisms are thermally activated has led to the development of a tool life model which directly relates tool life with cutting temperatures and is less parametrised than equations (5-5) and (5-6):

$$T = AT_F^{-B} \tag{5-7}$$

where T_F is the tool flank temperatures and A, B constants [Arsecularatne et al. 1996]. The above model can give good description of the wear phenomena at cutting temperatures higher than 800°C. It relies on analytical prediction of cutting temperatures [Arsecularatne et al., 1995] and it can be applied to both orthogonal and oblique cutting. The validity of the model depends on the level of accuracy of cutting temperatures prediction, as well as the determination of well-defined cutting operations range, wherein A and B can be considered as constants to be calculated by appropriate experimentation and statistical data processing.

The flank wear of carbide tools, cutting carbon steels, can also be analytically predicted using cutting forces and temperature, which are also predicted analytically [Usui et al. 1984]. The overall calculation is based on the derivation of a wear characteristic equation which correlates cutting temperature and normal stress on the flank wear land with the wear rate. Stress and temperature calculations are based on an energy method for predicting chip formation and cutting forces from orthogonal cutting data. The overall calculation procedure is rather complex and, on the basis of some rather crude assumptions, flank wear at time t can be calculated via the following equation [Matsumura et al. 1993]:

$$V_B = V_{B_0} + \int_0^t \frac{C v \sigma_f e^{-\frac{\lambda}{\vartheta_f}}}{\tan\gamma} dt \tag{5-8}$$

where V_B (mm) is the flank wear length, V_{B_0} (mm) is the initial flank wear, γ (rad) the relief angle, v (m/min) the cutting speed, θ_f the temperature of the flank wear land and σ_f the normal stress on the flank wear land. The wear-characteristic constants C, λ are determined via cutting tests. The main assumptions are that there is no appreciable initial wear, and a series of physical parameters, such as the strain rate on the chip surface and the specific wear rate over the wear land, are approximately constant. However, these and some other assumptions that have been made for the derivation of the above equation, only hold for a limited range of cutting conditions and refer to steady state wear rates.

5.5.2.3 Empirical tool life modelling

Analytical tool life models based on metal cutting theory result from the attempt to explain on a physical basis the mechanisms of tool wear. In most cases, complex formulations for tool life as a function of many variables are derived which are not easily applicable to machining optimisation practice. Furthermore, these models are rather incomplete, since they are obtained as a result of many assumptions and therefore their validity holds under restricted cutting conditions. In practical operating conditions, however, the machining process can not be easily controlled to ensure that all these preconditions are met. A different viewpoint adopted by many researchers and machining practitioners is based on endorsing some simple empirical models for tool life. These models are usually parametrised, so that there is a need to define the free parameters for a range of inputs, i.e. cutting conditions and tool/workpiece combinations. The most popular empirical models are the Taylor tool life formula of equation (5-3) and some variations of it, such as equation (5-4). A model that incorporates flank wear and nose wear into a single formulation for average tool wear versus time is [Billatos et al. 1986]:

$$W(t) = a_o + a_1 t^{1/2} + a_2 t^{3/2} \qquad (5\text{-}9)$$

where $W(t)$ is the accumulated average tool wear at the time instant t, while the coefficients a_1 and a_2 correspond to nose wear and flank wear respectively and are to be estimated, together with a_o, based on available cutting data for a range of cutting conditions and tool/workpiece combinations. Another model that relates the total amount of tool wear with cutting time has also been proposed. The model considers the cutting conditions as well as the total amount of flank wear as independent variables [Nagasaka and Hasimoto 1982]:

$$t = av^{n_1} f^{n_2} \exp\left[-\exp(b)V_B\right] \tag{5-10}$$

where t is the cutting time that results in flank wear of length V_B, v the cutting speed, f the feed rate and a, b, n, n_1, n_2 constants to be estimated. Tool life can be derived from equations (5-9), (5-10) for any given level of accumulated wear. Alternatively, tool life can be modelled according to:

$$T = \frac{C_T}{v^a h_e^m}, \quad a = a(h_e, v) \quad m = m(h_e, v) \tag{5-11}$$

where v is the cutting speed, h_e is called Woxen chip thickness and includes in one parameter the feed rate, nose radius, depth of cut and the cutting edge angle [Carlsson and Strand 1992]. The resemblance of the above equation to Taylor's formula (5-4) is obvious.

The free parameters (i.e. the constants) of the empirical tool life models described so far, are determined on the basis of available cutting data. In particular, initial estimates for these parameters are obtained based on machining data derived from machining handbooks or tool manufacturers. Once observed real-life cutting data become available, they are combined with the initial cutting data to obtain posterior (updated) tool life parameters, that best fit to the model adopted using appropriate regression methods, such as least squares estimation [Ermer 1970], or linear multiple regression [Yeo et al. 1989].

The main drawback of such methods, when employed to solve the machining economics problem, is that they treat the empirical formulae in a deterministic manner, without taking into account the inherent uncertainty about the evolution of the cutting process. There have been suggested several different approaches for tackling this uncertainty, but the most prevailing one considers tool life as a stochastic variable having a mean value properly represented by the empirical deterministic tool life equation adopted.

5.5.2.4 Probabilistic tool life modelling

Accepting that tool life is a random variable, brings another problem into consideration. That is the determination of tool life variability and tool life distribution. The benefit of taking into account tool life scatter becomes more significant as tool life variability increases. It has been argued that distributed tool life is responsible for increased machining costs to as high as 50% in some cases [Iakovou et al. 1996]. Tool life scatter originates from the large number of parameters influencing the wear process.

In a manufacturing environment the resources employed for production come from various sources and their quality and performance are subject to variation. For instance, the tools, fixtures and machine tools used may differ in stability and rigidity, resulting in significant deviation in machining performance. Tool life tests are often conducted under restricted conditions with minimum tool life variability. However, the statistical properties obtained from such tests reflect only the particular characteristics of the machine/tool/workpiece combination of the experiment. In practice, the machining data generated at any two different machining laboratories do not agree. This is even more true for shop floor conditions, where it is often difficult to control the cutting process, ensuring consistency of the cutting performance.

The experimental work on tool life testing carried out so far provides with enough evidence of tool life scatter. The size of tool life variation strongly depends on the nature of the experiments. For instance, it has been suggested that, for the same combination of tool/workpiece/type of cut, a standard deviation as high of 20% [Ramalingam 1982] or even 30% [Wager and Barash 1971] of the mean tool life value should be anticipated. Other studies have resulted in lower variation coefficient values - between 7% and 9% -for tool life distribution of ceramic tools [El Wardany and Elbestawi 1997]. Even for log-transformed tool life values, extensive experimentation has indicated that a 23% coefficient of variation should be considered for tool life distribution [Levi and Rossetto 1978].

A common simplification in machining practice is to reach optimisation decisions based on the determination of the first two of the tool life moments, i.e. the mean value by the empirical formula adopted and the variance by tool life testing experimentation. However, a more rigorous treatment for optimising the cutting conditions within the machining economics problem would require the determination of the shape of the tool life probability distribution function itself. This requirement has led many researchers to investigate whether tool life follows a characteristic distribution. As a result of significant experimental work many different functions have been suggested for probabilistic tool life modelling, including, normal [Wager and Barash 1971, Koulamas 1991, Zhou et al. 1990], log-normal [Levi and Rossetto 1978, Rossetto and Levi 1978], exponential [Ramalingam and Watson 1977], extreme value [Pandit 1978], Gamma [Ramalingam 1977, Iakovou et al. 1996], Rayleigh [Pandit 1978], Weibull [Billatos and Kendall 1990, Ramalingam and Watson 1977], Birnbaum-Saunders, [Billatos et al. 1986], Frechet distribution [Carlsson et al. 1992] , etc. Other more complex probability distribution functions have also been suggested which can approximate a range of functions, depending on the particular choice of some parameters. The determination of these parameters and consequently of the shape of the tool life probability distribution function should be based on available cutting data [Pandit 1978]. The more flexible such a function is in terms of its capability to represent observed tool life data, the more difficult usually becomes the task of determining the function parameters. Thus, a simple case of normally or log-normally distributed tool-life is usually considered.

5.5.2.5 Reliability and tool life modelling

It is often the case that the probabilistic approach to tool life modelling described above relies merely on empirical assumptions and has little physical basis. Indeed, there can hardly be found any theoretical justification for tool life being normally or log-normally distributed. A probabilistic approach that proceeds one step further is the one that does not look simply at tool life as a random variable, but, in doing so, considers tool wear as a stochastic process, where the tool is exposed to a series of hazards. Some of the hazards are of sufficient magnitude to cause irreversible damage to the tool. This approach is adopted when the tool life problem is treated from a reliability point of view [Liu and Makis 1996].

Reliability considerations are particularly beneficial in tool replacement policy decision making, where safe and cost efficient machining operation is pursued. The important considerations in this approach are related to the determination of the potential hazard functions, i.e. the conditional probability of failure within the time interval from t_0 to $t_0 + dt$ when it is known that there was no failure before t_0. This is not a trivial task, since one should identify the nature of the failure mechanisms and attribute to them certain hazard functions to adequately describe the risks that the cutting tool is exposed to. Failure models are required to consider single injury or multiple injury failure cases. These models should account for both combined hazards acting independently or in concert, where provision should be made for time dependent hazards [Ramalingam and Watson 1977, Ramalingam 1977].

One approach is to define the hazard functions by taking into account the tool and workpiece material properties and the operating environment on the basis of metal cutting theory and in particular of stress considerations. [Ramalingam et al. 1978]. Alternatively, hazard functions can intuitively be defined to describe the different causes of failure, such as poor quality control during tool manufacturing, non-homogeneity of workpiece condition, chemical wear that may cause tool chipping and failure due to excessive flank or crater wear [El Wardany and Elbestawi 1997]. Attributing distinct hazard functions to each one of the main failure causes may provide with more detailed tool wear modelling than the previously described simple probabilistic models. Furthermore, such reliability models appear to have stronger physical basis at the expense of increased complexity. Indeed, the computation needed in order to determine the whole set of parameters for each one of the hazard functions defined, is high and in some cases very complex.

5.5.2.6 Random processes and chaotic models of tool wear

In stochastic tool life modelling there is usually no clear way in which the cutting parameters are presented in the model. It is often observed that tool wear progress patterns may vary even under rather similar conditions. On the other hand, it is very difficult to associate tool life with the cutting conditions in a deterministic formulation using physical laws. The reason behind this difficulty is the inherent complexity and uncertainty of the cutting process. It has been argued that part of this complexity is due to the chaotic behaviour of the process in some circumstances. In particular, cutting process may show signs of irregular chatter, especially at high feed rates. The occurrence of chaotic oscillations during cutting has been theoretically derived by examining a two-dimensional dynamic model based on the geometry of orthogonal cutting. [Lin and Weng 1991]. The potential chaotic behaviour of the cutting process had been earlier suggested by Grabec [Grabec 1986]. Experimental results have also been found to verify that feed rate is a critical parameter that may lead the cutting process into chaotic behaviour [Khraisheh et al. 1995].

To overcome these difficulties a different stochastic model, i.e. a diffusion-threshold tool wear model has been developed with applicability to the tool replacement policy problem for drilling [Conrad and McClamroch 1987]. Accumulated tool wear is modelled by a diffusion process (Wiener process or Brownian motion), with a piecewise constant drift and piecewise linear variance. Diffusion processes are an important class of Markov processes with important applications in physics, population dynamics, genetics etc. [Gardner 1986]. The main benefit of this modelling approach is that it takes into account the potential chaotic nature of the cutting process, since Brownian motion is often a good first approximation for systems that exhibit chaotic or noisy behaviour. An equivalence between this model and Taylor's tool life formula has also been derived. Two classes of tool replacement policy have been postulated, based on an optimal control formulation. These classes are age replacement policies (i.e. based on machined parts count) and one step ahead replacement policies based on wear status information.Yet, it is questionable whether the wear process can indeed be modelled by such a Markov process. For instance, material properties may depend on past properties and not only on current conditions. However, any deviation from the Markov assumption would make the problem mathematically hardly tractable. In addition, the wear process often present some "jump" phenomena (e.g. catastrophic failure) that are not consistent with the continuity assumption of a diffusion process.

5.5.2.7 Computational intelligence and tool life modelling

Following the recent emergence of computational intelligence techniques, there have been a few reported attempts at neural network based tool life modelling. In particular, a feed forward neural network model has been proposed for tool life modelling based on a data set of 60 input/output patterns [Narayanan 1995]. The input patterns consist of cutting conditions combinations, i.e. cutting speed, feed rate and depth of cut while the output pattern is the observed tool life when turning Nitralloy 135 with a carbide tool. An average predictive accuracy of 6.2% is reported. A similar four-layered feed forward network has also been developed for turning grey cast iron (grey G-14) with a mixed-oxide ceramic cutting tool (type K090) [Ezugwu et al. 1995]. The network consists of one input layer with two input nodes, two hidden layers with 16 nodes per layer and a four nodes output layer. The input nodes are fed with the cutting speed and feed rate values, while the output nodes provide with a prediction for the tool life as well as the failure mode, i.e. excessive flank wear, catastrophic failure, poor surface finish. The best results obtained were 58.3% correct tool life prediction (within the 20% of the actual tool life) and 87.5% correct failure mode prediction. Even though these models are indicative of the mapping capabilities of multi-layer perceptron type neural networks for tool life modelling, they are very specialised and provide predictions for restricted cutting conditions and for a specific combination of tool/workpiece material.

A fuzzy model of tool life has also been developed on the basis of the development of a machining reference database from representative finish turning experiments [Fang and Jawahir 1994]. In particular the cutting data for turning work material AISI 4140 (high carbon steel) with an ISO P40 flat-faced tool has been used in order to develop fuzzy set mathematical models which quantitatively describe the machining performance. The fuzzy tool life model derived takes into account 8 inputs, e.g. cutting speed, feed rate, depth of cut, normal rake angle, inclination angle, tool cutting edge angle, nose radius and the workpiece hardness. The fuzzy model average predictive performance was 6.39% for major flank wear, 7.80% for crater wear and 7.72% for minor flank wear. However, the determination of the membership function parameters was based on rather intuitive criteria and there was no provision for optimisation on the basis of the available cutting data.

In the next two chapters a neural-fuzzy systems approach to tool life prediction is presented. This method is non-parametric in the statistical sense, i.e. it does not rely on any statistical assumption about the underlying tool life distribution. Neither does it depend on the validity of any tool life empirical formula, or theoretical knowledge of the wear process, since it can be completely data driven. Yet, it is capable of incorporating existing a priori knowledge about the cutting process and is easily implementable, thereby providing an attractive alternative tool life modelling approach.

NEUROFUZZY TOOL LIFE MODELLING

Structure Identification

6.1 Introduction

Tool life modelling is an essential part of detailed process modelling for a range of process planning activities. The determination of optimal tool replacement policies, as well as the optimisation of the cutting conditions are based on the availability of reliable models which can provide accurate tool life prediction for all the combinations of tool, workpiece and type of cut. Yet, it is very difficult to achieve reasonable accuracy in predicting tool life, without increasing significantly the complexity of the model and therefore of the computation involved. Furthermore, increased precision is usually achieved at the expense of the range of the applicability of the model. Thus, the validity of the model is often restricted within a small, predefined region of cutting conditions and workpiece/tool combinations.

It is rather non practical, to specify very high off-line tool life modelling accuracy requirements, since off-line modelling can not always foresee for wear phenomena occurring during the real-time operation. For example, the precise moment when a built-up-edge is detached from the tool rake face can not be predicted beforehand. In fact, even when in-process tool condition monitoring is employed, still such a task is not trivial. If considerable amount of the tool surface material is removed together with the built up edge, the tool behaviour in terms of wear resistance may dramatically change, as a result of significant weakening of the cutting edge. From that moment onwards, tool wear will progress in a different way and this can not be explicitly considered in the case of off-line tool life modelling. Generally speaking, tool wear alters the original tool geometry, resulting in different and often unexpected cutting performance. Therefore, effective tool life prediction should be pursued both off-line and on-line, thus taking into account not only the conditions under unworn tools but also those developing during the actual cutting process [Fang 1994]. Consequently, any effort to establish a very accurate off-line tool life model can hardly prove to be cost efficient. Instead, tool life modelling should seek to exploit the tolerance for imprecision and uncertainty, in order to balance simplicity with precision. Dealing with this sort of dilemma is at the very heart of the theory of fuzzy sets [Zadeh 1994].

Tool wear progress depends heavily on the particular machine/tool/workpiece combination. The cutting data derived from any two different laboratories or from different shop floors do not usually match in practice. Unless the knowledge that is uniquely relevant to the particular shop floor or laboratory considered can be captured and incorporated into a flexible tool life model, reliable tool life modelling would be very difficult to achieve. Thus, a tool life model suitable for practical application has to be built on the basis of available cutting data. Learning by examples rather than by instruction, is a typical characteristic of artificial neural networks. In view of these requirements for learning by examples, as well as for uncertainty and imprecision handling, this work examines the potential of employing neural-fuzzy methods for tool life modelling. The main objectives of the proposed tool life modelling are:

- To study the main requirements for the specification of a neurofuzzy tool life model.
- To derive a tool life model of reduced complexity.
- To provide a single generic model for a wide variety of combinations of cutting conditions, type of cut, cutting tool and working material.
- To build a model that is essentially data driven, i.e. to come up with a complete formulation for a tool life model, based on available cutting data.

- To accommodate a priori empirical knowledge about the turning process and in particular, results of tool life modelling using linear multiple regression analysis, recently carried out at Durham University [Alamin 1996].
- To obtain a flexible tool life model, which can be easily adjusted on the basis of future tool life data.

The main justification for the adopted neurofuzzy modelling approach is that fuzzy modelling can achieve reduced model complexity and provide a realistic and transparent tool life model, thus facilitating machining optimisation. In addition the requirement for a data-driven tool life model, capable of utilising available cutting data suggests that neural network - like learning capabilities should be incorporated into the model.

The rest of this thesis deals with the specification and development of a neural-fuzzy tool life model for turning operations. The development of the tool life model is carried out following two distinct identification phases, i.e. the structure identification and parameter optimisation. Structure identification involves the determination of the initial fuzzy inference system, including the determination of the inputs partition pattern, the size of the fuzzy rules base, as well as the selection of initial values for the system parameters. Parameter optimisation deals with the adaptation of the initial fuzzy system parameters, based on available cutting data. The neurofuzzy model has been developed using the Fuzzy Logic Toolbox of MATLAB (The Mathworks Inc.). In particular, the Adaptive, Network-based, Fuzzy Inference System (ANFIS) model [Jang 1993] is employed herein for tool life modelling. Apart from the earlier mentioned merits of neurofuzzy models, the choice of the ANFIS structure, is further justified by the following reasons:

1. The ability of specific classes of ANFIS models to behave as universal function approximators has been established [Jang 1993]. Therefore, it is possible to find an appropriate ANFIS model for tool life, provided that the quality of the available data is high. Within the context of system identification, sample data is considered to be of high quality when exhibits the following characteristics :

    - The available data is not too contaminated by noise or other unmodelled phenomena.

- The data set is complete (persistently exiting) in the sense that it contains sufficient information for all the range of values of the input variables.

Evidently, a completely data-driven model can not be of high quality if not based on rich data. The data availability issue will be further discussed in the next section.

2. ANFIS models have already found numerous engineering applications and have been tested on nonlinear system identification, chaotic time series prediction, adaptive noise cancellation, printed character recognition, automobile fuel consumption prediction etc [Jang, et al. 1997]. The nature of the fuel consumption prediction problem carries some similarity with the tool life prediction problem. In both cases a nonlinear model has to be built on the basis of available input - output pairs, where a mixture of continuous and multi-valued input variables is present.
3. Efficient algorithms for ANFIS model structure [Chiu 1994, Chiu 1996, Jang 1994] and parameter identification [Jang 1993, Jang and Mizutani 1996] based on available data have been developed.
4. The ANFIS architecture is directly supported by the MATLAB Fuzzy Logic Toolbox [Jang and Gulley 1995]. MATLAB is probably the most established software platform for technical computing.

The rest of this chapter deals with the structure identification problem for neurofuzzy tool life modelling. Issues such as inputs selection, data availability, input space partitioning as well as fuzzy rule base initial structure identification are addressed in the following sections. Some initialisation results are also presented followed by discussion.

6. 2 Tool life model structure identification design issues

The model structure identification involves the selection of relevant input and output variables, the determination of the structure of the fuzzy inference system, including the fuzzy rule base construction and the choice of initial premise and consequent parts of the fuzzy rules. Evidently, tool life is identified as the single output of the neurofuzzy model.

The input variables choice depends on the parameters influencing tool wear and the data availability. These issues are examined next.

6.2.1 Parameters influencing tool wear

Tool wear has been a subject for research for the last century or so. However, there has not been yet established a solid and unified theoretical framework that fully explains and accurately predicts the precise way in which tools wear [Mari and Gonseth 1995, Wardany and Elbestawi 1997, Arsecularatne et al. 1996, Trent 1991]. Nevertheless, tool life is known to be determined by a variety of factors related to cutting conditions, tool and workpiece material properties and tool/workpiece/machine tool combinations.

6.2.1.1 Cutting conditions

The relationship between cutting conditions, i.e. cutting speed, feed rate and depth of cut and tool life has long been investigated. The more aggressive the cutting conditions are the shorter is the expected tool life. Among cutting conditions parameters, cutting speed is generally considered as the predominant factor causing tool wear. High cutting speeds can bring about increased vibration, which in turn may affect the machine tool operation by creating defects on some of its components, such as bearings. Increase in feed rate also results in higher wear rates, but to a lesser extent than higher cutting velocity. Tool life is less sensitive to increase in depth of cut. However, the allowable increase in depth of cut is limited by a number of factors, such as the machine tool capability, the amount of metal to be removed, tooling capability, surface finish and occasionally by the shape of the workpiece [Alamin 1996].

6.2.1.2 Tool material properties

Tool wear progress depends heavily on the material chemical composition. In particular, important tool properties are physical and chemical stability at high temperatures and resistance to wear (toughness) and fracture (hardness).

6.2.1.3 Cutting tool geometry

A number of geometrical features related to the cutting tool give rise to different wear progress patterns. The main geometrical considerations related to tool wear are the tool rake angle, clearance or relief angle, entering or approach angle, inclination angle, nose radius, cutting edge angle and type of chipbreaker . An increase in the rake angle of the tool improves cutting efficiency but results in reduced tool life. Excessive increase in the tool rake angle weakens the cutting edge and entails higher hazards for tool fracture. A negative rake angle can be used to give greater strength to the cutting edge in occasions whenever this is needed to reduce tool failure risks. The tool wear rate is also reduced with even a small increase in the clearance angle. This, however, applies only to small angle values, since it is not possible to further increase the clearance angle without posing serious threat to the cutting edge strength. A slight increase in nose radius is generally beneficial for tool life, reducing the wear mainly to the minor but also to the major flank [Fang and Jawahir 1994]. On the other hand, the nose radius cannot be excessively increased without affecting the chip breakability and cutting efficiency. The choice of inclination angle mainly affects the magnitude of the cutting forces developed, whereas the significance of the cutting edge angle is more related to the surface finish achieved, the cutting forces developed and the chip breaking than directly to tool life.

6.2.1.4 Workpiece material properties

These are generally described by the qualitative term workpiece machinability. Of particular importance for tool life considerations are the material hardness, specific heat, tensile strength, as well as the finished surface of the workpiece.

6.2.1.5 Tool/workpiece/machine tool interface properties

There is a plethora of different factors influencing tool wear which are related to each individual cutting tool, workpiece and machine tool combination. These include the relative hardness of tool and workpiece materials, the chemical compatibility between tool and workpiece materials, the presence of abrasive particles such as scale layers on the surface of the workpiece, the condition (e.g. rigidity) of the machine tool, the condition of the cutting tool in terms of existing wear and the presence of lubricants and coolants.

6.2.2 Inputs selection for data-driven tool life modelling

One fundamental requirement for the tool life model is that it should be data driven. In practice, every cutting operation on the shop floor yields a set of approved cutting data which hold valuable information about the cutting performance for the particular combination of machine, tool and workpiece. It is crucial that this knowledge is exploited and not discarded, since it may help improving future operations. The significance of such approved feedback information has been recognised and provision for its utilisation is a key feature in various prototype systems recently developed at Durham University, including Computer Aided Process Planning systems for Turning [Maropoulos 1992, Maropoulos and Gill 1995], Tool Life Prediction and Management system for Turning [Alamin 1996, Maropoulos and Alamin 1996] and Machinability Assessment and Tool Selection for Milling [Carpenter 1996].

It should be noted that for a data driven model the input selection problem depends heavily on data availability. The main difficulty is related to the quality of the data, in terms of the richness of the provided information. If the available set of cutting data is only partially exciting, then the resulting model may be overtrained is some regions, while it may provide with inaccurate modelling in the areas where the relevant learning information was insufficient. Therefore, there is a need to examine all potential sources of training data to decide upon the learning strategy to follow.

6.2.2.1 Data availability

The main sources of cutting data are machining data handbooks [Metcut Research Associates 1980], tool manufacturer's handbooks, laboratory experiments, shop floor data and simulated data from existing models.

Real Life Data: Real life data can be obtained from the shop floor or by utilising a laboratory machine tool as a test rig of cutting experiments for producing machining data. Obtaining a sufficient training cutting data set from laboratory tests or shop floor operations is hardly achievable. It would require extensive experimentation to span the whole range of operations which would be both costly and time consuming.

Machining Handbooks and Tool Catalogues: The tabled data provided by machining handbooks and tool manufacturers include feed rate and cutting speed values for different combinations of workpiece material group and insert grade. The values given correspond to a certain tool life period, usually 10, 15, 20 or 30 min. Tool manufacturers suggest that for different values of intended tool life the cutting speed should be calculated by multiplying the given cutting speed values by empirical factors. Other correction factors may also be provided for tools with various nose radii. The main difficulty in employing data from machining handbooks or cutting tool manufacturers for building an ANFIS model is that this is only indicative data, which does not provide rich information to achieve a dense mapping between the input and output space. A further shortcoming when employing these type of data as training exemplars is that they often represent rather conservative scenarios, in order to minimise the risk to the cutting operation. Therefore, their accuracy in practice is limited. Further data may be derived by extrapolation from available data. However, a linear extrapolation is often inadequate and there is a need for a more realistic procedure for obtaining extra cutting process exemplars.

Simulated Data: Since it is hardly feasible to obtain sufficiently rich real life data, the potential of creating some "artificial" ones, from already existing tool life models has been considered. It would have been unwise to discard already existing tool life models, when it is possible to build on them, in order to achieve enhanced tool life prediction. Improved tool life data can be acquired if machining handbooks or tool manufacturers data is fit into existing tool life models, such as Taylor empirical formulae. Such a formula is the extended Taylor's tool life equation, earlier mentioned in chapter 5 and repeated here for convenience:

$$T = \frac{C}{v^{\frac{1}{\alpha}} f^{\frac{1}{\beta}} d^{\frac{1}{\gamma}}}$$

In the above equation T is the tool life (min), v the cutting speed (m/min), f the feed rate (mm/rev), d the depth of cut and α, β, γ are constants relevant to the tool/workpiece/type of cut combination. Such models offer a rather rough approximation of the actual machining performance. Modelling accuracy can be improved if the free parameters are estimated on the basis of available approved cutting data. Such a tool life model is already available in Durham University, where a prototype system for tool life prediction and management within the context of an integrated tool selection system has been developed [Alamin 1996, Maropoulos and Alamin 1996]. Adopting a knowledge based system approach, the system offers local approximations of tool life as a function of cutting speed and feed rate for each different combination of material group, type of cut (finishing, medium roughing and roughing) and insert grade [Alamin 1996]. Every such combination will be referred to hereafter as a "tool life group". For each tool life group, cutting data provided by tool manufacturers [Seco Tools AB 1993] are utilised to calculate the parameters of the following empirical formula:

$$T = \frac{C}{v^{\frac{1}{\alpha}} f^{\frac{1}{\beta}}} \tag{6-1}$$

where T stands for tool life (min), v for cutting speed (m/min) and f for feed rate (mm/rev), while C, α, β, are free parameters calculated by linear multiple regression. This is a simplification of the extended Taylor formula. The depth of cut is not explicitly present, but is indirectly taken into account in the constant C. Thus, a set of parameters was calculated for each one tool life group. Each set of parameters correspond to a different tool life group, with approximately 300 such groups been defined [Alamin 1996].

The present work utilises the result of the above tool life model in order to derive enhanced cutting data training patterns. The main advantage of this approach is that a rich data set can be derived. In addition, both the data provided by tool manufacturers or machining data handbooks, as well as Taylor's tool life empirical formula in the form of equation (6-1) are taken into account. The drawback of utilising such a cutting data set is related to the validity of equation (6-1) itself, regarding the degree of precision to which the estimated parameters C, α, and β are valid for each tool life group. However, it has already been shown that such a model is capable of providing with tool life predictions of reasonable accuracy in most cases [Alamin 1996]. It should be noted that utilising artificially created data from an already existing tool life model is equivalent with incorporating a priori knowledge for building the neurofuzzy model. Yet, it is not a prerequisite for building such an ANFIS model. Due to its universal approximation properties [Jang 1993] the neurofuzzy model has the capacity of being at least as good a model as the already existing one. An additional significant merit of ANFIS models is their adaptation flexibility on the basis of available training data. Indeed ANFIS models can be easily trained by employing the hybrid learning algorithm discussed in chapter 4, as well as other methods such as the backpropagation [Jang 1997] or the fast "Levenberg-Marquardt" algorithm [Jang and Mizutani, 1996]. Thus, any initial modelling mismatch can be overcome by training the neurofuzzy tool life model with available real life data.

6.2.2.2 Neurofuzzy model inputs

On the basis of the current data availability, five main parameters were identified as inputs to the ANFIS model:

- **Material class**: This is a categorical variable. Two broad categories of materials are considered herein, namely mild and alloy steels and stainless steels. Materials are classified according to the following table [Seco Tools AB 1993]:

Major Material Class	Material Class	Specification
Mild and alloy steels	1	Very soft, low carbon steels
	2	Free-cutting steels
	3	Structural steels, ordinary carbon steels
	4	High carbon steels, ordinary low alloy steels
	5	Normal tool steels
	6	*Difficult tool steels*
	7	*Difficult high-strength steels*
Stainless Steels	8	Free-cutting, austenitic stainless steels
	9	Moderately difficult austenitic stainless steels
	10	Austenitic and duplex stainless steels difficult to machine

Table 6.1 Material groups

Eight material classes were included, namely very soft steels, free cutting steels, structural steels, high carbon steels, normal tool steels, free cutting stainless steels, moderately and difficult to machine stainless steels, i.e. groups 1 to 5 and 8 to 10. An analytical table of individual material types that fall within the eight material groups examined is provided in Appendix B. Material class can be considered as ordinal variable within each major material class. It has been quantified on a 1-10 scale, according to table 6.1.

- **Insert grade**: Three types of carbide grades are employed, namely TP10, TP20 and TP35 [Seco Tools AB 1993] corresponding to ISO P10, P20 and P35 application ranges. Insert grade is a categorical variable but it is also linguistically meaningful, since a lower grade corresponds to a more wear resistant insert with lower toughness, whereas a higher grade is tougher but usually exhibits shorter useful cutting life. It has been quantified according to ISO ratings.

- **Type of Cut**: This is a linguistic variable with universe of discourse the linguistic values "finishing", "medium roughing" and "roughing". Type of cut is also a categorical variable. It is straightforward to assume that insert grade is also an ordinal variable. In section 6.2.1 the type of cut is not mentioned among the parameters influencing tool life. This is because the type of cut for each tool life group is essentially a linguistic description of the type of operation based on expert knowledge and taking into account the cutting conditions and the required surface finish. The surface finish in turn is determined by the feed rate and the nose radius for single point cutting tools. Considering the current data availbility, the type of cut has been quantified between 1 and 3, with the value 1 attributed to finishing, 2 to medium roughing and 3 to roughing.

This is a rather crude quantification for the type of cut, since it is conceivable that, for example, finishing cuts can be further distinguished between extra finishing, finishing and semi-finishing. Between two finishing cuts with the same cutting speed and feed rate, one which is performed with a depth of cut of 1.0 mm seems more appropriate to be attributed a higher membership value to the fuzzy set "finishing" than a deeper cut of 2.0 mm. The opposite could be said for their memberships to the fuzzy set "medium roughing". For example, one intuitive choice is to attribute to the first cut the membership values 1.0 and 0.1 to the fuzzy sets "finishing" and "medium roughing" respectively. The second cut could be described by the membership values 0.9 and 0.3 respectively. Such a distinction between types of cut is not supported by the existing tool life model [Alamin 1996] and therefore by the available cutting data. This data do not present any difference in the expected tool life between cuts of the same type but different depth of cut, i.e. the influence of depth of cut to the tool life for every tool life group is considered negligible. This should not be considered as a major problem, since tool life is generally much less sensitive to variations of depth of cut than changes in cutting speed or feed rate. A potential future improvement to the model developed here could look at how to incorporate the depth of cut into a more flexible fuzzification for the linguistic variable type of cut.

	Finishing	**Medium Roughing**	**Roughing**
Minimum	0.05	0.25	0.4
Maximum	0.35	0.5	1

Table 6.2: Feed rate extreme values for various types of cut (mm/rev)

MATERIAL CLASS	**TYPE OF CUT**	**INSERT GRADE**	**TAYLOR PARAMETERS**			**CUTTING SPEED BOUNDARY VALUES**	
			1/α	**1/β**	**C**	**MAX** (m/min)	**MIN** (m/min)
Very soft steel	Finishing	TP10	4.562	0.292	1.49E+13	490	455
Very soft steel	M Roughing	TP10	3.55	0	3.27E+10	430	380
Very soft steel	Roughing	TP10	3.566	0	2.50E+10	390	345
Free cutting steel	Finishing	TP10	4.584	0.298	7.84E+12	415	385
Free cutting steel	M Roughing	TP10	3.446	0	9.89E+09	365	320
Free cutting steel	Roughing	TP10	3.478	0	8.36E+09	330	290
Structural steel	Finishing	TP10	4.594	0.29	4.10E+12	370	310
Structural steel	M Roughing	TP10	3.603	0	1.36E+10	320	255
Structural steel	Roughing	TP10	3.42	0	3.43E+09	280	245
High carbon steel	Finishing	TP10	4.581	0.355	1.41E+12	310	260
High carbon steel	M Roughing	TP10	3.531	0	4.57E+09	270	210
High carbon steel	Roughing	TP10	3.665	0	6.49E+09	245	190
Normal tool steel	Finishing	TP10	4.626	0.344	8.56E+11	265	215
Normal tool steel	M Roughing	TP10	3.552	0	2.79E+09	225	175
Normal tool steel	Roughing	TP10	3.371	0	7.75E+08	210	160

Table 6.3 Pattern generation: Cutting speed boundary values and Taylor equation parameters for each tool life group [Alamin 1996]

MATERIAL CLASS	TYPE OF CUT	INSERT GRADE	TAYLOR PARAMETERS			CUTTING SPEED BOUNDARY VALUES	
			1/α	1/β	C	MAX (m/min)	MIN (m/min)
Free cutting stainless	Finishing	TP20	4.237	0.016	2.61E+10	170	120
Free cutting stainless	M Roughing	TP20	3.642	0	8.24E+08	150	100
Free cutting stainless	Roughing	TP20	3.456	0	2.61E+08	135	95
Free cutting steel	Finishing	TP20	4.612	0.292	4.22E+12	365	300
Free cutting steel	M Roughing	TP20	4.531	2.596	1.31E+11	305	175
Free cutting steel	Roughing	TP20	4.309	2.503	2.61E+10	280	160
High carbon steel	Finishing	TP20	4.626	0.344	8.56E+11	260	220
High carbon steel	M Roughing	TP20	4.424	2.623	1.53E+10	215	120
High carbon steel	Roughing	TP20	4.504	2.616	1.45E+10	195	110
Normal tool steel	Finishing	TP20	4.6	0.281	4.82E+11	230	190
Normal tool steel	M Roughing	TP20	4.469	2.475	1.29E+10	190	130
Normal tool steel	Roughing	TP20	4.559	2.511	1.33E+10	175	120
Structural steel	Finishing	TP20	4.581	0.343	1.69E+12	315	265
Structural steel	M Roughing	TP20	4.481	2.654	4.99E+10	265	150
Structural steel	Roughing	TP20	4.529	2.663	4.04E+10	240	135
Very soft steel	Finishing	TP20	4.584	0.302	6.91E+12	415	355
Very soft steel	M Roughing	TP20	4.456	2.628	1.67E+11	355	200
Very soft steel	Roughing	TP20	4.45	2.646	1.00E+11	320	180
Difficult castings stainless	Finishing	TP35	2.548	0.06	2.21E+06	120	65
Difficult castings stainless	M Roughing	TP35	4.307	0	2.18E+09	100	55
Free cutting stainless	Finishing	TP35	4.131	0.015	4.47E+10	210	165
Free cutting stainless	M Roughing	TP35	1.348	0.081	2.19E+04	175	120
Free cutting stainless	Roughing	TP35	1.394	0.082	2.40E+04	160	110
Free cutting steel	Finishing	TP35	4.601	0.327	2.63E+12	390	335
Free cutting steel	M Roughing	TP35	4.495	2.666	7.29E+10	285	160
Free cutting steel	Roughing	TP35	4.525	2.629	5.45E+10	255	145
High carbon steel	Finishing	TP35	4.602	0.373	4.79E+11	245	200
High carbon steel	M Roughing	TP35	4.495	2.76	1.33E+10	200	110
High carbon steel	Roughing	TP35	4.365	2.164	5.28E+09	180	100
Moderately difficult stainless	Finishing	TP35	4.122	0.021	7.06E+09	145	100
Moderately difficult stainless	M Roughing	TP35	3.9	0	1.29E+09	130	90
Moderately difficult stainless	Roughing	TP35	3.81	0	5.81E+08	120	70
Structural steel	Finishing	TP35	4.607	0.318	1.06E+12	285	225
Structural steel	M Roughing	TP35	4.578	2.694	4.04E+10	230	130
Structural steel	Roughing	TP35	4.453	2.463	1.75E+10	205	120
Very soft steel	Finishing	TP35	4.565	0.325	4.40E+12	390	335
Very soft steel	M Roughing	TP35	4.434	2.666	1.04E+11	330	185
Very soft steel	Roughing	TP35	4.52	2.654	1.07E+11	300	170
Normal tool steel	Finishing	TP35	4.599	0.302	3.30E+11	220	170
Normal tool steel	M Roughing	TP35	4.365	2.614	5.28E+09	180	100
Normal tool steel	Roughing	TP35	4.482	2.6	6.18E+09	165	95

Table 6.3 (cont.) Pattern generation: Cutting speed boundary values and Taylor equation parameters for each tool life group [Alamin 1996].

- **Feed rate**: It is a continuous variable defined in three different regions, one for each type of cut (Table 6.2). A significant overlap exists between finishing and medium roughing (0.25-0.35 mm/rev), as well as between medium roughing and roughing (0.4-0.5 mm/rev). This is because the type of cut is not uniquely defined by the feed rate but depends also on the rest of the cutting conditions.
- **Cutting speed**: It is also a continuous variable which is defined in different regions for each combination of material class, insert grade and type of cut. Cutting speed ranges for each tool life group are shown in Table 6.3 [Alamin 1996].

