Knowledge Management for Grinding Technology

Asmaa Alabed

A thesis submiutted to the University of Huddersfield in partial fulfilment of the requirements for the degree of Doctor of Philosophy

The University of Huddersfield

August 2011

Abstract

This thesis describes an investigation concerned with development of a grinding knowledge warehouse system (GKWS). Based on a study of previous work on knowledge management and technique for a selection of grinding conditions, the thesis proposes a novel methodology to deal with missing data in surface and cylindrical grinding using Genetic Programming.

The GKWS provides a guided tool for users to support the decision-making process to provide suggestions for selecting grinding conditions using rule-based reasoning (RBR) and case-based reasoning (CBR) and it can learn from new and previous grinding cases to improve and expand the CBR cases.

The GKWS developed a new methodology to deal with missing data in grinding operations. The new methodology is built on If-Then rules, mathematical equations and modelling using genetic programming (GP). Dealing with missing data improves the performance of knowledge discovery in the GKWS and the results of the CBR.

The GP is developed for modelling surface roughness in cylindrical and surface grinding. The developed GP model for surface grinding shows the ability to predict the surface roughness parameter especially when the GP terminals vary and the same material and wheel are used.

The discussion forum facilitates and supports transferring tacit knowledge into explicit knowledge where the users can exchange their ideas, send questions and answers, and pass on important links. The tacit knowledge is acquired directly from the knowledge engineers. The debate and discussion in GKWS will create new knowledge that is accessible and available when needed.

Acknowledgments

I would like to acknowledge the contribution of all those who assisted towards the completion of this thesis. First, I would like to thank my supervisor Dr Xun Chen, for his patience, support, help, encouragement and valuable guidance throughout my PhD study.

Thanks are expressed to company A for their contribution and suggestions in the first stage of this thesis.

Further thanks go to all members of computing and engineering department for their help and support.

Finally, I wish to thank my family and friends especially, my parents, my husband Saher, my friend Nadia, and my kids Hala, Omar and Jodie, for their support, encouragement and patience through my study without them this work would be impossible.

List Symbols of Acronyms

a	Depth of cut for cylindrical grinding and down feed for surface roughness
a_d	Dressing depth of cut
AI	Artificial Intelligence
b_s	Wheel width
CBR	Case-based Reasoning
CoP	Community of Practice
d_g	Grain size
d_w	Work diameter
d_{wo}	Workpiece diameter in exciting case
d_{wn}	Work piece diameter in the problem
D_{eq}	Equivalent diameter
DM	Data Mining
ENoPs	Electronic Networks of Practices
f_d	Feed rate
f_{do}	Feed rate in an exciting case
f_{dn}	Feed rate in the solution
FTDE	Free Text Data Entry
GA	Genetic Algorithm
GKWS	Grinding Knowledge Warehouse System
GP	Genetic Programming
h_{eq}	Equivalent grinding thickness
HTML	Hyper Text Markup Language
IS	Information System
k	Number of variables in the GP model

KDD	Knowledge Discovery in Database
MBR	Model Based Reasoning
n	Sample size
ODBC	Open Database Connectivity
PHP	Hypertext Pre-processor
Q'	Material removal rate
R_a	Surface roughness
$\overline{R_a}$	Surface roughness sample mean
R_{aGP}	GP value for surface roughness
R_{ao}	Workpiece surface roughness in an exciting case
R_{an}	Workpiece surface roughness in the problem
RBR	Rule-based Reasoning
s	Standard deviation
s_d	Dressing lead
s_t	Cross feed
SDE	Structured Data Entry
SQL	Structured Query Language
T_{av}	Average chip removal thickness
v_f	Feed rate
v_{f0}	Feed rate in an existing case
v_{fn}	Feed rate in the solution
v_w	Work speed
v_{wo}	Work speed in an existing case
v_{wn}	Work speed in the solution
v_s	Wheel speed
v_{so}	Wheel speed in an existing case
v_{sn}	Wheel speed in the solution
$V_w^{'}$	Specific work material removal
Z_w	Average chip removal rate

Table of Contents

List of Figures

List of Tables

CHAPTER 1 INTRODUCTION

This chapter presents the philosophy, motivation and aim behind the research undertaken, and it introduces the themes of knowledge management (KM) and grinding technology. The research methodology is identified and original contributions made by this thesis are summarised. Finally, a brief outline of the thesis structure is given.

1.1 Research Motivations and Aim

1.1.1 Overview of Knowledge Management (KM)

Many factors have contributed to the growth of knowledge management in manufacturing industry such as downsizing, outsourcing, revolution in information technology and deskilling (Stuart 1996; Davenport and Prusak 1998; O’Dell and Grayson 1998; Singh et al. 2006; Tiago et al. 2007; Kuivanen 2008; SHRM 2009; Chawla and Joshi 2010). According to these factors, manufacturing companies feel more pressure than ever to enhance the competitiveness of the manufacturing industry, maintain a well-trained workforce, and develop new innovative approaches and methods (Kuivanen 2008).

In recent years, organisations have tended to improve their efficiency, to automate their manual labour and to reduce redundancy, which has caused a decrease in informal employee communications and a reduction in tacit knowledge sharing (Sánchez and Palacios 2008). In Europe, the industry will lose a large amount of silent knowledge in a short period of time because a large number of employees will be retiring (Kuivanen 2008). The reduction in personnel will have a great impact on the operations that depend on the skills and experience of people, such as grinding technology. As people leave, organisations realise that they take valuable knowledge with them that had been kept in their minds (Connolly and Begg 1998; Hildreth, Wright and Kimble 1999; Kuivanen 2008; Sánchez and Palacios 2008).

The main objective of knowledge management is to manage knowledge processes (collect, store, retrieve and share), maintain current knowledge, and create new knowledge. Such new knowledge could be created by combining existing knowledge pieces or by a generation of novel concepts through knowledge sharing. An interesting characteristic of knowledge is that its value grows when shared (Bhirud, Rodrigues and Desai 2005; Tiago et al. 2007; Chawla and Joshi 2010). Further work observed that there are two additional key enablers that can create effective knowledge management systems – competency and infrastructure (Baveja, Shankar and Acharia 2009).

The explicit knowledge such as working procedures and databases, documents, memos, reports, best practices and /or process in organisation can be easily collected, stored, retrieved, shared and accessed at the convenience of employees and is well suited even for busy employees (Hildreth and Kimble 2002; Gillingham and Roberts 2006).

The tacit knowledge is found in people's heads or experience and it develops from direct experience of action. It could be shared through highly interactive conversion, storytelling, e-learning and sharing experiences (Wenger 1998; Hildreth, Wright and Kimble 1999; Hildreth and Kimble 2002; Bhirud, Rodrigues and Desai 2005; Coff, Coff and Eastvold 2006; Taminiau, Smit and De Lange 2009; Harris 2009).

The most challenging part of any KM programme is managing tacit knowledge (Denning 1998; Gurteen 1999; Hildreth, Wright and Kimble 1999; KPMG 2000; Kimble, Hildreth and Wright 2001; Du Plessis 2007; Taminiau, Smit and De Lange 2009). The challenge inherent with tacit knowledge is figuring out how to recognise, generate, share and manage it.

During data collection, researchers can observe missing data. Dropping all these observations and fitting a model to only the complete cases would be hugely inefficient and potentially biased (Horton and Kleinman 2007). Missing data might be missing completely at random or through an ignorable response mechanism that means the researcher can ignore the reasons for missing data. The reasons of missing data are often enormous, some due to experiment design and some due to chance. Dealing with missing data could be considered as a tacit knowledge. Since there are many different techniques available to input missing data such as maximum likelihood, regression model, weighing methods (Carpenter, Kenward and Vansteelandt 2006), and Bayesian approaches (Tang and Vemuri 2004; Ibrahim et al.

2005), in general, the researcher will investigate the advantages and disadvantages to these methods and then apply the selected method to input the missing data.

Information Technology (IT) in the form of e-mail, groupware, instant messaging, electronic database, video and audio recording, multimedia presentations, and related technologies can help facilitate the dissemination of tacit knowledge (Wiig 1997; Parlby and Taylor 2000; Santosus and Surmacz 2001; He and Li 2010; Choi, Lee and Yoo 2010). Hildreth and Kimble (2002) argued that another role of IT in KM is to make the tacit visible.

Knowledge sharing is a critical issue in any KM programme. Knowledge sharing is a key issue in order to enhance the innovation capability of firms and organisations (Sáenz, Aramburu and Rivera 2009). Taminiau, Smit and De Lange (2009) claim that the most important route to innovation is informal knowledge sharing.

Recent advances in information and communication technologies have led to the emergence of online structures where the primary purpose is more effective knowledge sharing and exchange. For example, an electronic network of practice (EnoP) enables individuals to interact and share their knowledge around a specific practice, regardless of physical time or location or prior personal familiarity (Teigland 2003; Teigland and Wasko 2004). Many organisations have used file servers, email and groupware as a collaborative tool. However, none of these tools are fundamentally designed to share knowledge (Stuart 1996; Coleman 1997; Baek et al. 1999). While intranet and groupware facilitate the creation of a tremendous amount of knowledge, it is very difficult to extract the exact knowledge efficiently from it. On the other hand, most of the time, people simply look for answers to their questions and they do not have the time to learn or/and search deeply about the subject (Baek et al. 1999; Sena and Shani 1999; Teigland 2003; Teigland and Wasko 2004).

1.1.2 Overview of Grinding Technology

Aiming at high quality parts at low costs and high-performance products in respect of part precision, surface integrity, machining efficiency and batch stability, manufacturing companies look for more accurate, reliable and efficient process planning with the support of computer aided manufacturing, computer aided process planning and flexible manufacturing systems (King and Hahn 1986; Oliveira et al. 2009).

Grinding is a material removal process using a grinding wheel, which is made up of random size, shape and orientation of abrasive grains. In other words, grinding is basically a chip-removal process in which the cutting tool is an individual abrasive grain (Kalpakjian 1991). Grinding is commonly selected for finishing operation because grinding has high accuracy and surface finish with a relatively high material removal rate. Existing techniques for the selection of grinding variables are data retrieval methods, empirical methods, and artificial intelligence (AI). Different approaches have been implemented to select grinding conditions using Case-based Reasoning (CBR), Rule-based Reasoning (RBR), and a hybrid approach (Rowe et al. 1994; Watson 1995; Rowe et al. 1988; Li et al. 1999; Li, Rowe and Mills 1999; Morgan et al. 2007; Malkin and Guo 2008).

Grinding is among the most challenging manufacturing processes and also one of the most complex modelling processes. This problem arises for different reasons that have been examined and investigated in the literature (Kegg 1983; King and Hahn 1986; Rowe, Bell and Brough 1987; Malkin 1989; Kalpakjian 1991; Tonshoff et al. 1992; Rowe et al. 1994; Vishnupad and Shin 1998; Chen 2002a; Chen 2002b; Zhou and Xi 2002; Hou and Komanduri 2003; Agarwal and Rao 2004; Brinksmeier et al. 2006; Kwak, Sim and Jeong 2006; Morgan et al. 2007; Choi and Shin 2007; Mukherjee and Ray 2008; Stepien 2009; Oliveira et al. 2009; Aurich et al. 2009; Hou, Li and Zhou 2010).

Firstly, the demand for a highly optimised and, at the same time, a more flexible grinding process results in high challenges concerning its stability (Oliveira et al. 2009). Grinding process planning and control tries to ensure high process stability by early identification of desirable grinding output. Numerous researches and approaches have been developed and investigated to improve the possibilities of process control in grinding (Midha, Zhu and Trmal 1991; Sakakura and Inasaki 1993; Gupta, Shishodia and Sekhon 2001; Choi and Shin 2007; Gallego 2007; Morgan et al. 2007; Malkin and Guo 2008). The stochastic nature of the grinding process affects the processes stability (Zhou and Xi 2002; Hou and Komandure 2003). The stochastic nature of the grinding process means that, during a material removal process, the wheel geometry varies with time when the grits participate in the process and removal of material from a workpiece**.** Because grinding is performed by a number of abrasives particles, which are randomly oriented in a grinding wheel, it is not possible to maintain the same surface finish and dimensional accuracy even though the input parameters of wheel, dressing and materials are the same.

Secondly, a large number of interactive inputs, in-process variables and thier responses need to be considered and controlled simultaneously (Michalski 2003;

Morgan et al. 2007; Mukherjee and Ray 2008; Shin, Subrahmanya and Choi 2008; Oliveira et al. 2009; Aurich et al. 2009). Some of these variables are displayed in table 1.1.

Thirdly, there is a complex relationship between the process of grinding variables and machine variables and work results (Morgan et al. 2007; Choi and Shin 2007; Shin, Subrahmanya and Choi 2008; Aurich et al. 2009; Oliveira et al. 2009; Hou, Li and Zhou 2010). Although the understanding and modelling of grinding processes is advanced, most models developed to date are seldom utilised in industrial or practical applications and the process planning is done in industry by trial and error on the machine (Morgan et al. 2007; Shin, Subrahmanya and Choi 2008; Oliveira et al. 2009; Hou, Li and Zhou 2010).

Finally, successful grinding in practice is highly dependent on the level of expertise of the machinists and engineers (Peters, Snoeys Decneut 1976; Rowe, Bell and Brough 1987; Rowe et al. 1994; Van der Spek and Spijkervet 1997; Aurich et al. 2009; Oliveira et al. 2009; Hou, Li and Zhou 2010). Such knowledge should cover all aspects in the manufacturing processes. The main industrial challenge for industry nowadays seems to be finding and keeping talented engineering staff who can deeply understand the fundamentals of grinding process planning and can apply them in practice (Chen 2002a; Morgan et al. 2007; Choi and Shin 2007; Oliveira et al. 2009). The design of grinding processes is mostly dependent on individual experiences of the process planner. Hence, an effective process planning, which is based on company-wide process knowledge, becomes more important (Oliveira et al. 2009).

Alabed and Chen (2009) proposed a methodology for knowledge warehouse systems in grinding to facilitate explicit and tacit knowledge sharing and retrieving and supporting the decision-making process for grinding process planning.

Oliveira et al. (2009) suggested integrating different methods of process analysis and a knowledge management for improving the possibilities of process control in grinding. Effective data management could reduce lead times considerably, but a generic solution is difficult since grinding is a complex process (Brinksmeier et al. 2006; Oliveira et al. 2009).

Table 1.1 A brief list of some selectable grinding processing variables

Variable	Definition
Wheel specification	Topography of a grinding wheel that includes important factors such as wheel width, wheel hardness, abrasive type, grain size, and bond type.
Wheel dressing	The dressing lead and depth are two important factors.
Material properties	Material properties include density, compounds, tensile strength, and hardness.
Grinding kinematics	Kinematics interaction between the topography of the grain of the wheel surface and the workpiece (Malkin and Guo 2008) such as depth of cut, contact length and undeformed chip thickness.

1.1.3 Research Aim

The aim of the research is to establish a reliable and flexible grinding knowledge warehouse system (GKWS) in order to facilitate a knowledge management process in grinding technology and improve the decision-making process for selecting grinding conditions, taking into account missing or incomplete data.

1.2 Research Objectives

In summary, the main objectives are:

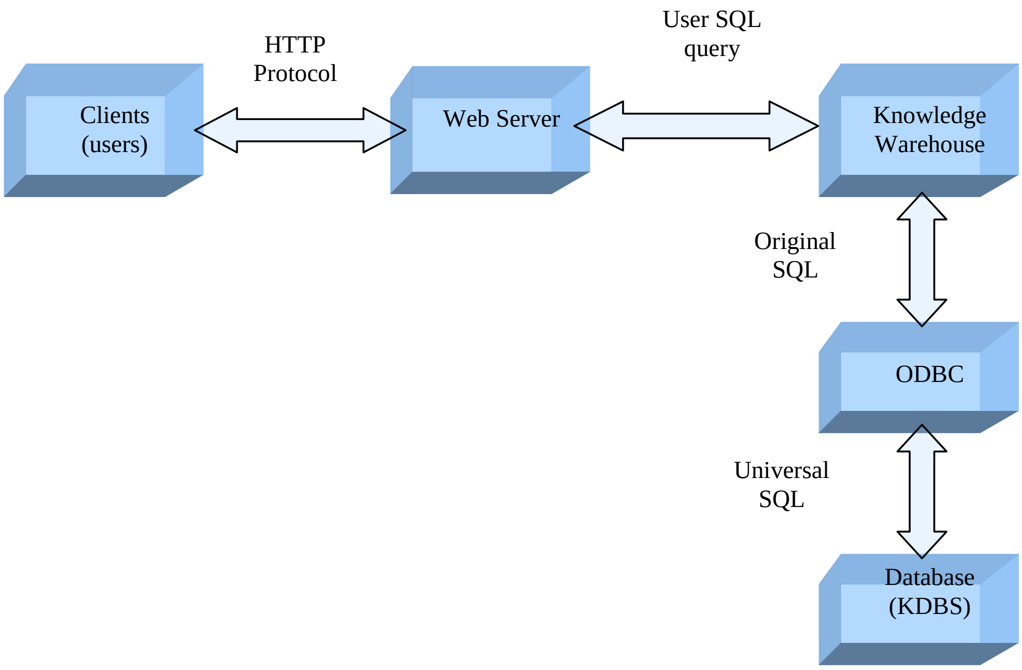
- To investigate the challenges in managing the knowledge of modelling grinding technology.
- To develop KM system that can retrieve appropriate grinding parameters efficiently and accurately.
- To develop a decision-making support method for selecting an initial grinding condition with limited available data and incomplete data.

1.3 Research Methodology

The RBR and CBR are used to provide the engineers and experts with the guidance to select the required parameters for a given grinding operation, taking into account limited grinding cases data and incomplete data. The GKWS provides users with an intelligent data acquisition form that will fill the missing key parameters using mathematical equations, empirical models, or Genetic Programming (GP) models. The core methodology of this research is using GP as the learning knowledge discovery for GKWS that takes advantage of the ability of GP to produce general solutions for modelling grinding surface roughness parameters.

The new GKWS encourages and facilitates the sharing of explicit and tacit knowledge by building a discussion forum that based on a survey of users. The categorised forum allows users to exchange their knowledge through storytelling, passing documents, or asking questions.

The grinding cases and grinding knowledge are kept in a knowledge warehouse (MySQL database). The users will be able to access and retrieve this knowledge by browsing the intranet and the GKWS is connected with a database server by using Open Database Connectivity (ODBC). The database server and web server will respond to any user query using PHP that is responsible for data manipulation. The data can be transferred to the intranet or Internet and users from different locations can access it by browsing the website constructed in a framework as shown in Figure 1.1.



1.4 Contributions

The results of this thesis contribute to the grinding technology by:

- Development of a GKWS to facilitate the knowledge management process in grinding technology and support the decision-making process for selecting grinding parameters.

- Establishment of a novel GP method in dealing with missing data. In the case study, the GP method was used to model a surface roughness parameter and the results are compared with different surface roughness models.

- The CBR is extended and adapted for selecting grinding conditions by increasing the number of saved grinding cases as the missing parameters are filled.

- The development of a categorised discussion forum to encourage knowledge sharing and retrieving of the communication between CoP that would encourage and facilitate an exchange of ideas and expertise.

- The contribution to the academic society by producing several publications.

1.5 Thesis Outline

The thesis layout displays a progression of research with the introduction and background information in Chapter 1. Chapter 2 surveys the literature of knowledge management and grinding technology and reviews the work related to the grinding knowledge warehouse system. Knowledge creation and sharing are discussed and explored. Chapter 2 explores the current related knowledge management software. Grinding modelling and its challenges are summarised. Finally, the research gap is identified.

Chapter 3 explains the steps for the development of the grinding knowledge warehouse system (GKWS), including investigation of user demand, system function analysis and general description of the GKWS. Two CoP communities are investigated and the results are presented and discussed. It introduces the methodology, main functions and capabilities of GKWS. The system function analysis is illustrated using a sequence diagram, a use case diagram and an activity diagram. General description of GKWS including the main sub-systems is introduced. The methodology of developing GKWS is discussed.

Chapter 4 presents the data presentation methodology for collecting and managing data. This stage is a very important stage for developing a knowledge warehouse. Although the number of grinding variables that are required to describe the grinding process is very large, the GKWS is recording the key parameters that are needed for selecting a grinding condition without losing any key knowledge that may have a generic application sense. The database module is designed and developed using free source MySQL. The data preparation function is established to ensure that useful knowledge is derived from the input data in knowledge discovery module.

Chapter 5 describes knowledge acquisition and problem-solving modules. The knowledge acquisition module is responsible for transforming the tacit to explicit the knowledge using a discussion forum. The problem-solving module is generated; the core of this module is Case-based Reasoning (CBR), Rule-based Reasoning (RBR), and Model-based Reasoning (MBR). Data transformation, knowledge inference engine and knowledge representation are explained and discussed in this chapter.

The learning knowledge discovery is discussed in Chapter 6. The main function of this module is to extract implicit, previously unknown and potentially useful models and patterns, and to modify and update the existing models and patterns. The methodology for dealing with missing data issue is identified. The novelty of using Genetic Programming (GP) as a technique to deal with missing data is explained. The GP model for the surface roughness variable is generated and the results are displayed and discussed.

Chapter 7 is about evaluation of GKWS that is divided into three parts: evaluation of technical GKWS, evaluation of user interaction, and measurement of GP model adequacy. The model adequacy is measured by applying GP into different datasets. Chapter 8 concludes the thesis with the thesis conclusions and future work discussion. Figure 1.2 displays the thesis structure.

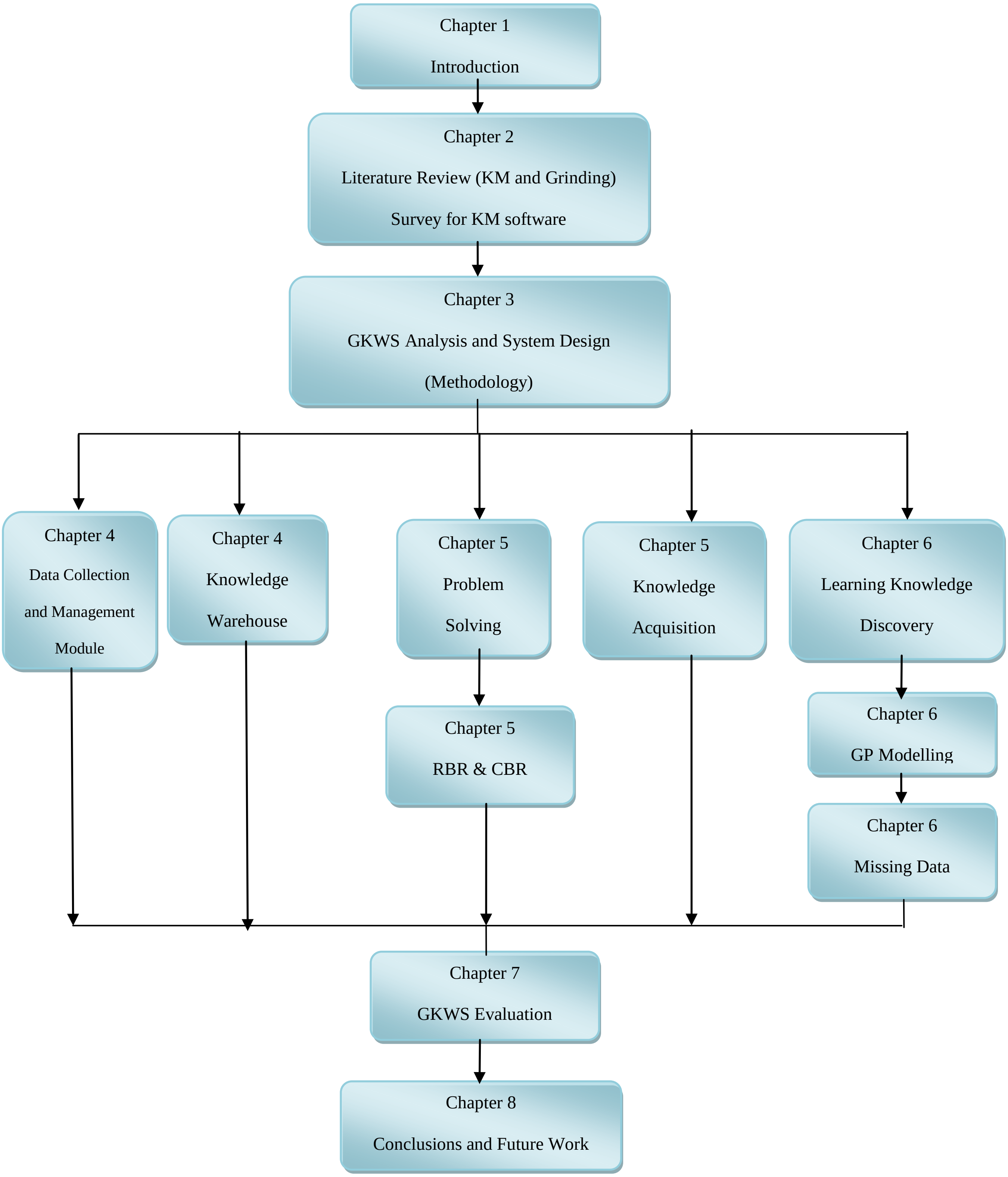


Figure 1.2 Schematic of thesis structure

CHAPTER 2 LITERATURE REVIEW

This chapter is divided into two main parts: knowledge management and grinding technology. The first part represents different definitions of knowledge and knowledge management in literature. The benefits and challenges of knowledge management are presented and knowledge creation is explained and managing explicit and tacit knowledge is discussed. Knowledge discovery in databases is briefly introduced including data and knowledge mining. The second part discusses grinding technology. The grinding process and its challenges are represented and identified in the literature. The investigation of the modelling grinding process is explained in the literature including knowledge management systems in the grinding process. Finally, the research gaps are identified.

2.1 Knowledge and Knowledge Management

2.1.1 Knowledge Definition

There are different definitions for knowledge in the literature (Nonaka 1991; Conklin 1996; Brown and Duguid 1998; Davenport and Prusak 1998; Wenger 1998; Gurteen 1999; Beckman 1999; Santosus and Surmacz 2001; Hildreth and Kimble 2002). In literature, many researchers draw distinctions between data, information, and knowledge (Davenport and Prusak 1998; Gurteen 1999; Grover and Davenport 2001), although these terms are often commonly used in day-to-day activities.

Gurteen (1999) differentiates between data, information and knowledge in a simple cake recipe example. Beckman (1997) proposed a five level hierarchy starting from data, information, knowledge, expertise and capability. Data is considered as raw material or facts or texts or numbers, while information is more structured and organised data. Knowledge is about reasoning and using data and information for an instance or a future guide. Knowledge depends on individual understanding and perspectives from information. The author believes that knowledge is a competitive resource that consists of experience, perspectives, concepts, and believes that it is indeed important for both individuals and organisations for problem-solving, decision-making, learning, creating and re-inventing what individuals know.

Conklin (1996) describes knowledge as formal and informal. He describes formal knowledge as that which is found in books, manuals and documents, and which can be easily shared in training courses. Informal knowledge is described as the knowledge that is applied in the process of creating formal knowledge. Brown and Duguid (1998) describe knowledge as know-what that can be shared by several sources and know-how. Know-how is the ability to put know-what into practice. Beckman (1999) differentiates between explicit, implicit and tacit knowledge. Regarding his definition of knowledge, tacit and implicit knowledge are found in the human mind and organisations but tacit can be accessed indirectly only with difficulty through knowledge elicitation and observation behaviour while implicit knowledge can be accessed through querying and discussion. On the other hand, explicit knowledge is found in documents and computers, which can be readily accessible, as well as documented into formal knowledge sources (Beckman 1999). Polanyi (1966) first defined tacit knowledge as knowledge that cannot be articulated or verbalised; it is a knowledge that resides in an intuitive realm.

Leonard and Sensiper (1998) describe knowledge as a continuum. Explicit and tacit knowledge are two extremes. The completely tacit knowledge that is semiconscious and unconscious is held in people's heads and bodies. On the other hand, completely explicit knowledge is codified, structured and accessible to people other than the individuals originating. Most knowledge of course exists between the extremes (Leonard and Sensiper 1998).

Regarding knowledge duality theory for Hildreth and Kimble (2002), hard knowledge is the part of what people know that can be articulated, captured, and stored and soft knowledge is the part of what people know that can't be articulated. According to the view of knowledge as duality, by implication, all knowledge is to some extent both hard and soft: it is simply the balance between the two variables (Hildreth and Kimble 2002). Wenger (1998) defines duality of knowledge that can fit into Hildreth and Kimble's (2002) definition of soft and hard knowledge. The first form (hard knowledge) is socially constructed knowledge. In anthropology, socio-psychological and sociological work knowledge tends to seen as a product of social activity. The second form of soft knowledge might be termed as internalised domain knowledge, for example, skills, experience, and expertise (Hildreth and Kimble 2002).

He and Li (2010) argue that tacit knowledge is high personalised and situated, and its cost of transfer is high. It is formed automatically by the subconscious and this forming and utilising are manifested by inspiration, skill, habit and belief, and so on (He and Li 2010).

Several views of knowledge have been explored in KM, most of them in the form of opposites, for example, formal/informal (Conklin 1996), know-what/know-how (Brown and Duguid, 1998), explicit/tacit (Nonaka 1991), and hard/soft (Hildreth and Kimble 2002). Nonaka (1991) differentiates between explicit and tacit knowledge. Tacit and explicit knowledge are not totally separated but mutually complementary entities. They can interact and exchange into each other in the creative minds of human beings. On the other hand, Leonard and Sensiper (1998) prefer to view knowledge as a continuum rather than a pair of opposites. They regard the two extremes as being tacit and explicit. They observe that most knowledge will reside somewhere between the extremes.

The author will follow Leonard and Sensiper's (1998) definition of knowledge that knowledge is a continuum and the extremes are explicit or tacit. As such, explicit knowledge is easily captured, codified, stored and shared and can be found in procedures, reports, instructions, and databases. Tacit knowledge is found in people's heads or experience and it has developed from direct experience of action, trial and error, and observation. It could be shared through highly interactive conversion, storytelling, e-learning facilities and by sharing experience (Wenger 1998; Hildreth, Wright and Kimble 1999; Hildreth and Kimble 2002, Bhiurd, 2005; Harris 2009).

2.1.2Knowledge Creation

Knowledge creation according to Nonaka and Takeuchi (1995) is a continuous and spiral conversion between explicit and tacit knowledge via four patterns of knowledge patterns of interactions, socialization, combination, internalisation and externalisation.

Nonaka and Takeuchi (1995) developed a model to describe how explicit and tacit knowledge interact and interchange into each other in the creative activities of human beings (see Figure 2.1).

Socialization involves the interaction between individuals through mechanisms such as observation, imitation or apprenticeships. Combination represents the mechanism of combining explicit knowledge through meeting and conversation or using information systems such as database, Internet and reports. Internalisation converts explicit knowledge into tacit knowledge through working groups and seminars. Externalisation converts tacit knowledge into explicit knowledge.

Socialization involves the interaction between individuals through mechanisms such as observation, imitation or apprenticeships. Combination represents the mechanism of combining explicit knowledge through meeting and conversation or using information systems such as database, Internet and reports. Internalisation converts explicit knowledge into tacit knowledge through working groups and seminars. Externalisation converts tacit knowledge into explicit knowledge as shown in table 2.1.

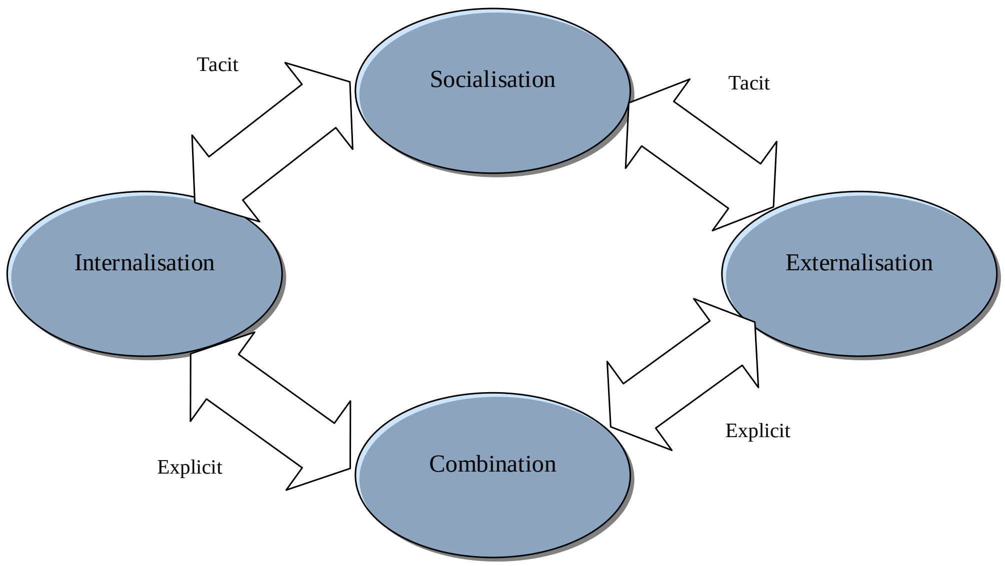


Figure 2.1 Interaction between explicit and tacit knowledge

Table 2.1 shows the activities that may facilitate and encourage knowledge interaction and exchange between explicit and tacit knowledge, taking into account Nonaka and Takeuchi's (1995) model. It can be concluded from table 2.1 that the same activity could encourage knowledge exchange tacit to tacit or tacit to explicit. For example in a brainstorming, part of the tacit knowledge could be transferred to explicit knowledge and the other part to tacit knowledge will be transferred to tacit knowledge depending on the knowledge taker's perspective from the brainstorming and their beliefs and their to share their expertise and tacit knowledge (Kreitner and Kinicki 1992; Szuianski 1996; Beckman 1997; Choi, Lee and Yoo 2010; He and Li 2010).

Table 2.1 Activities for knowledge exchange

	Tacit	Explicit
Tacit	Socialisation	Externalisation
	Brainstorming Storytelling Conversations Informal meeting E-learning facilities Community of practice	Working group Observations Seminars Informal visits Training Community of practice
Explicit	Internalisation	Combination
	Working group Observations Seminars Informal visits Training	Database Documents and reports Intranet Books Manuals and procedures

2.1.2 Knowledge Management

2.1.2.1 Knowledge Management Definition

Nowadays, Knowledge Management (KM) is widely known and practised in many large organisations; it will be useful to look back and try to give some perspective on how this old but new subject developed. Prusak (2001), one of the leading KM experts and one of a small group of practitioners who began to talk and write about knowledge management of the last few years, believed that the beginning of the knowledge management timeline was a conference held in Boston in early 1993 that several colleagues and Prusak organised—the first conference specifically devoted to knowledge management (Prusak 2001; Davenport, Prusak and Wilson 2003).

Unfortunately, there is no universal definition of KM. Davenport and Prusak (1998) define knowledge management as a fluid mix of framed experience, values, contextual information and expert insight that provides a framework for evaluating and incorporating new experiences and information. However, Petrash (1996) expresses the need for learning organisations to create, develop and share knowledge. Beckman (1997) defines knowledge management as being access to knowledge and expertise to create capabilities and superior performance, encourage innovation and enhance customer value. According to Alavi and Leidner (2001) and Chow et al. (2005), KM is the interdependent and distinctive processes or activities of knowledge creation, knowledge storage and retrieval, knowledge transfer, and knowledge application. Knowledge management activities are comprehensively discussed in the literature, which are summarised in Table 2.1.

All of these definitions agree on three important points. First, knowledge itself cannot be managed; what can be managed are knowledge processes of creation, storage, retrieval and sharing (Alavi and Leidner 2001; Chow et al. 2005; Park, Kim and Lee 2006).

Second, learning organisations must exist for knowledge management to be successful and reward the sharing of knowledge (Petrash 1996; Chinowsky 2007; Choi, Lee and Yoo 2010). Third, knowledge management is often facilitated by information technology; technology by itself is not KM (Meso and Smith 2000; Marwick 2001;

Lee and Choi 2003; Bhirud, Rodrigues and Desai 2005; Coff, Coff and Eastvold 2006; Wang, Klein and Jiang 2007). Lee and Choi (2003) confirmed the positive impact of IT on knowledge management. In the manufacturing industry, Wang, Klein and Jiang (2007) argue that information technology (IT) support for KM can directly enhance an organisation's knowledge-based activities.

The author believes that knowledge management is the creation, acquisition, analysis, storage, sharing, and manipulation of knowledge and expertise in a learning environment that encourages and rewards knowledge sharing where IT could support and facilitate these activities.

Table 2.2 Summary of KM Activities

Author and Publication	Knowledge Management Activities
Wiig (1993)	Creation Compilation Dissemination Application
O'Dell and Grayson (1998)	Identify Collect Adapt Organise Apply Share Create
Beckman (1997)	Identify Capture Select Store Share Apply Create Sell
Ruggles (1997)	Generation Codification Transfer
Meso and Smith (2000) Marwick (2001)	Use Search Creation Packaging
Alavi and Leidner (2001)	Creation Storage and retrieval Transfer Application

Chow et al. (2005)	Creation Storage Distribution Application
Park (2006)	Acquisition Organisation Utilisation Disposition Sharing
SHRM (2009)	Creating Acquiring Sharing Managing

2.1.2.2 Benefits from Knowledge Management

The benefits of investing in KM are deeply discussed and documented in the literature (Carrillo and Gaimon 2000; Gold, Malhotra and Segars 2001; Lee and Choi 2003; Jones 2003; Ferrada and Serpell 2009; Wang, Klein and Jiang 2007; Tiago et al. 2007; Chawla and Joshi 2010; Ovaska et al. 2010). To get the most value from a company's knowledge assets, knowledge management practitioners maintain that knowledge must be shared and served as the foundation for collaboration. The main benefits can be summarised as follows:

- improve competitive advantage and create new knowledge (KPMG 2000; Tiago et al. 2007; Chawla and Joshi 2010).
- improve customer service (KPMG 2000; Tiago et al. 2007).
- improve revenue growth and profit growth (KPMG 2000; Carrillo and Gaimon 2000).
- improve ability to innovate, the ability to anticipate surprises, improved coordination of efforts, rapid commercialisation of new products,

responsiveness to market changes, and reduced redundancy of information/knowledge (KPMG 2000; Gold, Malhotra and Segars 2001; Taminiau, Smit and De Lange 2009).

- foster effective tools for increasing productivity and performance in the manufacturing industry (Carrillo and Gaimon 2000; Gold, Malhotra and Segars 2001; Da Silva Garza et al. 2007; Ferrada and Serpell 2009).

2.1.2.3 Challenges of Knowledge Management

Culture, trust, creating a learning environment, IT support, and managing tacit knowledge are the main challenges facing the implementation of effective KM programmes.

- Culture and trust

  Culture and getting employees on board are some of the important challenges for any KM programme. The major problems that occur in KM programmes usually result because companies ignore or underestimate people and their cultural issues. To create a knowledge sharing culture, an organisation needs to encourage people to work together more effectively, to collaborate and to share - ultimately to make organisational knowledge more productive (Davenport and Prusak 1998; Gurteen 1999; Glasser 1999; Davenport, Prusak and Wilson 2003; Du Plessis 2007; Al-Alawi, Yousif and Fradoon 2007; Tiago et al. 2007). The old paradigm was “knowledge is power”; nowadays, it needs to be exchanged to “sharing knowledge is power” (Gravin 2003). Effective knowledge sharing and learning requires cultural change within organisation’s new management practices, senior management commitment and technology support (Havens and Knapp 1999; O’Dell and Grayson 1999; Chau 2005; Tiago et al. 2007). Trust is another critical challenge to any KM programme. Trust can be defined as maintaining reciprocal faith in each other in terms of intentioned behaviours (Kreitner and Kinicki 1992). According to Szuianski’s study (1996), the lack of trust among employees is one of the key barriers against knowledge exchange. Trust may facilitate and encourage knowledge exchange and creation. Lee and Choi (2003) confirmed the impact of trust on knowledge creation. Trust, relationships and dialogue are the foundation for building organisational knowledge sharing (SHRM 2009).

- Creating a learning environment

  In an environment where an individual’s knowledge is valued and rewarded, establishing a culture that recognise tacit knowledge and encourages employees to share it is critical. Organisations should motivate and reward employees to get the best from any KM programme (Petrash 1996; Gurteen 1999; Davenport, Prusak and Wilson 2003; Chinowsky 2007; Harris 2009; Choi, Lee, and Yoo 2010). Davenport, Prusak and Wilson (2003) argued that incentives and rewards create and support positive behaviours required for KM. Chinowsky (2007) addressed the challenge on how to transform an organisation from a focus on knowledge management to a focus on developing a learning culture.

- IT

While information technology could support and facilitate KM as discussed in the literature (Sena and Shani 1999; Hargadon and Sutton 2000; Meso and Smith 2000; Marwick 2001; Bhirud, Rodrigues and Desai 2005; Coff, Coff and Eastvold 2006; Du Plessis 2007; Wang, Klein and Jiang 2007; Choi, Lee, and Yoo 2010), KM is not a technology-based concept and IT is not the starting point of any KM programme.

- Managing tacit knowledge

The difficulty of managing tacit knowledge is one of the most important challenges for KM (Hildreth, Wright and Kimble 1999; KPMG 2000; Santosus and Surmacz 2001; Kimble, Hildreth and Wright 2001; Du Plessis 2007; Harris 2009; He and Li 2010). While IT in the form of databases, e-mail, groupware, instant messaging, electronic databases, video and audio recordings, multimedia presentations and related technologies can help facilitate the dissemination of tacit knowledge, identifying tacit knowledge in the first place is a major hurdle for most organisations (Gurteen 1999).

Hargadan and Sutton (2000) argued that KM makes tacit and explicit knowledge management possible, as both types of knowledge add value to the organisation. On the other hand, Coff, Coff and Eastvold (2006) developed a theory about how information technology can be applied to leverage tacit knowledge without transferring or codifying the knowledge applied into manufacturing companies and hospitals. In both cases, the technology allowed experts to work remotely and leverage their skills and knowledge across multiple locations.

Wong and Radcliffe (2000) argued that explicit knowledge requires an understanding process in order to collaborate with the tacit domain to perform needed tasks. He and Li (2010) claimed that tacit knowledge explicating activity is a distributed cognitive activity, whose success depends on interaction the following factors: individuals, artefacts, environment and sharing culture. These factors depend on each other and affect each other so none of the factors should be neglected. The challenge inherent with tacit knowledge is figuring out how to recognise, generate, and share it whereby it could then be easily managed.

2.1.2.4 Managing Explicit and Tacit Knowledge

As discussed earlier, the KM process is a continuous process that starts with knowledge creating and gathering, organising and refining, storing and retrieval and utilising (Hildreth, Wright and Kimble 1999; Alavi and Leidner 2001; Kimble, Hildreth and Wright 2001; Bhirud, Rodrigues and Desai 2005; Chow et al. 2005; Gillingham and Roberts 2006; SHRM 2009), as shown in Figure 2.3.

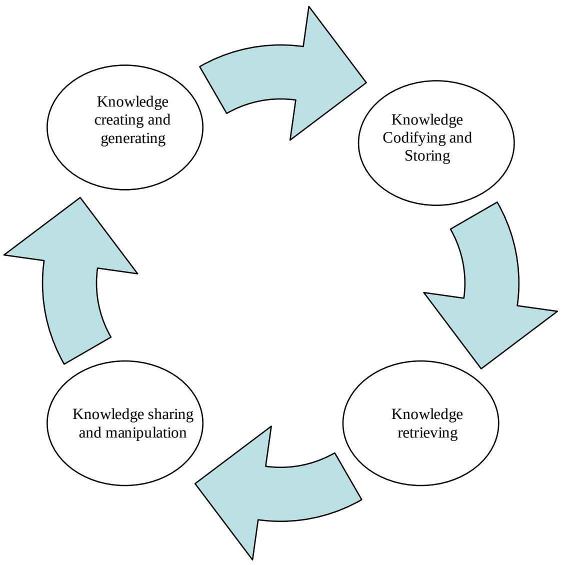


Figure 2.3 KM activities

The first step is knowledge acquired and captured from internal and external sources. The external sources are customers and suppliers' records, researching, net browsing and reading books. The internal resources are employee knowledge (tacit knowledge), database, procedures, reports, documents etc.

The second step is knowledge storage. Organisational memory is found in employees, organisational culture, processes and procedures, the physical workplace, and archives. Knowledge needs to be organised and codified so it will be easy to be retrieved and accessed. Technology plays an important role in retrieving knowledge; organisations must make sure that there is an efficient way of managing the content and making such that search engines and intranet browsers can retrieve the knowledge in an effective way.

Knowledge sharing is highly critical, which will be discussed deeply in the next section. Sharing knowledge has been facilitated by Internet technology, email, groupware etc.

Explicit knowledge can be written down, processed by information systems, codified or recorded, archived and protected by organisations (Hildreth, Wright and Kimble 1999; Kimble, Hildreth and Wright 2001; Gillingham and Roberts 2006). Many researchers argued that explicit knowledge can fit into this cycle but there is some knowledge that cannot be captured, codified and stored (Hildreth and Kimble 2002; Gillingham and Roberts 2006).

The management of explicit knowledge is well established but the sharing of tacit knowledge possesses greater problems (Hildreth, Wright and Kimble 1999; Kimble, Hildreth and Wright 2001; Santosus and Surmacz 2001; Gillingham and Roberts 2006; Harris 2009; He and Li 2010). KM is concerned with making knowledge visible and therefore developing knowledge processes. Knowledge creation is a continuous and dynamic interaction between explicit and tacit knowledge. Both types of knowledge should be found in any organisation. The most crucial part is how to recognise and transfer tacit knowledge to explicit knowledge (Nonaka and Takeuchi, 1995; Hildreth, Wright and Kimble 1999; Kimble, Hildreth and Wright 2001; Prusak 2001; Teigland 2003; Gillingham and Roberts 2006).

According to Hildreth, Wright and Kimble (1999), the first step to manage tacit knowledge is to understand the social processes that govern its constructions and its sustenance in an organisation. Community of Practice (CoP) has been identified as being a group where tacit knowledge is natured and sustained (Hildreth, Wright and Kimble 1999; Kimble, Hildreth and Wright 2001; Wenger 1998).

From Hildreth and Kimble's (2002) case studies, three methods can be discussed for tacit knowledge construction in CoP. Firstly, there is the gathering of domain knowledge (for example how to solve a particularly tricky problem). Secondly, the construction of knowledge of work practices specific to the community (for example knowledge of idiosyncrasies of an individual machine and how they are created). Finally, there is the knowledge that there are community constructs around the competencies of its members (for example through the appraisal of their war stories). These three methods could be regarded as being the tacit equivalent of the capture-codify-store approach of explicit knowledge management **(**Hildreth, Wright and Kimble 1999)**.**

Tacit knowledge could be shared through innovation days, organisation's internal conferences, storytelling, technology shows, best practice, sharing of artefacts via telephone conferencing and net meeting and internal training **(**Hildreth, Wright and Kimble 1999; Kimble, Hildreth and Wright 2001**).** Harris (2009) endorses the need for a focused approach to e-learning that facilitates social interaction and learning in order to harness the value of shared tacit knowledge.

He and Li (2010) discussed that tacit knowledge explicating activity is a distributed cognitive activity, whose success depends on interaction the following factors: individuals, artefacts, environment and sharing culture. These factors depend on each other and affect each other so none of the factors should be neglected. Further studies are needed to explore how to design these factors in the system so that the explication of tacit knowledge can be accomplished successfully (He and Li 2010).

The author describes how explicit and tacit knowledge resides in each KM activity in figure 2.4. For knowledge creating, explicit knowledge is created in documents reports, procedures, database entry, emails, and manuals while tacit knowledge is created by direct interactions, observations, brainstorming, informal meetings/visits, phone calls, thinking in people's heads, and expertise. The storage for explicit knowledge is in documents, reports, procedures, manuals, database, data or knowledge warehouse. On the other hand, tacit knowledge is stored in people's heads and perception but it could be transferred to explicit knowledge through groupware, CoP, informal meetings, video recordings, and knowledge/databases. Knowledge sharing and retrieving for explicit and tacit knowledge is accomplished by using reports, documents, Intranet, database, groupware, and knowledge warehouse.

Brainstorming, innovation days, storytelling, best practice and informal meetings could facilitate and encourage transferring tacit to explicit knowledge. The encouragement and rewarding system should be there all the time to promote knowledge sharing.

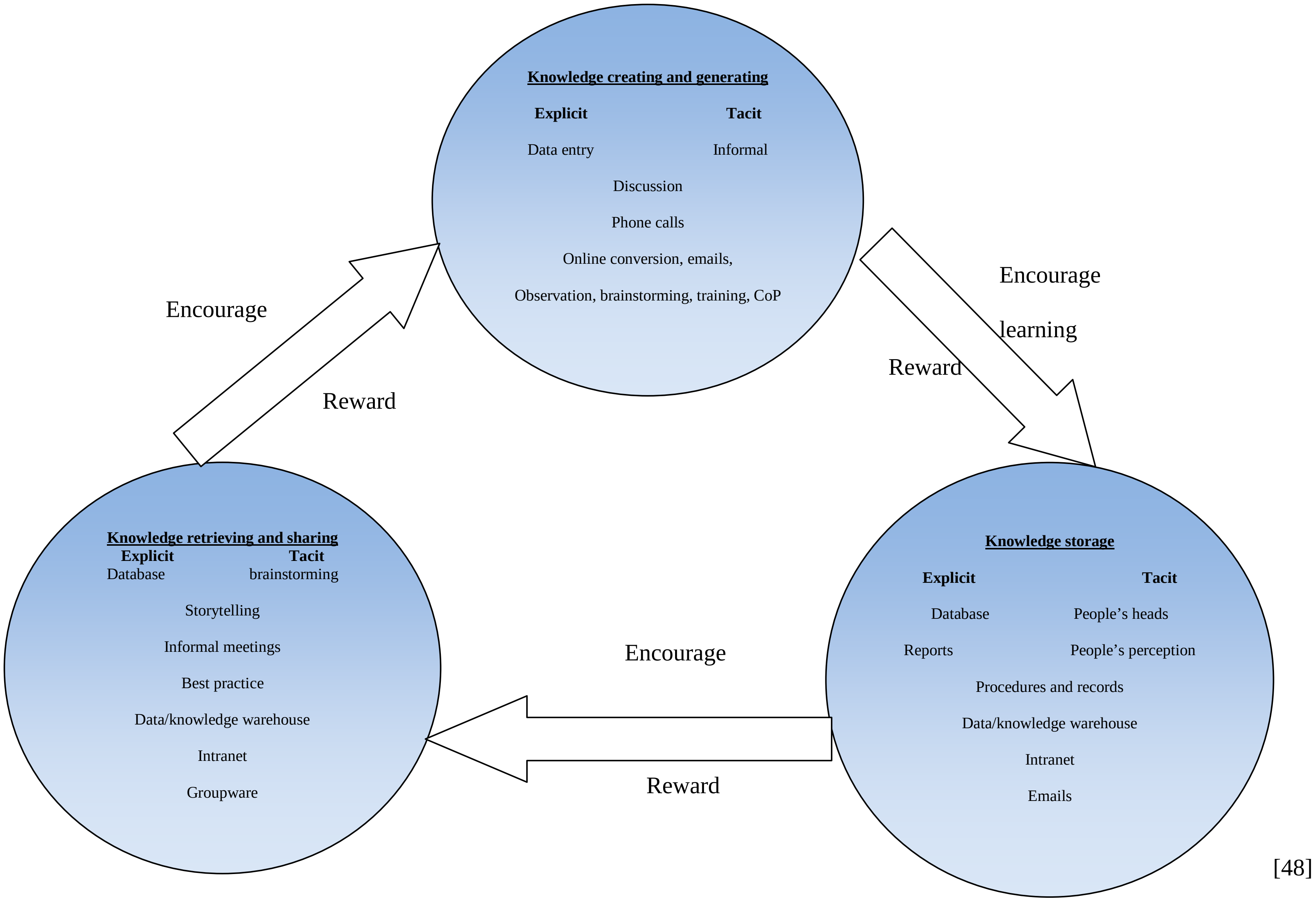


Figure 2.4 How explicit and tacit knowledge are found in KM

2.1.2.5 Knowledge Sharing

Knowledge sharing is a critical issue in any KM programme. Workers tend to form networks of expertise spontaneously: to facilitate individual learning and collaboration and to discuss work related problems together. Sometimes these networks transform into a Community of Practice (CoP) (Leave and Wenger 1991) and more recently virtual communities as Electronic Networks of Practices (ENoP) (Teigland and Wasko 2004).

In a CoP, employees who share a common interest for the field they work in come together on a regular basis to help each other, solve problems and to share and create knowledge collaboratively (Wenger 1998). Knowledge creating and sharing are two of the core activities of CoP. The CoPs provide the social structure in organisations for an interactive approach to KM.

Leave and Wenger (1991) first introduced the concept of CoP in 1991. They saw the acquisition of knowledge as a social process where people can participate in communal learning at different levels depending on their level of authority in the group.

Leave and Wenger (1991) originally described a CoP as “a set of relations among persons, activity and world, over time in relation with other tangential and overlapping CoPs”. In these communities, newcomers learn from older people by being allowed to participate in certain tasks relating to the practice of the community. Later on, Wenger, McDermott and Snyder (2002) defined CoP as a group of people who share a concern, a set of problems, or a passion about topic and who deepen their knowledge and expertise in this area by interacting on an ongoing .

CoP creates value by improving the performance of their members when they apply their knowledge in the performance of their job. Because practitioners belong at once to their CoP and their work teams, they are the direct “carriers” of knowledge. Involving practitioners in KM is also important for returning knowledge from the field (Wenger 2004). CoPs have been identified as effective tools for the variation and sharing of knowledge **(**Hildreth, Wright and Kimble 1999; Sharratt and Usoro, 2003; Wenger 2004).

Brown and Duguid (1998) developed the concept of networks of practice. This concept refers to the overall set of various types of informal, emergent social networks that facilitate learning and knowledge sharing between individuals conducting practice-related tasks.

Thus, in an effort to replicate traditional CoPs electronically, management in numerous organisations have invested in computer-mediated communication technologies to facilitate knowledge sharing regardless of time and space constraints. These emergences are defined as virtual communities of electronic networks of practices (ENoP) (Teigland and Wasko 2004).

Wenger (1998), Hildreth (2000), Kimble, Li and Barlow (2000), Santosus and Surmacz (2001), Sharratt and Usoro (2003) and Bhirud, Rodrigues and Desai (2005) highlighted the important use of shared artefacts between CoP or EnoP. Although the shared artefact does not solve the problem of tacit knowledge sharing in a distributed international environment, it can be of real benefit and can play a variety of useful roles to support the sharing of tacit knowledge.

Yang and Wei (2010) proposed a model that can numerically measure the performance of communities of practice and quantify the knowledge level of the knowledge workers. The model takes into account the tutor's willingness to share knowledge, the learner's motivation to learn knowledge, the learning by teaching effect, the learning by discussion effect, the effect of the reward system, and the ability of teaching and understanding.

Sáenz, Aramburu and Rivera (2009) claimed that knowledge sharing is a key issue in order to enhance the innovation capability of firms. Another research stressed the importance of informal knowledge sharing to enhance innovation in organisations (Taminiau, Smit and De Lange 2009).

2.1.2.6 Knowledge Sharing Technology

The impact of IT in knowledge management has been documented in the literature (Allee 1997; Kimble, Hildreth and Wright 2001; Santosus and Surmacz 2001; Teigland 2003; Teigland and Wasko 2004; Wang, Klein and Jiang 2007; Choi, Lee and Yoo 2010). Organisations are evolving from network applications to collaborative applications and online communication (Coleman 1997; Coleman 1999). By using computer and communications networks, people from different geographical areas can communicate and share their common target and efforts across time and space. After network applications, many companies moved forward to use groupware to share their knowledge. Groupware is an umbrella term describing technologies that support person-to-person collaboration. Groupware includes e-mail, electronic meetings, desktop video conferencing as well as systems for workflow and business process reengineering (Coleman 1997). Groupware is collaborative technology that provides tools to solve “collaboration oriented” problems. Much groupware software is available in the market such as Lotus Notes, MS Exchange, Tetranet Software etc.

Groupware can be used to send messages, files, data or documents between people and to facilitate the knowledge sharing that includes chatting, telephone, and video conferencing and management of group activities such as electronic calendars, project management, workflow etc (Allee 1997; Kimble, Hildreth and Wright 2001; Santosus and Surmacz 2001).

Many organisations are migrating from e-mail to electronic collaborative culture that will cause a huge increase of the generation of knowledge. This increase of knowledge generation will require corporate culture to evolve to the next step of managing this knowledge (Coleman 1999).

Many companies have used file servers, email and groupware as a collaborative tool (Baek et al. 1999). However, none of these tools are fundamentally designed for knowledge sharing. File servers are designed to provide teams with shared access to files and applications. They can provide a fast and reliable method to share files between a team, but not to find the knowledge in that file. E-mail is designed to send messages to one or more people. It is not intended to enable group communications. Groupware allows users to share knowledge and have fast and reliable access to files and applications, but it requires them to have the same groupware system in order to share knowledge (Coleman 1999).

While intranet and groupware facilitate the creation of a tremendous amount of knowledge, it is very difficult to extract the exact knowledge efficiently from it (Baek et al. 1999). By using the most advanced research technology, finding the right knowledge is still very difficult. This is because, in these search engines, they assume that the user has used the right keywords, which is not always the case. And even if the user used the right keywords, that word might have more than one meaning in a different context (Baek et al. 1999). Providing access to any data alone assumes that the users can read, understand and utilise the knowledge from that database or knowledge repository. These assumptions are not frequently valid because of the complexity of the decision-making, as no one can be trained in all areas, and there are time limitations. Most of the time, people need simply answers for their questions and they don't have the time to learn deeply about the subject.

In the manufacturing industry, Wang, Klein and Jiang (2007) argued that IT support for KM can directly enhance an organisation's knowledge-based activities and, thus, a firm's performance. Choi, Lee and Yoo (2010) found that IT has a positive impact on knowledge sharing. Also, it showed that sharing knowledge is not enough, since organisations must ensure that shared knowledge is applied and utilised in order to improve CoP performance.

2.1.3 Review of Current KM Software

Many tools and software are designed and developed to support the knowledge management process. There are compliance programmes to facilitate the knowledge-sharing process. Software has also become the means for capturing knowledge and ease of collaboration. This software tears down the figurative walls that confine knowledge within small groups and make it more widely available for large audiences, thus giving substantiation of an organisation's knowledge and also giving it some sense of longevity. A quick search on the availability of KM software on the

Internet has revealed many KM suites available online where some of the most related software suites are explained. Searching the Internet for current knowledge warehouse software and solutions using different combinations of key words (such as knowledge warehouse software, knowledge base warehouse software, collaborative tool, knowledge sharing) revealed that different packages of knowledge warehouse software are available from a number of vendors, including NOVA, ISYS, and SAP. The first search was carried out in November 2004; the related KM suites were compared in Table 2.2. Another search was carried out in September 2008 in order to investigate the most recent KM warehouse and collaborative suites. An overview of their features and comparison is shown in Table 2.3.

Different knowledge management software are designed and developed to optimise knowledge sharing and to take advantage of IT. The comparison is based on the KM activities: acquire knowledge, store and organise, retrieve and share. From the comparison between Table 2.2 and 2.3 for different KM suits, all of the software is designed to capture, store, retrieve and share structured (such as database and articles) and unstructured (such as online discussions and emails) data. The search engine for all of the suites can use natural language search and/or more advanced search such as Boolean and fielded search (Response, NOVA, Talisma, SAP, and Interspire) or by combining advanced full-text and linguistic techniques in Dieselpoint suite. Intelligent methods are used to discover sets of related documents such as fingerprints in Colleix suite and state-of-the-art algorithm in SAP.

Claromentis, NOVA, ISYs, SAP and Dieselpoint are focused on the search and retrieve processes of the knowledge while Magic Knowledge, Talisma, IntelliEnterprise, Interspire and Attivio’s Active Intelligence suites are more focused on search and documentation management. That means knowledge management processes (gathering, storing, retrieving and sharing) are all covered. It can be noticed that the search engine is becoming more and more advanced in recently used knowledge discovery and self-learning methods.

The search for collaborative tools revealed many different available suites; a sample is taken to investigate its features such as ICohere, CommunityZero, and Tomoye Ecco. These suites are focused on communications and collaboration between workers via a web browser or regular email or mobile devices, using ranking for answers and questions. The cost of implementing and maintaining any of these suites could be considered an important issue for any organisation.

Table 2.3 Comparison between KM suites (November 2004)

	Software Name				
Feature	**Claromentis**	**Magic Solution**	**Response**	**Knowledgebase.net**	**Knowledgebase NOVA**
Company Name	**HIS**	**Remedy**	**ComponentOne**	**KnowledgeBase**	**NOVA**
Acquiring Knowledge					
Structured	√	√	√	√	√
Unstructured	√	√	√	√	√
Store and Organise					
Support Database	√	√	√	**X**	√
Knowledge Warehouse	√	√	√	√	√
Retrieve (Search Engine)					
Natural Language	√	√	√	√	√
Advance Search	√	√	√	√	√
Self-learning	√	**X**	√	√	**X**
Share and dismiss					
Web-based	√	√	√	√	√

Table 2.4 Summary of KM suites (September 2008)

	ISYS 8	SAP	Talisma	Interspire	DieselPoint	IntelliEnterprise	Attivio	Collexis
Acquiring Knowledge								
Structured	√	√	√	√	√	√	√	√
Unstructured	√	√	√	√	√	√	√	√
Intranet	√	X	X	X	X	X	X	X
External resources	√	X	X	X	X	X	√	X
Store and Organise								
Database	√	√	√	√	√	√	√	√
Knowledge Warehouse	X	√	√	√	X	√	√	X
Retrieve								
Search Engine								
Natural Language	√	√	√	√	√	√	√	√
Advanced	√	√	√	√	√	√	√	√
Boolean	√	√	X	X	√	X	X	X
Fielded search	√	X	X	X	X	X	X	X
Proximity	X	√	X	X	X	X	X	X
Machine Learning Filtering	X	X	√	√	X	√	X	X
Self-learning	X	√	√	X	X	X	X	X

[57]

Knowledge Discovery	X	X	X	X	√	X	X	X
Share and dismissing								
Web	√	√	√	√	√	√	√	√
Intranet	√	X	X	√	X	√	X	X
Encourage Collaboration	X	X	√	X	X	√	X	X
Extra features		Automatic classifications and text mining	Custom fields for search engine feedback	Suggestion system	Linguistic techniques Suggestion engine Search by attributes			Fingerprints Suitable for education and search organisation
Most Powerful feature	Search engine	Search engine Text mining classifications	Documentation and sharing knowledge	Publishing, sharing, and collaboration	Search and navigation		Manage and update structure and unstructured data	Search for biomedical
Price	From $1,000							+ $24,000
Customers	Ford motors, Boeing, Toyota Australia			Vodafone, Xerox, Virgin				Mayo Clinic, Johns Hopkins University, the University of California, San Francisco, the University of South Carolina.

2.2 Knowledge Discovery in Databases

Knowledgediscovery in databases (KDD) is the process of extracting models and patterns from large databases. The term Data Mining (DM) is often used as a synonym for the KDD process although strictly speaking it is just a step within KDD (Sarker, Abbass and Newton 2002). In other words, DM is the application of specific algorithms to extract patterns and models from the data. The basic problem addressed by the KDD process is one of mapping low-level data into other forms that might be more compact, more abstract, or more useful (Fayyad, Piatetsky-Shapiro and Smyth 1996a).

KDD steps have been discussed in the literature. Fayyad, Piatetsky-Shapiro and Smyth (1996b) suggested nine steps: application domain understanding, creating a target data set, data cleaning and processing, data reduction and projection, choosing the mining task, choosing the mining algorithm, mining the data, interpreting the mined patterns, and consolidating the discovered knowledge.

Sarker, Abbass and Newton (2002) proposed thirteen steps: problem definition and determining the mining task, data description and selection, data conversion, data cleaning, data transformation, data reduction and projection, domain-specific data pre-processing, feature selection, choosing the mining algorithm, algorithm-specific data pre-processing, applying the mining algorithm, analysing and refining the results and knowledge consolidation (Sarker, Abbass and Newton 2002).

Bendoly (2003) summarised the steps into three phases: domain identification, strategy development application and results evaluation. It can be concluded from previous work on KDD that KDD focused on step of DM in all previous researches. The literatures studies on the KDD steps reveal that most of the studies have the same structure of the KDD process. The KDD is adopted in this research and its six steps can be summarised as follows: identify the domain, data selection, data cleaning, data transformation, data mining, and results evaluation.

2.2.1 Data and Knowledge Mining

Simoudis (1996), among others, discussed definitions of DM in the literature review. He related that DM is the process of extracting valid, previously unknown, comprehensible, and actionable information from a large database and using it to make crucial business decisions.

A different view is proposed by Fayyad, Piatetsky-Shapiro and Smyth (1966a), who stated that DM is a step in the KDD process that consists of applying data analysis and discovery algorithms that produce a particular enumeration of patterns (or models) over the data.

In DM, there are three primary components: model representation, model evaluation and search. Model representation is the language used to describe discoverable patterns. It is crucial that a data analyst fully understands the representational assumptions that may be inherent in a particular method.

Model evaluation is quantitative measurement on how the proposed models meet the goals of prediction and description.

The search method includes two components: parameter search and model search. In parameter search, the algorithm searches for the set of parameters that optimise the model evaluation criteria, given the observed data where the model is fixed. Model search acts as a loop over the parameters where the model representation is changing so that the family of models is considered.

The basic functions of the DM process consist of feature selection, summarisation, association, clustering, prediction, and classification.

- Feature selection is concerned with the identification of a subset of features that significantly contribute to the discrimination or prediction problem (Bradley, Mangasarian and Street 1998).

- Summarisation involves methods for finding a compact description for a subset of data. This is useful for understanding the importance of certain attributes when compared against each other (Hui and Jha 2000).

- Dependency modelling determines how to find a model that is related the various attributes.

- Clustering identifies a finite set of categories or clusters to describe the data (Jain and Dubes 1988). The categories may be mutually exclusive and exhaustive, or consist of a richer representation such as hierarchical or overlapping categories (Fayyad et al. 1996).

- Regression modelling is a function that maps a data value to a real-value.

- Classification is designed to predict the most likely state of a categorical variable given the values of the other variables.

Different listed techniques are available for data mining; this list should not be considered complete, but rather a sample of the techniques for data mining.

- **Genetic Programming (GP)** is considered as induction for classification and generalised rules (Goldberg 1989; Koza 1992). Freitas (1997) proposed a GP framework for two data mining tasks; classification and generalised rule induction based on relational algebraic operations, expressed by an SQL query. Wong (2001) presented a flexible knowledge discovery system called generic genetic programming, which combines genetic programming and inductive logic programming to induce production rules from knowledge represented in various knowledge formats. Elkaffas and Toony (2006) developed a framework for using genetic programming to induce classification rules from a database.

- **Artificial Neural Networks (ANNs)** are one of the most commonly used for DM. As a rule of thumb, ANNs are more accurate than many DM techniques and the choice of decision of the appropriate DM tool is usually a cost benefit analysis when it comes to real life applications.

- **Optimisation methods** provide another alternative set of techniques that produce robust results. A major problem with these techniques is scalability and slow convergence. Global optimisation can be combined with heuristics to overcome the slow performance of optimisation techniques (Bagirov, Rubinov and Yearwood, 2001).