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
5	3	35	0.535221	115.442	17.90747
3	3	10	0.646825	269.4412	16.73467
3	3	10	0.531361	254.0913	20.45215
3	1	35	0.06833	225.2801	36.18398
3	3	20	0.888796	239.8782	0.919819
1	2	35	0.347307	231.3477	57.058
4	2	35	0.358284	187.4931	13.75321
3	1	35	0.26761	265.1172	11.07116
1	2	10	0.493003	405.5766	18.03204
1	1	35	0.098026	375.3271	16.55063
3	1	10	0.340598	365.3371	9.449632
2	2	35	0.281089	199.0097	99.65873
2	1	10	0.173109	398.7314	15.8367
2	1	35	0.249718	387.8353	5.083784
4	1	10	0.3076	278.1124	13.60646
3	3	20	0.902525	209.9425	1.614949
2	2	20	0.495023	267.6423	8.155333
1	2	10	0.334173	409.8448	17.37419
3	1	20	0.234204	300.3192	12.41601
9	3	35	0.445486	87.24297	23.43579

Table 6.4 Pattern generation: An Example of Training Patterns

A total of fifty six tool life groups are considered, which correspond to a wide range of cutting process inputs. Approximately 255 training examples per group has been created to represent the cutting process performance in terms of expected tool life for each one of the tool life groups, totalling up to 14,262 patterns. This pattern set will be referred to hereafter as training pattern set. The patterns have been randomly generated for each tool life group, so that the set of feed rate and cutting speed values follows a two-variate uniform distribution with extreme values those of Table 6.2 and 6.3. The tool life value for each set of inputs is calculated according to equation (6-1), where α, β and C take the values shown in Table 6.3. Examples of some training patterns are shown in Table 6.4, while a partial list of the pattern set can be found in Appendix C. Each pattern consists of an input-output pair of the following type:

{Material Class, Type of Cut, Insert Grade (ISO P), feed rate (mm/rev), cutting speed m/min), tool life (min)}.

The large size of the patterns set ensures that the neurofuzzy model will receive sufficient information to achieve the desired predictive accuracy.

6.3 Determination of ANFIS architecture

Once a decision has been taken about input selection, the next step is the identification of a suitable ANFIS architecture. This includes the determination of the input space partitioning, the number of the fuzzy rules, the premise and consequent parts of the fuzzy rules as well as the initial premise and consequent parameters. The diagram of the ANFIS model selected, consisting of five inputs and one output is shown in Fig. 6.1.

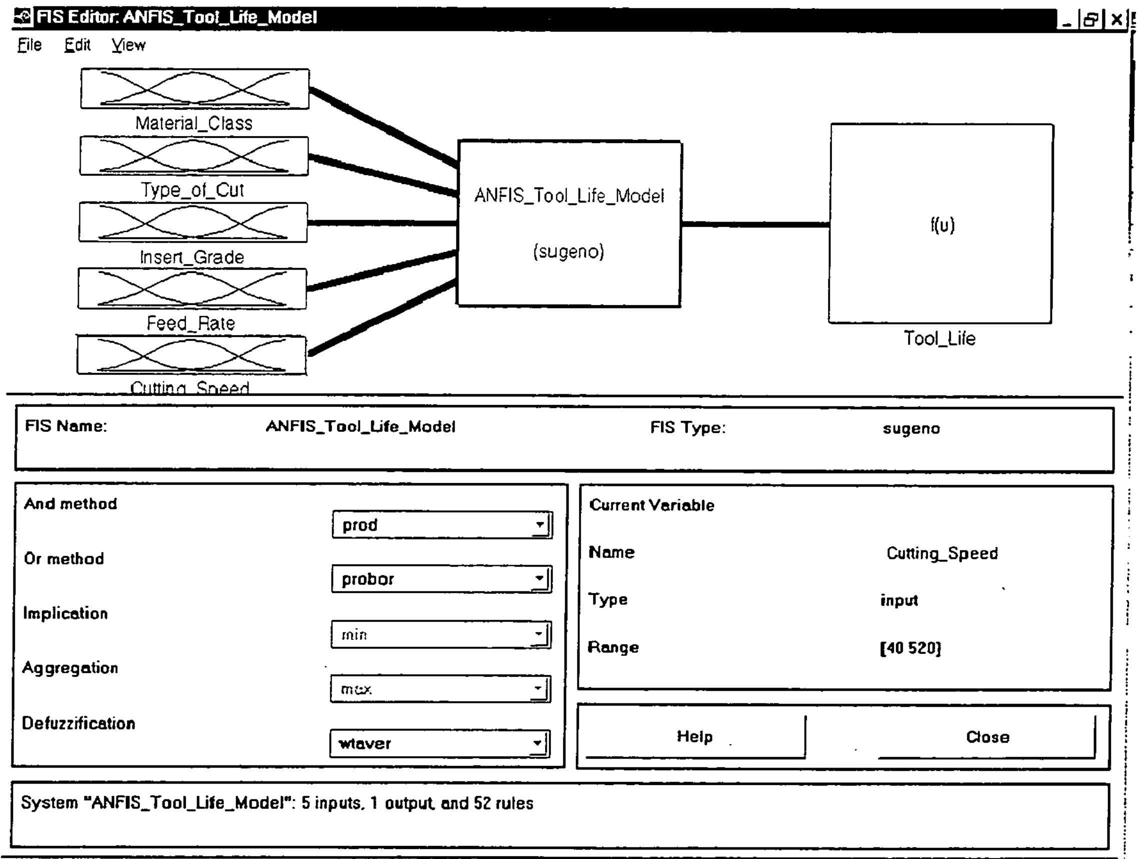

Figure 6.1: ANFIS Model for Tool Life

The Fuzzy Logic toolbox (v1.0) of Matlab [Jang and Gulley 1995] provides with algorithms for the identification Sugeno-type fuzzy inference systems (see chapter 4, [Takagi and Sugeno 1985]) with the following restrictions:

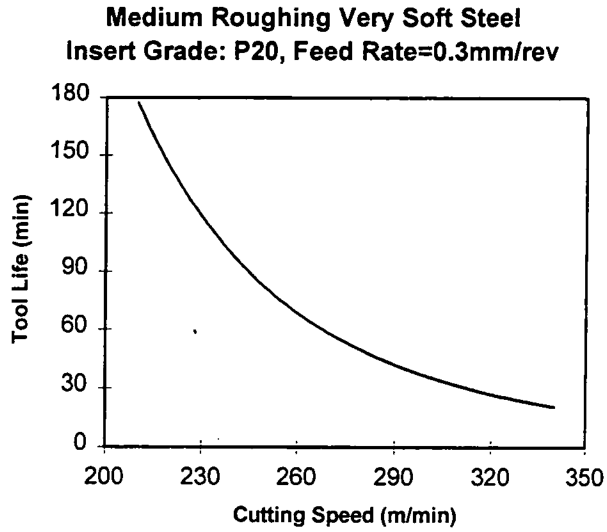

Figure 6.2: *Expected tool life against cutting speed for medium roughing cuts of very soft steel with constant feed rate [Alamin 1996].*

- First order Sugeno-type systems
- Weighted average defuzzification
- Rules weighting equal to unity.

The Sugeno type fuzzy inference system is an additive model of weighted local function approximations. A first order Sugeno fuzzy model is therefore a weighted sum of locally linear approximators. It should be noted that the weighting factors are not constants but are determined by the level of fulfilment of the fuzzy rules premise parts. The transition between the operating regimes of the fuzzy rules is gradual rather than abrupt, due to the overlap between the antecedent membership functions. This is why Sugeno fuzzy models achieve a smooth input-output mapping, when sufficient rule overlap exists.

To illustrate the point, a simple case of medium roughing cut of very soft steel with an ISO P20 insert, at a feed rate of 0.3mm/rev is considered. The expected tool life is calculated according to equation (6.1) and Table 6.3 [Alamin 1996] and is shown in Figure 6.2 against cutting speed. A simple Sugeno fuzzy model is constructed, comprising the following three rules:

1. IF Cutting Speed (v) is low then Tool Life $T = -2.28 \cdot v + 647.4$ (min)

2. IF Cutting Speed (v) is medium then Tool Life $T = -0.885 \cdot v + 298$ (min)

3. IF Cutting Speed (v) is high then Tool Life $T = -0.394 \cdot v + 153.9$ (min)

where the consequent parts are local linear approximations of the tool life function, obtained by simple linear regression. When the cutting speed is attributed crisp membership to the linguistic terms "low", "medium" and "high", as shown in Figure 6.3a, the input-output mapping is non-smooth (Figure 6.3b). In contrast, when cutting speed is fuzzified according to Figure 6.3c, a smooth input-output mapping is obtained (Figure 6.3d).

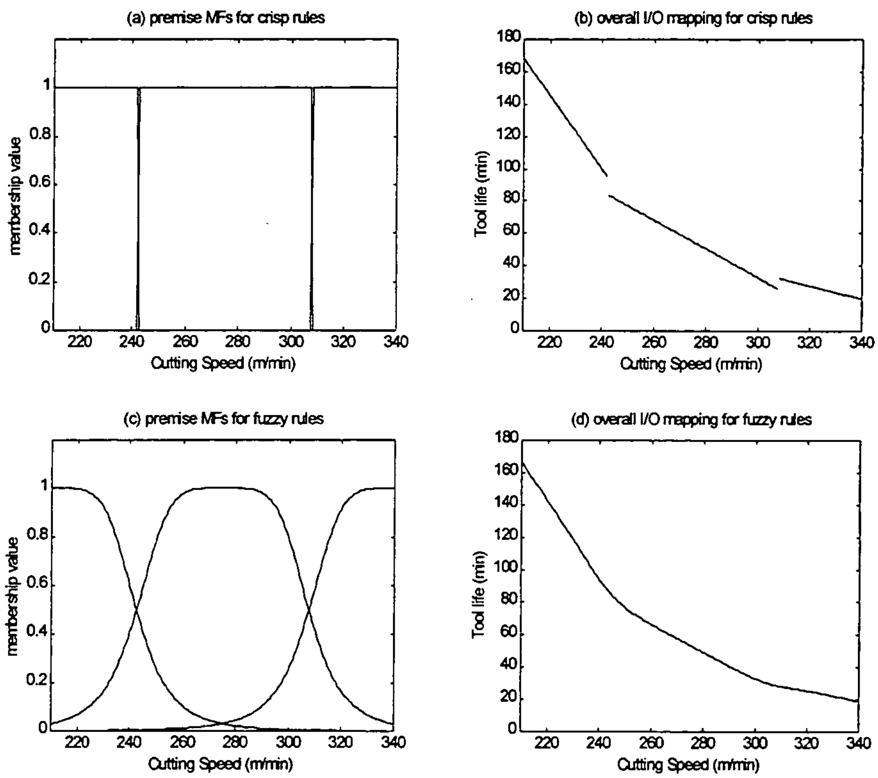

Figure 6.3. Contrast between modelling with crisp and fuzzy rules: (a)crisp partition of the cutting speed range, (b) input-output for crisp rules, (c) fuzzy partition of the cutting speed range, (d) input-output mapping with fuzzy rules

The example given is only indicative of the significance of employing overlapping fuzzy rules. The advantage of employing such rules has been exploited in the present work, which deals with the far more complex problem of obtained a single tool life model for a wide range of material classes, types of cut, insert grades, feed rate and cutting speed. To obtain such a model in the five-dimensional input space considered herein, a method for generating a reduced set of fuzzy rules based on the available cutting data is necessary. A grid partitioning of the input space suffers from the exponential growth in the number of fuzzy rules produced. Indeed, the number of fuzzy rules produced by grid partitioning of an n-dimensional input space into m regions per input is m^n. A grid partitioned 5-dimensional input space, such the one involved in the present tool life modelling problem, would require 243, 1024, 3125 or 7776 fuzzy rules for 3, 4, 5 or 6 membership functions defined per input. The subtractive clustering algorithm [Chiu 1994, Chiu 1996] has been employed to overcome this problem. As mentioned in chapter 4, this algorithm is by no means an optimal one. However it is a very simple and computationally efficient algorithm and is directly supported by the Fuzzy Logic Toolbox of Matlab. A good initial ANFIS model depends on the input space partitioning. The subtractive algorithm is only a means that provides an ANFIS structure with a good initial input space partitioning and a reduced set of fuzzy rules. Therefore, emphasis will be given to those particular characteristics that an ANFIS model should possess in order to stand a good chance of being adequate for initial state.

The premise membership functions are selected to be of Gaussian shape. This choice ensures smoother boundaries for the activation of the fuzzy rules rather than triangular or trapezoidal membership functions. Yet, Gaussian is a simple type of membership function, involving only two parameters for its definition. The subtractive algorithm can yield different results, depending on the particular choices made for the parameters relevant to the calculation of the density measure of each potential cluster (equations 4.9, 4.10), as well as on the stopping criteria mentioned in section 4.4. The algorithm parameters determine the number of the derived fuzzy rules, as well as the shape of the initial membership functions. For Gaussian membership functions the shape is determined by the centre and the spread of the Gaussian kernel. These parameters will be ultimately determined by the parameter optimisation algorithm employed. This can be the hybrid algorithm described in Chapter 4 [Jang 1993], the fast Levenberg-Marquardt algorithm [Jang 1996] or indeed any gradient-based nonlinear optimisation algorithm.

Alternatively, the fuzzy membership functions parameters can be optimised by probabilistic search methods, involving techniques such as evolutionary strategies or genetic algorithms [Cordon and Herrera 1995]. The drawback of gradient-based techniques is that they are often long and tedious procedures prone to become stack in local minima. Evolutionary strategies and genetic algorithms are well suited to find global optima, often at the expense of long search times. Evidently, the closest the fuzzy system parameters are to the optimal values, the easier the parameter optimisation task becomes. Therefore, it is important that a good initial state for the fuzzy system is found. For the above reasons, the subtractive clustering parameters have to be attributed appropriate values to ensure that the resulting fuzzy inference system posses the capacity of achieving a highly accurate mapping with a minimum number of fuzzy rules. Fortunately these choices are rather intuitive, as it is straightforward to decrease the number of the fuzzy rules obtained by setting, for example, higher values for the inhibition area around a cluster centre or by increasing the threshold value above which a potential cluster centre becomes definitely accepted as the core position of a new fuzzy rule. If background knowledge is available about the modelling task, it can simplify the determination of the initial membership function parameters. For example, the support of the initially derived membership functions can easily be determined by setting appropriate values to the cluster neighbourhood of influence parameters of equation 4.9. Yet, some initialisation trials are still necessary, in order to decide upon the model complexity required to achieve the desired mapping accuracy.

Once the premise parameters are identified, the consequent ones are calculated by the least squares method described in chapter 4. Thus, a completely defined initial ANFIS structure is derived.

6.4 Initialisation results

Several initialisation runs have been examined which resulted in a variety of ANFIS models comprising different number of fuzzy rules, ranging from 38 to 73. All the results obtained confirm that it is beneficial to employ overlapping fuzzy rules, as it was also demonstrated in the simple tool life modelling case examined earlier in this section. Three of these initialisation results are now presented as indicative examples.

All the input-output data pairs employed in the fuzzy model initialisation by the subtractive clustering algorithm are normalised within the region 0 to 1. Thus, equal significance of all inputs, regardless of specific scaling is ensured. The accuracy of the initial ANFIS model is cross-checked over two different pattern sets. The first is the one employed for the initialisation procedure and is a subset of the earlier mentioned in paragraph 6.2.2.2 complete training pattern set consisting of 3,000 patterns. This set will be referred to hereafter as the initialisation pattern set or simply initialisation set. The second is a separate set of 4978 patterns, used only for checking the mapping accuracy achieved. This set will be referred to as the checking pattern set or simply checking set. A partial but indicative list of this set is shown in Appendix D for the case of cutting very soft steel with an ISO P10 insert. The complete checking set contains cutting data such as those shown in Appendix D, but for the whole range of workpiece materials and insert grades. The tool life values in this set are calculated in exactly the same way as in the case of the training set. Thus, the checking set corresponds to a series of tests spanning a very wide range of inputs combinations that provides with a good indication of the modelling performance. The subtractive clustering algorithm parameters for the three different initialisation cases are selected according to Table 6.5.

Subtractive Clustering Parameters	**Comments**	**Parameter values for each initialisation run**		
		Run 1	Run 2	Run 3
Neighbourhood radii: specifies the range of influence of each cluster centre as a fraction of the data space in each data dimension (i.e. for each input and output)	Material Class	0.1	0.1	0.26
	Type of Cut	0.2	0.2	0.69
	Insert Grade	0.2	0.2	0.55
	Feed Rate	0.45	0.45	0.45
	Cutting Speed	0.2	0.2	0.2
	Tool Life	0.2	0.2	0.2
Inhibition area	Multiplied by the neighbourhood radius, determines the neighborhood of a cluster center within which the existence of other cluster centers is discouraged.	1.85	1.35	1.1
Upper threshold for cluster potential	The potential, as a fraction of the potential of the first cluster centre, above which another data point is accepted as a cluster center	0.55	0.45	0.4
Lower threshold for cluster potential	The potential, as a fraction of the potential of the first cluster centre, below which a data point can not be accepted as a cluster center	0.2	0.15	0.125

Table 6.5: ANFIS Initialisation: Subtractive clustering parameters

The first initialisation run differs from the second only in the inhibition area and threshold parameters. In the third initialisation run large values for the neighbourhood area parameters for the material class, type of cut and insert grade input dimensions are selected. Therefore, the support of the obtained membership functions for these inputs is much wider than in the case of the first two initialisation procedures. The impact of this difference on the achieved mapping performance of each individual initial ANFIS model is examined now in detail.

Table 6.6 shows initialisation results for three different initialisation procedures. The first initialisation results in an ANFIS model with 56 fuzzy rules and exhibits a root mean squared error (RMSE) of 5.51 and 5.8 for the initialisation and checking sets respectively. This is a rather poor initialisation but the situation improves with the second initialisation, which achieves a RMSE of 2.76 and 3.12 for the same sets with a 68 fuzzy rules base.

<table>
<tr><th rowspan="2">ANFIS Initialisation Run</th><th rowspan="2">Number of Fuzzy Rules</th><th colspan="2">RMSE Error</th><th rowspan="2">Comments</th></tr>
<tr><th>Initialisation Set</th><th>Checking Set</th></tr>
<tr><td>1</td><td>56</td><td>5.51</td><td>5.80</td><td rowspan="2">The subtractive algorithm parameters are such that the resulting membership functions for material class, type of cut and insert grade correspond to crisp rules.</td></tr>
<tr><td>2</td><td>68</td><td>2.76</td><td>3.12</td></tr>
<tr><td>3</td><td>51</td><td>1.96</td><td>1.92</td><td>A wider neighbourhood of influence is selected for each potential cluster centre in the case of material class, type of cut and insert grade resulting to a significant overlap of fuzzy rules.</td></tr>
</table>

Table 6.6: ANFIS Initialisation: Model size and initial RMSE

It seems that the addition of some extra fuzzy rules increased the initial mapping accuracy of the ANFIS model. However the third example appears to be rather odd since it achieves better accuracy with fewer fuzzy rules.

The explanation for the improved performance of the third model lies in the different deep structure of the model. In particular, the derived premise parameters of the third model for the material class, type of cut and insert grade inputs define membership functions with much wider support than those of the first two models. It has been mentioned in chapters 4, 5 that the real merit of fuzzy and neurofuzzy systems is that they provide with meaningful representations of knowledge. Indeed, by examining the membership functions of the ANFIS models, some conclusions are drawn about the mapping performance of the initial models in the following sections.

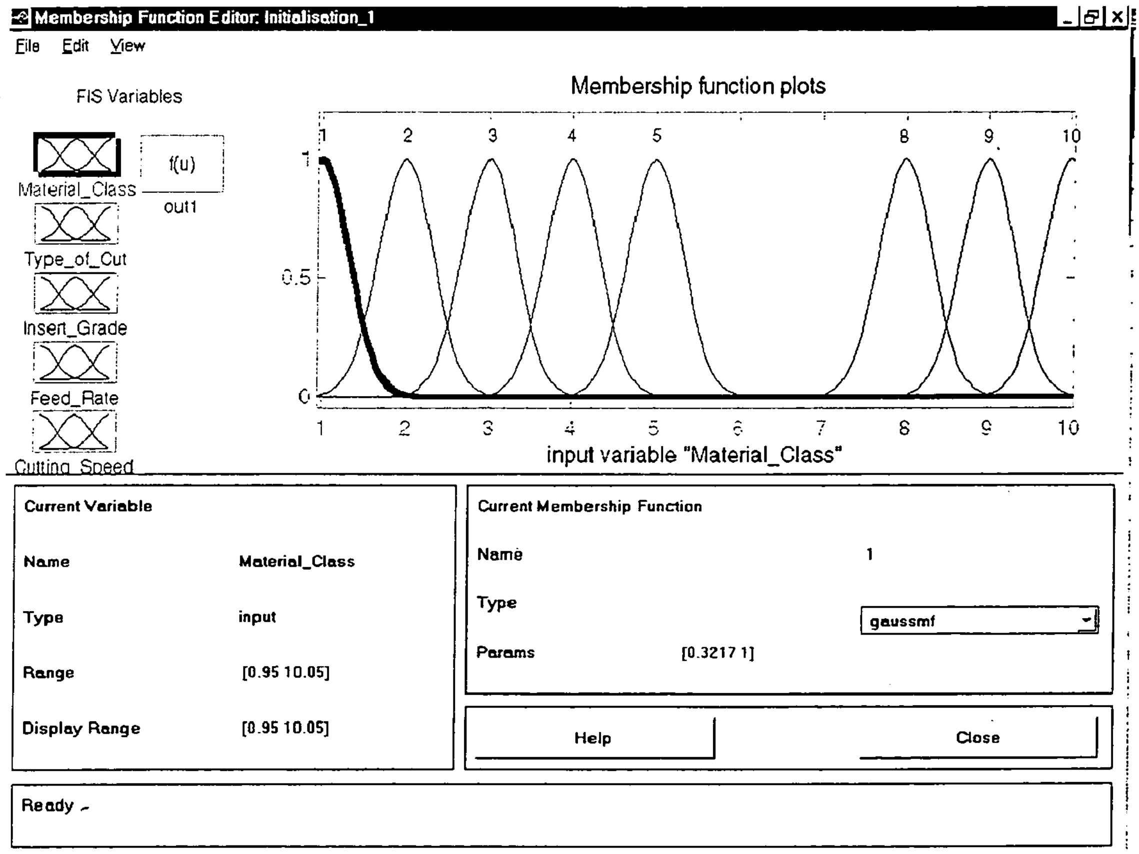

Figure 6.4. Material classes membership functions which equivalent to crisp set boundaries

6.4.1 Membership functions corresponding to crisp rules.

The membership functions derived from the initialisation runs 1, 2 (Table 6.5, 6.6) are examined first. The input space partition for the variable *Material_Class* is shown is Figure 6.4.

Material classes are quantified according to Table 6.1. A clear distinction between the different classes exists. Thus, a workpiece material either belongs to a specific class with membership function 1 or not at all, i.e. it belongs to it with membership function 0. In other words there are crisp boundaries separating material classes. This is consistent with conventional logic and in particular with the law of excluded middle. The implication of these sharp boundaries between the material classes is that the output of the ANFIS model is always triggered by those and only fuzzy rules that correspond to the relevant material class. The activation level of the rest of the fuzzy rules is zero.

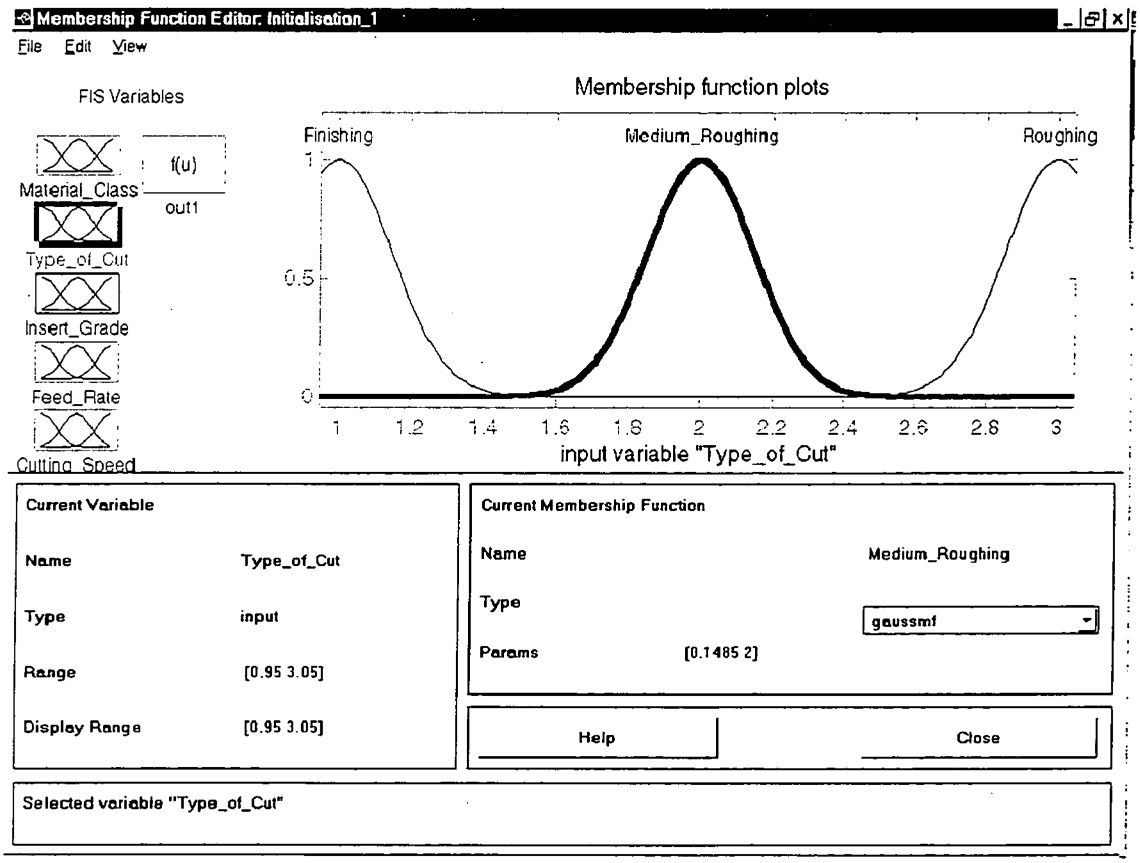

Figure 6.5. Type of cut membership functions which equivalent to crisp set boundaries

The same comment applies to the rest two of the categorical variables, i.e. the type of cut and the insert grade. Figures 6.5-6.6 depict the relevant membership functions. Again there are sharp boundaries separating the input variables into crisp sets. For example, a type of cut is either finishing or not. It makes no difference whether it is finishing with a feed rate as high as 0.3mm/rev which brings it close to medium roughing. Fuzzy rules model output. The same kind of rule activation should be expected in the case of the insert grade.

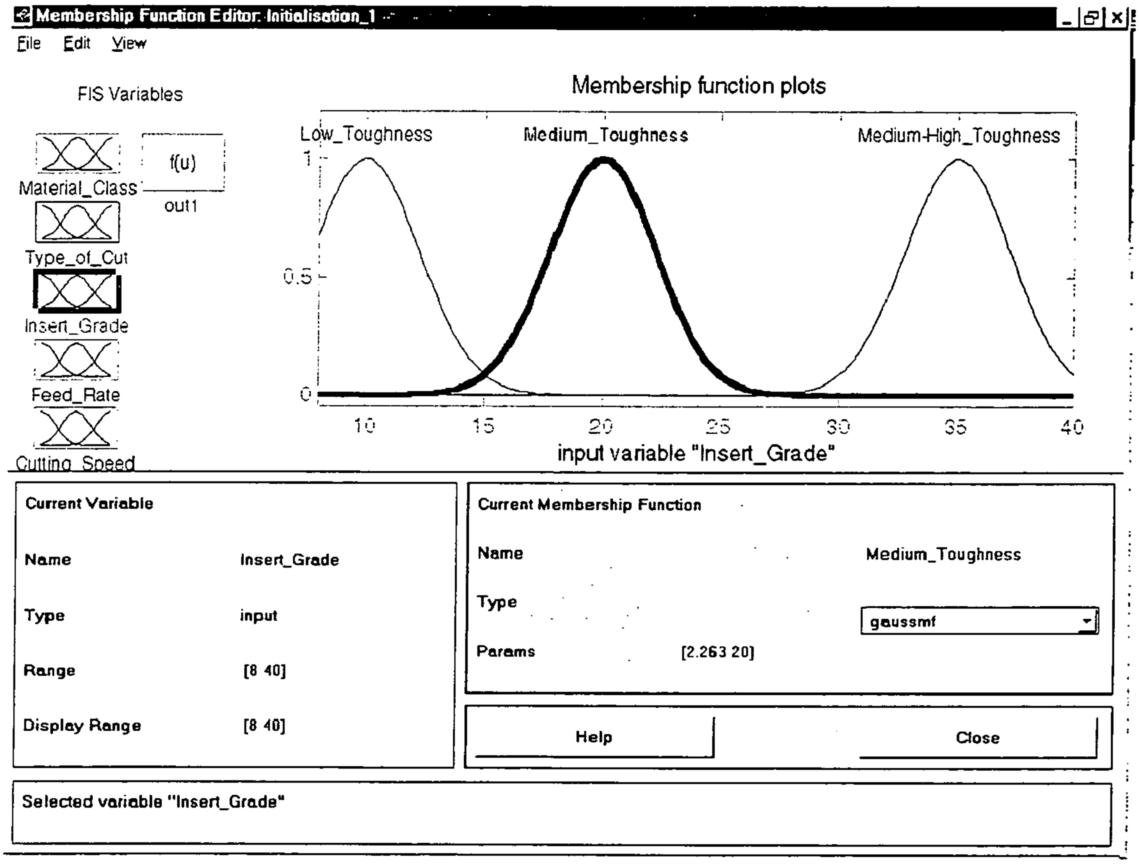

Figure 6.6. Insert Grade toughness membership functions which equivalent to crisp set boundaries

The second ANFIS initialisation run (Tables 6.5, 6.6) differs from the first only on the number of the fuzzy rules and not on the membership function parameters. Appropriate selection of the initialisation algorithm parameters (Table 6.5) modifies the acceptance criteria for the fuzzy rules. In particular, a significantly lower inhibition area value of 1.35 was defined in this case, allowing for more data points to develop a high potential to become accepted as cluster centres. Also the rejection and acceptance criteria have been slightly modified, but they are not directly comparable to those defined in the first initialisation run, as the data now develop higher potential values, due to the shrinking of the inhibition areas. Thus the total number of obtained fuzzy rules increases to 68. The RMSE error is reduced in both the initialisation and checking sets (Table 6.6) but still it is relatively high.

6.4.2 Membership functions corresponding to soft rules.

In contrast to the first two initialisation runs, the third one results in the determination of such premise parameters that allow for a "soft" definition of the boundaries between the material classes. These membership functions are shown in Figures 6.7, 6.8 and 6.8 for the case of material class, type of cut and insert grade respectively.

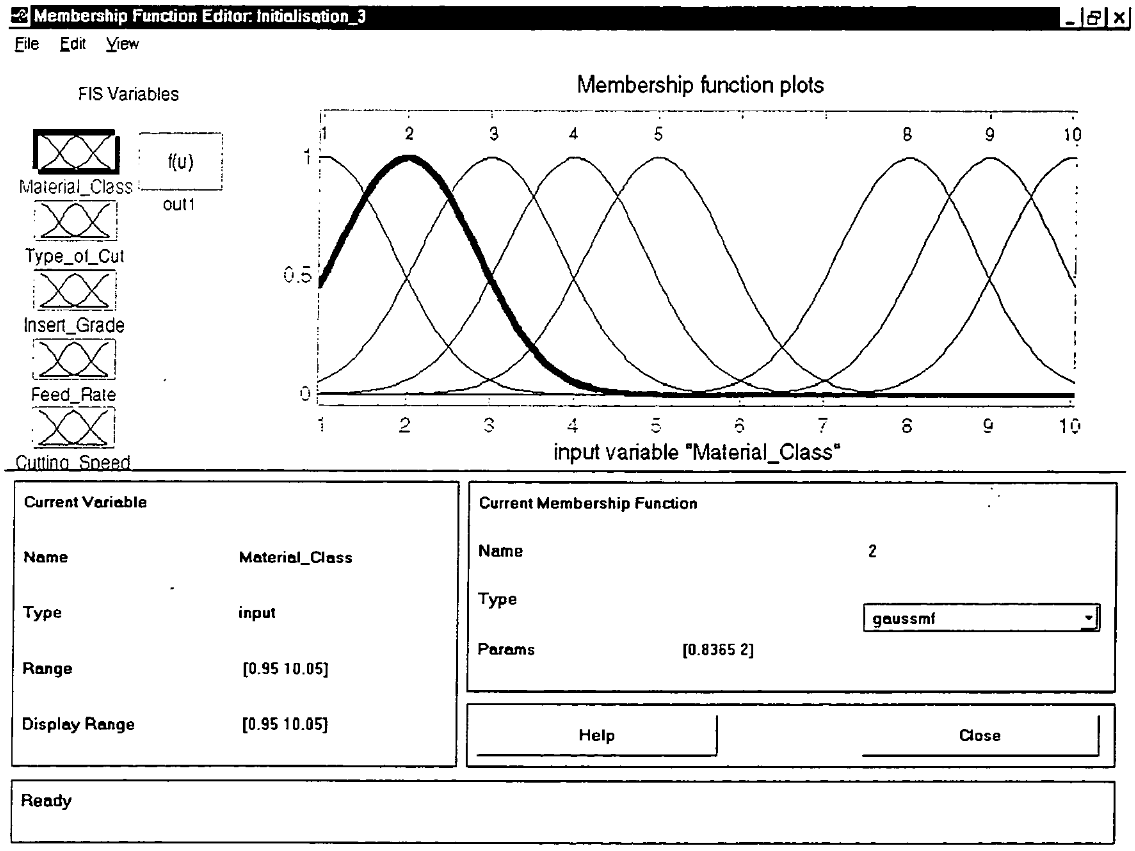

Figure 6.7. Material classes membership functions which equivalent to soft set boundaries

The degree to which each workpiece material belongs to a certain material class, i.e. the material class membership values are graphically shown in Figure 6.7 and mathematically expressed by the following equation:

$$\mathbf{A} \equiv \left[x_i \in x_j \right] = \begin{bmatrix} 1 & 0.49 & 0.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.49 & 1 & 0.49 & 0.06 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.06 & 0.49 & 1 & 0.49 & 0.06 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.06 & 0.49 & 1 & 0.49 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.06 & 0.49 & 1 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 1 & 0.49 & 0.06 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.49 & 1 & 0.49 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.49 & 1 \end{bmatrix} \tag{6-2}$$

where x_i , denotes workpiece material corresponding to the i-th class, i, j=1,2,3,4,5,8,9,10 and **A** is the material class membership values matrix. For instance a workpiece material identified as free cutting steel (material group 2) has the following membership values to the fuzzy set of material classes:

$$\mu(x_2) = \left\{ \frac{0.49}{1} + \frac{1}{2} + \frac{0.49}{3} + \frac{0.05}{4} + \frac{0.00}{5} + \frac{0.00}{8} + \frac{0.00}{9} + \frac{0.00}{10} \right\} \tag{6-3}$$

where the fuzzy set notation introduced in section 3.2 is employed, i.e. the numerators denote membership values and the denominators correspond to material classes quantified according to Table 6.1. Accordingly, the type of cut membership values for the third ANFIS initialisation are graphically illustrated in Figure 6.8 and in matrix format by the following equation:

$$\mathbf{B} \equiv \left[x_i \in x_j \right] = \begin{bmatrix} 1.00 & 0.15 & 0.00 \\ 0.15 & 1 & 0.15 \\ 0.00 & 0.15 & 1 \end{bmatrix} \tag{6-4}$$

where x_i stands for finishing, medium roughing or roughing for i=1,2,3 respectively. According to the above equation, the membership values of a finishing cut, e.g. type of cut 1 (section 6.2.2.2), to the set of types of cut is:

$$\mu(x_1) = \left\{ \frac{1.00}{finishing} + \frac{0.15}{medium\_roughing} + \frac{0.00}{roughing} \right\} \tag{6-5}$$

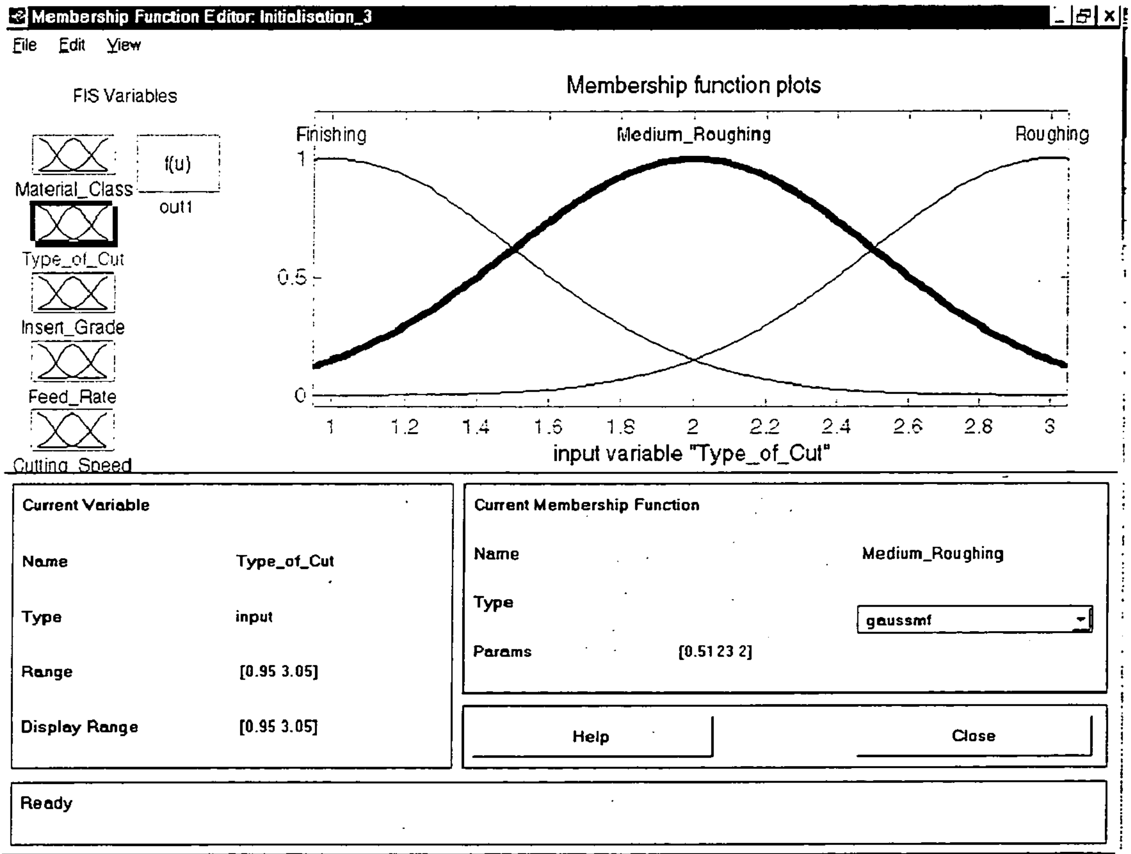

Figure 6.8. Type of cut membership functions which equivalent to soft set boundaries

Finally, the graphical representation of the insert grade input space partitioning obtained by the thirs ANFIS initialisation run is shown in Figure 6.9. The insert grade toughness membership values matrix is:

$$\mathbf{C} \equiv \left[x_i \in x_j \right] = \begin{bmatrix} 1.00 & 0.27 & 0.00 \\ 0.27 & 1 & 0.05 \\ 0.00 & 0.05 & 1 \end{bmatrix} \tag{6-6}$$

where x_i corresponds to P10, P20 or P35 for i=1,2,3 respectively. For instance, an ISO P20 insert has the following membership values to the set of ISO P grades:

$$\mu(x_{P20}) = \left\{ \frac{0.27}{P10} + \frac{1}{P20} + \frac{0.05}{P35} \right\} \tag{6-7}$$

Clearly, the initial premise parameters of the third initialisation are such that the derived ANFIS model better exploits the benefits of fuzziness and achieves improved fuzzy inference. The representation capabilities of the ANFIS network are enhanced, since the desired mapping is achieved through multiple firing not only of the fuzzy rules whose premise parts are well matched by the input vector, but also of those that appear to be of some relevance to the input values, without fully matching the premise conditions.