Michalski (2003) defines knowledge mining as concerned with developing and integrating a wide range of data analysis methods that are able to derive directly or incrementally new knowledge from large (or small) volumes of data using relevant prior knowledge. Knowledge mining process of deriving new knowledge can be characterised by the criteria inputted to the system, algorithms for generating new knowledge, and creating new knowledge from prior knowledge (Michalski 1983).

Raamesh and Uma (2009) applied attribute selection and clustering techniques to mine the knowledge in their test case system. Riel and Boonyasopon (2009) presented an application of knowledge mining to extract new explicit and implicit knowledge hidden in a large collection of electronic text documents. Vladislavleva et al. (2010) presented an approach for knowledge mining from a sparse and repeated dataset and variable selection and sensitivity analysis using genetic programming model ensembles, although Gilbert et al. (1998) and Keijzer (2004) have investigated the variable selection via genetic programming.

2.2.2 Missing Data and Data Mining

The missing data issue has been discussed in the literature (Little 1992; Affifi and Elashoff 1966; Pigott 2001; Raghunathan 2004; Ibrahim et al. 2005; Horton and Kleinman 2007). Missing data might be missing completely at random or missing at random.

Brown and Kros (2003) investigated the impact of missing data into the DM and knowledge discovery. Their study addressed that ignoring the missing data problem can introduce bias in the evaluated model and lead to inaccurate DM results (Brown and Kros 2003).

Different methods have been applied for dealing with missing data, such as regression imputation that predicts the missing value based on the relationships between the variables. Horton and Kleinman (2007) briefly reviewed the methods for incomplete missing observation in regression models as follows:

- The complete case only method is the simplest and easiest way to deal with missing data where the case with missing data is deleted. This method could be useful when there is a large amount of data collected and the amount of missing data is small (Pigott 2001; Horton and Kleinman 2007).

- Maximum likelihood is an alternative approach, which also assumes that missing data is missing at random. For each observation with missing data, multiple entries are created in an augmented dataset for each possible value of the missing covariates, and a probability of observing that value is estimated given the observed data and current parameter estimates (E-step) (Pigott 2001; Horton and Kleinman 2007). The augmented complete data dataset can then be used to fit the regression model, accounting for these weights. One of the complications of this method is the need to model the nuisance distribution of the covariates. In some settings with only a few categorical variables a saturated multinomial distribution can fit. Another complication for maximum likelihood relates to the calculation of the standard errors of estimates.

- Another approach to accounting for missing predictor data is the use of weighting methods. In this approach, a model for the probability of data being missing fits, and the inverse of these probabilities are used as weights for the complete cases (Horton and Kleinman 2007). This approach is mainly for a single missing predictor but becomes considerably less tractable with multiple missing variables.

- Bayesian approaches have been applied more generally. Ibrahim et al. (2005) described estimation with a prior distribution on the covariates, and there is a close relationship between the Bayesian approach, maximum likelihood and multiple imputation methods (Horton and Kleinman 2007).

- Multiple imputation is a combination of a number of imputation methods into a signal method. First, generating several possible values for missing observations are created that result in the creation of a number of “completed” datasets. Second, each of these completed datasets is analysed using a standard analysis procedure. Finally, the results are combined to obtain the multi-impute estimates.

2.3 Grinding Technology

2.3.1 Modelling and Knowledge Management System in Grinding

In literature, different methods were investigated by researchers to analyse model grinding process such as empirical models, analytical models, simulations, and finally artificial intelligence (AI) (Malkin 1989; Midha, Zhu and Trmal 1991; Chiu and Malkin 1993; Sakakura and Inasaki 1993; Li, Rowe and Mills 1999; Li et al. 1999;

Gupta, Shishodia and Sekhon 2001; Zhou and Xi 2002; Hecker and Liang 2003; Agarwal and Roa 2005; Kwak, Sim and Jeong 2006; Choi and Shin 2007; Choi et al. 2008; Stepien 2009). Tonshoff et al. (1992) and Brinksmeier et al. (2006) carried out comprehensive studies of grinding models.

The empirical models are functions of grinding kinematics conditions that are successful in industry but limited to the experimental conditions where wheel type, workpiece material and lubricant type needed to be adjusted (Malkin 1989; Choi et al. 2008). Kinematics models support the understanding of grinding processes on the basis of somewhat simplified local material removal process descriptions (Brinksmeier et al. 2006).

Empirical model methods tend to be of limited value since a small change in a variable that is uncontrolled can have a large effect on the model (Li, Rowe and Mills 1999; Li et al. 1999). Therefore, analytical models were developed to generate more generic models regardless of grinding wheel and workpiece material using simulation and probabilistic distribution (Chen and Rowe 1996a; Chen and Rowe 1996b; Chen et al. 1996; Chen et al. 1998; Zhou and Xi 2002; Hecker and Liang 2003; Agarwal and Rao 2005; Kwak, Sim and Jeong 2006; Stepien, 2009).

Knowledge-based expert systems in grinding normally use a knowledge base based on human experts and production rules, and they provide a solution through an inference procedure, e.g. on desirable grinding conditions for a given grinding situation or on selection of a grinding wheel (Midha, Zhu and Trmal 1991; Sakakura and Inasaki 1993; Gupta, Shishodia, and Sekhon 2001; Choi and Shin 2007; Gallego 2007; Morgan et al. 2007; Malkin and Guo 2008).

Midha, Zhu and Trmal (1991) presented a knowledge-based system that makes use of knowledge engineering and process modelling for the optimum selection of grinding parameters. It was designed to give the user a general recommendation of wheel speed, work speed and infeed rate for particular grinding situations. Sakakura and Inasaki (1993) developed an intelligent database system for grinding operations. The system consists of a grinding database, a grinding rule base, a learning module and a reasoning module. Genetic algorithms and fuzzy reasoning algorithms were applied to express and extract the relationships between parameters and results. The learning module evaluates practical grinding data in the grinding database and generates fuzzy rules, which are then stored in the rule base.

Chiu and Malkin (1993) developed a computerised simulation program for cylindrical plunge grinding operations to predict several process conditions such as the grinding forces, the power, and the actual material removal.

Li et al. (1999) developed an approach for selection of grinding process conditions using the blackboard approach. The approach has shown the ability to integrate different intelligent technologies into one system. The knowledge agents consist of case-based reasoning (CBR), neural network reasoning and rule-based reasoning (RBR) (Li, Rowe and Mills 1999; Li et al. 1999).

Gupta, Shishodia and Sekhon (2001) proposed an expert system for selecting grinding conditions. Based on the inputs from the user, and the interaction with the databases and other modules, the program suggests various process parameters. The knowledge is represented in form of if-then rules. In this approach optimal infeed value is computed according to a trial infeed value. Output parameters are computed by employing the trial infeed value.

Zhou and Xi (2002) applied stochastic distribution model of the grain to the kinematics analysis, which means the kinematics interaction between grain and the workpiece are no longer considered uniform. Hecker and Liang (2003) presented a prediction for surface roughness as a function of the wheel microstructure, the process kinematics conditions, and the material properties.

Agarwal and Rao (2005) proposed an analytical model for predicting surface roughness in ceramic grinding where they assumed that the profile of the groove generated by individual grain to be an arc of circle. Kwak, Sim and Jeong (2006) developed a response surface model to predict power and surface roughness in external cylindrical grinding.

Choi and Shin (2007) developed a generalised intelligent grinding advisory system for the optimisation of various grinding processes based on Lee and Shin's (2000) evolutionary strategies. The generalised intelligent grinding advisory system is a knowledge-based optimisation system that uses analytical models and empirical data, as well as heuristic rules. Process models are constructed by incorporating three knowledge representation methods: empirical models established by the training module using experimental data, analytical equations and heuristic knowledge extracted from production rules or expert knowledge.

Choi et al. (2008) developed a generalised grinding process model for surface roughness and grinding power. The model depends on coefficients that could be determined through a number of designed experiments, which could be considered as a drawback.

Aurich et al. (2009) presented a model based on the machine interaction. The model was more focused on describing the interaction of the grinding process and grinding machine set up, e.g. the grinding kinematics, the material and shape of workpiece and the properties of the wheel.

Stepien (2009) developed a probabilistic and analytical model for the grinding process based on assumption that the burns, vibrations and elastic deflections are ignored.

The conventional modelling strategies are not efficient when practical search space is too large, non-linear, and has a complex process such as a grinding operation. Recently, many researchers have been trying to develop non-conventional optimisation techniques such as revolutionary algorithms, genetic algorithms, ant colony algorithm, and fuzzy logic. These methods have been applied to grinding modelling problems because of their capability to handle uncertainty and flexibility (Sakakura and Inasaki 1993; Chen et al. 1998; Vishunpad and Shin 1998; Lee and Shin 2000; Saravanan and Sachithanandam 2001; Saravanan et al. 2002; Nandi and Banerjee 2005; Nandi and Pratihar 2004; Slowik and Slowik 2008). Since most of grinding knowledge is heuristics and rules that can be expressed on the form if-then, fuzzy logic-based schemes can maintain the benefits of the simple rule-based (if-then) systems while being able to manage the possible imprecision or vagueness of obtained knowledge.

Nandi and Pratihar (2004) proposed an automatic approach of a fuzzy logic controller using a genetic algorithm (GA). The key contribution was designing the knowledge base of a fuzzy logic controller (using a GA) for making predictions of power requirement and surface finish in grinding.

Nandi and Banerjee (2005) presented an intelligent approach for modelling of the cylindrical plunge grinding process based on fuzzy basis function neural network (FBF-NN) using a GA. The proposed structure of FBF-NN provides a way for developing a comprehensible (near complete) model with multi-dimensional output variables of the input-output relationships of a complex process, such as grinding in manufacturing.

Morgan et al. (2007) developed an intelligent grinding assistant that links with CNC machine and analyses of grinding performance of the machine in the real time. Gallego (2007) developed advanced software for centreless simulation. Malkin and Guo (2008) developed advanced and intelligent software that has the functions of simulation, calibration and optimisation for the grinding process. These systems can predict the grinding results regarding the cycle time, part form error, burn occurrence and size variation for a given grinding condition. However, for specific grinding problems, the process planning in industry is still done by trial and error on the machine (Morgan et al. 2007; Oliveira et al. 2009).

The trend of the future could be the combination of different types of models, process analysis methods, and knowledge management enabling the advantages of each approach to be exploited simultaneously (Tonshoff et al. 1992; Brinksmeier et al. 2006; Oliveira et al. 2009). Manufacturing companies could know much more about the planning and the problems of grinding processes (and how to solve them) if they handle their data and knowledge in a more reliable, efficient and accessible way.

Effective knowledge and data management could reduce lead times considerably (Brinksmeier et al. 2006; Oliveira et al. 2009).

Tyrolit, a grinding supplier company (Tyrolit 2010), has integrated a product finder search engine for customers to find their product where the user can select the industry, work process, material and brand.

Winterthur, an abrasive company (Winterthurusa, 2010), developed engineering work sheets for calculating grinding time. The grinding conditions are divided into input and output since the worksheet is concerned about calculating the grinding cycle including dressing and labour. Some important data is not recorded, such as material type, wheel name, dressing parameters (dressing lead and dressing depth), depth of cut, surface roughness and feed rate. Although Winterthur’s abrasive worksheet designed an intelligent sheet to calculate the grinding cycle time, important parameters for describing the quality of grinding parts on such surface roughness is not recorded.

2.4 Research Gap

From the literature review of KM, the main objective of KM is to manage knowledge processes to maintain current knowledge and create new knowledge. New knowledge is created by combining existing knowledge or by generation of novel concepts through knowledge sharing. An interesting characteristic of knowledge is that its value grows when shared (Bhirud, Rodrigues and Desai 2005). The problem with earlier technology was that it ignored the tacit aspects of knowledge.

The grinding process depends highly on skilled engineers or technicians who have both explicit and tacit grinding knowledge. In a highly dynamic market, these highly trained engineers or technicians may leave their job at any time, taking with them their knowledge and experience. As such, organisations need to find a way to keep their knowledge available when it is needed: not only the explicit knowledge but even the tacit knowledge.

Although understanding and modelling of grinding are advanced and have been investigated by many researchers with many initial experiments completed, it has not been implemented in the industry. From the above literature review, the models are usually valid only for specific cases (wheel or work material) or set-ups and none of these models or systems addressed the situation associated with missing or incomplete data. On the other hand, for effective case-based reasoning, the number of cases should be high. The available grinding cases may not have the same input and output parameters.

Recently, many researchers developed grinding models using a genetic algorithm, tabu search, and fuzzy neural network (Sakakura and Inasaki 1993; Choi et al. 1998; Vishunpad and Shin 1998; Cheol and Shin 2000; Saravanan and Sachithanandam 2001; Saravanan et al. 2002; Nandi and Banerjee 2005; Nandi and Pratihar 2004; Slowik and Slowik 2008). These approaches provide different roughness models and work well for the given conditions. However, models of specified format deduced from different sets of grinding data cannot be applied to other situations, since grinding is a complex and non-linear process. In practice, it is necessary to provide a model that can give guidance for a wider range of applications so as to cope with situations where the recorded data is incomplete or missing.

The main problems are identified as follows:

- Successful use of grinding in practice is highly dependent on the level of skills and experience of the machinist and engineer (King and Hahn 1986; Salmon 1992; Kegg 1993; Rowe et al. 1994; Chen 2002a; Morgan et al. 2007; Choi and Shin 2007; Aurich et al. 2009; Oliveira et al. 2009). If they leave the job, the company will lose their knowledge and experience.

- Grinding is still an unpredictable process; if there is no suitable knowledge and heavy reliance on experience because of large number of variables there will be involved and inadequate understanding of the relationships between these variables and the grinding process performance (Sririvason 1981; King and Hahn 1986; Salmon 1992; Kegg 1993; Rowe et al. 1994; Chen 2002b; Morgan et al. 2007; Choi and Shin 2007; Choi et al. 2008; Mukherjee and Ray 2008; Aurich et al. 2009; Oliveira et al. 2009; Hou, Li and Zhou 2010).

- The trend for the future could be the combination of different types of models, process analysis methods, and knowledge management, thus enabling the advantages of each approach to be exploited simultaneously (Tonshoff et al. 1992; Brinksmeier et al. 2006; Oliveira et al. 2009).

- Inherent variability of grinding behaviour even where the same grinding conditions from experience are employed (King and Hahn 1986; Salmon

1992; Kegg 1993; Rowe et al. 1994; Chen et al. 1999; Chen 2002a; Aurich et al. 2009; Oliveira et al. 2009).

- In grinding, it is difficult to find enough cases and reliable data process behaviour. Most grinding cases found in publications do not provide complete grinding parameter records (Malkin 1989; Salmon 1992; Rowe et al. 1994; Chen 1995; Li 1996; Chen et al. 1999; Chen 2002b).

- From the review of the current software, all of the discussed suites are more concerned about the explicit knowledge of such articles and procedures and their documentation, rather than the tacit knowledge, which is the most difficult to express and manage. The cost of implementation and maintenance of any current KM suites is considered as an important issue.

- While groupware and intranet facilitate the knowledge exchange and the variation of a tremendous amount of knowledge, it is very difficult to extract the exact knowledge efficiently from them (Baeck 1999; Lyer and Aronson 2000; Landqvist and Teigland 2005).

- The messages of ENoP technology are not stored in a single repository, which makes it difficult for newcomers to access and search for historical information (Lyer and Aronson 2000; Teigland 2003; Teigland and Wasko 2004; Landqvist and Teigland 2005).

Some of these problems will be investigated and solved in this thesis by:

- Providing an up-to-date knowledge warehouse of grinding processes, this would facilitate knowledge gathering, storing, organising and sharing for grinding parameters. Also, it would help the current and new workers to deal with any grinding case and keep records to facilitate the selection of grinding conditions in the future.

- Documenting the problems and solution that might occur or affect the grinding process. It would enhance the problem-solving time and improve customer satisfaction.

- Flexible and easy to used Internet-enabled knowledge system, to facilitate collaboration and knowledge capture. It would connect the employees together no matter where they are and provide them with 24-hour access to their system.

- Creating a database of all postings such that individuals could easily find previously discussed topics as well as educate employees on how to use the new technology. That will encourage and facilitate tacit knowledge sharing and exchange between employees.

- Provide a method that can give guidance for a wider range of applications so as to cope with a situation where the recorded data is incomplete or missing.

CHAPTER 3 GKWS ANALYSIS AND SYSTEM DESIGN

This chapter describes the process for the design and development of the grinding knowledge warehouse system (GKWS). The overall procedures of developing GKWS are illustrated. The user demand is investigated by comparing two community of practice (CoP) in a company A. system function analysis is conducted using a sequence diagram, a use case diagram and an activity diagram.   General description of GKWS is presented. Finally, the methodology of developing GKWS is identified.

3.1 Procedures of GKWS Development

Many factors must be considered in the design of GKWS, such as problem definition, well-defined objectives, a feasibility study, requirement investigation and user involvement and participation. All of these factors affect the successful development and implementation of GKWS. The procedures of overall GKWS development are shown in Figure 3.1. The following sections present a detailed description of system analysis, design and development stages.

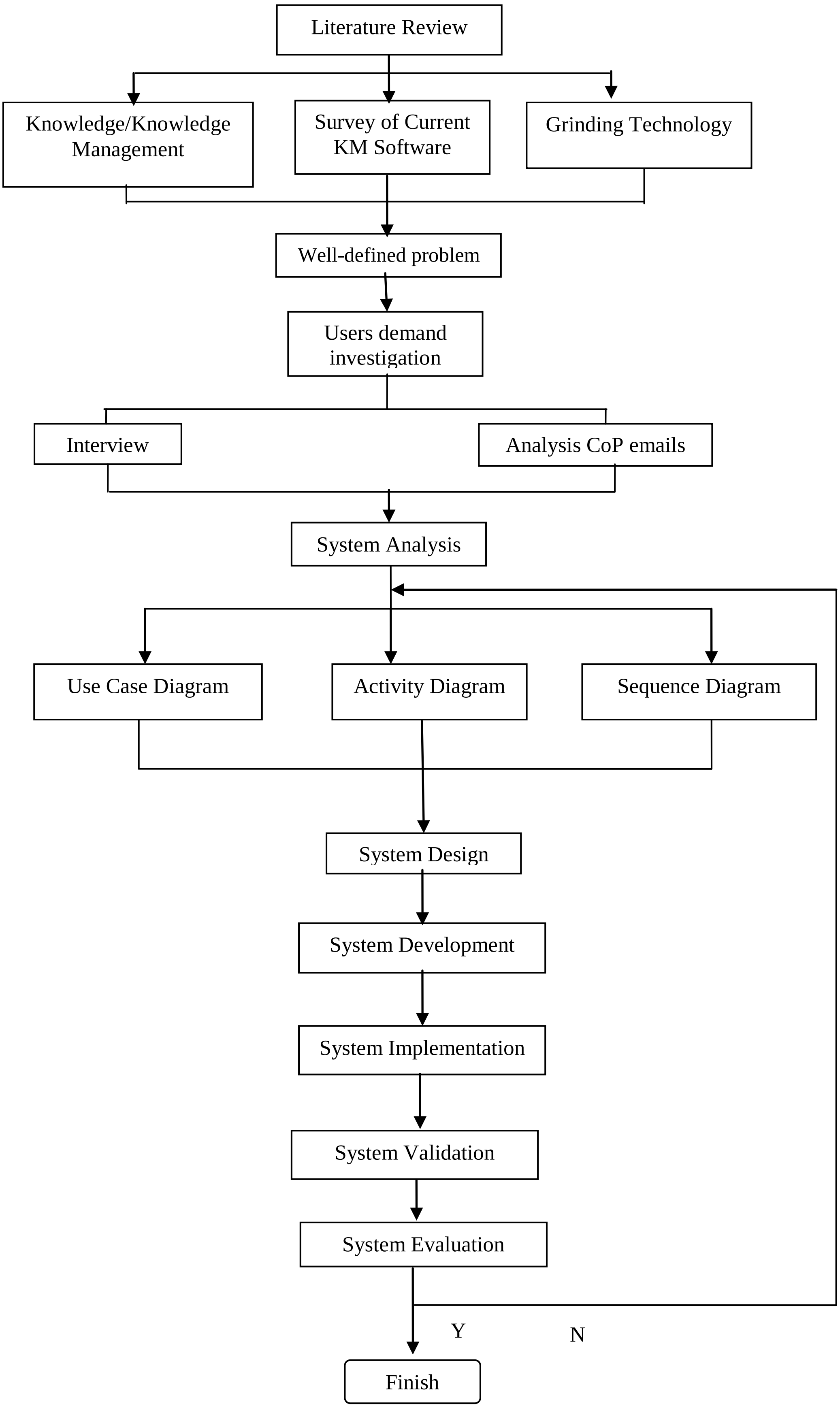


Figure 3.1 Procedure of GKWS development

3.1.1 Investigation of user demands

In order to investigate the user demands:

- a survey of relevant knowledge management (KM) software was carried out in chapter two,
- two representatives of CoP communities were interviewed from Company A,
- CoP emails were analysed from Company A. The results are discussed and summarised in this chapter.

3.1.2 Comparison Between the Performance of CoP in Two Manufacturing Operations

In order to develop an overview and awareness of the grinding operations and knowledge sharing techniques that took place in Company A, several interviews and discussions were carried out with CoP representatives in Company A. The interviews help the knowledge engineers to recognise the needs and demands of the users. For process planning in the grinding industrial floor, explicit knowledge is found in the machining data handbooks for wheel, dressing and material and the written sheets and procedures for previous ground parts. The tacit knowledge existed in the technician and engineer's heads. At Company A (until the date of the interview) the tacit knowledge was shared and exchanged through phone calls, informal meetings, and verbal questions on the industrial floor. For selected grinding conditions, a grinding expert can often make decisions based on their experience without taking all the required information or by using a method of trial and error. This implies that they make their decisions based on their tacit knowledge.

The interview feedback reveals that:

- there is a need for more reliable and efficient CoP system.
- Company A are looking to make the knowledge through CoP to be effectively exchangeable and retrievable.
- the CoP representative in Company A emphasised the importance of the CoP facilitator role to encourage the employee to use the CoP.

CoP is used as a sharing knowledge tool between the employees. These communities, up to March 2007, were using regular email for sharing and exchanging knowledge. Statistical analysis between two different communities was carried out to evaluate their performance. From the CoP emails of group A and B from March 2005 until November 2006, it can be noticed that the CoP tool has been mainly used for questions and answers, passing documents, calling for conferences or events, passing web links, sending NewsBox information and so on. Table 3.1 shows the result for Group A. The total number of emails in that period of time was 157.

Table 3.1 Number of emails for group A from March 2005-November 2006

Group A						
Questions and answers	Passing files	Calling for events	Web links	NewsBox	Others	Total
86	15	36	4	8	8	157

Figure 3.2 represents the percentage of each category in group A. It shows that 54% of the employees used CoP to share their knowledge by sending questions and answers.

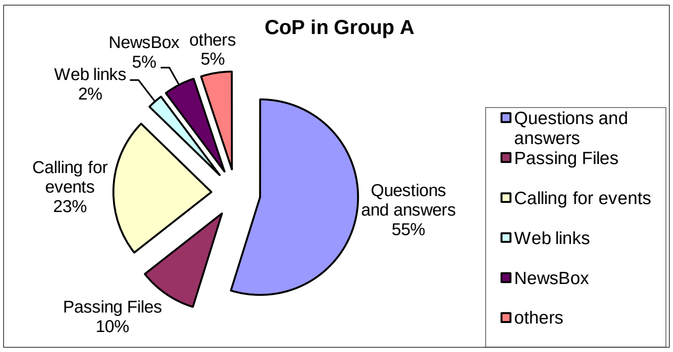


Figure 3.2 Percentage for Group A

While, for Group B, the total number of emails in the same period of time is only 33 emails, as shown in Table 3.2.

Table 3.2 Number of emails for group B from March 2005-November 2006

Group B					
Questions and answers	Passing files	Calling for events	Web links	Others	Total
16	11	4	2	0	33

Figure 3.3 represents the percentage of each category for group B. It shows that 49% of the employees used CoP to share their knowledge by sending questions and answers.

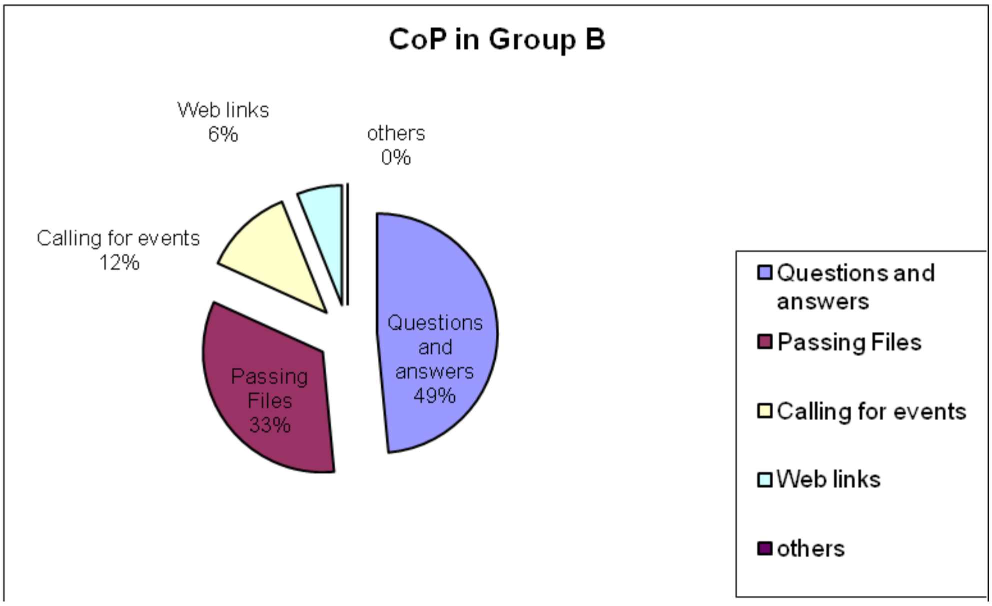


Figure 3.3 Percentages for Group B

Both cases showed that the highest percentage of usage of CoP was in the question and answer category, which means that CoP could be considered as a good tool for knowledge sharing, as shown in Table 3.3.

Table 3.3 Percentages for groups A and B

	Questions and answers	Passing files, web links and NewsBox	Calling for events	Others
Group A	54.9%	17.2%	22.9	5.1
Group B	48.5%	39.3%	12.1%	0%

The number of emails of group A is much more than the emails of group B, which means the CoP was used more frequently in group A than B, as shown in Table 3.3. The reasons were justified by CoP representative of group B: the lack of encouragement to use CoP, the CoP system is not reliable all the time, users prefer to contact each other directly rather than using emails and the frequent change of the CoP facilitator.

For group A, the number of asked questions is 40. From these questions, 10 questions have not been answered. That means 75% of these questions have been answered. Thirty-six of these questions (90%) were about the manufacturing process. The total number of answers is forty-six. The answers were classified into three categories: answers containing knowledge, contacts, or others. Seventy five percent of these answers contain manufacturing knowledge, as shown in Table 3.4.

Table 3.4 Analyses for the answered questions

Percentages		
Answers containing knowledge	Answers containing contact details	Others
75%	17%	8%

From the interview with Company A and the analysis of CoP email, it can be concluded that the CoP is a useful tool for sharing knowledge and creating new knowledge but it needs support and encouragement from the CoP representative. The problem with the current CoP is how to retrieve and extract the required knowledge efficiently with limited time and effort. The reliability and accessibility of CoP are important issues in order to share knowledge effectively.

3.2 System Function Analysis

Unified Modelling Language (UML) is a standardised graphical visualisation language that includes a set of graphic symbols and connectors (Ambler 2004). UML is being used to present the models of GKWS. The models are described with diagrams to permit engineers from different backgrounds to effectively visualise and understand the system. UML helps describe the system’s architecture and enhances understanding of its complex systems. UML is a general purpose modelling language that includes graphic notation techniques to create visual model for software engineers.

Ambler (2004) classified thirteen types of diagrams that can be categorised into three categories as follows:

- Structure diagrams emphasise the events that must take place in the system being modelled:
    - Component diagram
    - Class diagram
    - Composite structure diagram
- Behaviour diagrams emphasise what must happen in the system being modelled:
    - Use case diagram
    - Activity

- State machine

- Interaction diagrams are a subset of behaviour diagrams that emphasise the flow of control and data among other aspects of the system being modelled:
    - Sequence diagram
    - Interaction overview diagram
    - Timing diagram
    - Communication diagram

In this thesis, sequence diagram, activity diagram and use case diagram were selected in the phase of system function analysis in order to create a graphical presentation of the interactions between the actor and the GKWS activities.

3.2.1 Sequence Diagram

Figure 3.4 shows a decision-making sequence diagram that illustrates a visual representation of the scenario for creating the recommended grinding condition, filling the missing data, and the interaction between users, applications and various databases.

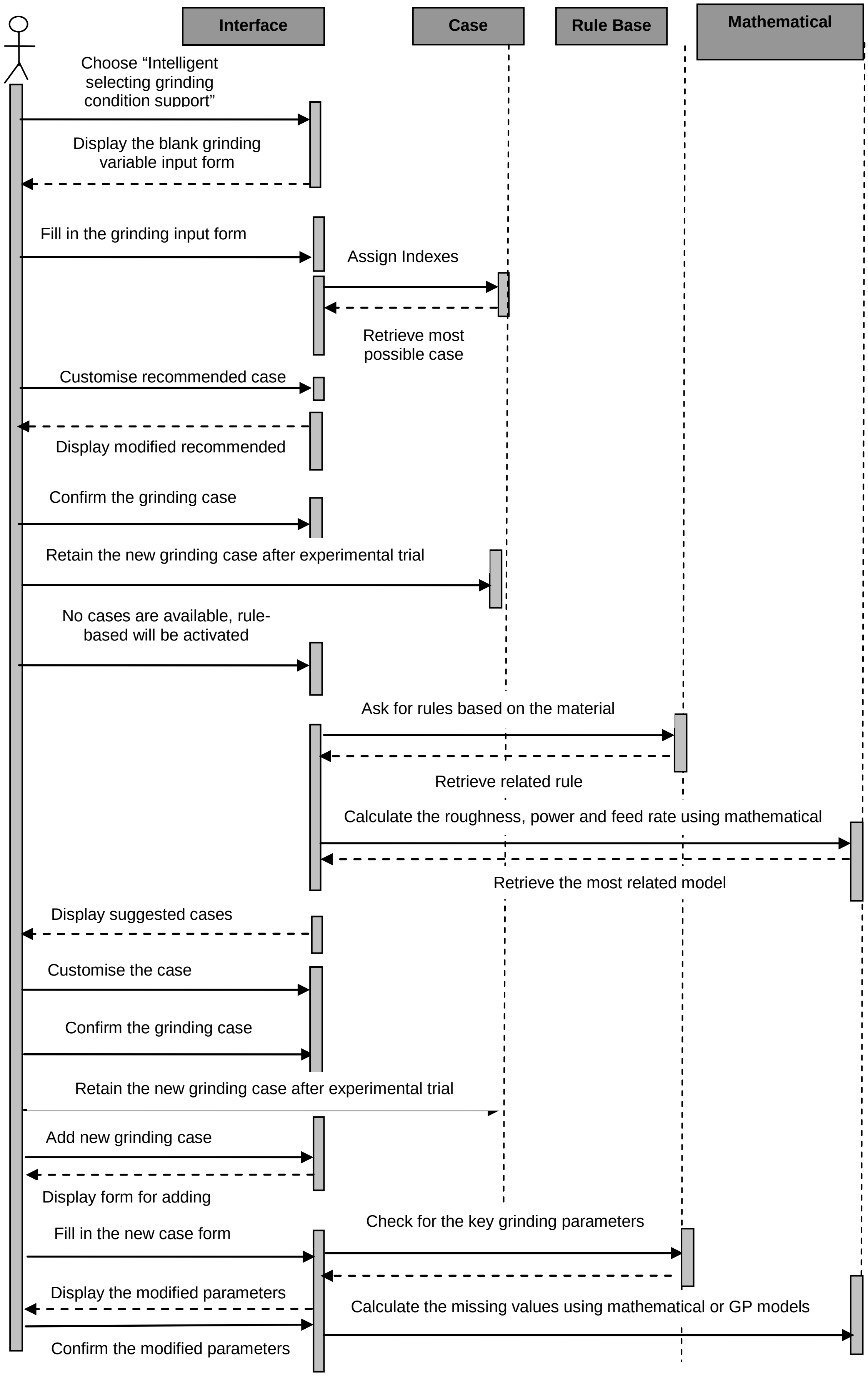


Figure 3.4 Sequence diagram for GKWS

3.2.2 Use Case Diagram

In the early stage of project development, a use case diagram is used to describe the real-world activities and represent the relationship between the actors and their actions, as shown in Figure 3.5. The actors in GKWS are the knowledge engineer and grinding engineer/technician. The knowledge engineers are responsible for managing security, validation rules, grinding cases, grinding historical data, grinding knowledge, rules based, cases based, mathematical equations, model, and CoP emails. The grinding engineer and technician are responsible for selecting grinding conditions, filling the missing parameters, searching for similar cases and validating their decisions. For CoP emails, the grinding engineers and technicians exchange and share their explicit and tacit knowledge, search for required knowledge and search for solutions for grinding problems.

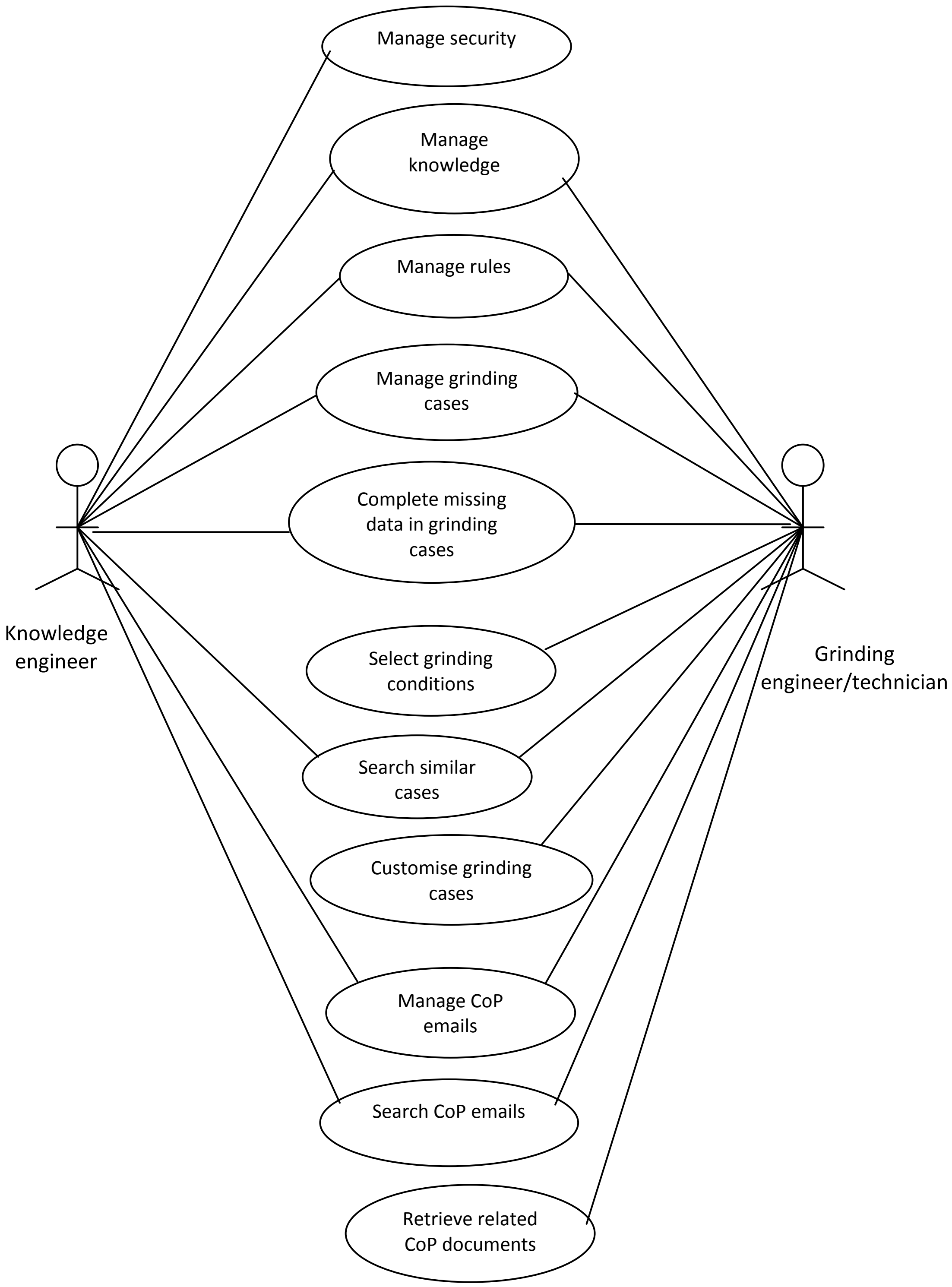


Figure 3.5 Use case diagram for GKWS

3.2.3 Activity Diagram

The activity diagram is a graphical representation for step-by-step of the flow of activities to either knowledge engineers or grinding engineers and GKWS. The activity diagram is used to describe the internal behaviour of the system and represent a flow driven by internally generated actions, as shown in Figure 3.6.

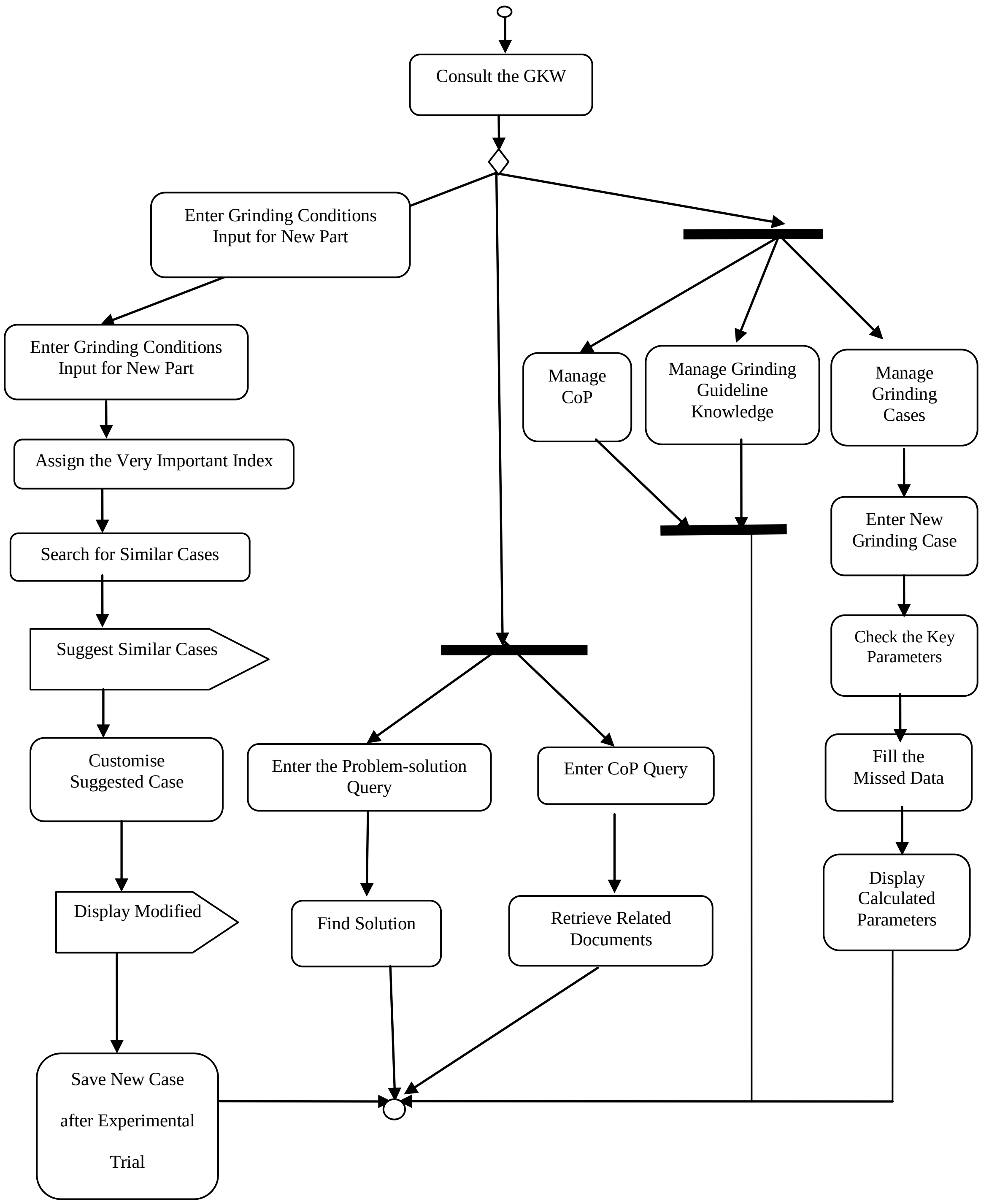


Figure 3.6 Activity diagram

3.3 General Description of the GKWS

The GKWS is developed based on the identified gap in the literature review and the analysis of CoP in Company A. The main aim of the GKWS is to facilitate transferring tacit knowledge into explicated knowledge that will be achieved by establishing a grinding knowledge warehouse and CoP collaborative tool. The knowledge engineers will be able to store and retrieve the required parameters for required grinding operation, taking into account any unrecorded key grinding conditions. GKWS can exhibit intelligent behaviour by understanding and processing grinding conditions in terms of concepts and relationships the way that humans undertake in two situations. Firstly, the GKWS will support the decision-making process for selecting grinding conditions, which will give the guidance for key required grinding conditions. Secondly, the GKWS will check the availability of the key required grinding conditions and will provide calculated values for the missing records.

The intelligent adding form will figure out the missing key data using if-then rules. In case of a missing record, the intelligent form will fill in the missing record using mathematical equations and/or GP models. The number of saved cases with the needed parameters will be increased in the GKWS, which will improve the performance of CBR for selecting grinding conditions. The GKWS will support the decision-making process for selecting grinding conditions using CBR and RBR.

On the other hand, the GKWS will have the capability to retrieve the previous grinding conditions according to grinding engineers criteria such as wheel, part number, material and/or grinding conditions.

In GKWS, the tacit knowledge is created and shared through the discussion forum tool. The knowledge engineers will be able to share and exchange their tacit and explicit knowledge through the discussion forum. The tacit knowledge in the form of questions and answers is more accessible and sharable now for employees at any time using the discussion forum category. The top level of the architecture of the design system is shown in Figure 3.7. The sub-systems are the main part for the GKWS that adopt the methodology of knowledge warehouse, which is described in the next chapter. The sub-systems will be discussed more deeply in the next sections.

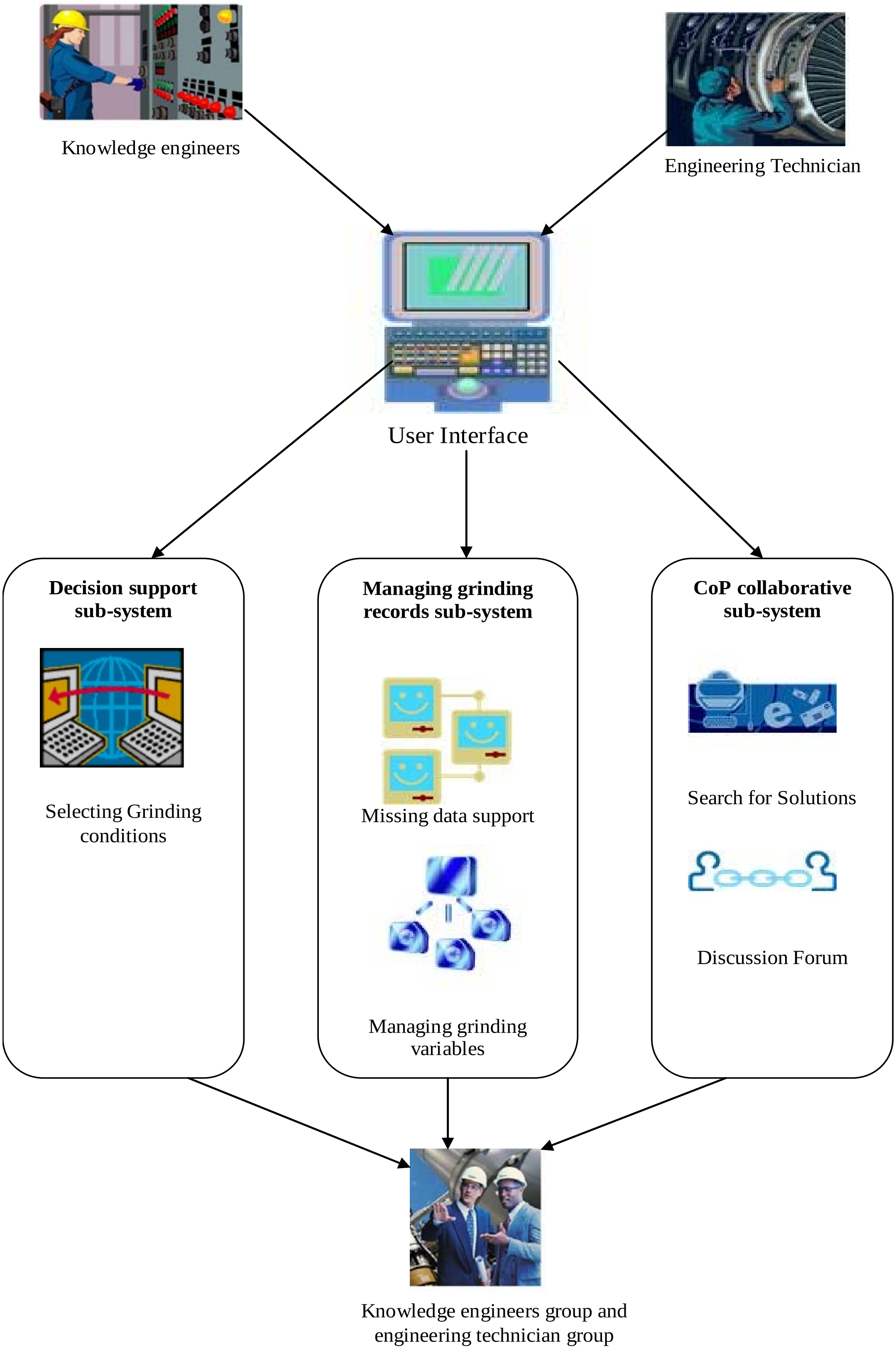


Figure 3.7 Top level of architect of design system

3.3.1 Decision Support Sub-system

The sub-system of decision support consists of an intelligent reasoning module. The CBR and RBR are developed in this sub-system. This sub-system is used to provide required grinding conditions for mainly new parts. Also, the user can retrieve the old cases for the same product too.

3.3.2 Managing Grinding Records Sub-system

Managing grinding records sub-system is not just responsible for managing the explicit knowledge but it goes further to complete missing or incomplete records. This sub-system is used to fill in the missing key parameters using if-then rules, mathematical equations or/and GP models.

3.3.3 CoP Collaborative Sub-system

CoP collaborative tool is basically a categorised discussion forum that is designed to facilitate and categorise communication between CoP members. CoP collaborative sub-system acts as a practical platform for sending and receiving messages between CoP members, extracting the required knowledge and searching for solutions. This sub-system will help to transfer, create, store, refine and retrieve the tacit knowledge by sharing knowledge through responding to questions and answers, general discussion and publishing best practice and storytelling. It also could generate new ideas and improve innovation.

3.4 The Methodology of the GKWS

In industry, choosing the correct grinding conditions is an important issue for manufacturing and technical engineers on the manufacturing floor. If the part has been ground before, the process will be just retrieving the previous grinding conditions. It is not as simple as it sounds, since sometimes for the same materials and same conditions the outputs of grinding performance, such as geometry and surface quality, will be different, which could be related to environment parameters such as coolant thermal parameters, ambience temperature, vibrations and so on. It is necessary to build up a repository warehouse to store and manage the knowledge management process for effective knowledge sharing and retrieval. When it is the first time for the part to be ground, the manufacturing and technical engineers will be looking for materials from the same group, hardness, required surface roughness and input parameters setting in order to make their decisions for selecting grinding conditions. The new GKWS system is developed for manufacturing processes, especially grinding technology, to facilitate and assist the decision-making process for selecting a grinding condition for the new part and efficient retrieval of the key parameters and comments on the previous ground part. The target users are manufacturing engineers and technicians working in the same company or in different geographical areas. Taking into account the advanced KM methodologies and technologies, the new GKWS will automatically check for the key grinding parameters and it adopts an intelligent algorithm to deal with missing record. That will increase the efficiency of the CBR system since the cases are kept up-to-date and the key parameters are available. The GKWS can provide output in the form of expert opinions, analyses and recommendations based on the CBR and RBR. The GKWS will link between the grinding parameters and the real experience in grinding floor by allowing users to add their comment and feedback.

The GKWS will encourage sharing knowledge and facilitate transferring explicit to tacit knowledge through discussion and storytelling that will be achieved through an online discussion forum. The GKWS web interface will allow the grinding CoP members to extract the exact knowledge they are looking for efficiently. The new knowledge web based system will help CoP members in grinding technology in the same company or in different geographical areas to collaborate and generate new ideas.

The GKWS is designed to facilitate and support the knowledge management process for grinding technology. It is not just creating, gathering, storing, and retrieving knowledge, moreover, it concerns keeping this knowledge up-to-date and maintaining it to contribute in generating new knowledge. It also enhances the retrieving and sharing of knowledge across the company no matter where the employees are distributed.

Based on the above discussion, the main functions for GKWS are:

- efficiently create, store, retrieve and share grinding conditions (explicit knowledge) in various forms and discovered knowledge (tacit knowledge).
- efficiently create, capture and store the CoP's knowledge in various formats such as files, forums, and images.

- manage and execute different tasks according to users' needs. For example, the users can retrieve grinding problems and their solutions or look for a remedy for a specific problem.

The GKWS includes seven modules, as shown in Figure 3.8. Figure 3.9 shows the knowledge management activities in the GKWS. These modules are explained briefly:

1. Data input module; the main task of this component is data preparation. It collects data through the data collection interface from grinding database and the interactions within the CoP. This component is not only to extract, load and integrate the data into the database facility, but also to periodically refresh the database to reflect data updates from user-system interaction.

2. The database module is created to store, retrieve and share data in various forms. This data can be transformed into the problem-solving module and then the learning knowledge discovery module, which guides the user to select the right operation conditions for grinding. The database module is based on relational database management system by using MySQL.

3. The knowledge acquisition module facilitates knowledge conversion from tacit to explicit knowledge; for instance, it directly acquires tacit knowledge from knowledge engineers or CoPs.

4. The core of the problem-solving knowledge discovery module is CBR, RBR, and MBR. This module includes data transformation, knowledge

inference engine and knowledge representation. The RBR and CBR are used to provide the engineers and workers with guidance to select the required grinding conditions or parameters for a given grinding operation taking into account the lessons learned from previous cases. The GKW provides a problem-solving module, which enables the user to find solutions for different grinding problems. The user can further browse different cases regarding different application criteria. The CBR and RBR in this research are based on the system proposed by Chen (1995) and Li (1996).

5. The learning knowledge discovery module extracts implicit, previously unknown and potentially useful rules and patterns, to modify and update the existing grinding case and complete missing records. This module provides the grinding engineer with the guidance to select the right conditions for a specific grinding operation. At the same time, it makes sure that the required grinding conditions are filled or calculated using mathematical equations or analytical or generated models.

6. The knowledge storage module facilitates efficient explicit knowledge storage and retrieval in various forms based on the required task. The knowledge warehouse manages the integration of a wide variety of knowledge, such as rules in rule base, cases in case base, model in model base etc. into a functioning whole. Various types of knowledge are included in this module such as numerical data, text streams and validated models, as well as the algorithm used for manipulating them.

7. The knowledge analysis management module provides the interfaces for target users to manage different analysis tasks.

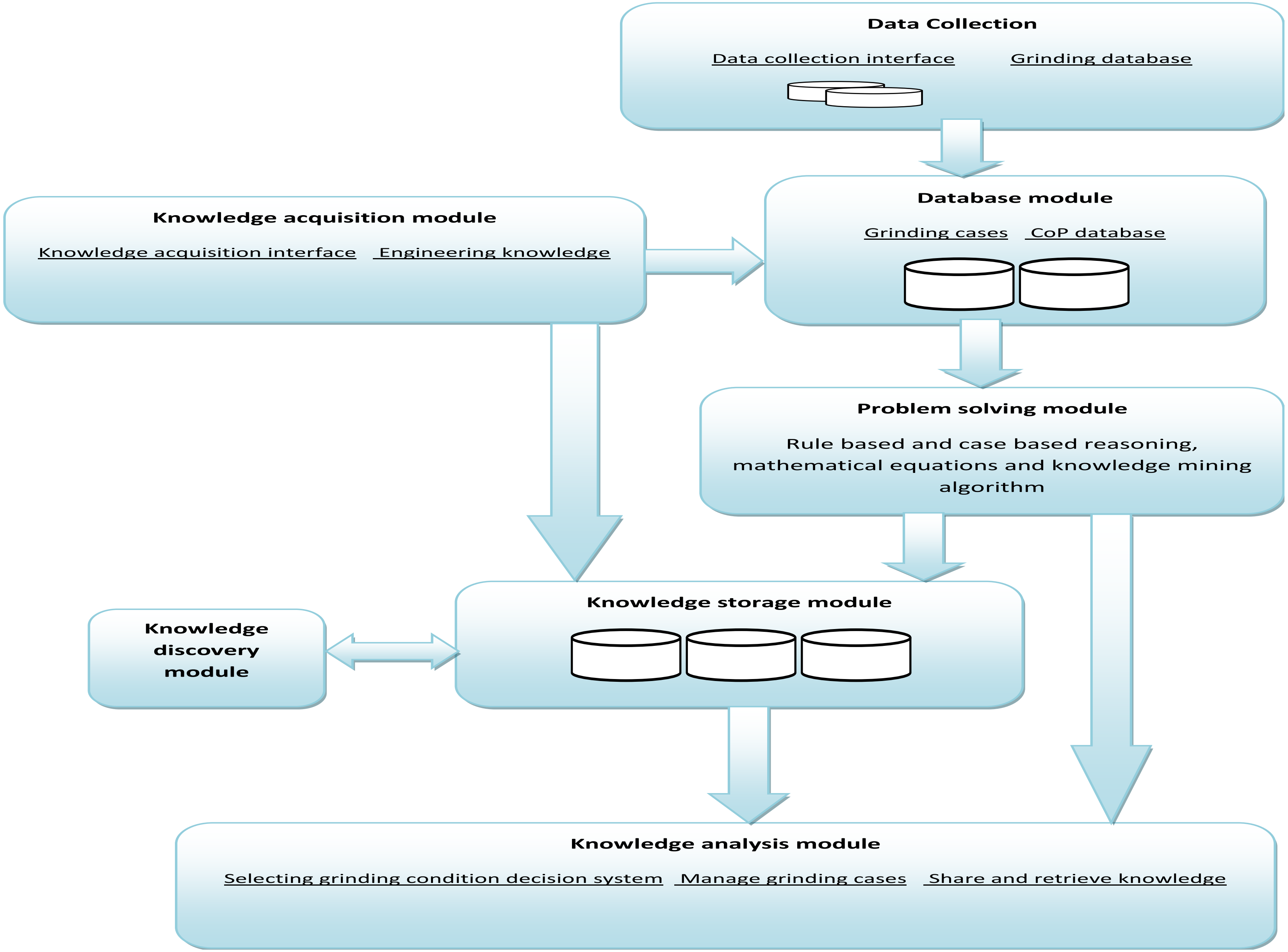


Figure 3.8 The GKWS architecture

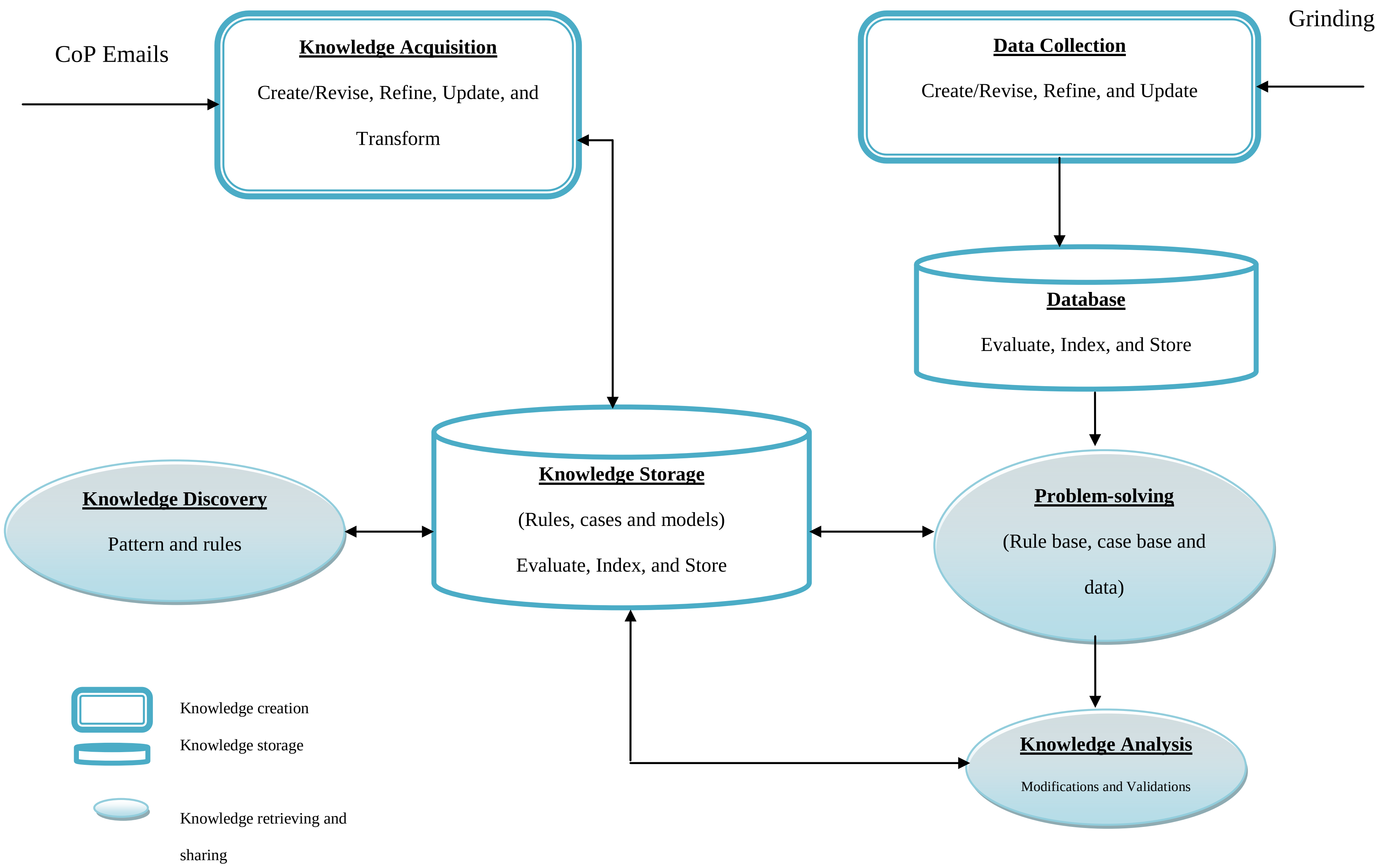


Figure 3.9 Knowledge management activities in GKWS

3.4.1 Technical Tools for GKWS

The GKWS has been designed and developed using Internet and database techniques (PHP-MySQL-Apache). The GKWS can run on stand-alone computers as well as a web server. The PHP- Hypertext Pre-processor worked as the script language, MySQL acted as the database server, and Apache was employed as the web server.

PHP, Apache, and MySQL are all part of the open source group of software programs. The open source movement is collaboration between some of the finest minds in computer programming. PHP is embedded within HTML; in other words, PHP are ordinary HTML pages that escape into PHP mode only when necessary.

PHP can be used to access different databases such as MySQL. MySQL and PHP are frequently used together. MySQL provides the database part and PHP provides the application part of the web database application while Apache acts as the web server. MySQL supports SQL language and has high performance. MySQL is a flexible system of authorisation and allows some or all database privileges to a specific user or group of users.

SQL commands are sent to the database server through MySQL-ODBC driver by PHP, as seen in Figure 3.10. The result from SQL queries can be retrieved and displayed using PHP. The results can be displayed using any web application.

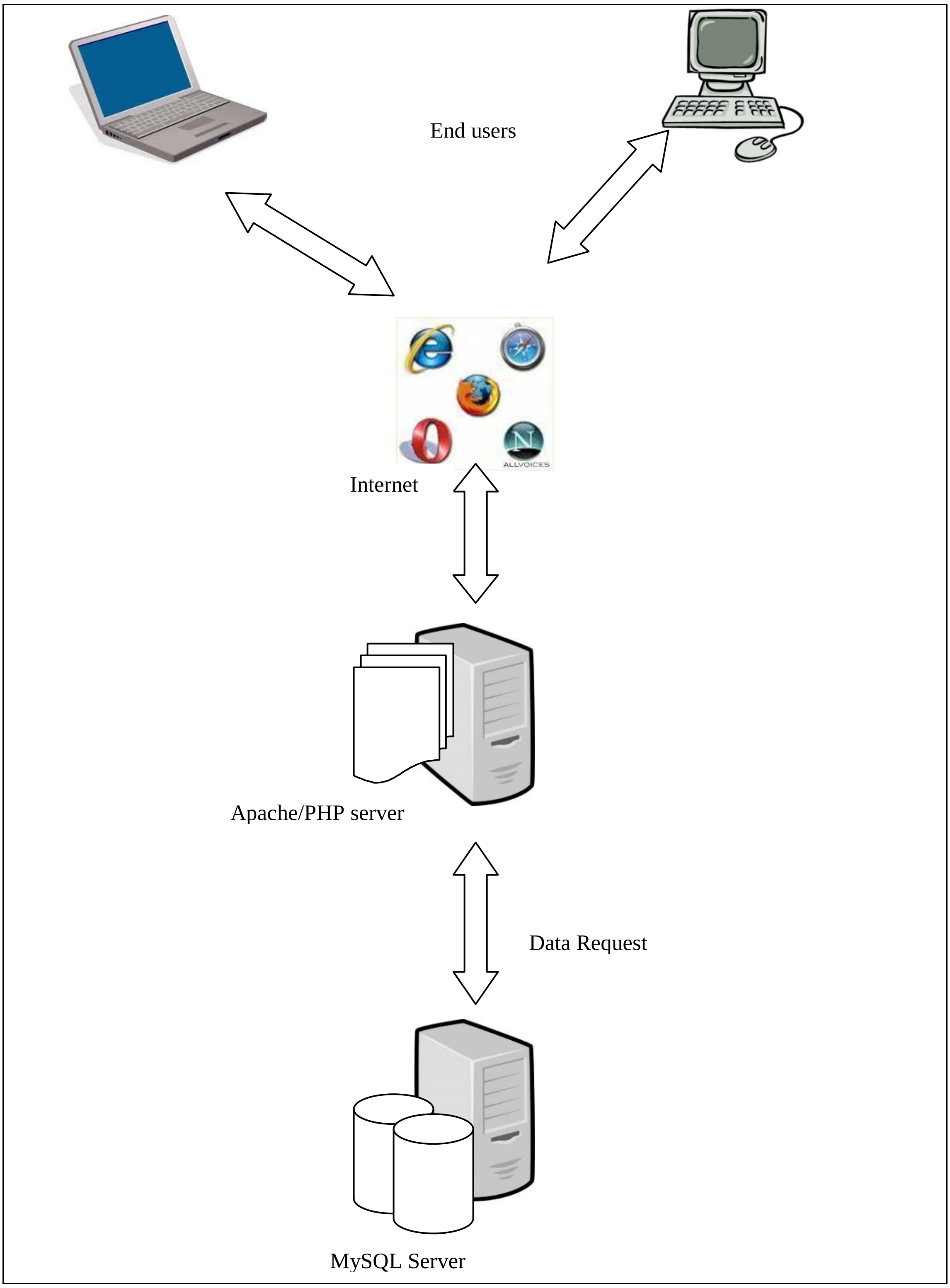


Figure 3.10 Technical system structures

3.5 Summary

The reader now realises that sharing knowledge, collaborating, minimising loss of expertise, quicker problem-solving, and better decision-making requires effective and efficient Information System (IS) and supportive culture. For organisations already using KM strategies, access to knowledge and expertise is speeding up problem-solving and response time to customers. Knowledge has no value if employees cannot share or access it. It will be more efficient for employees to retrieve and provide up-to-date knowledge to solve any problem with the click of a mouse.

The new KM system will be designed for manufacturing processes, especially grinding technology. It will help employees to share their tacit and explicit knowledge. The new knowledge base system will encourage and facilitate the sharing of tacit knowledge by allowing users to send their questions and then these questions and answers will be stored in a single repository that can be accessed by newcomers or searched for historical information. The new knowledge-based system will include an intelligent system to provide the employee with the ability to select the grinding conditions and calculate missing or not recorded data.

The CoP has been used as a tool of collaborative among company A workers. It was mainly used to send questions/answers, call for a meeting or seminar, and forward files or web links. From the meeting with a key representative of company A and the analysis of the CoP emails, the next step will be how to retrieve the knowledge efficiently from the CoP's emails. The reliability and accessibility of CoP are important issues in order to share and exchange knowledge effectively.

CHAPTER 4 DATA COLLECTION AND MANAGEMENT MODULE

This chapter represents part of the knowledge discovery in database (KDD) stage: data collection and management. This stage has two modules: the data collection and the database module. Based on understanding of the problem domain and the required data for selecting grinding condition, the data collection module is designed. The database module's design philosophy, development and management are also discussed. Also in this chapter, the knowledge storage module is developed.

4.1 Data Collection Module

The proposed web-based data collection module is completely paperless and confidential. The main task for this module is to collect the data from the user system interaction, the collection of related grinding cases and to control this data effectively. This module collects the data through the data collection interface from the grinding operational database and community of practice (CoP)'s members. All of this data may be needed for formulating the recommendations to select the grinding conditions variables, problem solving, and CoPs discussion forum.

The grinding operational database includes key grinding variables records and other related recodes, which are summarised in Table 4.1:

Table 4.1 Grinding parameters at a glance

Parameter Name	Parameter Details
Material parameters	Material composition, dimension, hardness, density, tensile, melting point, specific energy, and thermal conductivity.
Wheel parameters	Wheel name, abrasive type, bond type, grit size, grade, wheel diameter, wheel maximum speed, supplier and manufacturer.
Machine parameters	Name, country, description.
Dressing parameters	Name, type, country, dressing speed, dressing direction, number of dressing passing, redress life, dressing depth of cut, and dressing lead.
Coolant parameters	Name, delivery rate, and density.
Control parameters	Feed rate, work speed, depth of cut, spark-out time, wheel speed
Output quality parameters	Surface roughness, size tolerance, roundness

The CoP's emails are another input for this module. From the meeting with a key representative of Company A, and after the analysis of CoP for two groups, it was noticed that the CoP has mainly been used to send questions/answers, call for a meeting or seminar, and forward files or web links. The CoP discussion forum is built on these categories, so the user can go directly to the required link, which will make sharing and extracting knowledge easier.

The raw data in the proposed data collection module, including structured and unstructured data, is stored in structured database fields in order to speed up case retrieval (Grossman and Ophir 2001). The grinding conditions (wheel speed, work speed, depth of cut, surface roughness and so on) are structured data. The CoP's discussion forum, grinding articles, and grinding problems and solutions are unstructured data.

The data entry might be structured data entry (SDE) and free text data entry (FTDE). The SDE is perfect for further processing entered data, its presentation, computations, decision support etc. (Hanzlicek, Spidlen and Heroutova 2004). Examples of the structured data are feed rate, wheel speed, roughness, and wheel name.

The other type of data entry is free text data entry, which is able to quickly express any information obtained from a CoP member, independently of a predefined information structure, for example, a CoP discussion forum, articles, grinding problems and solutions and so on.

The framework for the data collection and database module is shown in Figure 4.1. In this data collection module, both data types are employed, because structured data is required for decision support for selecting the grinding operation variables, and free text-based data is capable of allowing CoPs to express and share their knowledge. Nevertheless, they will be applied in different functional models of the system, such as SDE in grinding variables information, FTDE in CoP's enquiries and grinding problems and solutions.

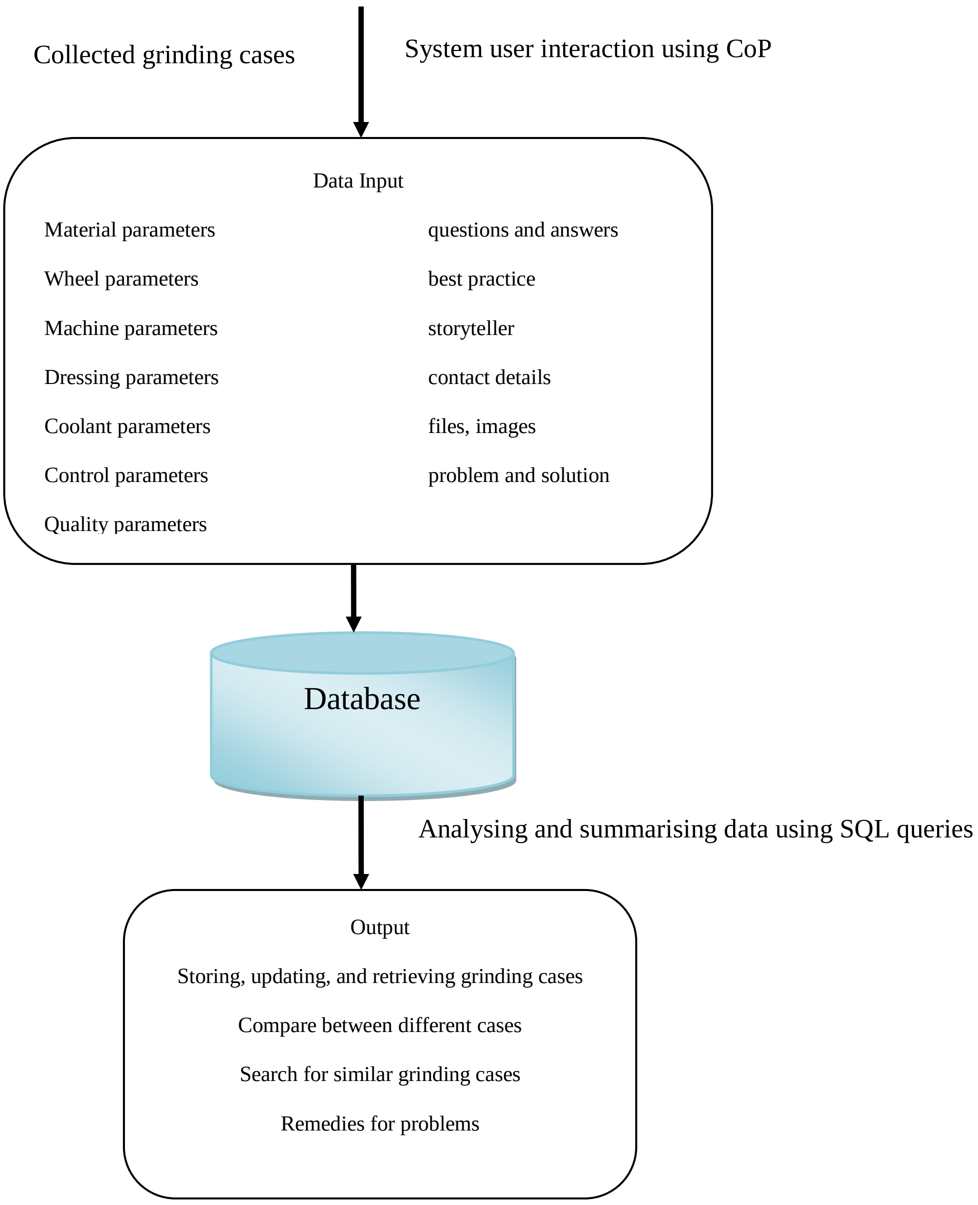


Figure 4.1 The framework for data collection and database module

4.1.1 Data Required for Selecting Grinding Conditions

In planning grinding operations, it is necessary to define the various inputs and outputs to develop relationships between these variables (Tonshoff et al. 1992; Li, Rowe and Mills 1999; Chen 2002a; Chen 2002b). Chen (2002b) distinguishes between input variables to the grinding machine and input to the grinding process, which occur at the wheelwork interface. The inputs to grinding machines are feed-rate or down feed, wheel and work speed, depth of dress, and spark out time.

The input to grinding process is the normal development at the wheel interface and describes the grinding behaviour during the grinding process, such as normal force, temperature, vibration, power etc. (Li, Rowe and Mills 1999; Chen 2002a; Chen 2002b). The process variables are affected by grinding conditions and affect the output variables.

The input variables for grinding conditions could be divided into uncontrolled and selectable variables. The uncontrolled variables are determined by the design requirements and have a significant effect on the grinding process, such as material properties, workpiece geometry, system rigidity, and power capacity. While the selectable parameters should be set and adjusted by the operator before the process based on previous grounded parts or trial and error the engineer’s knowledge and experience have a high impact on selecting these variables. These variables are cycle type, coolant, wheel speed, work speed, dressing lead, dressing depth, depth of cut and feed rate.

Output variables of the grinding system consist of the workpiece quality, productivity, and cost, which should meet the design and manufacturing requirements. The output variables are therefore the main variable to be controlled, such as power, surface finish, surface roughness, roundness, surface integrity, wheel wear, G-ratio, vibration, chatter, and grinding burn. The output process variables measure the performance of grinding operation that can be classified into process and performance parameters. The performance variables describe the quality of the ground part, such as surface roughness, surface integrity, and size tolerance. Figure 4.2 shows the grinding process variables in more detail.

The target of decision-making process for selecting grinding conditions based on the output variables must meet the requirements of the design and manufacturing requirements. In other words, the operator should set and adjust the selectable grinding conditions in order to satisfy constraints by imposed output variables and uncontrolled variables, which can not be changed by the operator. A part of the algorithms applied to open the case table to retrieve the similar grinding conditions according to one or more criteria, for example, part number, surface conditions, wheel and (or) material is presented as follows:

```
// select statement to retrieve the grinding conditions according to the user criteria) part number,
//whee, material and grinding conditions.
if (($mat != "" )&& ($_POST[cond]== "")  && ($_POST[part_id] != "")&& ($wheel != "") ){
    //retrieve the cases by material
  $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com,  wheel.wheel_id,
material.material_id, cases.part_id, casesrecord.wheel_speed, casesrecord.width AS width,
casesrecord.start_diameter, casesrecord.finish_diameter AS fd, casesrecord.hardness,
casesrecord.depth_cut AS d, casesrecord.`size_tolerance` , casesrecord.`roundness` ,
casesrecord.wheel_diameter, casesrecord.work_speed, casesrecord.feed_rate,
casesrecord.dressing_depth, casesrecord.dressing_lead, casesrecord.dressing_speed,
```

```
casesrecord.roughness, casesrecord.emp_id, casesrecord.Date, casesrecord.cal_roughness,
casesrecord.cal_feed_rate, casesrecord.cal_depth_cut
        FROM cases, wheel, casesrecord ,material
        WHERE cases.case_id = casesrecord.case_id
        AND cases.wheel_id = wheel.wheel_id
        AND cases.material_id = material.material_id
        AND cases.material_id =$mat
        and cases.part_id='$_POST[part_id]'
        AND cases.wheel_id =$wheel";
    }
  //if statement to make sure that the part is kept in the database
  if (mysql_num_rows($get_case_res)== 0 ) {
                echo "It seems that this part is a new one, go back to the new part
                <a href=\"main page.html\ new part></a>";
                exit;
     }
// get the required grinding conditions fields using select and while loop.

              $get_variable_field="select fields name  from table
 where (specify the required criteria)

$get_results= mysql_query($variable name)or die(mysql_error());

              // loop into the table to find out the needed grinding conditions using while loop
              while ($res=mysql_fetch_array($get_results)){
                             $variable_l=$res[mname_1];
                             $variable_2=$res[mname_2];
                            $variable_n=$res[mname_n];
// show up the required grinding conditions
}
```

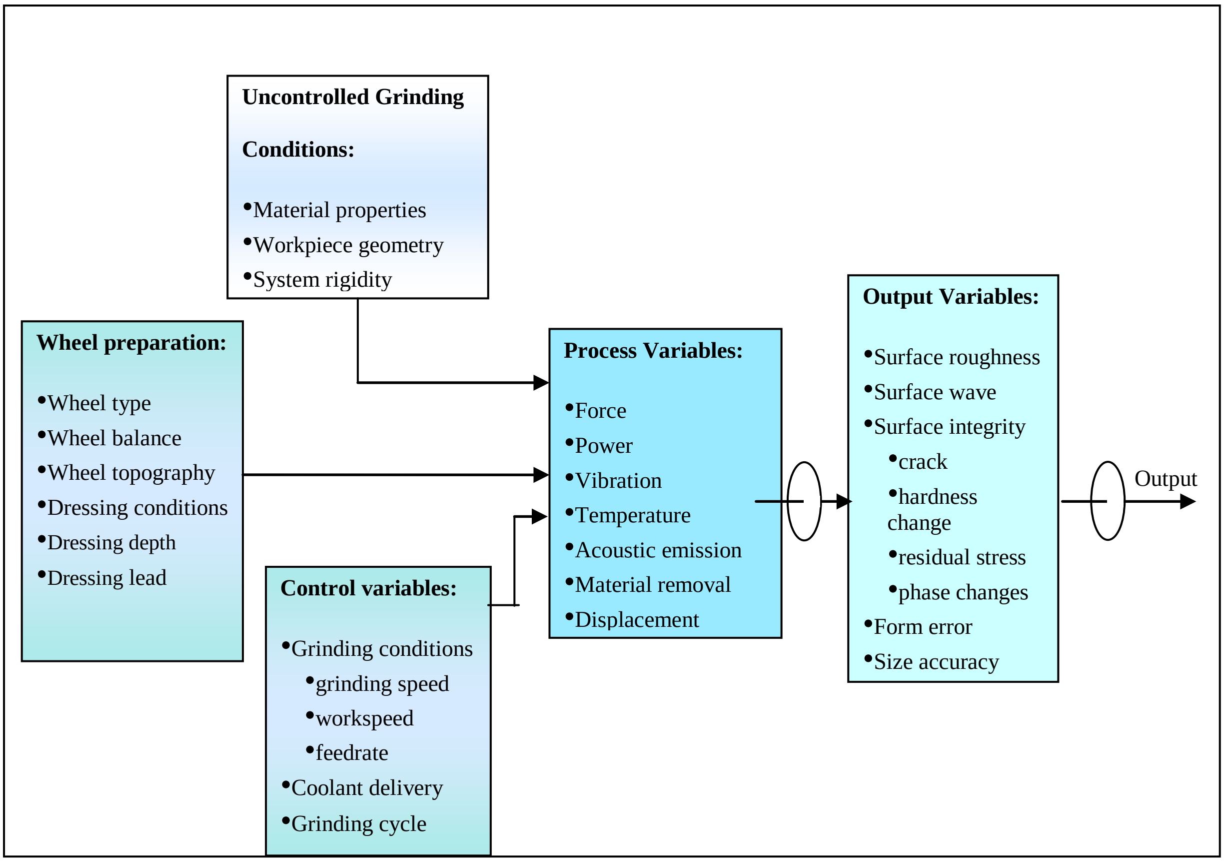


Figure 4.2 Grinding variables

4.1.2 Surface Roughness

Surface roughness is one of the important grinding variables to measure surface accuracy and precision. This variable is affected by many variables in grinding operation, such as:

- Grinding wheel; the grinding wheel characteristics have a direct effect on process efficiency, accuracy, surface roughness and surface integrity (Li, Rowe and Mills 1999). The best wheel for an application is a compromise between the ability to cut rapidly and the ability to hold form, maintain the surface roughness requirements and last a long time before dressing (King and

Hahn 1986). The specification of the grinding wheel includes five parts: abrasive type, abrasive grit size, grade, structure and bond. The surface finish is improved when the equivalent diameter increases and workspeed decreases, which means a higher number of grains of contact between the wheel and workpiece, leading to a chip thickness reduction and smoother surface finish.

- Dressing parameters; dressing prepares the cutting surface of the wheel by removing dulled grains or by cutting through them to present new shape grains (Li, Rowe and Mills 1999). Dressing conditions have an important influence on the grinding force, wheel wear, and workpiece surface creation. The dressing variables are dressing lead, dressing depth, number of passes, setting angle of the diamond, initial diamond shape, and diamond wear. The key selectable dressing variables are dressing lead and dressing depth. Results for many investigations (Pattinson and Lyon 1974; Verkerk and Pekelharing 1979) show that the coarser wheel dressing (high lead and large dressing depth) produce an open structure, which results in good cutting efficiency and lower grinding force but poorer workpiece roughness, whereas fine dressing produces a more closed structure to the wheel, which result in lower workpiece surface roughness but an inferior cutting process performance.

- Equivalent chip thickness; in practice, one of the most important and reliable basic parameters is the equivalent chip thickness because it correlates important parameters, such as work speed, wheel speed, depth of cut, and feed rate, as shown in Equation 4.1. The surface finish could be improved by reducing feed rate. For better surface roughness in grinding, the depth of cut should be minimised, as the surface roughness is inversely proportional to the depth of cut. On the other hand, there is a relationship between the equivalent chip thickness and normal force, tangential force, specific energy, roughness and G-ratio.

$$h_{eq} = \frac{v_w a}{v_s} = \frac{Q_w^{'}}{v_s} = \frac{\pi d_w v_f}{v_s} \qquad (4.1)$$

- Spark-out time; it affects the roundness and roughness of the workpiece. The spark-out period may be determined in the grinding process because the spark-out period required strongly depends on the stiffness of the workpiece, the stiffness of the machine and the efficiency of the removal process (Chen and Rowe 1999).

- Other parameters; on the other hand, the lubricant influences the surface finish and force also productivity and specific energy affects the surface roughness.

In other words, at least these parameters should be controlled in order to improve the surface roughness and the quality of the grounded part. As discussed earlier, the number of grinding variables that are required to describe grinding process is large. The relationship between process and performance parameters is highly non-linear in many cases. The successful selection for grinding conditions depends highly on understanding the complex relationship between the input and output variables that could be learnt through experience. In practice, it is difficult and unnecessary in many cases to record all the input and output grinding conditions to present an efficient knowledge for selecting grinding conditions. The grinding knowledge warehouse system (GKWS) records the key information that is needed for selecting the grinding condition without losing any key knowledge that may have a generic application sense.

4.2 Database Module

Through the data collection interface, related grinding data are transferred into the grinding and CoP databases that are contained in the database module. The database module retrieves a large variety of data from different resources, e.g. grinding control variables, input comments, and CoP enquiries. In addition, the databases may have numerous users with various requirements for viewing the databases. Therefore, the databases may have diverse purposes for further data management, such as grinding cases management, grinding cases research, and various statistics.

The database module is created to store, retrieve and share grinding cases. These cases are transformed to the problem-solving engine and then the learning knowledge discovery, which provides users with guidance to select the right conditions for manufacturing operation (for example, variables for grinding).

4.2.1 Design Philosophy

The database module is based on a relational database management system using MySQL, since MySQL is robust, easy to use, supports Structured Query Language (SQL) and open source. The database engine is built up using forms and queries. All data resides in the relational databases contained in this database module. This module implements the SQL, standard relational access language, to make use of the database indexing and query processing capabilities. The reasoning of the database module is accomplished by building SQL queries. The grinding cases have been manipulated by designing queries. SQL queries are designed to access the tables to retrieve the required details. The SQL is used for creating, editing, and updating the database. Different functions of the SQL are used, such as grouped by and summarising functions (MAX, SUM, MIN, GROUP BY).

The elementary component of this database engine is in the grinding cases table where the grinding details are stored and kept. The grinding cases table includes the input and output variables for the grinding operation such as feed rate, wheel speed, workpiece speed, removal rate, roughness, roundness etc. The grinding cases table is related to the machine, employee, material, dressing tool, grinding wheel, and coolant table. Also there is a relationship between these tables and the other tables shown in Figure 4.3. The tables and their fields are illustrated in Table 4.2.

Table 4.2 Database table fields for grinding

Table Name	**Field Name**
Cases	**Case_ID, Machine _ID, Grinding Wheel_ID, Dresser_ID, material_ID, Material_group_id, Part_id, Coolant_ID, Grinding_type, Condition_type, comment.**
Case record	**Case_rec_id, Case_Id, Wheel_speed, wheel diameter, abrasive type index, grits index, width, Start diameter, finish diameter, depth of cut, material id, material group index, roughness index, hardness index, work speed, feed rate, dressing depth of cut, dressing lead, dressing speed, dressing direction, number of dressing passing, re-dress life, size tolerance, roundness, surface roughness, hardness, G-ratio, equivalent diameter, similarity value, employee Id, date, calculated roughness, calculated depth of cut, calculated feed rate.**

Machine	Machine_ID, Machine Name, Country, Description, Image (optional)
Grinding wheel	Grinding Wheel_ID, Name, Type, Grade (hardness), Grit size, Bond, Abrasive, Country
Dressing tool	Dressing_ID, Dresser name, Dresser type, Dresser size, Country
Workpiece-Material	Workpiece_ID, Material_ID, Hardness, Strength, Hitch treatment, Size
Material	Material Group _ID, Material Group, Note
Coolant	Coolant_ID, Name, Description, Country
Employee	Employee ID, Employee_First Name, Employee_ Surname, Email, Job, Picture
Documents	Document_ID, Document, Created by (Employee_ID), Data Type, Date
Grinding type	Grinding No., Grinding Name, Details
Problems/solutions	Problem_ID, Problem Description, Solution, Solution Description
Copy-cases	Acts as a temporary storage for the matched cases, where the similarity calculation is carried out.
Problem-cases	Problem_ID, Cases_ID, Employee_ID

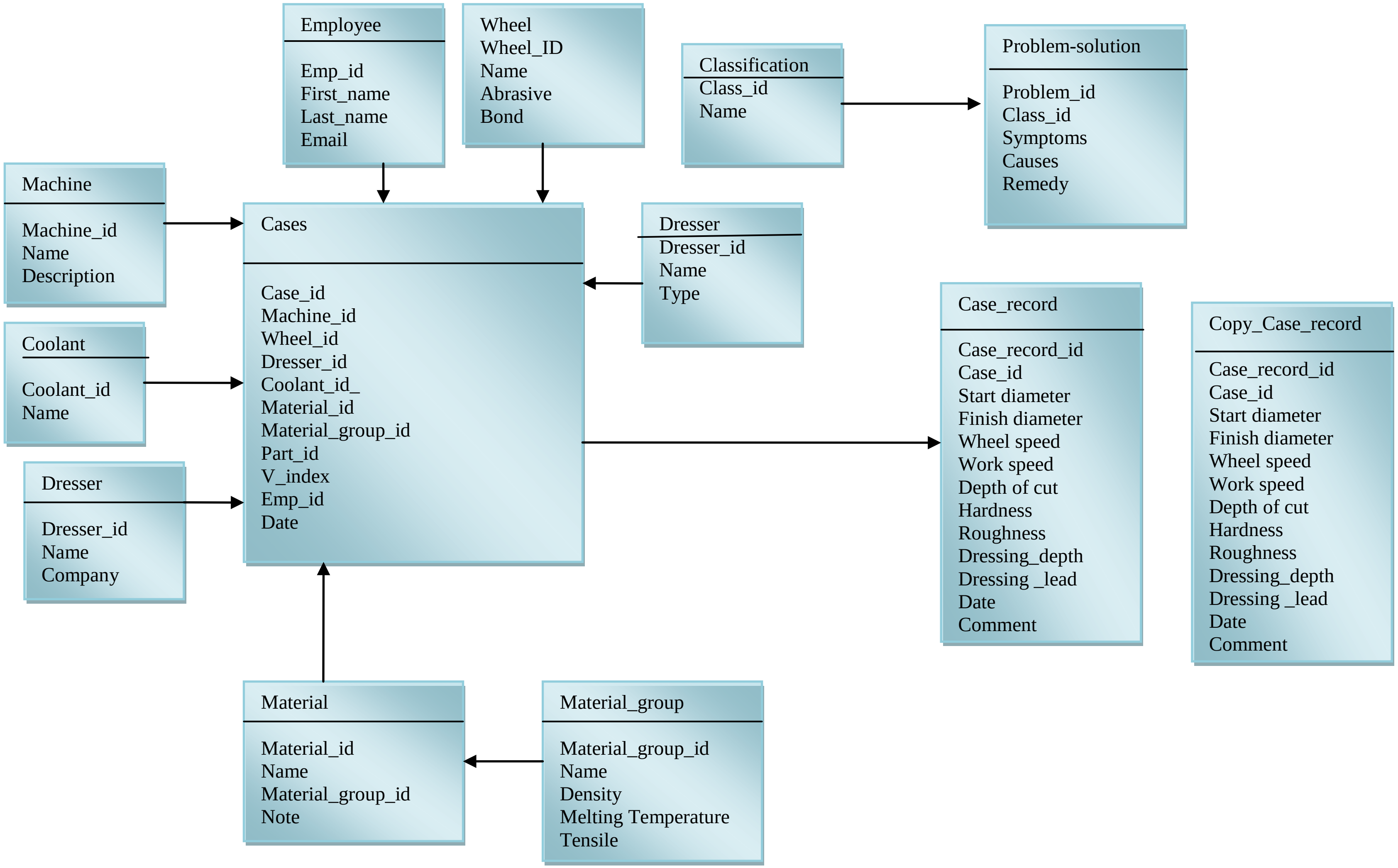


Figure 4.3 Relationship between grinding tables

The discussion forums for CoP communications contain categories, forum, topic, and message table, as shown in Table 4.3. The categories table stores different categories such as quality performance, problem-solving and more. The users can identify a category that will contain different discussion forums. The forums table holds different discussion forums that are a part of the category, such as general discussions, questions and answers, and best practices. The topic table stores the details of discussed topics, such as the subject, date, and who started it. The message table holds details of the discussion thread and the replies. In order to determine if there is any reply for a topic, a query will be run to see whether any messages has got that message as a parent. The forum has distinctive set of parts, as shown in Figure 4.4, which combine to create a system that makes discussion easy. The relationship between the tables is shown in Figure 4.5.

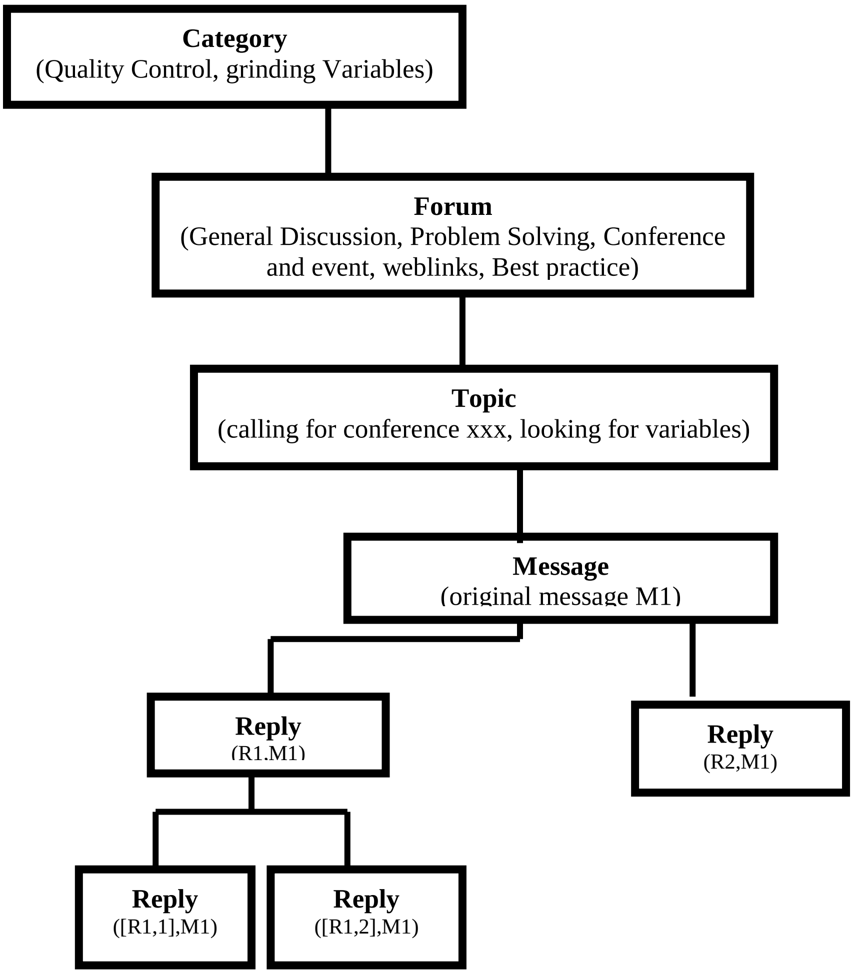


Figure 4.4 Structure of CoP Forum

Table 4.3 CoP tables

Table Name	Field Name
Category	Cat Id, name
Forums	Forum_id Cat_id name description
Topic	topic_id, date, emp_id, forum_id, subject
Message	Mesaage_id, topic_id, subject, body, date, parents

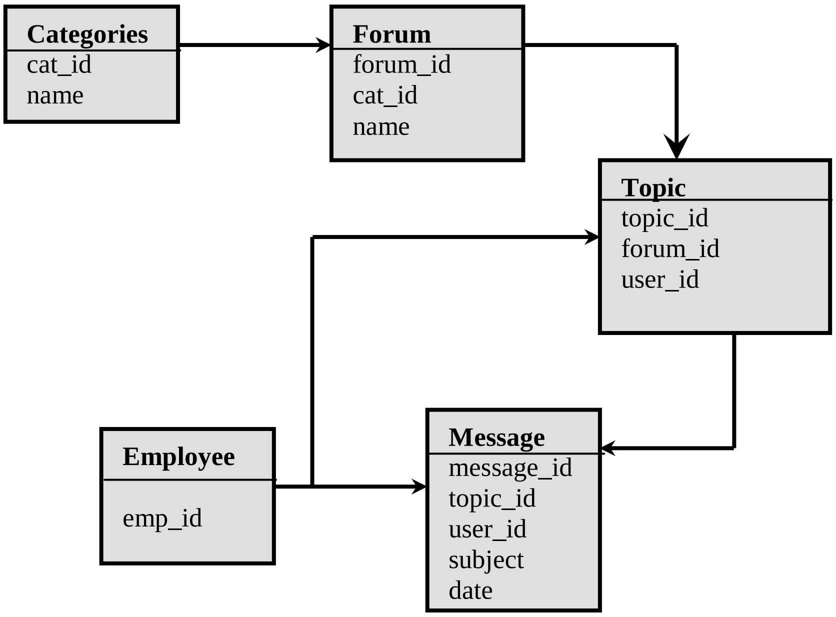


Figure 4.5 Relationship between CoP tables

4.2.2 Data Preparation

Data preparation is an essential step of knowledge discovery to ensure that useful knowledge is derived from the input data.

Data preparation steps are needed to identify noise, missing data, errors and inconsistencies etc. in the raw data, then to take an action like remove, correct or modify this data. On the other hand, this step helps to extract the possible target data subset based on understanding of the users' requirements. In this research, the process of data preparation includes the following steps:

- Data selection

Understanding the application domain and identifying the target of the GKWS user's requirement and selecting the needed dataset that was discussed earlier in section 4.1.1. The application domain is the grinding engineer's database for technicians, planning and quality control engineers, knowledge engineers for inserting the grinding cases and CoP members for exchanging their ideas. The goal of GKWS is to support the decision-making process for selecting grinding conditions and to facilitate knowledge collaboration between CoP members.

- Data pre-processing

Facilitating the data cleaning that consists of removal of noise, handling missing data and collecting needed data. The elimination of irrelevant data can improve the performance of the knowledge discovery process. The collected data in the database usually need further processes to be ready for use in the knowledge discovery module. The data collected for grinding conditions are either numbers (depth of cut, work speed, feed ate etc) or text (material name, wheel name etc). The system cleans the entered numbers by rejecting any entry containing alphabetic or unneeded characters (such as comma or backslashes) by using the following algorithm:

```
$Field_Name=trim($_POST[Form_Field_Name]);

 If (!(empty($Field_Name))){

  If (!(is_numeric($Field_Name))){

  $mistakes[] ='For field: Enter only NUMERIC characters<br>';
        } else{
```

```
//accept roughness entry and sanitize it

$ Field_Name = mysql_real_escape_string(stripslashes($Field_Name));}}
```

A warning message is shown for the user to be aware that there is a mistake or mistakes in the entered field. In this case, the entered number fields are cleansed from any noise (such as slash or comma or letters) and it is ready for further data retrieval. The key grinding conditions assigned indexes during data entry using the intelligent add grinding case form such hardness, roughness, material group, material, wheel abrasive type, wheel bond type in order to be ready for data mining in the future. An example for assigning index for hardness value is shown in the following algorithm:

```
//*******Assign an index for the hardness (primary index)*****************
  if ($_POST[hardness_field_name]== "<50RC") {
    $hardness_index="S";
    //echo"hardness index $hardness_index";
 }else if ($_POST[hardness_field_name]=="<50-58RC"){
     $hardness_index="M";
     //echo"hardness index $hardness_index";
 }else if ($_POST[hardness_field_name]==">58RC"){
     $hardness_index="H";
     //echo"hardness index $hardness_index";
}
```

In this research, the grinding cases are collected from different publications (Lee, Choi and Shin 2003; Choi and Shin 2007; Shin, Subrahmanya and Choi 2008) and theses (Chen 1995; Lee 2000). It was noticed that key data for CBR are missed. The data cleaning has adopted a methodology (which will be discussed in chapter 5) to complete the missing data by using mathematical calculations, rules and generation models.

4.3 Knowledge Storage

This module supports efficient explicit knowledge storage and retrieval in various forms based on the analysis task and provides the decision-maker with all phases of knowledge. The knowledge storage manages the integration of a wide variety of knowledge that includes the transaction data in database, rules in RBR, cases in CBR, mathematical equations, algorithm, text streams from expert system.

The rule base stores the guideline rules to select grinding conditions, validation rules to validate data entry, and rules to fill the incomplete data and knowledge.

The case base contains all grinding cases and the needed parameters for grinding operations. The knowledge base retains the inference knowledge from the problem-solving module, which includes similarity metric, weight metric and equations for modifying the case. The models and equations for calculating incomplete data are retained from the knowledge discovery to the knowledge storage, as shown in Figure 4.6.

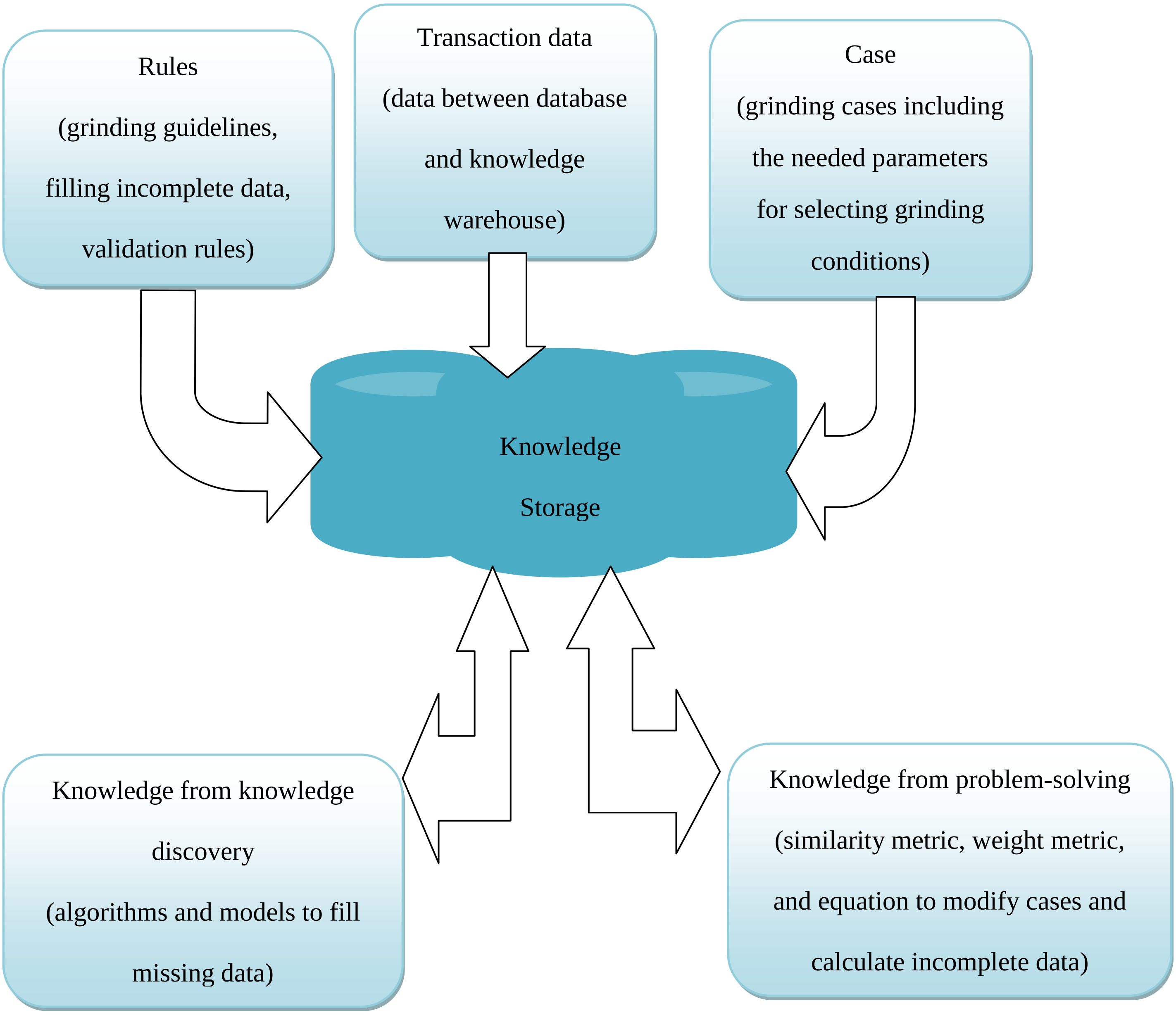


Figure 4.6 Knowledge storage in GKWS

4.4 Application Example

Some user interfaces for data collection will be shown in this section. Figure 4.7 illustrates the home page of GKWS that presents the most functions. Figure 4.8 illustrates the data acquisition intelligent form, which is designed to collect grinding parameters. The form uses a star symbol to indicate the important parameters. The drop down menu allows the user to select the material and then the material group will be selected automatically. The user will be promoted to enter the grinding conditions for work speed, wheel speed, dressing depth, dressing lead, depth of cut, surface roughness, and feed rate. The user can select from a drop down menu for machine name, wheel name, dresser name, and coolant name. If the user can't find the name from the drop down menu then they can add it first using the defined form then complete the grinding case. Figure 4.9 represents the free text data entry for the discussion forum. The user can express their ideas using the text field provided.

Figure 4.7 Home page screen

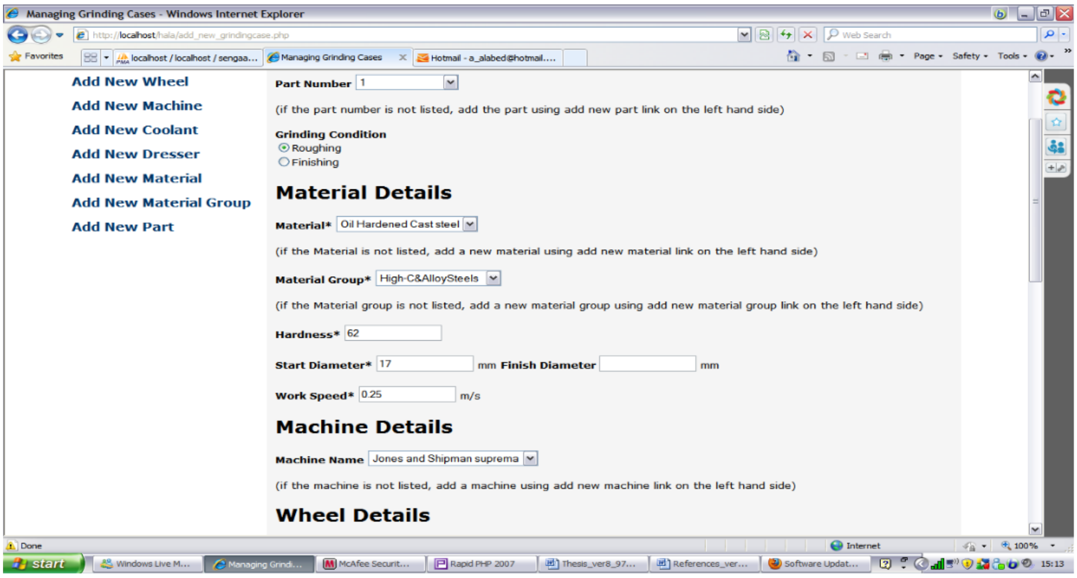


Figure 4.8 Form interface for collecting grinding variables

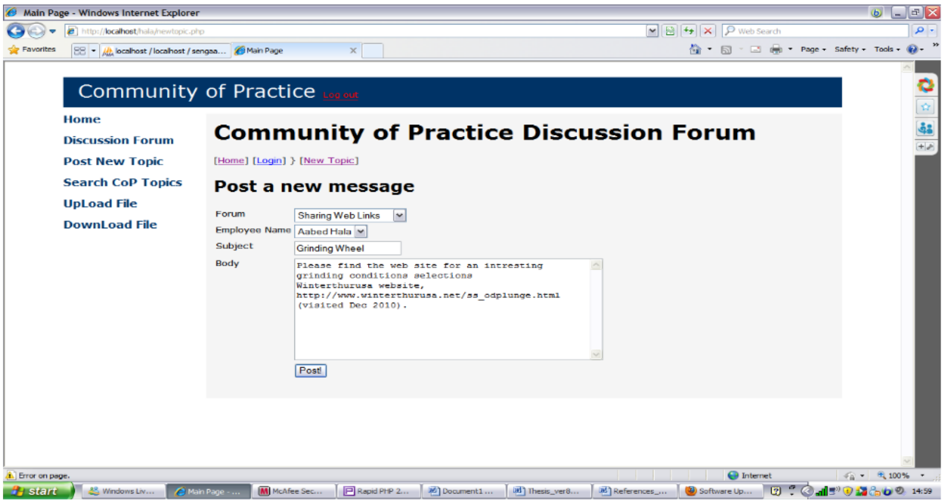


Figure 4.9 Free text data entry

4.5 Summary

Data collection and management stages are discussed in this chapter. Data collection is the first port of call for knowledge management and the discovery process. Therefore, the process to facilitate data collection is described. The required parameters for grinding cases are discussed and summarised. Database module is described including design philosophy and data preparation. The database design and structure are significant for reliable and effective storage and data retrieval. Therefore, the table's relationships and structure are discussed and presented in this chapter. The stored data is now ready for further operations.

CHAPTER 5 KNOWLEDGE ACQUISITION AND PROBLEM SOLVING MODULE

The knowledge acquisition module is designed and developed in this chapter. The core of the problem solving engines includes case-based reasoning (CBR), rule-based reasoning (RBR) and model-based reasoning (MBR). The rule-based and case-based systems are integrated to the grinding knowledge warehouse system (GKWS). Finally, an application example of selecting grinding conditions is demonstrated.

5.1 Knowledge Acquisition Module

Knowledge acquisition is the general name given to the process of eliciting, acquiring, and representing knowledge consisting of descriptions, relationships, and procedures in a specialised domain of interest (Benbasat and Dhaliwal 1989). Its major functions are to extract knowledge from expert(s), and analyse and formalise the knowledge into some computer understandable forms (Shaw and Woodward 1990). Interviews and observation techniques involve directly obtaining knowledge from a domain expert on how they do their job. The success of these methods is dependent on the questions asked, the knowledge engineer’s communication skills, and the ability and personality of the expert to articulate their knowledge (Yang 1995; Benbasat and Dhaliwal 1989; Wenger, Najdawi and Chung 2001). Since these methods rely on human interaction, automatic tools and in the Internet became more popular to acquire and elicit knowledge. Boose (1989) and Molnar and Sharda (1996) discussed the use of the Internet as a knowledge acquisition technique. Molnar and Sharda's (1996) study showed that email and discussion groups can be used for knowledge acquisition.

The knowledge acquisition module in GKWS is primarily responsible for converting knowledge from the tacit to the explicit automatically by:

- directly acquiring and gathering tacit knowledge from knowledge engineers or Community of Practices (CoPs) through the knowledge user interfaces.

- deriving the knowledge by calculations from relevant databases.

This module provides an interface for knowledge engineers and CoP experts to insert questions, browse related knowledge, and search for advice and solutions. The discussion forum is divided into three sub-forums: problem solving, quality control, and selecting grinding conditions. For example, the user can send a question to the CoP's members and, at the same time, the user can search for answers in the GKWS using an available search engine. Also, the user has the choice to go directly to the required link; for example, the user can click directly and post a new topic.

On the other hand, this module facilitates the knowledge acquisition process by advising engineers in selecting the grinding conditions. When a user keys in the input variables for a grinding operation, the system retrieves the most related case by using

CBR or RBR. The acquisition module includes a specified user interface to aid in one or more of the following processes:

- Operational rules election for CBR and RBR

Knowledge acquired through the knowledge acquisition interface is converted into a rule-based if-then format. This type of representation is most common for expert systems. For CBR, the system identifies the indexes for material group, hardness, roughness, wheel parameters using if-then rules; for example, when creating the index for roughness 1.4 micro m, the roughness index is 1;

```
if (($_POST[roughness]>=1.10) && ($_POST[roughness]<= 1.60)){
        $roughness_index=1;
 } else if (($_POST[roughness]>=0.70)&&($_POST[roughness]<= 0.90)){
        $roughness_index=2;
        } else if (($_POST[roughness]>=0.40)&&($_POST[roughness]<= 0.50)){
        $roughness_index=3;
          } else if (($_POST[roughness]>=0.20)&&($_POST[roughness]<= 0.35)){
        $roughness_index=4;
 } else if (($_POST[roughness]>=0.10)&&($_POST[roughness]<= 0.17)){
        $roughness_index=5 ;}
```

The identified primary indexes are taken for further processing to select the applicable cases by activating the following SQL statement:

```
SELECT * From `casesrecord` where `v_index` like '__$hardness_index%' and `v_index` like
'$matgroup_index%' and `v_index` like '%$roughness_index'";
```

The set of the applicable cases are located now and saved into a temporary table for further processing. The secondary indexes are used to calculate similarity value and choose the case with maximum similarity using the following SQL statement:

```
SELECT * FROM copy_casea WHERE sim= (Select Max(sim) FROM copy_casea)
```

For RBR, the rules are constructed using if-then expressions where the grinding conditions are selecting according to the user input.

- Grinding case management

The knowledge gathered from the engineering technicians or knowledge engineer through grinding case management is converted into if-then rules in order to automate the filling of missing or incomplete data using stored equations and models. For example, if the surface roughness is incomplete for material 4140, which is missing, then a model will be activated to predict surface roughness as follows:

```
If (material group == Super Alloys) &&(individual material=4140) && (hardness<51)){

 //the roughness could be calculated using Shin model

$he=1000*[depth_cut]*[work_speed]/[wheel_speed];

 $roughness=1.64*pow([dressing_depth],-0.021)*pow([d_lead],0.385)*pow($he,0.284);

$_SESSION['notify'] .= "The roughness could be calculated using Shin model, It is equal
$roughness <br>"
```

} else {

//if the material is not there then the system could calculate the surface roughness using the GP MataLab environment and the results will not be saved until it is proved by real grinding.

*$_SESSION['error'] .= "Ra could be calculated using Genetic Programming GP (MatLab extension)
";}*

5.2 Problem Solving Module

The core of this module is CBR and RBR. This module includes data transformation, knowledge inference engine and knowledge representation.

The CBR and RBR are based on the blackboard approach for selecting grinding conditions and intelligent grinding database (Li 1996; Chen 1998). CBR is an approach that seeks to identify a close match between a new operation to be performed and the characteristics of a previously successful case stored in a case base. The principle of the CBS is based on the blackboard approach developed by Li (1996), Li, Rowe and Mills (1999) and Li et al. (1999) as follows:

- Assign an index to each of the key features of the case specification. Indexation is employed to establish similarity between cases. The primary indexes are used to select the set applicable of cases. Initially, it is considered to be very important that material should have the same grindability and wheels should have the same abrasive type, bond type and grit size (Li et al. 1999).

- If there is a mismatch in this category, the case will be rejected. The primary indexes are material group, material hardness, surface roughness and wheel specification. Selection of an appropriate case is achieved when the user input specification index is matched with a case in the case base, as an SQL query will be activated.

- If the match were successful for one or more cases, a set of applicable cases would assume to be located from table cases and the details of the case are located from case records table. If not, the CBR process has failed and the program would turn to apply the RBR. The set of applicable cases will be saved in a temporary table for further calculations. That table works as a temporary storage for the applicable cases so the data retrieval and calculations processes will be faster.

- Calculate the similarity value for the retrieved past cases with similar secondery indexes where the case specification is matched with similar cases in the database memory. A similarity metric is proposed to judge similarity between a new case and the set of applicable cases determined by the primary indexes. The nearest case is the case that has the highest similarity value according to the following equation (Li et al. 1999)

$$Similarity = \frac{\sum_{i=1}^{n} Weight_i \times sim_i}{\sum_{i=1}^{n} weight_i} \qquad (5.1)$$

Weight represents the importance of each secondary parameter and the complexity of the modifications. The more important parameter has the higher value of the weight.

Sim_i represents the similarity for the secondary indexes. For the individual material, if the material is the same is defined as 1 otherwise sim(material)=0. For the rest of the secondary indexes Sim_i is calculated using:

$$Sim_i = 1 - \left(\frac{applicablecase_i - inputcase_i}{range} \right)^2 \tag{5.2}$$

The secondary indexes are used to choose the nearest case from those applicable cases determined by the primary indexes. If there is a mismatch of this category the case could be modified to fit the problem. The secondary indexes include individual material, roughness value, work diameter, wheel speed, and wheel diameter these represent the machining requirements and control parameters (Li et al. 1999).

- Adapt the case from the memory to match the new case specification. In the instance when similar cases are identified the nearest case is adopted. In most situations, the case retrieved would not exactly fit the case specification definition. The case has to be modified to confirm to the new requirements. Modification of a case is based on the following assumption:

    - For the same material and roughness groups, the wheel, fluid and wheel speed need not change.

    - The dressing lead is the most sensitive parameter for the surface roughness. This factor needs to be modified to satisfy the required surface roughness.

$$f_{dn} = \frac{f_{do} R_{an}^2}{R_{ao}^2} \tag{5.3}$$

- For changes in the workpiece diameter from diameter in an existing case to workpiece diameter in the problem, the feed rate is changed according to the equation 5.4.

$$v_{fn} = v_{fo} \frac{d_{wo} v_{sn}}{d_{wn} v_{so}} \tag{5.4}$$

- For changes in equivalent diameter from diameter in the existing case to the diameter in the problem, the work speed is changed to:

$$v_{wn} = v_{wo} \frac{d_{en} v_{sn}}{d_{eo} v_{so}} \tag{5.5}$$

Where $d_e = \dfrac{d_s d_w}{d_s + d_w}$

The modification of the value of dressing feed is not an exact process due to the uncertainty introduced by the shape of the dressing diamond, which will change with wear. However, with a little modification, the method is feasible. The flow-chart summaries the process, as shown in Figure 5.1.

- Test the new case. If the test is successful the output will be stored in the case and case records and the records in the temporary table will be dropped. If unsuccessful, the case is further modified and the loop repeated until success or the case has been exhausted.

The rules, expressed in Boolean, were developed from data that is given in tables such as those found in Machining Data Handbooks (Grinding Data Book 1992; Machinability Data Center 1980). The data for wheel specification and the value of dressing parameters were from The Grinding Data Book (Universal 1992). The data for values of the grinding parameters were from The Machining Data Handbook (Machining Data Handbook 1980). The RBR is activated when the CBR fails to find a matched case because there is not any matched case in the database system or very important indexes are missing. The recommendations for cylindrical grinding with conventional wheels are sorted into tables. The rules here are applied for selective materials, which are tool steel, cast iron, and super alloys. The data in the table can be encoded into a rule (see Appendix A for more details). An example of the used rule is as follows:

Rule 3

IF	workpiece material is cast iron
AND	material hardness > 50 Rc
AND	wheel is conventional wheel (C46MV)
AND	roughness < 0.8μm
THEN	wheel speed=28-33m/s
AND	work speed= 0.35-0.5m/s
AND	feed rate $\leq 0.013/2 \times 0.35 \times 1000(\pi \times d_w)$
AND	dressing depth= 0.012-0.019 mm
AND	dressing lead = 0.1 mm/r
AND	Coolant= Emulsifiable oils- light duty / Chemical and synthetics-light duty

The surface roughness model is calculated according to Equation 5.6 (Li 1996):

$$R_a = 0.821 a_d^{0.113} f_d^{0.290} [\mu m] \qquad (5.6)$$

Where a_d is dressing depth in mm

And f_d is dressing lead mm/rev.

The power model is calculated using Equation 5.7 (Chen et al. 1999):

$$P^{'} = 7.38 a_d^{-o.358} f_d^{-0.068} \text{ [W/mm]} \tag{5.7}$$

Where a_d is dressing depth in mm

And f_d is dressing lead mm/rev.

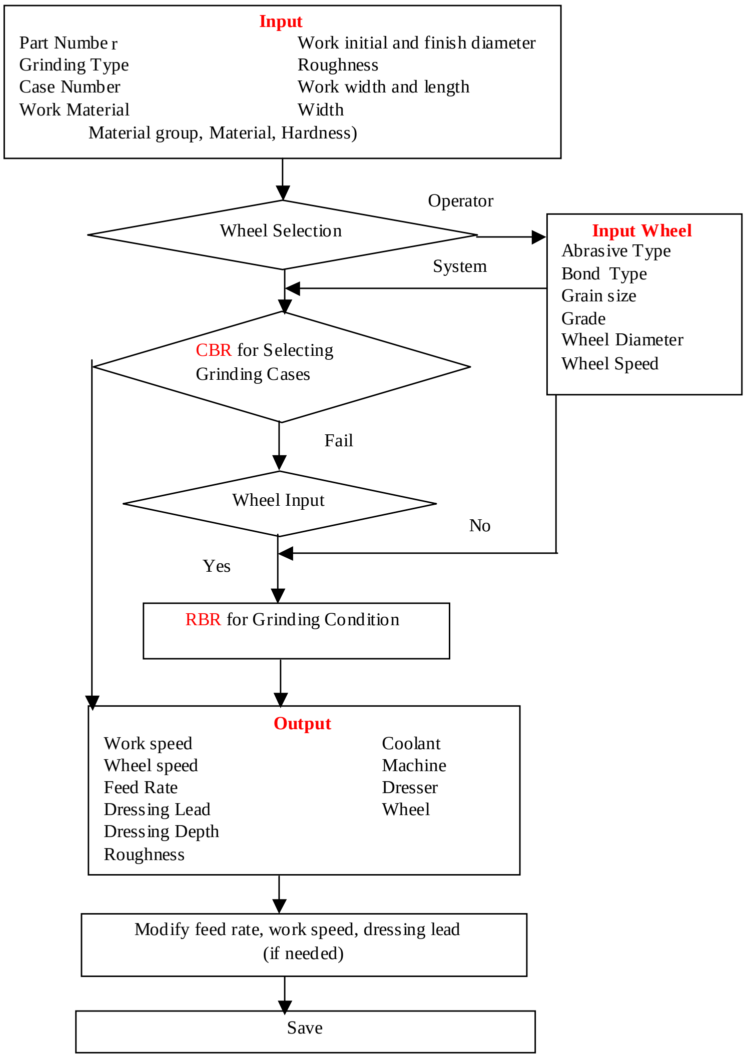


Figure 5.1 CBR and RBR flow chart

5.2.1 Inference Engine

The inference engine interprets the knowledge in the knowledge base, performs logical deduction and contains knowledge base modifications. The inference engine provides the reasoning strategy for searching the knowledge base to determine which rules apply to the situation and make the appropriate decision (Ranky 1990; Salmon 1992). Inference engines are forward chaining (data driven or antecedent reasoning) and/or backward chaining (goal driven or consequent making). Forward chaining is also called data driven inference mechanism. It starts with the available information as it is received and is trying to draw conclusions that are appropriate to the set goals.

Back word chaining or goal driven inference mechanism starts by specifying a goal. A hypothesis of how the specified goal to be achieved is established and the system backtracks through the rules to find evidence to support the hypothesis**.**

The approach adopted for selection of grinding conditions was the forward chaining reasoning. The basic reasoning method is a pattern-matching algorithm. In a predetermined order, the condition portion of the rule is compared with the current state of facts. When all the conditions of a rule are satisfied, then that rule becomes eligible for execution.

The rule-based system is automatically activated when the system fails to find a matched case. The minimum parameters are roughness, hardness, material and start diameter. If any of this data is missing, the system won’t be activated.

5.2.2 Application Example

GKWS supports the decision-making process for selecting grinding conditions for a newly machined part or to retrieve the grinding conditions for previous grounded part. The details for previous machined parts are stored in the database, so the retrieval for the grinding parameters is straightforward. The user can choose one or more criteria to view a pre-grounded part such as part number, surface conditions, wheel or (and) material, as shown in Figure 5.2. If the selected criteria are looking for material SCM435 and Wheel A60L8V:

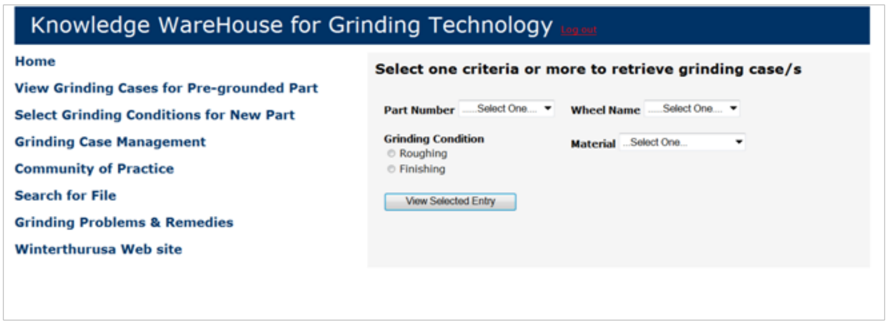


Figure 5.2 Old part key in screen

If the part has been machined before and the data is available in the database, the grinding parameters will show up, as seen in Figure 5.3.

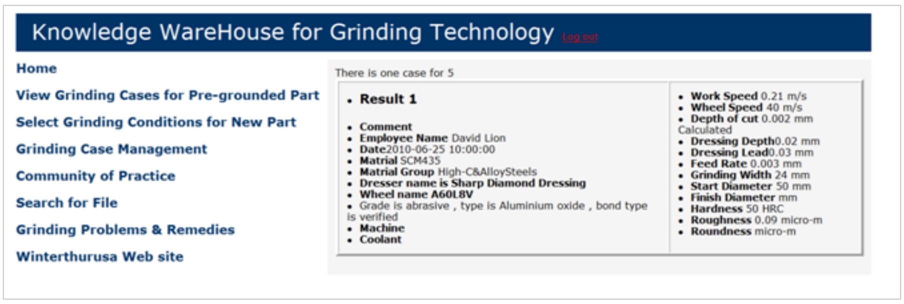


Figure 5.3 Retrieved grinding conditions for previous machined part

The new screen will show up if the part is new to allow the user to enter the key information, such as workpiece details (11060, hardness 62HRC, diameter is 20mm), where roughness is 0.3 micro m, work speed 0.6 m/s and wheel to be selected by the system as seen in Figure 5.4.

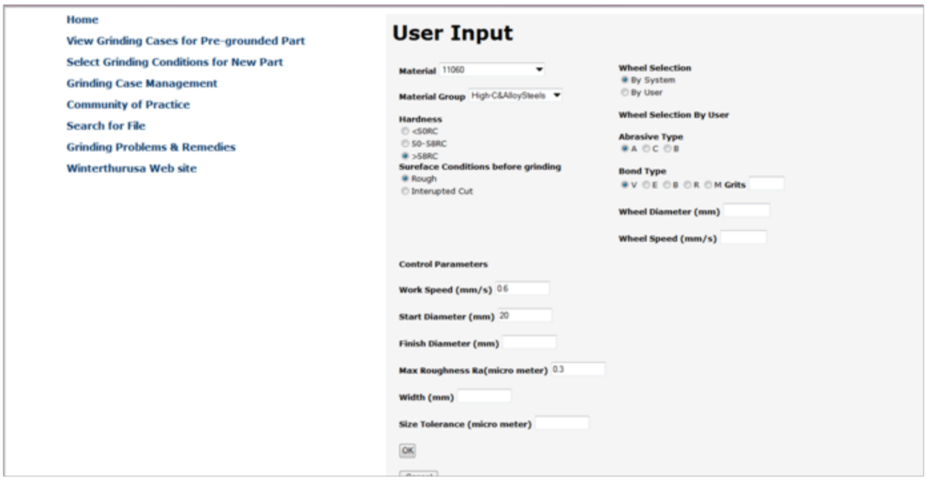


Figure 5.4 User Input Screen

The minimum data required for the case-based reasoning system to run are material, material group, hardness, roughness, and start diameter. If any of this data is missing the cased based system won't be able to run. The recommended case is located and modified, as shown in Figure 5.5.

Figure 5.5 Grinding conditions using CBR

The rule-based system is automatically activated when the CBR system fails to find a matched case. For example, the material is ASTM A3 from the Cast Iron group, hardness is between 50-58 HRC, roughness is 0.6micro-m and diameter is 50mm, as shown in Figure 5.6. The recommended case is selected using RBR is shown in Figure 5.7.

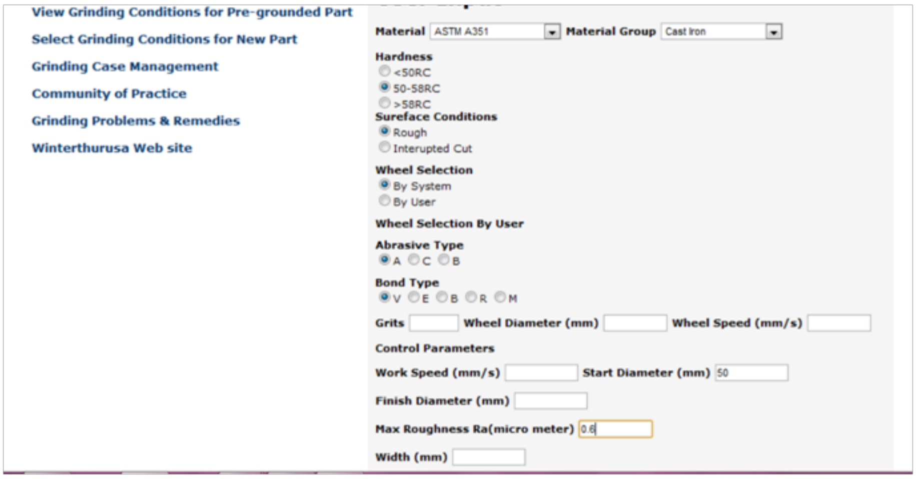


Figure 5.6 Input user interfaces for Cast Iron material

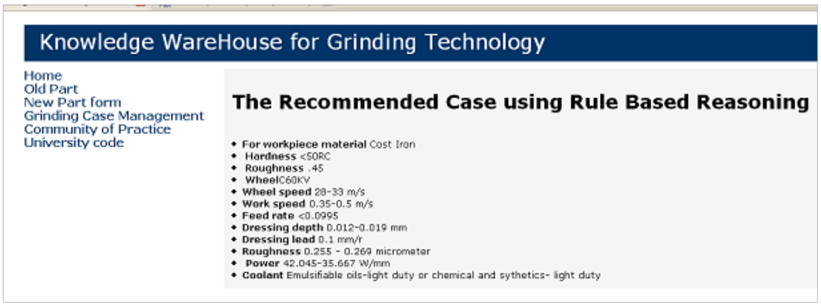


Figure 5.7 Recommended case using RBR

5.3 Summary

The selection of the grinding conditions in GKWS is based on AI techniques and historical saved information. A combination of CBR and RBR are employed to select grinding condition and update the knowledge base. The flow chart in Figure 5.8 summarises the main functions of the case and the rule-based system when the wheel is selected by the system. Figure 5.9 demonstrates the main functions of CBR and RBR if the user selects the wheel. When the wheel is selected by the system, the primary indexes are material, material group, hardness, roughness and important parameters are start diameter, and work speed. When user selects the wheel, the secondary indexes are material, material group, hardness, bond type, grits size, abrasive type and important parameters are individual material, roughness, workpiece diameter, wheel speed and wheel diameter.

The primary indexes are necessary to identify the applicable cases therefore if there is a mismatch the case will be rejected. The secondary indexes are used to determine the most similar case. The rule-based system will be activated when CBR fails to locate a similar case.

A fundamental characteristic of the CBR system is the requirement for sufficient cases to be saved in the database to cover the target specification. If insufficient cases are available for a search the system will retrieve misleading results. In other word, the main input parameters in CBR that should be entered by users are material group, material, wheel name, diameter, material hardness, wheel diameter, wheel speed, workpiece speed and surface roughness. The retrieved cases should have the same data available otherwise the case-based system won't be very accurate and the results are misleading. For example, if the input surface roughness value is missed from the recorded case in the database, taking into account that it is one of the very important parameters, no results will be matched with a case in the database, so the result will reject the CBR and activate the RBS. In this research, the system will ensure that the required parameters for activating CBR are not missed or incomplete in the saved recorded cases, in order to improve the performance and effectiveness of the CBR, which will be discussed in detail in Chapter 6.

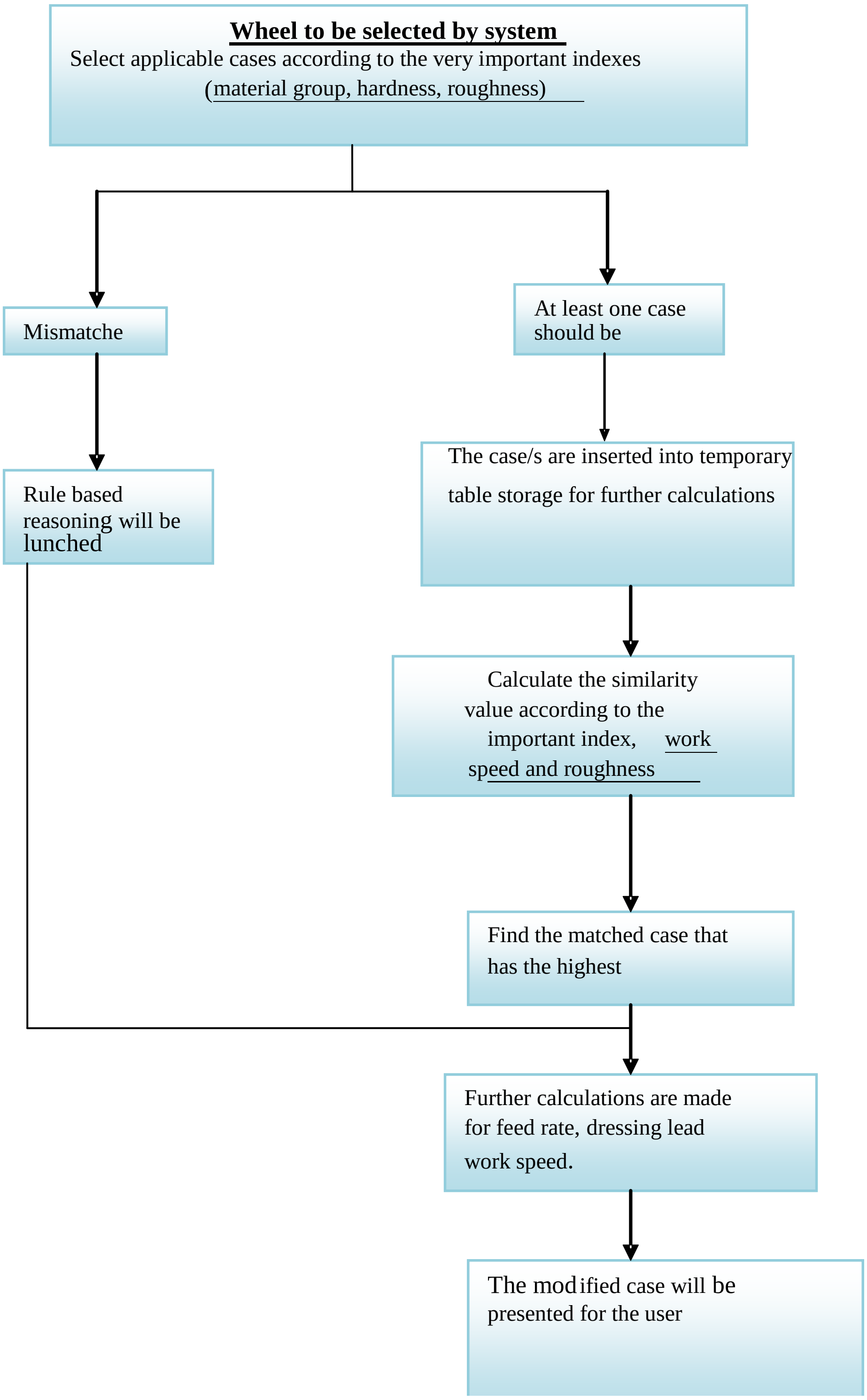


Figure 5.8 Flow chart for CBR and RBR in GKWS where the wheel is selected system

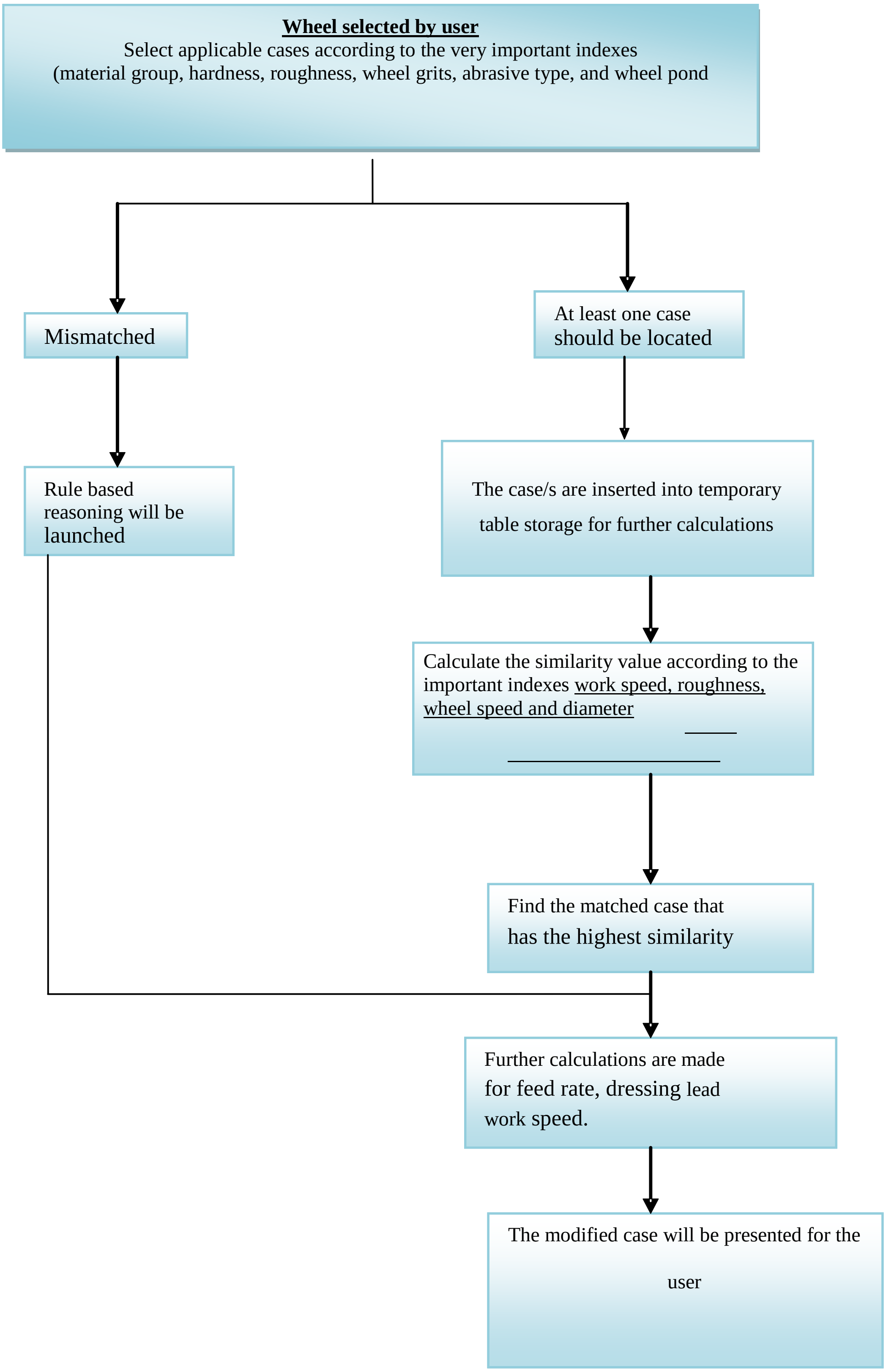


Figure 5.9 Flow chart for CBR and RBR in GKWS where the user selects the wheel

CHAPTER 6 LEARNING KNOWLEDGE DISCOVERY

This chapter describes the major contribution of this thesis, which is using genetic programming (GP) in modelling surface roughness and dealing with missing data. This chapter discusses further stages of learning knowledge discovery, which includes rule-based learning in the CoP forum and data mining (DM) using GP. The issue of missing data is explained and briefly reviews methods for incomplete data. GP model development for surface roughness is discussed and explained in detail in this chapter. At the end, the results of GP surface roughness are discussed.