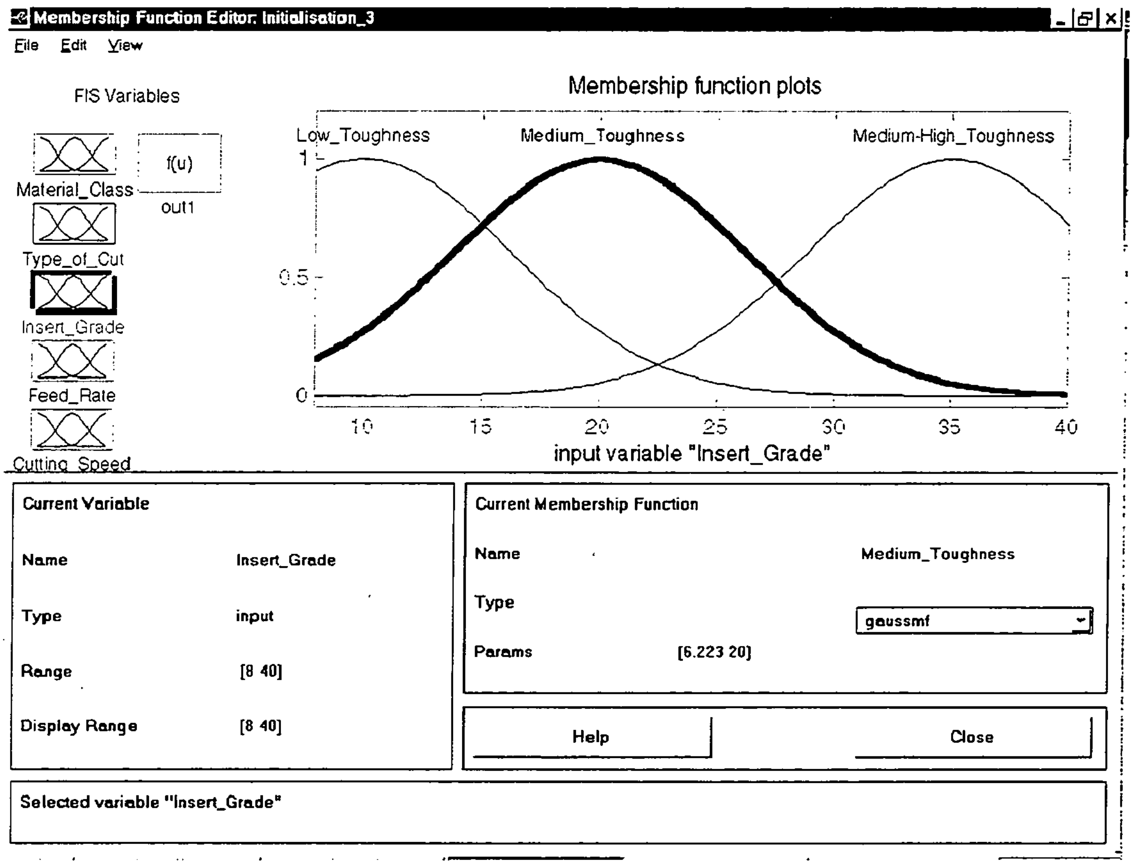

Figure 6.9. Insert grade toughness membership functions which equivalent to soft set boundaries

It could be argued that even the first two initialisation procedures might well have the potential to result in an accurate model, provided that an appropriate algorithm for modifying the premise parameters is available. However, in multidimensional modelling problems the error surface usually exhibits several local minima. Since the premise parameters are identified by a gradient search method as described in chapter 4, the possibility of the learning algorithm becoming stack in local minima is higher for initial states that are far from the global minimum. The learning task is further complicated by the fact that the available training data set attributes a few discrete values to input parameters such as material class, type of cut and insert grade. For example, there are no materials being quantified in the material class scale between the integer values of Table 6.1. Similarly, in the type of cut dimension, there are no training data that would lie within the integer values 1, 2 and 3. The same comment applies to insert grade, where only three grades are present in the training data set. Due to the scarcity of patterns along the material class, type of cut and insert grade scales, the model building problem becomes ill-conditioned. A modelling problem is said to be well-conditioned or well posed if a corresponding output exists within the predefined output range for each input vector, solution uniqueness is guaranteed and the actual input-output mapping is continuous, otherwise it is ill-conditioned or ill-posed [Haykin 1994].

Because of the ill-conditioning, large shallow slope areas are present in the error hypersurface, posing problems to the gradient learning algorithm convergence. Unless the initial state is close enough to the desired error surface minimum, the possibility of convergence to a local minimum is high. However, it should be emphasised that the model transparency achieved with neurofuzzy modelling allowed the determination of a good initial model state. Fuzzy models are particularly well suited for making a meaningful good initial choice of parameters by taking advantage of available empirical knowledge over the particular modelling problem, as well as of the interpretability of the model fuzzy rules activation. This is a significant advantage of neurofuzzy modelling in comparison to other modelling approaches such as multilayer perceptron-type neural networks, polynomial approximations etc.

6.4.3 Feed rate and cutting speed membership functions

Tool life is more sensitive to cutting speed variation than to changes in the feed rate. In addition, the range over which the input variables are defined for each tool life group is generally a smaller portion of the whole range of the allowable values for the cutting speed rather than for the feed rate. Therefore the support of the defined membership functions should be smaller for the case of the cutting speed. Therefore, according to Table 6.5, a high neighbourhood area value of 0.45 has been selected for feed rate and a smaller one (0.2) for cutting speed. This results in a relatively more dense partitioning of the cutting speed input space. On the other hand, the feed rate input space is less finely granulated, in order to reduce the size of the rule base. The shape of the feed rate and cutting speed membership functions is the same for all three initialisation procedures and is illustrated in Figures 6.10 and 6.11.

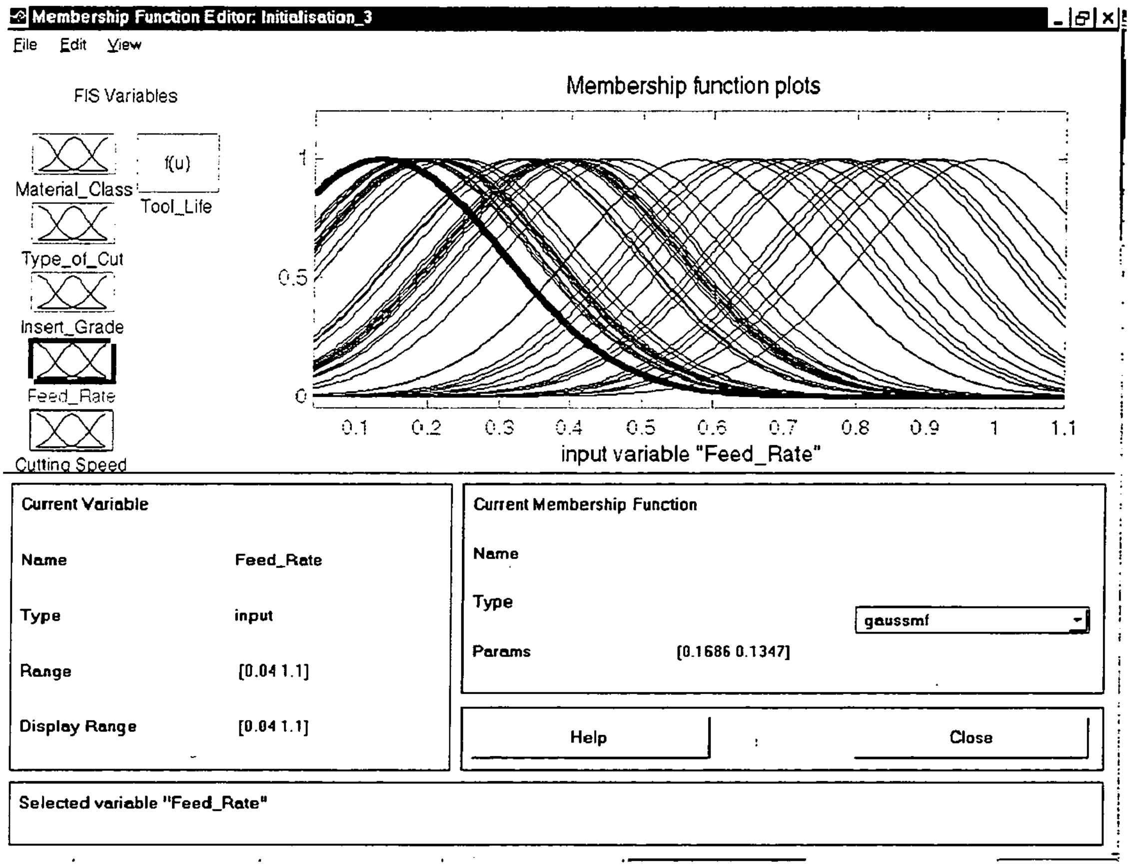

Figure 6.10. Feed rate membership functions

It is worth noting the contrast between the sparsely defined membership functions for the input parameters material class, type of cut and insert grade and the much more dense partitioning of the feed rate and cutting speed input spaces. In fact, according to the definition of ANFIS models in section 4.2 [Jang 1993], for each one input variable there are defined as many membership functions as many are the fuzzy rules. This is clearly shown in Figures 6.10 and 6.11, but it is not apparent in Figures 6.4-6.9. The reason for that is that the subtractive clustering algorithm attributes cluster centres only at data point positions. Since there are no data between the values 1,2,3,4,5,8,9,10 for material class, the antecedent membership functions relevant to the same material class are identical. The same applies to the other two categorical variables, i.e. type of cut and insert grade. Thus, ANFIS models may present some structural redundancy in the form of multiply defined identical membership functions.

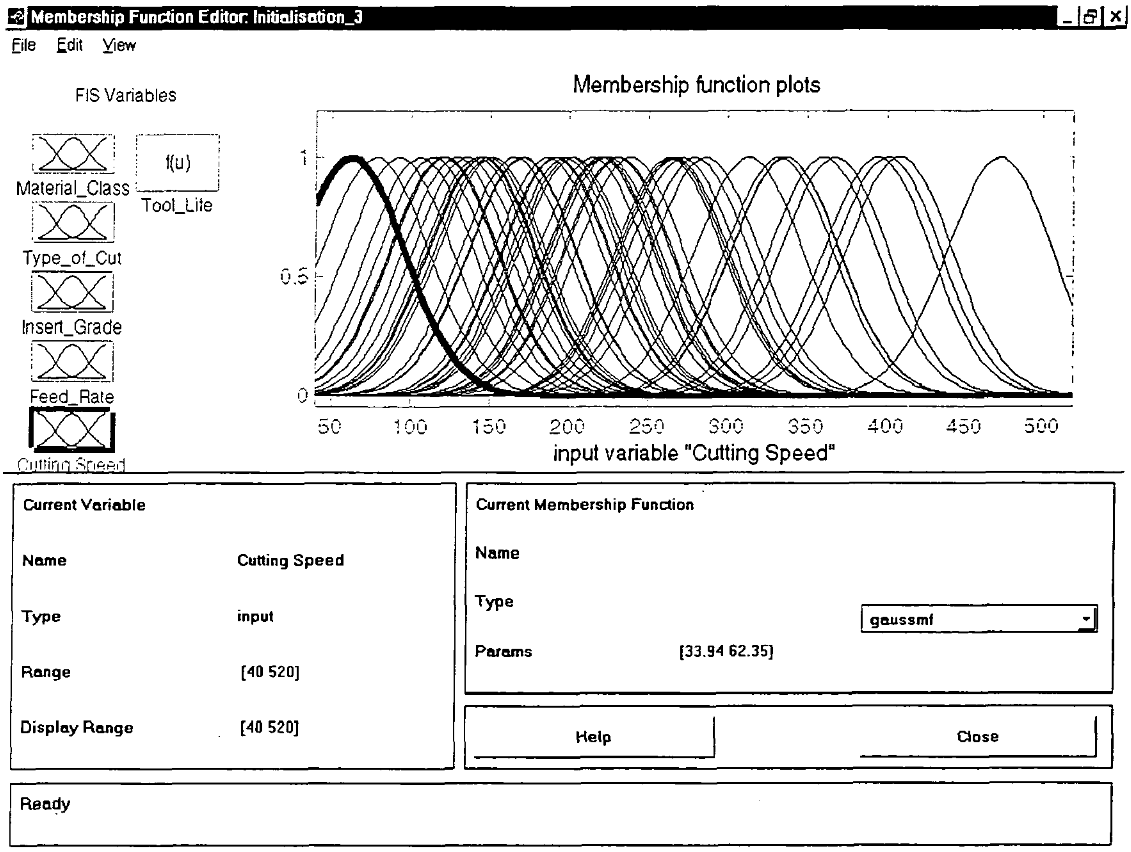

Figure 6.11. Cutting speed membership functions

Once appropriate parameters have been defined for the subtractive algorithm, a new initialisation run was performed over the complete training set of 14262 patterns, in order to reduce the possibility of an initial state fairly biased towards a small set of patterns. This resulted in an initial ANFIS model with 52 fuzzy rules and Gaussian membership functions of similar shape to the third initialisation case. This initial structure provides a good compromise between modelling accuracy and modelcomplexity. The fuzzy rule base is of reasonable size, considering the applicability range of the model. For example, a neural network-based tool life model has been reported for restricted inputs range with just two parameters taken into account, i.e. the feed rate and the cutting conditions, which resulted in a neural structure with no less than 16 hidden nodes [Ezugwu et al. 1995]. The initial root mean square error of mapping was 2.4221 and 2.2994 for the complete training and the checking pattern set respectively.

6.5 Conclusion

This chapter has examined the structure identification issue for tool life modelling. An initial ANFIS tool life model of reduced complexity has been derived by employing subtractive clustering and linear least squares to identify the number of fuzzy rules as well as the initial premise and consequent parts of the fuzzy inference system. The model is quite generic since it is relevant to a wide combination of workpiece materials, insert grades and cutting conditions. The model so far is completely data driven since all its parameters are obtained based on tool life cutting data. A priori knowledge, namely results of previous work on empirical tool life modelling previously carried out at Durham University [Alamin 1996, Maropoulos and Alamin 1996] has been used to create a persistently exciting machining data set.

It is important to realise that the membership functions presented so far are only the initial ones and their parameters will be modified after training in order to improve the achieved mapping. However, because learning will be initiated from a relatively "good" starting point, the deviation of the final premise parameters values from their initial positions should be rather small. Thus, fast convergence is expected. This is an advantage of the ANFIS model trained by the particular hybrid learning algorithm, mentioned in chapter 4, which is not shared by other computational intelligence modelling architectures such as multi-layer perceptrons trained by a gradient based algorithm, where a much slower convergence to the final state is anticipated. The model learning process is examined in the next chapter, where some indicative results followed by discussion are also included.

TOOL LIFE MODEL PARAMETER OPTIMISATION

Learning, results and discussion

7.1 Introduction

In the previous chapter the structure identification problem for neurofuzzy tool life modelling has been examined. An initial ANFIS model has been derived which serves as a good starting point for initiating the optimisation of the antecedent and consequent parameters. The subtractive clustering algorithm was employed for fuzzy rule base structure identification and antecedent parameters initialisation. The initial consequent parameters were calculated by simple linear least squares estimation. The fuzzy rule structure is intended to remain unchanged during adaptation, while both the premise and consequent parameters can be optimised by employing neural network - like learning algorithms.

The initialisation procedure did not rely on the availability of an expert for the design of the fuzzy rule base, as the overall process has been data-driven. The selection of the subtractive clustering algorithm parameters may seem to carry some arbitrariness. However, these choices are not meant to be optimal. They are simply based on human intuition about the cutting process, thus facilitating the derivation of a good initial ANFIS model. In principle, the subtractive algorithm parameters may have been chosen in a different way, so that the obtained initial premise membership functions support is more or less wide and the core positioning far away from the optimal one. Then, an efficient learning algorithm would have been required to optimise these parameters without becoming stack in local minima. Gradient based methods may not be appropriate if the error hypersurface is highly complex or if the parameter optimisation problem is ill-conditioned. In the latter case large shallow slope areas may be present on the error hypersurface, posing problems to the optimisation algorithm convergence. Then, other global optimisation methods have to be employed. Global optimisation can be pursued via probabilistic reasoning, i.e. genetic algorithms and evolutionary strategies. However, these methods usually involve long searches through the parameters space. It is often preferable to employ a fast optimisation algorithm, such as the hybrid learning algorithm described in chapter four, provided that a good initial model selection is feasible. The ease of obtaining a good initial model, which can later be optimised by following a neural network-like learning procedure, is a significant merit of neurofuzzy systems. This advantage has been exploited in the present work by employing a two stage strategy for the identification of a neurofuzzy model for tool life, i.e. first the determination of a good initial ANFIS model and second the optimisation of this model parameters based on training data.

This chapter deals with the parameter optimisation of the tool life model, based on the hybrid learning algorithm described in chapter four. The learning procedure requires a low number of training epochs, since it is initiated from a good starting point. The derived model is a good compromise between model size and mapping accuracy for a wide range of combinations of material class, type of cut, insert grade and cutting conditions. A series of indicative results are presented which verify the predictive performance of the model. An important point demonstrated in this chapter is that neurofuzzy modelling is not simply another black-box identification method. The input-output functional mapping is shown to be achieved in a meaningful way. Indeed, it is quite straightforward to interpret how the fuzzy rules are activated by a set of inputs and therefore to draw some conclusions about the overall modelling achieved.

The representation power, the simultaneous handling of both numerical and linguistic information and the convenience they present to neural network learning strategies are some of the attractive characteristics of neurofuzzy systems. The neurofuzzy tool life model presented in this thesis does not in fact depend on any arbitrary assumption, regarding the shape of the input-output relationship between cutting process inputs and tool life, such as in cases of tool life modelling based on empirical formulae or cutting process theory. To demonstrate this point, after the training procedure based on the training data set derived from the regression model [Alamin 1996] has been completed, an hypothetical case of significant model mismatch is considered. The ANFIS model is shown to capture with remarkable ease and satisfactory accuracy the new input-output mapping, requiring a very low number of training iterations. In addition, no statistical assumptions are required in order to derive the tool life model. The system is essentially data driven, but it is the nature of fuzzy inference systems to be capable of incorporating human linguistic knowledge, when the data availability does not allow the development of completely data-driven solutions.

7.2 ANFIS tool life model learning

The initial state for ANFIS learning was the 52 fuzzy rules model derived during the structure identification stage. ANFIS parameter optimisation has been carried out by the hybrid learning algorithm described in chapter 4.

7.2.1 Pattern sets

The ANFIS training was based on the same training pattern set with the initialisation. This set consists of 14262 input-output pairs, generated utilising the results of tool life modelling work based on multiple regression, recently carried out at Durham University [Alamin 1996], as explained in the previous chapter. The set was randomly split into two subsets of 7131 patterns, which were both used as a training and checking set in turn. The two subsets of the training pattern set are referred to as training set 1 and training set 2. They both cover the same input/output space. However, they contain different patterns to allow for cross-validation during the learning procedure.

The learning state has also been cross-checked by employing a third pattern set, consisting of 4978 cutting data patterns, which is the same with the checking set used during the initialisation phase described in the previous chapter. This set has never been employed as a basis either for initialisation or for learning. It has been used only for validation purposes. Enrolling separate training and checking sets is a basic form of regularisation, which constraints the search space to avoid solutions of high variance, i.e. overfit. Regularisation theory has been introduced in 1963 by Tikhonov as a method of solving ill-posed problems [Haykin 1994]. Within the context of function approximation, the basic idea of regularisation is to stabilise the solution by imposing smoothness constraints in the input-output mapping and thereby transforming an ill-posed problem into a well posed one [Girosi et al. 1995]. Early stopping of the training procedure based on cross-validation has shown to be equivalent to implicit regularisation [Sjoberg et al. 1995]. Cross-validation results to a more biased solution towards smooth models. In practice, simultaneous minimisation of both bias and variance in system identification is not feasible, since it would require infinitely large learning sets. Therefore bias is a price usually paid for reducing model variance [Haykin 1994]. This ensures the model's ability to generalise and not only to perform a pattern matching task.

7.2.2 Computing platform

The computing platform used for the learning procedure was the Fuzzy Logic Toolbox (v1.0) of Matlab (v4.2c1) on a Pentium PC 166MHz with 32MB RAM. The toolbox offers a very efficient means of rapid prototyping fuzzy logic systems and is equipped with a comprehensive graphical user interface environment to facilitate design and evaluation. The hybrid learning algorithm described in chapter 4 has been employed for ANFIS tool life model training. Even though ANFIS training required a small number of iterations, the computation performance of the platform was found to be very slow for large sizes of networks and training sets.

7.2.3 Learning progress and pattern outliers

The learning progress is illustrated in Figures 7.1-7.2. The figure 7.1 shows the root mean square error (RMSE) reduction throughout training, while figure 7.2 depicts the mean absolute error as a percentage of the tool life value of the regression model. An epoch is completed when the whole pattern set is fed into the network for updating the ANFIS parameters. The training is carried out in small batches of epochs. After each training batch is completed, the training set is then employed as checking set for the next batch and vice versa. The selection of appropriate learning rate for equation (4-20) is of crucial importance. The need for a low learning rate is common in batch mode learning cases, i.e. when the network parameters are updated after the whole set of patterns have been presented to the network as it is the case with the present algorithm. A high learning rate in batch-mode training often causes instability [Qin et al. 1992]. Therefore a low initial learning rate of 0.005 was selected, which was further reduced as training proceeded to assist convergence.

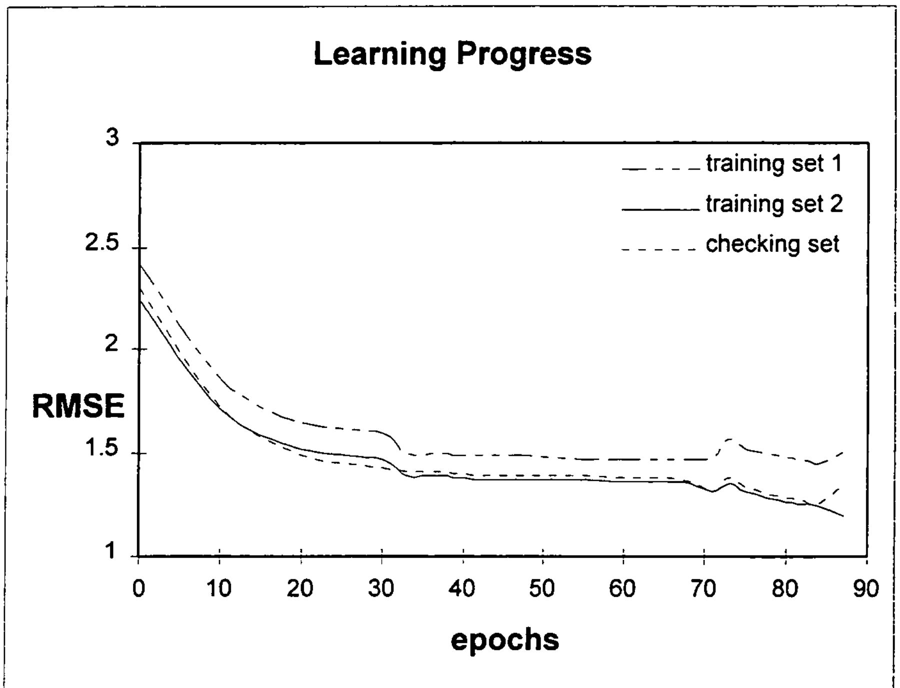

Figure 7.1: Root mean square error reduction during training

The RMSE appears to be lower in the case of the checking set. This is due to the fact that the independent checking set is not a randomly generated set but a well constructed series of tests, spanning a large area of the input space, where the model should be required to show its best performance. Therefore the existence of statistical outliers which exhibit large squared errors is less likely in the checking set than in the randomly generated training set.

The presence of statistical outliers in multivariate data is not a trivial problem. Generally speaking, in high dimension spaces a very small fraction of outliers can significantly degrade modelling accuracy [Rocke and Woodruff 1996]. In system identification tasks, outliers are related to very high squared error values. Estimation algorithms can be trapped in attempting to fit the outliers, instead of fitting the rest of the data. In fact, large squared error values are not always associated with the presence of statistical outliers and therefore appropriate choice of the error measure is needed in order to avoid confounding statistically well-behaved data with outliers. A distance measure that can be robust with respect to the presence of outliers is the Mahalanobis distance [Nadler and Smith 1993] between points in *n*-dimensional space:

$$d_{\Omega}^{2}(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^{T}\mathbf{\Omega}(\mathbf{x}-\mathbf{y}) \tag{7-1}$$

where **x**, **y** are points in the n-dimensional space and $\mathbf{\Omega}$ is a positive definite symmetric *nXn* matrix. Obviously, equation (7-1) reduces to the simple Euclidean distance when $\mathbf{\Omega}$ is the unity matrix. The pair of the distance measure and the matrix that defines it is often referred to as a metric. The determination of a good metric that is immune to the presence of statistical outliers has shown to be very difficult in high dimensional spaces [Rocke and Woodruff 1996]. When matrix $\mathbf{\Omega}$ is constant over the space, the metric is said to be global, otherwise local metrics are defined. For the purposes of the present work, a few training patterns which exhibit very high squared errors after initial training, will be treated as outliers and will be removed from the training set, to allow for further reduction in the cost function over the rest of the patterns.

The learning process progresses rapidly during the first 35 epochs (Figures 7.1-7.2). The mean absolute error over the checking set does not follow the same trend and exhibits a peak just after the twentieth iteration. After the first 35 epochs the error reduction is been retarded without any significant improvement in the cost function. This however is not an indication that the network mapping capacity has been exhausted. Careful examination of the training set reveals that the reason for the learning process becoming stuck at this stage is the presence of very few patterns, with exceptionally high squared error values. In particular, the size of this error is of one or two orders of magnitude higher than the squared error of the vast majority of the patterns.

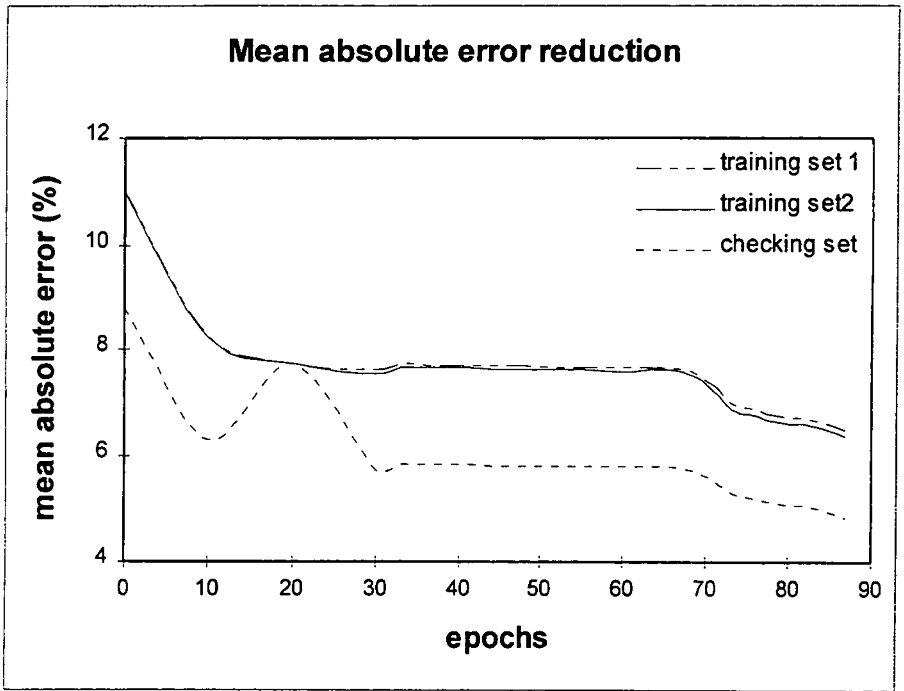

Figure 7.2: Mean relative absolute error reduction during training.

These few patterns, or outliers, correspond to boundary values of the feed rate (Table 6.2) and cutting speed (Table 6.3) ranges and are shown in Table 7.1. This should not come as a surprise, considering the source of the training cutting data, i.e. local models of tool life derived by multiple regression [Alamin 1996]. The functional mapping of these models is non-smooth at the boundaries separating the tool life groups. In addition, most of the patterns identified as outliers correspond to feed rate values that lie within the overlapping regions defined for finishing and medium roughing (0.25 - 0.35 mm/rev), as well as for medium roughing and roughing (0.4 - 0.5 mm/rev).

The definition of overlapping regions of feed rate for different types of cut is not wrong in principle. For instance, a cut with a feed rate of 0.26 mm/rev may be considered both as a finishing or medium roughing cut, depending on the depth of cut. However, this distinction is not supported by the cutting data derived from the regression model [Alamin 1996] and therefore the input-output relationship is poorly defined for some small regions of the input space. These regions appear to exhibit the highest squared errors. Even though their absolute error as a percentage of the regression model tool life value is not of a different order of magnitude than the average absolute error of the whole training set, the outliers dominate the training process by having very large corresponding "deltas" during the error backwards propagation procedure of the hybrid learning algorithm. As mentioned in chapter 2, these deltas are defined for each node as a function of the derivative of the activation of the node. Since the target mapping (i.e. the regression model) is non-smooth in the tool life groups boundaries, these deltas can acquire very high values. It is important to note that this behaviour is related to the particular source of cutting data and is less likely to be observed on approved real life data.

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)	ANFIS Output (min)	Squared Error
5	3	35	0.4190	97.27	72.96	39.75	1103.04
1	2	20	0.2622	228.99	172.11	139.56	1059.14
1	2	20	0.2934	212.25	179.68	152.78	723.20
5	3	35	0.4521	96.79	61.20	36.85	592.95
1	2	20	0.2871	219.89	162.60	139.09	552.82
5	2	35	0.3250	100.95	178.12	154.70	548.37
1	2	20	0.2851	223.47	154.05	132.78	452.52
1	2	20	0.2804	226.73	150.90	129.77	446.52
5	2	35	0.3273	101.52	170.61	150.19	416.76
3	3	35	0.4063	121.68	83.57	63.51	402.62
5	3	35	0.4640	97.78	54.64	35.24	376.46
3	3	35	0.4367	120.67	72.61	55.52	292.14
1	2	20	0.2588	245.97	129.49	112.66	283.40
1	2	20	0.2585	246.45	128.82	112.18	276.79
5	2	35	0.3165	104.35	165.18	149.02	261.01
3	2	20	0.2660	170.42	167.80	151.64	260.96
5	3	35	0.4278	105.13	48.77	34.01	217.97
5	3	35	0.4319	105.37	47.08	33.48	184.81
5	3	35	0.4176	107.85	46.32	33.23	171.34
3	3	35	0.4406	123.18	64.81	51.87	167.44
2	3	35	0.4162	146.27	87.02	74.58	154.86
5	2	35	0.3461	102.25	142.89	130.73	147.91
3	2	35	0.2650	150.17	156.95	144.96	143.95
3	2	35	0.4771	229.52	4.62	16.17	133.39
3	2	35	0.3070	136.24	164.87	154.20	113.71
9	3	35	0.8563	70.49	52.80	42.48	106.63
5	2	20	0.2535	131.69	130.17	119.91	105.38
9	3	35	0.7889	70.65	52.34	42.08	105.37
3	2	35	0.4234	229.06	6.43	16.50	101.43
1	2	35	0.4851	186.92	60.26	69.30	81.69
5	3	35	0.5526	95.81	38.01	29.15	78.40
3	2	35	0.4130	227.68	7.07	15.91	78.32
3	2	35	0.3884	229.48	8.04	16.88	78.18
9	3	35	0.6583	70.62	52.43	43.61	77.83
3	2	35	0.4497	225.65	5.85	14.60	76.49
3	2	35	0.4515	225.40	5.82	14.47	74.82
3	2	35	0.4824	223.31	5.08	13.50	70.84
3	2	35	0.4487	224.04	6.09	13.84	60.15

Table 7.1: Training set outliers

After the 67th iteration, 38 out of the 14262 patterns were identified as outliers (Table 7.1) and were excluded from the training pattern set. This choice was not meant to target at completely eliminating the outliers from the training set but rather than to show that ANFIS modelling accuracy can be enhanced if such outliers are removed. A significant reduction in the error over both the resulting training data set (training set 2) as well as the checking set was then noticed, while the squared error over the initial training set was slightly increased. Despite this increase, even the mean absolute error over the first training set has been reduced.

Since the training goal is the reduction of the squared error, the training procedure, before the exclusion of the outliers from the learning pattern set was in fact attempting to reduce mainly the error of the outliers which were dominating the overall process. When the learning process is allowed to proceed without been slowed down by these outliers, the mean absolute error is significantly reduced. This is because the patterns contributing to the mean absolute error are mainly those corresponding to small tool life values, where even a relatively small absolute error value may produce a large relative error. Yet, the training set still includes many patterns corresponding to boundary input/output values exhibiting rather high squared error values. Eventually, higher accuracy at the core of the input space could be achieved if the boundary accuracy requirements are slightly relaxed. A more sophisticated way of achieving that would be to introduce a robust error suppressor function [Kosko 1992]. In particular, the squared error criterion could be replaced by a robust error suppressor function of the form:

$$e' = \frac{a \cdot e}{(b \cdot e)^2 + 1} \tag{7-2}$$

where e corresponds to the non-suppressed and e' to the suppressed error and a, b are parameters that can be adequately defined to determine the range of influence and the magnitude of the error suppression. Such a function has been found beneficial in suppressing extremely high squared error values relevant to the presence of outliers in large random pattern sets for backpropagation training of feed forward neural networks with sigmoid activations [Emmanouilides and Petrou 1997]. Employing such an error suppressor formula in the case of ANFIS training would involve deviation from the linearity for the consequent parameters. Therefore, appropriate modification of the hybrid training algorithm would be required. For instance, the consequent parameters could also be identified by gradient descent. However, this would inevitably result in a slower training procedure.

Training results are presented later on this chapter. These results correspond to the network state after 87 epochs. The curves slope in Figs 7.1-7.2 indicate that further improvement in the network performance should be expected after the exclusion of the outliers from the pattern set. However, the training procedure under the present hardware/software configuration is quite slow, since the whole training process lasted approximately 96 hours of uninterrupted operation. One further step that could speed up training time would be an upgrade either in hardware or software. The software improvements that could be anticipated include a newer version of the computational platform (at the time that this thesis is written, version 2.0 of the Fuzzy Logic Toolbox is due to be announced and version 5.1 of Matlab has already become available), or obtaining a fast executable version of the ANFIS training algorithm by coding it directly into C++ or by using the C compiler which is an optional additional tool for Matlab. For the purposes of the present work it is important to note that even the current learning state of the obtained tool life model reveals the representation power of the neurofuzzy architecture employed. The ANFIS tool life model developed is shown to have the capacity of capturing the complex input/output relationship for a wide range of inputs combinations. A series of representative results are presented in the next section, which demonstrate the predictive performance of the model.

It should be mentioned that several training tests have been carried out with different initial states. These included a variety of ANFIS networks with a fuzzy rule base size ranging from 38 to 73 fuzzy rules and with various Gaussian membership function shapes, apart from those described in the previous chapter. The results presented here are a good compromise between the network size and modelling accuracy. Due to the bias/variance trade-off [Sjoberg et al. 1995], additional model complexity can enhance modelling accuracy at the expense of high modelling variance, i.e. overfit. On the other hand, ANFIS models of smaller fuzzy rule base size lack the modelling capacity to achieve a mapping of sufficient accuracy.

7.3 Results and discussion

Before considering the results obtained, the way the ANFIS model is activated by a set of inputs in order to provide a predicted value for tool life is first examined. This will give a good indication of how the model interprets the inputs, whilst providing an illustration of the meaningful way in which the overall mapping is achieved.

The ANFIS tool life model derived is a fuzzy inference system with the following characteristics:

- Singleton fuzzification with Gaussian membership functions.
- Algebraic product inference.
- Consequent parts corresponding to the first order Sugeno-type fuzzy model [Takagi and Sugeno 1985] defined in section 4.2 (equation 4-5), i.e. consequent functions are linear combinations of the inputs.
- Weighted average (centroid) defuzzification.

The rule base of the derived ANFIS tool life model consists of fuzzy rules of the following form:

IF (Material Class is MF_{1i}) and (Type of Cut is MF_{2i}) and (Insert Grade is MF_{3i}) and (Feed Rate is MF_{4i}) and (Cutting Speed is MF_{5i})

THEN (Tool Life is *f*{Material Class, Type of Cut, Insert Grade, Feed Rate, Cutting Speed})

where

- MF_{ji} denotes the Gaussian membership function associated with the *i*-th fuzzy rule and the *j*-th input, j=1,2,3,4,5 for the input variable material class, type of cut, insert grade, feed rate and cutting speed respectively.
- The input variables material class, type of cut, insert grade, feed rate and cutting speed are quantified as described in section 6.2.2.2.
- The consequent function $f\{\cdot\}$ is a linear function of the inputs.

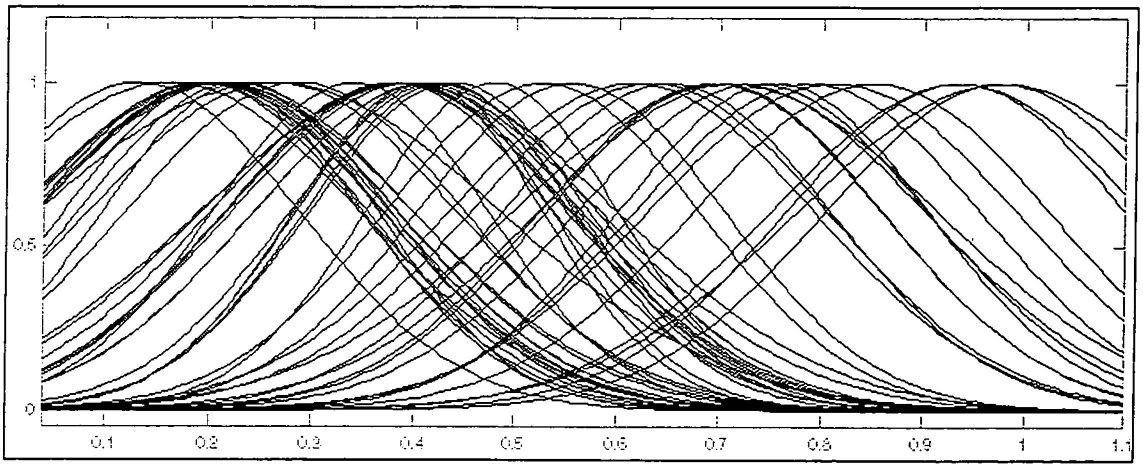

Figure 7.3: Feed rate membership functions after training

As explained in the previous chapter, each training data point attains specific discrete values in the input dimensions of material class, type of cut and insert grade. Because of the lack of cutting data patterns acquiring values between these discrete points, the gradient search through the parameter space can not move the initial membership functions far from their initial positions. The final shapes of the membership functions for these input variables remain practically unchanged after training and are similar to those shown in Figures 6.7-6.9. The final membership functions for the feed rate and cutting speed are depicted in Figures 7.3-7.4.