6.1 Knowledge Learning and Discovery Module

The knowledge learning and discovery module extracts implicit, previously unknown and potentially useful models and patterns, and then modifies and updates the existing models and patterns. This module provides grinding technologists with the guidance to select the right conditions for a specific grinding operation and dealing with missing data. On the other hand, it will utilise and save the time for the members of CoP by providing them with the most relative answer for their questions using an integrated search engine.

6.1.1 Rule-based Learning in the CoP Forum

The GKWS allow users to share and exchange their knowledge using a *Community of Practice Link*. On the other hand, it could facilitate transferring tacit knowledge to explicit knowledge. The knowledge is embedded in their general discussions, questions and answers, passing on important web links, storytelling, best practice and calling for event or conferences, as concluded from section 3.1.2. Users build up the knowledge, which means their participating and exchanging knowledge on the discussion forum will increase the knowledge in the knowledge storage repository. The user can participate by posting new topics or replaying a specific topic. For example, users can call for China International Abrasives & Grinding Technology Development Forum 2011 by posting the topic using Conference and event forum, as shown in Figure 6.1.

Figure 6.1 Posting a new topic in the conference and event forum

The GKWS allow users to extract and retrieve the needed knowledge using the CoP forum. The users can directly browse the needed knowledge using the categorised option in the discussion forum, as shown in Figure 6.2.

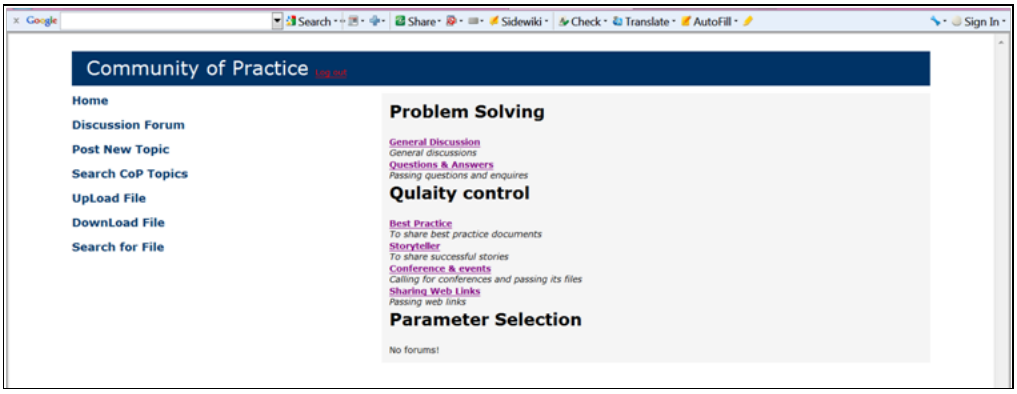


Figure 6.2 Categorised CoP forums

However, the user can extract the required knowledge using the search engine provided. The search engine is designed to use rules:

IF (causes) → Then (effects)

The user can look for the knowledge inside the forum by entering a key word and GKWS will retrieve the related results from the saved knowledge in the CoP database, which will save the time and effort for the users. The following algorithm will check for the search parameters:

```
if (!isset($keywordvar))

{  echo "<p>We don’t seem to have a search parameter!</p>";

  exit;}
```

Then, the SQL statement is built to retrieve the results of the searched keyword from different discussion forums:

$query = "SELECT * FROM `topic` WHERE `subject` LIKE '%".$trimmed."%'";

$numresults=mysql_query($query);

$numrows=mysql_num_rows($numresults);

For example, if the search keyword is *grinding wheel*, as shown in Figure 6.3:

Figure 6.3 Searching CoP topics

Then the related results will show up as shown in Figure 6.4. The results are combined from different forums. This case demonstrated that the forum provides a platform for questions and answers to enable a decision to be taken. Some tacit knowledge is embedded in these conversations.

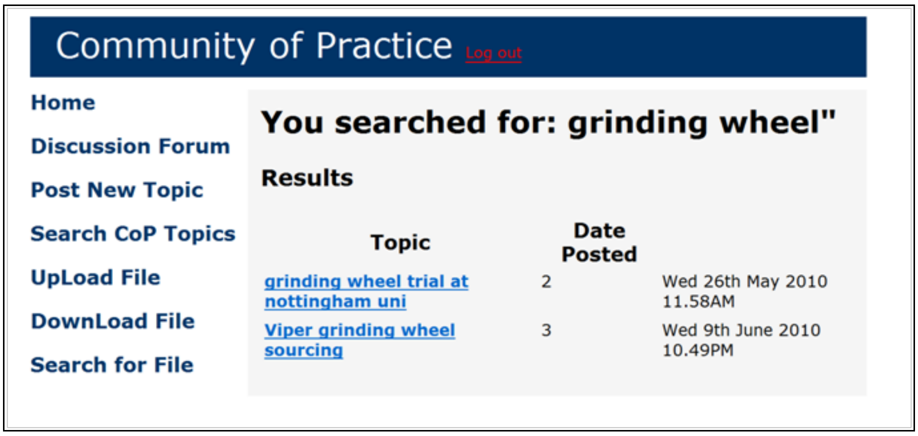


Figure 6.4 Searching results for "grinding wheel"

The users can browse the topics or add to the topic by using the reply link; for example, clicking on the first link "*grinding wheel trial at Nottingham uni*" will show up more details about the topic, as shown in Figure 6.6.

Figure 6.5 Detailed results for "*grinding wheel trial at Nottingham uni*"

On the other hand, the GKWS generated a search engine to look for the remedies for grinding technology. The knowledge engineers are responsible for updating and adding on new rules and knowledge. The grinding technicians can browse and search for remedies for their problems. The problems are categorised into different categories that will make it easier for the user to access their knowledge directly, as shown in Figure 6.6.

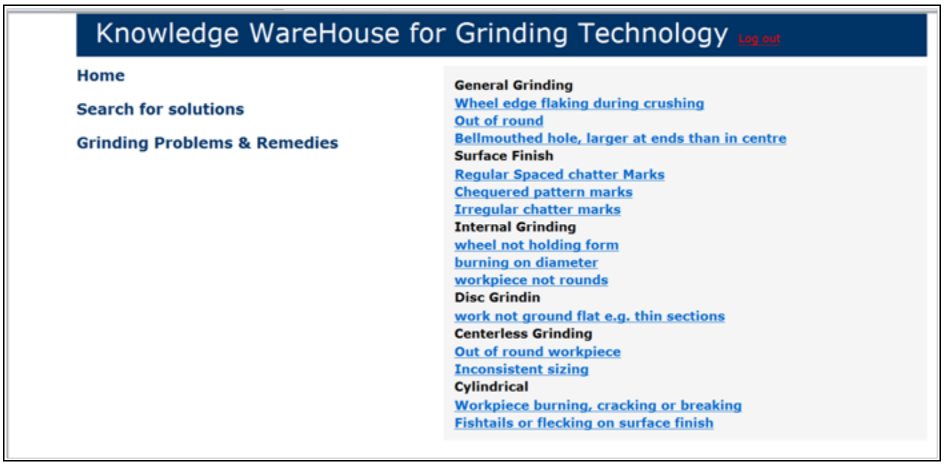


Figure 6.6 Problems and remedies main page

However, the users can search for specific problem symptoms using the search engine. The algorithm adopted for the search engine is

IF (causes) $\longrightarrow$ Then (effects)

The algorithm works as follows:

- check for the searched symptoms variable
- trim the white spaces from the search variables
- build SQL Query to look for a solution to the problem identified
- since Google is a reliable and fast search engine, it would be offered as an *alternative* if there were no results in the database

- display the results including symptoms, causes and remedies for the searched symptoms variable

For example, if the symptom is “burning”, the user will key it in the search field, as shown in Figure 6.7.

Figure 6.7 Problems and remedies search engine

The symptom, causes and remedies results will show up for the searched symptoms, as shown in Figure 6.8.

Figure 6.8 The results for symptoms search

The user can upload documents, images and pdf files using GKWS, as shown in Figure 6.9.

The user can look for the saved documents using related keywords by clicking on the search for file link; for example, if the keyword is grinding, as shown in Figure 6.10.

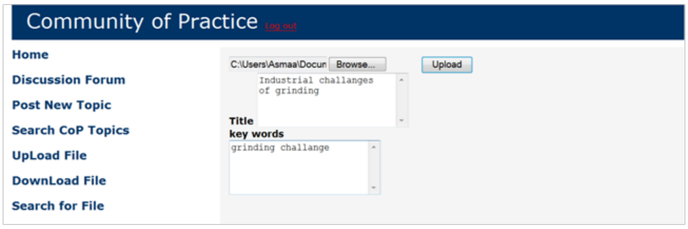


Figure 6.9 The uploading file user interface

Figure 6.10 Search for files user interface screen

The results will show up as seen in Figure 6.11.

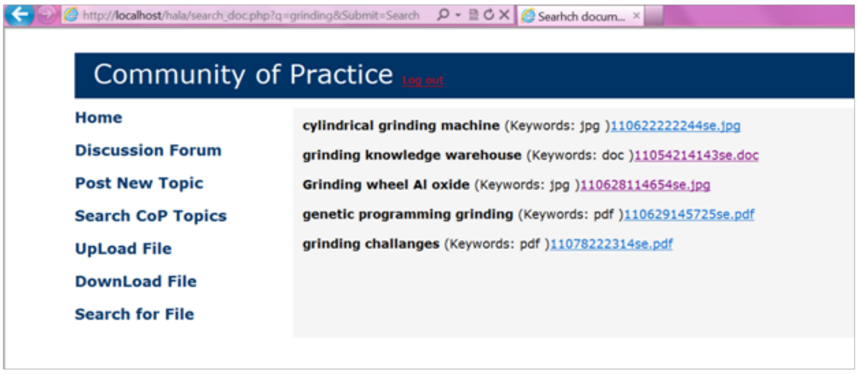


Figure 6.11 The files that contain grinding

6.1.2 Data Mining in GKWS

For CBR, initial grinding cases are needed to cover the grinding operation problem. In order to increase the flexibility, accuracy and efficiency of the CBR, primary and secondary indexes should be recorded. Ground surface roughness value is a critical parameter to calculate similarity for selection of an applicable case in CBR.

Most grinding cases found in publications do not provide a complete grinding case (Li 1995; Li et al. 1999; Morgan et al. 2007). The causes of missing data are often numerous, some due to design, and some due to chance. Some variables may not be collected from all grinding experiments. On the other hand, it could be that the researches depend on data collected from their experiments ignoring records with missing data or designing experiments according to their criteria. It could possibly be because researchers were concerned with a specific aspect of the grinding process rather than the whole grinding problem. For a grinding operation, grinding depends on large number of variables so different experiments focused on different parameters. For example, Choi et al. (2008) designed experiments for wheels where the recorded variables were dressing feed, dressing depth, wheel speed, work speed, infeed and equivalent chip of thickness, while a designing experiment for modelling surface roughness are wheel diameter, work diameter, dressing lead, dressing depth, wheel speed, work speed, depth of cut, and equivalent chip thickness.

For this research, grinding cases for cylindrical and surface grinding are collected in order to create a board picture of grinding operation. On the other hand, cylindrical and surface grinding has many parameters in common. There are many similar features between them. The data was collected from different articles (Lee, Choi and

Shin 2003; Choi and Shin 2007; Shin, Subrahmanya and Choi 2008) and theses (Chen 1995; Lee 2000). In this research the name of the dataset is written in the following format to simplify referencing dataset (operation type experimental/calculated surface roughness dataset (number)), as shown in Table 6.1. If the surface roughness value is calculated value then cal. word will appear before dataset name, as shown in Table 6.1. If the surface roughness value is experimental value then exp. word will appear before dataset name, as shown in Table 6.1.

From the collected cases presented in Table 6.1 it could be concluded that:

- The available grinding cases have not recorded the same set of input and output parameters.
- The primary indexes for CBR are available (represented in light grey in Table 6.1) in all datasets.
- The secondary indexes for CBR, which are highlighted in dark grey, are available except the experimental value for surface roughness in cylindrical cal. Dataset (2) and surface cal. dataset (2) (Choi and Shin 2007; Shin, Subrahmanya and Choi 2008).
- There are two datasets for cylindrical: cylindrical exp. dataset (1) and cylindrical cal. dataset (2) (Chen 1995; Shin, Subrahmanya and Choi 2008). The surface roughness value is experimental in cylindrical exp. dataset (1) dataset while in cylindrical cal. dataset (2) surface roughness values are calculated using the empirical model in Shin, Subrahmanya and Choi (2008).

An assumption is made that his calculated value is equal to the experimental surface roughness value.

- For surface grinding, two datasets are collected; surface exp. dataset (1) and surface cal. dataset (2) (Lee 2000; Choi and Shin 2007). The surface roughness value is experimental in surface expl. dataset (1) while surface cal. dataset (2) has calculated surface roughness value using the empirical model in Lee, Choi and Shin (2003). An assumption is made that his calculated value is equal to the experimental surface roughness value.

Table 6.1 Cylindrical and surface grinding datasets

	Cylindrical grinding		Surface grinding	
Reference	**(Chen 1995)**	**(Shin, Subrahmanya and Choi 2008)**	**(Lee 2000)**	**(Choi and Shin 2007)**
Name	Cylindrical exp. dataset (1)	Cylindrical cal. dataset (2)	Surface exp. dataset (1)	Surface cal. dataset (2)
Number of cases	17	21	21	20
Wheel	A465-K5-V30W	32-60-K-VBE	38A60K5VBE	38A60K5VBE
Machine tool	Jones and Shipman Series 10 and	Supertec G20-50CNC	Stanko 3G71	Mazak CNC machining centre
Material group	HS	LS	LS	LS
Hardness	62	40	50	50
Roughness	Experimental	Calculated from Model	Experimental	Calculated from Model
Work diameter	√	√	√	√
Work material	Oil hardened steel	4140	4140	4140
Wheel speed	√	√	√	√
Wheel diameter	√	√	√	√
Dressing depth	√	√	√	√
Dressing lead	√	√	√	√
Feed rate	√	-	-	√
Volumetric removal rate	√	-	-	-
Dressing direction	√	-	-	-
Number of dressing passing	√	-	-	-
Roundness	√	-	-	-
Size holding	√	-	-	-
G-ratio	√	-	-	-
Re-dress life	√	-	-	-
Dresser speed	√	-	-	-
Depth of cut	$a = \pi d_w v_f / v_w$	√	√	√
Specific energy	√	-	-	-

In the collected cases, the roughness value and material group value were the most likely to be incomplete and missed. For the material group parameter, it could be identified if the work material is known (Li 1996). Surface roughness value could be calculated using available surface roughness models.

In GKWS, a combination of CBR and RBR are employed to select a grinding condition and update the knowledge base. A fundamental characteristic of the CBR system is the requirement for sufficient cases to be saved in the database to cover the target specification. For a successful search, if insufficient cases are available, and the results of CBR will be inaccurate and misleading, then the system will activate RBR procedures.

In the CBR, the main input parameters that should be entered by users are the primary and secondary indexes; material group, hardness, wheel specifications are considered as primary and individual material, work diameter, wheel diameter, wheel speed, and surface roughness are considered as secondary indexes. The primary indexes are for identifying the applicable cases in CBR. If at least one of these parameters is missing in the saved cases in database, the CBR will be terminated. The secondary indexes are used to select the most similar case from the applicable cases. If one of the parameters is missing in the database, the results of CBR will not be realistic. Some parameters are necessary to modify the most similar case, such as feed rate, work speed and dressing lead while the retrieved cases should have these parameters available, otherwise the case based system won’t be very effective and the results are misleading.

In other words, the CBR would be more efficient if the work material, wheel parameters, hardness, roughness, work speed, work diameter, wheel speed, wheel diameter, dressing lead, dressing depth and feed rate value are available in the saved case.

As discussed in chapter 4 section 4.1.2, surface roughness is considered as an important grinding variable to measure surface accuracy and precision. Surface roughness describes the quality of grounded part, more precisely the geometry of grounded surface. On the other hand, surface roughness is one of the secondary indexes in selecting grinding conditions and calculates similarity from the CBR and RBR in GKWS, so surface roughness value should not be missed or incomplete in the database.

In data collection from real industry, experimental roughness value wasn’t published (in Choi and Shin 2007; Shin, Subrahmanya and Choi 2008) while the required parameters for activating the CBR are all recorded. As it has been explained earlier, the surface roughness value could be calculated using empirical models proposed in Shin, Subrahmanya and Choi (2008) and Lee, Choi and Shin (2003).

For the collected cases, two different grinding models were recognised for cylindrical and surface roughness grinding, which was used to calculate the roughness value. The generalised surface roughness model for cylindrical plunge processes material steel 4140 and wheel 38A60K5VBE is given by Shin, Subrahmanya and Choi (2008):

$$R_2 = R_a = 1.64 a_d^{-0.021} s_d^{0.385} h_{eq}^{0.284} \tag{6.1}$$

Where equivalent grinding thickness can be calculated using Equation 6.2:

$$h_{eq} = \frac{Q^{'}}{v_s} \tag{6.2}$$

$$Q^{'} = v_w a \tag{6.3}$$

The generalised surface roughness model for surface grinding processes is given by Lee, Choi and Shin (2003) in Equation 6.4:

$$R_3 = R_a = R_{a,0} = 12.9 s_d^{0.541} a_d^{0.34} (\frac{v_w}{v_s})^{0.38} (\frac{s_t}{b_s})^{0.43} \tag{6.4}$$

6.1.3 Methodology of Dealing with Missing Data

In order to tackle a missing data problem, the GKWS will be able to complete the missing data as soon as the case has been filled. Firstly, the system will send a warning message to users if the input parameter is not numeric or has unexpected characters. If the parameter couldn't be calculated, such as wheel, coolant, and start diameter, the GKWS will send a warning message to prompt users to complete the records. If the parameter can be calculated, the system will go to the next step.

Secondly, the system will try to fill the incomplete parameters that can be calculated and display it for the user to agree on these calculated variables. The depth of cut and feed rate will be saved automatically while the calculated surface roughness value will be saved after experimental approval. The system will complete the missing data using:

- mathematical equations for calculating depth of cut and feed rate etc.

- available models for surface roughness value,

- or/and available IF-Then rules as shown in Figure 6.12.

For example, in the data collected from Chen's (1995) thesis, the depth of cut was not recorded; in this case it was calculated using Equation 6.5:

$a = \pi d_w v_f / v_w$ Equation (6.5)

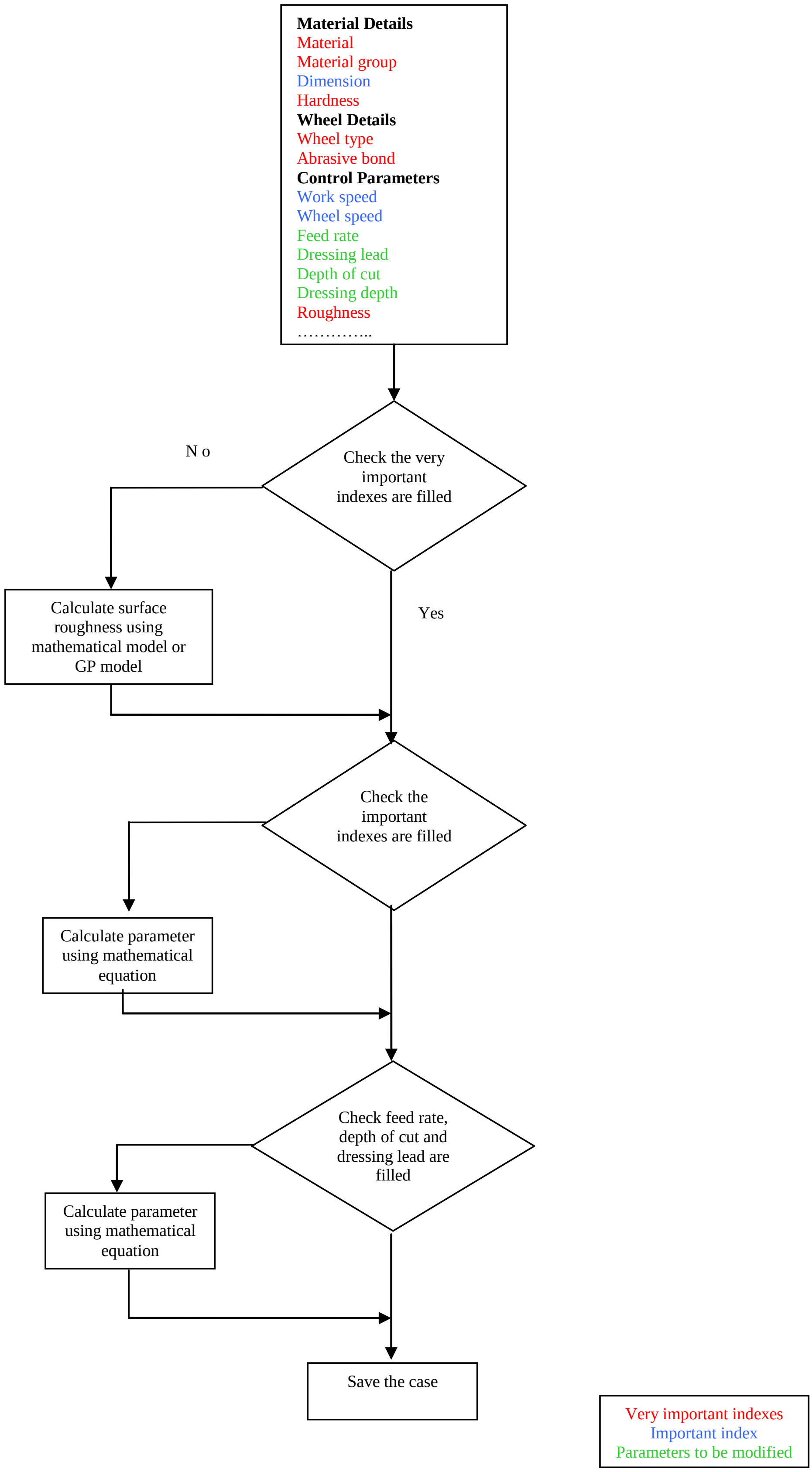


Figure 6.12 Methodology for filling missing data

6.1.4 Mathematical Equations

Mathematical equations are available to calculate some parameters in grinding, such as depth of cut, feed rate, volumetric removal rate, equivalent chip thickness and average chip removal rate. Table 6.2 shows some mathematical equations for cylindrical grinding.

Table 6.2 Mathematical equations

	Cylindrical grinding
Depth of cut	$a = \pi d_w v_f / v_w$
Volumetric material removal rate	$Q' = v_w \pi d_w$ $Q' = \pi d_w v_f b$
Equivalent grinding thickness	$h_{eq} = \frac{Q'}{v_s} = \frac{v_w . a}{v_s}$
Average chip removal rate	$Z_w = \pi d_w v_f$
Equivalent wheel diameter	$D_{eq} = d_s d_w / (d_s + d_w)$

The algorithm adopted in GKWS is established by using IF-Then rules and mathematical equations. When depth of cut or feed rate is calculated, their values will be identified for users to be aware that the value is not an experimental one. The algorithm is responsible for calculating depth of cut and feed rate.

1. If the feed rate value is missing and depth of cut, work speed, and start diameter are available, the feed rate will be calculated using Equation 6.7:

$$v_w = \pi . d_w . n_w \longrightarrow v_f = a . n_w \longrightarrow v_f = \frac{a v_w}{\pi d_w} \qquad \text{Equation 6.7}$$

If depth of cut or start diameter or work speed is missing, the next step will be triggered.

2. If the feed rate is missing and work diameter and material removal rate is available, it will be calculated using Equation 6.8:

$$Q' = \pi d_w v_f b \longrightarrow v_f = \frac{Q'}{\pi d_w b} \qquad \text{Equation 6.8}$$

The calculated feed rate could be used for further calculations in step (3).

3. The feed rate value could not be calculated if the depth of cut, work wheel and start diameter are missing or volumetric removal rate and wheel width are missing. The if loop will be terminated as follows:

4. If the depth of cut is missing and feed rate, work speed and start diameter are available, the depth of cut will be calculated using Equation 6.9:

$$a = \frac{\pi d_w v_f}{v_w} \qquad \text{Equation 6.9}$$

The calculated feed rate in step (2) could be used to calculate depth of cut if the feed rate is not recorded.

The if loop will be ended if feed rate, volumetric removal rate, work speed, or diameter is missing.

6.1.5 Genetic Programming Model

In the case where surface roughness value is not recorded or missed, the system will look for available empirical models. If there is no model in the system for the range of concern, a GP model will be adopted to fill the missed value.

6.1.5.1 Genetic Programming (GP)

In AI, GP is an evolutionary algorithm-based methodology inspired by biological evolution to find computer programs that perform a user-defined task. It is a specialisation of genetic algorithms (GA) where each individual is a computer program.

GP was selected method to deal with knowledge mining for missing data because of the following reasons:

- GP could produce an automatic computer program that can produce an output for a given set of input. On the other hand, GP can automatically create a general solution to model surface roughness problem in a form of a parameterised tree (Koza, 1992).

- The GP candidate solution usually includes both data and functions where terminals represented the variables and constants and functions represented the functions (Maimon and Rokach 2005). In other words, the GP solutions can represent the relationship between the dressing depth, dressing lead, speed ratio, geometrical contract length and equivalent chip thickness variables using functions for example the division, multiplications, and power.

- One of the unique capabilities of GP is its built-in power to select significant variables and gradually omit the variables that are not relevant while evolving the models (Vladislavleva et al. 2010).

- GP is the strongest argument for using symbolic regression on sparse data sets where symbolic regression can handle dependent and correlated variables and automatically discover various appropriate and diverse models (Vladislavleva et al. 2010). That would be helpful since grinding operations depend on large number of variables and there is a complex relationship between them as it was discussed earlier in chapter 2.

GP is developed to combine different surface roughness models into a generic model that is applicable for all conditions with minimum error. GP is based on the Darwinian theory of survival of the fittest. The first step is to create a random population then assess the individuals for their fitness. Having applied fitness function to all the individuals in the initial random population, the evolutionary process starts. The new population will be formed by crossover or mutation and so on. The process will be terminated when the maximum number of generation is reached or when specific performance criterion is achieved.

The GP steps are:

- Determining the set of terminals that could find system variables or constants.

- Determining the set of functions; that is, all possible functions that can be composed recursively from the available set of functions.

- Determining the fitness measure where each individual in a population is assigned a fitness value as a result of its interaction with the environment.

- Determining the parameters for the run, such as population size, number of generations, and minor parameters.

- Determining the method for designating a result and the criterion for terminating a run. The algorithm can be terminated when either a specified total number of generations have been run or when some performance criterion is satisfied.

6.1.5.2 Selecting GP terminals

A literature survey was performed and selected representative models for surface roughness, which are summarised in Table 6.2 and Table 6.3 for surface and cylindrical grinding respectively. The common variables among those models were identified as significant parameters affecting grinding surface roughness, which can be summarised as follows:

Single variables $a_d, d_g, s_d, v_w, v_s, D_{eq}, d_w, d_s, a$

Product variables $a.d_e$

Ratio variables $\frac{v_w}{v_s}, \frac{a}{d_s}, \frac{a_d}{s_d}$

Mixed variables $\frac{a.v_w}{v_s}$

Among these variables, some have been already identified as physical meaning as follows:

$\sqrt{a.d_{eq}}$ geometrical contract length (mm)

$d_{eq} = \frac{d_s.d_w}{d_s + d_w}$ equivalent diameter (mm)

$q = \frac{v_w}{v_s}$ speed ratio

$h_{eq} = \frac{v_w.a}{v_s}$ equivalent chip thickness (μ m)

Table 6.3 Surface roughness models for cylindrical grinding

Reference	Grinding Wheel	Work Material	Grinding Model for Cylindrical Grinding
Lindsay and Hahn (1973)	2A80K4VFMB	AISI 52100	$T_{av} = 12.5x10^3 \frac{d_g^{16/27} a^{19/27}}{D_{eq}^{8/27}} (1 + \frac{a_d}{s_d}) s_d^{16/27} (\frac{v_w}{v_s})^{16/27}$ $R_a = 0.487 T_{av}^{0.30}$ $for 0 < T_{av} < 0.254$ $R_a = 0.7866 T_{av}^{0.72}$ $for 0.254 < T_{av} < 2.54$
Chen, Jung and Inasaki (1989)	B56-3	Si_3N_4	$R_a = \gamma.a_m^{\varepsilon}$ $a_m = w^2 (\frac{v_w}{v_s}) \sqrt{a.d_e}$
Shin, Subrahmanya and Choi (2008)	32**A**-60-K-VBE	4140 Alloy Steel	$R_{a0} = 1.64 a_d^{-0.021} s_d^{0.385} h_{eq}^{0.284}$

Table 6.4 Surface roughness models for surface grinding

Reference	Grinding Wheel	Work Material	Grinding Model for Surface Grinding
Lindsay and Hahn (1973)	2A80K4VFMB	AISI 52100	$T_{av} = 12.5 \times 10^3 \frac{{d_g}^{16/27} a_p^{19/27}}{D_e^{8/27}} (1 + \frac{a_d}{s_d}) {s_d}^{16/27} (\frac{v_w}{v_s})^{16/27}$ $R_a = 0.487{T_{av}}^{0.30}\ for 0 < T_{av} < 0.254$ $R_a = 0.7866T_{av}^{0.72}\ for 0.254 < T_{av} < 2.54$ p
Chen, Jung and Inasaki (1989)	B56-3	Si_3N_4	$R_a = \gamma . a_m^{\varepsilon}$ $a_m = w^2 (\frac{v_w}{v_s}) \sqrt{a.d_e}$
Lee, Choi and Shin (2003)	38A60K5VBE	4140 Alloy Steel	$R_3 = R_a = R_{a,0} = 12.9 s_d^{0.54_1} a_d^{0.34} (\frac{v_w}{v_s})^{0.38} (\frac{s_t}{b_s})^{0.43}$

The number of variables should be reduced to have a more simplified and condensed model. Recently, many statistical analysis studies (Lee, Choi and Shin 2003; Choi and Shin 2007; Shin, Subrahmanya and Choi 2008) were carried out to determine the most significant variables affecting surface roughness. Among the parameters in the cylindrical grinding, the equivalent chip thickness was the most significant parameter. For surface grinding, the depth of cut is not a major variable for calculating surface of roughness (Lee, Choi and Shin 2003; Choi and Shin 2007). In the latest general model for surface roughness (Lee, Choi and Shin 2003; Choi and Shin 2007) the ratio of the cross-feed over the wheel width was considered as an additional parameter. The equivalent diameter was found to be insignificant for surface and cylindrical grinding (Lee, Choi and Shin 2003; Choi and Shin 2007; Shin, Subrahmanya and Choi 2008). Therefore, the variables that will be considered in the GP models are shown in Table 6.4 for surface grinding and Table 6.5 for cylindrical grinding. The fitness function was the minimum error, which is the summation of the absolute difference between the expected surface roughness value and the surface roughness value from the dataset.

Table 6.5 GP terminals for surface grinding

Surface Grinding	
Objective	Find the general model for surface roughness then compare the result with the calculated value from surface roughness model by Lee, Choi and Shin's (2003) model.
Terminal set "common parameters between the two functions"	X1= a_d µm, X2= s_d mm, X3= $\frac{v_w}{v_s}$, X4= $\frac{s_t}{b_s}$, X5= $a.d_e$, X6= h_{eq} µm (16/27)0.5926,(19/27)0.7037,(8/27)0.2963,0.4587,0.7866,0.3,0.72, -1,0.5,12.5,10,100,100,0.1,0.01,0.001, 0, 1, 2,3,4,5,6,7,8,9
Functions set	+, -, *, /,power
Standardised fitness	Minimum error (for each individual, the sum of the absolute difference between the expected roughness value and the value returned from the individual on all fitness cases.)
Control parameters	Population (M), Generation G, 4 times (100,200), 4 times (100,300)
Termination	The algorithm can be terminated when either a specified total number of generations have been run or when a performance criterion is satisfied.

Table 6.6 GP terminals for cylindrical grinding

Cylindrical Grinding	
Objective	Find the general model for surface roughness then compare the result with the calculated value from Shin's surface roughness model by Shin (Shin, Subrahmanya and Choi 2008).
Terminal set "common parameters between the two functions"	X1= a_d mm, X2= s_d mm, X3= $\frac{v_w}{v_s}$, X4= $a.d_e$, X5= h_{eq} µm, (16/27)0.5926,(19/27)0.7037,(8/27)0.2963,0.4587,0.7866,0.3,0.72, -

	1,0.5,12.5,10,100,100,0.1,0.01,0.001, 0, 1, 2,3,4,5,6,7,8,9
Functions set	+, -, *, /,power
Standardised fitness	Minimum error (for each individual, the sum of the absolute difference between the expected roughness value and the value returned from the individual on all fitness cases.)
Control parameters	Population (M), Generation G, 4 times (100,200), 4 times (100,300)
Termination	The algorithm can be terminated when either a specified total number of generations have been run or when a performance criterion is satisfied.

6.1.5.3 GP Model Development

A GP lab toolbox that has been developed by Silva (Silva 2004) for a Mat lab environment is used for generating GP generations. For more details of the results see Appendix (B); the procedures for generating a GP model are as follows:

- The datasets were tested using a GP lab toolbox, with different generation and individual sizes. Each dataset was tested three times for generation size 100 and individual 200 and the size of the individuals was increased to 300 for another three runs.

- In order to generate a generic surface roughness model for cylindrical grinding (surface grinding) all collected data from cylindrical grinding (surface grinding) are tested to generate a GP model, taking into account that the work material and wheel are different.

- The generated GP models for different datasets were analysed and compared with the experimental value surface roughness or calculated the surface roughness value from the model.

6.1.5.4 Discussion

For cylindrical grinding datasets, the grinding work material and wheel are different for different datasets, as shown in Table 6.6. For example, hardness and material group for cylindrical exp. dataset (1) are 62HRC and high steel respectively where for cylindrical cal. dataset (2) it is 40HRC and low steel. It can be noticed that the grinding work material, hardness, and grinding wheel are the same for all the datasets in surface grinding. For cylindrical cal. dataset (2), it can be noticed that the surface roughness value has a wide range that is due the increase of the dressing lead, dressing depth and work speed for the last two records.

The coefficient of determination R^2 is calculated for different GP models using Equation 6.10 to determine the fitness of the model.

$$R^2 = 1 - \frac{SS_e}{S_y} = 1 - \frac{\sum (y_{ijk} - \hat{y}_{ijk})^2}{\sum (y_{ijk} - \bar{y})^2} = 1 - \frac{\sum (R_a - \hat{R}_a)^2}{\sum (R_a - \bar{R}_a)^2} \quad \text{Equation (6.10)}$$

Table 6.7 Datasets values for cylindrical and surface grinding

Dataset name	Cylindrical exp. dataset (1)	Cylindrical cal. dataset (2)	Surface exp. dataset (3)	Surface cal. dataset (4)
References	(Chen 1995)	(Shin, Subrahmanya Choi 2008)	(Lee 2000)	(Choi and Shin 2007)
Number of cases	17	21	21	20
Wheel	A465-K5-V30W	32-60-K-VBE	38A60K5VBE	38A60K5VBE
Material group	HS	LS	LS	LS
Hardness	62	40	50	50
Roughness value	Experimental	Calculated	Experimental	Calculated
Work material	Oil hardened steel	Alloy steel 4140	Alloy steel 4140	Alloy steel 414
Dressing depth (mm)	0.005-0.025	0.025-2.33	0.008-0.020	0.005-0.015
Dressing lead	0.05-0.25	0.1-2.75	**0.12**	0.05-0.15

(mmmin/rev)				
Wheel speed (m/s)	**33**	28, 30.5, 33	**33**	22, 27, 18
Work speed (m/s)	0.25	0.33, 0.42, 0.5, 1.5, 2.5	0.1,0.2,0.3	0.05, 0.07, 0.09
Depth of cut (mm)	0.00213	0.012-0.018	0.01-0.03	0.005-0.075
Surface roughness (μm)	(0.22-0.45)	(0.419-2.23)	(0.25-0.57)	(0.318-0.65))

In cylindrical exp. dataset (1), a comparison between calculated value for surface roughness using GP and the experimental value is shown in Figure 6.13. The coefficient of determination is 0.47. In this dataset, the wheel speed, work speed, and depth of cut are fixed, so the input terminals for GP model are constant, such as the speed ratio, geometrical contact and equivalent chip thickness. That could affect the response of the generated GP model and explain the poor coefficient of determination value, as seen in Figure 6.13.

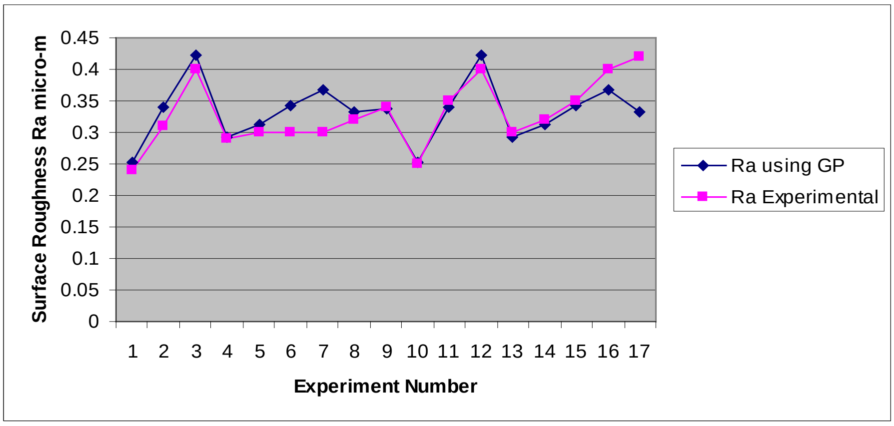


Figure 6.13 Comparison between Ra experimental value and using GP model for cylindrical exp. dataset (1)

A comparison between surface roughness value calculated using a GP model and surface roughness calculated by Shin, Subrahmanya and Choi (2008) for cylindrical cal. dataset (2), which is shown in Figure 6.14. The coefficient of determination is 0.97. The grinding conditions for this dataset were varied, which could explain the good fit of the GP model, as seen in Figure 6.14, and the generated surface roughness model was based on surface roughness value calculated using Equation 6.1. In other words, the experimental error in surface roughness is minimised.

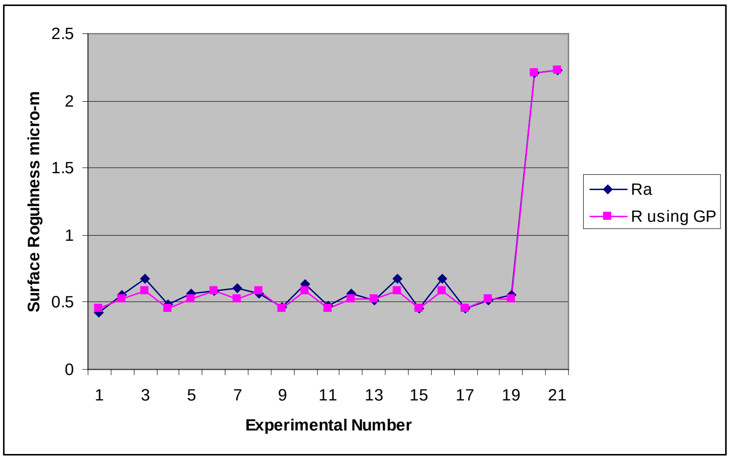


Figure 6.14 Comparison between Ra value and using GP model for cylindrical cal. dataset (2)

For surface grinding surface exp. dataset (3), the GP model for surface roughness (based on experimental roughness value) provides a good fit for surface roughness, as seen in Figure 6.15. The coefficient of determination is 0.93. This dataset is designed to calculate surface roughness value where the wheel speed, work speed and depth of cut vary, so the input terminals for GP model so the speed ratio, geometrical contact and equivalent chip thickness vary as well.

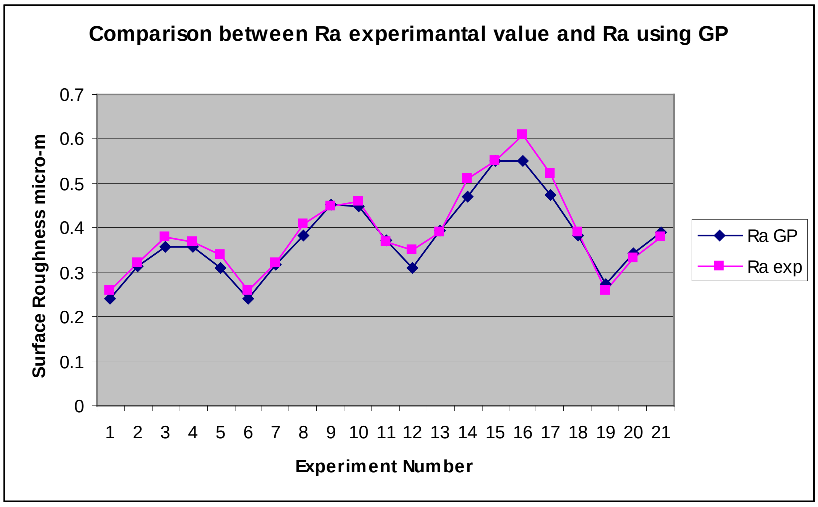


Figure 6.15 Comparison between Ra value and using GP model for surface exp. dataset (3)

For surface cal. dataset (4), a comparison between surface roughness using the GP model and surface roughness calculated by Choi and Shin (2007) is shown in Figure 6.16. The coefficient of determination is 0.79. The GP model wasn't a good fit for surface roughness value, which could be because the GP terminals were constants. In other words, GP's terminals $\frac{S_t}{b_s}$, speed ratio and geometrical contact were constant for the first eight records and from the ninth till seventieth records all the GP terminal were fixed.

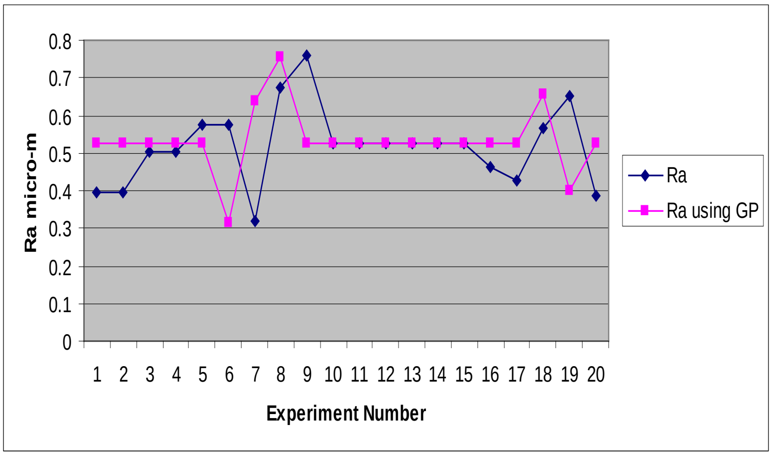


Figure 6.16 Comparison between Ra value and using GP model for surface cal. dataset (4)

For surface grinding, all of the data sets from Lee (2000) and Choi and Shin (2007) are collected together to generate a general GP model for calculating surface roughness, where the surface roughness is calculated from Lee, Choi and Shin's (2003) model. The coefficient of determination is 0.91. It can be noticed that the work material and wheel are the same for all datasets, as seen in Figure 6.17.

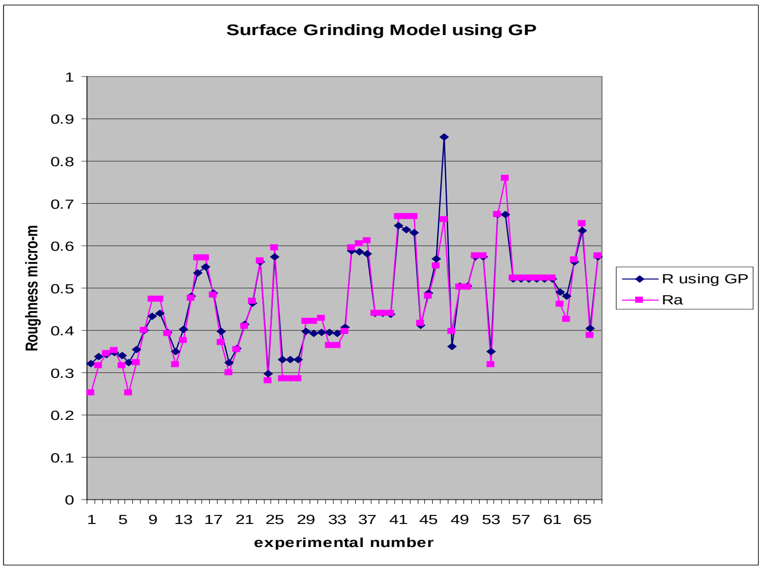


Figure 6.17 Comparison between Ra value and using a GP model for all surface grinding datasets

In this research, the experimental surface roughness value was collected from Chen's (1995) thesis for cylindrical grinding and from Lee's (2000) thesis for surface grinding. It can be noticed from Figures 6.13 and 6.14 for cylindrical grinding and Figures 6.15 and 6.16 for surface grinding that the GP model generated provided a better fit for surface grinding data. That could be explained because the data in the surface grinding dataset was structured to model surface roughness in grinding, which means the GP terminal varies while in the cylindrical experimental dataset, some of the GP terminals were constants such as speed ratio, equivalent chip thickness, and geometrical contacts, which affected the development of the GP model.

For cylindrical cal. dataset (2) and surface cal. dataset (4), the experimental value for surface roughness was not recorded so it has been assumed that the calculated value from Equations 6.1 and 6.3 for cylindrical and surface grinding, respectively, was equal to the experimental value. In other words, the GP model was built on calculated surface roughness values. The cylindrical cal. dataset (2) generated a good surface roughness model using GP, as seen in Figure 6.14, while for the surface grinding the surface roughness generated using GP was not a good fit, as seen in Figure 6.9. That could be justified for surface grinding dataset since all GP input terminals were constant between the 9^{th} till 17^{th} record respectively, which affected the response of the GP model.

In GKWS, the generalised surface roughness model for cylindrical plunge processes material steel 4140 and wheel 38A60K5VBE is given by Shin, Subrahmanya and Choi (2008) and was used to calculate surface roughness value using Equation 6.1.

In order to avoid biased surface roughness values, the calculated value will not be added until an experiment should be carried out first.

6.2 Screenshot of the Main Functions of GKWS

The main functions of GKWS are described in the following figures, 6.18 and 6.19 (see Appendix C for GKWS PHP and HTML source).

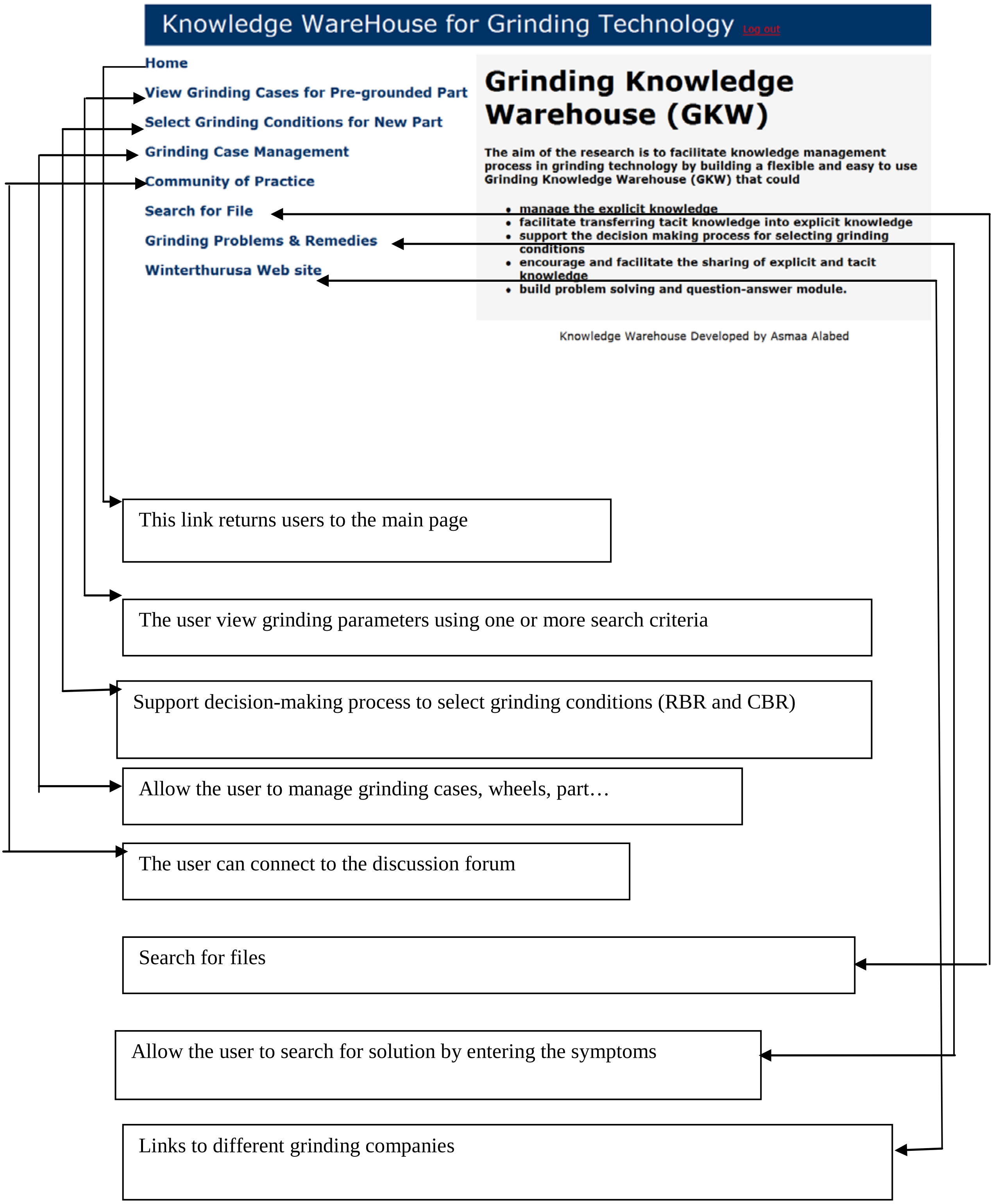


Figure 6.18 GKWS main screen

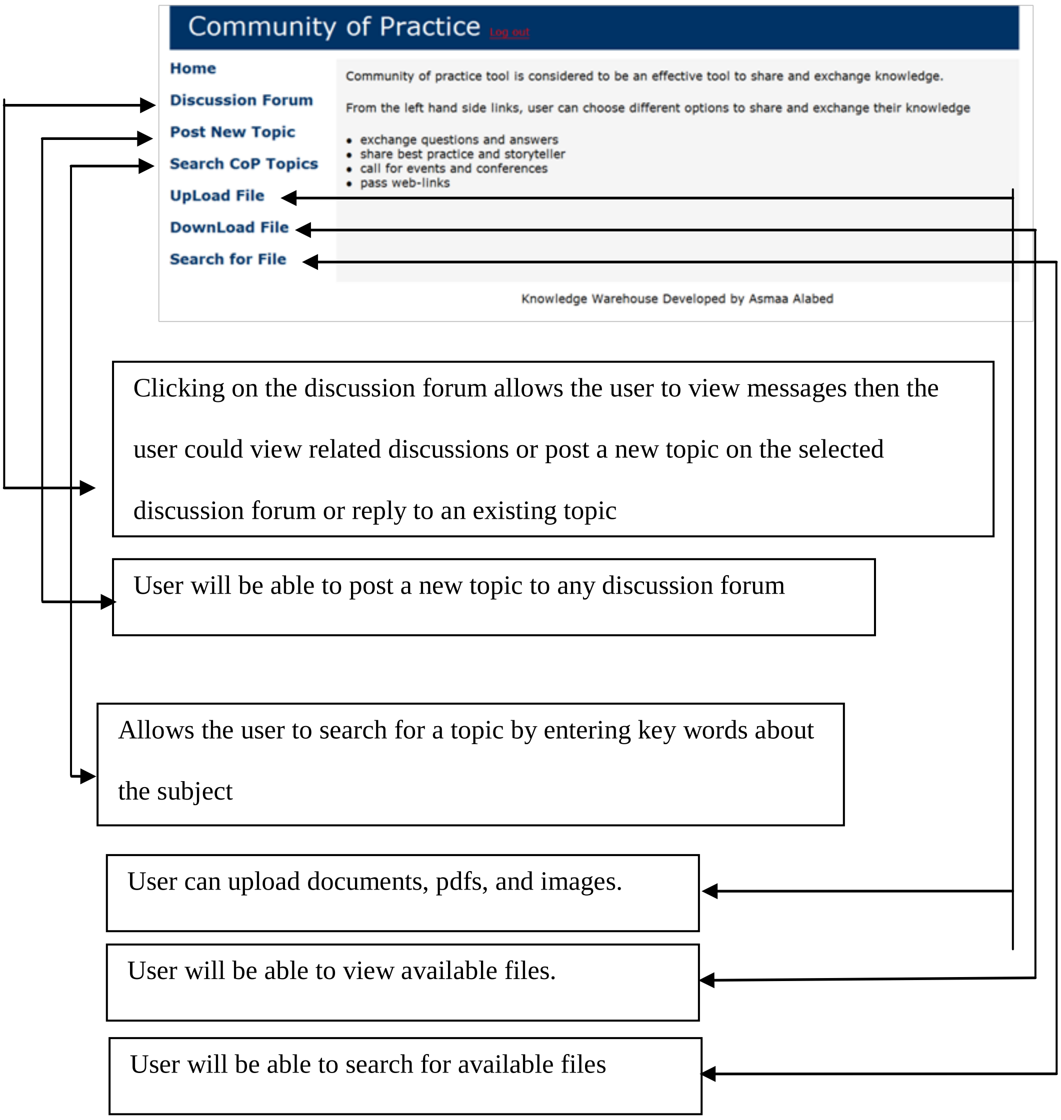


Figure 6.19 CoP screen

The users can add a new grinding case to the system by clicking on the *case management link* on the left hand side of the main page. The part number, wheel, dresser, coolant, and machine could be selected from the drop down list. If it is not there the user can add it by clicking on the required link on the left hand side of the page. The user chooses the material from the drop down menu and the material group will be selected automatically.

For example, material is Oil hardened steel, hardness 62HRc, diameter is 17 mm, work speed 0.25m/s. wheel speed 33m/s. dressing lead 0.15mm, dressing depth 0.01mm, feed rate 0.01 mm, wheel A465-K5-V30W, coolant Castrol Hysol, dresser rotary dresser, machine Jones and Shipmens, and surface roughness is 0.3 micro-m, as shown in Figure 6.20.

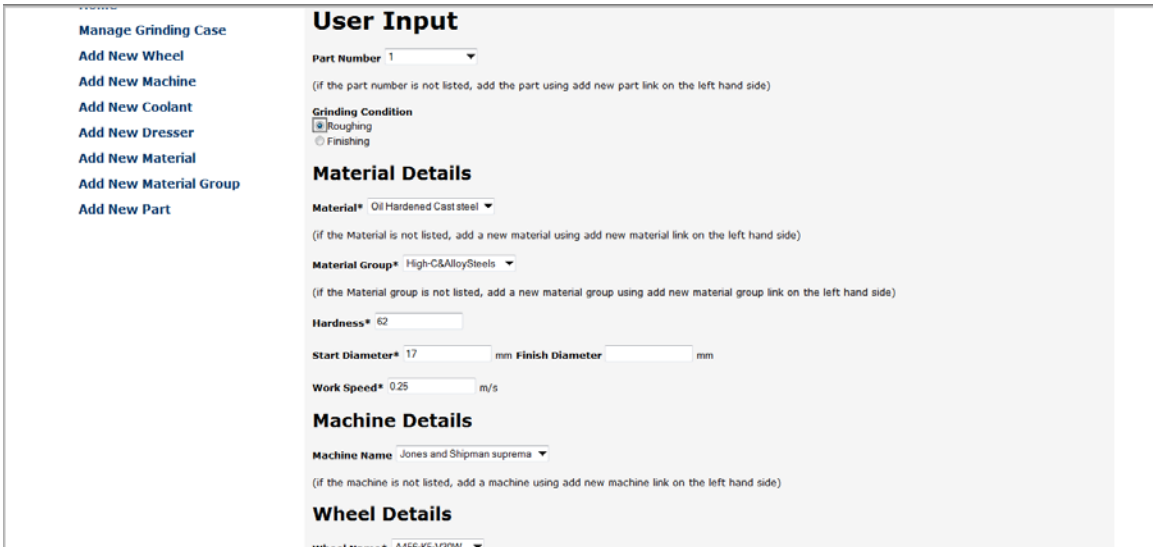


Figure 6.20 the Input form for grinding case

The depth of cut is not recorded in this example; the GKWS will calculate the value and notify the user that depth of cut is calculated as shown in Figure 6.21.

Figure 6.21 User is notified that depth of cut is calculated

If the user agrees on the value it could be typed in the form and then the user can click on the add button, as shown in Figure 6.22.

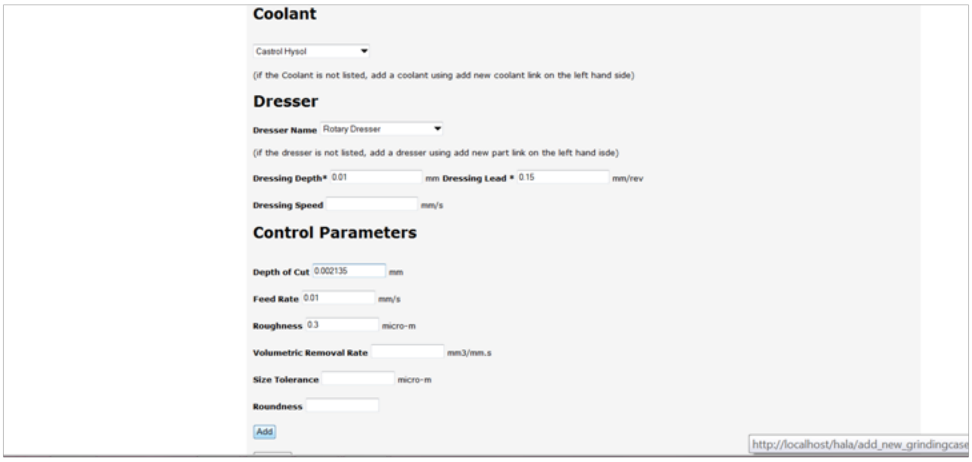


Figure 6.22 User typed in the depth of cut

Figure 6.23 The case is added

The user can add a newrecode by clicking on the first link or add a new case by clcicking on the second link, as shown in Figure 6.23.

Example 2, material 4140 alloy steel (HRC 40), wheel 32A-60-K-VBE,wheel speed 33 m/s, start diameter 38mm, work speed 0.5m/s, dressing depth 0.3 mm, dressing lead 0.16 mm, and depth of cut 0.01 mm. The user will be notified that feed rate value is missing and the calculated value will be 0.0414 mm, which will show up as in Figure 6.22. Also, the surface roughess value is missing, so the value is calculated using Shin's mode Equation 6.1, which is 0.49 micro-m, as shown in Figure 6.24. If the user agrees with the result, the value will be entered and added to the database.

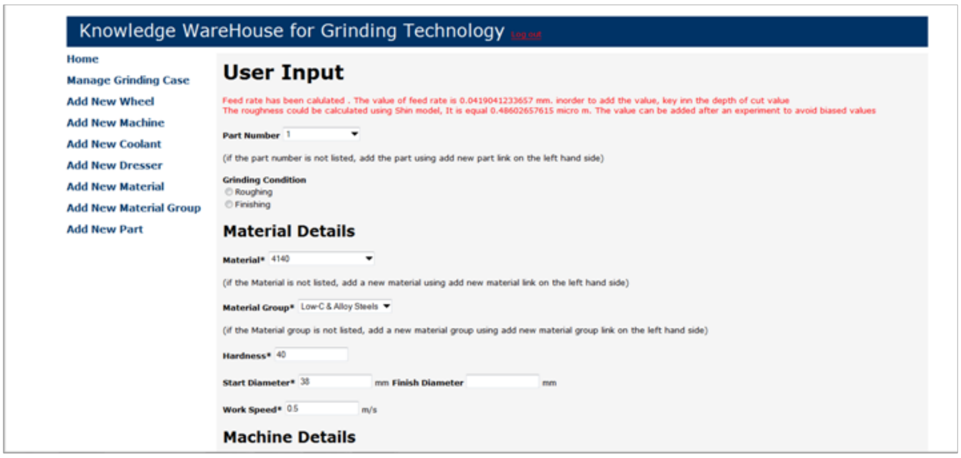


Figure 6.24 The calculated feed rate and surface roughness value

The required parameters are marked with (*) on this page. In case any of these field parameters is empty or contains a non-numerical value (letters or slashes), the user will be notified. For example, if the start diameter is missed and work speed value is (0,3 instead of 0.3) the warning message will appear on the top of the page, as shown in Figure 6.25.

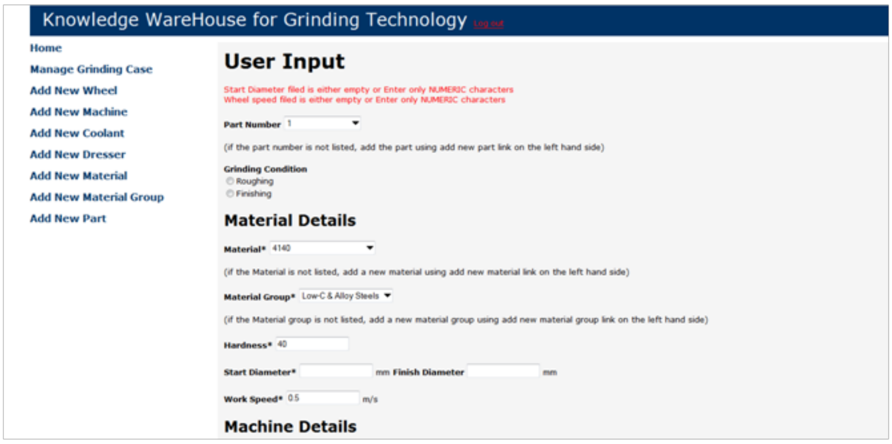


Figure 6.25 Warning message for missing data

The GKWS users will have username and password in order to login into the system and share their knowledge as shown in figure 6.26.

Figure 6.26 Security Screen

6.3 Conclusions and Summary

- This work shows novelty in using GP in modelling surface roughness in grinding. Also it shows novelty in using GP as tool for solving missing data problem. The GP generated model for surface grinding could be used to predict the missing surface roughness value for the same grinding type, material (alloy steel 4140 50 HRc) and wheel (38A60K5BE) for roughness value between (0.25-0.57). The relation between the variables in the GP models complies with the cylindrical and surface grinding mechanical behaviour.

- The model generated for surface grinding was applied for different datasets that has the same material and wheel. The coefficient of determination was very good and the equation was implied with the grinding relationships.

- From the experimental data available, GP could provide a good result for surface roughness even the range is small, as seen in Figures 6.13, 6.14, 6.15 and 6.16.

- If the GP terminals were constants, the GP model cannot present the effect of this parameter, as shown in Figure 6.17.

- GP model is not applicable to predict different materials performance.

- GP is capable of generating a model from non-structured conditions as shown in Figure 6.13; on the other hand, it gives reasonable prediction from structured data, as shown in Figures 6.14 and 6.15.

- The aim of the GP model is to provide reasonable prediction for surface roughness for industrial engineering application.

- However, the form has the capability to calculate missing records. The intelligent form for managing grinding cases sends warning messages to the user to indicate that important records are missing. The material group will be assigned automatically once the material is defined. The drop down menu for wheel, coolant, machine, dresser name and part number offers easier selection and accessibility to the records. However, if the defined name or number is not there, the user can update it by clicking on the defined link. The important parameters, such as; work diameter, work speed, wheel speed and so on are automatically cleaned and cleansed from any noise errors, such as non-numerical characters or slashes.

CHAPTER 7 SYSTEM EVALUATION AND DISCUSION

The GKWS has been designed and developed as described in the previous chapters; the next stage is the evaluation stage. This stage is important to assess the performance of GKWS. The evaluation of GKWS is divided into two stages: evaluating the technical tool and the performance of GP model.

7.1 Evaluation of GKWS

The performance of GKWS is evaluated to ensure that the system performance is at acceptance level for potential users. The system is evaluated from the following aspects:

- The user interaction and technical tools for the GKWS development,
- The performance of GP models in dealing with missing surface roughness values by comparing this model with different models.

7.2 Evaluation of User Interaction

The user interaction and technical tools were tested by two experts in grinding technology and computing. The user's interaction evaluation covers how the user interacts with the GKWS, and the GKWS features and objectives. The main feedbacks from the expertise were:

- The users can browse and surf the contents of the GKWS easily using the structured links on the left hand side of the main page.
- The GKWS interface is user friendly.
- In general, the system has already shown its purpose as a collection of knowledge and it is renewable as well.

7.3 Measure of GP Model Adequacy

In order to evaluate the GP models created for different datasets in section 6.1.5, different evaluation methods are used, such as:

- Absolute value for relative error
- Test of significance of GP models
- Confidence intervals and standard error
- Applying GP model to experimental dataset.

7.3.1 Absolute Value for Relative Error

A useful parameter to evaluate the surface roughness model using GP is the absolute value of relative error between surface roughness experimental value and calculated value from the GP model, as shown in Equation 7.1:

$$\text{Absolute relative error} = \frac{|R_a - R_{aGP}|}{Ra} \times 100 \qquad 7.1$$

For cylindrical exp. dataset (1), the maximum percentage of relative error is calculated (22%) and the average percentage for relative error is 6.6% (as seen in Figure 7.1), which is considered a good prediction of surface roughness, where a typical scattering is present.

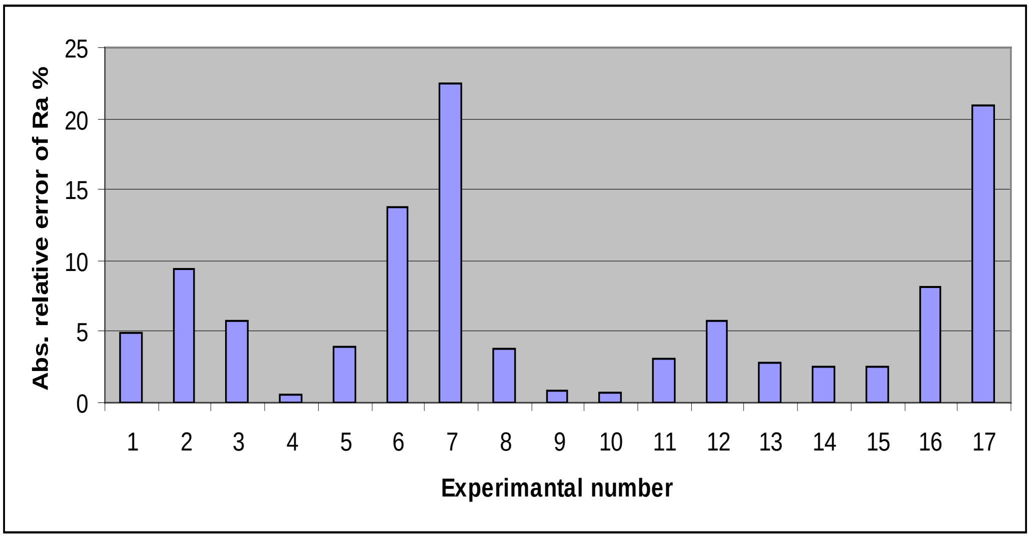


Figure 7.1 Absolute relative error between experimental and calculated surface roughness using GP for cylindrical exp. dataset (1)

The absolute value of relative error between the surface roughness value using GP and the surface roughness value in cylindrical cal. dataset (2) value is presented in Figure 7.2. It can be observed that the maximum error is 14.5% and the average relative error is 5.6%. This is considered a good prediction value for surface roughness compared to cylindrical exp. dataset (1) value.

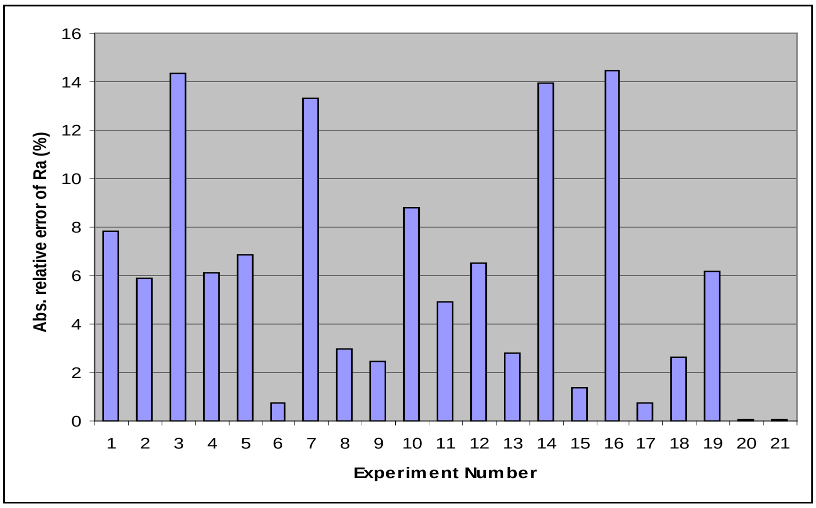


Figure 7.2 Absolute relative error between experimental and calculated surface roughness using GP for cylindrical cal. dataset (2)

The absolute value of relative error between the GP and experimental surface roughness value is described in Figure 7.3. It can be observed that the maximum error is 11.1% and the average relative error is 4.7%. This is considered a very good prediction value for surface roughness compared to the value in cylindrical exp. dataset (1) and cylindrical cal. dataset (2).

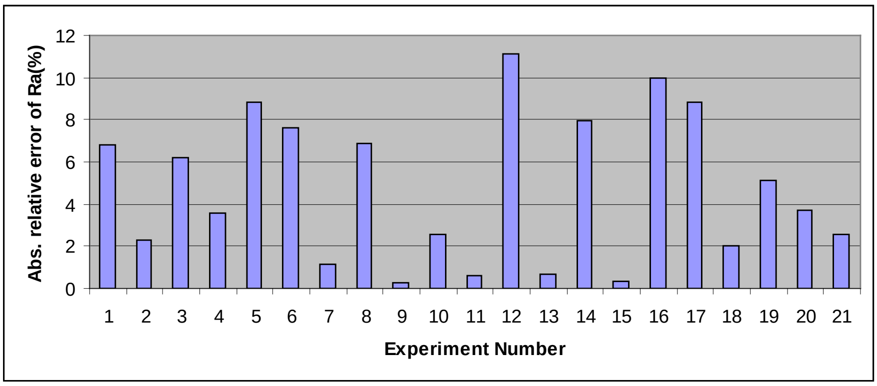


Figure 7.3 Absolute relative error between surface roughness experimental and calculated using GP for surface exp. dataset (3)

7.3.2 Test for Significance of a GP Model

The test for significant of a GP model is a test to determine whether a relationship exists between the response variable y and a subset of regressor variables $x_1, x_2, ...$

The appropriate hypotheses are:

$$H_o : \beta_1 = \beta_2 = ... = \beta_k = 0 \qquad 7.2$$

$$H_1 : \beta_j \neq 0 \text{ for at least one } j \qquad 7.3$$

Rejection of $H_o : \beta_1 = \beta_2 = ... = \beta_k = 0$ implies that at least one of the regressor variables $x_1, x_2, ...$ contributes significantly to GP model.

$$F_0 = \frac{SS_R / k}{SS_E /(n-p)} \qquad 7.4$$

$$SS_E = \sum_{i=1}^{i=n} \left(R_{ai} - R_{GPi}\right)^2 \qquad 7.5$$

$$S_{yy} = \sum_{i=1}^{n} \left(R_{ai} - \overline{R}_{ai}\right)^2 \qquad 7.6$$

$$S_{yy} = SS_R + SS_E \qquad 7.7$$

Where

R_{ai} the actual value roughness value

R_{aGP} GP value for surface roughness

$\overline{R}_{ai}$ the average value for experimental surface roughness

n number of observation or experiments

k number of variables in the GP model

$$p = k + 1 \quad 7.8$$

The hypothesis is rejected if the computed F_0 from Equation 7.4 is greater than $f_{\alpha,k,n-p}$ as shown in Table 7.1.

Table 7.1 Test for significance where α=0.05

Data Set	n	k	p	F_0	$f_{\alpha=0.05,k,n-p}$	Accepted/Rejected
Surface exp. dataset (3)	21	6	7	30.99	2.67	Rejected
Cylindrical cal. dataset (2)	21	5	6	333.97	2.34	Rejected
All Cylindrical	75	2	3	9.22	3.15	Rejected
All Surface	67	6	7	152.04	2.25	Rejected

The hypothesis for surface cal. dataset (4) and cylindrical exp. dataset (1) would be rejected if α value is increased, as shown in Table 7.2. For example, if α is increased to 0.1, the GP model for surface cal. dataset (4) will be rejected, which means that there is a significant relationship between surface roughness using GP and at least one of the variables in the model.

Table 7.2 Test for significance where α=0.1

Data Set	n	k	p	F_0	$f_{\alpha,k,n-p}$	Accepted/Rejected
Surface cal. dataset (4)	20	2	3	2.7	$\alpha=0.1$ 2.64	Rejected
Cylindrical exp. dataset (1)	17	4	5	1.55	$\alpha=0.25$ 3.26	Rejected

7.3.3 Confidence Interval

If the sample size is relatively small (n < 30), the usual assumption is that the population is random sample from normal distribution. This leads to confidence intervals based on *t* distribution.

$$\overline{R}-t_{\alpha/2,n-1}s/\sqrt{n}\le R\le \bar{R}-t_{\alpha/2,n-1}s/\sqrt{n} \qquad 7.12$$

Where the standard division (*s*) is $s=\sqrt{\dfrac{\sum(R_a-\overline{R_a})^2}{n-1}}$ 7.13

And *n* is the sample size

Confidence interval tables are presented in Table 7.3.

Table 7.3 Confidence intervals

Cases	95% Confidence intervals
Cylindrical exp. dataset (1)	n=17,Ra= 0.343±0.0406
Cylindrical cal. dataset (2)	n=21, Ra= 0.708±0.231
Surface cal. dataset (3)	n=21, Ra=0.392±0.0447
Surface cal. dataset (4)	n=20, Ra= 0.504 ± 0.0464

7.3.4 Applying GP Model on Experimental Data

The surface exp. dataset (3) has the minimum relative absolute error and for that reason the dataset is investigated more deeply. On the other hand, the GP model for surface exp. dataset (3) was generated based on experimental surface roughness value (Lee 2000). This model was compared to Lee, Choi and Shin's (2003) model and applied to different datasets that have the same grinding material and wheel.

The coefficient of determination (R^2) was calculated for surface roughness model using GP and for Lee, Choi and Shin's (2003) model, which is the same (0.94). Figure 7.4 represents the GP trees for surface roughness model with best fitness value. Figure 7.5 shows the graphical outputs from GP for surface roughness modelling. The first graph from the left hand side represents the structure complexity in obtaining the best surface roughness so far. The second figure in the right hand side gives the most information regarding the GP progression for a specific dataset. It can be seen that the fitness (accuracy) decreases (the smaller fitness measure the better) over the number of generations. The depth of nodes increases, which can give a more comprehensive and detailed classification solution. Figure 7.6 represents comparison between experimental surface roughness value, value calculated from GP model and Lee, Choi and Shin's (2003) model. Another useful parameter to evaluate the surface roughness model using GP is the absolute value of relative error. In this case, this value is compared to the surface roughness value calculated from Lee, Choi and Shin's (2003) model where the surface roughness is missing. The parameter is shown in Figure 7.7, where it can be observed that the maximum error is 11.1 and the average error is 4.7%. This is considered a good prediction value for surface roughness.

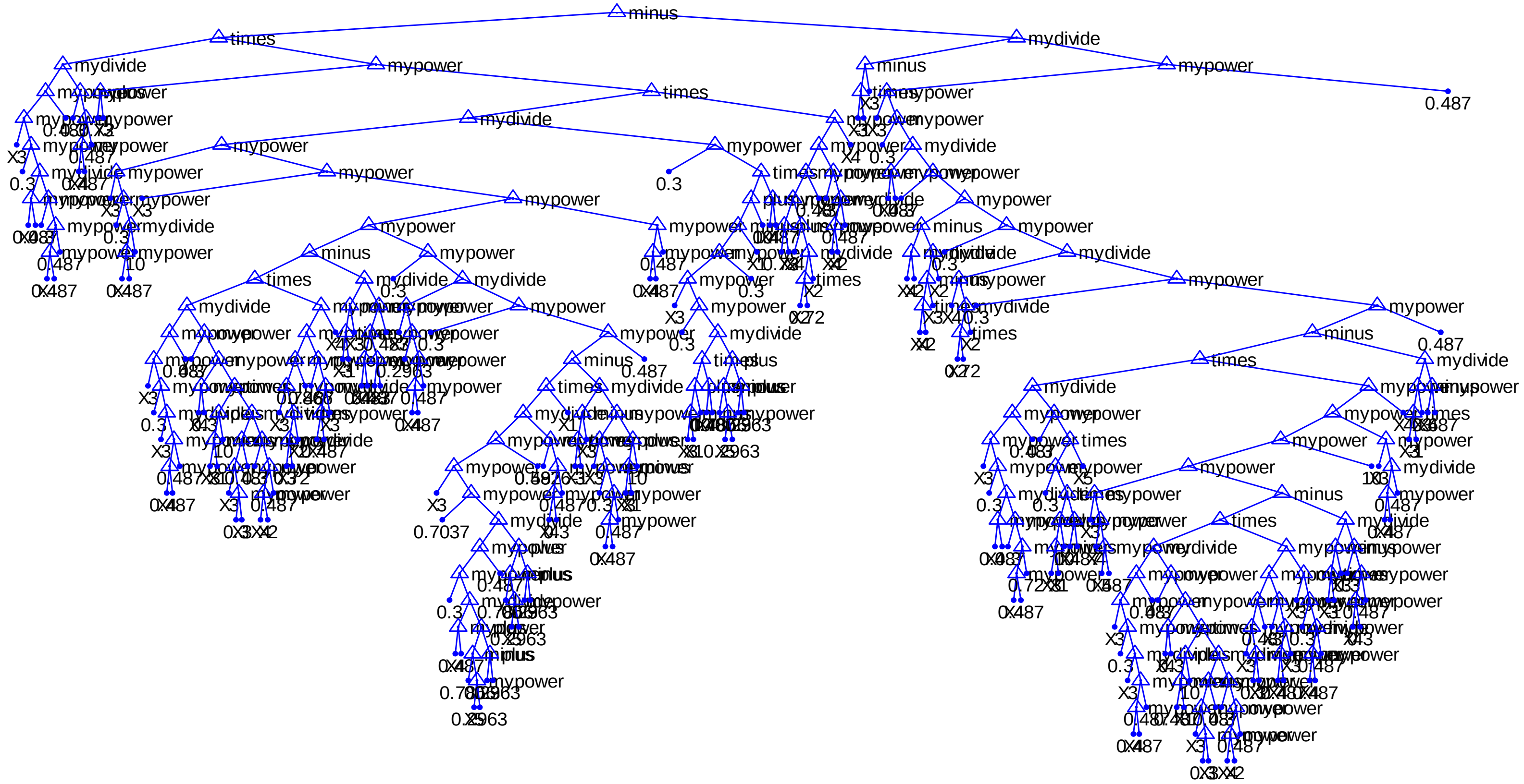


Figure 7.4 The tree output for Ra modelling using GP (See Appendix B for programming code page 276 for Table 3 terminal set as dataset X and expected output as dataset Y)

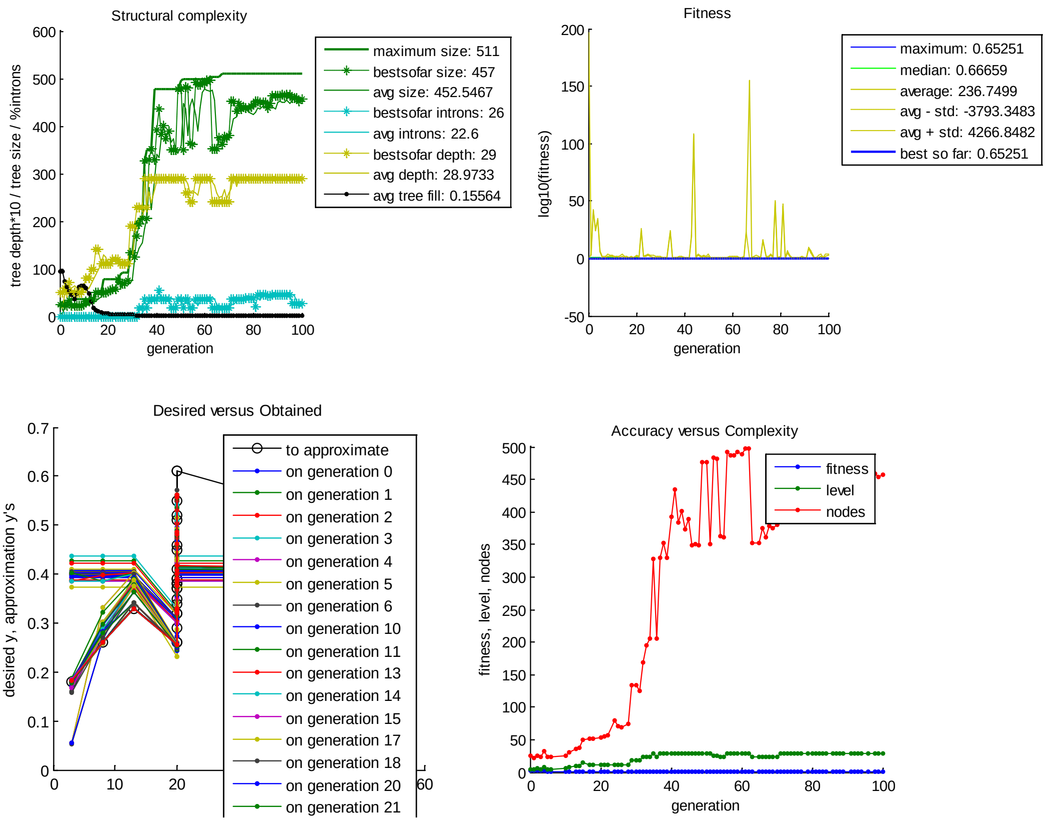


Figure 7.5 Graphical output for Ra modelling using GP

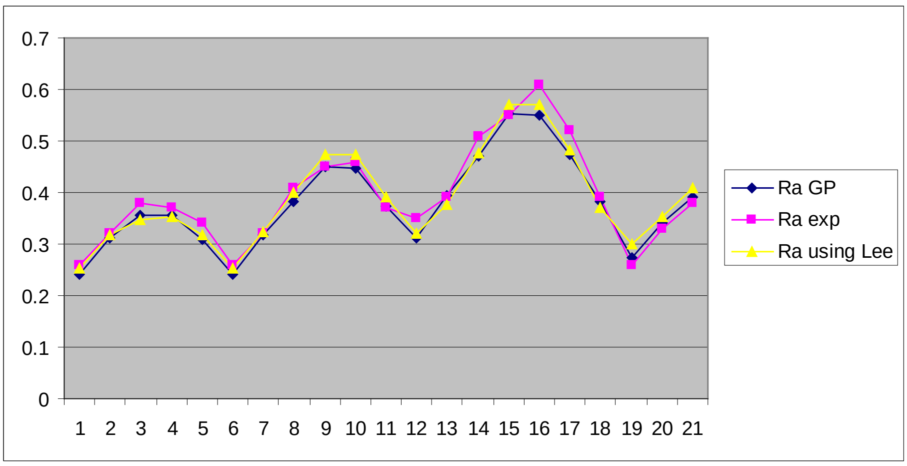


Figure 7.6 Comparison between Ra experimental value, Ra using GP and Lee, Choi and Shin's (2003) model

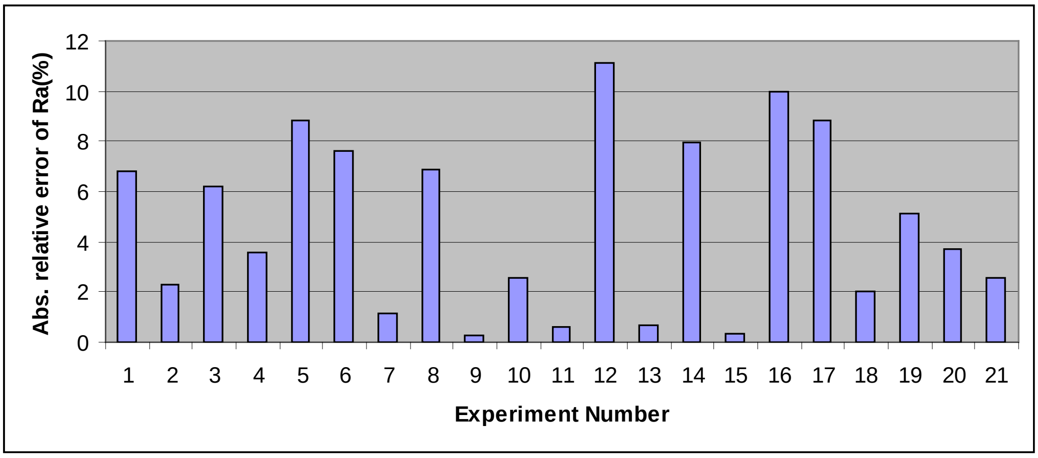


Figure 7.7 Absolute relative error between Ra experimental and Ra using GP

The GP model for surface roughness is:

$$R_a = \left(0.72+\frac{v_w}{v_s}\right)^{\dfrac{0.3\left(\frac{s_t}{b_s}\right)\left(10-a_d+\left(\frac{v_w}{v_s}\right)^{0.3(o.3)^{\frac{\left(\frac{v_w}{v_s}-a_d+10\right)\left(\frac{s_t}{b_s}\right)^{0.487}}{1.39-(ad_e)^{0.2963}}}}\right)}{\left(\left(\frac{v_w}{v_s}\right)^{0.3\frac{\left(\frac{s_t}{b_s}\right)^{0.487}}{10}}\right)^{\left(\frac{v_w}{v_s}\right)^{r}}}\left(\left(\frac{v_w}{v_s}\right)^{0.3^{0.3\frac{\left(\frac{s_t}{b_s}\right)^{0.487}}{\left(\frac{s_t}{b_s}\right)^{0.237}}}}\right)^{0.487}} - \frac{2\left(\frac{v_w}{v_s}\right)}{\left((s_d)^{0.3\frac{\left(\frac{s_t}{b_s}\right)^{0.487}}{0.3^{\left(\frac{\frac{s_t}{b_t}-s_d\frac{s_t}{b_s}+\frac{v_w}{v_s}}{s_d}\right)^{0.3}}}}\right)^{0.487}} \quad (7.14)$$

Where
$$r = \frac{0.3^{\left(\frac{s_t}{b_s}\right)\left(10-a_d+\left(\frac{v_w}{v_s}\right)^{0.3\left(0.3^{\frac{\left(\frac{v_w}{v_s}-a_d+10\right)\left(\frac{s_t}{b_s}\right)^{0.487}}{1.3829-(a.d_e)^{0.2963}}}\right)''}\right)}}{\left(\left(\frac{v_w}{v_s}\right)^{0.3\frac{\left(\frac{s_t}{b_t}\right)^{0.487}}{10}}\right)^{\left(\frac{v_w}{v_s}\right)^{(r1)^{0.237\left(\frac{s_t}{b_s}\right)}}}} \quad ;$$

$$r_1 = (r_2)^{a_d}\left(\frac{\left[\left(\frac{v_w}{v_s}\right)(0.7037)^{\frac{(0.3)^{0.487\frac{\left(\frac{s_t}{b_s}\right)^{0.487}}{}}}{2(1.3829-(a.d_g)^{0.2963})}}\right]^{0.487}}{0.5926^{\left(\frac{s_t}{b_s}\right)^{0.1416}}}\right) - \frac{-2\left(\frac{v_w}{v_s}\right)}{\left(\frac{v_w}{v_s}\right)^{\left(\frac{v_w}{v_s}-a_d+10\right)\left(\frac{s_t}{b_s}\right)^{0.237}}}$$

$$r_2 = \frac{\left(\frac{v_w}{v_s}\right)^{0.487(0.3)\left(\frac{\frac{v_w}{v_s}}{\left(\frac{s_t}{b_s}\right)^{0.237}}\right)}}{\left(\frac{s_t}{b_s}\right)^{0.3(\frac{v_w}{v_s}-a_d+10)\left(\left(\frac{\left(\frac{v_w}{v_s}\right)^{0.487(0.3^{\frac{v_w}{v_s}})}}{0.3\left(\frac{s_t}{b_s}\right)^{0.487 s_d}}\right)\right)}} - \frac{2\frac{v_w}{v_s}}{\left(\left(\frac{s_t}{b_s}\right)^{0.237\frac{v_w}{v_s}}\right)^{0.487}}$$

0.3

The surface roughness model using GP model has been applied for another set of surface grinding data (Lee, Choi and Shin 2003) and it is compared to Lee, Choi and Shin's (2003) surface roughness model, as seen in Figure 7.8. The coefficient of determination calculated was 0.97. Another useful parameter to evaluate the surface roughness model using GP is the absolute value of relative error. In this case, this value is compared to the surface roughness value calculated from Lee, Choi and Shin's (2003) model where the surface roughness is missing. The parameter is shown in Figure 7.9, where it can be observed that the maximum error is 12.4% and the average error is 3.8%. This is considered a good prediction value for surface roughness.

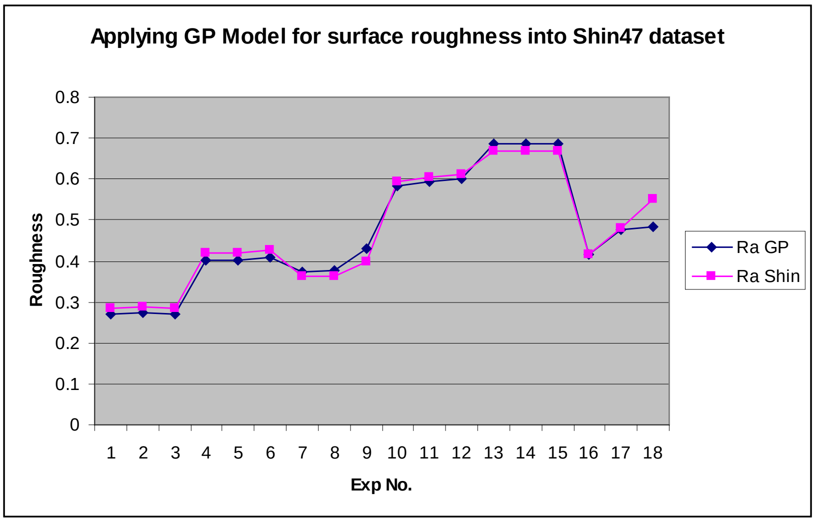


Figure 7.8 Comparison between Ra experimental value and Ra using GP and Lee, Choi and Shin's (2003) model

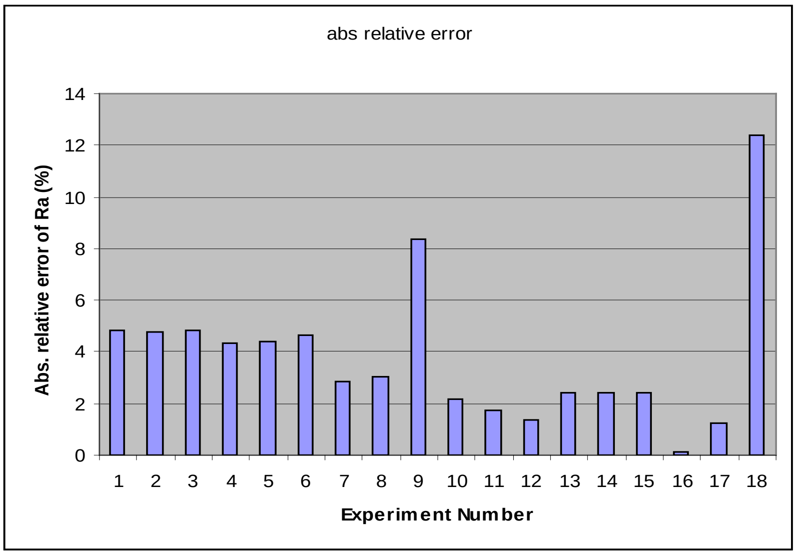


Figure 7.9 Absolute relative error between Ra calculated using Lee, Choi and Shin's (2003) model and Ra using GP

In order to generate a surface roughness model using GP for a structured dataset and relatively large data where it is 47 cases, data from Lee, Choi and Shin (2003) was tested to generate surface roughness model using GP as shown in Figure 7.10. The surface roughness from Lee, Choi and Shin (2003) article is calculated using Equation 6.4.

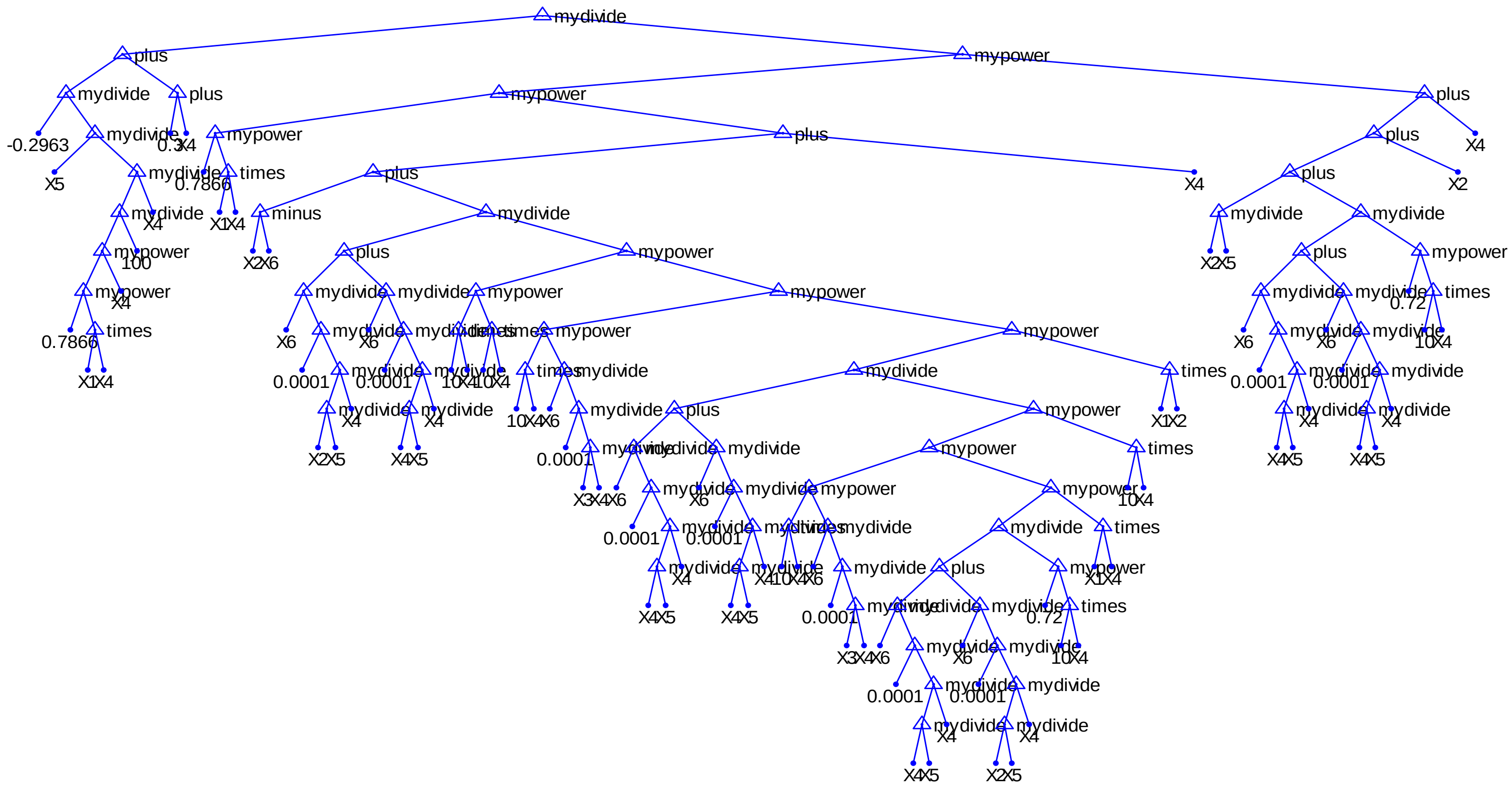


Figure 7.10. The tree output for Ra modelling using GP (See appendix B for programming code page 276 Table 5)

Figure 7.11 describes comparison between surface roughness value using GP and surface roughness in the tested dataset. The absolute value of relative error between the surface roughness value using GP and the surface roughness in the dataset is shown in Figure 7.12. It can be observed that the average error is 5.9%. This is considered a very good prediction value for surface roughness.

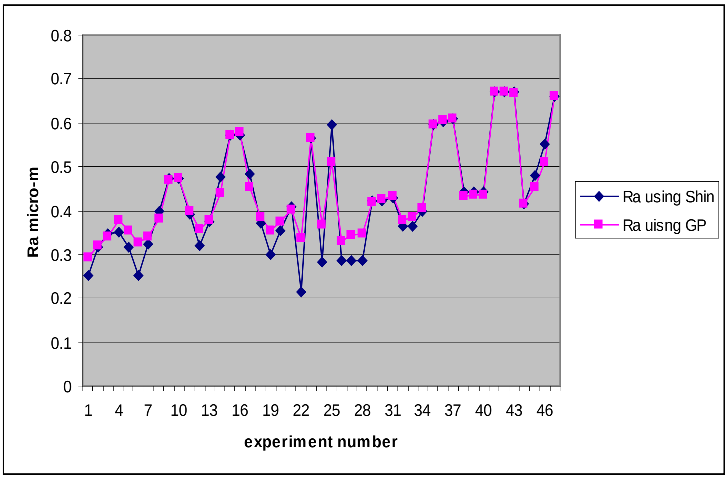


Figure 7.11 Comparison between Ra value and using a GP model for surface grinding

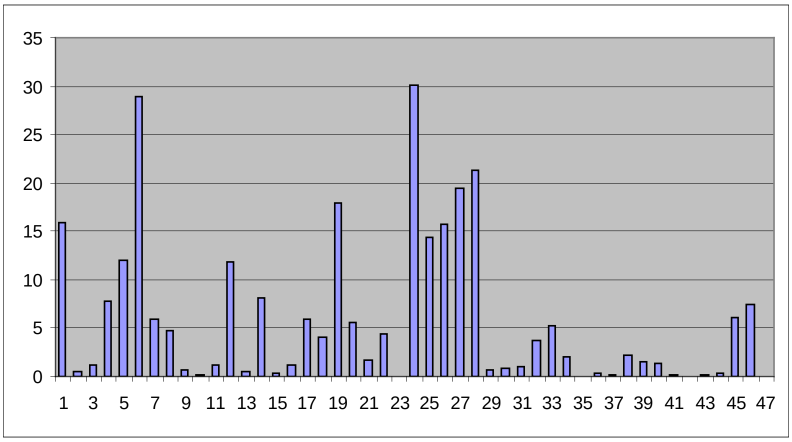


Figure 7.12 Absolute relative error between surface roughness in Lee, Choi and Shin (2003) and calculated using GP for surface grinding

For surface grinding, the best GP generated model was applied surface exp dataset (1), as shown in Figure 7.13. The coefficient of determination is 0.85.

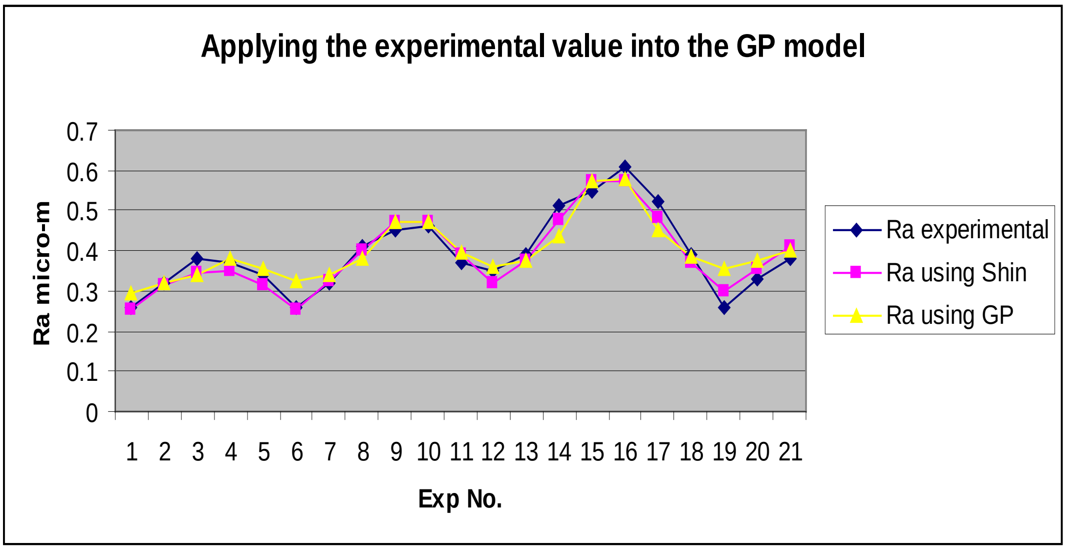


Figure 7.13 Comparison between experimental Ra, Ra using GP model and Ra using Shine's model for surface grinding

7.4 Summary

This chapter evaluates the performance of GKWS from different aspects. The user interface feedback from the users was positive and promising for GKWS. The system will need further assessment by the target users and in the industry.

The GP modelling for surface roughness was evaluated for different datasets. The GP provides a better prediction for surface roughness where the GP terminals vary than constants GP terminals, as proved in surface roughness experimental dataset (3). The GP has a similar accuracy as normal statistics regression methods such as Lee, Choi and Shin's (2003) surface roughness model.

CHAPTER 8 CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

As a part of this study, a knowledge support system was designed and developed in order to facilitate knowledge acquisition in grinding technology and assist the decision-making process for selecting grinding conditions by compensating for missing data. The objective of the study was to encourage and facilitate the knowledge management process in advanced technology. In particular the focus was on sharing and retrieving tacit knowledge by building a CoP forum and using Genetic Programming (GP) to deal with missing data by using previously stored grinding cases.

The GKWS provides a guided tool for users to support the decision-making process to provide suggestions for selecting grinding conditions using RBR and CBR and it can learn from new and previous grinding cases to improve and expand the case-based cases.

The GKWS has the potential to manage knowledge processes in grinding technology. The GKWS interfaces collect the grinding parameters and allow users to update new information that is relevant to grinding processes, such as new wheels, new material or new coolant etc. The collected data is stored and taken for further processes to be retrieved in case of previous grounded cases. The automated records would save time and effort for current and new employees as well as allow for training sessions for new employees.

The intelligent form for managing grinding cases sends warning messages to the user to indicate that important records are missing. The GKWS ensures that data entry is cleansed automatically from the first step by removing any noise such as non numerical characters and/or slashes. The important grinding conditions such as depth of cut, feed rate and surface roughness are calculated using mathematical equations and empirical models. The GKWS developed a new methodology to deal with missing data in grinding operations. The new methodology is built on IF-Then rules, mathematical equations and modelling using GP. Dealing with missing data improves the performance of knowledge discovery in the GKWS and the results of the CBR.

The GKWS is capable of completing the missing or not recorded grinding variables that will enable the wide use of the whole system. A novelty of dealing with missing data or incomplete records has been explained using GP.

From the evaluation and investigation of the GKWS, it showed some potential benefits of using the system to support the decision-making process. It will save the time and effort for target users by showing the most related grinding case. In order to increase the efficiency of the reasoning system of GKWS, using mathematical equations and/or GP method solves the missing data issue. The search for similar cases using CBR is more accurate and useful where the grinding cases are kept with sufficient information for further retrieving. The efficiency of CBR has also been improved due to the fact that the incomplete data has been filled using mathematical equations, whereas before the CRB would have terminated and RBR will be activated.

The GP is developed for modelling surface roughness in cylindrical and surface grinding. Unlike the genetic algorithm, the GP has the ability to represent the relationship between variables using functions. The developed GP model for surface grinding shows the ability to predict the surface roughness parameter especially when the GP terminals vary and the same material and wheel are used.

The discussion forum facilitates and supports transferring tacit knowledge into explicit knowledge where the users can exchange their ideas, questions and answers, and pass on important links. The knowledge sharing and retrieving are encouraged and facilitated using a categorised forum where all the posts are saved and ready for further operations such as searching, sharing or updating. The tacit knowledge is acquired directly from the knowledge engineers. The tacit knowledge could be exchanged through online storytellers, best practice, or general discussions. The users can input a key word and look for related output in the discussion forum. The employee will be able to participate and contribute to the GKWS at any time and wherever they are located. The debate and discussion in GKWS will create new knowledge that is accessible and available when needed.

The basic search engine is capable of finding the required data using a field or Boolean search. The GKWS allows the users to browse grinding problems and their remedies or look for remedies for a specific problem, which will speed up the response time for solving problems in grinding operations.

The GKWS is designed using Open Source Software tools and PHP-MySQL-Apache tools, which will keep the cost of the software to the minimum. The GKWS is reliable and bug free. It can be concluded from the research project that:

- The developed GKWS has the capability to store and retrieve information for grinding operation.
- The GKWS can facilitate the communication for a grinding practise community.
- The GKWS integrated various AI technologies to provide knowledge support for grinding operations and recommendation of selection process condition parameters.
- The project has demonstrated that the GP method is a feasible tool dealing with missing data problems. It can provide similar or even better accuracy than statistic regression method.

8.2 Future Work

Although promising results have been achieved in GKWS, the GKWS could be improved by:

- Collecting more real cases from the industry for further assessment of the efficiency of the system. GKWS should be tested with a large number of cases from the industry.

- Investigating modelling surface roughness using GP where the data is structured and large numbers of cases are collected, since the established GP model results provide evidence that the generated model for surface roughness in surface grinding has good predictive capabilities.

- Investigating the feasibility of automating the input for the GKKWS by integrating the GP Mat-lab to the GKWS system so the result will come to the system straight away.

- Modifying the search engine for the CoP forum using more advanced techniques. While the current search engine adopted Boolean and identified keywords to look for required knowledge, the system could be improved using indexing and tagging.

- The evaluation of entire system acceptance by target users has not been covered in this research due time constraints. The GKWS needs to be assessed by users for their feedback and comments, which will improve it.

- Applying more security for GKWS since it has the potential to be published on the Internet.

References

Affifi, A. and Elashoff, R. (1966). Missing observation in multivariate statistics 1: review of the literature. *Journal of the American Statistical Association* 61: 595-604.

Alabed, A. and Chen, X. (2009). *Development of a knowledge warehouse for grinding.* In: Proceedings of Computing and Engineering Annual Researchers' Conference 2009: CEARC'09. University of Huddersfield, Huddersfield, pp. 99-104. ISBN 978186218085.

Agarwal, S. and Rao, P. V. (2005). A probabilistic approach to predict surface roughness in ceramic grinding. *International Journal of Machine Tools & Manufacture* 45: 609-616.

Alavi, M. and Leidner, D. E. (2001). Knowledge management and knowledge management systems: conceptual foundations and research issues. *MIS Quarter* 25(1): 107-136.

Al-Alawi, A. I., Yousif N. and Fradoon Y. (2007). Organisational culture and knowledge sharing. *Journal of Knowledge Management* 11(2): 22-42.

Allee, V. (1997). *The knowledge evolution: expanding organisational intelligence*. Butterworth-Heinemann, p.231.

Ambler, S. W. (2004). *Model Driven Development with UML 2*. The Object Primer, 3rd edition: Agile, Cambridge University Press.

Aurich, J. C., Biermann, D., Blum, H., Brecher, C., Carstensen, C., Denkena, B., Klocke, F., Kroger, M., Steinmann, P. and Weinert, K. (2009). Modelling and simulation of process: machine interaction in grinding. *Production Engineering Research Development* 3:111-120.

Baek S., Liebowitz, J., Prasad, S. Y. and Granger, M. (1999). Intelligent agents for knowledge management, Ch 11 in Knowledge Management Handbook, edited by Liebowitz, J., by CRC Press LLC.

Bagirov, A. M., Rubinov, A. M. and Yearwood, J. (2001). A heuristic algorithm for feature selection based on optimization techniques, Chapter 2, *Heuristic and Optimization for Knowledge Discovery* (this volume), Idea Group Publishing, USA.

Beckman, T. (1999). The current state of Knowledge Management, Ch 1 in *Knowledge Management Handbook*, edited by Liebowitz, J., by CRC Press LLC.

Beckman, T. (1997). A methodology for knowledge management. *International Association Of Science And Technology For Development (IASTED) AI and Sot Computing Conference*, Banff, Canada.

Benbasat, I. and Dhaliwal, J. S. (1989). A framework for the validation of knowledge acquisition, Knowledge Acquisition 1(2).

Bendoly, E. (2003). Theory and support for process frameworks of knowledge discovery and data mining from ERP systems. *Information & Management* 40: 639-647.

Bhirud, S., Rodrigues, L. and Desai, P. (2005). Knowledge sharing practices in KM: A case study in Indian software subsidiary. *Journal of Knowledge Management Practice*. Available at www.tlainc.com/articl103.htm - 60k [Accessed April 2007].

Bradley, P. S., Mangasarian, O. L. and Street, W. N. (1998). Feature selection via mathematical programming. *Informs Journal on Computing* 10(2): 209-217.

Brinksmeier, E., Aurich, J. C., Govekar, E., Heinzel, C., Hoffmeister, H. W., Peters, J., Rentsch, R., Stephenson, D. J., Uhlmann, E., Weinert, K. and Wittmann, M. (2006). Advances in modeling and simulation of grinding processes. *Annals of the CIRP* 55(2): 667-696.

Brown, J. S. and Duguid, P. (1998). Organising knowledge. *California Management Review* 40(3): 90-111.

Brown, M. L. and Kros, J. (2003). Data mining and the impact of missing data. *Industrial Management and Data Systems*. 103(8): 611-621.

Boose, J. H. (1989). A survey of knowledge acquisition techniques and tools. *Knowledge Acquisition* 1(1): 39-58 (March).

Carpenter, J. R., Kenward, M. G. and Vansteelandt, S. (2006). A comparison of multiple imputation and doubly robust estimation for analyses with missing data, *Journal of the Royal Statistical Society*, Series A, 169: 571-584.

Carrillo, J. E. and Gaimon, C. (2000). Improving manufacturing performance through process change and knowledge creation. *Management Science* 46(2): 265-288.

Chau, P. Y. (2005). ChXX: Why Knowledge Management Fails, edited by Jennet, M., *Productivity Impact from Knowledge Management*, London: Idea Group Inc.

Chawla, D. and Joshi, H. (2010). Knowledge management practices in Indian industries – a comparative study. *Journal of Knowledge Management* 14(5): 708-725.

Chen, C., Jung, Y. and Inasaki, I. (1989). Surface, cylindrical and internal grinding of advanced ceramics. *ASME PED-V39 Grinding Fundamentals and Applications* 39: 201-211.

Chen, X. (1995). Strategy for the Selection of Grinding Wheel Dressing Conditions. Liverpool John Moores University. PhD thesis.

Chen, X. and Rowe W. B. (1996a). Analysis and Simulation of the Grinding Process – Part I, Generation of the Grinding Wheel Surface. *International Journal of Machine Tools & Manufacture* 36(8): 871-882.

Chen, X. and Rowe, W. B. (1996b). Analysis and Simulation of the Grinding Process – Part II, Mechanics of Grinding. *International Journal of Machine Tools & Manufacture* 36(8): 883-896.

Chen, X., Rowe, W. B., Mills, B. and Allanson, D. R. (1996). Analysis and Simulation of the Grinding Process – Part III, Comparison with Experiment. *International Journal of Machine Tools & Manufacture* 36(8): 897-906.

Chen, X., Rowe, W. B., Mills, B. and Allanson, D. R. (1998). Analysis and Simulation of the Grinding Process – Part IV, Effects of Wheel Wear. *International Journal of Machine Tools & Manufacture* 38(1): 41-49.

Chen, X. and Rowe, W. B. (1999). Modelling Surface Roughness Improvement in Grinding. *Journal of Engineering Manufacture*. *Proceedings of ImechE* 213(B1) Part B: 93-96.

Chen, X., Rowe W., Allanson, D. R. and Mills, B. (1999). A grinding power model for selection of dressing and grinding conditions. *Transactions of ASME* 121:632-637.

Chen, X. (2002a). Knowledge structure use for manufacturing technology. Proceeding of the 18th National Conference on Manufacturing Engineering and Management 261-265.

Chen, X. (2002b). Systematic consideration of grinding process monitoring. *Proceedings of the 8th Chinese Automation and Computing Society Conference* in the

UK, 175-178.

Chen, Y. (1998). *A Generic Intelligent Control System for Grinding*. PhD thesis. Department of Mechanical Engineering. Liverpool John Moores University. UK.

Chinowsky, P. (2007). Knowledge management to learning organization connection. *Journal of Management in Engineering* 23(122): 742-597.

Claromentis. http://www.claromentis.com [Accessed November 2004].

ComponentOne. http://www.componentone.com/pages.aspx?pagesid [Accessed November 2004].

Chiu, N. and Malkin, S. (1993). Computer simulation for cylindrical plunge grinding. *Ann. CIRP* 42(1): 383-387.

Choi S. Y., Lee H. and Yoo Y. (2010). The Impact of information technology and transactive memory systems on knowledge sharing, Application, And Team Performance: A Field Study. *MIS Quarterly* 34(4): 855-870.

Choi, T. and Shin, Y. (2007) Generalized intelligent grinding advisory system. *International Journal of Production Research* 45(8): 1901-1935.

Chow, H. K. H, Choy, K. L., Lee, W. B. and Chan, F. T. S. (2005). Design of a knowledge-based logistics strategy system. *Expert System Application* 29: 272-290.

Coff, R. W., Coff, D. C. and Eastvold, R. (2006). The knowledge-leveraging paradox: how to achieve scale without making knowledge imitable. *Academy of Management Review* (31)2: 452-465.

Coleman, D. (1997). Groupware: Collaborative Strategies for Corporate LANs and Intranet", Prentice Hall.

Coleman, D. (1999). Groupware: Collaboration and Knowledge Sharing Ch 12 in Knowledge Management Handbook, edited by Liebowitz, J. CRC Press LLC.

Conklin, E. J. (1996). Designing organizational memory: preserving intellectual assets in a knowledge economy. *Glebe Creek*, MD: CogNexus Institute. Available at: http://cognexus.org/dom.pdf [Accessed April 2006]

Connolly, T. and Begg, C. (1998). *Database Systems*. 2nd Edition, Addison Wesley.

Davenport, T. H., Prusak, L. and Wilson, J. H. (2003). Who's bringing you hot ideas and how are you responding? *Harvard Business Review*. 81(2): 59-64.

Davenport, T. H. and Prusak, L. (1998). *Working knowledge: How organisations manage what they know*. Harvard Business School Press.

Denning, S. (1998). *What is knowledge management?* (A background paper to the World Development Report 1998). www.stevedenning.com Available at: http://www.stevedenning.com/knowledge.htm [Accessed May 2005].

Du Plessis, M. (2007). Knowledge Management: What Makes Complex implementation Successful. *Journal of Knowledge Management* 11(2): 91-101.

Elkaffas, S. M. and Toony, A. A. (2006). Applications of Genetic Programming in Data Mining, World Academy of Science, Engineering and Technology 17 2006.

Fayyad, U., Piatetsky-Shapiro, G. and Smyth, P. (1996a). From Data Mining to Knowledge Discovery. *American Association for Artificial Intelligence* 37-54.

Fayyad, U., Piatetsky-Shapiro, G. and Smyth, P. (1996b). *Knowledge discovery and data mining: Towards a unifying framework*. The second international conference on Knowledge Discovery and Data Mining. Portland. AAAI Press. 82-88.

Fayyad, U., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R. (1996). *Advances in Knowledge Discovery and Data Mining*. MIT Press. Cambridge. MA.

Ferrada, X. and Serpell, A. (2009). Knowledge management and the construction Industry *Revista De La Construction* 8(1): 46-58.

Freitas, A. A. (1997). A genetic programming framework for two data mining tasks: classification and generalized rule induction, Genetic Programming: Proc. 2nd Annual Conf. (Stanford University, July 1997), 96-101. Morgan Kaufmann.

Gallego, I. (2007). Intelligent centerless grinding: global solution for process instabilities and optimal cycle design. *Annals of the CIRP* 56(1): 347-352.

Da Silva Garza, A. G., Gómez, A., Franzoni Velázquez, A. L. and Morales, V. C. (2007). Promoting productivity in manufacturing companies in developing countries: An information system for managing and querying knowledge bases in the automotive industry in Mexico. *Information Technology for Development* 13: 253-268.

Gillingham, H. and Roberts, B. (2006). Implementing Knowledge Management: A Practical approach. *Journal of Knowledge Management Practice* 7(1). (Available at http://www.tlainc.com/articl107.htm [Accessed Janurary 2011]).

Gilbert, R. J., Goodacre, R., Shann, B., Kell, D. B., Taylor J. and Rowland. J. J. (1998). Genetic programming-based variable selection for high-dimensional data. In Genetic Programming: Proceedings of the Third Annual Conference, pp. 109-115, Wisconsin, USA, Morgan Kaufmann.

Glasser, P. (1999). Knowledge factors, CIO magazine. (Available at http://www.cio.com.au/article/107326/knowledge_factor/ [Accessed 2005]).

Gold, A., Malhotra, A. and Segars, A. (2001). Knowledge Management: An Organizational Capabilities Perspective. *Journal of Management Information Systems* 18(1): 185-214.

Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Wesley Longman, Inc.

Garvin, D. A. (2003). *Building a Learning Organization*. Harvard Business Review on Knowledge Management. Harvard Business School Press. Cambridge Mass.

Grinding Data Book. (1992). *Universal Grinding Wheel Co. Ltd*.UK.

Grossman, D. and Ophir, F. (2001). Integrating structured data and text. *Intelligent Enterprise* 4(14) : 22-23.

Grover, V. and Davenport, T. H. (2001). General perspectives on knowledge management: Fostering a research agenda. *Journal of Management Information systems success* 18: 5-11.

Gupta, R., Shishodia, K. S. and Sekhon, G. S. (2001). Optimization of grinding process parameters using enumeration method. *Journal of Materials Processing Technology* 112(1): 63-67.

Gurteen, D. (1999). Creating a Knowledge Sharing Culture. *Knowledge Movement Magazine* 2(5).

Hanzlicek, P., Spidlen, J. and Heroutova, H. (2004). User interface of MUDR Electronic Health Record. *Studies In Health Technology And Informatics* 105: 190-201.

Harris, R. J. (2009). Improving tacit knowledge transfer within SMEs through e-collaboration. *Journal of European Industrial Training* 33(3): 215-231.

Hargadon, A. and Sutton, R. I. (2000). Building an innovation factory. *Harvard Business Review* 78(3): 157-66.

Havens, C. and Knapp, E. (1999). Easing into knowledge management. *Strategy & Leadership* 27(2): 4-10.

He, M. and Li, Y. (2010). Exploiting distributed cognition to make tacit knowledge explicating. *Journal of Software Engineering & Applications* 3: 273-279.

Hecker, R. L. and Liang S. Y. (2003). Predictive modeling of surface roughness in grinding. *International Journal of Machine Tools & Manufacture* 43:755-761.

Hildreth, P. M. and Kimble C. (2002). The duality of knowledge. *Information Research* 8(1).

Hildreth, P., Wright, P. and Kimble, C. (1999). Knowledge management: are we missing something? *In The 4th UKAIS Conference, York:* 347-356.

Hildreth, P. C. (2000). Communities of Practice in the Distributed International Environment. *Journal of Knowledge Management* 4(1): 27-37. MCB, University Press.

Horton, N. J. and Kleinman, K. P. (2007). Much Ado About Nothing: A Comparison of Missing Data Methods and Software to Fit Incomplete Data Regression Models. *The American Statistician* 61(1): 79-92.

Hou, Y., Li, C. and Zhou Y. (2010). Applications of High-Efficiency Abrasive Process with CBN Grinding Wheel. *Engineering* 2: 184-189.

Hou, Z. B. and Komanduri, R. (2003). On the mechanics of the grinding process – Part I. Stochastic nature of the grinding process. *International Journal of Machine Tools & Manufacture* 43: 1579-1593.

Hui, S. and Jha, G. (2000). Data mining for customer service support. *Information & Management* 38: 1-13.

Ibrahim, J. G., Chen, M.-H., Lipsitz, S. R. and Herring, A. H. (2005). Missing-data methods for generalized linear models: A comparative review. *Journal of the American Statistical Association* 100: 332-346.

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for Clustering Data*. Englewood Cliffs. NJ: Prentice-Hall.

Jones, R. (2003). Measuring the benefits of knowledge management at the financial services authority: a case study. *Journal of Information Science* 29: 475-487.

Kalpakjian, S. (1991). *Manufacturing Processes for Engineering Materials*, 2nd edition. Wesley Publishing Company.

Keijzer, M. (2004). Scaled symbolic regression. *Genetic Programming and Evolvable*

Machines 5(3): 259-269.

Kegg, R. L. (1983). Industrial Problems in Grinding. *Anna's of the CIRP* 32(2): 559.

King, R. I. and Hahn, R. S. (1986). *Handbook of Modern Grinding Technology*. Chapman and Hall. New York.

Kimble, C., Hildreth, P. and Wright, P. (2001). Communities of practice: going Virtual. *In: Knowledge Management and Business Innovation*, edited by Y. Malhota, 216-230. Hershey. PA: Idea Group Publishing.

Kimble, C., Li, F. and Barlow, A. (2000). Effective Virtual Teams Through Communities of Practice, Glasgow. University of Strathclyde. Strathclyde Business School.

Kreitner, R. and Kinicki, A. (1992). *Organizational Behavior*. Homewood. IL: Richard D. Irwin.

Koza, J. R. (1992). *Genetic Programming on the programming of computers by means of natural selection*. New York MIT Press.

KPMG Consulting (2000). Knowledge management research report 2000. Available at: www.kpmgconsulting.com [Accessed April 2007].

Kwak, J., Sim, S. and Jeong, Y. (2006). An analysis of grinding power and surface roughness in external cylindrical grinding of hardened SCM440 steel using the response surface method. *International Journal of Machine Tools & Manufacture* 46: 304-312.

Knowledgebase. http://www.knowledgbase.net [Accessed November 2004].

Kuivanen, R. (2008). The future of the manufacturing industry in Europe. *International Journal of Productivity and Performance Management* 57(6): 488-493.

Landqvist, F. and Teighland, R. (2005). Collective action in electronic networks of practice: an empirical study of three online social structures. *Proceedings of the Second Communities and Technologies Conference*, Besselaar edited by Michelis, G. D., Preece, J. and Simone C., XV 467.

Leave, J. and Wenger, E. (1991). *Sustained learning: legitimate peripheral participation*. Cambridge University Press, Cambridge.

Lee, C. W. (2000). *Intelligent modeling and optimisation of grinding process*. PhD. thesis. Purdue University, USA.

Lee, C. W. and Shin, Y. C. (2000). Evolutionary modelling and optimization of grinding processes. *International Journal of Production Research* 38(12): 2787-2813.

Lee, C. W., Choi, T. and Shin, Y. C. (2003). Intelligent model-based optimisation of the surface grinding process for heat treated 4140 steel alloys with Aluminium oxide grinding wheels. *Journal of Manufacturing and Engineering* 125: 65-76.

Lee, H. and Choi, B. (2003). Knowledge management enablers, processes, *and* organizational performance*: an* Integrative view *and* empirical examination. *Journal of Management Information Systems* 20(1): 179-228.

Leonard, D. and Sensiper, S. (1998). The role of tacit knowledge in group innovation. *California Management Review* 40(3): 112-132.

Li, Y. (1996). Intelligent Selection of Grinding Conditions. PhD thesis, Department of Mechanical Engineering. Liverpool John Moores University, UK.

Li, Y., Rowe, W. B. and Mills, B. (1999). Study and selection of grinding conditions part 1: grinding conditions and selection strategy. *Proc Instn Mech Engrs* 213(B): 119-129.

Li, Y., Rowe, W. B., Chen, X. and Mills, B. (1999). Study and selection of grinding conditions part 2: a hybrid intelligent system for selection of grinding conditions. *Proc Instn Mech Engrs* 213(B): 131-142.

Little, R. J. A. (1992). Regression with missing X's: A review. *Journal of the American Statistical Association* 87: 1227-1237.

Lindsay, R. P. and Hahn, S. (1973). On the surface finish - metal removal relationship in precision grinding. *Annals of the CIRP* 22(1): 105-106.

Lyer, S. and Aronson, E. J. (2000). Knowledge management: practices and challenges, *Industrial Management and Data Systems* 100(1): 17-21.

Machining Data Handbook (1980). *Machinability Data Centre*. 3rd Edition. Vol. 2. Metcut Research Associates Inc. USA.

Magic. www.remedy.com/solutions/magic [Accessed November 2004].

Maimon, O. and Rokack, L. (2005). Data mining and knowledge discovery handbook, Chapter 20 - Evolutionary algorithms for data mining, Springer.

Malkin, S. (1989). *Grinding Technology Theory and Applications of Machining with*

Abrasives. Ellis Harwood Limited.

Malkin, S. and Guo, C. (2008). Grinding Technology: Theory and Applications of Machining with Abrasives (second ed.). 257-280.

Marwick, A. D. (2001). Knowledge management technology. *IBM System Journal* 40(4): 814-829.

Meso, P. and Smith, R. A. (2000). Resource-based view of organizational knowledge management systems. *Journal of Knowledge Management* 4(13): 224-234.

Michalski, R. S. (2003). *Knowledge Mining: A Proposed New Direction*, Sanken Symposium on Data Mining and Semantic Web, Osaka University, Japan.

Michalski, R. S. (1983). Theory and Methodology of Inductive Learning, in *Machine Learning: An Artificial Intelligence Approach*, R. S. Michalski, J. Carbonell and T. Mitchell (Eds.), pp. 83-134, Morgan Kaufman Publishing Co., Palo Alto.

Midha, P. S., Zhu, C. B. and Trmal, G. J. (1991). Optimum selection of grinding parameters using process modeling and a knowledge-based system approach. *Journal of Material Processing Technology* 28: 189-198.

Molnar, K. K. and Sharda, R. (1996). Using the internet for knowledge acquisition in expert systems development: a case study. *Journal of Information Technology* 11: 223-234.

Morgan, M. N., Cai, R., Guidotti, A., Allanson, D. R., Moruzzi, J. L. and Rowe, W. B. (2007). Design and implementation of an intelligent grinding assistant system. *International Journal of Abrasive Technology* 1(1): 106-135.

Mukherjee, I. and Ray, P. K. (2008), A modified tabu search strategy for multiple-response grinding process optimisation, *Int. J. Intelligent Systems Technologies and Applications* 4(1/2): 97-122.

Nandi, A. K. and Banerjee, M. K. (2005). FBF-NN-based modelling of cylindrical plunge grinding process using a GA. *Journal of Materials Processing Technology* 162(163): 655-664.

Nandi, A. K. and Pratihar, D. K. (2004). Automatic design of fuzzy logic controller using a genetic algorithm—to predict power requirement and surface finish in grinding. *Journal of Materials Processing Technology* 148: 288-300.

Nonaka, I. and Takeuchi, H. (1995). *The Knowledge-Creating Company: how*

Japanese companies create the dynamics of innovation. Oxford University Press.

Nonaka, I. (1991). The knowledge creating company. *Harvard Business Review* 69 (Nov-Dec): 96-104.

Novo. http://www.novosolutions.com/knowledge-base-software/ [Accessed November 2004].

O'Dell, C. and Grayson, C. J. (1998). If we only knew what we know: identification and transfer of internal best practices. *California Management Review* 40(3): 154-174.

Oliveira, J. F. G., Silva, E. J., Guo, C. and Hashimoto, F. (2009). Industrial challenges in grinding, *CIRP Annals - Manufacturing Technology* 58: 663-680.

Ovaska, E., Evesti, A., Henttonen, K., Palviainen, M. and Aho, P. (2010). Knowledge based quality-driven architecture design and evaluation. *Information & Software Technology* 52(6): 577-601.

Park, Y., Kim, S. and Lee, S. (2006). Knowledge management system for fourth generation R & D: KNOWVATION. *Technovation* 26: 595-602.

Parlby, D. and Taylor, R. (2000). The power of knowledge: a business guide to knowledge management. Available at: www.kpmgconsulting.com/index.html [Accessed April 2006].

Pattinson, E. J. and Lyon, J. (1974). The collection of data for the assessment of grinding wheel dressing treatment. *Proceedings of The Fifteenth International Machine Tool Design and Research* 317.

Peters, J., Snoeys, R. and Decneut, A. (1976). The proper selection of grinding conditions in cylindrical plunge grinding. *Ann. CIRP* 26(1): 387-394.

Petrash, G. (1996). Managing knowledge assets for value. *Knowledge Based Leadership Conference*, Boston.

Pigott, T. D. (2001). A review of methods for missing data. *Educational Research and Evaluation* (7)4: 353-383.

Polanyi, M. (1966). *The Tacit Dimension*. Doubleday Anchor, New York, NY.

Prusak, L. (2001). Where did knowledge management come from? *IBM Systems Journal* 40(4): 1002-1007.

Raamesh, L. and Uma G. V. (2009). Knowledge Mining of Test Case System.

International Journal on Computer Science and Engineering 2(1): 69-73.

Raghunathan, T. E. (2004). What do we do with missing data? Some options for analysis of incomplete data. *Annual Review of Public Health* 25: 99-117.

Ranky, P. G. (1990). *Manufacturing database management and knowledge based expert systems*. CIMware Limited, Guildford, Surrey, England.

Riel, A. and Boonyasopon, P. (2009). A Knowledge Mining Approach to Document Classification. *AIJSTPME* 2(3): 1-10.

Rowe, W. B., Bell, W. F. and Brough, D. (1987). Limit chart for high removal rate centreless grinding. *Int. J. Mach. Tools Mf.* 27(1): 15-25.

Rowe, W. B., Li, Y., Inasaki, I. and Malkin, S. (1994). Applications of artificial intelligence in grinding. Keynote paper. *Ann. CIRP* 43(2): 521-531.

Rowe, W. B., Pettit, J. A., Boyle, A. and Moruzzi, J. L. (1988). Avoidance of thermal damage in grinding and prediction of the damage threshold. *Ann. CIRP* (1): 327-330.

Ruggles, R. (1997). *Tools for knowledge management: an introduction*, Ruggles, R., ed. Knowledge Management Tools. Butterworth-Heinemann.

Sánchez, M. S. and Palacios, M. A. (2008). Knowledge-based manufacturing enterprises: evidence from a case study. *Journal of Manufacturing Technology Management* 19(4): 447-468.

Sáenz, J., Aramburu, N. and Rivera, O. (2009). Knowledge sharing and innovation performance: A comparison between high-tech and low-tech companies. *Journal of Intellectual Capital* 10(1): 22-36.

Sakakura, M. and Inasaki, I. (1993). Intelligent data base for grinding operations. *Annals of the CIRP Manufacturing Technology* 42(1): 379-382.

Salmon, S. C. (1992). *Modern Grinding Process Technology*. McGraw Hill. USA.

Santosus, M. and Surmacz, J. (2001). The ABCs of Knowledge Management, CIO magazine. Available at http://www.cio.com/research/knowledge/edit/kmabcs.html [Accessed April 2005].

Saravanan, R. and Sachithanandam, M. (2001). Genetic algorithm (GA) for multivariable surface grinding process optimisation using a multi-objective function model. *International Journal of Advanced Manufacturing Technology* 17: 330-338.