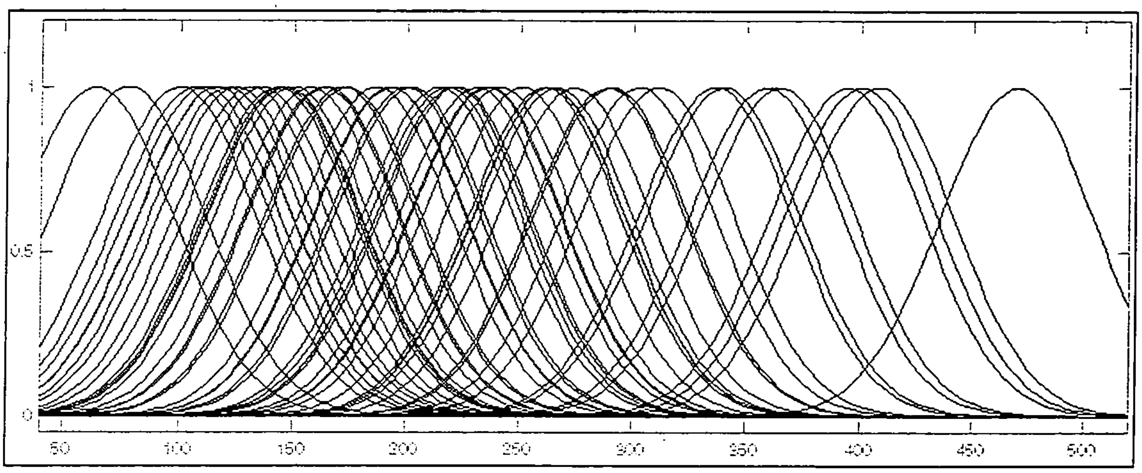

Figure 7.4: Cutting speed membership functions after training

7.3.1 The rule firing mechanism

Depending on the positioning of the core of the premise membership functions, each fuzzy rule can be regarded as relevant to a particular tool life group. Accordingly, the correspondence between fuzzy rules and tool life groups is summarised in Table 7.2, where the way that the fuzzy rules are activated for a given combination of inputs is illustrated. The grey shaded areas correspond to defined tool life groups. For each one of these groups tool life data were included into the training and checking sets.

Material Class	Finishing			Medium Roughing			Roughing		
	P10	P20	P35	P10	P20	P35	P10	P20	P35
1	15	33	7	16	52	40	25	39	42
2	8	3		6	20	30,47	35	27	19,51
3			29	31	36		13	48	
4	9	1	4	2		22,41	43	10	14
5	24		38		11,49		21	37	44,50
8		26	18		17	28		32	23
9						5			45
10			12,34			46			

Table 7.2: Correspondence between tool life groups and fuzzy rules; the numbers indicate fuzzy rules relevant to a specific combination of material class, type of cut and insert grade. The grey shaded areas correspond to defined tool life groups.

The fifty two fuzzy rules derived have premise membership functions whose core correspond to certain tool life groups. A fuzzy rule number appearing at a specific cell, implies that the core of the antecedent membership functions of this rule corresponds to this particular combination of material class, type of cut and insert grade, indicated by the position of the cell. For instance, finishing free cutting steel (material class 2) with P10 grade, invokes the 8th fuzzy rule to fire. The firing strength depends also on whether the feed rate and cutting speed values correspond to the core of this rule premise membership functions. It is worth noting that several tool life groups lack a clear representation within the fuzzy rule base. However, these input space regions are computationally supported by the fuzzy rules which are relevant to adjacent inputs combinations. On the other hand, some tool life groups are represented by more than one fuzzy rules.

Figures 7.5-7.6 illustrate the fuzzy rules firing mechanism and the output provided by the ANFIS tool life model for the following set of inputs:

- Material class: Free cutting steel (class 2).
- Type of cut: Finishing.
- Insert grade: P10.
- Feed rate: 0.15 mm/rev.
- Cutting speed: 415 m/min.

The cutting speed value has actually been defined as the upper boundary value for this type of operation (Table 6.3). The expected tool life according to the regression model is 13.76 min, while the ANFIS model yields 13.6 min which corresponds to a 1.16% relative modelling error. This set of inputs triggers 7 out of the fuzzy rules with varying firing strengths, shown in Table 7.3.

Fuzzy Rule	Firing Strength	Strength Order
3	0.0179	5
6	0.0047	6
8	0.8739	1
9	0.0001	7
15	0.1269	2
16	0.0268	4
33	0.1088	3

Table 7.3: Fuzzy rules firing strength for finishing free cutting steel with an ISO P10 insert

In figure 7.5, a screen damp of the rule viewer window of the Fuzzy Logic Toolbox illustrates the complete set of fuzzy rules. A more detailed insight in the rule firing mechanism is provided by figure 7.6, where only the most relevant fuzzy rules are depicted, sorted according to firing strength order. This figure demonstrates an important point, related to the representation power of neurofuzzy systems. That is the ease of interpreting how the model actually achieves the input-output mapping.

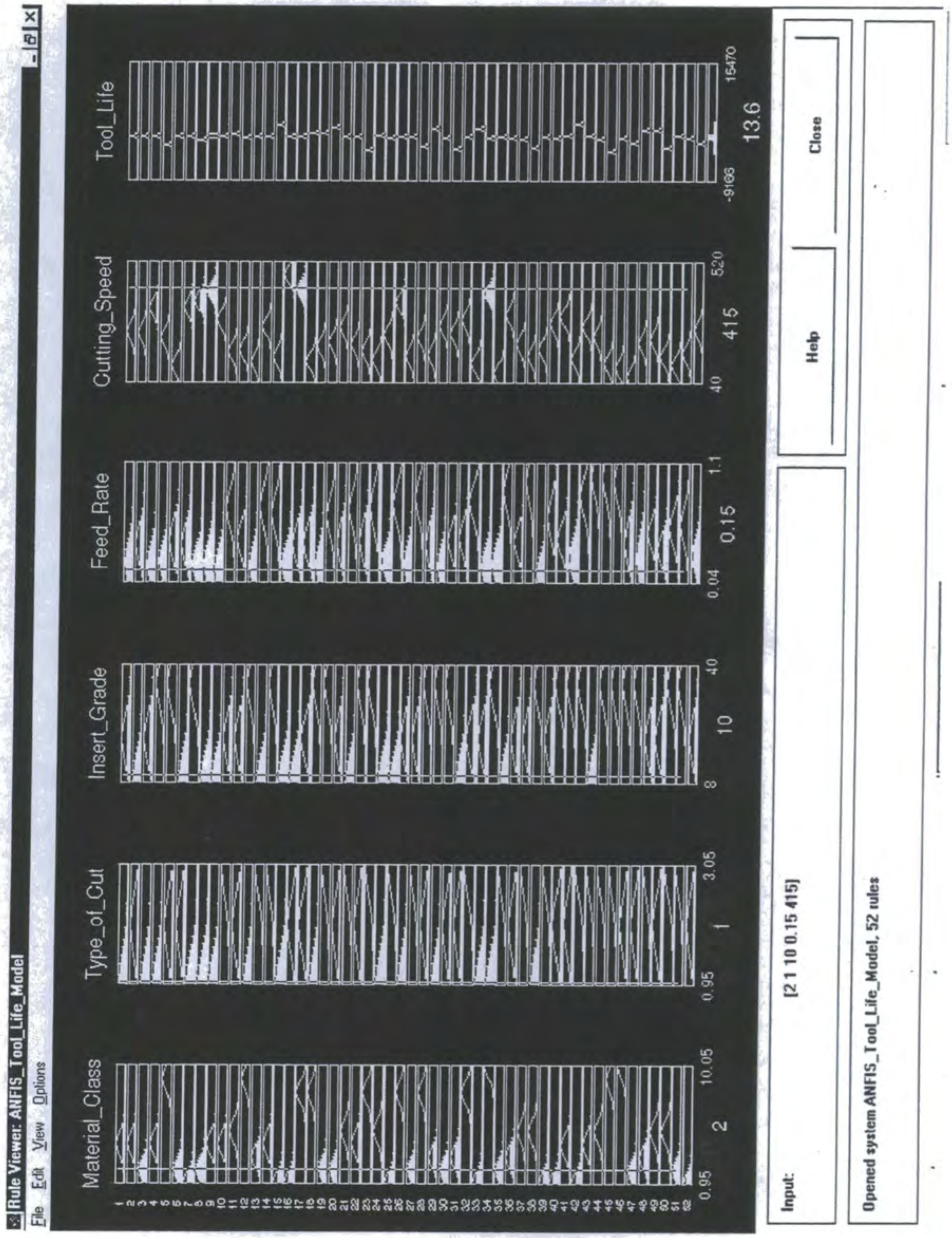

Figure 7.5: Fuzzy rules firing for finishing free cutting steel with an ISO P10 insert

For instance, all five premise membership functions of the 8th fuzzy rule (1st in Figure 7.6) are almost fully matched. In contrast, in the case of the 15th fuzzy rule (2nd in Figure 7.6) the type of cut, insert grade and feed rate membership functions are almost fully matched, whereas both the material class and cutting speed membership functions are only partially matched. In the example considered, the degree of compatibility of the inputs to the premise parts of the rest of the fuzzy rules is even smaller and is practically zero for any other rule not listed in Table 7.3.

It is evident from the above that, even though a single model is derived for the whole range of inputs, it is still straightforward to recognise which exactly rules are activated for any combination of inputs. Therefore, neurofuzzy models provide with a transparent input-output mapping. This feature is not shared by other black-box modelling tools, such as the popular feed forward neural networks with sigmoid activations, or by polynomial approximators. Because of their transparency, neurofuzzy inference systems are often called grey-box models. When a neurofuzzy model performs poorly in some region, modelling accuracy can be improved by simply adding an extra fuzzy rule to cover this region, or by splitting an existing rule into two [Kosko 1997]. The best place to place a new fuzzy rule is where it most reduces the approximation error between the fuzzy system output and the approximand function.

A different example is illustrated in Figures 7.7-7.8. The inputs to the ANFIS model are:

- Material class: Difficult to machine stainless steel (class 10).
- Type of cut: Medium roughing.
- Insert grade: P35.
- Feed rate: 0.4 mm/rev.
- Cutting speed: 90 m/min.

The regression model suggests a tool life of 8.33 min, while the ANFIS network predicts 7.52 min. Again, figure 7.7 illustrates the complete set of fuzzy rules, wheareas in figure 7.7 the most relevant fuzzy rules are shown, sorted according to firing strength order. The fuzzy rules firing strengths are shown in Table 7.4.

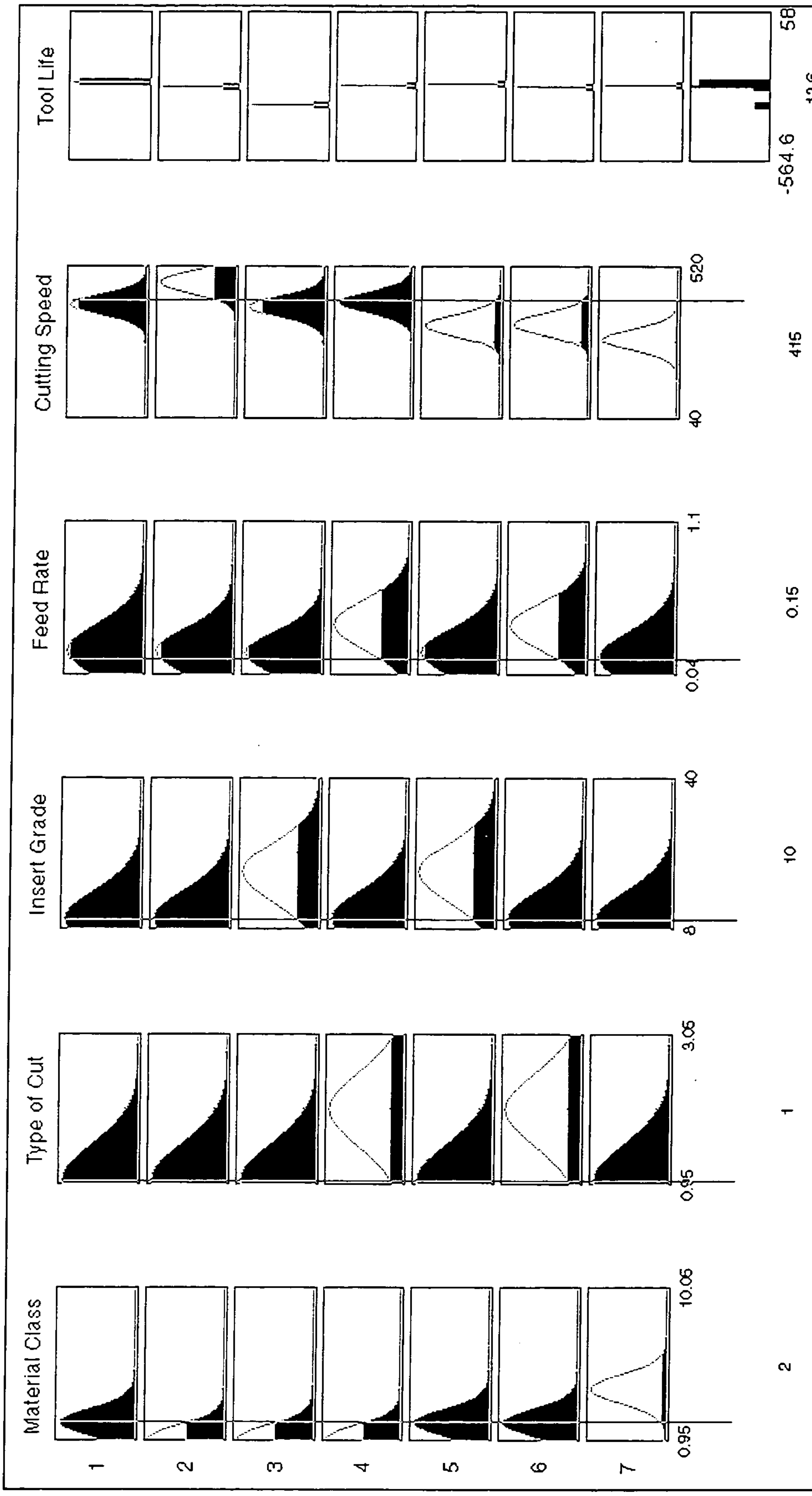

Figure 7.6: Firing of most relevant fuzzy rules for finishing free cutting steel with an ISO P10 insert (rules sorted according to firing strength).

Fuzzy Rule	Firing Strength	Strength Order
5	0.4449	2
12	0.024	4
17	0.0019	6
18	0.0001	8
23	0.0014	7
26	0.0001	9
28	0.0124	5
34	0.0784	3
46	0.7354	1

Table 7.4: Fuzzy rules firing strength for medium roughing difficult to machine stainless steel with an ISO P35 insert

Here the input parameters material class and insert grade are such that the ANFIS mapping receives computational support from a few extra fuzzy rules which correspond to membership functions relatively close to those of the fuzzy rules which fire at full strength.. In particular, rule 46 is shown to be activated at almost full strength, as all its premise parts are nearly fully matched. Computational support is provided mainly by the 5^{th} rule, whose all premise parts apart from the material class are almost fully matched. In other words, the 5^{th} fuzzy rule, which corresponds to medium roughing moderately difficult to machine stainless steel (material class 9) is shown to be partially activated.

This is well justified rule firing, since the material classes 9, 10 are adjacent. Cutting performance on moderately difficult to machine stainless steels should present similarities with the performance on difficult to machine stainless steels. In addition, the cutting speed value of 90mm/rev belongs also to the cutting speed range defined for medium roughing moderately difficult stainless steels. To a much lesser extent, rules 12, 17, 28 and 34 also fire. The rest of the fuzzy rules firing strength is practically zero. Having examined both of the above examples, it becomes clear that the whole firing mechanism is activated in a quite straightforward and meaningful way. In a neurofuzzy inference system that has been adequately trained to perform an input-output mapping, the overall modelling task is accomplished in a natural way, i.e. in the form of fuzzy IF-THEN rules, which is easily interpreted by a human.

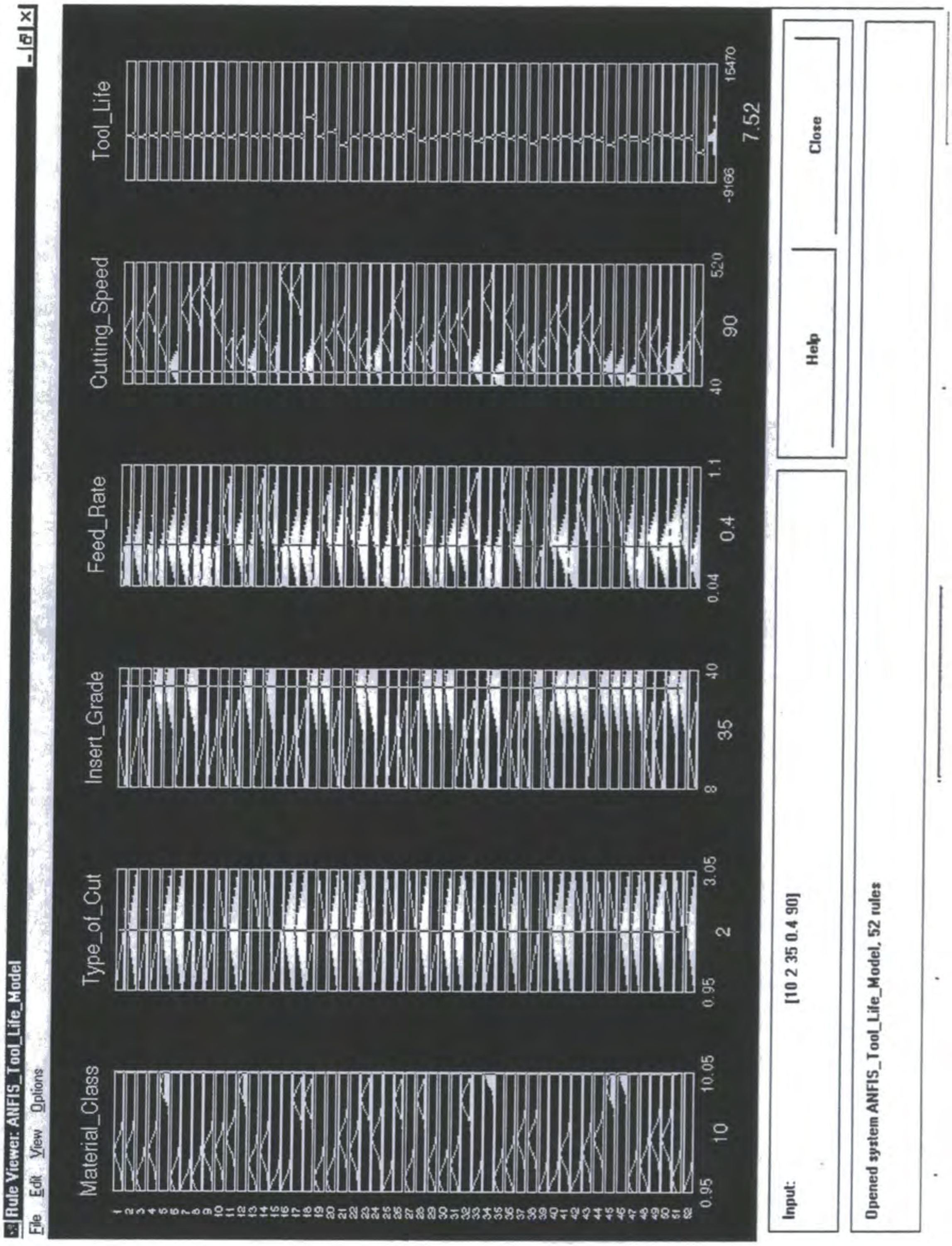

Figure 7.7: Fuzzy rules firing for medium roughing difficult to machine stainless steel with an ISO P35 insert

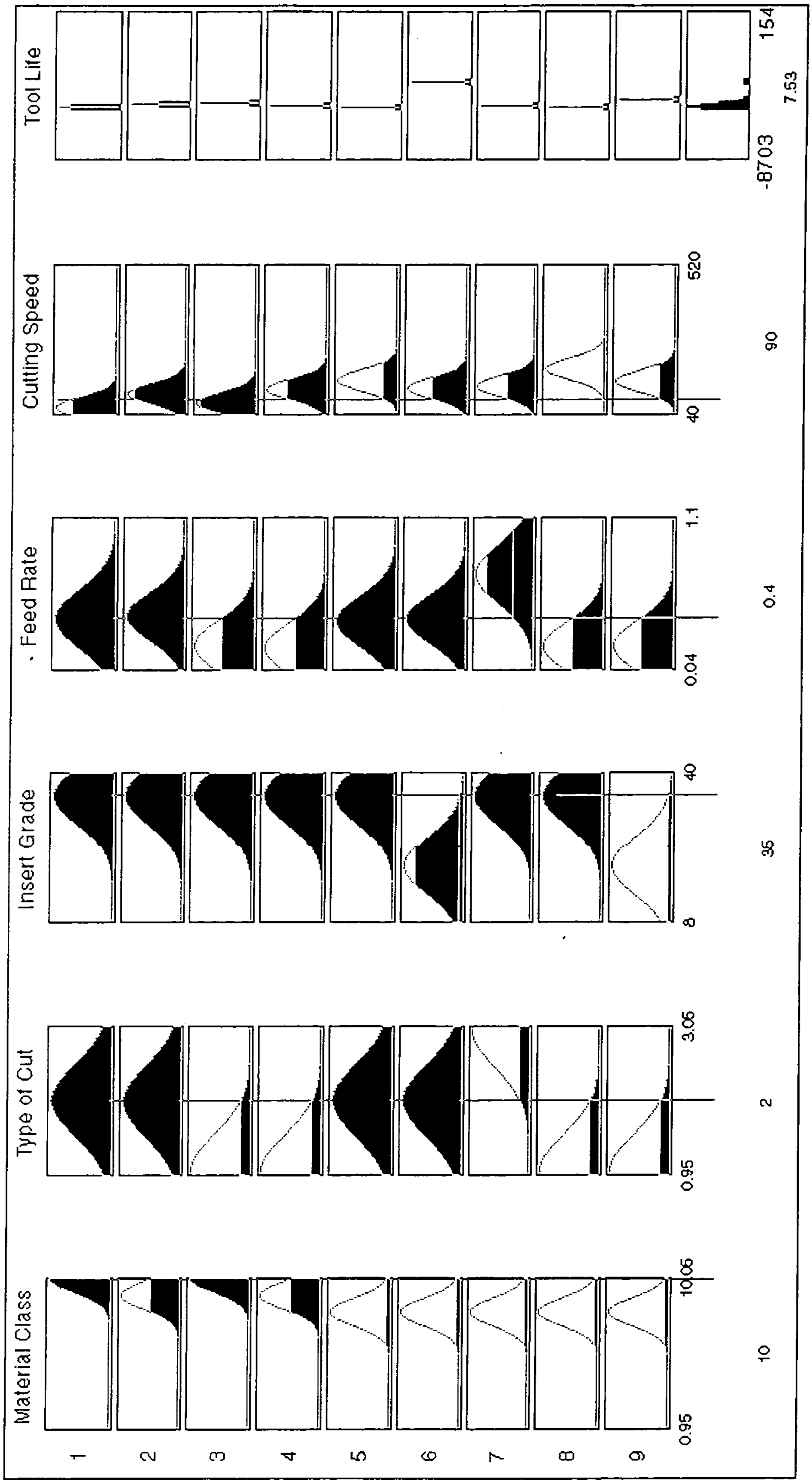

Figure 7.8: Firing of most relevant fuzzy rules for medium roughing difficult to machine stainless steel with an ISO P35 insert (rules sorted according to firing strength)

It is worth noting that a neurofuzzy model, such as the one described in this thesis, requires the existence of a well-constructed training data set in order to come up with a completely data driven model. This ensures that there is sufficient information to support the creation of at least one fuzzy rule having its premise parts activated at every valid combination of inputs. Thus, the possibility of having a range of inputs that fail to activate any of the fuzzy rules can be eliminated. Less dependence on data availability can be achieved by intuitive definition of the fuzzy rules, based on available expert knowledge. For instance, it is rather straightforward to add an extra fuzzy rule to be activated at a predefined range of inputs. This should be done when it is felt that the available data is not rich enough to span the particular region, but some rough guess can be made about the expected outcome, based on existing expert knowledge. In many occasions, and in particular in machining practice, an expert can often provide with a vague guess about the anticipated tool life, usually in linguistic terms such as short, medium, long etc. Such an intuitive guess should suffice to initially define additional fuzzy rules. The parameters of this rule can later be optimised on the basis of even scarce available machining data.

7.3.2 Indicative results

Having examined the rule firing mechanism, some indicative results are now presented, covering a wide range of inputs combinations. Figures 7.9-7.15 illustrate the ANFIS model tool life predictions versus cutting speed variations, compared to those of the regression model employed for the derivation of the training tool life data [Alamin 1996]. It should be emphasised once again that the modelling accuracy is tested on cutting data never employed either for ANFIS initialisation or for training. Therefore, the good mapping performance achieved indicates that the ANFIS tool life model can generalise the knowledge contained in the training patterns and not only perform a pattern matching task.

First the ANFIS tool life predictions for cutting free cutting steel with a TP10 insert are shown in Figure 7.9. Results are presented for all types of cut. In all cases the ANFIS mapping performance is highly accurate, exhibiting root mean squared error (RMSE) values well below unity and mean relative absolute error between 0.8% and 2.6%.

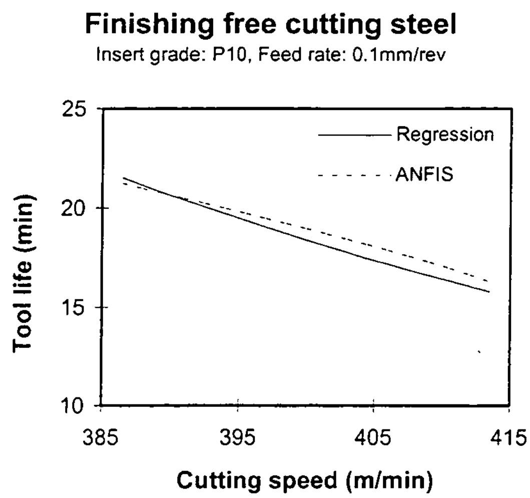

Figure 7.9a: RMSE=0.51, mean abs.err=2.59%

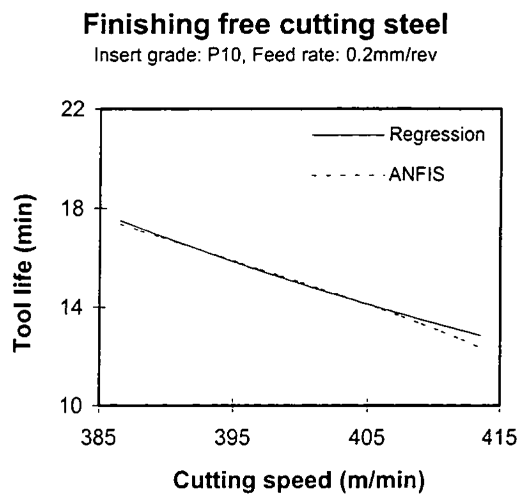

Figure 7.9b: RMSE=0.17, mean abs. err.=0.81%

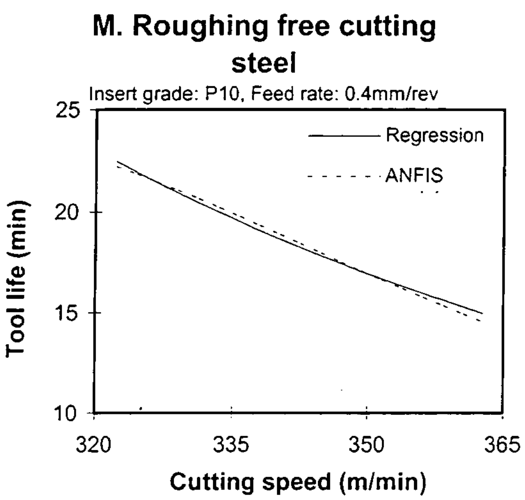

Figure 7.9c: RMSE=0.21, mean abs.err=1.05%

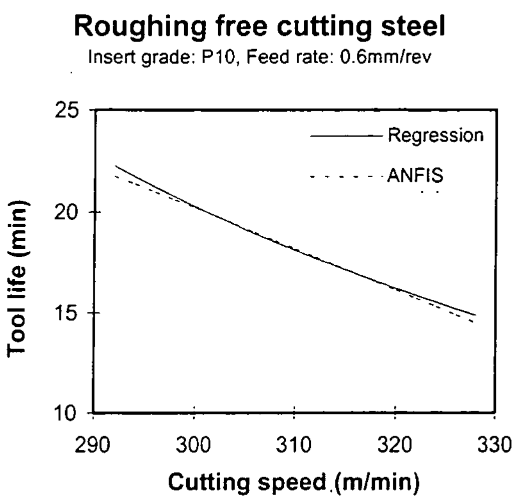

Figure 7.9d: RMSE=0.20, mean abs.err=0.83%Figure

7.9: ANFIS tool life predictions for turning free cutting steel with an ISO P10 insert

Figure 7.10. Again, the ANFIS performance is very good in all cases of type of cut, i.e. finishing, medium roughing and roughing. The root mean square error is well below 1.0 and the mean relative absolute error ranges between 1-3%.

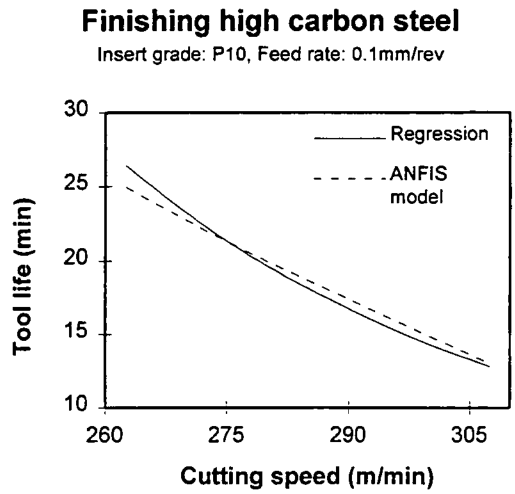

Figure 7.10a: RMSE=0.64, mean abs.err=2.93%

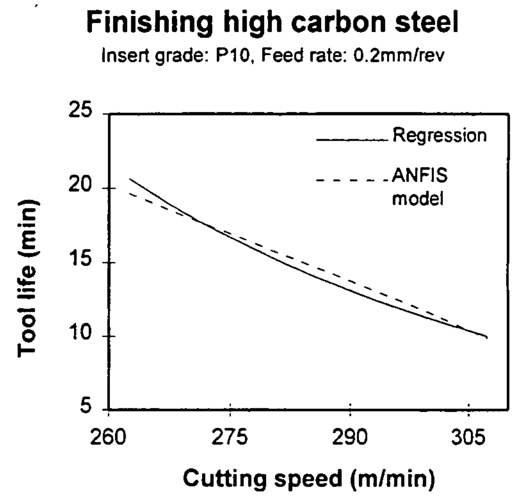

Figure 7.10b: RMSE=0.50, mean abs. err.=3.02%

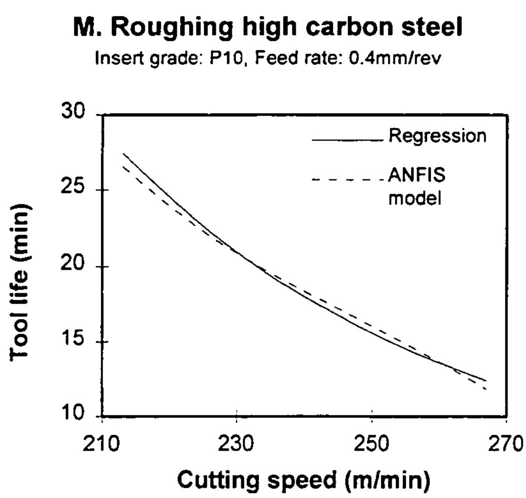

Figure 7.10c: RMSE=0.21, mean abs.err=1.05%

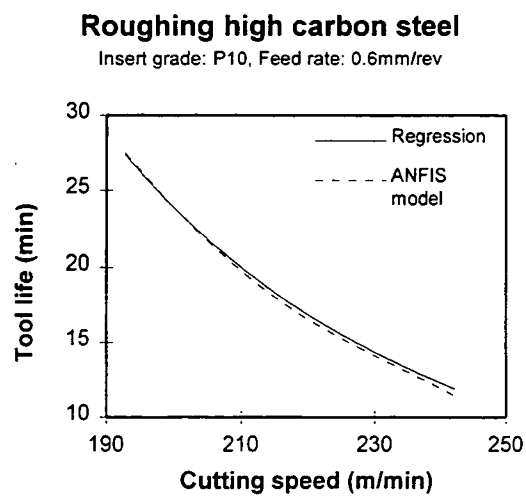

Figure 7.10d: RMSE=0.26, mean abs.err=1.47%

Figure 7.10: ANFIS tool life predictions for turning high carbon steel with an ISO P10 insert.

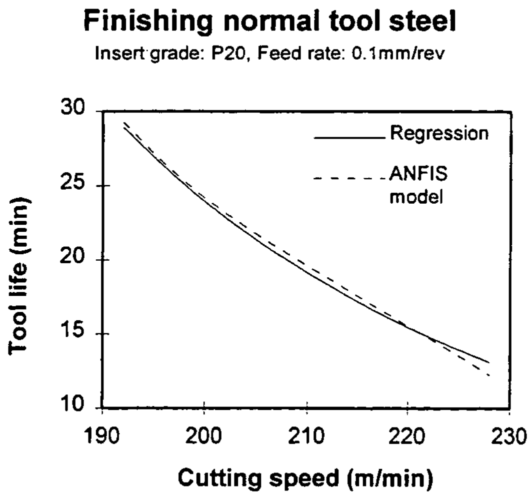

Figure 7.11a: RMSE=0.39, mean abs.err=1.93%

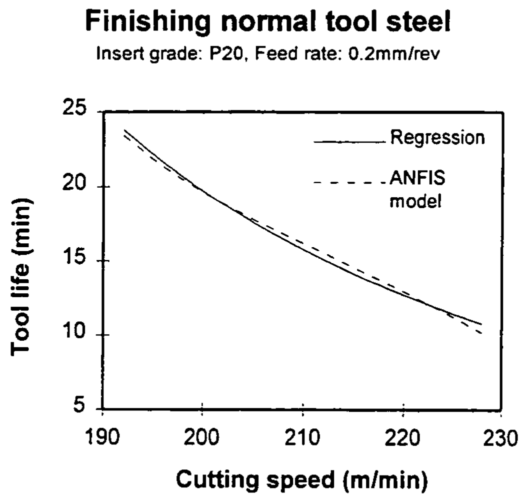

Figure 7.11b: RMSE=0.32, mean abs. err.=1.92%

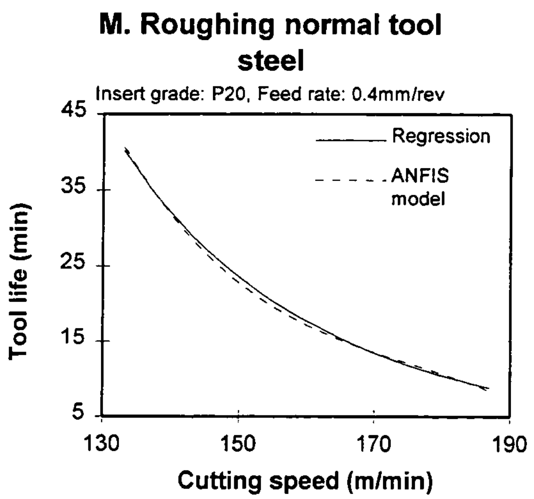

Figure 7.11c: RMSE=0.21, mean abs.err=1.05%

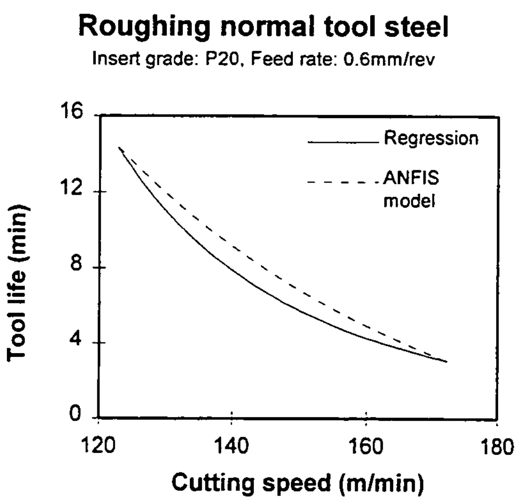

Figure 7.11d: RMSE=0.88, mean abs.err=11.86%

Figure 7.11: ANFIS tool life predictions for turning normal tool steel with an ISO P20 insert

Similar results are also obtained for finishing and medium roughing of normal tool steels with an ISO P20 insert (Figure 7.11). However, a large mean value of relative absolute error is noticed for roughing normal tool steel with the same insert grade. It is worth noting that this does not correspond to an equally high root mean square error value. This is a case of a high relative absolute error appearing in a region of small tool life values.

Equally good mapping is achieved in the case of finishing and medium roughing free cutting stainless steel with a ISO P20 insert (Figure 7.12). The root mean squared error is less than 0.5, while the mean absolute relative error is below 2.3%. Again, the accuracy is poorer for the roughing case, which presents a RMSE value of 1.12. The corresponding mean absolute relative error is 4.88%.

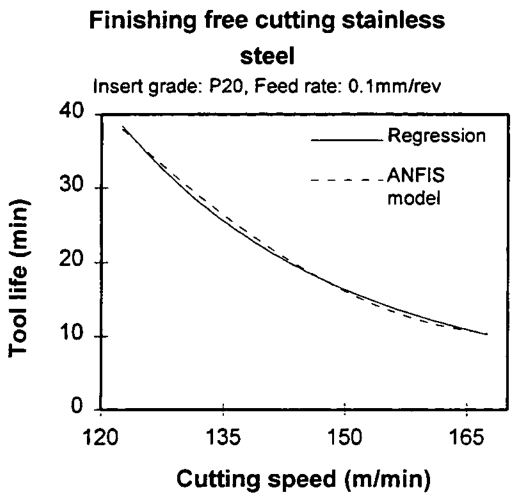

Figure 7.12a: RMSE=0.49, mean abs.err=2.25%

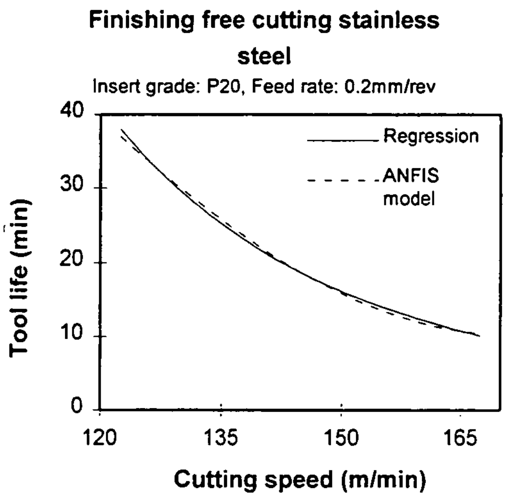

Figure 7.12b: RMSE=0.41, mean abs. err.=1.88%

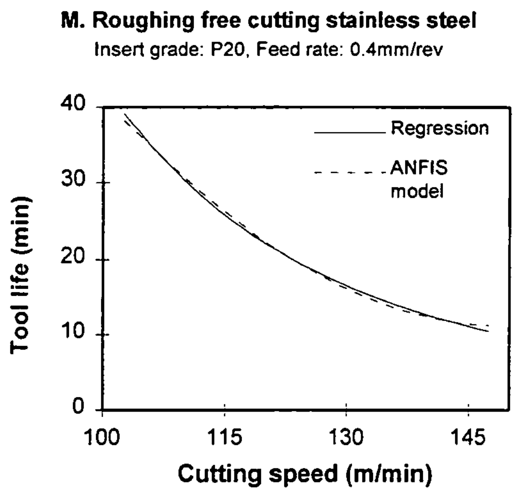

Figure 7.12c: RMSE=0.40, mean abs.err=1.88%

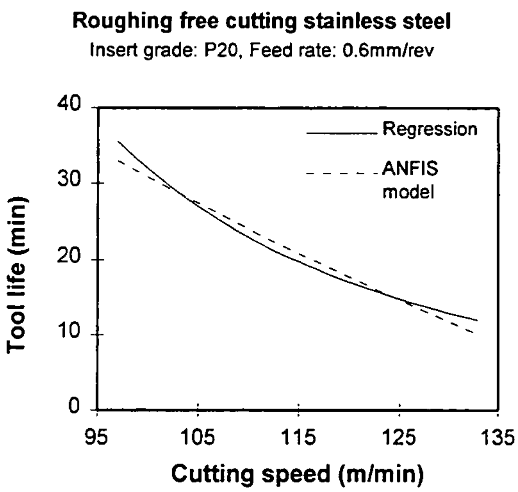

Figure 7.12d: RMSE=1.12, mean abs.err=4.88%

Figure 7.12: ANFIS tool life predictions for turning free cutting stainless steel with an ISO P20 insert

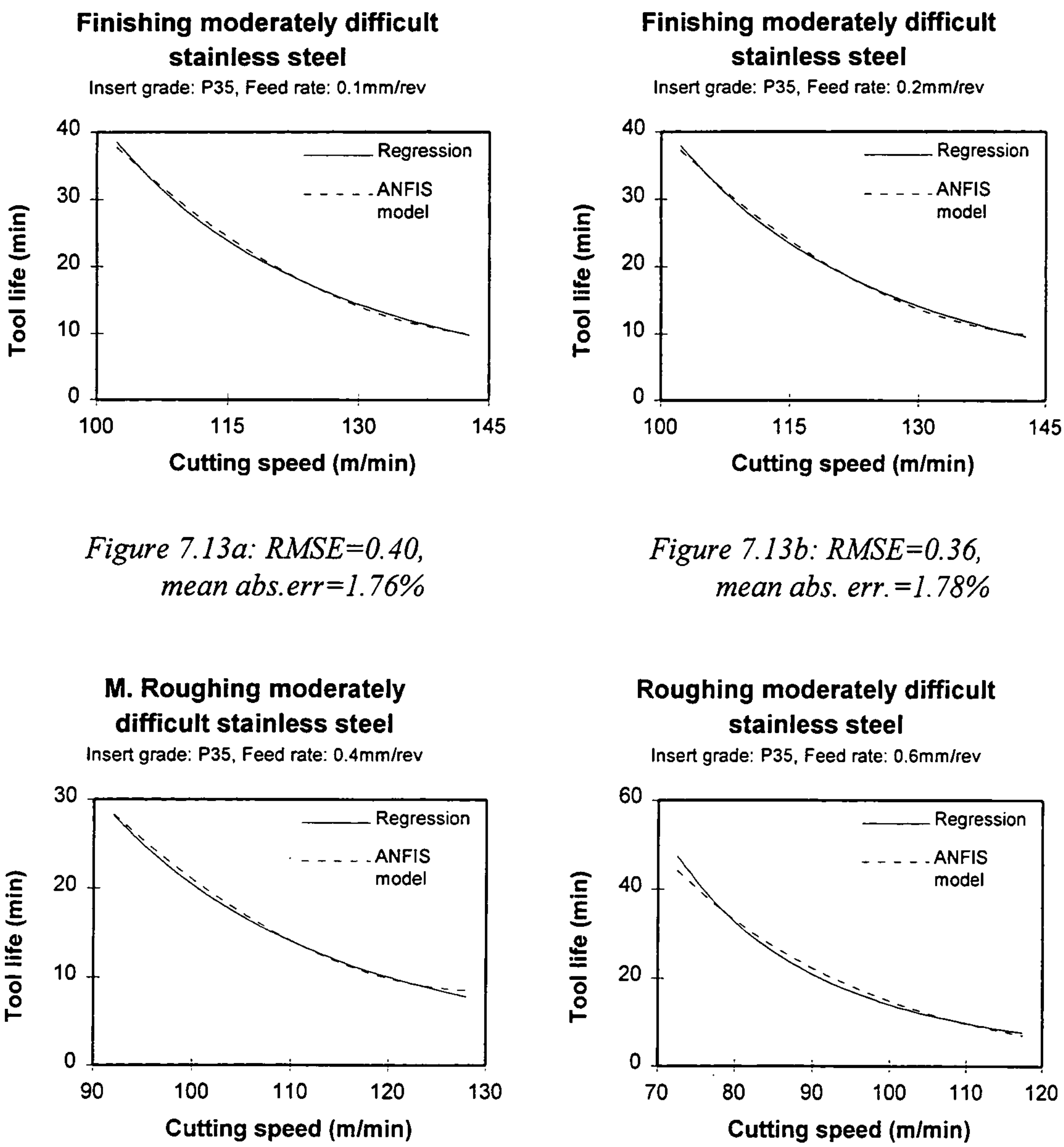

Figure 7.13a: RMSE=0.40, mean abs.err=1.76%

Figure 7.13b: RMSE=0.36, mean abs. err.=1.78%

Figure 7.13c: RMSE=0.37, mean abs.err=2.21%

Figure 7.13d: RMSE =1.17, mean abs.err=4.63%

Figure 7.13: ANFIS tool life predictions for turning moderately difficult stainless steel with an ISO P35 insert

The same remarks apply when examining the results obtained for turning moderately difficult stainless steel (Figure 7.13), as well as in the case of turning difficult to machine stainless steel (Figure 7.14) with an ISO P35 insert. In the latter case, only finish and medium roughing turning operations are defined. The mapping performance deteriorates for cutting speed values close to the upper defined boundary value for the medium roughing case in Figure 7.14.

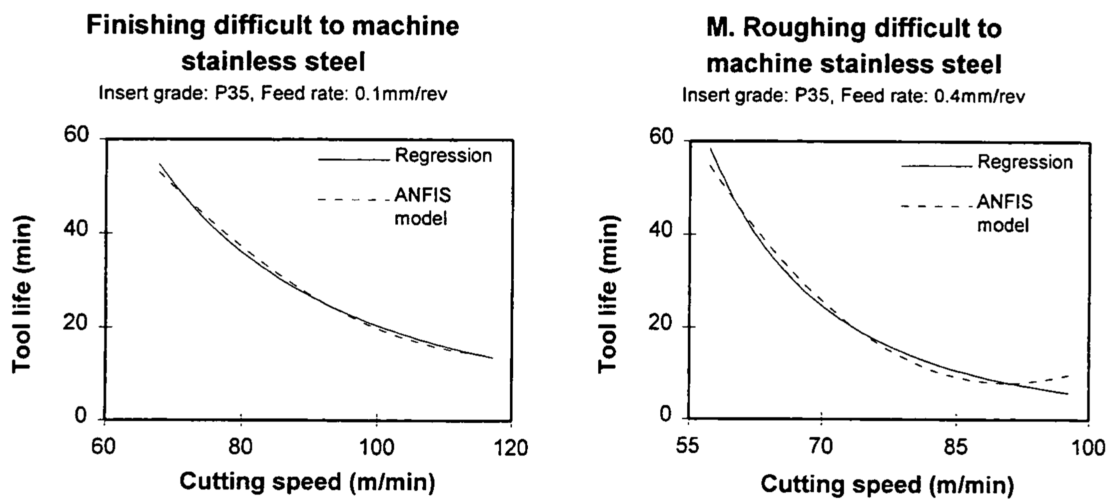

Figure 7.14a: RMSE =0.72, mean abs.err=2.20%

Figure 7.14b: RMSE =1.69, mean abs. err.=11.25%

Figure 7.14: ANFIS tool life predictions for turning difficult to machine stainless steel with an ISO P35 insert

Some examples of poor modelling performance are shown in Figure 7.15. In particular figures 7.15a-b illustrate cases of high root mean square error values appearing at cases of long tool life, while the performance in terms of the mean value of he relative absolute error is not equally poor. The opposite case is demonstrated by Figures 7.15c-d, where the RMSE values are now rather small. However, because the tool life is now short, the result is a mean relative absolute error higher than 10%.

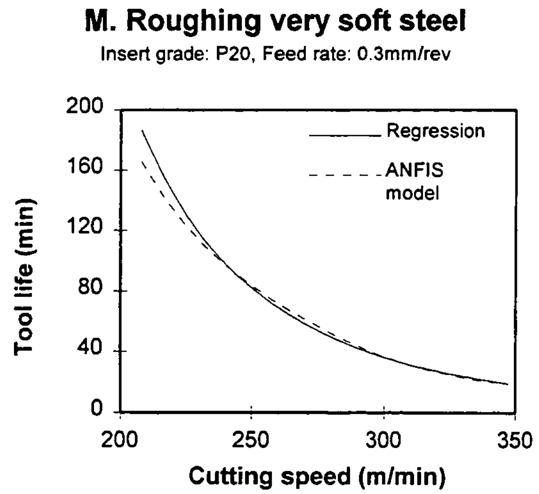

Figure 7.15a: RMSE=6.30, mean abs.err=3.28%

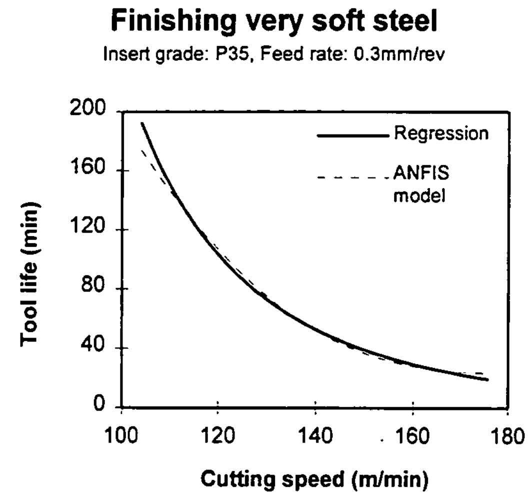

Figure 7.15b: RMSE=5.14, mean abs. err.=4.95%

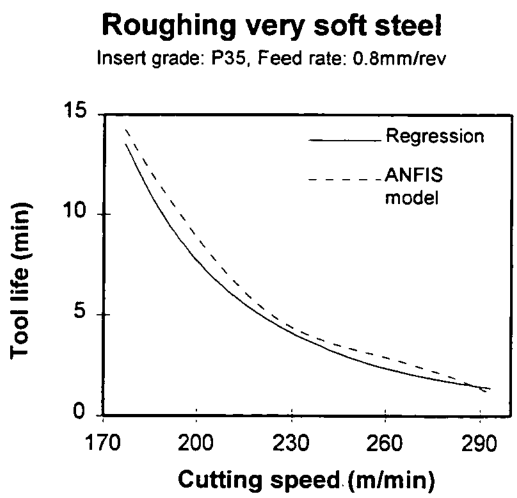

Figure 7.15c: RMSE=0.68, mean abs.err=13.39%

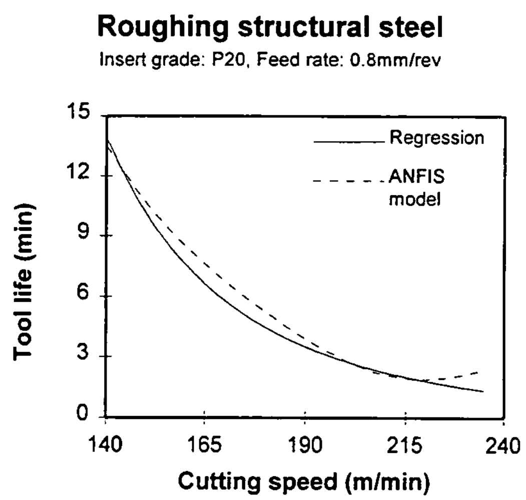

Figure 7.15d: RMSE=0.60, mean abs.err=14.32%

Figure 7.15: ANFIS tool life predictions for a range of inputs which correspond to either very long or very short tool life values

The results presented reveal that the ANFIS model obtained after 87 iterations of training, have captured the complex functional relationship between the five inputs and the output, i.e. it can predict the tool life given the material class, type of cut, inset grade, feed rate and cutting speed.

The graphs in figures 7.9-7.15 illustrate the predictive performance of the developed tool life model over a series of well constructed tests, none of them being employed either for ANFIS initialisation or as a training example during the learning phase. In most cases the model provided predictions of sufficient accuracy. The overall mapping appears to be smooth, with no abrupt prediction error variations throughout the series of tests. In other worrds, a model of low variance has been derived, i.e. no overfit has been observed. Therefore, it can be said that the ANFIS tool life model obtained offers a good compromise between modelling accuracy and model complexity. The least accurate results are obtained at or close to the boundary values of the inputs or outputs.

So far the ANFIS model has been shown capable of capturing the functional relationship between the five input variables and the tool life. The functional mapping learned is actually that of a series of local models, based on multiple regression performed over the Taylor equation (6-1). [Alamin 1996] The question is how such a model would perform against cutting data that do not necessarily conform to that equation. Such a case is examined in the next section.

7.3.3 Testing ANFIS tool life model adaptation capacity

The regression model employed for the derivation of the cutting data was shown to offer good tool life predictions in many cases [Alamin 1996]. However there is no guarantee that a functional relationship such that of Taylor equation in the form of equation (6-1) would be adequate for all cases of machining with single point cutting tools. Many other tool life equations have been suggested in the literature, as discussed in chapter 5. The neurofuzzy tool life model presented in this work is not restricted to follow the mapping produced by the regression model. It is flexible enough to capture tool life relationships which deviate from the model employed for obtaining the training data. To demonstrate this, an hypothetical case of cutting performance following a different input-output mapping than that of equation (6-1) is now considered. In particular, it is assumed that the expected tool life can be modelled by the following equation:

$$T = \frac{C}{D + v^{\frac{1}{\alpha}} f^{\frac{1}{\beta}}} \tag{7-3}$$

This is a modification of the Taylor equation (6-1) which admits an additive term, D, in the denominator. This term is assumed to receive different values for each tool life group. This equation is consistent with Taylor formula for D=0, but can lead to significant variation in the tool life response curve for non-zero values of D. The presence of an additive term in the denominator of a tool life equation has been suggested in the literature, for modelling the tool life of tungsten carbide tools at high cutting temperatures (higher than 800°C) [Mari and Gonseth 1993]. However, equation (7-3) is not introduced here to accurately model tool life and therefore there is no intention to give a physical interpretation of the model mismatch introduced. It is simply a plausible relationship between tool life and cutting conditions, upon which the ANFIS model adaptation ability will be tested.

It is now assumed that the actual machining performance observed on a shop floor machine tool is consistent with equation (7-3). The question is whether the ANFIS tool life model developed so far can actually be used for tool life prediction, based on cutting data obtained from this revised tool life model.

MATERIAL CLASS	TYPE OF CUT	INSERT GRADE	NEW MODEL PARAMETERS				CUTTING SPEED BOUNDARY VALUES	
			1/α	1/β	C	D	MAX (m/min)	MIN (m/min)
Very soft steel	Finishing	TP10	4.562	0.292	1.49E+13	2.07E+11	490	455
Very soft steel	M Roughing	TP10	3.55	0	3.27E+10	4.97E+08	430	380
Very soft steel	Roughing	TP10	3.566	0	2.50E+10	4.20E+08	390	345
Free cutting steel	Finishing	TP10	4.584	0.298	7.84E+12	1.37E+11	415	385
Free cutting steel	M Roughing	TP10	3.446	0	9.89E+09	1.95E+08	365	320
Free cutting steel	Roughing	TP10	3.478	0	8.36E+09	8.87E+07	330	290
Structural steel	Finishing	TP10	4.594	0.29	4.10E+12	6.58E+10	370	310
Structural steel	M Roughing	TP10	3.603	0	1.36E+10	1.46E+08	320	255
Structural steel	Roughing	TP10	3.42	0	3.43E+09	4.39E+07	280	245
High carbon steel	Finishing	TP10	4.581	0.355	1.41E+12	2.23E+10	310	260
High carbon steel	M Roughing	TP10	3.531	0	4.57E+09	8.80E+07	270	210
High carbon steel	Roughing	TP10	3.665	0	6.49E+09	1.06E+08	245	190
Normal tool steel	Finishing	TP10	4.626	0.344	8.56E+11	1.08E+10	265	215
Normal tool steel	M Roughing	TP10	3.552	0	2.79E+09	5.16E+07	225	175
Normal tool steel	Roughing	TP10	3.371	0	7.75E+08	1.07E+07	210	160

Table 7.5 Pattern generation for revised tool life model: Cutting speed boundary values and revised tool life equation parameters for each tool life group

MATERIAL CLASS	TYPE OF CUT	INSERT GRADE	NEW MODEL PARAMETERS				CUTTING SPEED BOUNDARY VALUES	
			1/α	1/β	C	D	MAX (m/min)	MIN (m/min)
Free cutting stainless	Finishing	TP20	4.237	0.016	2.61E+10	3.95E+08	170	120
Free cutting stainless	M Roughing	TP20	3.642	0	8.24E+08	1.17E+07	150	100
Free cutting stainless	Roughing	TP20	3.456	0	2.61E+08	3.34E+06	135	95
Free cutting steel	Finishing	TP20	4.612	0.292	4.22E+12	4.56E+10	365	300
Free cutting steel	M Roughing	TP20	4.531	2.596	1.31E+11	2.47E+09	305	175
Free cutting steel	Roughing	TP20	4.309	2.503	2.61E+10	5.17E+08	280	160
High carbon steel	Finishing	TP20	4.626	0.344	8.56E+11	1.39E+10	260	220
High carbon steel	M Roughing	TP20	4.424	2.623	1.53E+10	3.05E+08	215	120
High carbon steel	Roughing	TP20	4.504	2.616	1.45E+10	1.85E+08	195	110
Normal tool steel	Finishing	TP20	4.6	0.281	4.82E+11	6.32E+09	230	190
Normal tool steel	M Roughing	TP20	4.469	2.475	1.29E+10	2.03E+08	190	130
Normal tool steel	Roughing	TP20	4.559	2.511	1.33E+10	1.74E+08	175	120
Structural steel	Finishing	TP20	4.581	0.343	1.69E+12	2.61E+10	315	265
Structural steel	M Roughing	TP20	4.481	2.654	4.99E+10	9.76E+08	265	150
Structural steel	Roughing	TP20	4.529	2.663	4.04E+10	4.80E+08	240	135
Very soft steel	Finishing	TP20	4.584	0.302	6.91E+12	1.00E+11	415	355
Very soft steel	M Roughing	TP20	4.456	2.628	1.67E+11	3.26E+09	355	200
Very soft steel	Roughing	TP20	4.45	2.646	1.00E+11	1.96E+09	320	180
Difficult castings stainless	Finishing	TP35	2.548	0.06	2.21E+06	3.46E+04	120	65
Difficult castings stainless	M Roughing	TP35	4.307	0	2.18E+09	2.28E+07	100	55
Free cutting stainless	Finishing	TP35	4.131	0.015	4.47E+10	7.25E+08	210	165
Free cutting stainless	M Roughing	TP35	1.348	0.081	2.19E+04	4.11E+02	175	120
Free cutting stainless	Roughing	TP35	1.394	0.082	2.40E+04	3.85E+02	160	110
Free cutting steel	Finishing	TP35	4.601	0.327	2.63E+12	4.25E+10	390	335
Free cutting steel	M Roughing	TP35	4.495	2.666	7.29E+10	7.40E+08	285	160
Free cutting steel	Roughing	TP35	4.525	2.629	5.45E+10	1.05E+09	255	145
High carbon steel	Finishing	TP35	4.602	0.373	4.79E+11	5.14E+09	245	200
High carbon steel	M Roughing	TP35	4.495	2.76	1.33E+10	1.44E+08	200	110
High carbon steel	Roughing	TP35	4.365	2.164	5.28E+09	6.78E+07	180	100
Moderately difficult stainless	Finishing	TP35	4.122	0.021	7.06E+09	7.25E+07	145	100
Moderately difficult stainless	M Roughing	TP35	3.9	0	1.29E+09	1.65E+07	130	90
Moderately difficult stainless	Roughing	TP35	3.81	0	5.81E+08	7.35E+06	120	70
Structural steel	Finishing	TP35	4.607	0.318	1.06E+12	2.10E+10	285	225
Structural steel	M Roughing	TP35	4.578	2.694	4.04E+10	4.85E+08	230	130
Structural steel	Roughing	TP35	4.453	2.463	1.75E+10	3.50E+08	205	120
Very soft steel	Finishing	TP35	4.565	0.325	4.40E+12	7.91E+10	390	335
Very soft steel	M Roughing	TP35	4.434	2.666	1.04E+11	1.63E+09	330	185
Very soft steel	Roughing	TP35	4.52	2.654	1.07E+11	2.04E+09	300	170
Normal tool steel	Finishing	TP35	4.599	0.302	3.30E+11	5.80E+09	220	170
Normal tool steel	M Roughing	TP35	4.365	2.614	5.28E+09	9.24E+07	180	100
Normal tool steel	Roughing	TP35	4.482	2.6	6.18E+09	1.20E+08	165	95

Table 7.5 (cont.) Pattern generation for revised tool life model: Cutting speed boundary values and revised tool life equation parameters for each tool life group

A procedure for creating cutting data out of the new model is followed, similar to the one used in chapter 6 for obtaining the checking data set. Parameter D is randomly chosen in the region of 1%-2% of the value of C. The rest of the parameters are assumed to be the same with those of the original regression model. Table 7.5 shows the values of *D* for each tool life group. For convenience, the rest of the parameters for each tool life group are also listed.

The constant term, *D*, introduces a significant distortion in the original tool life model of equation (6-1). Yet, it does not affect the physical relationship that exists between the cutting conditions and the expected tool life for each tool life group. Therefore it is expected that the structured knowledge contained in cutting data derived from the revised model should be similar to that of the original regression model. If such is the case, there is a strong case for an ANFIS tool life model, which has captured the structured knoledge about the cutting process, to be easily optimised to adapt in the new situation. That would be very beneficial, since that structured knowledge relevant to any shop floor machine tool should remain roughly unchanged. What is expected to vary significantly is the actual machining performance. Indeed, an expert machinist is able to make rough guesses about the expected outcome of a machining operation, using linguistic descriptions in terms such as short, medium or long tool life. These guesses are based on information describing the type of cut, material class, insert grade or cutting conditions. However, these guesses are not sufficient to base planning decisions upon them. A neurofuzzy model that has captured the relevant structured knowledge in the form of IF-THEN rules and can still be easily tuned to optimise its predictions, based on available cutting data would be a particularly attractive modelling tool.

A cutting data set similar to the checking data set employed in chapter 6 and earlier in this chapter has been created consisting of the same 4978 patterns. What is different now is the output of each pattern, i.e. the expected tool life. The original model would yield a mean absolute prediction error relative to the actual tool life of the revised model as high as 31.62% with a root mean squared error value of 13.48. The large error gives an indication of the size of the distortion introduced in the tool life model. It should be noted that this model mismatch is not caused by a simple linear scaling of the expected tool life, but the effect of the parameter *D* is non-linear with respect to the change in the expected tool life. The ANFIS model has been tested on the new data set and has shown a mean absolute prediction error of 33.41%, which corresponds to a root mean squared error value of

13.15. Obviously, this is unacceptable performance and it is desirable to re-train the ANFIS model based on available cutting data.

The new cutting data set has been randonly shuffled and split into two subsets, containing 2489 patterns. When one of these subsets was employed as training set the other one was used for cross-validation in a way similar to that described in section 7.3. The whole re-training required just 15 training epochs. An initial learning rate of 0.001 was selected, which was gradually reduced to 0.0003. The final root mean squared error value of the prediction error was reduced from 13.15 to 0.4652, corresponding to a mean absolute relative error of 2.85%. Thus the ANFIS model has been succesfully re-trained to capture the new input-output mapping with remarkable ease and sufficient accuracy. It is worth noting that the final RMSE value of the predicion error is much lower than the RMSE error in the case of the modelling based on the original regression model. The reasons for this result are:

- The training based on the initial model was based on a random pattern set. The presence of statistical outliers in that set inhibits the convergence of the learning algorithm to a solution that fits mainly the good data, instead of fitting the outliers as well. On the other hand, the training of the revised model was based on cutting data from a series of well constructed cutting tests, randomly reshuffled. Therefore, the risk of the learning procedure being slowed down by some outliers was lower in that case.
- The introduction of the parameter *D* in the revised model results in shorter tool life values. Thus, the RMSE is expected to be lower. However, the mean absolute relative error, which is now calculated against lower tool life values is not decreased accordingly.

Figures 7.16-7.17 demostrate some results for the updated ANFIS tool life modelling. An illustrative example for each major material class is given, i.e, both for mild and alloy, as well for stainless steels. In these figures the new model input-output mapping is contrasted with the original regression model to give an indication of the size of the model mismatch introduced.

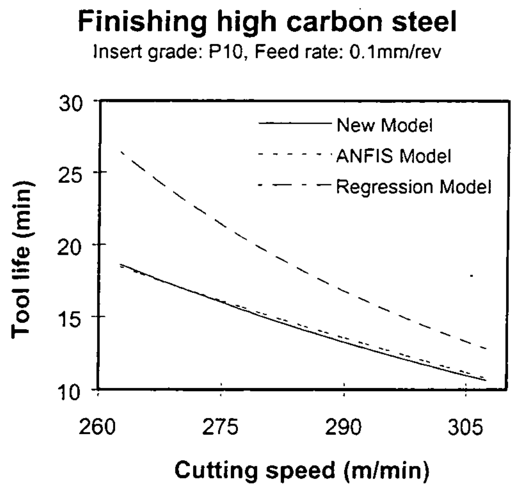

Figure 7.16a: RMSE=0.21, mean abs.err=1.47%

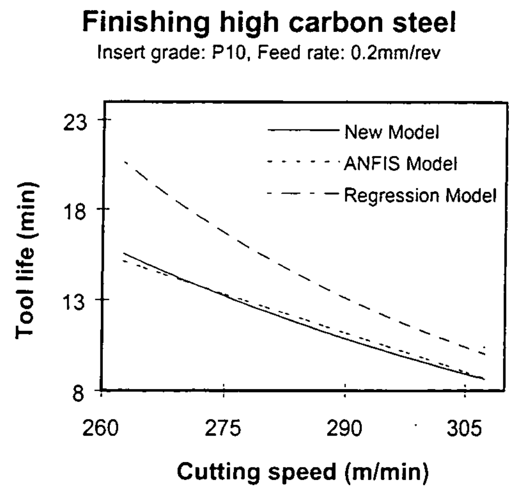

Figure 7.16b: RMSE=0.24, mean abs. err.= 1.90

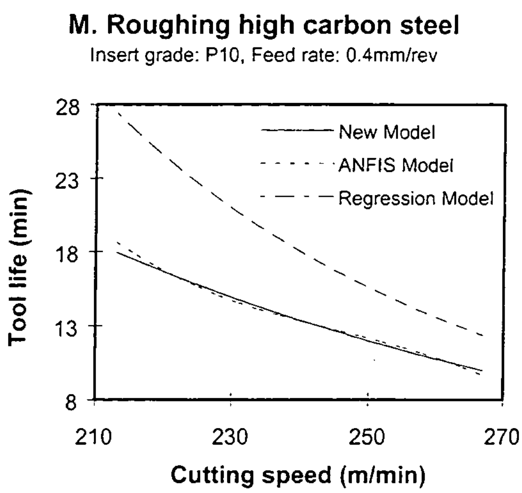

Figure 7.16c: RMSE=0.23, mean abs.err=1.29%

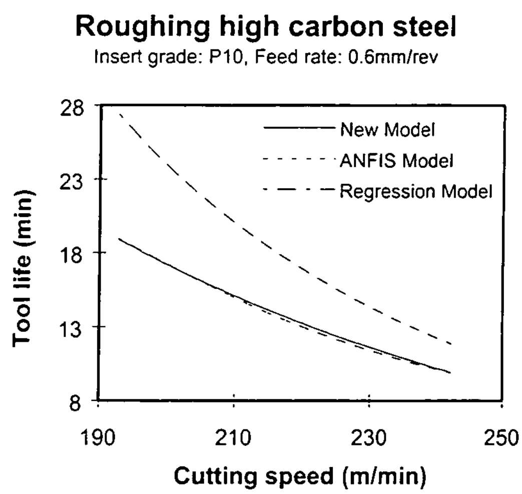

Figure 7.16d: RMSE=0.15, mean abs.err=1.03%

Figure 7.16: ANFIS tool life predictions for turning high carbon steel with an ISO P10 insert

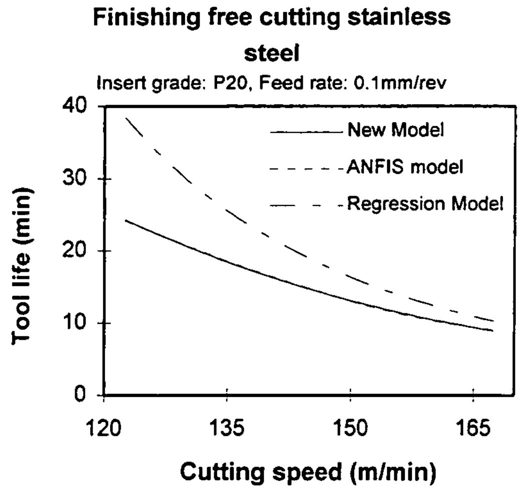

Figure 7.17a: RMSE=0.097, mean abs.err=0.62%

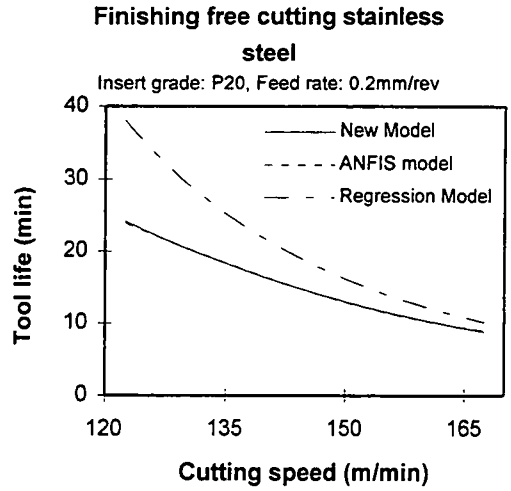

Figure 7.17b: RMSE=0.10, mean abs. err.= 0.58%

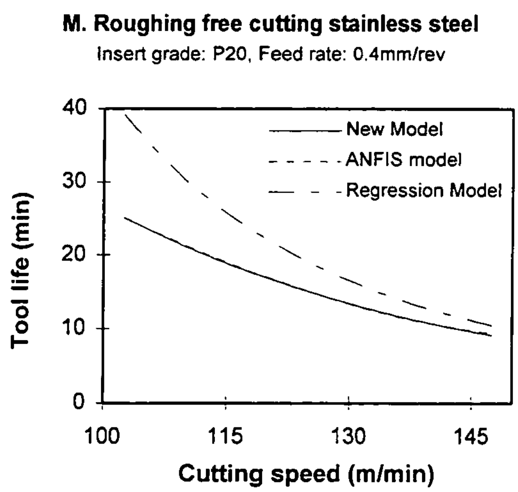

Figure 7.17c: RMSE=0.10, mean abs.err=0.59%

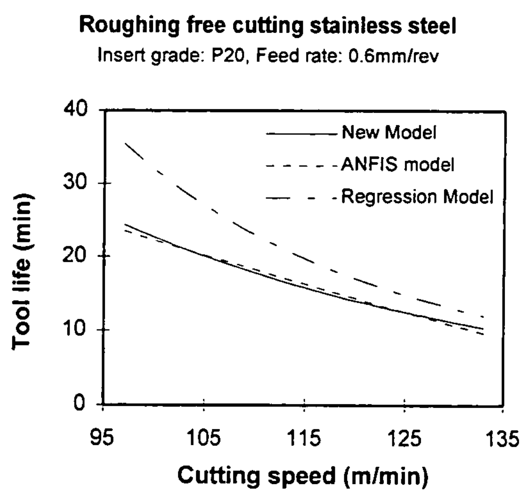

Figure 7.17d: RMSE=0.44, mean abs.err=2.45%

Figure 7.17: ANFIS tool life predictions for turning free cutting stainless steel with an ISO P20 insert

Figure 7.16 illustrates a case of machining high carbon steel with an ISO P10 insert. Examples are given for each type of cut. The root mean squared error is below 0.25 in all cases, while the mean absolute relative error is kept below 2%. Similarly, in figure 7.17 cases of cutting free cutting stainless steel are considered. Both the RMSE value, as well as the mean absolute relative error are extremely low in all cases apart from the roughing example, where they are still is below 0.5 and 2.5% respectively.

Thus, the ANFIS tool life model has shown remarkable flexibility in capturing input-output relationships significantly different from the one it was initially constructed for. This is an important result, since it indicates that it is appropriate to construct an initial neurofuzzy model based on a-priori knowledge about the cutting process and then re-train it to adjust on available cutting data. Any existing model, either based on cutting theory, or on multiple regression upon empirical tool life formulae or other equation can be employed for the creation of training data for obtaining an initial fairly accurate neurofuzzy model. The latter is in principle capable of performing as good a mapping as the initial theoretical or regression model. In addition it can extract structured knowledge about the cutting process in the form of IF-THEN rules. These rules can easily be interpreted by a human expert and conclusions may be drawn about the way in which the neurofuzzy model achieves a particular input-output mapping. Finally, the neurofuzzy model can be re-trained on the basis of available real life data.

All the above features compose a strong modelling framework, particularly useful for detailed process modelling. Thus, process planning and optimisation tasks, such as the optimisation of cutting conditions or the determination of optimal tool replacement policies can be greatly facilitated.

CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK

8.1 Discussion

A neurofuzzy tool life model for turning operations has been developed, which can facilitate machining optimisation. The computational platform on which the present research work was carried out was a 166MHz Pentium computer with 32MB of RAM, equipped with the Fuzzy Logic Toolbox (v 1.0) of the MATLAB (v. 4.2c1) software package for numerical computation and visualisation (The Mathworks Inc.). The adaptive, network-based, fuzzy inference system (ANFIS) architecture has been selected to perform the desired input-output mapping. The finally derived model predicts tool life for a wide range of inputs, i.e. combinations of workpiece material, type of cut, insert grade, feed rate and cutting speed.

A literature review has been carried out on topics such as tool wear and tool life, neural networks, fuzzy sets theory and neural fuzzy systems integration. Existing tool life models are usually either over simplistic or very complicated for machining optimisation practice. Deterministic empirical or theoretical tool life models fail to handle the uncertainty which is inherent to the tool wear process. The prevailing method for handling uncertainty in tool life prediction is a probabilistic one, either in the form of stochastic interpretation of the available tool life formulae or, less commonly, reliability-based modelling. However, tool wear variability can not be easily explained by simple probabilistic models, while the most flexible ones are too complex to be of practical use.

Fuzzy sets theory offers an attractive alternative as a means for handling uncertainty. This work has explored the potential of employing fuzzy inference methods for tool life prediction. It has been concluded that the developed model should be data-driven, in order to be applicable to machining optimisation practice. However, implementations of pure fuzzy systems rely heavily on the designer's own intuition and expertise, without a straightforward way of incorporating knowledge contained in numerical data, i.e. cutting process data. Neural networks are recognised as computational structures well suited to perform learning by examples. Yet, they suffer from lack of power in representing structured knowledge in a way that is relatively easily interpretable by humans. Therefore, the research focused on the development of a neurofuzzy model for tool life. Such a model combines both the merits of neural networks and fuzzy logic systems, i.e. the learning capability and the representation power respectively. The main characteristics of the developed model are:

- It is a typical case of a fuzzy inference system, since all the elements comprising such a system are present, i.e. fuzzifier, fuzzy rule base, inference engine and defuzzifier.
- It is completely data-driven, i.e. both the structure and the parameter identification are successfully carried out solely on the basis of available tool life data. This is achieved by employing a neural network-like training algorithm after an initialisation stage which yields a fairly accurate ANFIS model from the patterns set. Thus, structured knowledge about the cutting process performance is extracted from numerical data.
- The obtained neurofuzzy tool life model is of generic nature. It exhibits adequate predictive performance for a wide range of combinations of workpiece material class, type of cut, insert grade, feed rate and cutting speed.
- The complexity of the model is rather low, considering that the overall mapping is achieved by just a 52 fuzzy rules model.
- Available a priori knowledge about the cutting process is employed for the creation of the training exemplars, by accommodating results of tool life modelling research recently carried out at Durham University [Alamin 1996].

- The model is non-parametric in the statistical sense, i.e. it does not rely on any statistical assumption about the underlying probability distribution function of tool wear.
- It is a flexible model, as it is easily adjustable on the basis of new tool life data. Evidently, if the initial tool life data are of poor quality there will be more need for adaptation with real life data.
- Even though the training examples are derived by utilising existing tool life modelling, the universality of ANFIS models as function approximators ensures that any functional deviation from the initial model may be captured by employing further additional training or by adding extra fuzzy rules. Therefore the developed model does not rely on any empirical formula or analytical tool life equation.
- The meaningful way in which tool life prediction is achieved facilitates the interpretation of the results. Therefore, it is easy to add extra fuzzy rules at a region where it is desired to improve the mapping performance or where no information was included in the initial training cutting data set.

During the development of the model, some important conclusions were drawn at both the initialisation and the parameter identification phase. In the structure identification phase the data availability problem was discussed and tool life training examples were extrapolated from an existing tool life model. Input space partitioning was a major concern during that stage. It was found that the representation power of the neurofuzzy inference system was better exploited if ordinal categorical variables such as material class, type of cut and insert grade are "softly" defined. Therefore, soft boundaries for the input space partitioning were defined, which allow for a mapping from a specific input pattern to a tool life value to be achieved by receiving computational support from inputs combinations that have a degree of similarity to the pattern under consideration. The definition of these soft boundaries between classes of inputs produced improved mapping performance with a reduced set of fuzzy rules.

The learning phase required a low number of epochs. However, under the computational platform employed the whole learning procedure was very time consuming. Nonetheless, once the learning has been completed, the response time required for the recall phase is negligible. Significantly faster training times could be achieved by improving the computational or hardware platform employed. It is anticipated that "coding" the overall learning process into a programming language, such as C++ will drastically reduce the CPU time required. Time performance can also be improved by software or hardware upgrades. Ultimately, fuzzy logic systems can be implemented on to hardware, by taking advantage of the growing number of dedicated fuzzy hardware and fuzzy logic processors that are now becoming commercially available.

A second shortcoming observed was the relatively poor predictive performance of the model at inputs combinations close to the defined boundary values. The reasons for this is related to the source of the cutting data set used for training. The training data has been randomly obtained by a series of locally-valid models derived by multiple regression [Alamin 1996]. Random pattern generation can ensure rich training data, but can also create statistical outliers. In addition, the mapping produced by the training data is non smooth in the boundaries between tool life groups. This poses problems to the hybrid algorithm employed for the optimisation of the premise parameters of the model. In particular, the algorithm estimates the gradient of the error hypersurface at each network node with respect to each one antecedent parameter. Because of the non-smoothness of the function defined by the training data, the estimate of the gradient attains large values in the boundaries between different classes, i.e. tool life groups. Thus, parameter updating is misguided, as it tries to optimise parameters based on training samples which are adjacent in the input space but produce very different gradients, as they belong to different classes. The learning algorithm tries to reduce the large squared errors appearing in few patterns lying close at the boundaries between tool life groups and gets trapped in a local minimum.

It was found appropriate to exclude a very small number of patterns outliers from the training data set These outliers were found to dominate the learning procedure by introducing very large error values into the error-based training algorithm. Releasing the training process from the burden of achieving accurate mapping for those patterns has allowed the neurofuzzy model predictive performance to improve. Alternatively these outliers may be allowed to influence the training process, thus contributing to the boundary spaces mapping, by introducing a robust error suppressor cost function, instead of the simple quadratic criterion employed. However, this would call for a different learning algorithm than the simple least squares estimation for the consequent parts linear parameters. The predictive performance of the finally derived model was checked over a series of well constructed tests, spanning a wide range of inputs combinations, i.e. cutting speed and feed rate variations for all defined combinations of workpiece material class, type of cut and insert grade. The finally achieved root mean squared error was 1.19 and 1.35 for the training and checking set respectively.