Saravanan, R., Asokan, P. and Sachidanandam, M. (2002). A multi objective genetic algorithm (GA) approach for optimization of surface grinding operations. *International Journal of Machine Tools & Manufacture* 42: 1327-1334.

Sarker, R., Abbass, H. and Newton, C. (2002). Introducing data mining and knowledge discovery, In Sarker, R. and Abbass, H. and Newton, C. (Eds.), *Heuristics and Optimisation for Knowledge Discovery*, Idea Group Publishing.

Sena, J. and Shani, A. B. (1999). Intellectual capital and knowledge creation: towards an alternative framework, Ch 8 in *Knowledge Management Handbook*, edited by Liebowitz, J., 1999 by CRC Press LLC.

Sharratt, M. and Usoro, A. (2003). Understanding knowledge sharing in online communities of practice. *Electronic Journal of Knowledge Management* 1: 187-196.

Shaw, M. L. G. and Woodward, J. B. (1990). Modelling expert knowledge, *Knowledge Acquisition* 2(3): 179-206.

Shin, Y., Subrahmanya, N. and Choi, T. (2008). Generalized practical models of cylindrical grinding processes. *International Journal of Machine Tools & Manufacture* 48: 61-72.

SHRM (2009). Leveraging HR and knowledge management in a challenging economy. *HR Magazine* (54)6: 1-9.

Silva, S. (2004). A Genetic Programming Toolbox for Matlab - version 3.ECOS - Evolutionary and Complex Systems Group, University of Coimbra, Portugal, http://gplab.sourceforge.net/.

Simoudis, E. (1996). Reality check for data mining. *IEEE Expert* 11: 26-33.

Singh, M. D., Shankar, R., Narain, R. and Kumar, A.(2006) .Survey of knowledge management practices in Indian manufacturing industries. *Journal of Knowledge Management* 10(6): 110-128.

Slowik, A. and Slowik, J. (2008). Multi-objective optimization of surface grinding process with the use of evolutionary algorithm with remembered Pareto set. *International Journal of Advanced Manufacturing Technology* 37: 657-669.

Sririvason, K. (1981). Application of the regeneration spectrum method to workpiece regenerative chatters in grinding. *Proceedings of the 9th North America Manufacturing Research Conference* 283-289, University Park.

Stepien, P. (2009). A probabilistic model of the grinding process. *Applied*

Mathematical Modeling 33: 3863-3884.

Stuart, A. (1996). 5 Uneasy Pieces, CIO magazine. Available at http://www.cio.com/archive/060196_uneasy_1.html [Accessed April 2005].

Szuianski, G. (1996). Exploring internal stickiness: Impediments to the transfer of best practice within the firm. *Strategic Management Journal* 17(10): 27-43.

Taminiau, Y., Smit, W. and De Lange, A. (2009). Innovation in management consulting firms through informal knowledge sharing. *Journal of Knowledge Management* 13(1): 42-55.

Tang, N. and Vemuri, V. (2004). Web-based knowledge acquisition to impute missing values for classification. Proceeding of the IEEE/WIC/ACM International Conference on Web Intelligence.

Teigland, R. (2003). *Knowledge networking: structure and performance in networks of practice.* Ph.D. Stockholm: Stockholm School of Economics.

Teigland, R. and Wasko, M. (2004). Extending richness with reach: participation and Knowledge Exchange in Electronic Networks of Practice. Chapter 19 *in* Hildreth, P. & Kimble, C. (eds.), *Knowledge Networks: Innovation Through Communities of Practice*, London: Idea Group Inc.

Tiago, M. T., Couto J. P., Tiago F. G. and Vieira A. C. (2007). Knowledge management: An overview of European reality. *Management Research News* 30(2): 100-114.

Tonshoff, H. K., Peters, J., Inasaki, T. and Paul, T. (1992). Modeling and simulation of grinding processes. *Annals of CIRP* 41(2): 677-688.

Tyrolit http://www.tyrolit.com/page.cfm?vpath=index [visited Dec 2010].

Van der Spek, R. and Spijkervet, A. (1997). A Knowledge management: dealing intelligently with knowledge. *Knowledge Management and Its Integrative Elements*. Liebowitz, J. & Wilcox, L., eds. CRC Press.

Verkerk, J. and Pekelharing, A. J. (1979). The influence of the dressing operation on productivity in precision grinding. Keynote paper. *Ann. CIRP* 28(2): 487-495.

Vishnupad, P. and Shin, Y. C. (1998). Intelligent optimization of grinding processes using fuzzy logic. *Proc Instn Mech Engrs* 212(B): 647-660.

Vladislavleva, K., Veeramachaneni, K., Burland, M., Parcon, J. and O'Reilly, U. (2010). Knowledge mining with genetic programming methods for variable selection in flavor design. In Proceedings of GECCO 2010, pp. 941-948.

Wang, E., Klein G. and Jiang J. J. (2007). IT support in manufacturing firms for a knowledge management dynamic capability link to performance. *International Journal of Production Research* 45(11): 2419-2434.

Wenger, W. P., Najdawi, M. K. and Chung, Q. B. (2001). Selection of knowledge acquisition techniques based upon the problem domain characteristics of production and operations management expert systems. *Expert Systems* 18(2): 76-85.

Watson, I. (1995). Developing industrial awareness of case based reasoning technology in Europe. ECN p. 5. Publishing Company.

Wenger, E. (1998). Communities of Practice Learning as a Social System. Published in the Systems Thinker.

Wenger, E. (2004). Knowledge management as a doughnut: shaping your knowledge strategy through communities of practice. *Ivey Business Journal*. Available at http://www.ibj.ca/view_article.asp?intArticle_ID=465 [Accessed April 2007].

Wenger, E., McDermott, R. and Snyder, W. (2002). Cultivating Communities of Practice: a guide to managing knowledge, Harvard Business School Press, USA.

Wiig, K. (1997). Knowledge Management: Where did it come from and where will it go? *Expert Systems with Applications*, Pergamon Press/Elsevier, 14.

Winterthurusa http://www.winterthurusa.net/ss_odplunge.html [visited Dec 2010].

Wong, M. L. (2001). A flexible knowledge discovery system using genetic programming and logic grammars, Decision Support Systems 31: 405-428.

Wong, W. L. P. and Radcliffe, D. F. (2000). The tacit nature of design knowledge. *Technology Analysis and Strategic Management* 12(4): 493-512.

Yang, C.-L. and Wei, S.-T. (2010). Modelling the performance of CoP in knowledge management. *Total Quality Management* (21)10: 1033-1045.

Yang, H. L. (1995). Information/Knowledge acquisition methods for decision support systems and experts systems. *Information Processing and Management* 31: 47-58.

Zhou, X. and Xi, F. (2002). Modeling and predicting surface roughness of the grinding process. *International Journal of Machine Tools & Manufacture* 42: 969-977.

Appendix A

Rules for Rule Based Reasoning

Rule 1	
IF	workpiece material is tool steel
AND	material hardness > 50 Rc
AND	wheel is conventional wheel (WA80JV)
AND	roughness < 0.8μm
THEN	wheel speed=28-30m/s
AND	work speed= 0.3-0.5m/s
AND	feed rate $\leq 0.012/2 \times 0.3 \times 1000(\pi \times d_w)$
AND	dressing depth= 0.012-0.019 mm
AND	dressing lead = 0.1 mm/r
AND	Coolant= Emulsifiable oils- heavy duty / Chemical and synthetics-heavy duty

Rule 2	
IF	workpiece material is cast iron
AND	material hardness <50 Rc
AND	wheel is conventional wheel (C60KV)
AND	roughness < 0.8μm
THEN	wheel speed=28-33m/s
AND	work speed= 0.35-0.5m/s
AND	feed rate $\leq 0.025/2 \times 0.3 \times 1000(\pi \times d_w)$
AND	dressing depth= 0.012-0.019 mm

AND dressing lead = 0.1 mm/r

AND Coolant= Emulsifiable oils- heavy duty / Chemical and synthetics-heavy duty

Rule 3

IF workpiece material is cast iron

AND material hardness > 50 Rc

AND wheel is conventional wheel (C46MV)

AND roughness < 0.8μm

THEN wheel speed=28-33m/s

AND work speed= 0.35-0.5m/s

AND feed rate $\leq 0.013/2 \times 0.35 \times 1000(\pi \times d_w)$

AND dressing depth= 0.012-0.019 mm

AND dressing lead = 0.1 mm/r

AND Coolant= Emulsifiable oils- light duty / Chemical and synthetics-light duty

Rule 4

IF workpiece material is superalloys

AND wheel is conventional wheel

AND roughness < 0.8μm

THEN wheel speed=15-18m/s

AND work speed= 0.25-0.5m/s

AND feed rate $\leq 0.005/2 \times 0.25 \times 1000(\pi \times d_w)$

AND dressing depth= 0.012-0.019 mm

AND dressing lead = 0.1 mm/r

AND Coolant= oils- heavy

Rule 5

IF workpiece material is aluminium alloys (C46JV)

AND wheel is conventional wheel

AND	roughness < 0.8μm
THEN	wheel speed=28-33m/s
AND	work speed= 0.25-0.77m/s
AND	feed rate $\leq 0.013/2 \times 0.25 \times 1000(\pi \times d_w)$
AND	dressing depth= 0.012-0.019 mm
AND	dressing lead = 0.1 mm/r
AND	Coolant= Emulsifiable oils- light duty / Chemical and synthetics-light duty

Appendix B

Cylindrical exp. dataset (1)

Table 2 GP terminals for cylindrical exp. dataset (1)

Terminal set (data file x)					Expected Output Data file (Y)
Dressing depth	Dressing lead	Speed ratio	Geometrical contact	Equivalent chip thickness	Surface roughness (micro-m)
a_d (mm)	s_d (mm)	$\frac{v_w}{v_s}$	$a.d_e$ (mm)	h_{eq} (µm)	R_a (µm)
0.005	0.05	0.008	0.034694	0.01712	0.24
0.005	0.15	0.008	0.034694	0.01712	0.31
0.005	0.25	0.008	0.034694	0.01712	0.4
0.015	0.05	0.008	0.034694	0.01712	0.29
0.015	0.15	0.008	0.034694	0.01712	0.3
0.015	0.25	0.008	0.034694	0.01712	0.3
0.025	0.05	0.008	0.034694	0.01712	0.3
0.025	0.15	0.008	0.034694	0.01712	0.32
0.025	0.25	0.008	0.034694	0.01712	0.34
0.005	0.05	0.008	0.034694	0.01712	0.25
0.005	0.15	0.008	0.034694	0.01712	0.35
0.005	0.25	0.008	0.034694	0.01712	0.4
0.015	0.05	0.008	0.034694	0.01712	0.3
0.015	0.15	0.008	0.034694	0.01712	0.32
0.015	0.25	0.008	0.034694	0.01712	0.35
0.025	0.05	0.008	0.034694	0.01712	0.4
0.025	0.15	0.008	0.034694	0.01712	0.42

Run 1A

(Individual size100, Population size 200)

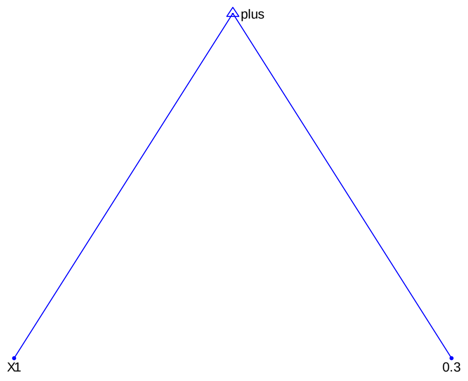
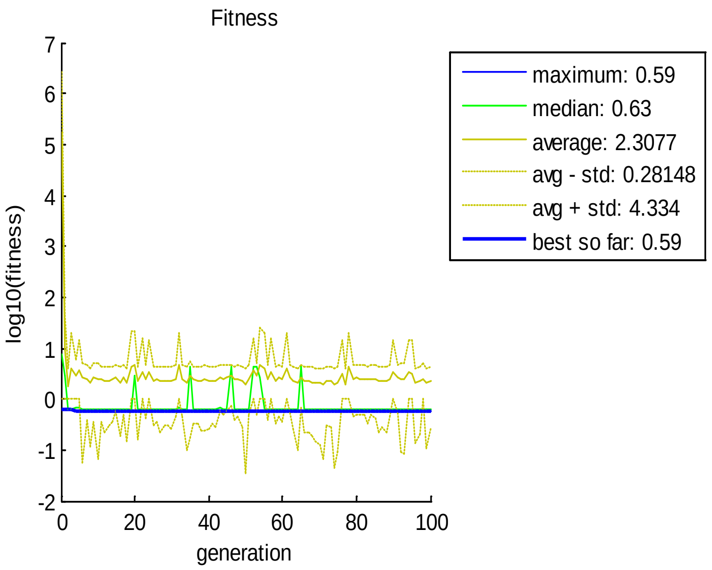


Figure 3 The best tree output for Cylindrical exp. dataset run (1A)



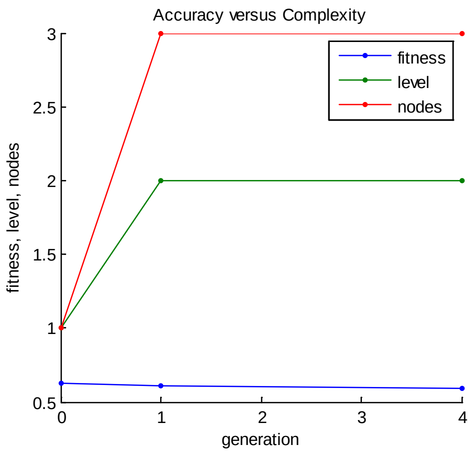


Figure 4Graphical output for cylindrical ex. dataset(1) run(1A)

Run 2A

(Individual size100, Population size 200)

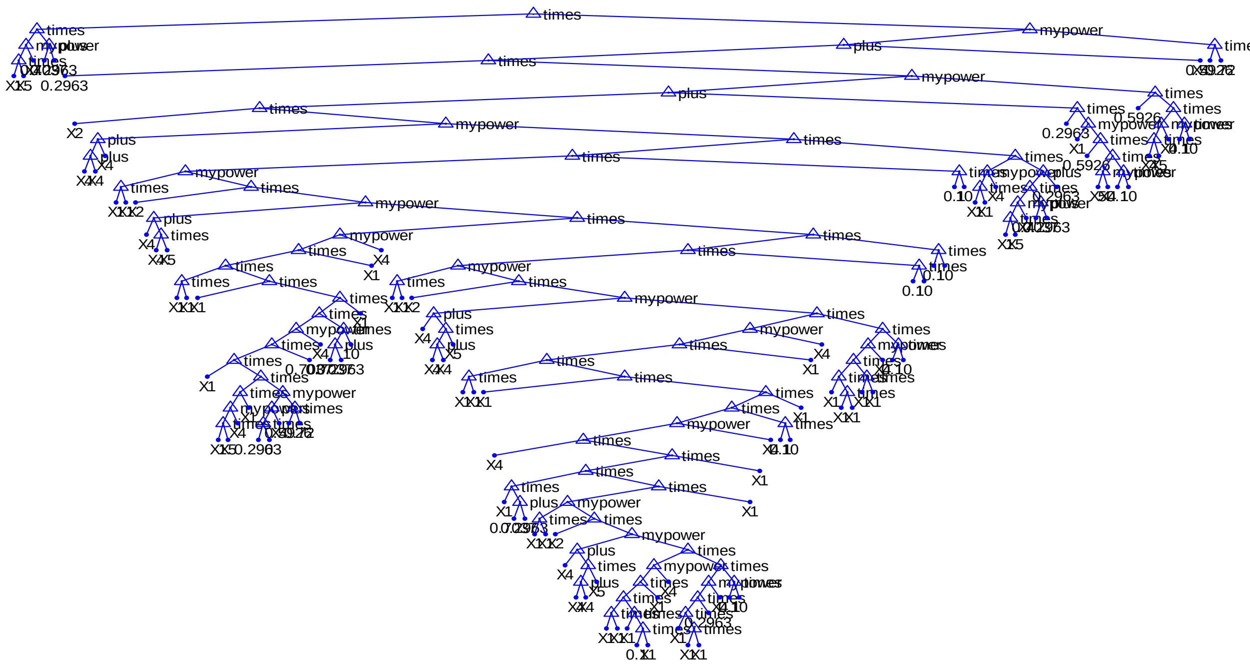
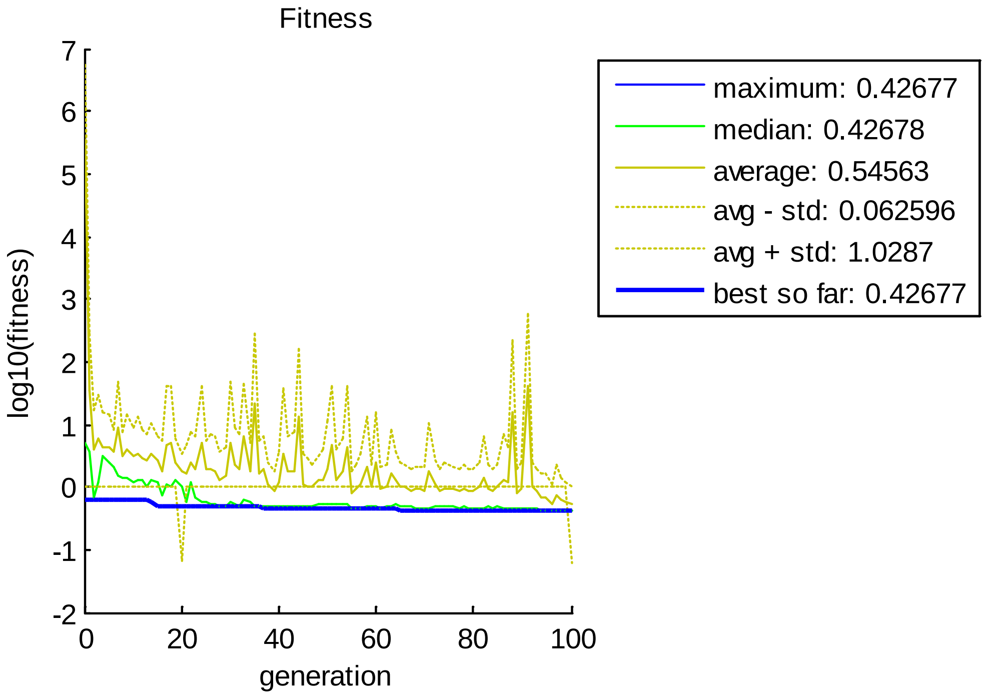


Figure 5 The best tree output for cylindrical exp. dataset (1) run (2A) using GP programming code page 276 for Table 1 terminal set as dataset X and expected output as dataset Y



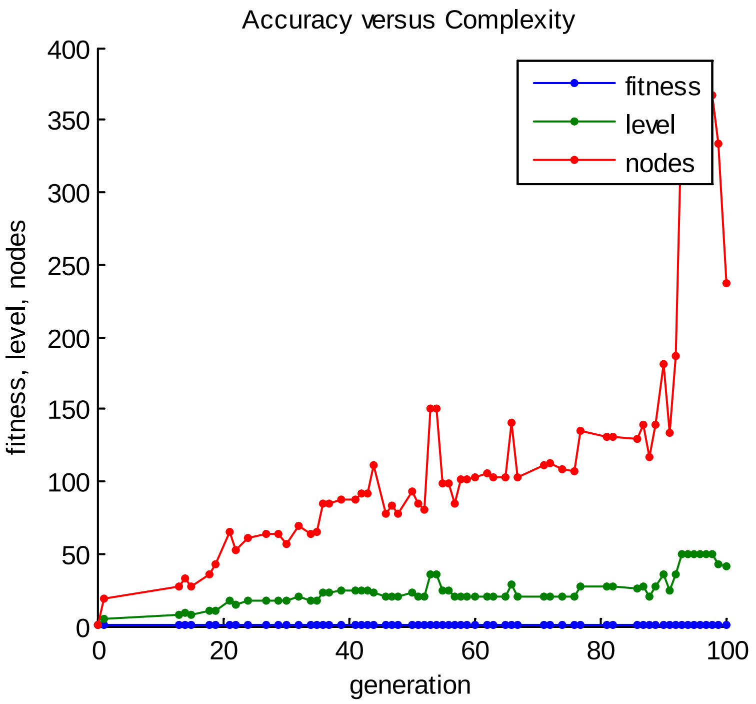


Figure 6 Graphical output for Cylindrical exp. dataset (1) run (2A)

Run 3A

(Individual size100, Population size 200)

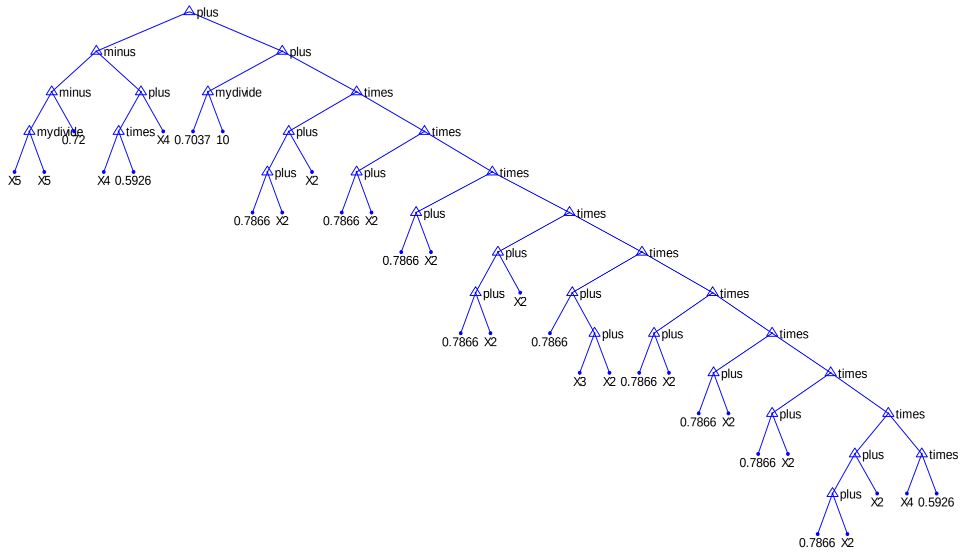


Figure 7 The best tree output for cylindrical exp. dataset(1) run 3A

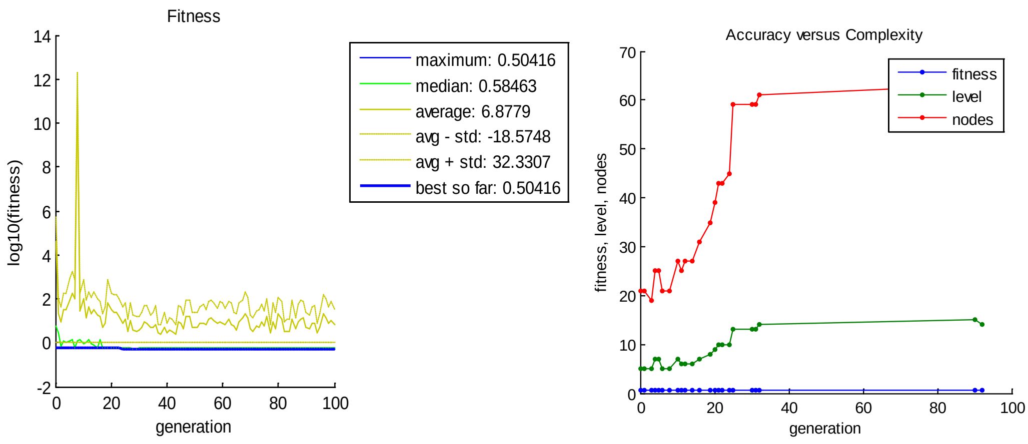


Figure 8 Graphical output for Cylindrical exp dataset(1) run 3A

Run1B

(Individual size100, Population size 300)

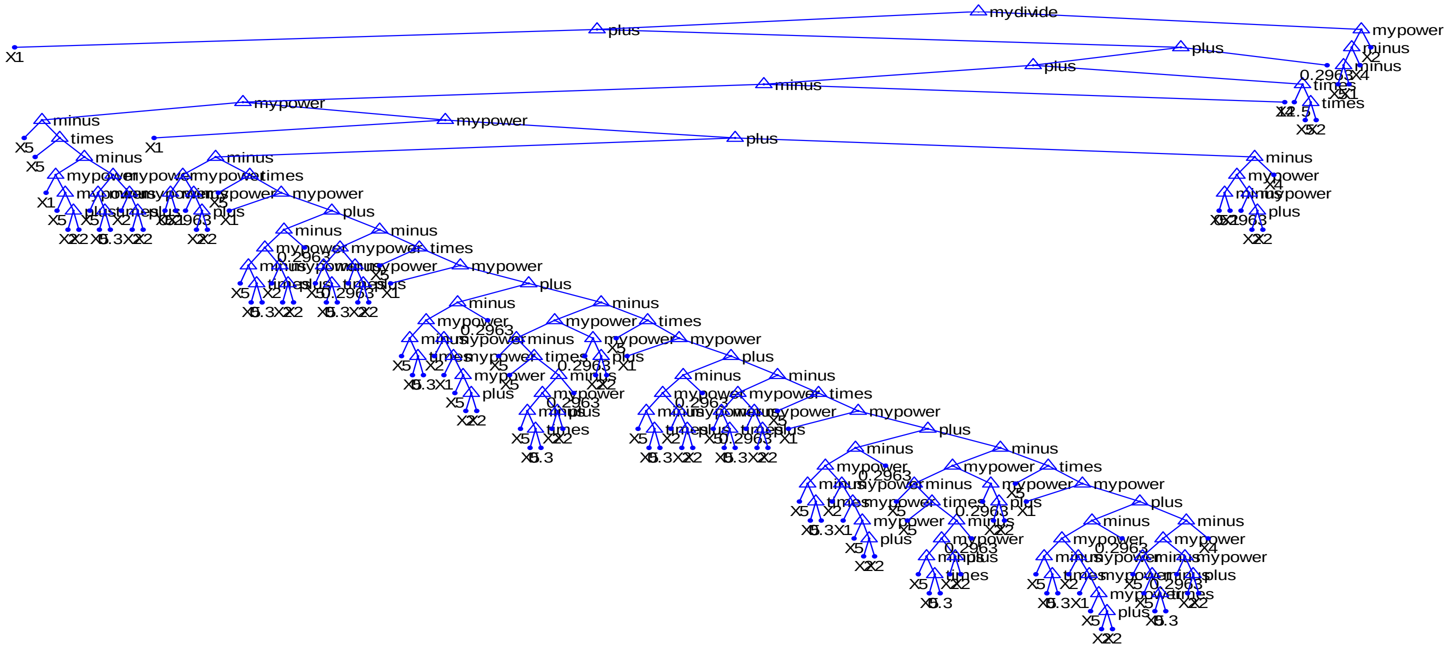


Figure 9 The best tree output for cylindrical exp. dataset (1) run 1B using GP programming code page 276 for Table 1 terminal set as dataset X and expected output as dataset Y

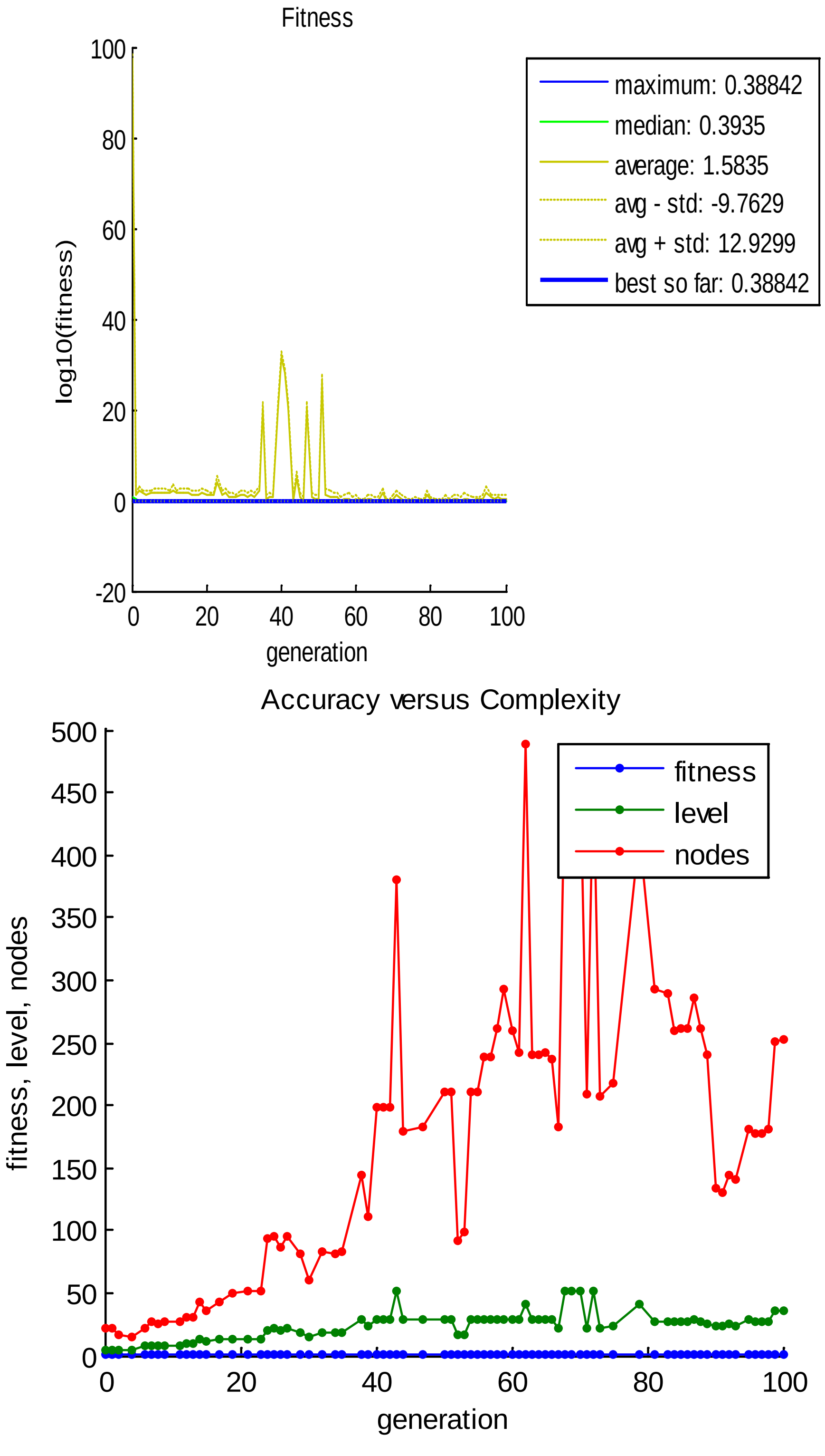


Figure 10 Graphical output for cylindrical exp. dataset (1) run 1B

Run 2B

(Individual size100, Population size 300)

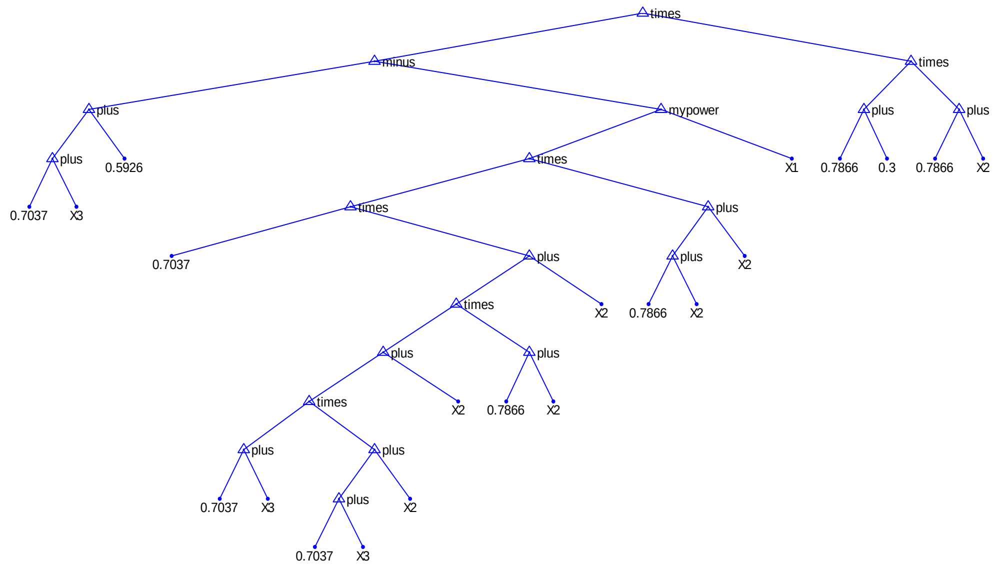


Figure 11 The best tree output for Cylindrical exp. dataset (1) run 2B

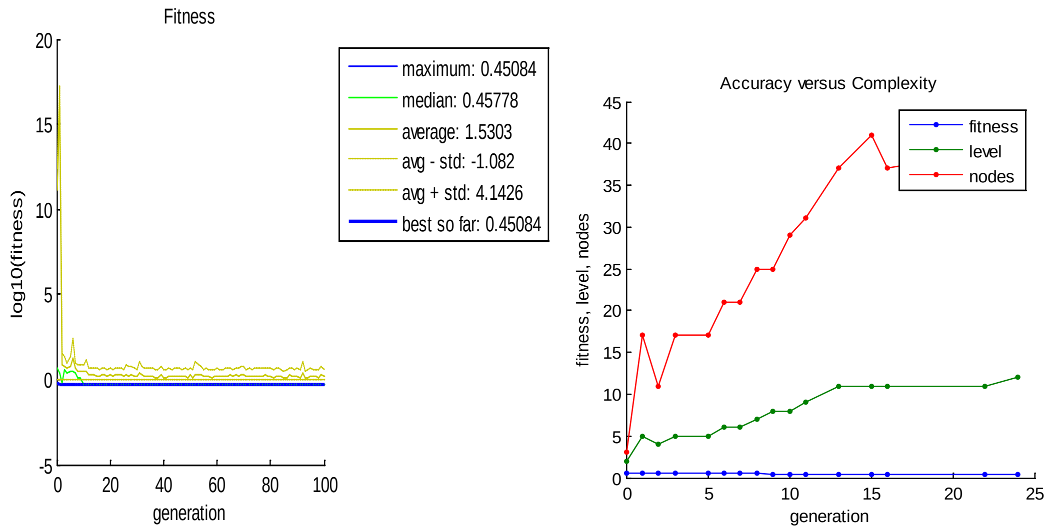


Figure 12 Graphical output for cylindrical exp. dataset(1) run 2B

Run 3B

(Individual size100, Population size 300)

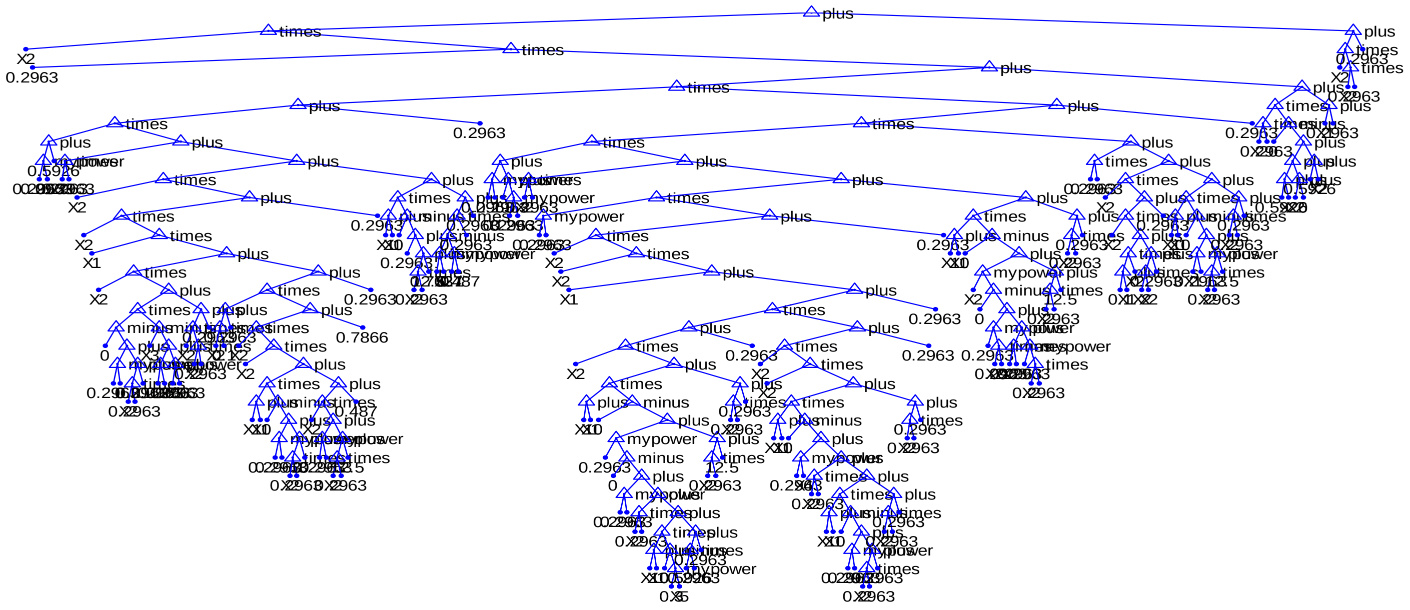
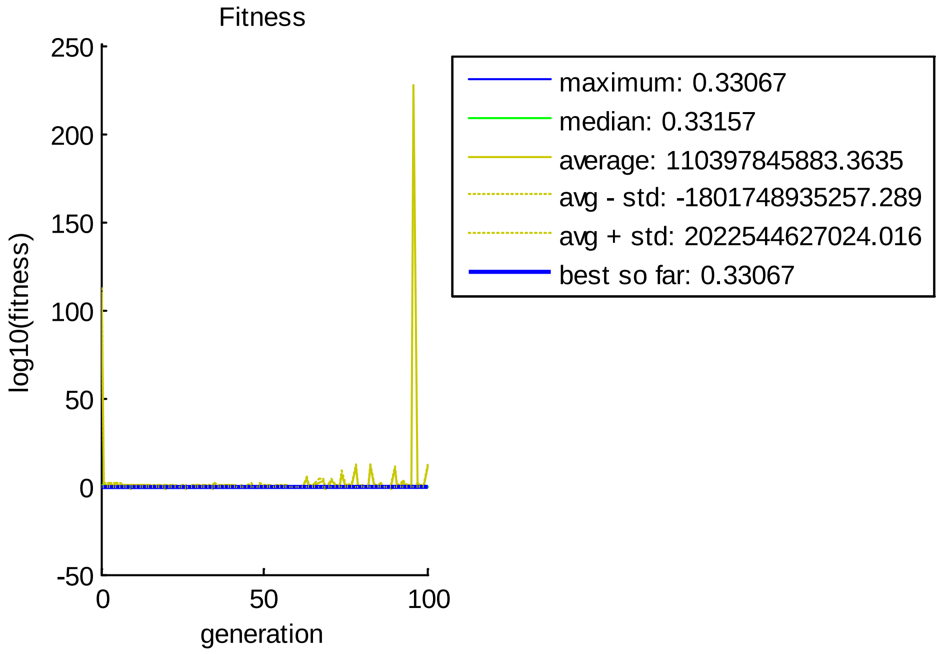


Figure 13 The tree output for cylindrical exp. dataset (1) run 3B using GP programming code page 276 for Table 1 terminal set as dataset X and expected output as dataset Y



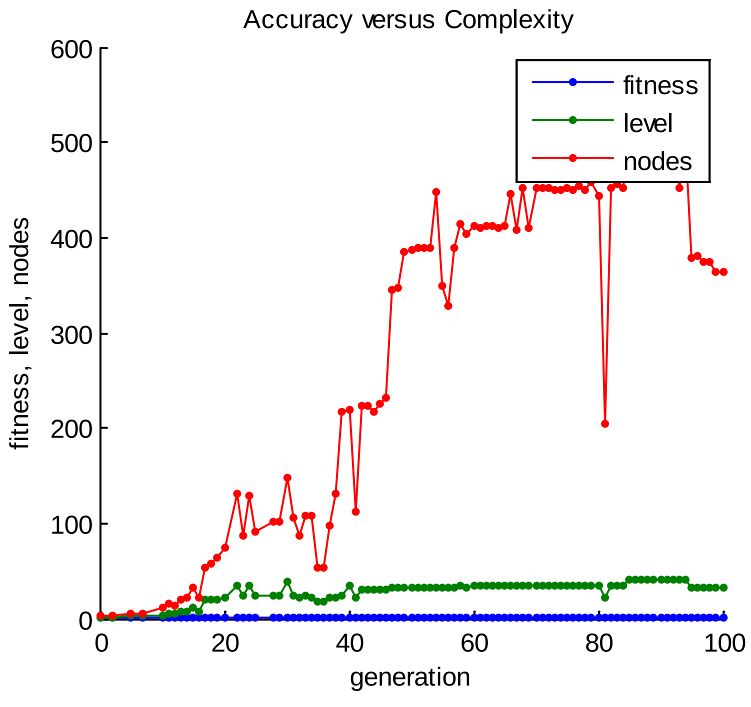


Figure 14 Graphical output for cylindrical exp. dataset (1) run 3B

Cylindrical cal. dataset (2)

Table 3GP terminals for cylindrical cal. dataset(2)

Terminal set Data file (x)					Expected output Data file (Y)
Dressing depth	Dressing lead	Speed ratio	Geometrical contact	Equivalent chip thickness	
a_d (mm)	s_d (mm)	$\frac{v_w}{v_s}$	$a.d_e$ (mm)	h_{eq} (µm)	R_a (µm)
0.025	0.1	0.011786	0.41304	0.1411	0.419
0.03	0.16	0.01377	0.5163	0.206557	0.557
0.036	0.22	0.015152	0.61956	0.272727	0.679
0.025	0.1	0.012727	0.88632	0.229091	0.481
0.03	0.16	0.017857	0.59088	0.214286	0.563
0.036	0.22	0.01082	0.7386	0.162295	0.586
0.025	0.16	0.015152	1.1295	0.272727	0.605
0.03	0.22	0.011786	0.753	0.141429	0.565
0.036	0.1	0.01377	0.94125	0.206557	0.463
0.025	0.22	0.017857	0.41304	0.214286	0.638
0.036	0.1	0.015152	0.5163	0.227273	0.475
0.03	0.16	0.011786	0.61956	0.212143	0.561
0.025	0.16	0.01082	0.7413	0.162295	0.51
0.036	0.22	0.015	0.88956	0.27	0.676
0.03	0.1	0.016393	0.59304	0.196721	0.458
0.025	0.22	0.015152	0.94125	0.227273	0.68
0.036	0.1	0.01082	1.1295	0.194754	0.455
0.03	0.16	0.012727	0.753	0.152727	0.511
0.025	0.16	0.01377	0.7386	0.206557	0.559
1.25	2.75	0.04918	0.5163	0.737705	2.21
2.33	2	0.081967	0.5163	1.229508	2.23

Run 1A

(Individual size 100, population size 200)

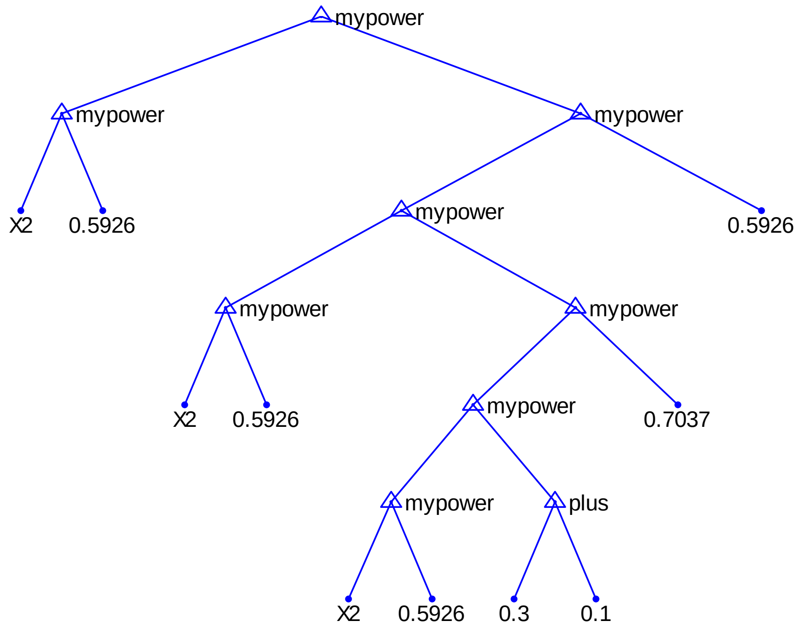


Figure 15The best tree output for cylindrical cal. dataset(2) run 1A

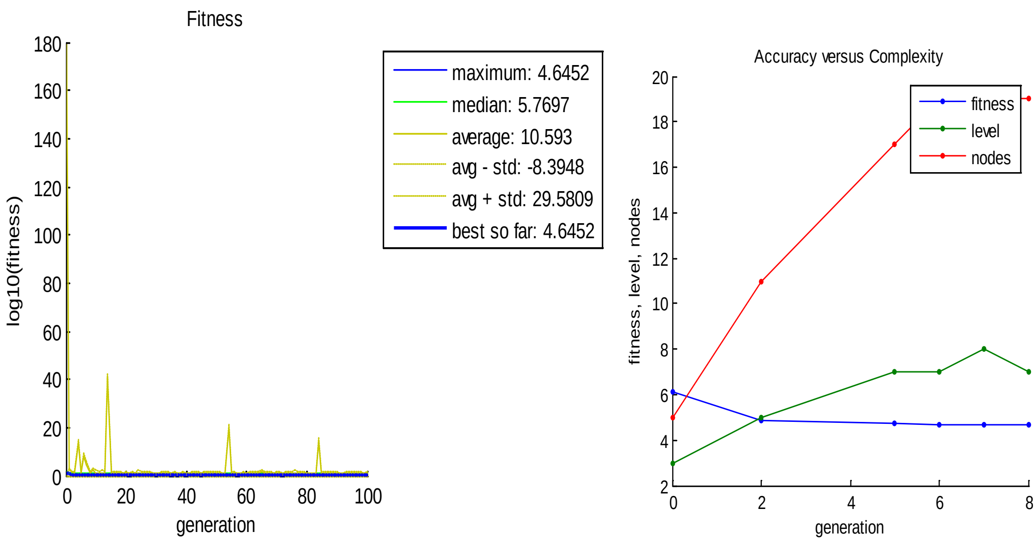


Figure 16 Graphical output for cylindrical cal. dataset(2) run 1A

Run 2A

(Individuals size 100, population size 200)

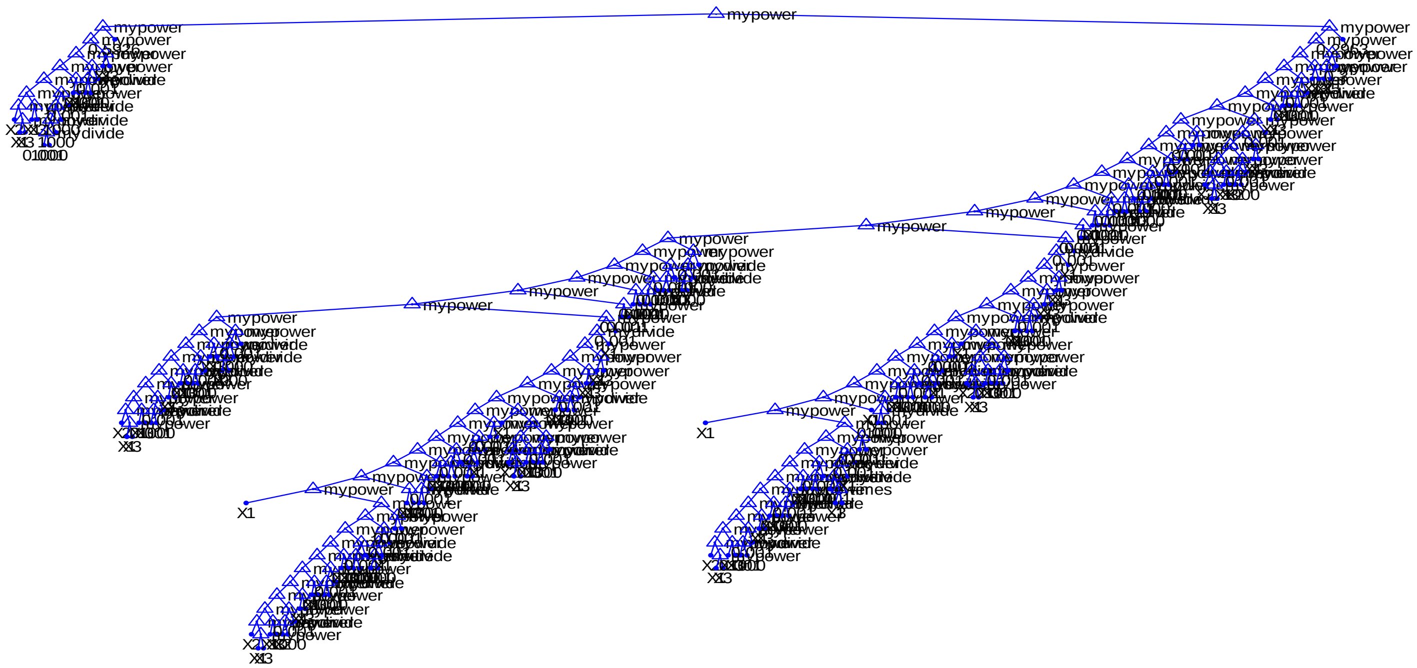
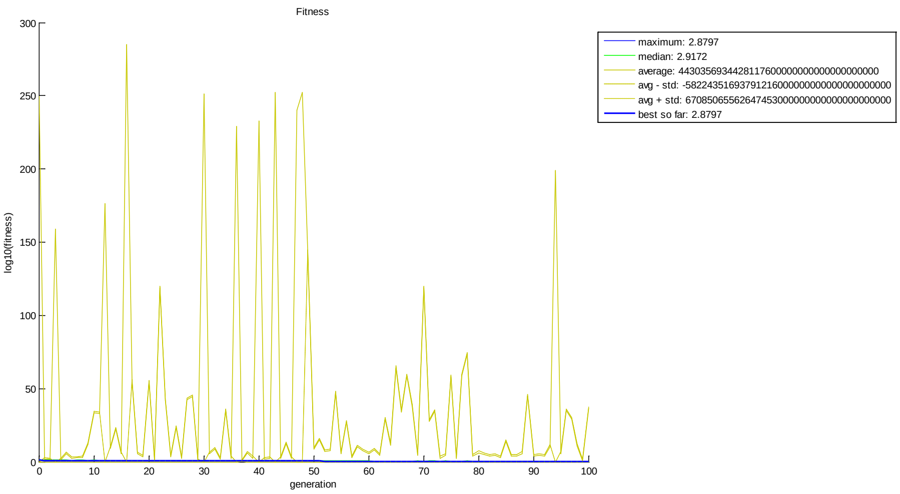


Figure 17 The tree output for cylindrical cal. dataset(2) run 2A using GP programming code page 276 for Table 2 terminal set as dataset X and expected output as dataset Y



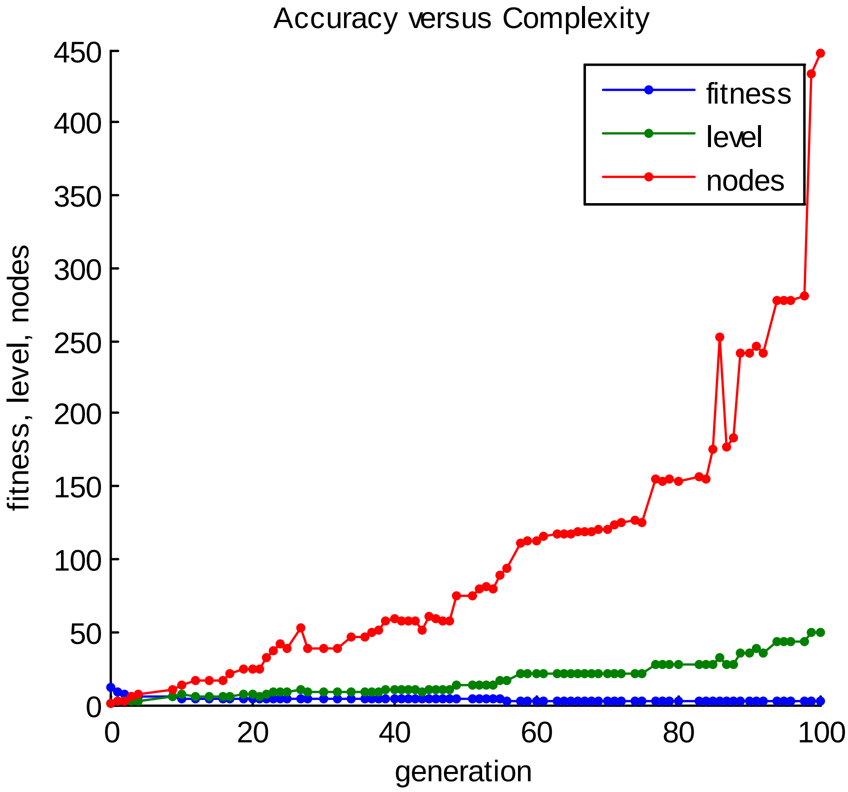


Figure 18 Graphical output for cylindrical cal. dataset(2) run 2A

Run 3A

(Individuals size 100, population size 200)

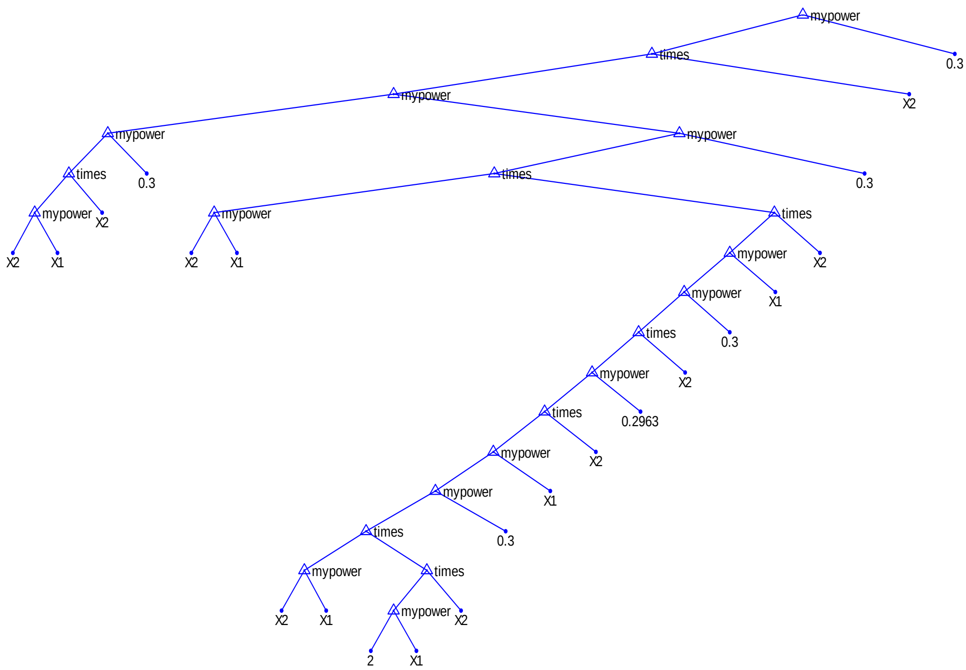


Figure 19 The best tree output for cylindrical cal. dataset(2) run 3A

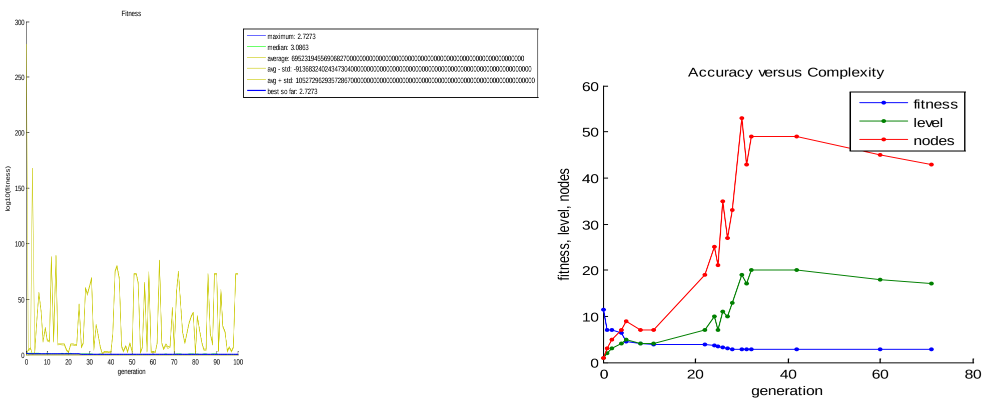


Figure 20 Graphical output for cylindrical cal. dataset (2) run 3A

Run 1B

(Individuals size100 ,Population size 300)

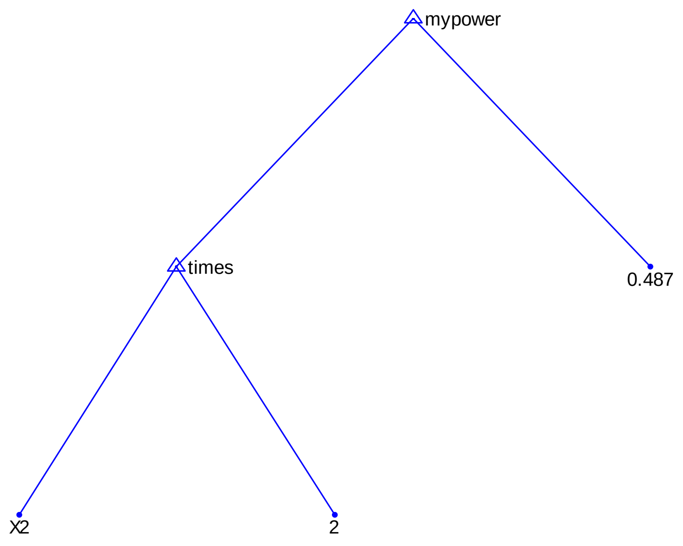


Figure 21 The best tree output for Cylindrical cal. dataset (2) run 1B

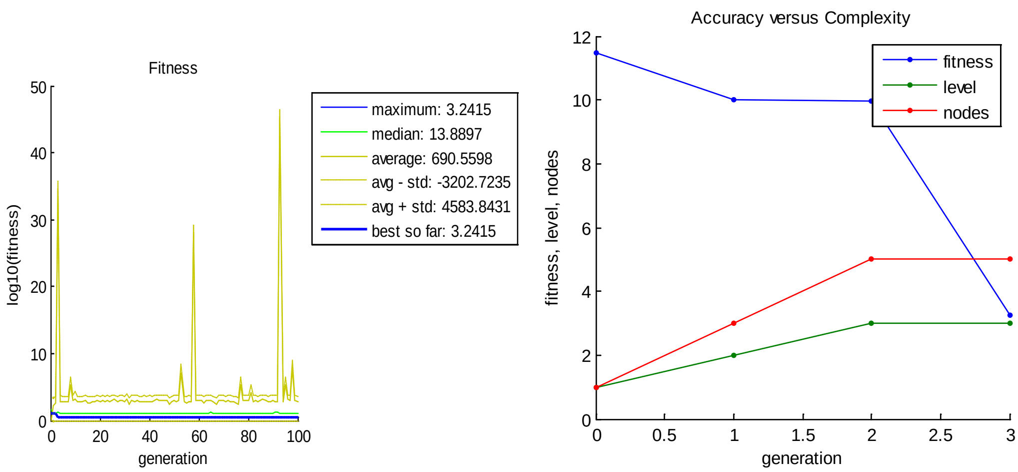


Figure 22 Graphical output for cylindrical cal. dataset (2) run1B

Run 2B

(Individuals size100 , Population size 300)

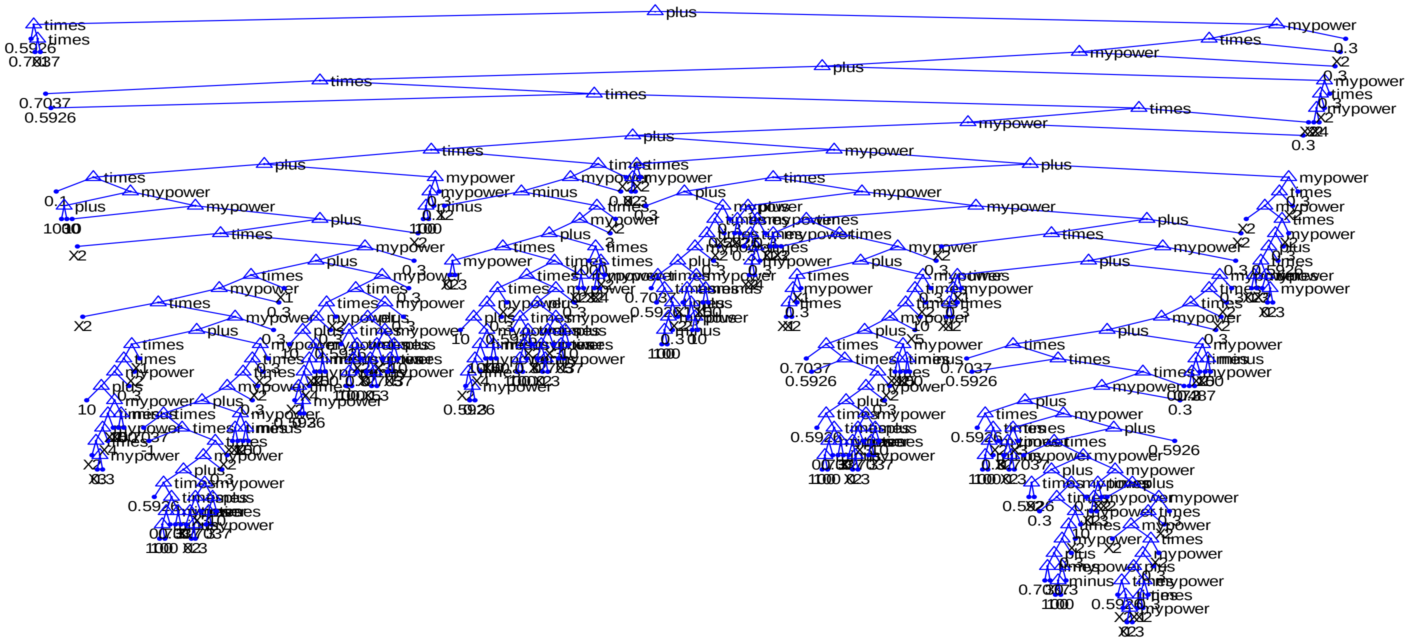


Figure 23 The best output tree for cylindrical cal. dataset (2) run2B using GP programming code page 276 for Table 1 terminal set as dataset X and expected output as dataset Y

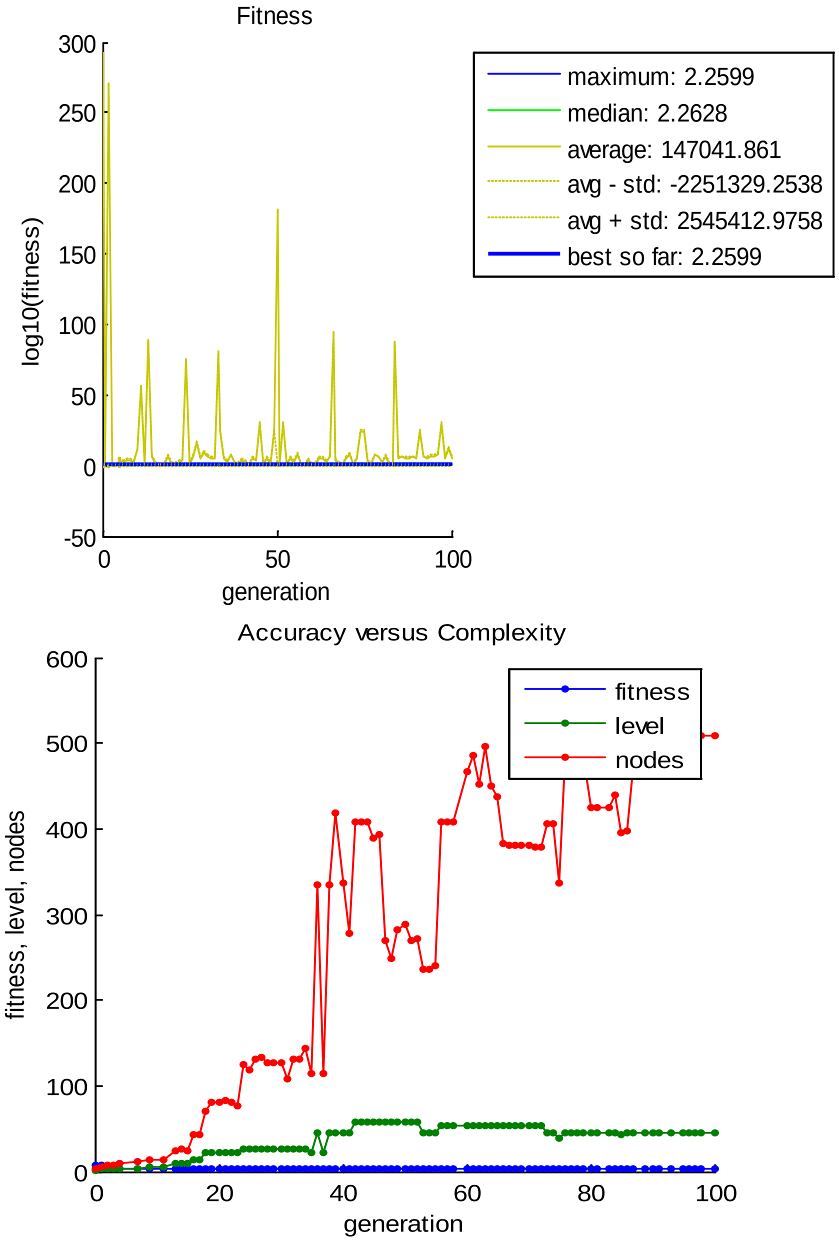


Figure 24 Graphical output for cylindrical cal. dataset(2) run 2B

Run 3B

(Individuals size100 , Population size 300)