It is important to note that cutting data derived from the shop floor or laboratory experiments are less likely to define non smooth mappings at the boundaries between tool life groups. Therefore, training based on such data is less likely to be trapped on local minima, as a smoother error hypersurface should be expected. However, sufficiently rich information can hardly be obtained from such data, as this would be both costly and time consuming. This prohibits model-building solely based on real-life data. Yet, such data can easily be employed for tuning an already existing neurofuzzy tool life model. Thus, this thesis advocates the use of simulated cutting data for deriving an initial neurofuzzy tool life model, which is flexible enough to be easily refined at a later stage, based on approved, shop floor data. The flexibility of the developed neurofuzzy tool life model has been demonstrated by re-training the model based on a new set of cutting data. This set has been created by introducing significant model mismatch into the series of local models employed for obtaining the initial cutting data. The ANFIS model was shown to capture with remarkable ease and sufficient accuracy the new mapping.

8.2 Future directions

Finally, this thesis concludes with some remarks on the applicability of the developed neurofuzzy tool life model. Tool life prediction is designated to facilitate machining optimisation activities, such as optimal determination of cutting conditions and tool replacement strategy decision making. The training examples provided to the ANFIS network correspond to mean values of expected tool life. Therefore the neurofuzzy model has learned to perform a mapping from a set of inputs to the expected tool life value. However, the mean value of expected tool life does not suffice to describe the uncertainty about tool life. From a probabilistic point of view, it would be beneficial to estimate confidence intervals for these values. This work has adopted a non-probabilistic approach to uncertainty handling, i.e. that of fuzzy sets theory. What would be relevant in a fuzzy systems approach is uncertainty representation via fuzzy sets.

The output of the ANFIS model developed is essentially a crisp value, which results from the weighted average defuzzification of the results of each one fuzzy rules firing. The output of the fuzzy rules are in fact fuzzy singletons. In order to take full advantage of the representation and reasoning power of fuzzy systems, the fuzzy tool life model should be put into the more general framework of fuzzy optimisation of machining processes. Such a fuzzy optimisation problem should receive as input the non-defuzzified output of the fuzzy tool life model. This value is in essence a fuzzy set and being such contains information about its uncertainty. For instance, a tool life output fuzzy set with wide support corresponds to a highly uncertain outcome, while an output tool life fuzzy set with narrow support carries increased certainty about the tool life value. A fuzzy model that would better exploit the strengths of fuzzy sets theory would be one which considers non-singleton fuzzification and defuzzification, which are in principle more appropriate for information processing of highly uncertain or corrupted by noise data. Indeed non-singleton fuzzy systems ensure smoother transition between decision boundaries and are more robust in dealing with ill-defined parameters.

There have been very few attempts so far reported in the literature which treat machining optimisation problems by a fuzzy sets theory approach. These are related cutting parameters selection [Balazinski and Bellerose 1994] tool selection and cutting conditions determination [Chen et al. 1995] and machining process selection [Huang et al. 1996]. However, in all these examples the determination of the premise and consequent parts parameters is rather arbitrary and is not optimised on the basis of available machining process data. Therefore, the present work should be considered only as a first step towards a systematic treatment of machining process optimisation problems from the viewpoint of fuzzy sets theory. Other steps to be taken should consider extending the current model for a wider range of operations, as well as obtaining neurofuzzy models for other critical output parameters in machining operations, such as surface roughness, dimensional and tolerance accuracy, machining costs etc. Such models could eventually be incorporated into a flexible formulation for the tool selection and cutting conditions optimisation problems, as well as for a new approach to fuzzy logic based tool replacement policy decision making.

REFERENCES

Abe S. and Lan M.-S., 1995, "Fuzzy rules extraction directly from numerical data for function approximation", IEEE Trans. Sys., Man, Cybern., 25(1), 119-129.

Agapiou J.S., 1992, "Optimization of multistage machining systems, Part 1: Mathematical solution", Trans. ASME, J. Eng. Ind., 114, 524-531.

Alamin B., 1996, "Tool life prediction and management for an integrated tool selection system", Ph.D. Thesis, University of Durham, UK.

Altintas Y., 1992, "Prediction of cutting forces and tool breakage in milling from feed drive current measurements", Trans. ASME, J. Eng. Ind., 114, 386-392.

Altintas Y. Newell N. and Ito M., 1996, "Modular CNC design for intelligent machining, Part 1: Design of a hierarchical motion control module for CNC machine tools", Trans. ASME, J. Manuf. Science Eng., 118, 506-513.

Altintas Y. and Munasingue W.K., 1996, "Modular CNC design for intelligent machining, Part 2: Modular integration of sensor based milling process monitoring and control tasks", Trans. ASME, J. Manuf. Science Eng., 118, 514-521.

Arabshahi P., Choi J.J., Marks II R.J. and Caudell T.P., 1996, "Fuzzy parameter adaptation in optimization: some neural net training examples", IEEE Comp. Sci. Engng., 3(1), 57-65.

Arsecularatne J.A., Mathew P. and Oxley P.L.B., 1995, "Prediction of chip flow direction and cutting forces in oblique machining with nose radius tools", Proc. Inst. Mech. Eng., Part B, 209, 305-315.

Arsecularatne J.A., Fowle R.F., Mathew P. and Oxley P.L.B., 1996, "Prediction of tool life in oblique machining with nose radius tools", Wear, 198, 220-228.

Arsovski S.M., 1983, "Wear sensors in the adaptive control of machine tools", Int. J. Prod. Res., 21(3), 347-356.

Balazinski M and Bellerose M, 1994, "Application of fuzzy logic techniques to the selection of cutting parameters in machining processes", Fuzzy Sets and Systems, 63, 307-317.

Bezdek J.C., 1994, "What is computational intelligence", in "Computational intelligence, imitating life", ed. by Zurada J.M., Marks II R.J. and Robinson C.J., IEEE Press, 1-12.

Billatos S.B., Bayoumi A.E., Kendall L.A. and Saunders S.C., 1986, "A statistical model for certain tool materials with applications to machining", Wear, 112, 257-271.

Billatos, S.B. and Kendall L.A., 1990, "A replacement model for multi-tool transfer lines", Trans. ASME, J. Eng. Ind., 112, 253-259.

Blum, T. and Inasaki, I., 1990, "Study on acoustic emission from the orthogonal cutting process", Trans. ASME, J. Eng. Ind., 112(3), 203-211.

Brown M. and Harris C.J., 1995, "A perpective and critique of adaptive neurofuzzy systems used for modelling and control applications", Int. J. Neural Sys., 6(2), 197-220.

Buckley J.J. and Hayashi Y., 1994, "Fuzzy neural networks", in "Fuzzy sets, neural networks and soft computing", ed. by Yager R.R. and Zadeh L.A., Van Nostrand Reinhold.

Byrne G., Dornfeld D., Inasaki I., Ketteler G., Konig W. and Teti R., 1995, "Tool condition monitoring (TCM) - The status of research and industrial application", Annals of the CIRP, 44(2) 541-567.

Carlsson T.E. and Strand F., 1992, "A statistical model for prediction of tool life as a basis for economical optimization of the cutting process", Annals of the CIRP, 41(1), 79-82.

Carolan T.A., Hand D.P., Barton J.S., Jones J.D.C, Wilkinson P. and Reuben R.L., 1996, "Assessment of tool wear in milling using acoustic emission detected by a fiber-optic interferometer", Trans. ASME, J. Manuf. Sci. Eng., 118, 428-433.

Carpenter G.A. and Grossberg S., 1994, "Fuzzy ARTMAP: a synthesis of neural networks anf fuzzy logic for supervised categorization and nonstationary prediction", in "Fuzzy sets, neural networks and soft computing", ed. by Yager R.R. and Zadeh L.A., Van Nostrand Reinhold.

Carpenter I.D., 1996, "Machinability assessment and tool selection for milling", Ph.D. Thesis, University of Durham, UK.

Chen J.C., 1996, "A fuzzy-nets tool-breakage detection system for end-milling operations", Int. J. Adv. Manuf. Technol., 12(3), 153-164.

Chen T. and Chen R., 1995, "Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems", IEEE Trans. Neural Networks, 6(4), 911-917.

Chen Y., Hui A. and Du R, 1995, "A fuzzy expert system for the design of machining operations", Int. J. Mach. Tools Manufact., 35(12), 1605-1621.

Chen Y., Li X., Orady E., 1996, "Integrated diagnosis using information-gain-weighted radial basis function neural networks", Comp. Ind. Eng., 30(2), 243-255.

Chiu S., 1994, "Fuzzy model identification based on cluster estimation", J. Intelligent & Fuzzy Sys., 2(3), 267-278.

Chiu S., 1996, "Method and software for extracting fuzzy classification rules by subtractive clustering", Biennial Conf. of the North American Fuzzy Information Processing Society - NAFIPS 1996, 461-465.

Chryssolouris G., 1992, "Manufacturing systems: theory and practice", Springer-Verlang New York Inc.

Chryssolouris G., Domroese M., Beaulieu P., 1992, "Sensor synthesis for control of manufacturing processes", 114, 158-174.

Cordon O. and Herrera G., 1995, "A general study on genetic fuzzy systems", in "Genetic algorithms in engineering and computer science", pp.35-59, J. Wiley.

Dagli C.H. (ed.), 1994, "Artificial neural networks for intelligent manufacturing", Chapman & Hall.

Dasarathy P.V., 1997, "Sensor fusion potential exploitation - Innovative architectures and illustrative applications", Proceedings of the IEEE, 85(1), 24-38.

Dickerson J.A. and Kosko B., 1996, "Fuzzy function approximation with ellipsoidal rules", IEEE Trans. Sys., Man, Cybern., 26(4), 542-560.

El Wardany T.I. and Elbestawi M.A., 1997, "Prediction of tool failure rate in turning hardened steels", Int. J. Adv. Manuf. Techn., 13, 1-16.

Emmanouilides C. and Petrou L., 1997, "Identification and control of anaerobic digesters using adaptive, on-line trained neural networks", Computers Chem. Engng., 21(1), 113-143.

Ermer D.S., 1970, "A Baysian model of machining economics for optimization by adaptive control", Trans. ASME, J. Engng. Ind., 95, 628-632.

Ezugwu E.O., Arthur, E.L. and Hines E.L., 1995, "Tool wear prediction using artificial neural networks", J. Materials Proc. Technology, 49, 255-264.

Fang X.D., 1994, "Experimental investigation of overall machining performance with progressive tool wear at different tool faces", Wear, 173, 171-178.

Fang X.D. and Jawahir I.S., 1994, "Predicting total machining performance in finish turning using integrated fuzzy-set models of machinability parameters", Int. J. Prod. Res., 32(4), 833-849.

Gardner W.A., 1986, "Introduction to random processes with applications to signals and systems", Macmillan Publ. Co.

Girosi F., Jones M. and Poggio T., 1995, "Regularization theory and neural network architectures", Neural Computation, 7, 219-269.

Udo G. J. and Gupta Y.P., 1994, "Applications of neural networks in manufacturing systems", Prod. Plan. & Control, 5(3), 258-270.

Grabec I., 1986, "Chaotic dynamics of the cutting process", Int. J. Mach. Tools Manuf., 28, 19-32.

Grapec I. and Kuljianic E., 1994, "Charecterization of manufacturing processes based upon acoustic emission analysis be neural networks", CIRP Annals, 43(1), 77-80.

Gupta M.M. and Rao D.H., 1994, "Neuro-control systems: a tutorial", in "Neuro-control systems, theory and applications", IEEE Press.

Hassoun M.H., 1995, "Fundamentals of artificial neural networks", MIT Press.

Haykin S., 1994, "Neural Networks, a comprehensive foundation", McMillan College Publishing Company, Inc.

Hertz J., Krogh A. and Palmer R.G., 1991, "Introduction to the theory of neural computation", Addison-Wesley Publishing Company.

Hirota K. and Sugeno M (editors), 1995, "Industrial applications of fuzzy logic technology in the world", World Scientific Publishing.

Hope A.D., Javed M.A., Littlefair G. and Rao B.K.N., 1996, "Intelligent multi-sensor tool wear monitoring system for unmanned machining environments", 5th Int. Conf. on profitable condition monitoring, Harrogate UK, ed. Rao, B.K.N., BHR Group Ltd., 287-297.

Huang S.H. and Zhang H.-C., 1994, "Artificial neural networks in manufacturing: concepts, applications, and perspectives", IEEE Trans. Compon., Packag. And Manuf. Techn., 17(2), 212-228.

Huang S.H. and Zhang H.-C., 1995, "Neural-expert hybrid approach for intelligent manufacturing: a survey", Computers in Industry, 26, 107-126.

Huang S.H., Zhang H-C, Sun S. and Li H.H., 1996, "Function approximation and neural-fuzzy approach to machining process selection", IEEE Trans. Compon. Packag. and Manuf. Techn., Part C, 19(1), 9-18.

Hunt K.J., Haas R. and Murray-Smith R., 1996, "Extending the functional equivalence of radial basis function networks and fuzzy inference systems", IEEE Trans. Neural Networks, 7(3), 776-781.

Iakovou E., Ip C.M. and Koulamas C., 1996, "Adaptive tool replacemenr policies in machining economics", Trans. ASME, J. Manuf. Science. Eng., 118, 658-663.

Iwaka K. and Moriwaki T., 1976, "An application of acoustic emission measurement to in-process sensing of tool wear", CIRP Annals, 25(1), 21-26.

Jain A.K., Mao J. and Mohiuddin K.M, 1996, "Artificial neural networks: a tutorial", IEEE Computer, 29(3), 31-44.

Jang, J.-S.R., 1993, "ANFIS: Adaptive-Network-Based Fuzzy Inference System", IEEE Trans. Sys., Man and Cybern., 23(3), 665-685.

Jang, J.-S.R., 1995, "Neuro-fuzzy modelling and control", Proc. IEEE, 83(3), 378-405.

Jang J.-S.R., 1994, "Structure determination in fuzzy modelling: a fuzzy CART approach", Proc. Of the 3rd IEEE Conf. on Fuzzy Systems, World Congress on Computational Intelligence, pp. 480-485.

Jang, J.-S.R. and Gulley, N, 1995, "Fuzzy Logic Toolbox for Use with MATLAB", The Mathworks Inc.

Jang J.-S.R. and Mijutani E., 1996, "Levenberg-Marquardt method for ANFIS learning", Biennial Conf. of the North American Fuzzy Information Processing Society - NAFIPS 1996, 87-91.

Jang J.-S.R., Sun, C.-T., 1993, "Functional equivalence between radial basis function networks and fuzzy inference systems", IEEE Trans. Neural Networks, 4(1), 156-159.

Jang, J-S.R., Sun, C-T and Mitzutani, E., 1997, "Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence", Prentice Hall Int.

Jansson, D.G. Rourke, J.M. and Bell, A.C., 1984, "High speed roughness measurement", Trans. ASME, J. Eng. Ind, 106(1), 34-39.

Jeon J.U. and Kim S.W., 1988, "Optimal flank wear monitoring of cutting tools by image processing", Wear, 127, 207-217.

Jin L., Gupta M.M. and Nikiforuk P.N., 1996, "Approximation capabilibties of feedforward and recurrent neural networks", in "Intelligent control systems: theory and applications", 234-264, IEEE Press.

Khraisheh M.K., Pezeshki C., Bayoumi A.E., 1995, "Time series based analysis for primary chatter in metal cutting", J. Sound Vibr., 180(1), 67-87.

Kim H.M. and Mendel J.M., 1995, "Fuzzy basis functions: comparisons with other basis functions", IEEE Trans. on Fuzzy Systems, 3(2), 158-167.

Klir G.J. and Yuan B., 1995, "Fuzzy sets and fuzzy logic: theory and applications", Prentice-Hall International, Inc.

Ko T.J. and Cho D.W., 1992, "Fuzzy pattern recognition for tool wear monitoring in diamond turning", CIRP Annals, 41(1), 125-128.

Ko T.J., Cho D.W. and Jung M.Y., 1995, "On-line monitoring of tool breakage in face milling using a self-organised neural networks", J. Manuf. Sys., 14(2), 80-90.

Kosko B., 1992, "Neural networks and fuzzy systems, a dynamical approach to machine intelligence", Prentice-Hall International, Inc.

Kosko B., 1997, "Fuzzy engineering ", Prentice-Hall International, Inc.

Kosko B. and Dickerson, 1995, "Function approximation with additive fuzzy systems", in "Theoretical aspects of fuzzy control", ed. by Nguyen H.T., Sugeno M., Tong R.M and Yager R.R. (IEEE 2nd Int. Conf. on Fuzzy Systems, 1993 San Francisco, Calif.), 313-347, John Wiley & Sons.

Koulamas C., 1991, "Simultaneous determination of optimal machining conditions and tool replacement policies in constrained machining economics problems by geometric programming", Int. J. Prod. Res. 29(12), 2407-2421.

Lee K.-M., Kwak D-H. and Kwang H.L., 1996, "Fuzzy inference neural network for fuzzy model training", IEEE Trans. Sys., Man, Cybern., 26(4), 637-645.

Leopold J., 1996, "Process monitoring of 3D-cutting inserts using optical methods", J. Materials Proc. Technology, 61, 34-38.

Levi R. and Rossetto S., 1978, "Machining economics and tool life variation, Part 1: Basic considerations and their practical implications", Trans. ASME, J. Eng. Ind., 100, 393-396.

Levy E.K., Tsai C.L., Groover M.P., 1976, "Analytical investigation of the effect of tool wear on the temperature variations in a metal cutting tool", Trans. ASME, J. Eng. Ind., 251-257.

Lin C.-J., 1996, "A fuzzy adaptive learning control network with on-line strauctural and parameter learning", Int. J. Neural Sys., 7(5), 569-590.

Lin C.-T. and Lee C.S.G, 1994, "Reinforcement structure/parameter learning for neural network-based fuzzy logic control systems", IEEE Trans. on Fuzzy Systems, 2(1), 46-63.

Lin C.-T. and Lee C.S.G, 1996, "Neural fuzzy systems: a neurofuzzy synergism to intelligent systems", Prentice-Hall Inc.

Lin C.-T., Lin C.-J. and Lee C.S.G., 1995, "Fuzzy adaptive learning control with on-line neural training", Fuzzy sets and systems, 71, 25-45.

Lin C.-T. and Lu Y.-C., 1995, "A neural fuzzy system with fuzzy supervised learning", IEEE Trans. Fuzzy Systems, 30(2), 169-189.

Lin C.-T. and Lu Y.-C., 1996, "A neural fuzzy system with fuzzy supervised learning", IEEE Trans. Sys,. Man. Cybern., 26(5), 744-763.

Lin J.S. and Weng C.I., 1991, "Nonlinear dynamics of the cutting process", Int. J. Mech. Sci., 33(8), 645-657.

Lindley D.V., 1987, "The probability approach to the treatment of uncertainty in artificial intelligence and expert systems", Statistical Science, 2(1), 17-24.

Liu H. and Makis V., 1996, "Cutting-tool reliability assessment in variable machining conditions", IEEE Trans. Reliability, 45(4), 573-581.

Liu J.J. and Dornfeld D.A., 1996, "Modelling and analysis of acoustic emission in diamond turning", Trans. ASME, J. Manuf. Science Eng., 118, 199-207.

Lundholm T., Bergstrom E., Enarson D., Harder L., Lindstrom B., Nicolescu M. and Nilsson B., 1992, "New techniques applied to adaptive controlled machining", Robotics & Comp-Int Manuf., 9(4/5), 383-389.

Malakooti B.B., Zhou Y.Q. and Tandler E.C, 1995, "In-process regressions and adaptive multi-criteria neural networks for monitoring and supervising machining operations", J. Intelligent Manuf., 6, 53-66.

Mari D. and Gonseth D.R., 1993, "A new look at carbide tool life", Wear, 165, 9-17.

Marks II R.J. (editor), 1994, "Fuzzy logic technology and applications", IEEE Press.

Maropoulos P.G. and Hinduja S., 1990, "Automatic tool selection for finish turning", Proc Instn Mech Engrs, 204(B), 43-51.

Maropoulos P.G. and Hinduja S., 1991, "Automatic tool selection for rough turning", Int. J. Prod. Res., 29(6), 1185-1204.

Maropoulos P.G., 1992, "Cutting tool selection: an intelligent methodology and its interfaces with technical and planning functions", Proc Instn Mech Engrs, 206(B), 49-59

Maropoulos P.G., 1995, "Review of research in tooling technology, process modelling and process planning. Part I: Tooling and process modelling", Comp. Integrated Manuf. Sys., 8(1), 5-12.

Maropoulos P.G. and Alamin B., 1995, "Intelligent tool selection for machining cylindrical components, part 2: results from the testing of the knowledge-based module", Proc. Instn. Mech. Engrs., 209(B), 183-192.

Maropoulos P.G., Gill P.A.T., 1995, "Intelligent tool selection for machining cylindrical components. Part 1: logic of the knowledge-based module", Proc Instn Mech Engrs, 209(B), 173-181.

Maropoulos P.G. and Alamin B., 1996, "Integrated tool life prediction and management for an intelligent tool selection system", J. Materials Proc. Technology, 61, 225-230.

Matsumura T., Obikawa T., Shirakashi T. and Usui E., 1993, "Autonomous turning operation planning with adaptive prediction of tool wear and surface roughness", J. Manuf. Sys., 12(3), 253-262.

Mehrotra K., Mohan C.K. and Ranka S., 1997, "Elements of artificial neural networks", MIT Press.

Mendel J.M., 1995, "Fuzzy logic systems for engineering: a tutorial", Proc. IEEE, 83(3), 345-377.

Metcut Research Associates, 1980, "Machining Data Handbook", Machinability Data Center.

Mitra S. and Pal S.K., 1995, "Fuzzy multi-layer perceptron, inferencing and rule generation", IEEE Trans. Neural Networks, 6(1), 51-63.

Monostori L. and Barschdorf D., 1992, "Artificial neural networks in intelligent manufacturing", Rob. & Comp. Integr. Manuf. 9(6), 421-437.

Monostori L. Markus A., Van Brussel H.V. and Westkampfer E, 1996, "Machine learning approaches to manufacturing", Annals of the CIRP, 45(2), 675-712.

Mouzouris G.C. and Mendel J.M, 1997, "Nonsingleton fuzzy logic systems: theory and application", IEEE Trans. on Fuzzy Systems, 5(1), 56-71.

Nadler M. and Smith E.P., 1993, "Pattern recognition engineering", J. Wiley & Sons.

Nagasaka K. and Hashimoto F., 1982, "The establishment of a tool life equation considering the amount of tool wear", Wear, 81, 21-31.

Narayanan V., 1995, "Systems for the prediction of process parameters", J. Mat. Proc. Techn., 54, 64-69.

Nie J., 1995, "Constructing fuzzy model by self-organizing counterpropagation network", IEEE Trans. Sys., Man, Cybern., 25(6), 963-970.

Nie J., Lee T.H. and Linkens D.A., 1997, "A note on the integration of fuzzy systems with neural networks under a TLTT framework", Fuzzy Sets and Systems, 87, 277-289.

Pal S.K. and Mitra S., 1992, "Multilayer perceptron, fuzzy sets and classification", IEEE Trans. Neural Networks, 3(5), 683-697.

Pandit S.M. and Kashou S., 1982, "A data dependent system strategy for on-line tool wear sensing", Trans. ASME, J. Eng. Ind., 104(3), 217-223.

Park J.-J. and Ulsoy A.G., 1992, "On-line tool wear estimation using force measurement and a non-linear observer", Trans. ASME, J. Dyn Sys. Meas. Control, 114, 666-672.

Park J.-J. and Ulsoy A.G., 1993, "On-line flank wear estimation using and adaptive observer and computer vision, Part 1: Theory", Trans. ASME, J. Eng. Ind., 115, 30-36.

Qin S.-Z., Su H-T and McAvoy T.J, 1992, "Comparison of four neural net learning methods for dynamic system identification", IEEE Trans. Neural Networks, 3(1), 122-130.

Ramalingam S., 1977, "Tool-life distributions, Part 2: Multiple-injury tool-life model", Trans. ASME, J. Eng. Ind., 99, 523-528.

Ramalingam S. and Watson J.D., 1977, "Tool-life distributions, Part 1: Single-injury tool-life model", Trans. ASME, J. Eng. Ind., 99, 519-522.

Ramalingam S., 1982, “Tool wear, tool life distribution and consequences”, in “On the art of cutting metals - 75 years later”; ed. By Kops L. and Ramalingam S., The Winter annual meeting of the ASME, Phoenix, Arizona, November 14-19, 1982, ASME, PED (Publication).

Rocke D.M. and Woodruff D.L., 1996, "Identification of outliers in multivariate data", J. of the American, Statistical Association, 91(435), 1047-1061.

Rossetto S. and Levi R., “Machining economics and tool life variation, Part 2: Application to models for machining processes”, Trans. ASME, 100, 397-402.

Ryabov O., Mori K. and Kasashima N., 1996, “An In-process direct monitoring method for milling tool failures using a laser sensor”, CIRP Annals, 45(1), 97-100.

Saade J.J., 1996, "A unifying approach to defuzzification and comparison of the outputs of fuzzy controllers", IEEE Trans. on Fuzzy Systems, 4(3), 227-237.

Sadat A.B., 1994, “Tool wear measurement and monitoring techniques for automated machining cells”, Procs of the Energy Sources Technology Conference, N. Orleans, USA, 61, 103-115.

Sadat A.B. and Raman S., 1987, “Detection of tool flank wear using acoustic signature analysis”, Wear, 115, 265-272.

Seco Tools AB, 1993, “Seco turning tools catalogue”, Seco Tools AB, Sweden.

Shaw M. C, 1984, "Metal cutting principles", Oxford University Press.

Sheikh A.K., Kendall L.A. and Pandit S.M., 1980, “Probabilistic optimisation of multitool machining operations”, Trans. ASME, J. Eng. Ind., 102, 239-246.

Shin Y.C. Abodelmonem A.H. and Kumara S. (ed.), 1992, "Neural networks in manufacturing and robotics", presented at "The winter annual meeting of the American Society of Mechanical Engineers, Anaheim, California, November 8-13, 1992", ASME PED-Vol.57.

Sjoberg J., Zhang Q., Ljung L., Benveniste A., Delyon B., Glorennec P.-Y. , Hjalmarsson H. and Juditsky A., 1995, "Nonlinear black-box modeling in system identification: a unified overview", Automatica, 31(12), 1691-1724.

Sun C.-T., 1994, "Rule-base structure identification in an adaptive-network-based fuzzy inference system", IEEE Trans. Fuzzy Systems, 2(1), 64-73.

Takagi T. and Sugeno M., 1985, "Fuzzy identification of systems and its applications to modeling and control", IEEE Trans. Sys., Man and Cybern., 15(1), 116-132.

Tan F.P. and Creese R.C., 1995, "A generalised multi-pass machining model for machining parameter selection in turning", Int. J. Prod. Res., 1995, 33(5), 1467-1487.

Taylor F.W., 1906, "On the art of cutting metals", Trans. ASME, 28, 31-350.

Trent E.M., 1991, "Metal Cutting", Butterworth-Heinmann Ltd.

Udo G. J. and Gupta Y.P., 1994, "Applications of neural networks in manufacturing systems", Prod. Plan. & Control, 5(3), 258-270.

Uehara K. , 1973, "New attempts for short time tool life testing", Annals of the CIRP, 22(1) 23-24.

Ulsoy A.G. and Koren Y., 1989, "Applications of adaptive control to machine tool process control", IEEE Control Sys. Mag., June, 33-37.

Ulsoy A.G and Koren Y., 1993, "Control of machining processes", Trans. ASME, J. Dyn. Sys. Meas. Control, 115, 301-307.

Ulsoy A.G., Koren Y. and Rasmussen F., "Principal developments in the adaptive control of machine tools", Trans. ASME, J. Dyn. Sys. Meas. Control, 108, 107-112.

Usui E., Shirakashi T. and Kitagawa T., 1984, "Analytical prediction of cutting tool wear", Wear, 100, 129-151.

Wang D.X., Zuo M.J., Qi K.Z. and Liang M., 1996, "On-line tool adjustment with adaptive tool wear function identification", Int. J. Prod. Res., 34(9) 2499-2515.

Wang L.-X., 1994, "Adaptive fuzzy systems and control, design and stability analysis", Prentice-Hall Inc.

Wang L.-X., 1995, "Design and analysis of fuzzy identifiers of nonliear dynamic systems", IEEE Trans. Autom. Control, 40(1), 11-23.

Weller E.J., Schorier H.M. and Weichbrodt B., 1985, "What sound can be expected from a worn tool?", Trans. ASME, J. Eng. Ind., 91, 525-534.

Wilcox S.J., Reuben R.L. and Souquet P., 1997, "The use of cutting force and acoustic emission signals for the monitoring of tool insert geometry during rough face milling", Int. J. Mach. Tools Manufact., 37(4), 481-494.

Yager R.R. and Filev D.P., 1994a, "On a flexible structure of fuzzy systems models", in "Fuzzy sets, neural networks and soft computing", ed. by Yager R.R and Zadeh L.A., 1-28, Van Nostrand Reinhold.

Yager R.R. and Filev D.P., 1994b, "Approximate clustering via the mountain method", IEEE Trans. Sys., Man, Cybern., 24, 1279-1284.

Yen J., Langari R., Zadeh L.A. (editors), 1995, "Industrial applications of fuzzy logic and intelligent systems", IEEE Press.

Yeo S.H., 1989, "Towards enhancement of machinability data by multiple regression", J. Mech. Work. Technology, 19, 85-99.

Zadeh L.A., 1965, "Fuzzy Sets", Information and Control, 8, 338-353.

Zadeh L.A., 1973, "Outline of a new approach to the analysis of complex systems and decision processes", IEEE Trans. Sys., Man Cybern., 3(1), 28-44.

Zadeh L.A., 1994, "Fuzzy logic, neural networks and soft computing", Communications of the ACM, 37(3), 77-84.

Zhang X., Hang C.-C., Tan S and Wang P.-Z., 1996, "The min-max function differentiation and training of fuzzy neural networks", IEEE Trans. Neural Networks, 7(5), 1139-1150.

Zhou C., Chandra J. and Wysk, R., 1990, "Optimal cutting tool replacement based on tool wear status", Int. J. Prod. Res., 28(7), 1357-1367.

APPENDIX A: TOOL WEAR MEASUREMENT

A.1 Off-line tool wear measurement

Off-line measurements take place after the actual cutting processes has been terminated and can be direct or indirect. A number of different approaches can be adopted for off-line determination of tool wear. These include optical methods, contact gauges, electrical methods, radioactive techniques and loss of material measurements [Sadat A.B. 1994]. A brief description of these approaches follows.

A.1.1 Optical Methods

The simplest and most easily implemented case is by human inspection using a toolmaker's microscope, fitted with a micrometric measuring scale [Alamin 1996, Maropoulos and Alamin 1996, Ezugwu et al. 1995, Fang 1994]. Tool wear is quantified by observing the distance between the cutting edge and the bottom of the worn surface. Human intervention in tool wear measurement can be avoided if a computer vision technique is employed to quantify the state of tool wear. The computational methods used should be capable of performing human-like reasoning in order to determine the state of tool wear, a task that is not trivial. The general principle of these methods is that light reflected by the illuminated tool wear zone and received by appropriate sensing equipment is processed in order to give an indication of the actual wear status. The tool image is usually coded into a binary thresholded image where the worn surface appears white, while the unworn as black background [Jeon and Kim 1988]. Another example of off-line optical tool wear measurement is by laser light projection through a diffraction grating on the rake face of the tool where narrow shadow stripes are formed. The deflection of the stripes corresponds to the crater wear. A 3-D image of crater wear is then obtained by processing the deflection of each one of the stripes. However, this method adds hardware complexity, as it requires accurate control of the lighting conditions. In a recent paper, tool surface topography is obtained by employing projected fringes techniques [Leopold 1996].

A 16-bit grey scale 512X512 pixel digitised tool surface image is derived by using a charged coupled device (CCD) camera which captures the tool surface image magnified by special lens. The overall image processing is completed within a few seconds and different surface maps are derived, including 3D-maps of the cutting insert, without any need for removing the insert from the tool holder mounted on the machine tool

Apart from direct wear-land measurements, optical methods can also be employed for indirect measurements. In particular, surface roughness measurement can give an indication of the wear status of the cutting tool. For instance, a laser beam can be employed to illuminate the workpiece surface. The surface roughness of the machined workpiece is then established by the scattering pattern from the surface, which is scanned across a linear array of lead selenide detectors [Janson et al. 1984].

A.1.2 Contact Gauges

These can be employed to check either the tool geometry or workpiece dimensions. In the first case the measurement is direct, whereas in the second tool wear is estimated indirectly. The sensitivity of these probes to deflections can be of the order of one micrometer. Contact gauges can be connected either electrically or with non-contact interface, like inductive transmission. They have become increasingly popular and many machine-tool manufacturers are now building them into their products as standard equipment [Sadat 1994].

A.1.3 Electrical methods

The principle of these indirect measurement techniques lies with the change of the resistance of the tool- workpiece interface, as the shape and size of the contact area change because of the wear progress. One example of this short of techniques is the bonding of a thin film of metal or graphite on the clearance surface of the tool [Uehera 1973]. As tool wear progresses, the size of the contacting surface changes, so as the electrical resistance. The change of the resistance of the film can be used as a measure of the size of tool wear.

A.1.4 Radioactive techniques

If a small quantity of radioactive material is implanted on the tool flank face, the loss of radioactivity which can be sensed with a Geiger-Muller tube, can indicate the condition of the tool. As long as tool wear is within acceptable limits, radioactive transmission from the insert can be detected by the sensor. When radioactivity seizes, the tool is considered to be worn out and ought to be replaced [Arsovski 1983]. The main drawbacks of these methods are related to its relative slow response, as well as to safety hazards, due to exposure to radioactivity.

A.1.5 Measurement of loss of tool material

As the wear process progresses during cutting, tool particles are carried away by the chip. One alternative method to tool wear measurement is by analysing the chip to obtain a measure of tool material loss. The chips can be collected and soaked into acid. The remains could then be examined for concentration in tungsten (for tungsten carbide tools), which may give a good indication of the wear size. However this is a lengthy laboratory based procedure, unsuitable for short time response.

A.2 On-line tool wear state identification

The main drawbacks of off-line tool wear measurements are [Sadat 1994]:

- The need to disengage the tool from the workpiece in order to carry out the measurement.
- They are usually time consuming.
- They can not be carried out under the severe and hostile conditions of the cutting process.
- The tool occasionally needs cleaning to reduce uncertainty or noise in measurements.

Thus, it is often required that tool wear is measured on-line (in-process). Due to the complexity and severity of the cutting process, on-line tool wear monitoring sensors are required to fulfill certain requirements [Byrne et al. 1995]:

- The measurement should take place as close to the cutting process as possible.
- The static and dynamic stiffness of the machine tool should remain unaltered.
- There should be no additional restrictions on cutting conditions or working space.
- The sensors used should be resistant to dirt and chips and immune to electromagnetic or mechanical interference.
- Sensor functionality should be independent of the particular tool or workpiece employed.
- The sensor must posses appropriate metrological characteristics.

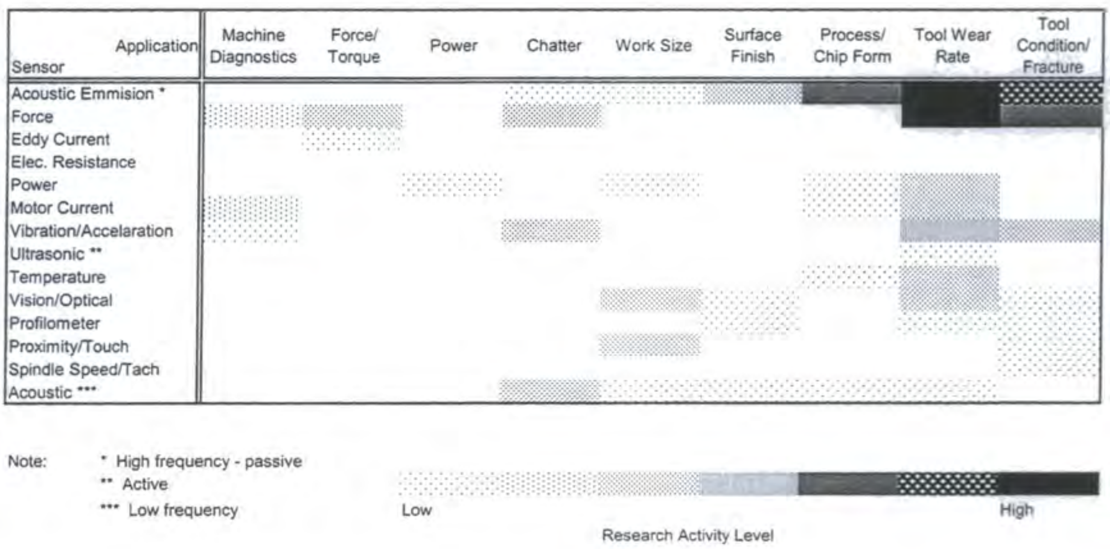

Fig. A.1 Summary of research activity in on-line cutting process monitoring [Byrne et al. 1995]

Because of the additional requirements that on-line sensors must fulfill compared to the off-line ones, real time tool wear measurement is difficult to be achieved without increased complexity and costs. This is even more true for on-line direct measurement methods, which usually rely on the availability of appropriate vision technology and are still considered to be rather complex for practical use. However, progress in computer vision is rapid and the appearance of new, vision-based tool wear measurement examples vision is rapid and the appearance of new, vision-based tool wear measurement examples is becoming increasingly frequent. A more popular alternative is to measure tool-wear indirectly on the basis of other process-states measurements, like cutting forces, acoustic emission etc., which are correlated with tool wear. The main advantages of this approach is its simplicity and lower cost. On the other hand, determining the tool wear status from such measurements is usually not a trivial task, as the interrelationship between the measured parameters and tool wear is not fully known and the sensorial information is often corrupted by noise. Among the auxiliary variables that can be measured for indirect tool wear state identification, acoustic emission (AE) of the cutting region, cutting forces and vibration measurements are the most commonly employed. A summary of the research activity for on-line tool wear identification is depicted in Fig. 5.1 [Byrne et al. 1995]. A brief description of the main on-line tool wear measuring methods follows.

A.2.1 Optical methods

The principle of these techniques is the same with off-line optical methods both in the case that wear-land is directly measured or when it is indirectly estimated by inference from other measurements (e.g. surface roughness, workpiece dimensions etc.). However, the high operational requirements for on-line functionality, tend to complicate the overall tool wear estimation problem. Thus, most of the on-line optical wear measurement techniques developed so far are more laboratory based than industrially relevant methods. However, the increasing availability of computational processing power at affordable prices is promising more on the development of highly intelligent vision based on-line tool wear measurement methods. A representative example of the current state in vision - based direct wear measurement can be found in [Park and Ulsoy 1993], where a computer vision system is reported to provide very accurate on-line flank wear estimation. This estimation is then employed by an on-line adaptive observer to calibrate the wear estimation provided by a cutting-forces based tool wear monitoring system. The integrated flank wear monitoring system has been experimentally tested and was found to provide excellent estimations even when crater wear was not negligible, as long as flank wear was the predominant type of wear. An on-line tool geometry measurement system for milling has also been developed, based on a laser displacement meter [Ryabov et al. 1996]. The system involves a preprocessing stage for removing the noise from the signals detected as well as a hybrid signal processing technique for tool failure detection. The overall system exhibits good tool wear estimation performance for flank wear greater than 40 μm. Optical methods for indirect tool wear estimation via surface roughness measurement has also been reported [Byrne et al. 1995], but they should rather be considered of laboratory nature at present. The main problems that on-line optical methods for tool wear measurement have to overcome in order to become of practical use in an industrial environment are related to noise introduced by chips and dirts, the complexity of both the filtering and the signal processing stage involved and the increased costs due to the additional equipment required. This is probably the reason for the very slow acceptance that on-line optical sensors have found in industry.

A.2.2 Forces-based wear measurement

The cutting tool and the tool holder have to withstand three main reactionary cutting forces during single point cutting operations, namely the tagential, axial and radial forces. The tangential component is the predominant force component and is generated by the rotational movement of the workpiece. The axial force is relatively smaller and is the result of the feed movement of the tool along the workpiece. The smallest force component is the radial one which depends on the approach angle. There is an affluence of sensors for force-based measurements. Six different methods of obtaining tool wear state by carrying out force related measurements can be distinguished. A brief discussion on them follows [Byrne et al. 1995].

Direct measurement dynamometers. They can provide with very accurate measurements of cutting forces. They usually come as four 3D-component force transducers, mounted under high preload between two plates and fitted under the tool for turning operations [Ko and Cho 1992], and under the workpiece or in the tool holder [Ko et al. 1995, Chen et al. 1996, Chen 1996] for multi point cutting. Due to the piezoelectric nature of the measurements, static forces over long time period can not be captured. In addition, they remain unprotected from overload and therefore are more suitable for research purposes than for practical industrial application. Their high cost is a further disadvantage.

Plates and rings. In order to achieve over-load protection, the piezoelectric force measuring elements can be embedded into thin plates, so that they undergo only a fraction of the total force. Another alternative is to fit strain gauges into such plates. However, in the latter case the rigidity of the plate is much lower. These measuring systems can easily be mounted on lathes between the turret housing and the cross slide or the turret disc. They can also be retrofitted in machining centers behind the spindle flange. However, in most of the attempts to fit them on machining centers they suffer from being exposed to disturbances like thermal expansion of the spindle, spindle oil temperature change, thermal displacement of the headstock etc.

Pins, extension sensors. These can indirectly measure the cutting force by detecting the extension of force bearing machine elements. Though fitting is easy, deciding the exact fitting positions is not straightforward. They are applicable mainly to tool failure identification, due to their low sensitivity.

Displacement measuring. Measurement of the displacement or bending of the tool may provide with valuable information about the wear status of the insert. This can be achieved by mounting non-contact displacement sensors on the tool or the spindle nose. The main drawback of these measurement techniques is that their accuracy can be questionable under the presence of chips, dirt and the coolant or lubricant used. When installed on tool turrets they can be employed for collision identification, but otherwise the information they provide is too much corrupted by noise to be of any practical use for tool wear identification purposes.

Bearings. Spindle rolling-contact bearings with strain gauges fitted on them can provide with force measurements indirectly related to the wear process. They usually require low-pass filtering to eliminate interference from the ball contact frequency. This inevitably results in filtering out high spectral components of the process signal together with the noise. Alternatively, force-measuring bushings which strain gauges mounted on the internal surface of their hollow body, can be fitted between the housing and the normal contact bearing. The main disadvantage of these bushings are that they reduce the rigidity of the spindle.

Force and torque at tool holders and spindles. The forces developing on the tool holder [Park and Ulsoy 1992] of a single point cutting tool or the torque developing on the spindle during cutting with multi point cutting tools are directly related with the cutting forces and therefore with tool wear. However, the costs involved can be high due to the need to fit sensors to each one tool holder. On the other hand, retaining the same torque measurement accuracy for the whole operating range of the machine tool involved, requires the development of a very complex sensorial system. In addition the measured signal has to be transmitted from the sensor on a non-contact basis. For single point cutting operations the are already available in the market tool holders with integrated force sensors on them.

A.2.3 Acoustic emission analysis

The cutting process produces high frequency elastic stress waves, known as acoustic emission (AE), which propagate through the machine structure. The source of these waves is sporadic energy release due to friction on the rake or flank face, plastic deformation in the shear zone, crack formation, cutting edge fracture, chip breakage, material phase transformation etc. [Hope et al. 1996, Blum and Inasaki 1990, Iwata and Moriwaki 1977]. Within the last few year several different AE sensors have been commercially available. These overcome most of the problems that older sensors had. They exhibit improved performance even under high temperatures and large coolant volumes, as well as resistance to abrasive wear from the chips. The AE transducers used are of piezoelectric nature and they are usually attached on the machine tool surface. The exact placement of the transducer is of critical importance when employing AE sensors, because of the attenuation that the transmitted through the tool structure signal undergoes. A number of different signal transmission methods have been proposed, including the use of a coolant stream for transmitting the waves [Byrne et al. 1995], inductive non-contact signal transmission to a receiver which can be fitted on the machine tool, transmission via fluids other than coolants , spring steel acoustic waveguides, fiber optics interferometry [Carolan et al. 1996] and thin film technology.

The signatures of AE signals can be classified into continuous or burst type [Sadat 1994]. Depending on the type of the AE signal captured, the signal processing methods used for obtaining tool wear related information include time series, root mean square (RMS) voltage, and count (or count rate) analysis. AE methods have found numerous applications for on-line tool wear state identification, but they are generally more successful in tool breakage detection [Grabec and Kuljanic 1994, Malakooti et al. 1995, Carolan et al. 1996, Liu and Dornfeld 1996]. Their applicability to multi-point tool wear identification is restricted by the complexity of the composite signal generated by the different cutting edges, thus making very difficult to identify uneven wear on a single insert [Wilcox et al. 1997].

A.2.4 Vibrations

The dynamic interaction between the tool, the workpiece and the machine tool during the cutting process cause vibrations on the machine structure. The main sources of these vibrations are sequential changes occurring at the shear zone of the workpiece material due to compression and sliding, as well as friction variations at the tool/workpiece interface [Hope et al. 1996]. These vibrations can be monitored with relative ease by mounting accelerometers on or near the tool holder. In case of multi point cutting tools the sensor can be fitted on the working table. The signals obtained, are correlated with tool wear and appropriate signal processing can extract tool wear state information from them [Weller et al. 1969], [Pandit and Kashou 1982]. A critical point when employing vibration measurements for indirect tool wear state identifications is the transmission path of the vibration signal from the tool/workpiece/machine tool interface towards the sensor and the impact it might have on the quality of the signal received. Vibration signal processing usually involves a feature selection phase followed by some sort of pattern recognition technique for the determination of the process state.

A.2.5 Motor current and effective power

Measurement of motor current or effective power of feed drives or spindles can be an effective and yet cheap alternative for cutting process monitoring [Altintas 1992].However, the feed power can not always give a clear indication of the wear state, since it is not solely consumed by the cutting process. The current or power signals from feed motors, can often be confounded with signals originated from the friction components in the guideways which may be of quite significant magnitude. These signals may vary with different lubrication states or traversing rates [Byrne et al. 1995]. Furthermore, due to the integrating nature of motor power measurements, timely detection of short time tool wear state degradation events can not be easily achieved before any consequent damage has already occurred.

A.2.6 Cutting Temperatures

Many of the tool wear mechanisms are thermally activated. Therefore, the cutting temperatures developed during the cutting process could give a good indication of the tool wear state as well as the wear rate, since increased wear give rise to higher temperature due to friction and energy conversion. There has been a lot of research over many years on developing cutting temperature sensors. As a result of that there has been a plethora of different temperature sensing methods [Groover and Kane 1971, Groover et al. 1977, Levy et al. 1976]. The main limitations when attempting to determine tool wear state by cutting temperature measurements are [Sadat 1994].

- The need for calibrating temperature with regards to tool wear state.
- There should be a way of knowing in advance the relationship between tool wear and cutting temperatures under varying cutting conditions.
- Tool chipping and fracture can not be detected as they are not thermally activated wear types.
- Low reliability due to slow measurement response time.
- No applicability for intermittent cutting processes.

Thus far, all the methods developed for wear state recognition via cutting temperature measurements are still of pure laboratory nature and almost unfeasible for intermittent cutting processes like milling and drilling [Byrne et al. 1995].

A.2.7 Tool-Workpiece distance variations

As tool wear progresses the distance between the machined surface and a fixed point on the cutting tool varies. This distance, can be sensed by proximity sensors, and provide with an indication of the tool wear status. An example of workpiece-tool distance measurement for cutting process state identification can be found in [El Gomayel and Bregger 1986], where an electromagnetic probe is employed to measure the workpiece diameter change as tool wear progresses. The variations in the gap between the sensor and the machined surface induces a voltage output indirectly related to the wear status of the tool. The advantage of these methods is that they do not require any contact with the workpiece material while the main drawbacks are [Alamin 1996]:

- Strict operating limits due to sensitivity of the probes to high temperatures
- Possible interference by chips.
- Reduced accuracy in cased of deflections or misalignments on the machine tool or tool holder or when vibrations occur.
- Increased probability of false measurements because of insert or toolholder thermal expansion.
- Time lag between the exact time of cut and the time when the workpiece diameter is measured.

A.2.8 Surface roughness measurement

The surface quality of the machined workpiece is influenced by the tool wear state and progress. Therefore, surface roughness measurements should give an indication of tool wear. On-line surface roughness measurements can be obtained using contacting or non-contacting sensors [Janson 1984]. The applicability of these methods is limited because of the time lag between the cutting and measuring time and the possibility of false measurements if the workpiece surface has not been cleaned.

A.2.9 Sound

Low frequency spectra of sound produced from the cutting process can also be analysed to give an indication of the wear state [Sadat and Raman 1987]. The main difficulty in this approach arises due to noise produced by causes other than the machining process, which often overlaps with the spectrum of the measuring signal. Sound monitoring-based tool wear estimation is also still at laboratory research stage, without any practical industrial applications so far.

The simultaneous use of multiple sensors (**sensor fusion**) can compensate for the weaknesses of each individual sensor with respect to accurate, on-line tool wear state identification. Sensor fusion technology, integrated with intelligence capabilities has already found numerous applications in several areas [Dasarathy 1997]. This technology offers vast area for improvement of current on-line tool wear measurement and process monitoring systems. [Rangwala and Dornfeld 1990, Leem et al. 1995, Hope et al. 1996, Chryssolouris et al. 1992].

APPENDIX B: MATERIAL CLASSES

Group	AISI	Werkstoff	DIN	BS	AFNOR	SS
1	1006	1.0201	St 36	-	Fd 5	1160
Very Soft Steel	1010	1.1121	Ck 10	045 M 10	XC 10	1265
	-	1.1121	St 37-1	4360 40 A		1300
	A27 65-35	1.0443	GS-45	A1	E 23-45 M	1305
	-	1.0416	GS-38	-	230-400 M	1306
	A570-36	1.0038	RSt 37-2	4360 40 C	E 24-2 NE	1311
	A573-81 65	1.0116	St 37-3	4360 40 B	E 24-U	1312
	A515 65	1.0345	H I	1 501 161	A 37 CP	1330
	1015	1.0401	C 15	080 M 15	CC 12	1350
	1022	1.1133	GS-20Mn 5	120 M 19	20 M 5	1410
	A36	-	St 44-2	4360 43 A	NFA 35-501 E 28	1411
	A573-81	1.0144	St 44-3	4360 43 C	E 28-3	1412
	-	-	StE 320-3Z	1 501 160	-	1421
	-	1.0425	H II	-	A 42 CP	1432
	1025	1.1158	Ck 25	050 A 20	X C 25	1450

Group	AISI	Werkstoff	DIN	BS	AFNOR	SS
2	1213	1.0715	9 SMn 28	230 M 07	S 250	1912
Free Cutting Steel	(12L13)	1.0718	9 SMnPb 28	-	S 250 Pb	1914
	-	1.0723	15 S 20	210 A 15	S 300	1922
	(12L14)	1.0737	9 SMnPb 36	-	S 300 Pb	1926
	(12L13)	1.0718	9 SMnPb 28	-	-	1940
	1140	1.0726	35 S 20	212 M 36	35 MF 4	1957
	1151	1.0727	45 S 20	212 M 44	45 MF 4	1973

Group	AISI	Werkstoff	DIN	BS	AFNOR	SS
3	1015	1.1141	Ck 15	080 M 15	XC 18	1370
Structural Steel	A27 70-36	1.0551	GS-52	A2	280-480 M	1505
	1035	1.0501	C 35	060 A 35	AF 55 C 35	1550
	1035	1.1181	Ck 35	080 A 32	XC 38	1572
	A148 80-40	1.0553	GS-60	A3	320-560 M	1606
	1043	1.0503	C 45	080 M 46	AF 65 C 45	1650
	1055	1.0535	C 55	070 M 55	-	1655
	1042	1.1191	Ck 45	080 A 47	XC 45	1660
	A537 1	1.0473	19 Mn 6	1 501 224	A 52 CP	2101
	A662 C	1.0436	ASt 45	1 501 224	A 48 FP	2103
	A738	1.0577	ASt 52	1 501 224	A 52 FP	2107
	-	1.057	St 52-3	4360 50 B	E 36-3	2132
	A572-60	-	17 MnV 6	4360 55 E	NFA 35-501 E 36	2142
	A572-60	1.89	StE 380	4360 55 E	-	2145

APPENDIX B (CONT): MATERIAL CLASSES

Group	AISI	Werkstoff	DIN	BS	AFNOR	SS
4 High Carbon Steel	1042	1.1191	Ck 45	080 M 46	-	1672
	1064	1.1221	Ck 60	060 A 62	XC 65	1678
	1070	1.1231	Ck 67	070 A 72	XC 68	1770
	1080	1.1248	Ck 75	060 A 78	XC 75	1774
	1095	1.1274	Ck 101	060 A 96	XC 100	1870
	9254	1.0904	55 Si 7	250 A 53	55 S 7	2090
	1335	1.1167	36 Mn 5	150 M 36	40 M 5	2120
	5120	1.0841	St 52-3	150 M 19	20 MC 5	2172
	A387 12-2	1.7337	16 CrMo 4 4	1 501 620	15 CD 4.5	2216
	A182 F-22	1.738	10 CrMo 9 10	1 501 622	12 CD 9.10	2218
	4130	1.7218	25 CrMo 4	CDS 110	25 CD 4	2225
	6150	1.8159	50 CrV 4	735 A 50	50 CV 4	2230
	4135	1.233	35 CrMo 4	708 A 37	34 CD 4	2234
	-	1.8515	31 CrMo 12	722 M 24	30 CD 12	2240
	4142	1.2332	47 CrMo 4	708 M 40	42 CD 4	2244
	4140	1.7225	42 CrMo 4	708 M 40	42 CD 4	2244
	5140	1.7045	42 Cr 41	530 A 40	42 C 4 TS	2245
	5155	1.7176	55 Cr 31	527 A 60	55 C 3	2253
	52100	1.3505	100 Cr 6	534 A 99	100 C 6	2258
	8620	1.6523	21 NiCrMo 2	805 H 20	20 NCD 2	2506
	5115	1.7131	16 MnCr 5	527 M 17	16 MC 5	2511
	A204A	1.5415	15 Mo 3	1 501 240	15 D 3	2912
	A355A	1.8509	42 CrAlMo 7	905 M 39	40 CAD 6.12	2940
	403	1.4	X6 Cr 13	403 S 17	Z 8 C 13	2301
	(410S)	1.4001	X7 Cr 14	(403 S 17)	Z 8 C 13	2301
	410	(1.4006)	G-X10 Cr 13	410 S 21	Z 10 C 13 M	2302
	405	1.4724	X6 CrAl 13	405 S 17	Z 8 CA 12	-
	430	1.4016	X6 Cr 17	430 S 17	Z 8 C 17	2320
	434	1.4113	X6 CrMo 17	434 S 17	-	2325
	416	1.4005	X12CrS 13	416 S 21	Z11 CF 13	2380
	430F	1.4104	X12 CrMoS 17	420 S 37	Z 13 CF 17	2383
	409	1.4512	X5 CrTi 12	409 S 19	Z 6 CT 12	-
	430Ti	1.451	X6 CrTi 17	-	Z 4 CT 17	-

Group	AISI	Werkstoff	DIN	BS	AFNOR	SS
5 Normal Tool Steel	W 1	1.1545	C105W1	BW1A	Y 105	1880
	-	1.2108	90 CrSi 5	-	-	2092
	O 1	1.251	100 MnCrW 4	BO1	8 Mo 8	2140
	-	-	31 NiCrMo 13 4	830 M 31	-	2534
	4340	1.6582	34 CrNiMo 6	817 M 40	35 NCD 6	2541
	-	1.6746	32 NiCrMo 14 5	830 M 31	35 NCD 14	-
	S 1	1.2542	45 WCrV 7	BS1	55 WC 20	2710
	420	1.4021	X20 Cr 13	420 S 37	Z 20 C 13	2303
	(420)	1.4028	X30 Cr 13	420 S 45	Z 30 C 13	-2304
	(420)	1.4031	X40 Cr 13	-	Z 40 C 14	-2304
	-	1.4923	X22 CrMoV 12 1	-	-	-
	431	1.4057	X20 CrNi 17 2	431 S 29	Z 15 CN 16-02	2321
	440B	1.4112	X90 CrMoV 18	-	-	-

APPENDIX B (CONT): MATERIAL CLASSES

Group	AISI	Werkstoff	DIN	BS	AFNOR	SS
8 Free Cutting Stainless Steel	304	1.4301	X5 CrNi 18 10	304 S 10	Z 5 CN 18-09	2333
	304H	1.4948	X6 CrNi 18 11	304 S 51	Z 5 CN 18-09	2333
	303	1.4305	X10 CrNiS 18 9	303 S 31	Z 8 CNF 18-09	2346
	304L	1.4306	X2 CrNi 18 10	304 S 11	Z 3 CN 19-11	2352
	305	1.4312	X8 CrNi 18 12	305 S 19	-	-
	302	-	X12 CrNi 18 9	302 S 31	Z 10 CN 18-09	2330
	301	1.431	X12 CrNi 17 7	301 S 21	Z 11 CN 17-08	2331
	CF-8	1.4308	X6 CrNi 18 9	304 C 15	Z 6 CN 18-10 M	2333

Group	AISI	Werkstoff	DIN	BS	AFNOR	SS
9 Moderately Difficult Stainless Steel	321	1.4541	X6 CrNiTi 18 10	321 S 31	Z 6 CNT 18-10	2337
	347	1.455	X6 CrNiNb 18 1	347 S 31	Z 6 CNNb 18-10	2338
	316	1.4436	X5 CrNiMo 17 1	316 S 33	Z 6 CND 19-12-03	2343
	316Ti	1.4571	X8 CrNiMoTi 17	320 S 31	-	-
	316	1.4401	X5 CrNiMo 17 1	316 S 31	Z 7 CND 17-11-02	2347
	316L	1.4404	X2 CrNiMo 17 1	316 S 11	Z 3 CND 17-12-02	2348
	316Ti	1:4571	X6 CrNiMoTi 17	320 S 31	Z 6 CNDT 17-12-0	2350
	316L	1.4435	X2 CrNiMo 18 1	316 S 13	Z 3 CND 18-14-03	2353
	317	(1.4449)	X5 CrNiMo 17 1	317 S 16	-	-
	310S	1.4845	X12 CrNi 25 20	310 S 16	Z 12 CN 25-20	2361
	317L	1.4438	X2 CrNiMo 18 1	317 S 12	Z 2 CND 19-15-04	2367
	-	1.4418	X4 CrNiMo 16 5	-	Z 6 CND 16-04-01	2387
	304LN	1.4311	X2 CrNiN 18 10	304 S 61	Z 2 CN 18-10 AZ	2371
	309S	1.4833	X6 CrNi 22 13	309 S 13	Z 15 CN 24-13	-
	CF-8M	1.4408	X6 CrNiMo 18 1	304 C 15	-	2343

Group	AISI	Werkstoff	DIN	BS	AFNOR	SS
10 Stainless Steel Difficult to Machine	S44400	1.4521	X1 CrMoTi 18 2	-	-	2326
	202	1.4371	X3 CrMnNiN 18	284 S 16	Z 8 CMN 18-08-05	-
	S30815	1.4893	X8 CrNiNb 11	-	-	2368
	CA6-NM	1.4313	(G-)X4 CrNi 13	(425 C 11)	Z 4 CND 13-04 M	2385
	660	1.498	X5 NiCrTi 25 15	-	Z 8 NCTV 25-15 B	2570
	(S31726)	1.4439	X2 CrNiMoN 17	-	Z 3 CND 18-14-06	-
	330	1.4864	X12 NiCrSi 16	NA 17	Z 12 NCS 35-16	-
	309	-	X15 CrNi 23 13	309 S 24	Z 15 CNS 20-12	-
	310	1.4841	X15 CrNiSi 25 2	314 S 31	Z 15 CNS 25-20	-
	(329)	(1.446)	X4 CrNiMo 27 5	-	Z 5 CND 27-05 AZ	2324
	S32304	1.4362	X2 CrNiN 23 4	-	Z 2 CN 23-04 AZ	2327
	S30415	1.4891	X5 CrNiNb 18 1	-	-	2372
	316LN	1.4406	X2 CrNiMoN 17	316 S 61	Z2 CND 17-12 AZ	2375
	316LN	1.4429	X2 CrNiMoN 17	316 S 63	Z2 CND 17-13 AZ	2375
	S31500	1.4417	X2 CrNoMoSi 18	-	-	2376
	S31803	1.4462	X2 CrNiMoN 22	318 S 13	Z3 CND 22-05 AZ	2377
	CN-7M	1.4539	(G-)X1 NiCrMoC	-	Z1 NCDU 25-02 M	2564
	No8904	1.4539	X2 NiCrMoCu 2	904 S 13	Z1 NCDU 25-20	2562
	S31254	1.4547	X1 CrNiMoN 20	-	-	2378
	S31753	-	X2 CrNiMoN 18	-	-	-
	-	-	X2 CrNiMoN 25	-	-	-
	S32750	1.441	X3 CrNiMoN 25	-	-	2328
	-	-	X5 NiCrN 35 25	-	-	-
	S17400	1.4542	X5 CrNiCuNb 17	-	-	-

APPENDIX C: PARTIAL LIST OF TRAINING PATTERNS

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
5	3	35	0.5352	115.44	17.907
3	3	10	0.6468	269.44	16.735
3	3	10	0.5314	254.09	20.452
3	1	35	0.0683	225.28	36.184
3	3	20	0.8888	239.88	0.920
1	2	35	0.3473	231.35	57.058
4	2	35	0.3583	187.49	13.753
3	1	35	0.2676	265.12	11.071
1	2	10	0.4930	405.58	18.032
1	1	35	0.0980	375.33	16.551
3	1	10	0.3406	365.34	9.450
2	2	35	0.2811	199.01	99.659
2	1	10	0.1731	398.73	15.837
2	1	35	0.2497	387.84	5.084
4	1	10	0.3076	278.11	13.606
3	3	20	0.9025	209.94	1.615
2	2	20	0.4950	267.64	8.155
1	2	10	0.3342	409.84	17.374
3	1	20	0.2342	300.32	12.416
9	3	35	0.4455	87.24	23.436
4	1	35	0.2846	237.23	8.991
1	3	35	0.5146	177.19	42.925
1	1	10	0.0995	467.02	19.397
4	2	10	0.3217	218.74	24.986
8	3	35	0.4691	142.55	25.387
8	2	20	0.3709	116.39	24.661
5	3	10	0.5435	189.89	16.156
1	2	20	0.4684	246.19	27.140
8	3	20	0.7642	110.70	22.494
4	1	35	0.2461	221.61	12.986
1	2	35	0.4564	258.65	16.794
5	2	10	0.3665	201.25	18.308
8	2	35	0.3766	158.51	25.619
10	1	35.	0.3303	83.89	29.640
4	2	10	0.3181	212.43	27.710
2	3	20	0.6624	194.06	10.149
5	1	10	0.2405	221.15	19.888
2	2	35	0.4091	279.29	7.988
1	1	20	0.0587	405.47	18.053
5	3	35	0.5379	160.57	4.028
5	1	20	0.0595	196.92	29.790
5	1	20	0.0507	219.04	19.089
9	3	35	0.4840	82.69	28.743
4	2	35	0.4797	162.71	11.623
1	3	10	0.9938	389.88	14.423
2	3	10	0.6637	313.70	17.356
1	3	20	0.9661	300.25	1.037
5	2	35	0.4231	177.01	7.700
4	1	35	0.3114	226.18	10.828
8	1	35	0.1320	190.25	17.700
5	1	20	0.2127	207.92	16.218
1	2	35	0.3504	264.51	30.774
5	3	35	0.8908	155.97	1.236
4	3	10	0.6571	217.20	17.677
9	2	35	0.4968	116.33	11.314
3	2	20	0.3225	243.28	20.431
3	2	35	0.4662	162.41	23.935
1	2	35	0.3899	266.89	22.242
4	3	35	0.8160	152.08	2.448
4	3	10	0.4756	191.38	28.109
8	3	20	0.7062	116.56	18.823
4	2	20	0.4930	168.60	13.737
10	2	35	0.3101	71.44	22.534
8	1	20	0.3222	155.52	13.749

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
2	1	35	0.0922	339.05	13.071
5	1	10	0.1899	231.06	17.611
2	3	20	0.4839	242.06	8.594
2	1	10	0.0644	411.53	18.401
4	2	10	0.2898	210.38	28.674
3	3	35	0.5461	182.79	6.587
9	2	35	0.2713	107.37	15.464
5	3	20	0.9502	158.86	1.394
5	1	35	0.1933	185.96	19.809
10	1	35	0.3175	106.12	16.322
2	2	20	0.2692	236.95	68.880
9	1	35	0.3205	106.51	31.799
5	1	35	0.1231	200.12	16.197
8	1	20	0.2931	150.41	15.865
5	3	10	0.8358	202.09	13.097
5	3	35	0.9723	117.30	3.531
5	1	35	0.1907	207.76	11.945
2	3	35	0.8965	240.23	1.226
4	3	20	0.7241	179.04	2.402
1	1	10	0.3051	471.60	13.375
3	3	10	0.8820	278.11	15.016
5	3	20	0.9197	140.06	2.687
2	3	10	0.5111	322.37	15.786
1	1	35	0.1276	347.03	21.730
10	2	35	0.2817	70.97	23.187
1	1	35	0.0532	379.46	19.197
9	3	35	0.4031	116.67	7.743
5	2	20	0.3839	184.94	10.216
2	2	35	0.3826	241.31	18.421
2	1	20	0.0786	348.78	16.674
2	1	35	0.2923	342.74	8.527
4	3	20	0.6411	136.33	11.275
2	1	35	0.3290	336.33	8.949
8	3	20	0.6125	129.39	13.120
2	3	35	0.7562	193.64	5.089
8	2	35	0.3589	132.57	32.725
5	3	35	0.5342	106.20	26.158
4	3	10	0.6246	202.62	22.803
3	2	10	0.2585	282.65	20.043
5	1	20	0.2268	208.36	15.773
4	3	35	0.7011	134.25	5.861
4	3	10	0.5113	231.54	13.986
10	2	35	0.3303	80.21	13.685
4	2	20	0.4240	163.61	23.301
4	2	10	0.3771	252.37	15.081
3	2	10	0.4863	255.98	28.644
5	3	35	0.8386	114.53	5.772
5	1	10	0.1691	245.08	13.957
2	3	10	0.4707	320.25	16.152
5	1	10	0.3072	254.40	9.565
4	1	10	0.1032	285.42	17.805
5	1	10	0.3148	217.39	19.629
4	3	10	0.8833	222.60	16.155
4	1	10	0.3203	301.67	9.241
2	3	35	0.7080	252.44	1.822
4	1	35	0.3153	243.01	7.746
5	3	35	0.8286	164.51	1.175
4	2	35	0.3470	193.94	12.902
2	3	20	0.9563	217.80	2.462
4	2	35	0.3140	165.90	34.316
9	3	35	0.9716	86.77	23.927
9	2	35	0.2997	118.68	10.465
3	3	20	0.8777	234.41	1.056
4	2	35	0.2921	199.11	18.452

APPENDIX C (CONT): PARTIAL LIST OF TRAINING PATTERNS

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
5	2	10	0.4040	184.72	24.820
4	3	35	0.6711	149.92	3.979
3	3	10	0.4397	253.62	20.581
1	1	10	0.0978	475.64	17.936
3	2	20	0.4752	200.00	17.567
3	2	35	0.4622	196.61	10.211
1	2	10	0.3215	412.74	16.946
3	3	35	0.9576	194.10	1.265
9	3	35	0.5311	108.98	10.040
5	2	35	0.3037	130.73	68.787
10	1	35	0.2234	109.15	15.514
4	1	35	0.1776	235.93	10.996
3	2	20	0.4166	188.30	32.627
5	3	35	0.4856	154.26	6.292
8	3	20	0.9204	108.05	24.459
3	2	20	0.3576	240.63	16.308
4	1	35	0.2256	217.33	14.673
3	2	10	0.4834	300.94	15.991
3	1	10	0.2105	316.53	20.995
3	2	20	0.3558	207.02	32.437
2	3	35	0.8627	254.04	1.054
2	2	20	0.3863	198.83	59.674
3	2	35	0.2688	168.69	88.654
8	3	35	0.5579	150.39	23.229
2	1	20	0.0773	321.20	24.497
2	3	10	0.7814	327.80	14.895
4	3	35	0.5948	120.76	13.281
3	2	35	0.4492	160.17	28.190
1	2	10	0.4840	412.69	16.952
2	2	35	0.3368	236.56	28.302
9	1	35	0.1789	122.00	18.391
1	2	20	0.3602	234.44	67.296
8	2	20	0.2844	111.41	28.919
4	2	35	0.4337	150.22	21.991
2	3	20	0.7997	210.43	4.467
5	3	35	0.4928	125.70	15.153
4	2	10	0.3300	263.73	12.910
4	3	10	0.7211	195.66	25.920
2	1	20	0.2729	352.04	11.108
1	2	20	0.2871	219.89	162.598
2	1	10	0.0948	400.86	18.493
5	1	35	0.1688	201.37	14.309
5	2	20	0.4357	161.75	13.589
2	3	35	0.5277	181.41	17.599
2	3	20	0.9734	222.56	2.146
2	2	35	0.3727	251.20	16.488
1	2	10	0.4471	407.51	17.731
5	1	20	0.3281	226.59	9.666
1	2	35	0.3762	213.01	66.481
5	3	10	0.5378	188.42	16.583
4	2	10	0.3900	210.92	28.416
1	3	10	0.5789	382.48	15.443
4	3	35	0.5763	100.05	32.327
8	2	20	0.3029	100.00	42.861
4	2	20	0.4546	147.35	30.825
1	1	20	0.0747	372.77	24.675
2	3	10	0.6465	299.89	20.297
9	1	35	0.2568	129.90	14.092
2	2	35	0.3670	180.28	76.306
3	1	20	0.3352	309.82	9.519
3	3	20	0.4494	160.50	34.903
4	3	35	0.6628	132.52	7.003
5	2	10	0.4754	179.10	27.701
9	2	35	0.4933	105.92	16.304

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
4	2	10	0.2862	269.72	11.925
3	2	10	0.2754	315.41	13.502
5	3	20	0.7478	142.49	4.176
5	2	20	0.4475	151.65	16.973
8	2	35	0.3228	164.61	24.656
3	1	20	0.2213	295.65	13.602
10	2	35	0.2508	84.35	11.020
10	1	35	0.1808	69.88	48.954
8	1	35	0.2118	196.59	15.348
4	3	10	0.9319	234.32	13.387
8	2	35	0.4400	136.12	31.066
3	3	35	0.5781	123.34	33.001
1	2	35	0.4513	298.99	9.101
1	1	10	0.1997	461.76	16.664
3	3	20	0.8748	175.13	3.989
1	1	20	0.2661	365.93	18.304
8	3	20	0.8771	95.77	37.113
8	1	35	0.3123	170.53	27.454
1	3	20	0.5155	241.85	14.314
2	2	20	0.4609	293.80	6.434
2	3	10	0.5024	304.80	19.183
1	3	20	0.6191	316.57	2.661
8	1	20	0.0875	151.77	15.566
2	2	10	0.2583	349.50	17.004
4	1	10	0.1986	278.85	15.699
1	3	35	0.5094	226.48	14.534
2	1	20	0.1971	336.13	15.119
2	3	20	0.7631	204.76	5.651
3	2	35	0.4939	207.52	6.671
1	3	10	0.4763	370.44	17.309
1	1	20	0.1377	382.91	18.137
5	3	10	0.6735	203.27	12.843
9	3	35	0.9868	92.04	19.114
2	1	35	0.2011	381.23	5.905
8	2	20	0.4923	119.45	22.439
1	3	35	0.4076	208.08	38.544
3	2	35	0.3412	142.01	102.607
5	2	35	0.4123	157.91	13.558
10	2	35	0.4952	61.66	42.480
1	2	35	0.4111	190.48	86.144
1	2	35	0.3549	310.63	14.587
1	1	35	0.0596	346.15	28.152
4	3	10	0.8972	229.69	14.403
5	1	20	0.2215	197.15	20.478
9	1	35	0.1775	109.77	28.438
5	3	35	0.7442	145.43	2.700
5	2	35	0.3328	178.50	13.907
5	2	20	0.3404	174.44	17.869
9	1	35	0.3193	112.86	25.052
3	2	10	0.4872	264.81	25.352
2	1	20	0.2928	309.64	19.666
4	3	10	0.9017	223.27	15.978
3	3	10	0.9410	271.94	16.214
3	1	35	0.1066	266.48	14.490
5	3	10	0.5830	168.44	24.200
4	3	10	0.7545	215.26	18.269
8	2	35	0.4568	173.74	22.288
3	2	20	0.4903	162.47	41.026
2	2	10	0.4890	335.74	19.529
8	2	35	0.4078	170.74	23.029
8	2	20	0.4827	146.89	10.564
8	3	20	0.6426	127.04	13.976
8	3	20	0.8279	122.36	15.911
1	2	20	0.3287	297.90	29.442

APPENDIX D: PARTIAL LIST OF VALIDATION PATTERNS

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
1	1	10	0.0750	456.75	23.314
1	1	10	0.0750	458.50	22.911
1	1	10	0.0750	460.25	22.516
1	1	10	0.0750	462.00	22.129
1	1	10	0.0750	463.75	21.751
1	1	10	0.0750	465.50	21.381
1	1	10	0.0750	467.25	21.018
1	1	10	0.0750	469.00	20.662
1	1	10	0.0750	470.75	20.314
1	1	10	0.0750	472.50	19.973
1	1	10	0.0750	474.25	19.639
1	1	10	0.0750	476.00	19.312
1	1	10	0.0750	477.75	18.991
1	1	10	0.0750	479.50	18.677
1	1	10	0.0750	481.25	18.369
1	1	10	0.0750	483.00	18.068
1	1	10	0.0750	484.75	17.772
1	1	10	0.0750	486.50	17.482
1	1	10	0.0750	488.25	17.198
1	1	10	0.1000	456.75	21.435
1	1	10	0.1000	458.50	21.065
1	1	10	0.1000	460.25	20.702
1	1	10	0.1000	462.00	20.346
1	1	10	0.1000	463.75	19.999
1	1	10	0.1000	465.50	19.658
1	1	10	0.1000	467.25	19.324
1	1	10	0.1000	469.00	18.997
1	1	10	0.1000	470.75	18.677
1	1	10	0.1000	472.50	18.364
1	1	10	0.1000	474.25	18.057
1	1	10	0.1000	476.00	17.756
1	1	10	0.1000	477.75	17.461
1	1	10	0.1000	479.50	17.172
1	1	10	0.1000	481.25	16.889
1	1	10	0.1000	483.00	16.612
1	1	10	0.1000	484.75	16.340
1	1	10	0.1000	486.50	16.074
1	1	10	0.1000	488.25	15.812
1	1	10	0.1500	456.75	19.042
1	1	10	0.1500	458.50	18.713
1	1	10	0.1500	460.25	18.390
1	1	10	0.1500	462.00	18.075
1	1	10	0.1500	463.75	17.766
1	1	10	0.1500	465.50	17.463
1	1	10	0.1500	467.25	17.167
1	1	10	0.1500	469.00	16.876
1	1	10	0.1500	470.75	16.592
1	1	10	0.1500	472.50	16.313
1	1	10	0.1500	474.25	16.041
1	1	10	0.1500	476.00	15.773
1	1	10	0.1500	477.75	15.512
1	1	10	0.1500	479.50	15.255
1	1	10	0.1500	481.25	15.003
1	1	10	0.1500	483.00	14.757
1	1	10	0.1500	484.75	14.516
1	1	10	0.1500	486.50	14.279
1	1	10	0.1500	488.25	14.047
1	1	10	0.2000	456.75	17.508
1	1	10	0.2000	458.50	17.205
1	1	10	0.2000	460.25	16.909
1	1	10	0.2000	462.00	16.618
1	1	10	0.2000	463.75	16.334
1	1	10	0.2000	465.50	16.056
1	1	10	0.2000	467.25	15.783
1	1	10	0.2000	469.00	15.517
1	1	10	0.2000	470.75	15.255
1	1	10	0.2000	472.50	14.999
1	1	10	0.2000	474.25	14.748
1	1	10	0.2000	476.00	14.502
1	1	10	0.2000	477.75	14.262
1	1	10	0.2000	479.50	14.026
1	1	10	0.2000	481.25	13.795
1	1	10	0.2000	483.00	13.568
1	1	10	0.2000	484.75	13.346
1	1	10	0.2000	486.50	13.128
1	1	10	0.2000	488.25	12.915
1	1	10	0.2500	456.75	16.403
1	1	10	0.2500	458.50	16.120
1	1	10	0.2500	460.25	15.842
1	1	10	0.2500	462.00	15.570
1	1	10	0.2500	463.75	15.304
1	1	10	0.2500	465.50	15.043
1	1	10	0.2500	467.25	14.788
1	1	10	0.2500	469.00	14.538
1	1	10	0.2500	470.75	14.293
1	1	10	0.2500	472.50	14.053
1	1	10	0.2500	474.25	13.818
1	1	10	0.2500	476.00	13.588
1	1	10	0.2500	477.75	13.362
1	1	10	0.2500	479.50	13.141
1	1	10	0.2500	481.25	12.924
1	1	10	0.2500	483.00	12.712
1	1	10	0.2500	484.75	12.504
1	1	10	0.2500	486.50	12.300
1	1	10	0.2500	488.25	12.100
1	1	10	0.3000	456.75	15.553
1	1	10	0.3000	458.50	15.284
1	1	10	0.3000	460.25	15.021
1	1	10	0.3000	462.00	14.763
1	1	10	0.3000	463.75	14.510
1	1	10	0.3000	465.50	14.263
1	1	10	0.3000	467.25	14.021
1	1	10	0.3000	469.00	13.784
1	1	10	0.3000	470.75	13.552
1	1	10	0.3000	472.50	13.324
1	1	10	0.3000	474.25	13.102
1	1	10	0.3000	476.00	12.883
1	1	10	0.3000	477.75	12.669
1	1	10	0.3000	479.50	12.460
1	1	10	0.3000	481.25	12.254
1	1	10	0.3000	483.00	12.053
1	1	10	0.3000	484.75	11.856
1	1	10	0.3000	486.50	11.663
1	1	10	0.3000	488.25	11.473
1	2	10	0.3000	382.50	22.200
1	2	10	0.3000	385.00	21.693
1	2	10	0.3000	387.50	21.200
1	2	10	0.3000	390.00	20.722