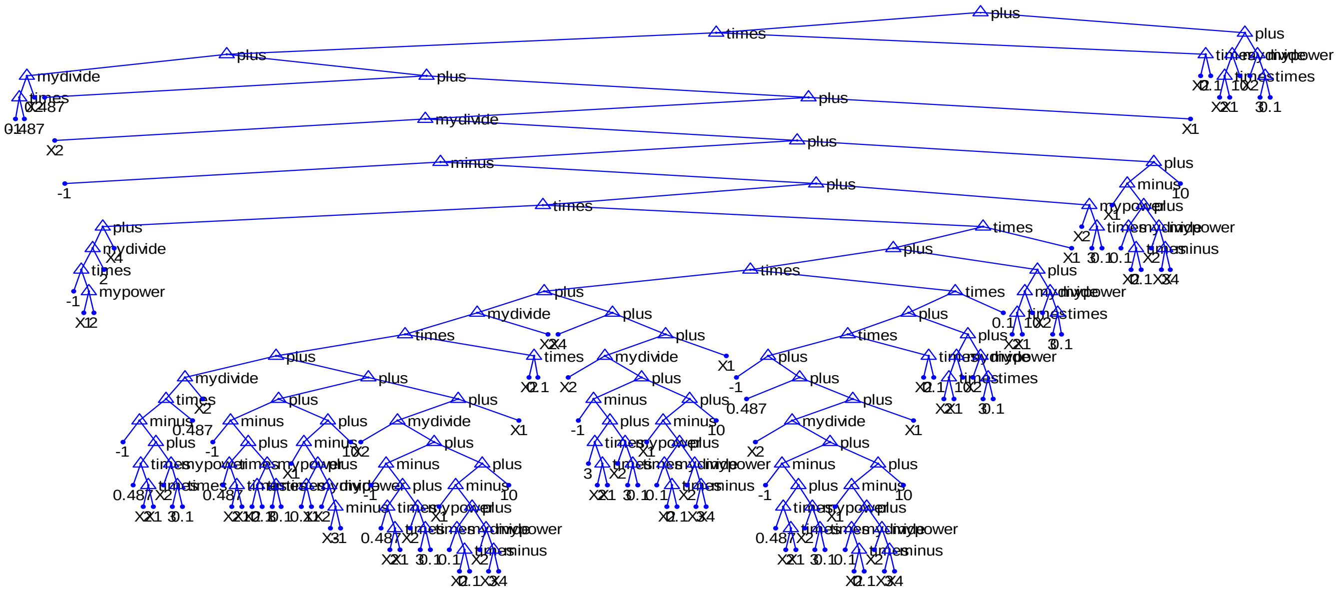


Figure 25 The best tree output for cylindrical cal. dataset(2) run 3B using GP programming code page 276 for Table 1 terminal set as dataset X and expected output as dataset Y

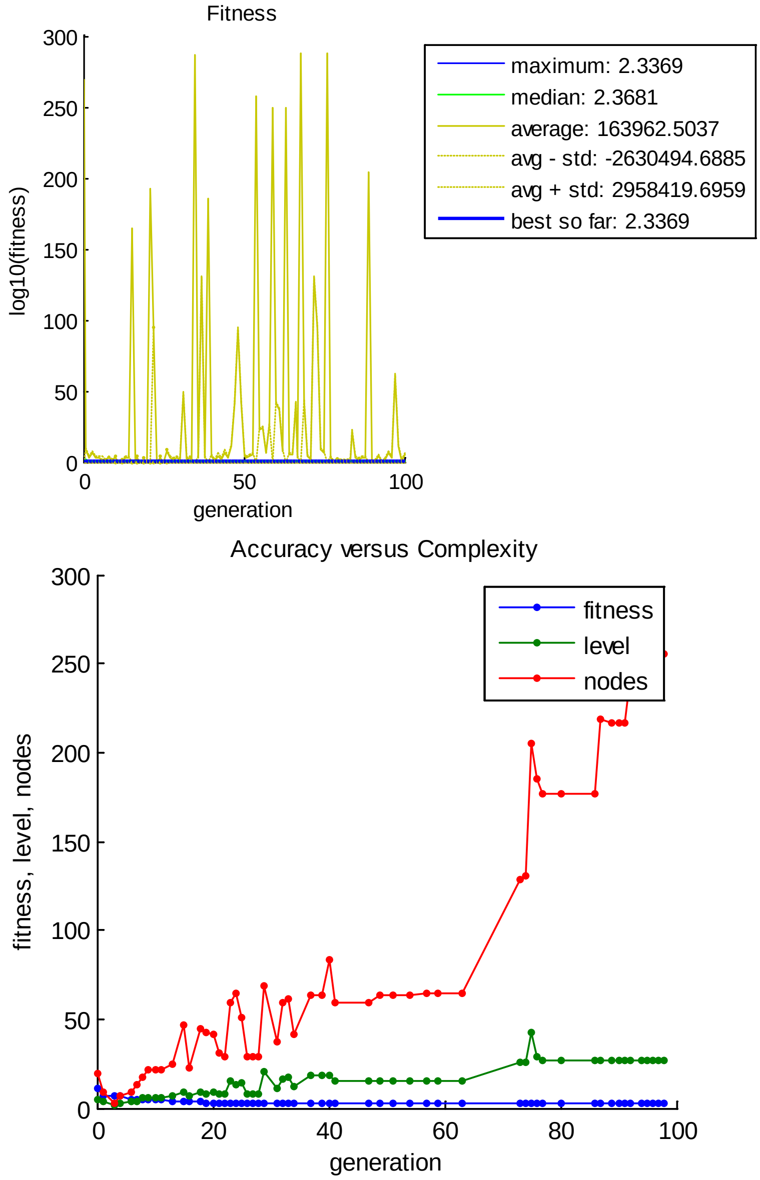


Figure 26 Graphical output for cylindrical cal. dataset(2) 3B

Surface grinding exp. dataset(3)

Table 4 Surface grinding exp. dataset(3)

Terminal Set Data file (X)						Expected output Data file (Y)
Dressing depth	Dressing lead	Speed ratio	St/bs	Geometrical contact	Equivalent chip thickness	Surface Roughness
a_d (µm)	s_d (mm)	$\frac{v_w}{v_s}$	$\frac{s_t}{b_s}$	$a.d_e$ (mm)	h_{eq} (µm)	R_a (µm)
20	0.12	0.00303	0.024016	2.54	3.03E-05	0.26
20	0.12	0.006061	0.022047	2.54	6.06E-05	0.32
20	0.12	0.009091	0.018898	2.54	9.09E-05	0.38
20	0.12	0.009091	0.019685	7.62	0.000273	0.37
20	0.12	0.006061	0.022047	7.62	0.000182	0.34
20	0.12	0.00303	0.024016	7.62	9.09E-05	0.26
20	0.12	0.00303	0.04252	2.54	3.03E-05	0.32
20	0.12	0.006061	0.037795	2.54	6.06E-05	0.41
20	0.12	0.009091	0.03937	2.54	9.09E-05	0.45
20	0.12	0.009091	0.03937	7.62	0.000273	0.46
20	0.12	0.006061	0.03622	7.62	0.000182	0.37
20	0.12	0.00303	0.041732	7.62	9.09E-05	0.35
20	0.12	0.00303	0.06063	2.54	3.03E-05	0.39
20	0.12	0.006061	0.057087	2.54	6.06E-05	0.51
20	0.12	0.009091	0.06063	2.54	9.09E-05	0.55
20	0.12	0.009091	0.06063	7.62	0.000273	0.61
20	0.12	0.006061	0.058661	7.62	0.000182	0.52
20	0.12	0.00303	0.058661	7.62	9.09E-05	0.39
8	0.12	0.006061	0.040157	5.08	0.000121	0.26
13	0.12	0.006061	0.040157	5.08	0.000121	0.33

Run 1A

(Individual size 100, population size 200)

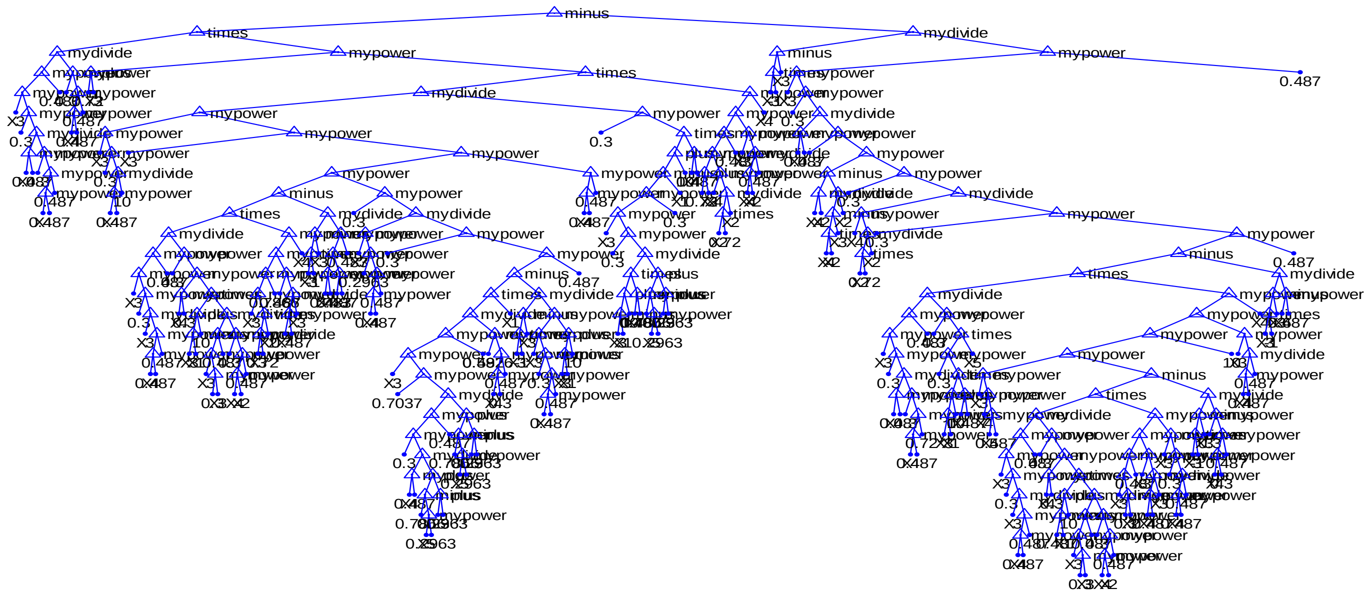


Figure 27 The best tree output for surface grinding exp. dataset (3) run 1A using GP programming code page 276 for Table 3 terminal set as dataset X and expected output as dataset Y

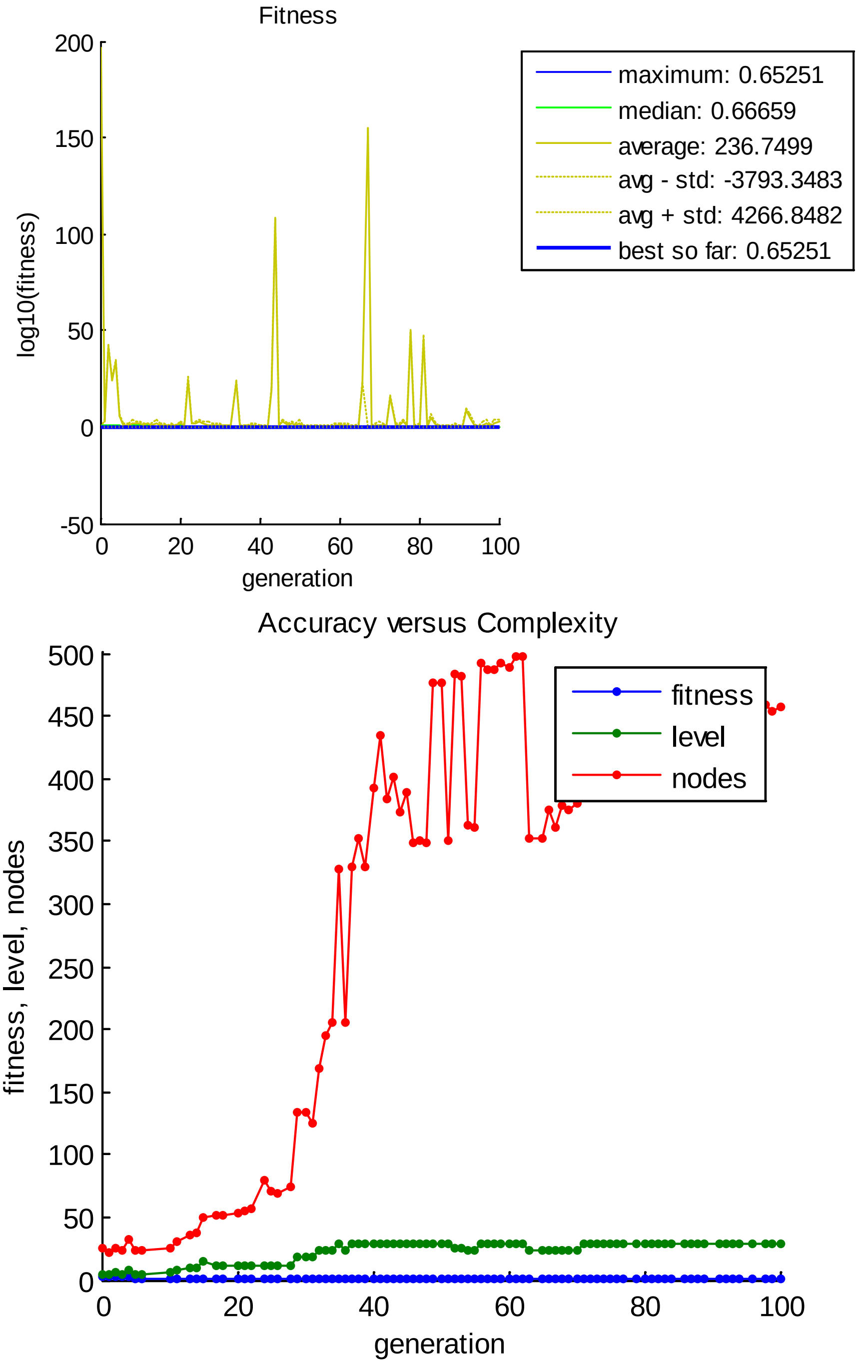


Figure 28 Graphical output for surface exp. dataset (3) run 1A

Run 2A

(Individual size 100, population size 200)

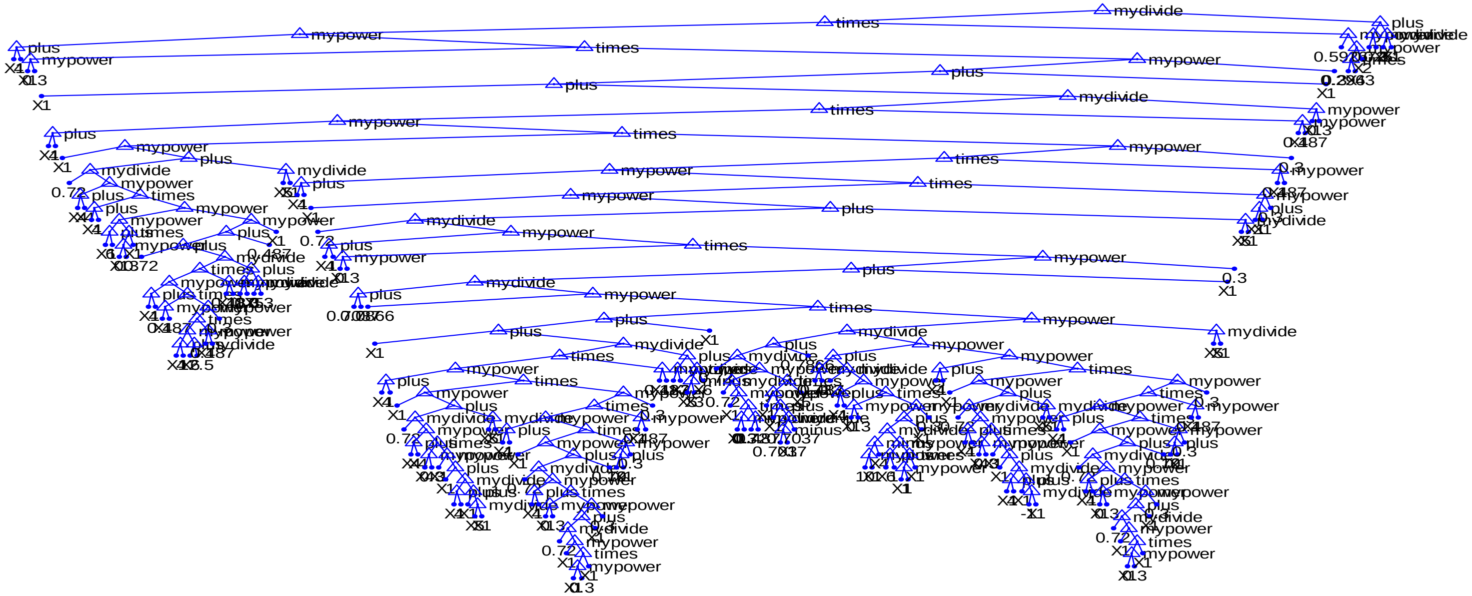


Figure 27 The best tree output for surface grinding exp. dataset (3) run 2A using GP programming code page 276 for Table 3 terminal set as dataset X and expected output as dataset Y

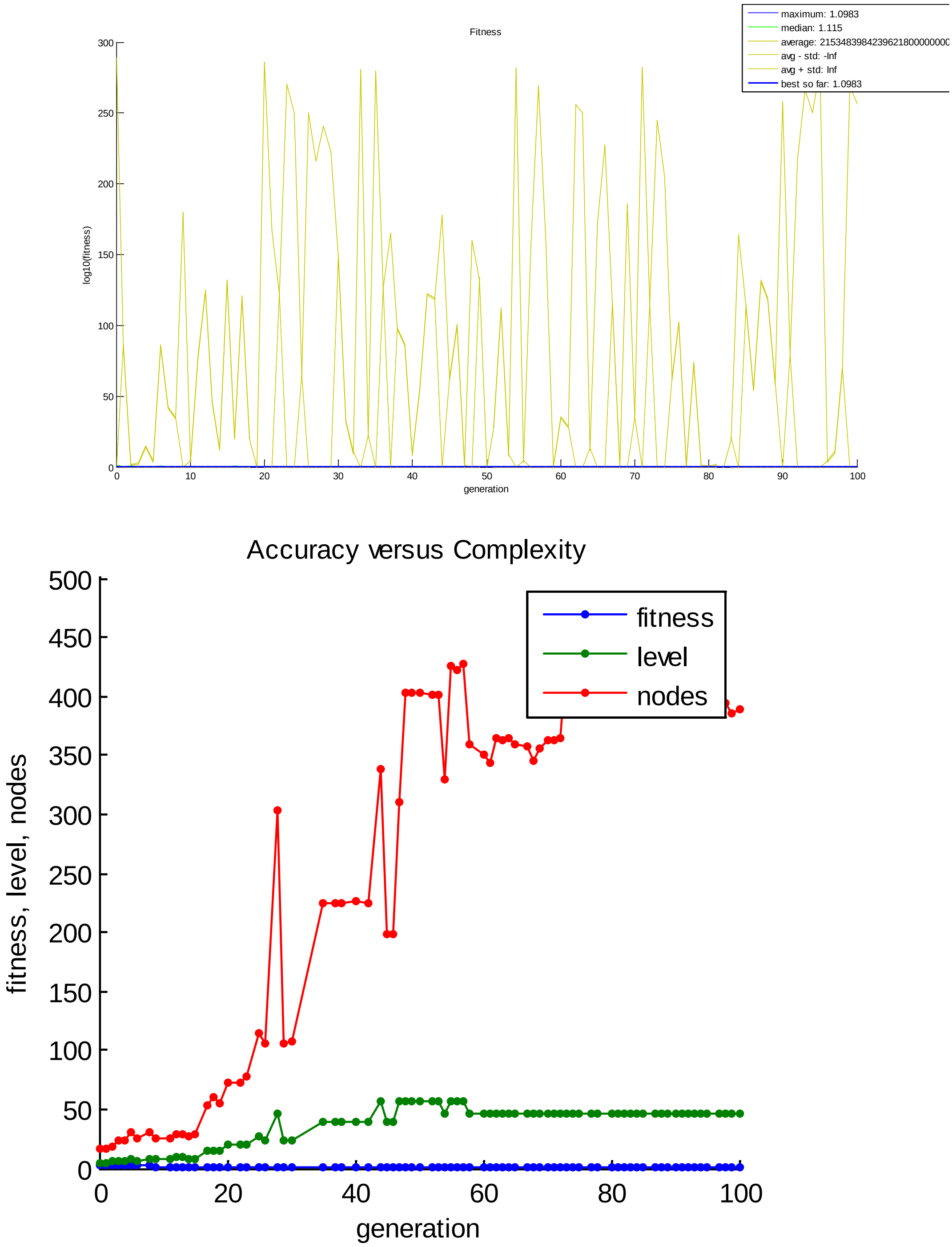


Figure 28 Graphical output for surface exp. dataset (3) run 2A

Run 3A

(Individual size 100, population size 200)

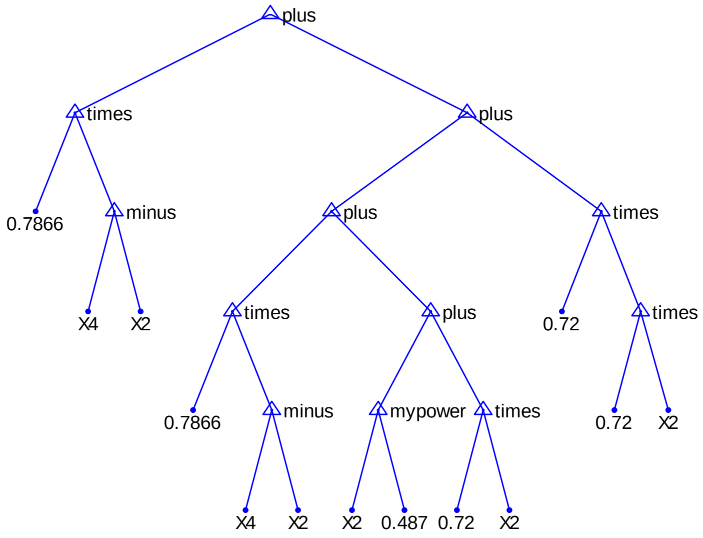


Figure 29 The best tree output for surface grinding exp. dataset (3) run 3A

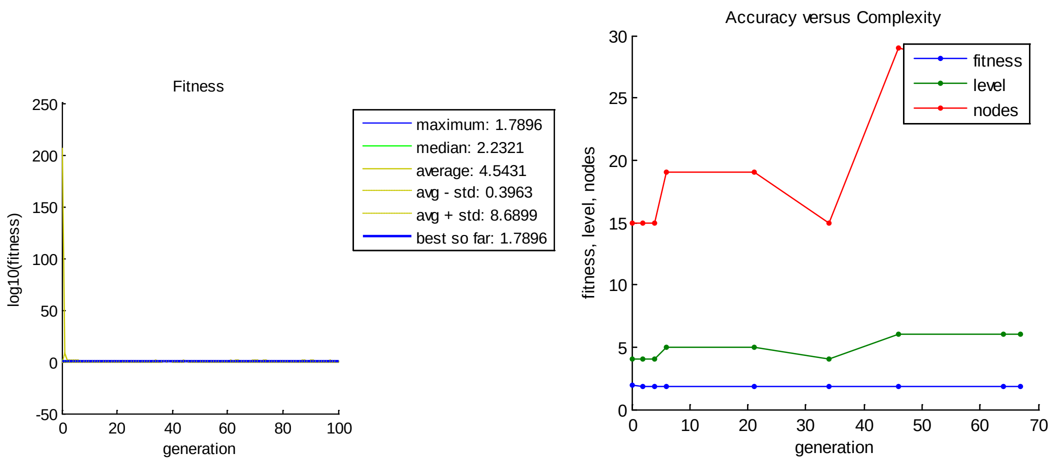


Figure 30 Graphical output for surface exp. dataset (3) run 3A

Run 1B

(Individuals 100, Population 300)

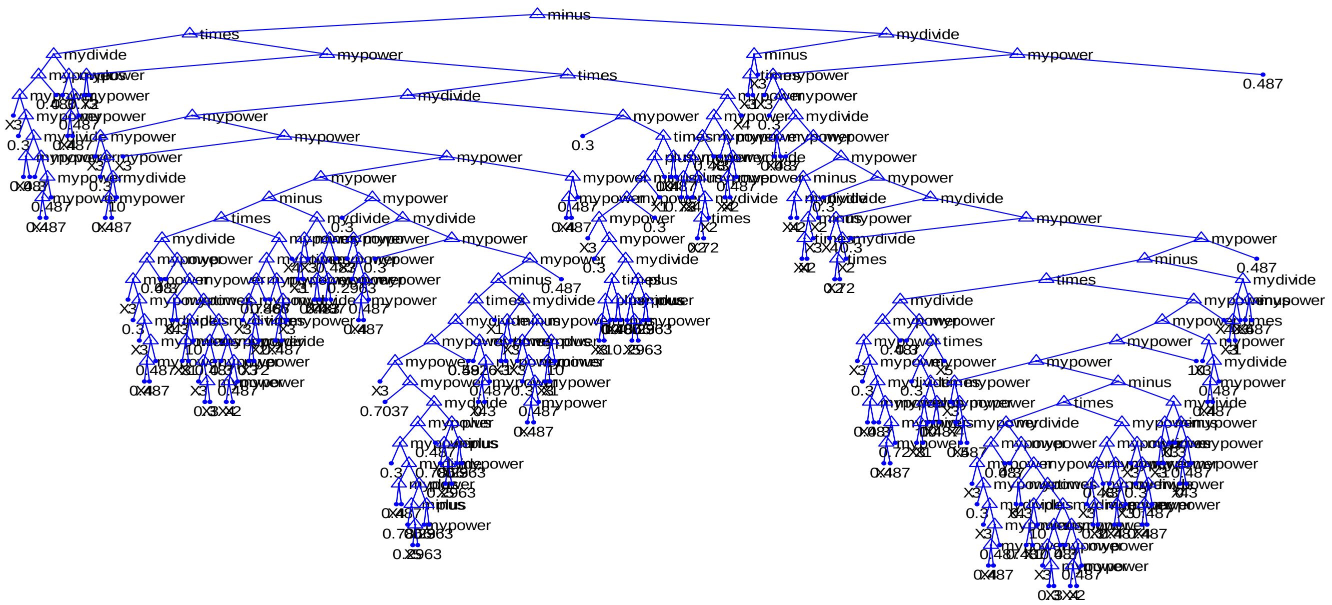


Figure 31 The best tree output for surface grinding exp. dataset (3) run 1B using GP programming code page 276 for Table 3 terminal set as dataset X and expected output as dataset Y

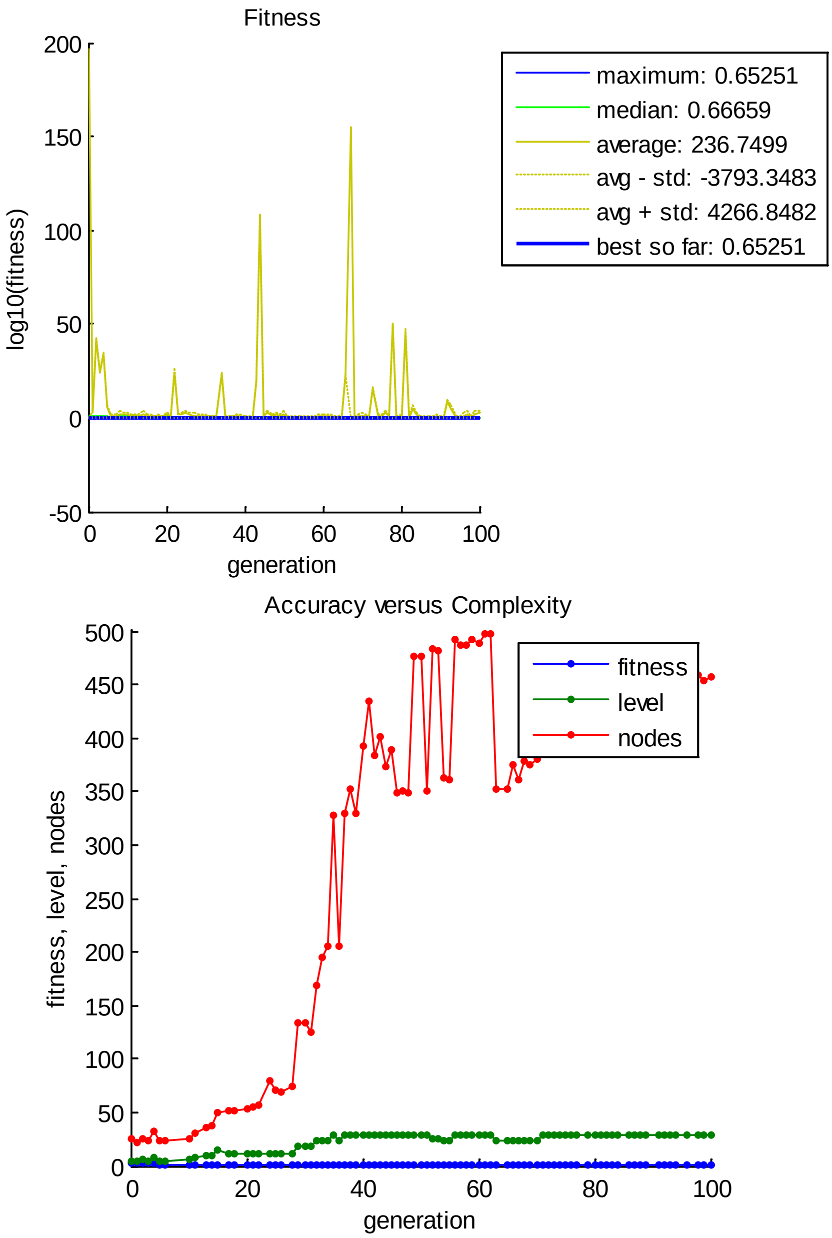


Figure 32 Graphical output for surface exp. dataset (3) run 1B

Run 2B

(individuals 100, Population 300)

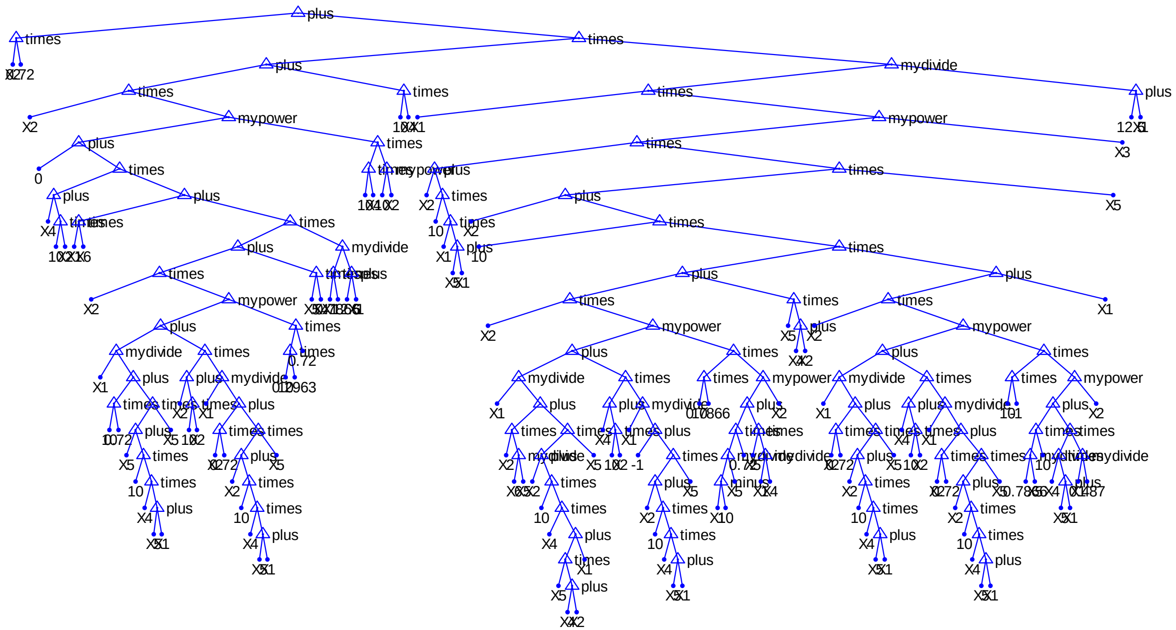


Figure 33 The best tree output for surface grinding exp. dataset (3) run 2B using GP programming code page 276 for Table 3 terminal set as dataset X and expected output as dataset Y

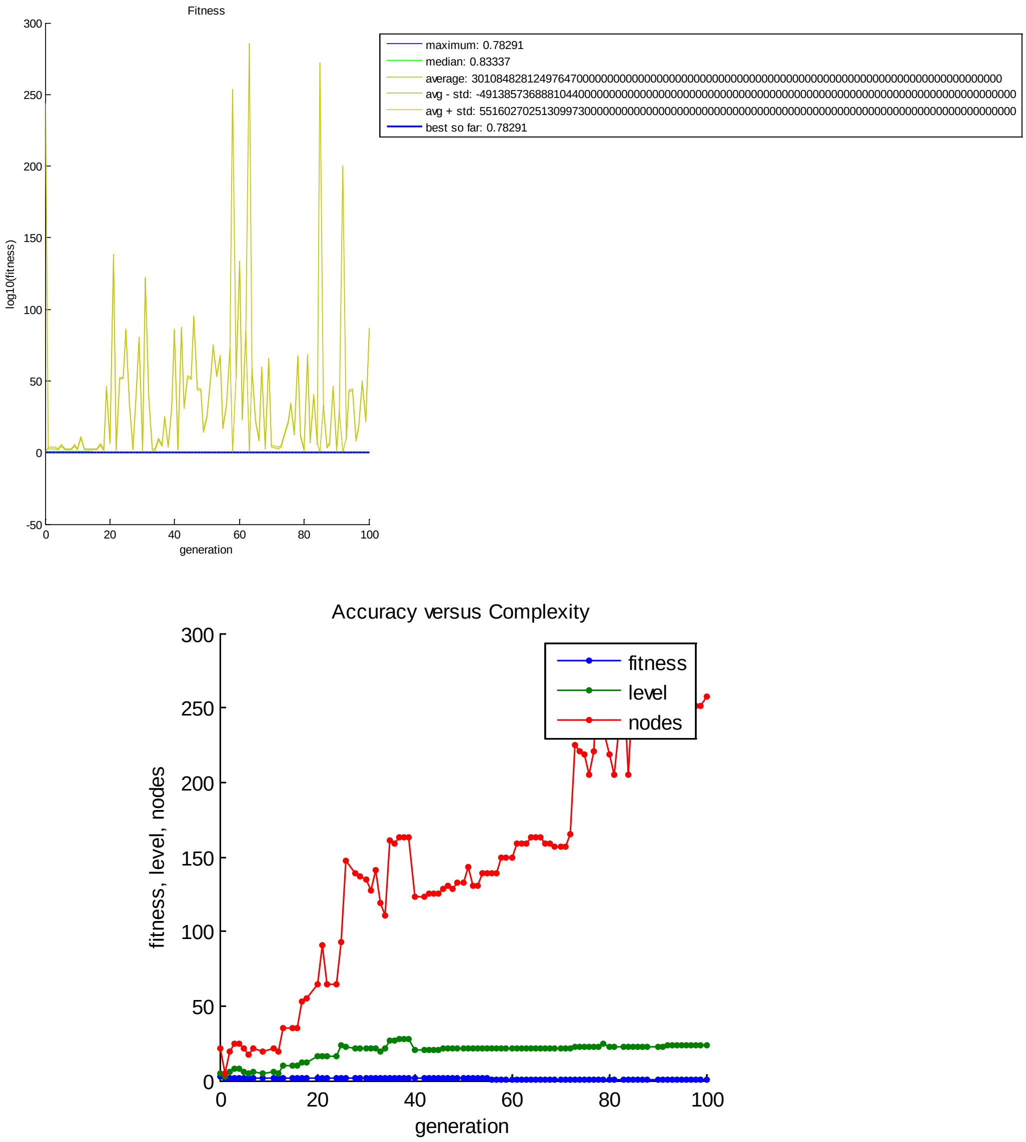


Figure 34 Graphical output for surface exp. dataset (3) run 2B

Run 3B

(Individuals 100, Population 300)

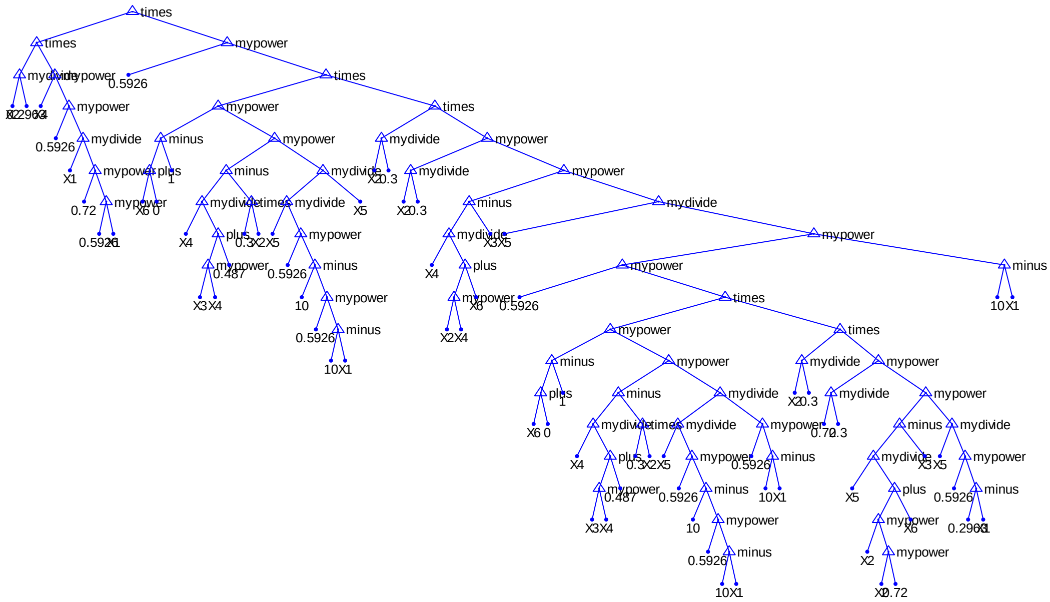


Figure 35 The best tree output for surface grinding exp. dataset (3) run 3B using GP programming code page 276 for Table 3 terminal set as dataset X and expected output as dataset Y

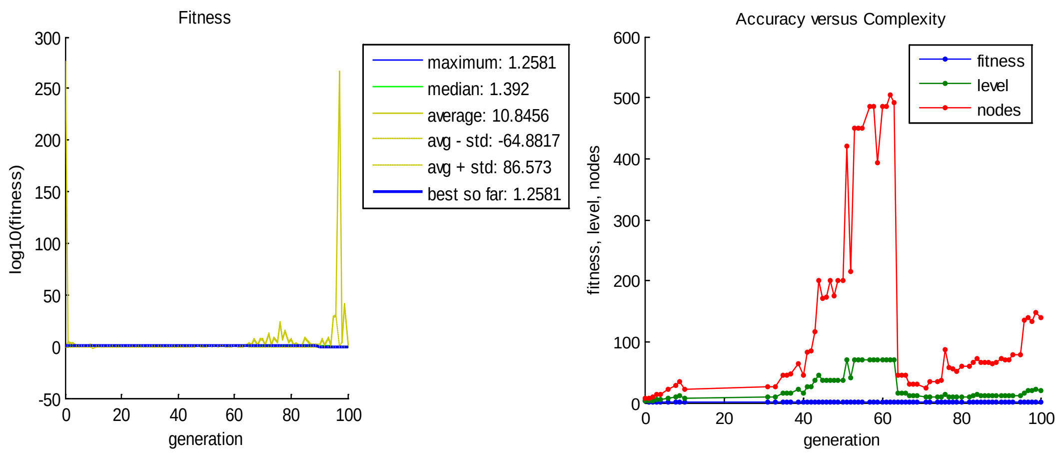


Figure 36 Graphical output for surface exp. dataset (3) run 3B

Surface grinding cal. dataset (4)

Table 4 Surface grinding exp. dataset(4)

Terminal Set Data file (X)						Expected output Data file (Y)
Dressing depth	Dressing lead	Speed ratio	St/bs	Geometrical contact	Equivalent chip thickness	Surface Roughness
a_d (µm)	s_d (mm)	$\frac{v_w}{v_s}$	$\frac{s_t}{b_s}$	$a.d_e$ (mm)	h_{eq} (µm)	R_a (µm)
5	0.15	0.004091	0.11811	1.64	4.09091E-05	0.397
10	0.15	0.004091	0.11811	1.64	4.09091E-05	0.397
10	0.15	0.004091	0.11811	1.64	4.09091E-05	0.502
15	0.15	0.004091	0.11811	1.64	4.09091E-05	0.502
15	0.15	0.004091	0.11811	1.64	4.09091E-05	0.576
15	0.05	0.004091	0.11811	1.64	4.09091E-05	0.577
15	0.2	0.004091	0.11811	1.64	4.09091E-05	0.318
15	0.25	0.004091	0.11811	1.64	4.09091E-05	0.673
15	0.15	0.003182	0.11811	0.82	1.59091E-05	0.759
15	0.15	0.003182	0.11811	1.23	2.38636E-05	0.524
15	0.15	0.003182	0.11811	1.64	3.18182E-05	0.524
15	0.15	0.003182	0.11811	1.64	3.18182E-05	0.524
15	0.15	0.003182	0.11811	2.46	4.77273E-05	0.524
15	0.15	0.003182	0.11811	2.87	5.56818E-05	0.524
15	0.15	0.002273	0.11811	1.64	2.27273E-05	0.524
15	0.15	0.001852	0.11811	1.64	1.85185E-05	0.461
15	0.15	0.003889	0.11811	1.64	3.88889E-05	0.426
15	0.15	0.003182	0.19685	1.64	3.18182E-05	0.566
15	0.15	0.005	0.03937	1.64	0.00005	0.653
15	0.15	0.004091	0.11811	1.64	4.09091E-05	0.388

Run 1A

(Individual size 100, population size 200)

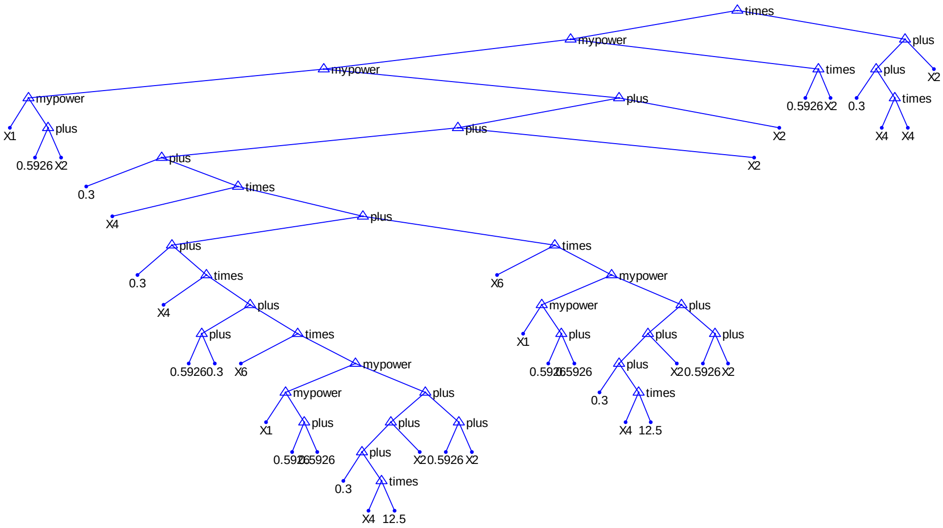


Figure 29 The best tree output for surface cal. dataset (4) run 1A

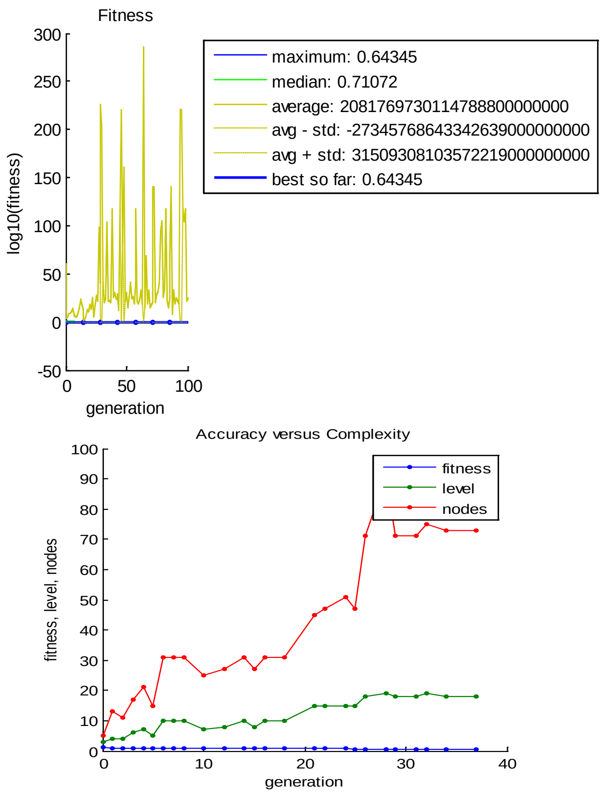


Figure 30 Graphical output for surface cal. dataset (4) run 2A

Run 2A

Individual size 100, population size 200

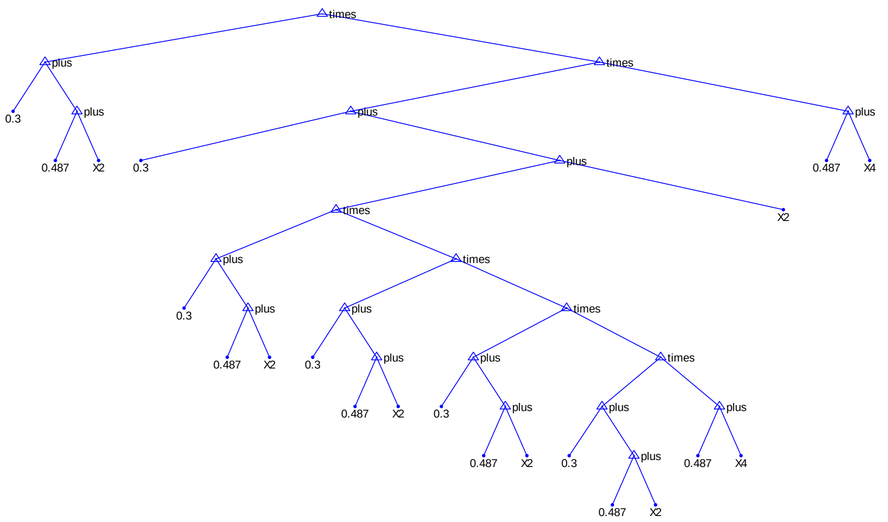


Figure 31 The best tree for surface cal. dataset (3) run 2A

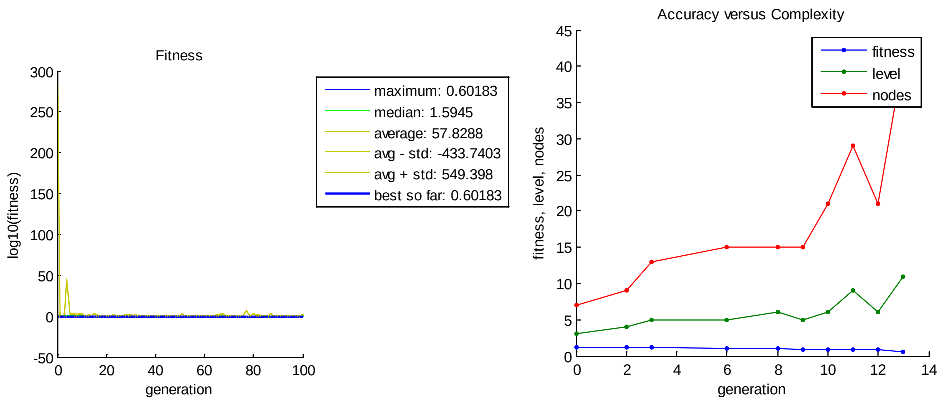


Figure 32 Graphical output for surface cal. dataset (4) run 2A

Run 3A

Individual size 100, population size 200

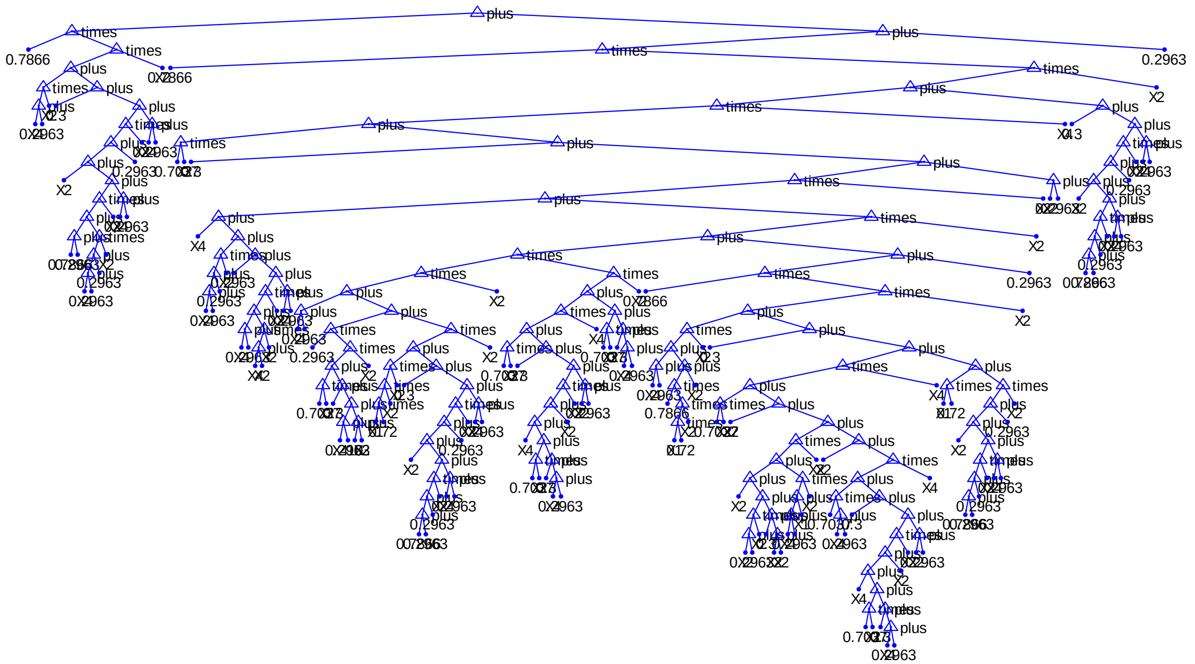
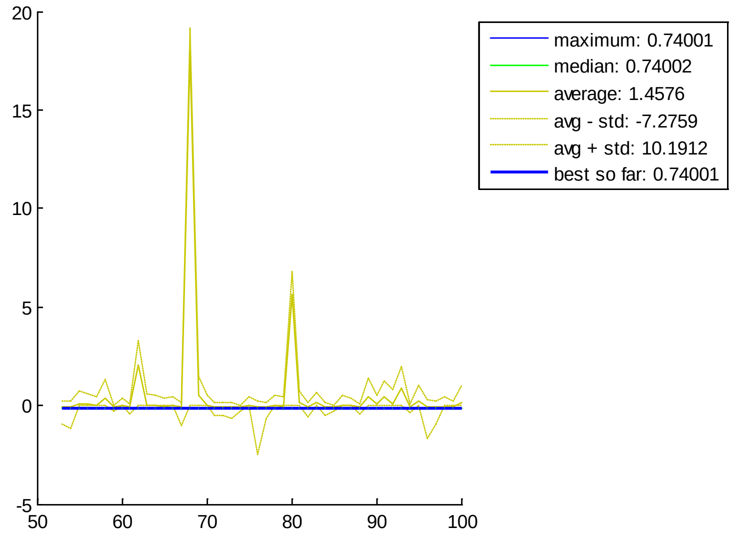


Figure 41 The best tree for surface cal. dataset (4) run 3A using GP programming code page 276 for Table 4 terminal set as dataset X and expected output as dataset Y



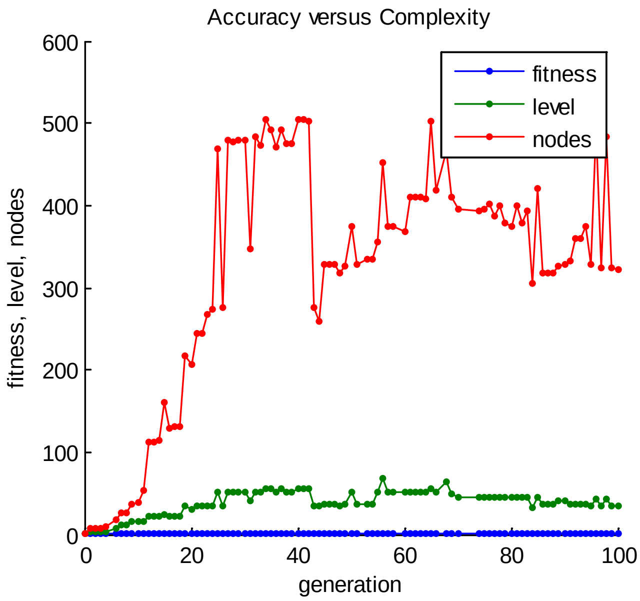


Figure 42 Graphical output for surface cal. dataset (4) run 3A

Run 1B

(Individual size 100, Population size 300)

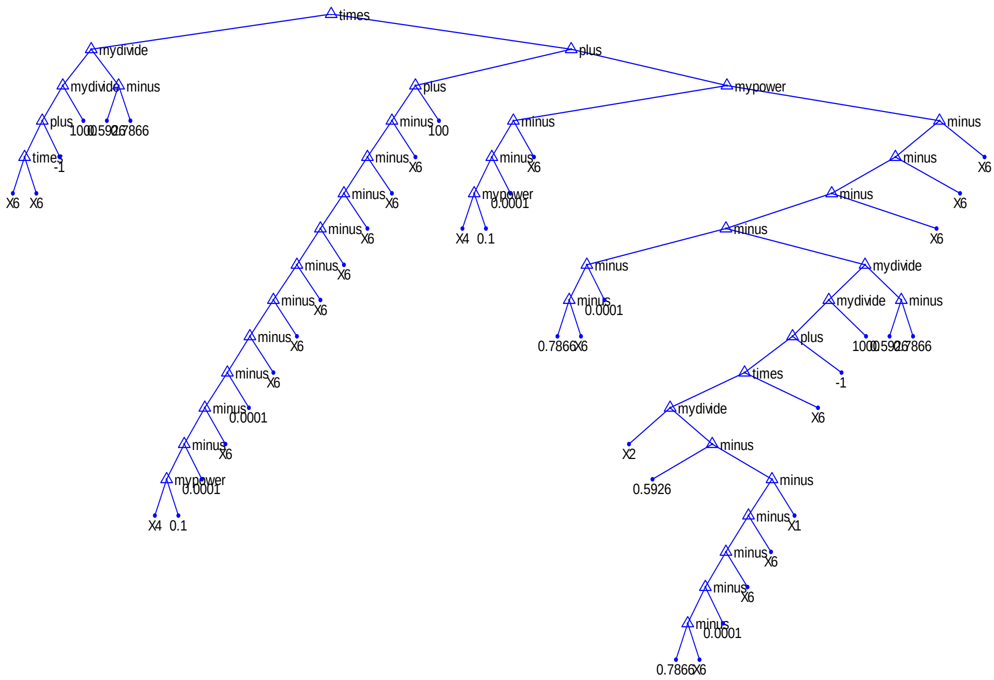


Figure 43 The best tree for surface cal. dataset (4) run 1B

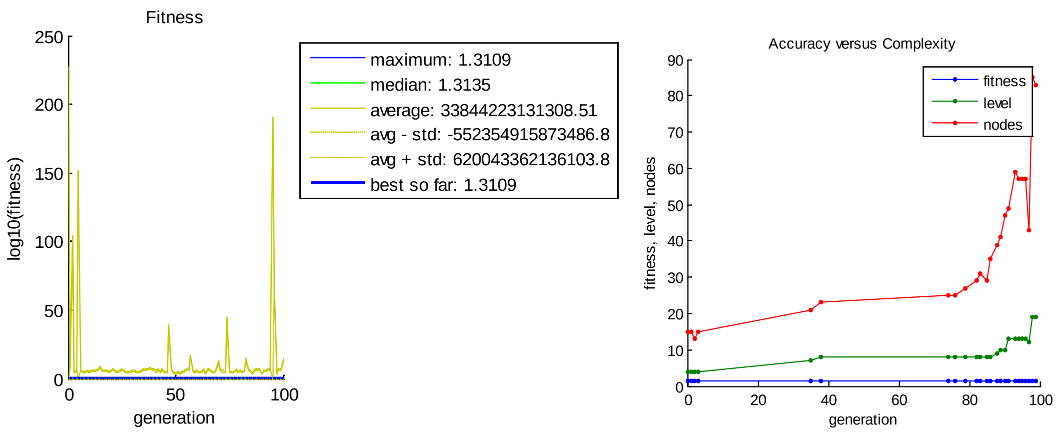


Figure 44 Graphical output for surface cal. dataset (4) run 1B

Run 2B

(Individual size 100, Population size 300)

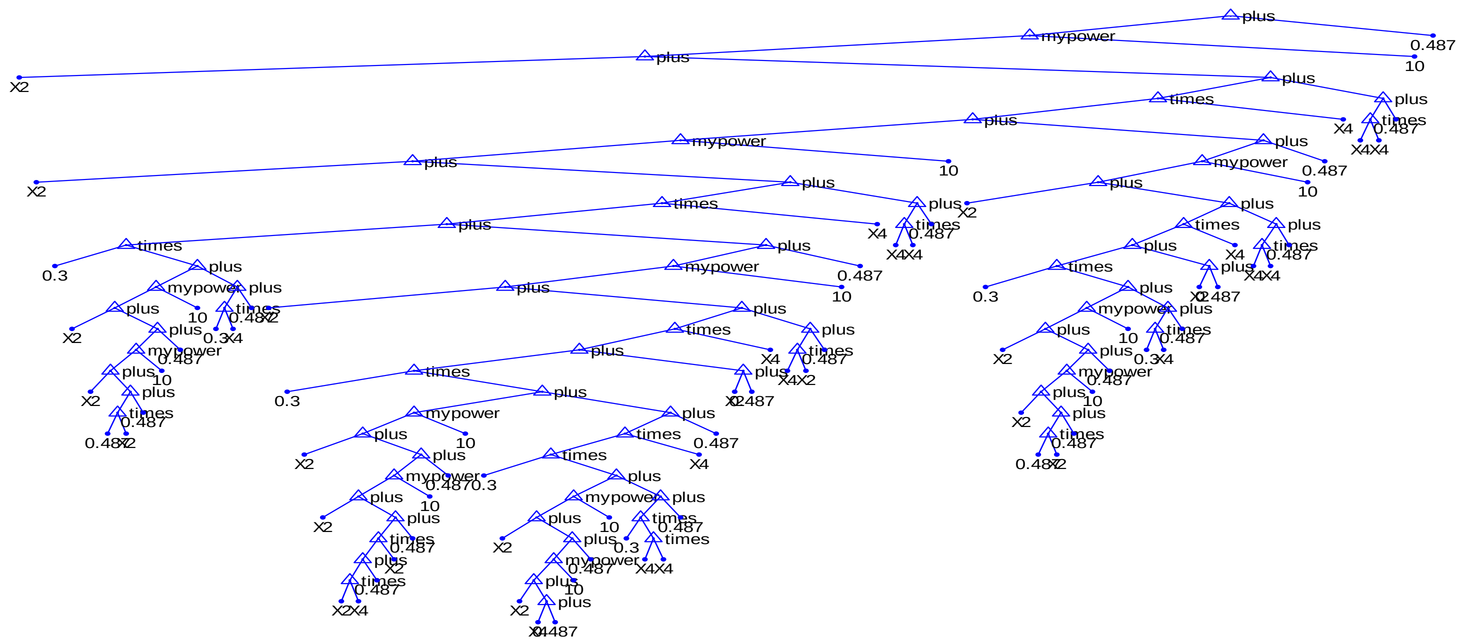
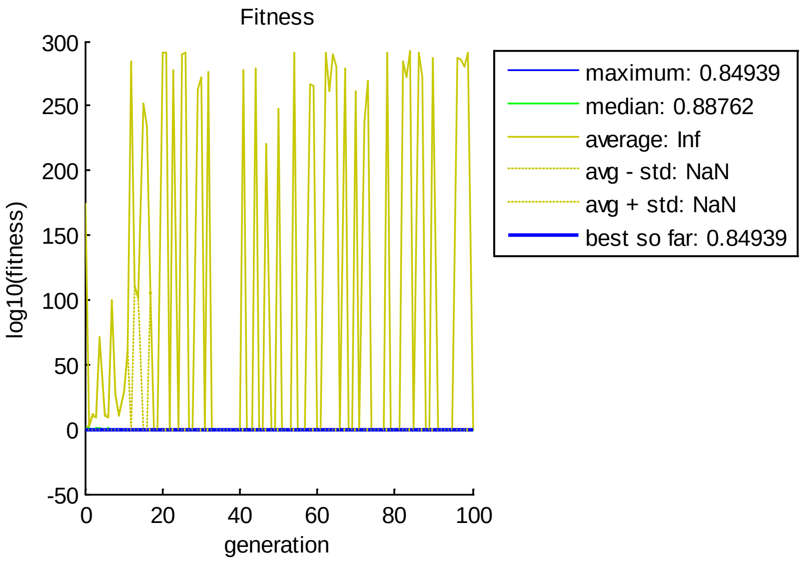


Figure 45 The best tree for surface cal. dataset (4) run 2B using GP programming code page 276 for Table 4 terminal set as dataset X and expected output as dataset Y



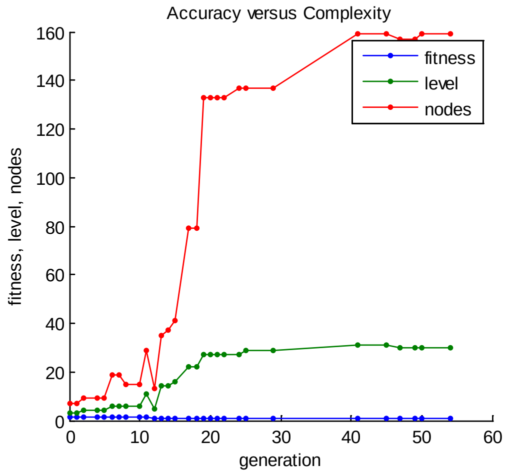


Figure 46 Graphical output for surface cal. dataset (4) run 2B

Run 3B

(Individual size 100, Population size 300)

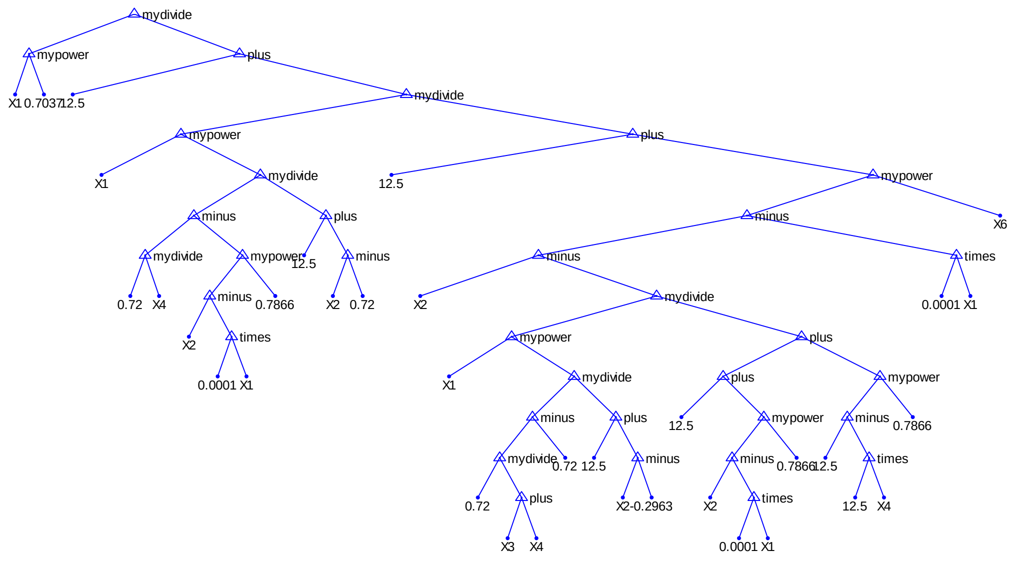


Figure 47 The best tree for surface cal. dataset (4) run 3B

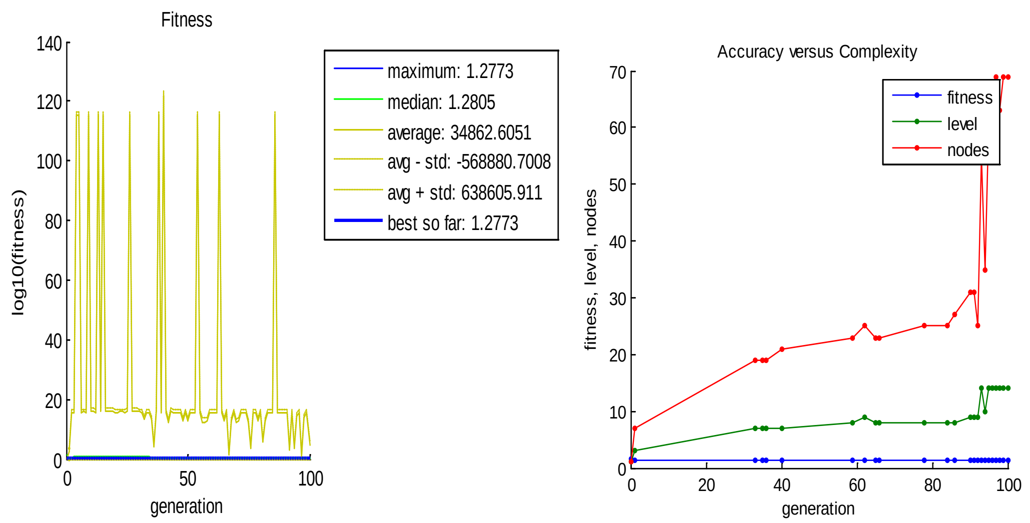


Figure 48 Graphical output for surface cal. dataset (4) run 3B

Lee, Choi, and Shin (2003)

Table 5 Surface grinding Lee, Choi and Shin (2003)

a_d (µm)	s_d (mm)	$\frac{v_w}{v_s}$	$\frac{s_t}{b_s}$	$a.d_e$ (mm)	h_{eq} (µm)
Dressing depth	Dressing lead	Speed ratio	St/bs	Geometrical contact	Equivalent chip thickness
20	0.12	0.00303	0.024016	2.54	3.0303E-05
20	0.12	0.006061	0.022047	2.54	6.0606E-05
20	0.12	0.009091	0.018898	2.54	9.0909E-05
20	0.12	0.009091	0.019685	7.62	0.00027273
20	0.12	0.006061	0.022047	7.62	0.00018182
20	0.12	0.00303	0.024016	7.62	9.0909E-05
20	0.12	0.00303	0.04252	2.54	3.0303E-05
20	0.12	0.006061	0.037795	2.54	6.0606E-05
20	0.12	0.009091	0.03937	2.54	9.0909E-05
20	0.12	0.009091	0.03937	7.62	0.00027273
20	0.12	0.006061	0.03622	7.62	0.00018182
20	0.12	0.00303	0.041732	7.62	9.0909E-05
20	0.12	0.00303	0.06063	2.54	3.0303E-05
20	0.12	0.006061	0.057087	2.54	6.0606E-05
20	0.12	0.009091	0.06063	2.54	9.0909E-05
20	0.12	0.009091	0.06063	7.62	0.00027273
20	0.12	0.006061	0.058661	7.62	0.00018182
20	0.12	0.00303	0.058661	7.62	9.0909E-05
8	0.12	0.006061	0.040157	5.08	0.00012121
13	0.12	0.006061	0.040157	5.08	0.00012121
20	0.12	0.006061	0.040157	5.08	0.00012121
3	0.12	0.006061	0.040157	5.08	0.00012121
51	0.12	0.006061	0.040157	5.08	0.00012121
20	0.06	0.006061	0.040157	5.08	0.00012121
20	0.24	0.006061	0.040157	5.08	0.00012121
20	0.12	0.002941	0.032677	3.81	4.4118E-05
20	0.12	0.002941	0.033071	7.62	8.8235E-05
20	0.12	0.002941	0.032677	11.684	0.00013529
20	0.12	0.008824	0.030709	11.684	0.00040588
20	0.12	0.008824	0.030709	7.62	0.00026471
20	0.12	0.008824	0.03189	3.81	0.00013235