APPENDIX D (CONT): PARTIAL LIST OF VALIDATION PATTERNS

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
1	2	10	0.3000	392.50	20.257
1	2	10	0.3000	395.00	19.805
1	2	10	0.3000	397.50	19.367
1	2	10	0.3000	400.00	18.940
1	2	10	0.3000	402.50	18.526
1	2	10	0.3000	405.00	18.123
1	2	10	0.3000	407.50	17.732
1	2	10	0.3000	410.00	17.351
1	2	10	0.3000	412.50	16.980
1	2	10	0.3000	415.00	16.620
1	2	10	0.3000	417.50	16.269
1	2	10	0.3000	420.00	15.928
1	2	10	0.3000	422.50	15.596
1	2	10	0.3000	425.00	15.273
1	2	10	0.3000	427.50	14.958
1	2	10	0.3500	382.50	22.200
1	2	10	0.3500	385.00	21.693
1	2	10	0.3500	387.50	21.200
1	2	10	0.3500	390.00	20.722
1	2	10	0.3500	392.50	20.257
1	2	10	0.3500	395.00	19.805
1	2	10	0.3500	397.50	19.367
1	2	10	0.3500	400.00	18.940
1	2	10	0.3500	402.50	18.526
1	2	10	0.3500	405.00	18.123
1	2	10	0.3500	407.50	17.732
1	2	10	0.3500	410.00	17.351
1	2	10	0.3500	412.50	16.980
1	2	10	0.3500	415.00	16.620
1	2	10	0.3500	417.50	16.269
1	2	10	0.3500	420.00	15.928
1	2	10	0.3500	422.50	15.596
1	2	10	0.3500	425.00	15.273
1	2	10	0.3500	427.50	14.958
1	2	10	0.4000	382.50	22.200
1	2	10	0.4000	385.00	21.693
1	2	10	0.4000	387.50	21.200
1	2	10	0.4000	390.00	20.722
1	2	10	0.4000	392.50	20.257
1	2	10	0.4000	395.00	19.805
1	2	10	0.4000	397.50	19.367
1	2	10	0.4000	400.00	18.940
1	2	10	0.4000	402.50	18.526
1	2	10	0.4000	405.00	18.123
1	2	10	0.4000	407.50	17.732
1	2	10	0.4000	410.00	17.351
1	2	10	0.4000	412.50	16.980
1	2	10	0.4000	415.00	16.620
1	2	10	0.4000	417.50	16.269
1	2	10	0.4000	420.00	15.928
1	2	10	0.4000	422.50	15.596
1	2	10	0.4000	425.00	15.273
1	2	10	0.4000	427.50	14.958
1	2	10	0.4500	382.50	22.200
1	2	10	0.4500	385.00	21.693
1	2	10	0.4500	387.50	21.200
1	2	10	0.4500	390.00	20.722
1	2	10	0.4500	392.50	20.257
1	2	10	0.4500	395.00	19.805

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
1	2	10	0.4500	397.50	19.367
1	2	10	0.4500	400.00	18.940
1	2	10	0.4500	402.50	18.526
1	2	10	0.4500	405.00	18.123
1	2	10	0.4500	407.50	17.732
1	2	10	0.4500	410.00	17.351
1	2	10	0.4500	412.50	16.980
1	2	10	0.4500	415.00	16.620
1	2	10	0.4500	417.50	16.269
1	2	10	0.4500	420.00	15.928
1	2	10	0.4500	422.50	15.596
1	2	10	0.4500	425.00	15.273
1	2	10	0.4500	427.50	14.958
1	3	10	0.5000	347.25	21.796
1	3	10	0.5000	349.50	21.300
1	3	10	0.5000	351.75	20.818
1	3	10	0.5000	354.00	20.350
1	3	10	0.5000	356.25	19.896
1	3	10	0.5000	358.50	19.454
1	3	10	0.5000	360.75	19.025
1	3	10	0.5000	363.00	18.607
1	3	10	0.5000	365.25	18.202
1	3	10	0.5000	367.50	17.808
1	3	10	0.5000	369.75	17.424
1	3	10	0.5000	372.00	17.051
1	3	10	0.5000	374.25	16.689
1	3	10	0.5000	376.50	16.336
1	3	10	0.5000	378.75	15.992
1	3	10	0.5000	381.00	15.658
1	3	10	0.5000	383.25	15.333
1	3	10	0.5000	385.50	15.016
1	3	10	0.5000	387.75	14.707
1	3	10	0.6000	347.25	21.796
1	3	10	0.6000	349.50	21.300
1	3	10	0.6000	351.75	20.818
1	3	10	0.6000	354.00	20.350
1	3	10	0.6000	356.25	19.896
1	3	10	0.6000	358.50	19.454
1	3	10	0.6000	360.75	19.025
1	3	10	0.6000	363.00	18.607
1	3	10	0.6000	365.25	18.202
1	3	10	0.6000	367.50	17.808
1	3	10	0.6000	369.75	17.424
1	3	10	0.6000	372.00	17.051
1	3	10	0.6000	374.25	16.689
1	3	10	0.6000	376.50	16.336
1	3	10	0.6000	378.75	15.992
1	3	10	0.6000	381.00	15.658
1	3	10	0.6000	383.25	15.333
1	3	10	0.6000	385.50	15.016
1	3	10	0.6000	387.75	14.707
1	3	10	0.7000	347.25	21.796
1	3	10	0.7000	349.50	21.300
1	3	10	0.7000	351.75	20.818
1	3	10	0.7000	354.00	20.350
1	3	10	0.7000	356.25	19.896
1	3	10	0.7000	358.50	19.454
1	3	10	0.7000	360.75	19.025
1	3	10	0.7000	363.00	18.607

APPENDIX D (CONT): PARTIAL LIST OF VALIDATION PATTERNS

Material Class	Type of Cut	Insert Grade (ISO P)	Feed Rate (mm/rev)	Cutting Speed (m/min)	Tool Life (min)
1	3	10	0.7000	365.25	18.202
1	3	10	0.7000	367.50	17.808
1	3	10	0.7000	369.75	17.424
1	3	10	0.7000	372.00	17.051
1	3	10	0.7000	374.25	16.689
1	3	10	0.7000	376.50	16.336
1	3	10	0.7000	378.75	15.992
1	3	10	0.7000	381.00	15.658
1	3	10	0.7000	383.25	15.333
1	3	10	0.7000	385.50	15.016
1	3	10	0.7000	387.75	14.707
1	3	10	0.8000	347.25	21.796
1	3	10	0.8000	349.50	21.300
1	3	10	0.8000	351.75	20.818
1	3	10	0.8000	354.00	20.350
1	3	10	0.8000	356.25	19.896
1	3	10	0.8000	358.50	19.454
1	3	10	0.8000	360.75	19.025
1	3	10	0.8000	363.00	18.607
1	3	10	0.8000	365.25	18.202
1	3	10	0.8000	367.50	17.808
1	3	10	0.8000	369.75	17.424
1	3	10	0.8000	372.00	17.051
1	3	10	0.8000	374.25	16.689
1	3	10	0.8000	376.50	16.336
1	3	10	0.8000	378.75	15.992
1	3	10	0.8000	381.00	15.658
1	3	10	0.8000	383.25	15.333
1	3	10	0.8000	385.50	15.016
1	3	10	0.8000	387.75	14.707

APPENDIX E: ANFIS TOOL LIFE MODEL

Fuzzy Logic Toolbox FIS File

This Appendix contains a list of the trained ANFIS Tool Life Model, as stored in ASCII format, readable by the Fuzzy Logic Toolbox of Matlab. This file ('tlm.fis') contains all the information necessary to fully define the developed ANFIS model.

```
[System]
Name='ANFIS Tool Life Model'
Type='sugeno'
NumInputs=5
NumOutputs=1
NumRules=52
AndMethod='prod'
OrMethod='probor'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='wtaver'

[Input1]
Name='Material Class'
Range=[0.95 10.05]
NumMFs=52
MF1='in1mf1':'gaussmf',[0.839611290765144 3.9882325470629]
MF2='in1mf2':'gaussmf',[0.837572288903139 4.00203023702909]
MF3='in1mf3':'gaussmf',[0.835709435728068 1.99893368956057]
MF4='in1mf4':'gaussmf',[0.833178904065055 4.00926623652907]
MF5='in1mf5':'gaussmf',[0.843029711717404 9.00647439701348]
MF6='in1mf6':'gaussmf',[0.836982693887178 2.00080393885859]
MF7='in1mf7':'gaussmf',[0.832716049903644 0.996947114430338]
MF8='in1mf8':'gaussmf',[0.837002230588852 1.9994803747844]
MF9='in1mf9':'gaussmf',[0.838759036297879 4.00208019086678]
MF10='in1mf10':'gaussmf',[0.82297257012132 4.01367509894336]
MF11='in1mf11':'gaussmf',[0.828920858224994 5.00464933064628]
MF12='in1mf12':'gaussmf',[0.838370862304256 9.001206642183]
MF13='in1mf13':'gaussmf',[0.836379677470144 3.00037497229304]
MF14='in1mf14':'gaussmf',[0.846712833622288 4.00225287707321]
MF15='in1mf15':'gaussmf',[0.836487782218365 0.99998798574326]
MF16='in1mf16':'gaussmf',[0.836567237722962 1.00005574852378]
MF17='in1mf17':'gaussmf',[0.837534597105762 8.00039929612833]
MF18='in1mf18':'gaussmf',[0.836418687625158 8.0000546320052]
MF19='in1mf19':'gaussmf',[0.862418629551945 2.03980187424775]
MF20='in1mf20':'gaussmf',[0.852136180012853 1.98874188678648]
MF21='in1mf21':'gaussmf',[0.837775065142432 4.99867473331848]
MF22='in1mf22':'gaussmf',[0.839369873169264 4.08148660856996]
MF23='in1mf23':'gaussmf',[0.836434661012348 7.99900420203887]
MF24='in1mf24':'gaussmf',[0.835134034718811 5.0006467318425]
```

MF25='in1mf25':'gaussmf',[0.836430481546806 0.999936966056145]
MF26='in1mf26':'gaussmf',[0.836912003789954 8.00006394188163]
MF27='in1mf27':'gaussmf',[0.829552792080852 2.00856015138132]
MF28='in1mf28':'gaussmf',[0.835501259909587 8.00034861928828]
MF29='in1mf29':'gaussmf',[0.834477837176952 3.00103650725865]
MF30='in1mf30':'gaussmf',[0.825625061972228 1.99711532945172]
MF31='in1mf31':'gaussmf',[0.836549260624672 3.0011205370978]
MF32='in1mf32':'gaussmf',[0.836048086414035 8.00225358401078]
MF33='in1mf33':'gaussmf',[0.839447493905707 1.00210483469015]
MF34='in1mf34':'gaussmf',[0.836483646795377 10.0000497145509]
MF35='in1mf35':'gaussmf',[0.837167870982392 1.99985867361935]
MF36='in1mf36':'gaussmf',[0.84432262004326 2.9980757679618]
MF37='in1mf37':'gaussmf',[0.837746006160487 4.99895021352469]
MF38='in1mf38':'gaussmf',[0.840254293245963 4.99576497941034]
MF39='in1mf39':'gaussmf',[0.836930258091213 1.00034299053164]
MF40='in1mf40':'gaussmf',[0.835846497111754 0.999489928859456]
MF41='in1mf41':'gaussmf',[0.836474022873929 3.89982046087518]
MF42='in1mf42':'gaussmf',[0.839822997665874 1.00315451204941]
MF43='in1mf43':'gaussmf',[0.837227700444017 3.99928524999468]
MF44='in1mf44':'gaussmf',[0.83513367478156 5.00171326404428]
MF45='in1mf45':'gaussmf',[0.837121015181546 8.99957455950808]
MF46='in1mf46':'gaussmf',[0.840964436647407 9.99627482502456]
MF47='in1mf47':'gaussmf',[0.846077949657182 1.9420520068242]
MF48='in1mf48':'gaussmf',[0.832285032905224 2.99560696784648]
MF49='in1mf49':'gaussmf',[0.842163120198199 4.99406764989049]
MF50='in1mf50':'gaussmf',[0.840653698566677 4.9986532968315]
MF51='in1mf51':'gaussmf',[0.819075515613615 1.99675875350226]
MF52='in1mf52':'gaussmf',[0.844111523848874 1.00900257663559]

[Input2]
Name='Type_of_Cut'
Range=[0.95 3.05]
NumMFs=52
MF1='in2mf1':'gaussmf',[0.511062734207259 0.999385537008225]
MF2='in2mf2':'gaussmf',[0.515180797901788 2.00018380311025]
MF3='in2mf3':'gaussmf',[0.513396475418636 1.00054727547884]
MF4='in2mf4':'gaussmf',[0.520331913526307 1.00411347782657]
MF5='in2mf5':'gaussmf',[0.515406869940631 1.98874390053113]
MF6='in2mf6':'gaussmf',[0.510846089302102 1.99946234080718]
MF7='in2mf7':'gaussmf',[0.510501184698391 0.999081036137462]
MF8='in2mf8':'gaussmf',[0.512208344427701 0.999954702061251]
MF9='in2mf9':'gaussmf',[0.511974673815564 0.999840686622766]
MF10='in2mf10':'gaussmf',[0.51972911614928 2.99612399689049]
MF11='in2mf11':'gaussmf',[0.512349481327761 1.98174732563016]
MF12='in2mf12':'gaussmf',[0.511890570303671 0.999820104480592]
MF13='in2mf13':'gaussmf',[0.512488291615271 2.99990053024313]
MF14='in2mf14':'gaussmf',[0.517061227643768 2.99762092597335]
MF15='in2mf15':'gaussmf',[0.512225414685384 0.999962270705073]
MF16='in2mf16':'gaussmf',[0.512548480324114 1.99996649420262]
MF17='in2mf17':'gaussmf',[0.512768144207152 2.00178519476636]
MF18='in2mf18':'gaussmf',[0.512403172239085 1.00004504473938]
MF19='in2mf19':'gaussmf',[0.530390640392823 2.99079738227083]
MF20='in2mf20':'gaussmf',[0.519266093888731 1.99343289588172]
MF21='in2mf21':'gaussmf',[0.513447403494712 2.9994086168333]
MF22='in2mf22':'gaussmf',[0.539506585778905 2.00484511327932]
MF23='in2mf23':'gaussmf',[0.51256758732871 2.99986271086749]
MF24='in2mf24':'gaussmf',[0.512817355204691 1.00023857768061]
MF25='in2mf25':'gaussmf',[0.512512333435495 2.99989027275031]

MF26='in2mf26':'gaussmf',[0.512078765802663 0.999918605617935]
MF27='in2mf27':'gaussmf',[0.51547682197272 2.99835745141612]
MF28='in2mf28':'gaussmf',[0.52006684349732 2.00589104644721]
MF29='in2mf29':'gaussmf',[0.51363592229251 1.00067332426842]
MF30='in2mf30':'gaussmf',[0.509131557228933 1.97470076325093]
MF31='in2mf31':'gaussmf',[0.511125766831476 2.00858219830436]
MF32='in2mf32':'gaussmf',[0.512414287160344 2.99994104296685]
MF33='in2mf33':'gaussmf',[0.512012906402969 0.99985104766992]
MF34='in2mf34':'gaussmf',[0.514790558417956 1.00136017284172]
MF35='in2mf35':'gaussmf',[0.512425154929821 2.99993291665285]
MF36='in2mf36':'gaussmf',[0.50508700855058 2.00725700339787]
MF37='in2mf37':'gaussmf',[0.516092882538878 2.99804658034113]
MF38='in2mf38':'gaussmf',[0.517862146339825 1.0028031006943]
MF39='in2mf39':'gaussmf',[0.515966056262286 2.99811297231711]
MF40='in2mf40':'gaussmf',[0.512998042270215 2.00078301293325]
MF41='in2mf41':'gaussmf',[0.526812320509704 2.02033219431462]
MF42='in2mf42':'gaussmf',[0.514510849259739 2.9988688798823]
MF43='in2mf43':'gaussmf',[0.512591008837982 2.99985045503329]
MF44='in2mf44':'gaussmf',[0.520331025494307 2.99584442101858]
MF45='in2mf45':'gaussmf',[0.512513253384293 2.99988945328562]
MF46='in2mf46':'gaussmf',[0.517777004180118 2.01300786064757]
MF47='in2mf47':'gaussmf',[0.520966975491127 2.01179796744863]
MF48='in2mf48':'gaussmf',[0.524424587172543 2.99423775373891]
MF49='in2mf49':'gaussmf',[0.506593886302192 2.01306544045647]
MF50='in2mf50':'gaussmf',[0.507716045912694 3.00234815052854]
MF51='in2mf51':'gaussmf',[0.499937336953209 3.00623809898729]
MF52='in2mf52':'gaussmf',[0.516970621663829 2.00555310986635]

[Input3]
Name='Insert_Grade'
Range=[8 40]
NumMFs=52
MF1='in3mf1':'gaussmf',[6.22276691237997 19.9995911868863]
MF2='in3mf2':'gaussmf',[6.22271157382575 10.0001025828954]
MF3='in3mf3':'gaussmf',[6.22342756708962 20.0004810793601]
MF4='in3mf4':'gaussmf',[6.22294074469671 34.9998218253147]
MF5='in3mf5':'gaussmf',[6.22280090734426 34.9998916325118]
MF6='in3mf6':'gaussmf',[6.2227043337384 10.000103899686]
MF7='in3mf7':'gaussmf',[6.22254260364002 34.9999990019608]
MF8='in3mf8':'gaussmf',[6.22267347168728 10.0000124543609]
MF9='in3mf9':'gaussmf',[6.22263037373686 10.0000487977609]
MF10='in3mf10':'gaussmf',[6.22194713089794 19.9998852305689]
MF11='in3mf11':'gaussmf',[6.22407683959132 20.0004298952831]
MF12='in3mf12':'gaussmf',[6.22257269617821 34.9999863014616]
MF13='in3mf13':'gaussmf',[6.22252968267118 9.99999251986609]
MF14='in3mf14':'gaussmf',[6.22190334956314 35.0002758237789]
MF15='in3mf15':'gaussmf',[6.22254129586046 10.0000016265755]
MF16='in3mf16':'gaussmf',[6.22256994240017 10.0000184607757]
MF17='in3mf17':'gaussmf',[6.22267085377901 20.0000652251005]
MF18='in3mf18':'gaussmf',[6.2225945398181 34.9999772511262]
MF19='in3mf19':'gaussmf',[6.22272292599302 34.9999240328679]
MF20='in3mf20':'gaussmf',[6.22568068845977 20.0000221341723]
MF21='in3mf21':'gaussmf',[6.22262292142423 10.0000440674904]
MF22='in3mf22':'gaussmf',[6.22284765385583 34.9998700314875]
MF23='in3mf23':'gaussmf',[6.22256585863306 34.9999891495609]
MF24='in3mf24':'gaussmf',[6.22279751370181 10.0001532765778]
MF25='in3mf25':'gaussmf',[6.22260520360412 10.000040730539]
MF26='in3mf26':'gaussmf',[6.22253756082414 20.0000009087481]

MF27='in3mf27':'gaussmf',[6.22328596234804 20.0010094840771]
MF28='in3mf28':'gaussmf',[6.22268436017267 34.9999399844671]
MF29='in3mf29':'gaussmf',[6.22253280386264 35.0000019947386]
MF30='in3mf30':'gaussmf',[6.22320197977444 34.9997195839694]
MF31='in3mf31':'gaussmf',[6.22296765536401 10.0002736140346]
MF32='in3mf32':'gaussmf',[6.22293495278116 20.0001491534989]
MF33='in3mf33':'gaussmf',[6.22190927216584 19.999752405026]
MF34='in3mf34':'gaussmf',[6.22255400481233 34.9999940551913]
MF35='in3mf35':'gaussmf',[6.22263677714949 10.000060372964]
MF36='in3mf36':'gaussmf',[6.22048285053625 19.9995135784656]
MF37='in3mf37':'gaussmf',[6.22230650829589 19.9999466614774]
MF38='in3mf38':'gaussmf',[6.22260636896263 34.9999729194369]
MF39='in3mf39':'gaussmf',[6.22236078799917 20.0000693238615]
MF40='in3mf40':'gaussmf',[6.22283785400182 34.9998780022978]
MF41='in3mf41':'gaussmf',[6.22569157668148 34.9986992182077]
MF42='in3mf42':'gaussmf',[6.22249100139074 35.0000201150965]
MF43='in3mf43':'gaussmf',[6.2224599176863 9.9999537845772]
MF44='in3mf44':'gaussmf',[6.22248113042806 35.0000258379165]
MF45='in3mf45':'gaussmf',[6.22257104189758 34.9999869872477]
MF46='in3mf46':'gaussmf',[6.22255212877488 34.9999948334196]
MF47='in3mf47':'gaussmf',[6.22330307045788 34.9996854687522]
MF48='in3mf48':'gaussmf',[6.22377712432812 20.0000284770248]
MF49='in3mf49':'gaussmf',[6.22117622789735 20.00002864501]
MF50='in3mf50':'gaussmf',[6.22212236189346 35.000172146639]
MF51='in3mf51':'gaussmf',[6.22338306610185 34.9996500834767]
MF52='in3mf52':'gaussmf',[6.22128535185088 20.00067389788]

[Input4]
Name='Feed_Rate'
Range=[0.04 1.1]
NumMFs=52
MF1='in4mf1':'gaussmf',[0.18248257394129 0.216278489810257]
MF2='in4mf2':'gaussmf',[0.167489786813141 0.383371910305347]
MF3='in4mf3':'gaussmf',[0.172902561528851 0.195183491032074]
MF4='in4mf4':'gaussmf',[0.151462953015122 0.224893788073255]
MF5='in4mf5':'gaussmf',[0.167918786799699 0.412224670287816]
MF6='in4mf6':'gaussmf',[0.160558801108976 0.376828134456824]
MF7='in4mf7':'gaussmf',[0.144433544400388 0.204607341176076]
MF8='in4mf8':'gaussmf',[0.192096060657611 0.205771370649799]
MF9='in4mf9':'gaussmf',[0.186865829677612 0.159248912601984]
MF10='in4mf10':'gaussmf',[0.164053834374213 0.695628522226047]
MF11='in4mf11':'gaussmf',[0.141515482635469 0.421051036923264]
MF12='in4mf12':'gaussmf',[0.174320109283156 0.193125188000766]
MF13='in4mf13':'gaussmf',[0.188995445177814 0.712460979380842]
MF14='in4mf14':'gaussmf',[0.160186467870622 0.648566436756726]
MF15='in4mf15':'gaussmf',[0.167604302836305 0.201415014106634]
MF16='in4mf16':'gaussmf',[0.168979432290299 0.386654677257424]
MF17='in4mf17':'gaussmf',[0.171571505476658 0.394074820156871]
MF18='in4mf18':'gaussmf',[0.170625210453505 0.204435949891655]
MF19='in4mf19':'gaussmf',[0.174383255282825 0.630979560663197]
MF20='in4mf20':'gaussmf',[0.194971505485469 0.388546934541834]
MF21='in4mf21':'gaussmf',[0.166368076995881 0.691932251006768]
MF22='in4mf22':'gaussmf',[0.167482328209409 0.412409688403867]
MF23='in4mf23':'gaussmf',[0.207940361965482 0.706138046867884]
MF24='in4mf24':'gaussmf',[0.133502174440215 0.230463760424683]
MF25='in4mf25':'gaussmf',[0.1713642300732 0.733115385590328]
MF26='in4mf26':'gaussmf',[0.174870761653858 0.207557670331888]
MF27='in4mf27':'gaussmf',[0.169220002533738 0.786826823029781]

MF28='in4mf28':'gaussmf',[0.175990707538798 0.378089249541077]
MF29='in4mf29':'gaussmf',[0.161918187919031 0.191448675865143]
MF30='in4mf30':'gaussmf',[0.125941496746071 0.410596268791723]
MF31='in4mf31':'gaussmf',[0.137273018631559 0.435210507710151]
MF32='in4mf32':'gaussmf',[0.151557098681649 0.753276761425567]
MF33='in4mf33':'gaussmf',[0.156717184607301 0.189376477041522]
MF34='in4mf34':'gaussmf',[0.174428030266592 0.209302021784826]
MF35='in4mf35':'gaussmf',[0.174475960258023 0.849828540935297]
MF36='in4mf36':'gaussmf',[0.122297375883703 0.40474475952321 2]
MF37='in4mf37':'gaussmf',[0.182945811559519 0.933465171119527]
MF38='in4mf38':'gaussmf',[0.157539633046581 0.128868061442596]
MF39='in4mf39':'gaussmf',[0.178510626595105 0.613725905844523]
MF40='in4mf40':'gaussmf',[0.13279408163193 0.426148456794489]
MF41='in4mf41':'gaussmf',[0.178722710592248 0.262550737962477]
MF42='in4mf42':'gaussmf',[0.180754542715465 0.811140435317355]
MF43='in4mf43':'gaussmf',[0.17149071722576 0.975327677067175]
MF44='in4mf44':'gaussmf',[0.199639888624238 0.976107029307731]
MF45='in4mf45':'gaussmf',[0.16765991907747 0.932990489846102]
MF46='in4mf46':'gaussmf',[0.204194282212528 0.395593629034963]
MF47='in4mf47':'gaussmf',[0.225100187611417 0.259118173694144]
MF48='in4mf48':'gaussmf',[0.124028197350606 0.476934576844436]
MF49='in4mf49':'gaussmf',[0.134773357700969 0.33818977172222]
MF50='in4mf50':'gaussmf',[0.153800756853645 0.525839628035747]
MF51='in4mf51':'gaussmf',[0.150265512885776 0.554084357996792]
MF52='in4mf52':'gaussmf',[0.164797083021869 0.284092870885892]

[Input5]
Name='Cutting_Speed'
Range=[40 520]
NumMFs=52
MF1='in5mf1':'gaussmf',[33.9414673457142 240.630570209739]
MF2='in5mf2':'gaussmf',[33.9410770732004 235.471635616967]
MF3='in5mf3':'gaussmf',[33.9411333987625 336.125459103468]
MF4='in5mf4':'gaussmf',[33.9410367789749 217.184156458841]
MF5='in5mf5':'gaussmf',[33.9409707708885 106.109351185253]
MF6='in5mf6':'gaussmf',[33.9411830434862 339.731560019425]
MF7='in5mf7':'gaussmf',[33.9411174188614 364.246068244851]
MF8='in5mf8':'gaussmf',[33.9411748407732 400.38739206804]
MF9='in5mf9':'gaussmf',[33.9412176596807 290.22702329304]
MF10='in5mf10':'gaussmf',[33.940477241437 162.303868911741]
MF11='in5mf11':'gaussmf',[33.940725399114 170.946589458728]
MF12='in5mf12':'gaussmf',[33.9410832247674 120.578675942381]
MF13='in5mf13':'gaussmf',[33.9411624162066 262.862756400319]
MF14='in5mf14':'gaussmf',[33.9413434536205 142.311541305217]
MF15='in5mf15':'gaussmf',[33.9411335086759 469.790795604997]
MF16='in5mf16':'gaussmf',[33.941141531722 407.753392111814]
MF17='in5mf17':'gaussmf',[33.941184977714 124.448972170391]
MF18='in5mf18':'gaussmf',[33.9411075210509 185.781093829151]
MF19='in5mf19':'gaussmf',[33.9408707136047 201.341576734995]
MF20='in5mf20':'gaussmf',[33.9407629952126 260.872692746364]
MF21='in5mf21':'gaussmf',[33.9410158288538 192.004255967325]
MF22='in5mf22':'gaussmf',[33.9417782160342 163.447770279843]
MF23='in5mf23':'gaussmf',[33.9411348310852 130.961751663974]
MF24='in5mf24':'gaussmf',[33.941143371772 230.325921724434]
MF25='in5mf25':'gaussmf',[33.9411304208692 358.678892951018]
MF26='in5mf26':'gaussmf',[33.9410930266271 145.935716222382]
MF27='in5mf27':'gaussmf',[33.9412767044599 219.986857842435]
MF28='in5mf28':'gaussmf',[33.9410419895243 149.078348989409]

MF29='in5mf29':'gaussmf',[33.9413211248142 266.117086898485]
MF30='in5mf30':'gaussmf',[33.9420738792568 234.742158521128]
MF31='in5mf31':'gaussmf',[33.9412683677878 288.623704011083]
MF32='in5mf32':'gaussmf',[33.9406350139076 115.801854643292]
MF33='in5mf33':'gaussmf',[33.9411839093658 394.137880545956]
MF34='in5mf34':'gaussmf',[33.9412248703387 77.5911827071631]
MF35='in5mf35':'gaussmf',[33.9411374179857 310.855481522926]
MF36='in5mf36':'gaussmf',[33.9411374662644 192.581216827054]
MF37='in5mf37':'gaussmf',[33.9411074217564 145.546586850157]
MF38='in5mf38':'gaussmf',[33.941001799549 172.780218956045]
MF39='in5mf39':'gaussmf',[33.9408578356968 274.057711402709]
MF40='in5mf40':'gaussmf',[33.9413244672225 302.498305842805]
MF41='in5mf41':'gaussmf',[33.9425158706037 135.64414999267]
MF42='in5mf42':'gaussmf',[33.9411380562025 263.115035290529]
MF43='in5mf43':'gaussmf',[33.9411190144497 221.550416028376]
MF44='in5mf44':'gaussmf',[33.9412931125757 115.483170497421]
MF45='in5mf45':'gaussmf',[33.9412397301029 99.8729901369938]
MF46='in5mf46':'gaussmf',[33.9412844483991 63.4146814662695]
MF47='in5mf47':'gaussmf',[33.9426258625388 199.091070705498]
MF48='in5mf48':'gaussmf',[33.9411429772135 213.697321846744]
MF49='in5mf49':'gaussmf',[33.9412537194432 144.302772368579]
MF50='in5mf50':'gaussmf',[33.9412182864445 110.366733678738]
MF51='in5mf51':'gaussmf',[33.9417637915234 156.816950701404]
MF52='in5mf52':'gaussmf',[33.9420632057302 250.665760760231]

[Output1]
Name='Tool_Life'
Range=[0 190]
NumMFs=52
MF1='out1mf1':'linear',[-26.8120108024098 125.16572144532 -7.57016905544154 34.2510909773725 -1.47590244567306 537.142539683259]
MF2='out1mf2':'linear',[4.04465315444996 -16.7832683565706 -7.22935195497158 -23.6353305126749 -0.0506044334264723 124.535938972592]
MF3='out1mf3':'linear',[-23.6729788767767 120.652737044123 -2.11087575986417 -15.5974611787023 -0.198729538751353 59.0369445683674]
MF4='out1mf4':'linear',[8.66931837802606 -33.8464403557105 63.300800055714 -18.4151335845702 -0.165851310439887 -2164.38377815168]
MF5='out1mf5':'linear',[262.78436682395 -20.1281618941483 -98.7490789325033 38.677238996226 -2.19333912547129 1327.59143772795]
MF6='out1mf6':'linear',[-8.69909279179083 -6.78279922863471 -9.20042398472361 18.3685450958438 -0.160897202086156 186.104060954335]
MF7='out1mf7':'linear',[-5.10343419033659 68.9244409616706 23.6418286303561 -17.4226544369578 -0.361088812970463 -747.19102990968]
MF8='out1mf8':'linear',[-21.4490465874135 -7.70277045508048 2.86647682794408 -35.0923791871701 -0.125918053864476 113.769240547206]
MF9='out1mf9':'linear',[-15.4263384221805 -23.5656145615906 6.61931252360515 -27.8650622584641 -0.274667565285152 120.609201574486]
MF10='out1mf10':'linear',[-8.65477182825537 -83.4219794736606 -15.1515532457058 34.1580565836815 0.570803155720422 434.364689204007]
MF11='out1mf11':'linear',[-99.5402516538056 10.804894264111 1.12227078249131 -801.843052247284 -4.12470092510899 1867.63806591598]
MF12='out1mf12':'linear',[83.6702891611288 -62.9606768552895 -39.6791079431751 -16.3940989187737 -1.00340556284613 810.629232267875]
MF13='out1mf13':'linear',[-12.9134453403524 -60.1677545819183 0.975810366138617 18.3471961982028 -0.543208042460076 366.98502806552]
MF14='out1mf14':'linear',[-7.47911891397561 -422.672026370927 -58.2478331084786 37.0635018328274 0.389496740013001 3268.70239362714]

MF15='out1mf15':'linear',[-35.4818009490012 -16.9096287453786 7.11764929510455 -27.1032317019642 -0.24846783352523 121.435351198154]
MF16='out1mf16':'linear',[-15.3342074906291 -5.46226845919954 -0.266647015183064 -15.979075947134 -0.152850572949915 116.131195067535]
MF17='out1mf17':'linear',[-819.086348470396 -15.567833692516 378.396594434101 -32.7572184192135 -0.99004777687935 -889.027517616015]
MF18='out1mf18':'linear',[21.7333753408074 -44.0128390055473 -31.672004063626 -9.44397726924365 0.427158009790074 885.40963748996]
MF19='out1mf19':'linear',[-1.53956599673775 -936.028539645717 -4.07032282997525 -215.489477831747 -0.904322764874932 3341.6027769596]
MF20='out1mf20':'linear',[-293.924912088145 -25.384459154085 -9.95235730705808 -767.642144672278 -3.33737173095925 2322.76230971171]
MF21='out1mf21':'linear',[12.6512398574582 -40.0224236902664 0.988495035148972 -6.03725426523481 -0.128006514883073 82.0937465005728]
MF22='out1mf22':'linear',[-41.3340883290551 12.5290584343355 94.3743504106806 -762.991633937348 -3.34992380034564 -1978.72031528269]
MF23='out1mf23':'linear',[-15.3866115618014 -131.884765384897 -1.05735719485334 38.0178065021685 -0.421345233687942 630.002895494348]
MF24='out1mf24':'linear',[43.1489884037966 -61.0076646786268 2.62263607002801 -37.1076620159349 -0.227441531063712 -117.097391747942]
MF25='out1mf25':'linear',[-13.00964460289 -64.761446973835 -2.23918814578104 -2.54661492387196 -0.14704502780113 302.976012077807]
MF26='out1mf26':'linear',[-257.057473869682 -86.8446203274923 122.634417098877 -5.41211225418007 -0.857892163369921 -187.952755953653]
MF27='out1mf27':'linear',[-8.76312495092025 871.477664462993 9.08150820134106 10.4156482305944 -0.0746426916741322 -2773.3565407175]
MF28='out1mf28':'linear',[65.3590782724991 13.6411393014052 -73.836905271053 -99.3113820079359 1.93418789218101 1663.51090188252]
MF29='out1mf29':'linear',[-4.06772438268184 138.133579057949 7.79290738681373 -12.6614122259695 -0.39943948958523 -271.948215128915]
MF30='out1mf30':'linear',[48.8629879598401 6.49402393292996 88.5476454398389 -98.7571175946389 -2.2628248864202 -2522.25286836778]
MF31='out1mf31':'linear',[-22.8724818534618 3.77704090654783 29.5480075802479 10.3363641661177 -0.226760019196169 -143.335818263801]
MF32='out1mf32':'linear',[-273.702586634334 -24.302748310876 -20.0095753128157 -5.70229072815814 -0.647033138185793 2760.28900673718]
MF33='out1mf33':'linear',[-26.9397914456427 5.95488886002379 10.9801075441316 -27.368752893706 -0.657874214872661 80.2749383280767]
MF34='out1mf34':'linear',[-92.9051875135942 447.495172270418 -0.488858873501532 22.8939374306904 -2.02909641828448 689.175784353187]
MF35='out1mf35':'linear',[-15.7537778556052 158.240003548708 -2.89750928009152 -10.1188855368327 -0.228184105132747 -314.360203822617]
MF36='out1mf36':'linear',[2.84994082323956 51.5432595773066 -5.85478789305346 -540.549233799723 -2.11872892344839 574.072880543233]
MF37='out1mf37':'linear',[-6.818648536745 1178.19364958851 -2.62309203307647 -17.8145055798583 0.0769529192343427 -3443.65197006092]
MF38='out1mf38':'linear',[-25.2875737538594 92.2556696551507 -1.70382264467395 -72.3520465989903 -0.584820703713338 231.355505549271]
MF39='out1mf39':'linear',[-1.38228650863343 137.808845439774 -1.83581792072464 -9.74689712150592 -0.00238117475964385 -367.489578265649]
MF40='out1mf40':'linear',[15.1355710990018 0.258260700376292 -15.4845180316565 -53.2014908688465 -0.315739308674758 655.834255473653]
MF41='out1mf41':'linear',[-122.579114494703 -83.4372213983649 -186.720100925098 -2172.08341055986 -9.05587528894754 8940.23462882704]
MF42='out1mf42':'linear',[2.51432585185038 -210.932178972627 1.86132759523841 -22.5988158016821 -0.0515116014027216 601.759431009381]
MF43='out1mf43':'linear',[-11.6913194260693 222.383664605488 -4.20027724466384 18.3079830780634 -0.30064520059 -513.414499652057]

MF44='out1mf44':'linear',[-2.37176931110114 1551.65056266005 -8.62824498234049 -27.5935091967997 -0.0384548060718405 -4303.68625191179]
MF45='out1mf45':'linear',[-15.5948769018872 31.0261489326754 -0.332050326073988 -35.9913994968415 -0.915137509626556 201.473990326665]
MF46='out1mf46':'linear',[-78.9795018826674 45.567124678327 -8.68065887193724 194.91053766844 -5.55046089737823 1210.13048105229]
MF47='out1mf47':'linear',[-108.145968585341 -69.7695995888284 -85.3783185369149 -1738.83779901314 -3.94022560203012 4661.76220040143]
MF48='out1mf48':'linear',[3.01676796830677 -780.845693960939 -1.48472735271557 -71.4508279132255 -0.0919266891561015 2434.59474944979]
MF49='out1mf49':'linear',[127.808100458108 13.4044640574799 -3.23515424371852 -1128.3188402232 -6.07483159509322 359.530784888466]
MF50='out1mf50':'linear',[-37.9457855324815 -136.559717096254 4.33041983983749 -48.9368285911301 -0.369554441860075 552.149811244502]
MF51='out1mf51':'linear',[-25.5202389579938 -32.2267352048578 -8.59189399903089 -237.191063289705 -2.24295715915809 941.045586633377]
MF52='out1mf52':'linear',[-400.016661713625 62.029698231843 37.2668908547018 -1122.78442680183 -2.6519111334468 347.800434157336]

[Rules]
1 1 1 1 1, 1 (1) : 1
2 2 2 2 2, 2 (1) : 1
3 3 3 3 3, 3 (1) : 1
4 4 4 4 4, 4 (1) : 1
5 5 5 5 5, 5 (1) : 1
6 6 6 6 6, 6 (1) : 1
7 7 7 7 7, 7 (1) : 1
8 8 8 8 8, 8 (1) : 1
9 9 9 9 9, 9 (1) : 1
10 10 10 10 10, 10 (1) : 1
11 11 11 11 11, 11 (1) : 1
12 12 12 12 12, 12 (1) : 1
13 13 13 13 13, 13 (1) : 1
14 14 14 14 14, 14 (1) : 1
15 15 15 15 15, 15 (1) : 1
16 16 16 16 16, 16 (1) : 1
17 17 17 17 17, 17 (1) : 1
18 18 18 18 18, 18 (1) : 1
19 19 19 19 19, 19 (1) : 1
20 20 20 20 20, 20 (1) : 1
21 21 21 21 21, 21 (1) : 1
22 22 22 22 22, 22 (1) : 1
23 23 23 23 23, 23 (1) : 1
24 24 24 24 24, 24 (1) : 1
25 25 25 25 25, 25 (1) : 1
26 26 26 26 26, 26 (1) : 1
27 27 27 27 27, 27 (1) : 1
28 28 28 28 28, 28 (1) : 1
29 29 29 29 29, 29 (1) : 1
30 30 30 30 30, 30 (1) : 1
31 31 31 31 31, 31 (1) : 1
32 32 32 32 32, 32 (1) : 1
33 33 33 33 33, 33 (1) : 1
34 34 34 34 34, 34 (1) : 1
35 35 35 35 35, 35 (1) : 1
36 36 36 36 36, 36 (1) : 1
37 37 37 37 37, 37 (1) : 1
38 38 38 38 38, 38 (1) : 1

39 39 39 39 39, 39 (1) : 1
40 40 40 40 40, 40 (1) : 1
41 41 41 41 41, 41 (1) : 1
42 42 42 42 42, 42 (1) : 1
43 43 43 43 43, 43 (1) : 1
44 44 44 44 44, 44 (1) : 1
45 45 45 45 45, 45 (1) : 1
46 46 46 46 46, 46 (1) : 1
47 47 47 47 47, 47 (1) : 1
48 48 48 48 48, 48 (1) : 1
49 49 49 49 49, 49 (1) : 1
50 50 50 50 50, 50 (1) : 1
51 51 51 51 51, 51 (1) : 1
52 52 52 52 52, 52 (1) : 1