20	0.12	0.002941	0.05748	3.81	4.4118E-05
20	0.12	0.002941	0.057874	7.62	8.8235E-05
20	0.12	0.002941	0.070472	11.684	0.00013529
20	0.12	0.008824	0.068504	11.684	0.00040588
20	0.12	0.008824	0.070866	7.62	0.00026471
20	0.12	0.008824	0.072835	3.81	0.00013235
20	0.12	0.002941	0.090157	3.81	4.4118E-05
20	0.12	0.002941	0.090157	7.62	8.8235E-05
20	0.12	0.002941	0.090157	11.684	0.00013529
20	0.12	0.008824	0.090157	11.684	0.00040588
20	0.12	0.008824	0.090157	7.62	0.00026471
20	0.12	0.008824	0.090157	3.81	0.00013235
13	0.12	0.005882	0.059843	7.62	0.00017647
20	0.12	0.005882	0.059843	7.62	0.00017647
30	0.12	0.005882	0.059843	7.62	0.00017647
51	0.12	0.005882	0.059843	7.62	0.00017647

Run 1A

(individual size 100, population size 200)

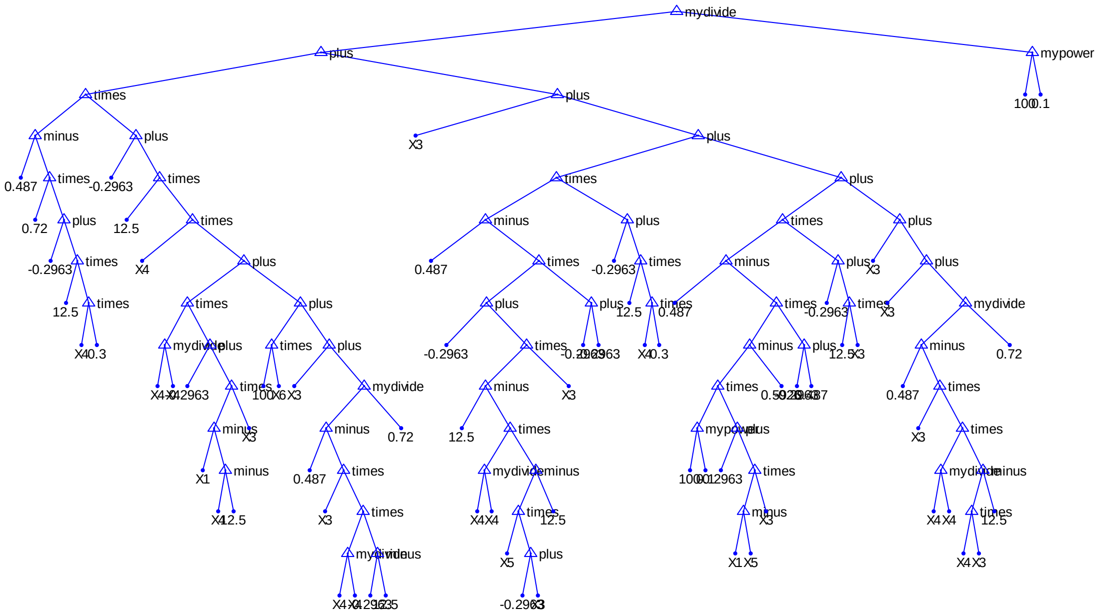


Figure 49 The best tree for surface grinding Lee, Choi and Shin 2003 run 1A using GP programming code page 276 for Table 5 terminal set as dataset X and expected output as dataset Y

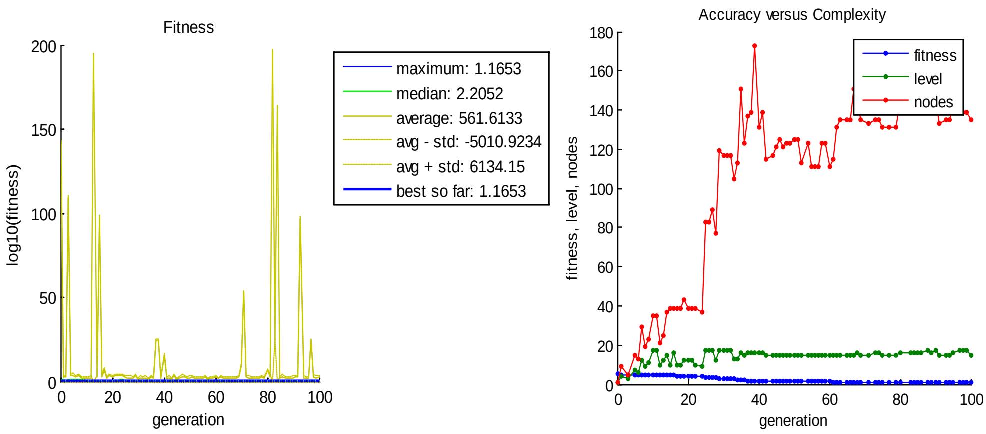


Figure 50 Graphical output for surface grinding Lee, Choi and Shin 2003 run 1A

Run 2A

(Individual size 100, Population size 200)

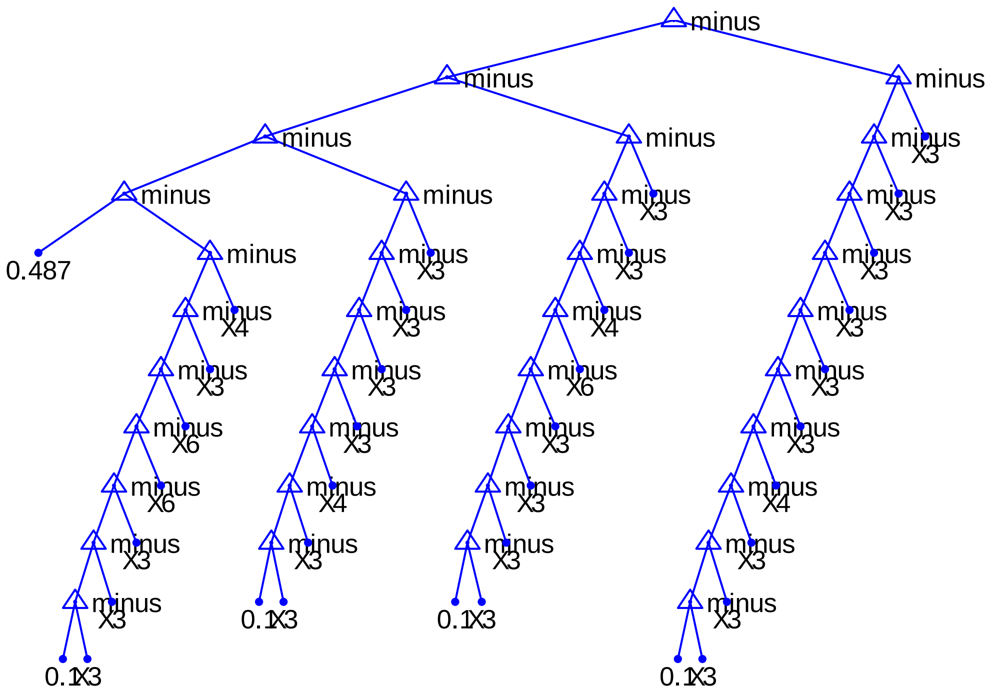


Figure 51 The best tree for surface grinding Lee, Choi and Shin 2003 run 2A using GP programming code page 276 for Table 5 terminal set as dataset X and expected output as dataset Y

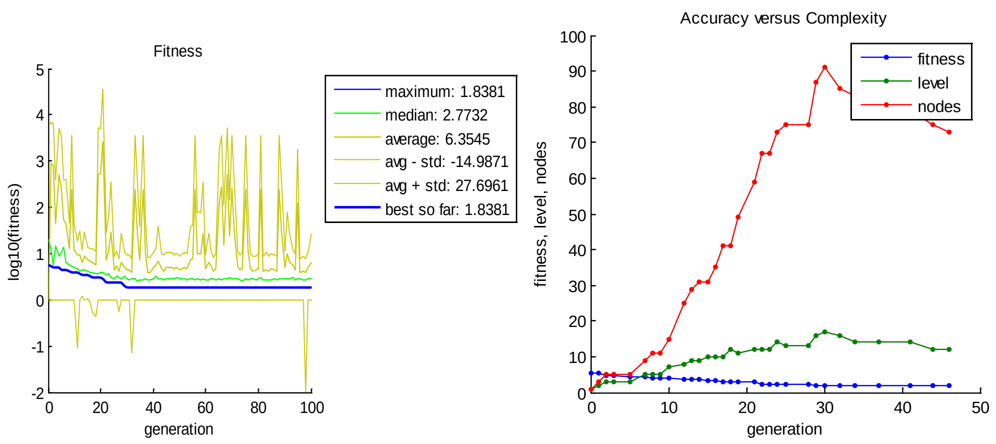


Figure 52 Graphical output for surface grinding Lee, Choi and Shin 2003 run 2A

Run 3A

(Individual size 100, Population size 200)

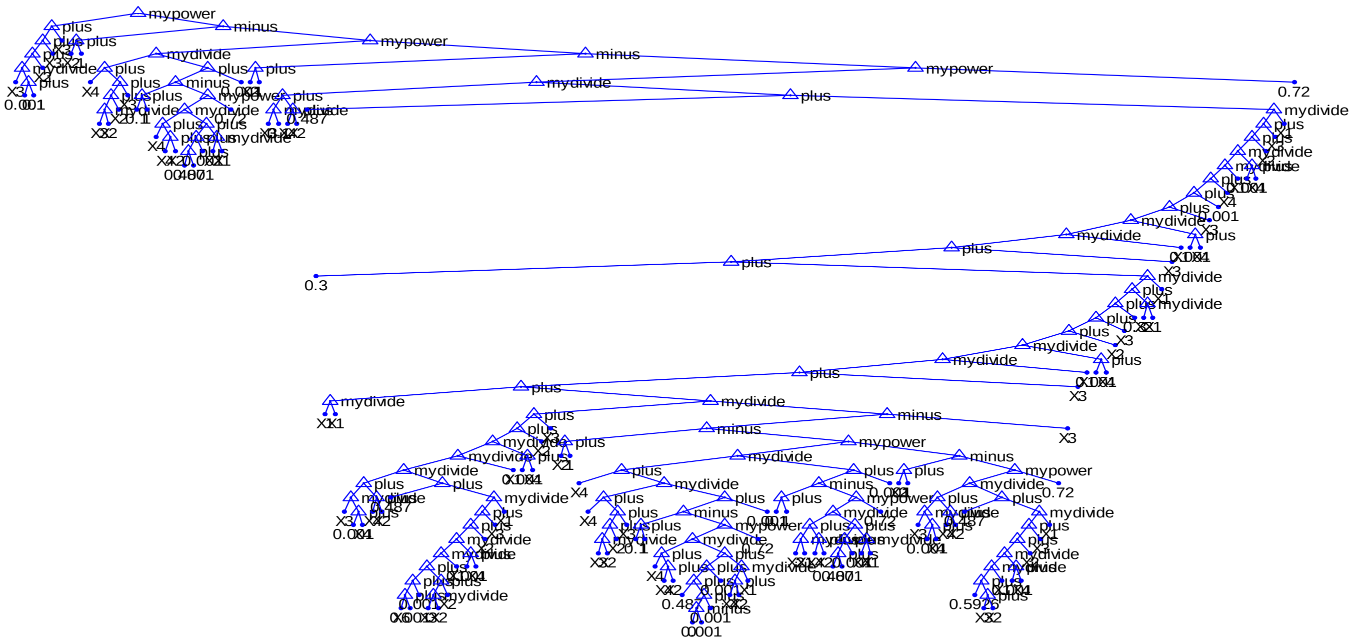
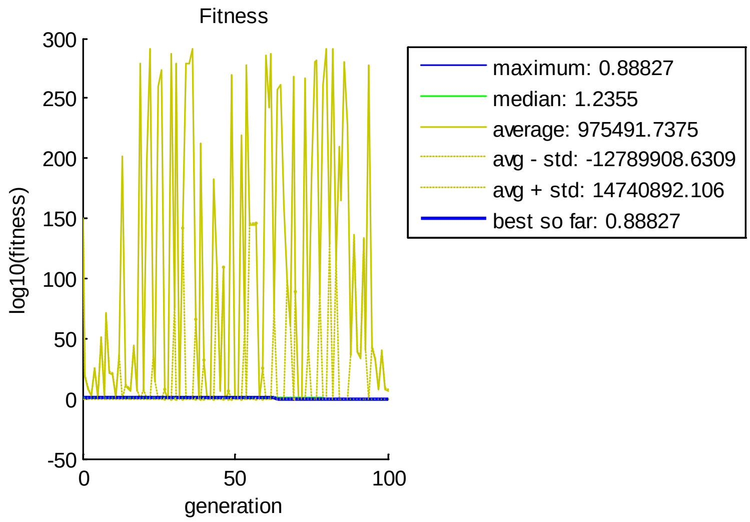


Figure 53 The best tree for surface grinding Lee, Choi and Shin 2003 run 3A using GP programming code page 276 for Table 5 terminal set as dataset X and expected output as dataset Y



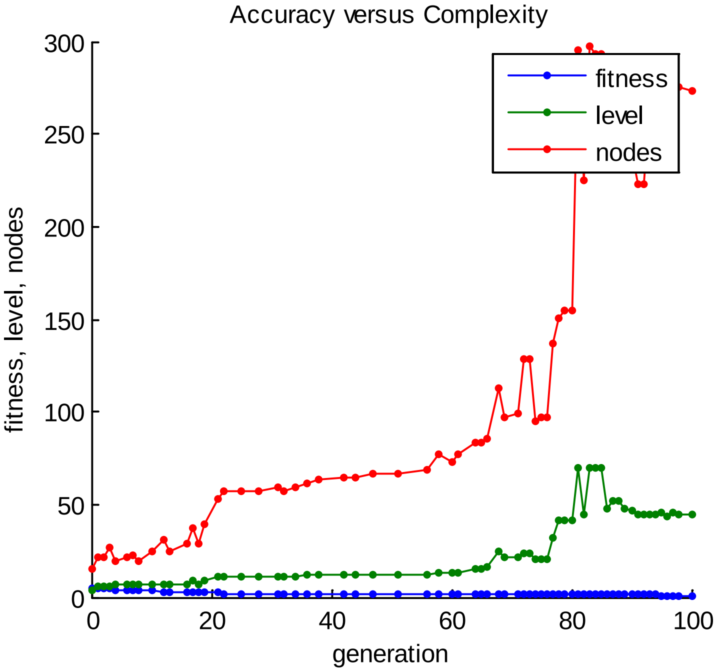


Figure 54 Graphical output for surface grinding Lee, Choi and Shin 2003 run 3A

Run 1B

(Individual size 100, Population size 300)

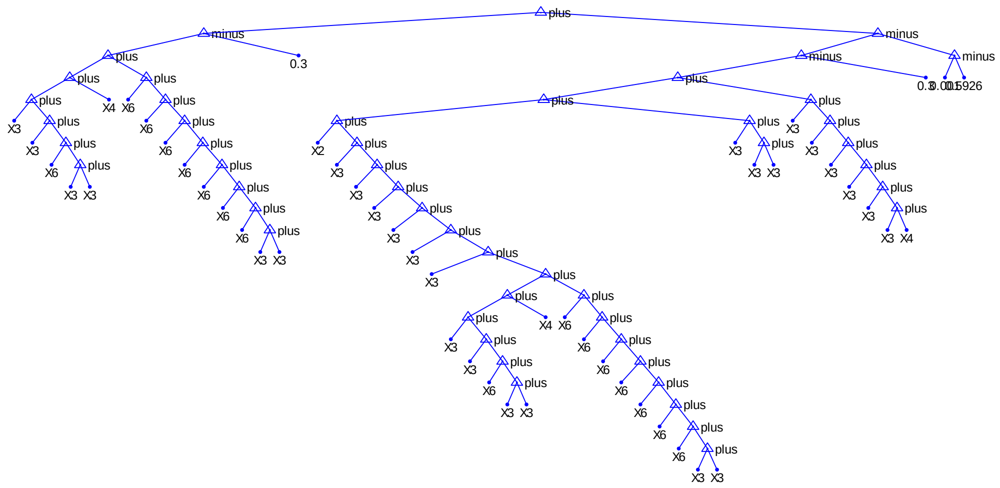


Figure 55 The best tree for surface grinding Lee, Choi and Shin 2003 run 1B

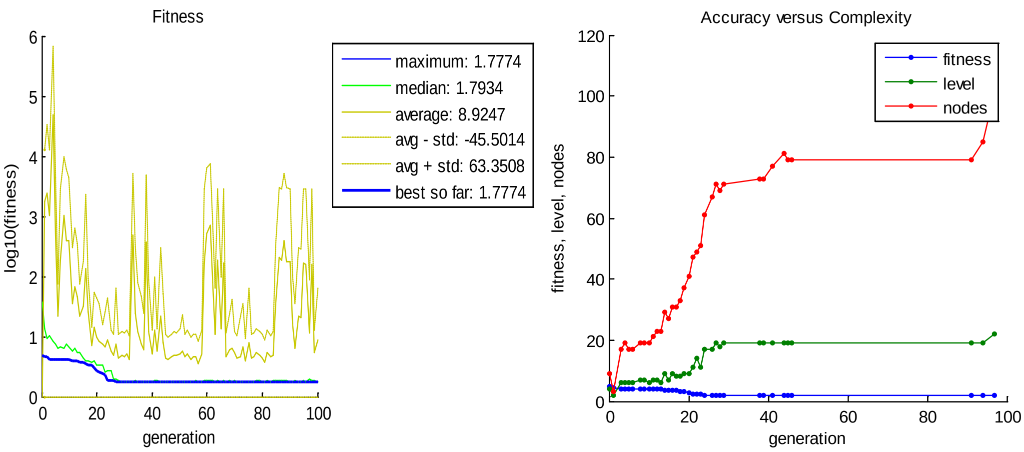


Figure 56 Graphical output for surface grinding Lee, Choi and Shin 2003 run 1B

Run 2B

(Individual size 100, Population size 300)

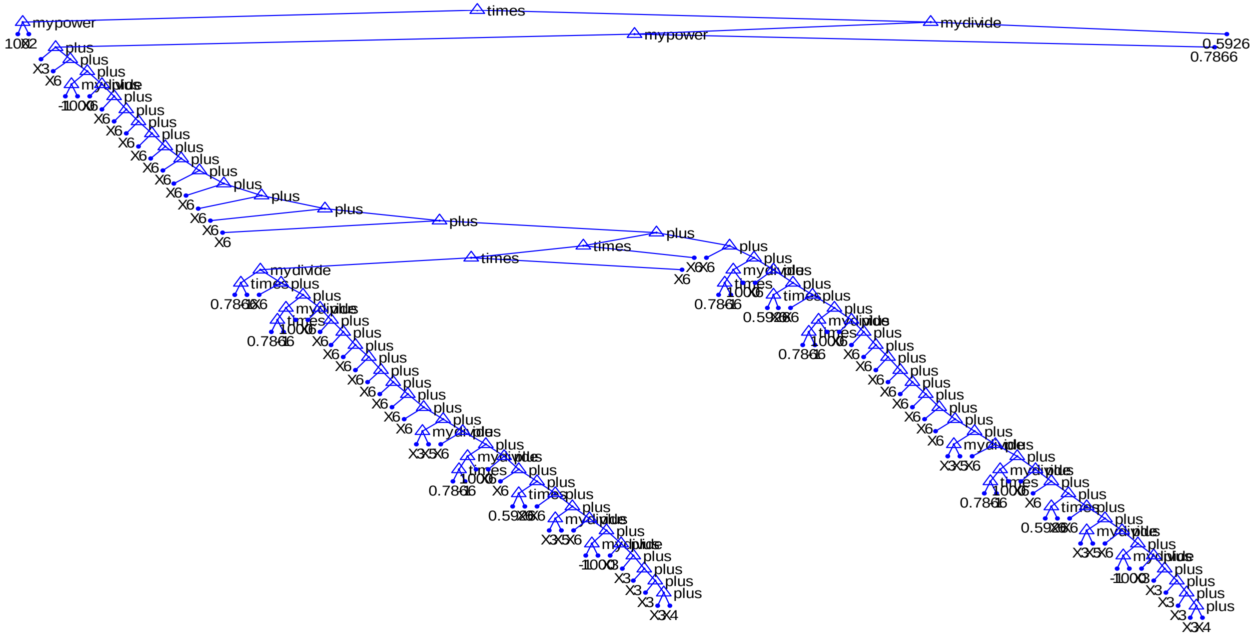


Figure 57 The best tree for surface grinding Lee, Choi and Shin 2003 run 2B using GP programming code page 276 for Table 5 terminal set as dataset X and expected output as dataset Y

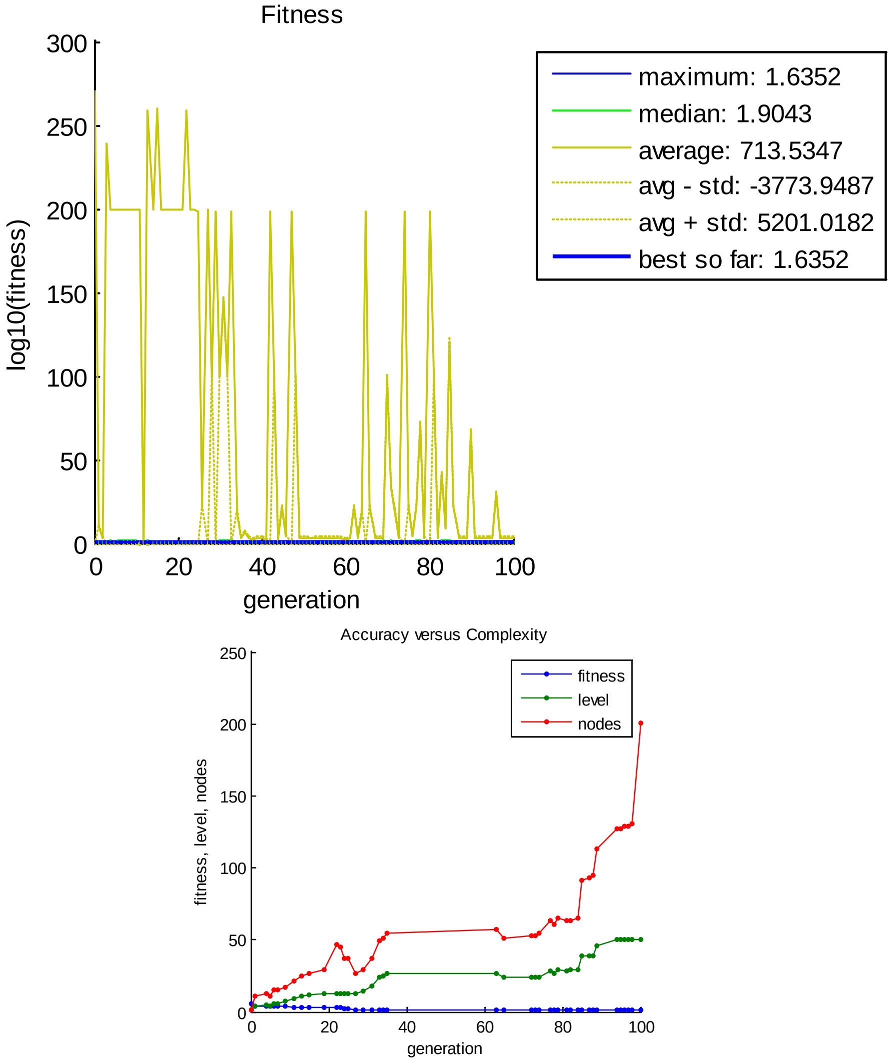


Figure 58 Graphical output for surface grinding Lee, Choi and Shin 2003 run 2B

Run 3B

(Individual size 100, Population size 300)

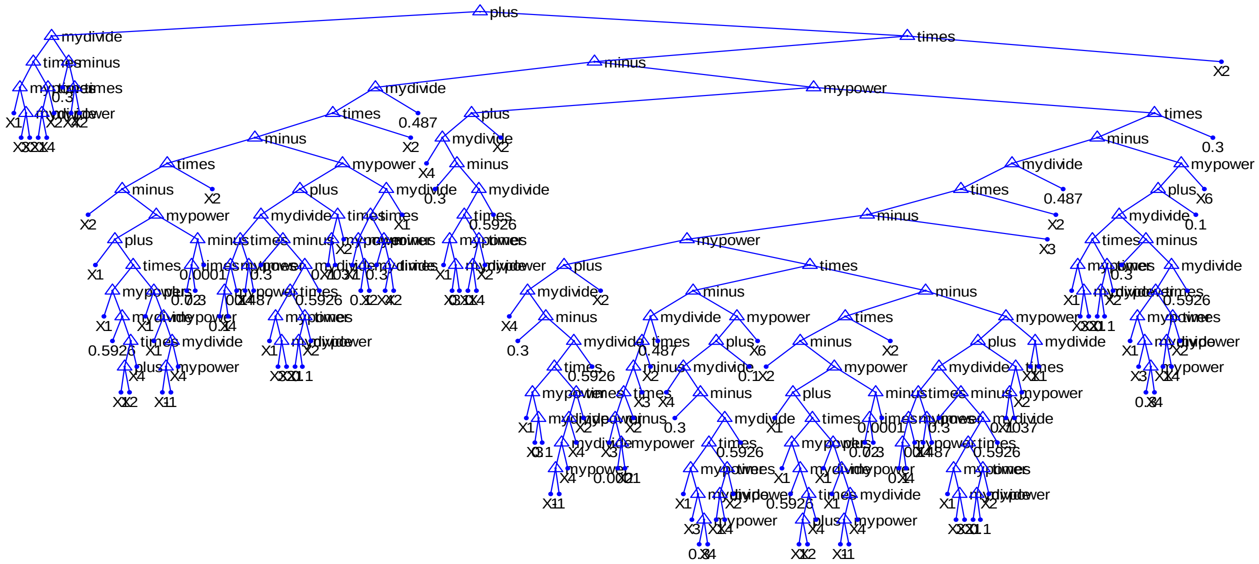
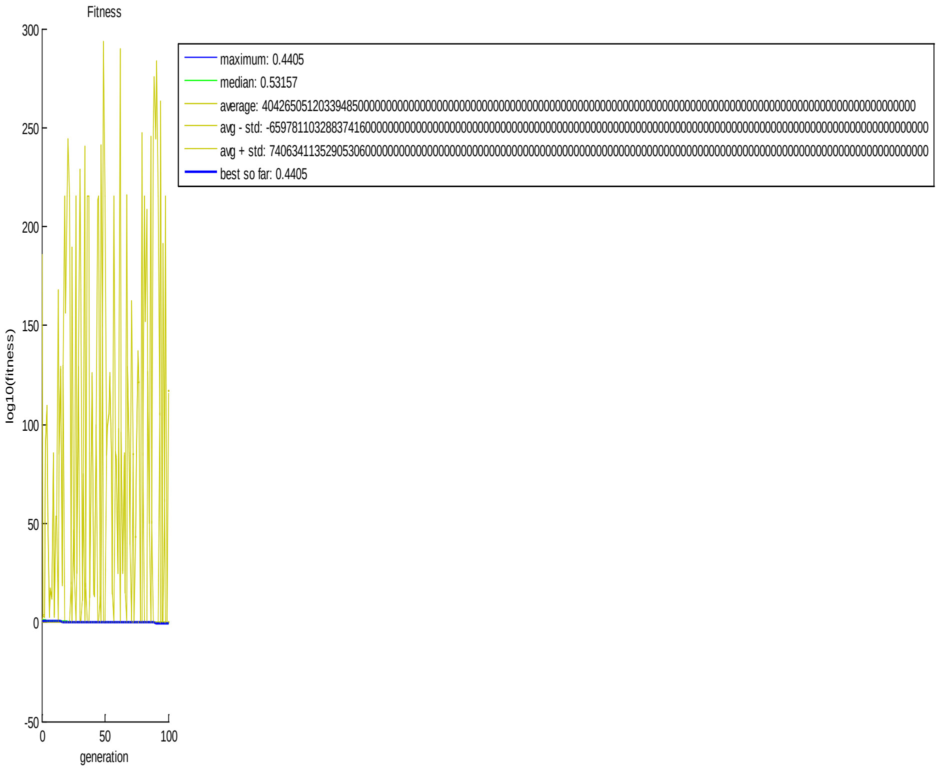


Figure 59 The best tree for surface grinding Lee, Choi and Shin 2003 run 3B using GP programming code page 276 for Table 5 terminal set as dataset X and expected output as dataset Y



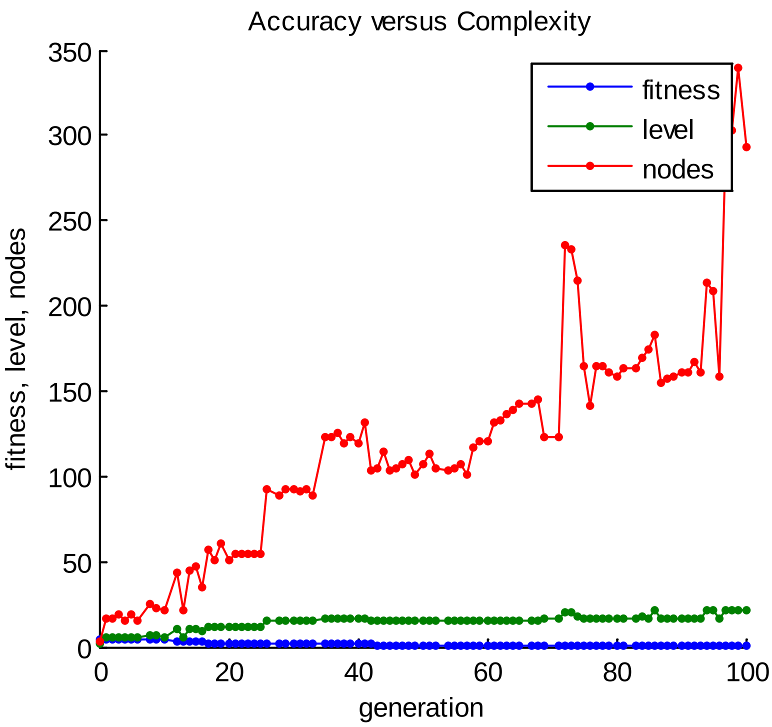


Figure 60 Graphical output for surface grinding Lee, Choi and Shin 2003 run 3B

Genetic Programming Code

```
% demonstration function to illustrate its usage
function [v,b]=demoparity
%   Copyright (C) 2003-2007 Sara Silva (sara@dei.uc.pt)
%   This file is part of the GPLAB Toolbox
params=resetparams;
[v,b]=gplab(200,500,params);    %population 200 and generations 51
%    First, generate initial population using ramped half and half        %
method
params.initpoptype='rampedinit';
% set the function such as +, _, *. ?, Power and exp
params=setfunctions(params,'plus',2,'minus',2,'times',2,    'mydivide',    2,
'mypower' ,2);
% set the terminals which are wheel speed, wheel diameter, work %speed, work
diameter, dressing lead, dressing depth, grain size, and %roughness
params=setterminals(params,'workspeed','workdia','wheeldia','wheelspeed','gr
ainsize','roughness','dressinglead','dressingdepth');
% set the genetic operations
 params=setoperators(params,'crossover',2,2,'mutation',1,1);
%p.calcfitness='regfitnessv11';  %used instead of class fitness for
%backpropagation type learning rule
% validate the new individual that produced by GP
params.survival='keepbest';      % keep best individual
params.sampling='tournament';
params.elitism='replace';
params.survival='fixedpopsize';
% import data from the text file
params.datafilex='GP_data_Chen2.txt'; %  the input variables
params.datafiley='GP_out_Chen2.txt';    %  the output(roughness)

% extra data file for the input and expected output
params.testdatafilex='patterns_input_chendata.txt';
params.testdatafiley='targets_roughness_chendata.txt';
params.operatorprobstype='variable';
params.usetestdata=1;

%  measure row fitness
params.calcfitness='regfitness';
params.lowerisbetter=1; % if the fitness value is lower this is best

 %  calculate more information about the evolutionary process
params.calcdiversity={};
params.fixedlevel=0;
params.dynamiclevel='1'; %allows tree bloat if fitness increases
 params.tournamentsize=0.1; %number of new individuals to be created using

%GP tournament size is 10% of designated population set
%   GP will run until the max generation indicated by user is reached or
%until a stop condition is reached
params.hits='[100 0 50 10]';

%  stop if the best individual produces exact results in all fitness %cases
%and stop if the best individual produces result within plus %minus 10% of
%expected result
%graphic GP output
params.graphics={'plotfitness','plotdiversity','plotcomplexity','plotoperato
rs'};
desired_obtained(v,[],1,0,[]);      %desired v obtained graphical o/p
accuracy_complexity(v,[],0,[]);     % accuracy v complexity graphical o/p
figure
plotpareto(v);                      % pareto GP plot
drawtree(b.tree);                     % plot GP tree representing best found
%data separation rule
```

Appendix C

PHPand HTML Source

```
<?php
//connect to the cop database
$conn=mysql_connect("localhost","root","password") or die(mysql_error());
mysql_select_db("sengaa2",$conn) or die(mysql_error());
?>
<?php include('header.php');
#script5.8- index.php
$page_title='knowledge warehouse';
include('headder.html');
?>
<h1>Grinding Knowledge Warehouse (GKW)</h1>
<p><strong>The aim of the research is to facilitate knowledge management process in grinding technology by
building
a flexible and easy to use Grinding Knowledge Warehouse (GKW) that could
<ul> <li>manage the explicit knowledge</li>
  <li>facilitate transferring tacit knowledge into explicit knowledge</li>
  <li>support the decision making process for selecting grinding conditions</li>
  <li>encourage and facilitate the sharing of explicit and tacit knowledge</li>
  <li>build problem solving and question-answer module.</li>
</ul>
 </strong></p>
 <?php
 include('footer.html');
 ?>
<html>
<head>
<title>Login form</title>
</head>
<body>
<H1> Log in Form</H1>
<form action="login.php"  method="POST">
<table>
<tr>
   <td>Username</td>
   <td><input type="text" name="username"></td>
</tr>
<tr>
		<td>Password</td>
		<td><input type="password" name="password"></td>
</tr>
<tr>
		<td></td>
		<td><input TYPE="SUBMIT" name="submit" value="login!"></td>
</tr>
</table>
</form>
<H2> New user can register by clicking <a href="register.php">on this link</a> </H2>
```

```
</body>
</html>
</li>
<?php ob_start(); session_start();
include('config.php');
//add the programmer name
$config_admin="Asmaa Alabed";
$config_adminemail= "a_alabed AT hotmail DOT com";
//add the location of the forum
$config_basedir="http:localhost/touploadGrinding";
// check for the required fields
if ((!$_POST['username'] )|| (!$_POST['password'])){
        echo "Please, enter the username and password,thanks";
        header("Location: login_form.html");
        exit;
        }
if($_POST['submit']) {
        $sql="select * from emp where username= '"
        . $_POST['username'] . "' AND password ='"
        . $_POST['password'] . "';";
        $result=mysql_query($sql) or die(mysql_error());
        $numrows=mysql_num_rows($result);
if($numrows ==1) {
        $row=mysql_fetch_assoc($result);
        if ($row['active'] ==1){
                $_SESSION['USERNAME']=$row['username'];
                $_SESSION['USERID']=$row['emp_id'];
                header("location:login_success.php");
        // "clcik to enter <strong><a href='index.php'></a></strong>";
                //the user is authorid
                    }
                 else {
                 //require("header.php");
                $msg= "This account is not active";
                    print "$msg";
            }
   }      else {
            $sql="select * from emp where username= '"
           . $_POST['username'] . "' OR password ='"
           . $_POST['password'] . "';";
           $result=mysql_query($sql) or die(mysql_error());
           $numrows=mysql_num_rows($result);
           if($numrows ==1) {
                 echo "Please,check the username or password!";
                 echo "[<a href='login.php'> Click to try again </a>]";
             } else {echo "You are not authorised user";}
        }
}
?>
<?php session_start();
if(!isset($_SESSION['USERNAME'])){
$msg = "Invalid user name or password.";
$error = 1;
}else{
$msg = "Welcome! ".$_SESSION['USERNAME']." ,You have been successfully Logged in.";
$error = 0;
}
?>
<br /><br /> <br /><br />
<table id="Table_01" width="450" height="230" border="0" align='center' cellpadding="0" cellspacing="0">
<div id='msg' align='center' ><?php echo $msg ; ?> </div>
<?php if ($error==0){ ?>
<div id='msg' align='center' align='center'><?php echo "<a href='index2.php' class='join'>Click here if your
browser dose not supports automatically redirection </a>";?></div>
```

```
<div id='msg' align='center' align='center'><?php echo "<meta http-equiv='Refresh' content='3 ;
URL=index2.php'>";?></div>
<?php }else{ ?>
<div id='msg' align='center' align='center'><?php echo "<a href='index.php' class='join'>Click here if your browser
dose not supports automatically redirection </a>";?></div>
<div id='msg' align='center' align='center'><?php echo "<meta http-equiv='Refresh' content='3 ;
URL=index.php'>";?></div>
<?php }?>
    </td></tr>
</table>
<br /><br />
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=is-8859-1"/>
  <title><?php echo $page_title; ?>
  </title>
  <style type="text/css" media="screen">
  body {background-color:#ffffff;}
  .content {
                background-color: #f5f5f5;
                padding-top: 10px; padding-right: 10px; padding-bottom: 10px; padding-left:10px;
                marging-top:10px; margin-right:10px; margin-bottom:10px;margin-left:10px;
                }
    a.navlink:link{ font-size:16px ;color:#003366;
    text-decoration: none;
                }
   a.navlink:visited { font-size:16px; color:#003366;
   text-decoration: none;
                }
     a.navlink:hover { font-size:16px ;color:#ccccc;
    text-decoration: none;
                }
   td{
              font-family: verdana, Arial,
              Helvetica, sans-serif; font-size:13px;
              vertical-align:top;
              }
  .title{
              font-size:28px;font-weight:normal;color:#ffffff;
              margin-top:5px; margin-bottom:5px;
              paddong-top:5px; padding-bottom:5px; padding-left:20px;
              }
 </style>
 </head>
 <body>
 <table width="90%" border="0" cellspacing="10"
 cellpadding="0" align="center">
 <tr>
 <td colspan="2" bgcolor='#003366'>
 <p class="title"> Knowledge WareHouse for Grinding  Technology <a href='logout.php' style="font-size:
12px;color: #FF0000;" >Log out</a> </p></td></tr>
 <tr>
 <td valign="top" nowrap="nowrap">
 <b><a href="index2.php" class="navlink">Home</a><br/><br/>
 <a href="oldpart.php" class="navlink">View Grinding Cases for Pre-grounded Part</a><br/><br/>
  <a href="newpartform.php" class="navlink">Select Grinding Conditions for New Part </a><br/><br/>
 <a href="add_new_grindingcase.php" class="navlink">Grinding Case Management</a><br/><br/>
 <a href="cop.php" class="navlink">Community of Practice</a><br/><br/>
  <a href="doc_searchform.html" class="navlink">Search for File</a><br/><br/>
 <a href="problem_solution.php" class="navlink">Grinding Problems & Remedies</a><br/><br/>
    <a href="http://www.winterthurusa.net/ss_odplunge.html " class="navlink">Winterthurusa Web
site</a><br/><br/>
 </td>
 <td valign="top" class="content">
 <!--Script 12.1- headder.html -->
```

```
<?php include('header.php');
#script5.8- index.php
$page_title='knowledge warehouse(old part)';
include('headder.html');
?>
<?php
//this code is to select and view an old part case according to the user input
if ($_POST[op] != "view") {
$get_wheelname="SELECT `wheel_id`,`wheel_name` FROM `wheel";
$get_wheelname_res= mysql_query($get_wheelname) or die (mysql_error());
$get_matgroup="SELECT `material_group_id` as id, `name`, `mg_index` FROM `material_group` order by
`name`";
$get_matgroup_res=mysql_query($get_matgroup)or die(mysql_error());
$get_material="SELECT `material_id`,`name` from material order by 'name' ";
$get_material_res=mysql_query($get_material) or die(mysql_error());
   //haven't seen the form, so show it
          $display_block = "<h3>Select one criteria or more to retrieve grinding case/s </h3>";          //get
grinding type from the records
          $get_list="SELECT `grinding_type_id` as id ,`name` FROM `grinding_type` ORDER By `name`";
          $get_list_res=mysql_query($get_list) or die (mysql_error());
 $get_part="SELECT `part_id` FROM `part_details`";
$get_part_res= mysql_query($get_part) or die(mysql_error());
                              //has records, so get the result and dispaly in a form
                    $display_block .= "
<form method=\"post\" action=\"$_SERVER[PHP_SELF]\">
<table cellspacing=1 cellpadding=10 border=0>
<tr>
<td>
   <P><strong>Part Number</strong>
<select name=\"part_id\">
<option value=\"\" >......Select One.... </option>";
while ($recsp=mysql_fetch_array($get_part_res)){
          $id=$recsp['part_id'];
          $display_name=$recsp['part_id'];
    if ($id==$_POST[part_id]) {
          $display_block .= "<option value=\"$id\" selected >";}
   else{
   $display_block .= "<option value=\"$id\">";}
          $display_block .= "$display_name</option>";
          }
$display_block.="
</select>
     </td>
     <td>
   <P><strong>Wheel Name</strong>
<select name=\"sel_wheel\" value="".$_POST[sel_wheel]."">
<option value=\"\" >......Select One.... </option>";
while ($recs=mysql_fetch_array($get_wheelname_res)){
          $id=$recs['wheel_id'];
          $display_name=$recs['wheel_name'];
   if ($id==$_POST[sel_wheel]) {
          $display_block .= "<option value=\"$id\" selected >";}
   else{
   $display_block .= "<option value=\"$id\">";}
          $display_block .= "$display_name</option>";
          }
$display_block.="
</select>
      </td>
      </tr>
      <tr>
      <td>
    <P><strong>Grinding Condition</strong><br>
 <input type=\"radio\" name=\"cond\" value=\"R\">Roughing<br>
  <input type=\"radio\" name=\"cond\" value=\"F\">Finishing<br>
```

```
            </td>
            <td>
                 <P><strong>Material</strong>
<select name=\"sel_material\" onChange=\"getInfo();\"><br>
<option value=\"\" >...Select One... </option>";
while ($recs=mysql_fetch_array($get_material_res))  {
              $id=$recs['material_id'];
              $display_name=$recs['name'];
              $display_block.= "<option value=\"$id\">
              $display_name</option>";
}
$display_block.= "
</select>
                     </td>
                     </tr>
   <tr>
     <td>
              <p><input type= \"submit\" name=\"submit\"
                value=\"View Selected Entry\"></p>
               <input type=\"hidden\" name=\"op\" value=\"view\">
    </td>
    </tr>
               </FORM>";
} else if ($_POST[op]=="view")           {
   // $display_block= " part is $_POST[part_id] ";
            $mat=$_POST[sel_material];
             $wheel=$_POST[sel_wheel] ;
    $display_block="
                         // retrieve the cases by part number and surface condiitons
      if ((($_POST[cond]== 'R') || ($_POST[cond]== 'F') )&& ($mat == "" ) && ($_POST[part_id] != "")&&
($wheel == "") ) {
      $get_var="SELECT cases.case_id, cases.part_id, cases.comment as
com,casesrecord.`wheel_speed`,casesrecord.`width` as
width,casesrecord.`start_diameter`,casesrecord.`finish_diameter` as
fd,casesrecord.`hardness`,casesrecord.`depth_cut` as d, casesrecord.`size_tolerance`,
casesrecord.`roundness`,casesrecord.`wheel_diameter`,casesrecord.`work_speed`,casesrecord.`feed_rate`,casesreco
rd.`dressing_depth`,casesrecord.`dressing_lead`,casesrecord.`dressing_speed`,casesrecord.`roughness`,casesrecord.
`emp_id`,casesrecord.`Date`, casesrecord.`cal_roughness`,casesrecord.`cal_feed_rate`,casesrecord.`cal_depth_cut`
    from casesrecord, cases
       WHERE cases.case_id=casesrecord.case_id
 and cases.part_id='$_POST[part_id]'and `condition_type`='$_POST[cond]' ";
  $display_block="     part and surface con first if";
  }
  //((($_POST[cond]== "") && ($_POST[cond]== "")))
 if  (($mat == "" )&& ($_POST[cond]== "")   && ($_POST[part_id] != "")&& ($wheel == "") )  {
$get_var="SELECT cases.case_id, cases.part_id, cases.comment as com,
casesrecord.`wheel_speed`,casesrecord.`width` as width,casesrecord.`start_diameter`,casesrecord.`finish_diameter`
as fd,casesrecord.`hardness`,casesrecord.`depth_cut` as d, casesrecord.`size_tolerance`,
casesrecord.`roundness`,casesrecord.`wheel_diameter`,casesrecord.`work_speed`,casesrecord.`feed_rate`,casesreco
rd.`dressing_depth`,casesrecord.`dressing_lead`,casesrecord.`dressing_speed`,casesrecord.`roughness`,casesrecord.
`emp_id`,casesrecord.`Date`, casesrecord.`cal_roughness`,casesrecord.`cal_feed_rate`,casesrecord.`cal_depth_cut`
   from casesrecord, cases
       WHERE cases.case_id=casesrecord.case_id
      and cases.part_id='$_POST[part_id]' ";
       $display_block="     retrieve by part id working";
     }
        //retreieve the cases by material
     if (($mat != "" )&& ($_POST[cond]== "")   && ($_POST[part_id] == "")&& ($wheel == "") ){
    $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com,  material.material_id,
cases.part_id, casesrecord.wheel_speed, casesrecord.width AS width, casesrecord.start_diameter,
casesrecord.finish_diameter AS fd, casesrecord.hardness, casesrecord.depth_cut AS d, casesrecord.`size_tolerance`
, casesrecord.`roundness` , casesrecord.wheel_diameter, casesrecord.work_speed, casesrecord.feed_rate,
casesrecord.dressing_depth, casesrecord.dressing_lead, casesrecord.dressing_speed, casesrecord.roughness,
casesrecord.emp_id, casesrecord.Date, casesrecord.cal_roughness, casesrecord.cal_feed_rate,
casesrecord.cal_depth_cut
```

```
        FROM cases, material, casesrecord
        WHERE cases.case_id = casesrecord.case_id
        AND cases.material_id = material.material_id
        AND cases.material_id =$mat";
         $display_block="      if for materila";
    }
 //retrieve material, part and conditions
     if ((($_POST[cond]== 'R') || ($_POST[cond]== 'F') )&& ($mat != "" ) && ($_POST[part_id] != "")&&
($wheel == "") ){
     $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com, material.material_id,
cases.part_id, casesrecord.wheel_speed, casesrecord.width AS width, casesrecord.start_diameter,
casesrecord.finish_diameter AS fd, casesrecord.hardness, casesrecord.depth_cut AS d, casesrecord.`size_tolerance`
, casesrecord.`roundness` , casesrecord.wheel_diameter, casesrecord.work_speed, casesrecord.feed_rate,
casesrecord.dressing_depth, casesrecord.dressing_lead, casesrecord.dressing_speed, casesrecord.roughness,
casesrecord.emp_id, casesrecord.Date, casesrecord.cal_roughness, casesrecord.cal_feed_rate,
casesrecord.cal_depth_cut
        FROM cases, material, casesrecord
        WHERE cases.case_id = casesrecord.case_id
        AND cases.material_id = material.material_id
        AND cases.material_id =$mat
        and cases.part_id='$_POST[part_id]' and `condition_type`='$_POST[cond]'";
         $display_block="      part mat conditions";
    }
 //conition and material are selected
    if ((($_POST[cond]== 'R') || ($_POST[cond]== 'F') )&& ($mat != "" ) && ($_POST[part_id] == "")&&
($wheel == "") ){
     $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com, material.material_id,
cases.part_id, casesrecord.wheel_speed, casesrecord.width AS width, casesrecord.start_diameter,
casesrecord.finish_diameter AS fd, casesrecord.hardness, casesrecord.depth_cut AS d, casesrecord.`size_tolerance`
, casesrecord.`roundness` , casesrecord.wheel_diameter, casesrecord.work_speed, casesrecord.feed_rate,
casesrecord.dressing_depth, casesrecord.dressing_lead, casesrecord.dressing_speed, casesrecord.roughness,
casesrecord.emp_id, casesrecord.Date, casesrecord.cal_roughness, casesrecord.cal_feed_rate,
casesrecord.cal_depth_cut
        FROM cases, material, casesrecord
        WHERE cases.case_id = casesrecord.case_id
        AND cases.material_id = material.material_id
        AND cases.material_id =$mat
        and `condition_type`='$_POST[cond]'";
         $display_block="      mat and conditions";
    }
 //wheel is selected
    if (($mat == "" )&& ($_POST[cond]== "")   && ($_POST[part_id] == "")&& ($wheel != "") ){

    $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com,  wheel.wheel_id, cases.part_id,
casesrecord.wheel_speed, casesrecord.width AS width, casesrecord.start_diameter, casesrecord.finish_diameter AS
fd, casesrecord.hardness, casesrecord.depth_cut AS d, casesrecord.`size_tolerance` , casesrecord.`roundness` ,
casesrecord.wheel_diameter, casesrecord.work_speed, casesrecord.feed_rate, casesrecord.dressing_depth,
casesrecord.dressing_lead, casesrecord.dressing_speed, casesrecord.roughness, casesrecord.emp_id,
casesrecord.Date, casesrecord.cal_roughness, casesrecord.cal_feed_rate, casesrecord.cal_depth_cut
        FROM cases, wheel, casesrecord
        WHERE cases.case_id = casesrecord.case_id
        AND cases.wheel_id = wheel.wheel_id
        AND cases.wheel_id =$wheel";
         $display_block="      wheel if";
    }
          //wheel and material are selected
       if (($mat != "" )&& ($_POST[cond]== "")   && ($_POST[part_id] == "")&& ($wheel != "") ){
    $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com,  wheel.wheel_id,
material.material_id, cases.part_id, casesrecord.wheel_speed, casesrecord.width AS width,
casesrecord.start_diameter, casesrecord.finish_diameter AS fd, casesrecord.hardness, casesrecord.depth_cut AS d,
casesrecord.`size_tolerance` , casesrecord.`roundness` , casesrecord.wheel_diameter, casesrecord.work_speed,
casesrecord.feed_rate, casesrecord.dressing_depth, casesrecord.dressing_lead, casesrecord.dressing_speed,
casesrecord.roughness, casesrecord.emp_id, casesrecord.Date, casesrecord.cal_roughness,
casesrecord.cal_feed_rate, casesrecord.cal_depth_cut
        FROM cases, wheel, casesrecord ,material
```

```
        WHERE cases.case_id = casesrecord.case_id
        AND cases.wheel_id = wheel.wheel_id
        AND cases.material_id = material.material_id
        AND cases.material_id =$mat
        AND cases.wheel_id =$wheel";
         $display_block="      wheel mat if";
    }
   //wheel, part numer and material are selected
     if (($mat != "" )&& ($_POST[cond]== "")  && ($_POST[part_id] != "")&& ($wheel != "") ){
       //retreieve the cases by material
    $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com,  wheel.wheel_id,
material.material_id, cases.part_id, casesrecord.wheel_speed, casesrecord.width AS width,
casesrecord.start_diameter, casesrecord.finish_diameter AS fd, casesrecord.hardness, casesrecord.depth_cut AS d,
casesrecord.`size_tolerance` , casesrecord.`roundness` , casesrecord.wheel_diameter, casesrecord.work_speed,
casesrecord.feed_rate, casesrecord.dressing_depth, casesrecord.dressing_lead, casesrecord.dressing_speed,
casesrecord.roughness, casesrecord.emp_id, casesrecord.Date, casesrecord.cal_roughness,
casesrecord.cal_feed_rate, casesrecord.cal_depth_cut
        FROM cases, wheel, casesrecord ,material
        WHERE cases.case_id = casesrecord.case_id
        AND cases.wheel_id = wheel.wheel_id
        AND cases.material_id = material.material_id
        AND cases.material_id =$mat
        and cases.part_id='$_POST[part_id]'
        AND cases.wheel_id =$wheel";
         $display_block="      wheel, part, mat if";
    }
    //wheel, part, materila and conditons
       if ((($_POST[cond]== 'R') || ($_POST[cond]== 'F') )&& ($mat != "" ) && ($_POST[part_id] != "")&&
($wheel != "") ){
    $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com,  wheel.wheel_id,
material.material_id, cases.part_id, casesrecord.wheel_speed, casesrecord.width AS width,
casesrecord.start_diameter, casesrecord.finish_diameter AS fd, casesrecord.hardness, casesrecord.depth_cut AS d,
casesrecord.`size_tolerance` , casesrecord.`roundness` , casesrecord.wheel_diameter, casesrecord.work_speed,
casesrecord.feed_rate, casesrecord.dressing_depth, casesrecord.dressing_lead, casesrecord.dressing_speed,
casesrecord.roughness, casesrecord.emp_id, casesrecord.Date, casesrecord.cal_roughness,
casesrecord.cal_feed_rate, casesrecord.cal_depth_cut
        FROM cases, wheel, casesrecord ,material
        WHERE cases.case_id = casesrecord.case_id
        AND cases.wheel_id = wheel.wheel_id
        AND cases.material_id = material.material_id
        AND cases.material_id =$mat
        and cases.part_id='$_POST[part_id]'
        and `condition_type`='$_POST[cond]'
        AND cases.wheel_id =$wheel";
         $display_block="      wheel part mat cond if";
    }
    //wheel and part number
      if (($mat == "" )&& ($_POST[cond]== "")   && ($_POST[part_id] != "")&& ($wheel != "") ){
    $get_var="SELECT cases.case_id, cases.material_id, cases.comment as com,  wheel.wheel_id, cases.part_id,
casesrecord.wheel_speed, casesrecord.width AS width, casesrecord.start_diameter, casesrecord.finish_diameter AS
fd, casesrecord.hardness, casesrecord.depth_cut AS d, casesrecord.`size_tolerance` , casesrecord.`roundness` ,
casesrecord.wheel_diameter, casesrecord.work_speed, casesrecord.feed_rate, casesrecord.dressing_depth,
casesrecord.dressing_lead, casesrecord.dressing_speed, casesrecord.roughness, casesrecord.emp_id,
casesrecord.Date, casesrecord.cal_roughness, casesrecord.cal_feed_rate, casesrecord.cal_depth_cut
        FROM cases, wheel, casesrecord
        WHERE cases.case_id = casesrecord.case_id
        AND cases.wheel_id = wheel.wheel_id
        and cases.part_id='$_POST[part_id]'
        AND cases.wheel_id =$wheel";
    }
    //wheel, part, and conditions
       if ((($_POST[cond]== 'R') || ($_POST[cond]== 'F') )&& ($mat == "" ) && ($_POST[part_id] != "")&&
($wheel != "") ){
    $get_var="SELECT cases.case_id,cases.comment as com,  wheel.wheel_id, cases.part_id,
casesrecord.wheel_speed, casesrecord.width AS width, casesrecord.start_diameter, casesrecord.finish_diameter AS
```

```
fd, casesrecord.hardness, casesrecord.depth_cut AS d, casesrecord.`size_tolerance` , casesrecord.`roundness` ,
casesrecord.wheel_diameter, casesrecord.work_speed, casesrecord.feed_rate, casesrecord.dressing_depth,
casesrecord.dressing_lead, casesrecord.dressing_speed, casesrecord.roughness, casesrecord.emp_id,
casesrecord.Date, casesrecord.cal_roughness, casesrecord.cal_feed_rate, casesrecord.cal_depth_cut
        FROM cases, wheel, casesrecord
        WHERE cases.case_id = casesrecord.case_id
        AND cases.wheel_id = wheel.wheel_id
        and cases.part_id='$_POST[part_id]'
        and `condition_type`='$_POST[cond]'
        AND cases.wheel_id =$wheel";
         $display_block="      wheel part cond";
    }
    // wheel condition
         if ((($_POST[cond]== 'R') || ($_POST[cond]== 'F') )&& ($mat == "" ) && ($_POST[part_id] == "")&&
($wheel != "") ){
 $get_var="SELECT cases.case_id, cases.comment as com,  wheel.wheel_id, cases.part_id,
casesrecord.wheel_speed, casesrecord.width AS width, casesrecord.start_diameter, casesrecord.finish_diameter AS
fd, casesrecord.hardness, casesrecord.depth_cut AS d, casesrecord.`size_tolerance` , casesrecord.`roundness` ,
casesrecord.wheel_diameter, casesrecord.work_speed, casesrecord.feed_rate, casesrecord.dressing_depth,
casesrecord.dressing_lead, casesrecord.dressing_speed, casesrecord.roughness, casesrecord.emp_id,
casesrecord.Date, casesrecord.cal_roughness, casesrecord.cal_feed_rate, casesrecord.cal_depth_cut
        FROM cases, wheel, casesrecord
        WHERE cases.case_id = casesrecord.case_id
        AND cases.wheel_id = wheel.wheel_id
         and `condition_type`='$_POST[cond]'
        AND cases.wheel_id =$wheel";
         $display_block="      wheel cond if last";
    }
$get_var_res=mysql_query($get_var)or die(mysql_error());
   $countrows=mysql_num_rows($get_var_res);
                if ($countrows==1){
                                $display_block.="There is one case for $_POST[part_id] <br>";
                } else {
                                $display_block.="<li> <strong> Number of Records:  $countrows </strong> <br>";
                }
                                                $dispaly_block .= " </ul>";
$count=1;
       while ($variable_info=mysql_fetch_array($get_var_res)){
          $caseid=$variable_info[case_id];
  //get material and material group variable
                $get_material="select material.name as mname, case_id from cases,material where
cases.material_id=material.material_id and cases.case_id='$caseid'";
                $get_material_res= mysql_query($get_material)or die(mysql_error());
                while ($res=mysql_fetch_array($get_material_res)){
                                    $material=$res[mname];
}
       //get material group
                $get_mgroup = "select material_group.name as gname, cases.case_id from cases, material_group where
cases.material_group_id=material_group.material_group_id and cases.case_id='$caseid'";
                $get_mgroup_res=mysql_query($get_mgroup)or die(mysql_error());
                                while ($res=mysql_fetch_array($get_mgroup_res)){
                                    $mgroup=$res[gname];
                                                                }
          $get_dresser="select cases.case_id,dresser.name as dressername ,casesrecord.dressing_depth as
depth,casesrecord.dressing_lead as lead
               from cases,dresser, casesrecord
                where cases.dresser_id=dresser.dresser_id and cases.case_id='$caseid'";
 $get_dresser_res=mysql_query($get_dresser)or die(mysql_error());
   while ($dresser_info=mysql_fetch_array($get_dresser_res)){
                      $dresser=$dresser_info[dressername];
                      $depth=$dresser_info[depth];
                      $lead=$dresser_info[lead];
       }
               //get grinding wheel variables
```

```
  $get_wheelname="select  wheel.wheel_name as wheelname,
cases.case_id,wheel.grade,wheel.abrasive_type,wheel.bond from cases,wheel where cases.wheel_id =
wheel.wheel_id and  cases.case_id='$caseid'";
  $get_wheel_res = mysql_query($get_wheelname)or die(mysql_error());
   while ($wheel_info = mysql_fetch_array($get_wheel_res)){
        $wheel=$wheel_info[wheelname];
                $grade=$wheel_info[grade];
        $abrasive_type=$wheel_info[abrasive_type];
        $bond=$wheel_info[bond];
    }
     //get the machine details
   $get_machine="select machine.name as mname,cases.case_id from machine,cases where
cases.machine_id=machine.machine_id and cases.case_id='$caseid'";
   $get_machine_res=mysql_query($get_machine)or die(mysql_error());
   while ($res=mysql_fetch_array($get_machine_res)){
                          $machine=$res[mname];
                                  }
            //get the coolant details
   $get_coolant="select coolant.name as cname,cases.case_id from cases,coolant where
cases.coolant_id=coolant.coolant_Id and cases.case_id='$caseid'";
   $get_coolant_res=mysql_query($get_coolant)or die(mysql_error());
   while ($res=mysql_fetch_array($get_coolant_res)){
                          $coolant=$res[cname];
                       }
     // get the employee name
   $get_emp= "SELECT casesrecord.case_id,emp.first_name as fn, emp.last_name as ln, CASESRECORD.Date as
d FROM casesrecord,emp  WHERE  casesrecord.emp_id = emp.emp_id and casesrecord.case_id='$caseid' ";
         $get_emp_res=mysql_query($get_emp)or die(mysql_error());
         while($res=mysql_fetch_array($get_emp_res)) {
         $last_n=$res[ln];
         $first_n=$res[fn];
         $date=$res[d];
         }
                $work_speed=$variable_info[work_speed];
        $comm=$variable_info[com];
                $wheel_speed=$variable_info[wheel_speed];
                $feed_rate=$variable_info[feed_rate];
                $width=$variable_info[width];
                $start_diameter=$variable_info[start_diameter];
                $finish_diameter=$variable_info[fd];
                $hardness=$variable_info[hardness];
                $roughness2=$variable_info[roughness];
                $roundness=$variable_info[roundness];
                $dressing_depth=$variable_info[dressing_depth];
                $dressing_lead=$variable_info[dressing_lead];
        $depth=$variable_info[d];
        $cal_depth=$variable_info[cal_depth_cut];
        $cal_rough=$variable_info[cal_roughness];
        $cal_feed=$variable_info[cal_feed_rate];
        $depth_cut=$variable_info[cdepth];
        if ($cal_depth== '0' ){
             $cal_depth=' ';
             } else{
             $cal_depth='Calculated';}
   //for feed rate
             if ($cal_feed=='0'){
             $cal_feed='';
             }   else {
             $cal_feed='Calculated';}
    //for roughness
             if ($cal_rough=='0'){
             $cal_rough='';
             } else{
             $cal_rough='Calculated';}
  $display_block .="
```

```
        <table cellspacing=1 cellpadding=10 border=3>
<tr>
<td>
        <li><h4> Result  $count</h3>
                        <li><strong>Comment</strong>$comm
                        <li><strong>Employee Name</strong> $first_n $last_n
                        <li><strong>Date</strong>$date
                        <li><strong> Matrial </strong>$material
                        <li><strong> Matrial Group</strong> $mgroup
                        <li><strong>Dresser name is $dresser</strong> <br>
                        <li><strong> Wheel name $wheel </strong><br>
       <li>Grade is $grade abrasive , type is $abrasive_type , bond type is $bond
       <li> <strong>Machine </strong>$machine
       <li><strong>Coolant</strong> $coolant  </td>
       <td>
                                                  <li><strong> Work Speed </strong> $work_speed
m/s
                              <li><strong>Wheel Speed</strong> $wheel_speed m/s
                      <li><strong>Depth of cut </strong> $depth mm $cal_depth
                              <li><strong>Dressing Depth</strong>$dressing_depth mm
                              <li><strong>Dressing Lead</strong>$dressing_lead mm
                              <li><strong>Feed Rate </strong>$feed_rate  mm  $cal_feed
                              <li><strong>Grinding Width </strong>$width   mm
                              <li><strong>Start Diameter</strong> $start_diameter  mm
                              <li><strong>Finish Diameter </strong>$finish_diameter  mm
                              <li><strong>Hardness </strong>$hardness HRC
                              <li><strong>Roughness </strong>  $roughness2 micro-m $cal_rough
                              <li><strong>Roundness </strong>$roundness micro-m</td>
                      </tr>";
                              $count++;
  }
                              }
      $display_block.="</ul>";
 ?>
<HTML>
<HEAD>
<TITLE>Retrived Case</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block"; ?>
</BODY>
</HTML>
<?php include('header.php');
#script5.8- index.php
$page_title='knowledge warehouse (new part)';
include('headder.html');
?>
<?php
//this code will allow user to input the case varaible
if ($_POST[op] != "view") {
$get_matgroup="SELECT `material_Group_id` as id, `name`, `mg_index` FROM `material_group` ORDER By
`name`";
$get_matgroup_res=mysql_query($get_matgroup)or die(mysql_error());
$get_material="SELECT `material_id`,`name`FROM `material` `name`
order by `name`";
$get_material_res=mysql_query($get_material) or die(mysql_error());
$display_block="<h3>User Input</h3>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\">
<P><strong> Material Group</strong>
<select name=\"sel_matgroup\">
<option value=\"\" >..Select One...</option>";
while ($recs=mysql_fetch_array($get_matgroup_res))  {
        $id= $recs['id'];
        $display_name=$recs['name'];
        $display_block .= "<option value=\"$id\">
```

```
            $display_name</option>";
}
$display_block.= "
</select>
<P><strong>Material</strong><br>
<select name=\"sel_material\">
<option value=\"\" >...Select One... </option>";
while ($recs=mysql_fetch_array($get_material_res))  {
            $id=$recs['material_id'];
            $display_name=$recs['name'];
            $display_block.= "<option value=\"$id\">
            $display_name</option>";
}
$display_block.= "
</select>
<P><strong>Hardness</strong><br>
<input type=\"radio\" name=\"hardness\" value=\"<50RC\" ><50RC<br>
<input type=\"radio\" name=\"hardness\" value=\"<50-58RC\" >50-58RC<br>
<input type=\"radio\" name=\"hardness\" value=\">58RC\" >>58RC<br>
<P><strong>Sureface Conditions</strong><br>
<input type=\"radio\" name=\"surfce_cond\" value=\"Rough\" >Rough<br>
<input type=\"radio\" name=\"surfce_cond\" value=\"Interupted Cut\" >Interupted Cut<br>
<P><strong>Wheel Selection</strong><br>
<input type=\"radio\" name=\"wheel_sel\" value=\"By System\" checked>By System<br>
<input type=\"radio\" name=\"wheel_sel\" value=\"By User\" >By User<br>
<P><strong>Wheel Selection By User</strong><br>
<P><strong>Abrasive Type</strong><br>
<input type=\"radio\" name=\"abrasive_type\" value=\"A\" >Aluminum Oxide
<input type=\"radio\" name=\"abrasive_type\" value=\"C\" >Silicon Carbide
<input type=\"radio\" name=\"abrasive_type\" value=\"B\" >CBN<br>
<P><strong>Bond Type</strong><br>
<input type=\"radio\" name=\"bond_type\" value=\"V\" >Vitrified
<input type=\"radio\" name=\"bond_type\" value=\"E\">Shellac
<input type=\"radio\" name=\"bond_type\" value=\"B\">Resin
<input type=\"radio\" name=\"bond_type\" value=\"R\">Rubber
<input type=\"radio\" name=\"bond_type\" value=\"M\">Metal<br>
<P><strong>Grits Size</strong>
<input type=\"radio\" name=\"grits\" value=\"A\" >10
<input type=\"radio\" name=\"grits\" value=\"B\" >12
<input type=\"radio\" name=\"grits\" value=\"C\" >14
<input type=\"radio\" name=\"grits\" value=\"D\" >16
<input type=\"radio\" name=\"grits\" value=\"E\" >18<br>

<P><strong>Wheel Diameter</strong>
<input type=\"text\" name=\"w_diameter\" size=30><br>
<P><strong>Wheel Speed (mm/s)</strong>
<input type=\"text\" name=\"w_speed\" size=30><br>
<P><strong>Start Diameter (mm)</strong>
<input type=\"text\" name=\"start_diam\" size=30><br>
<P><strong>Finish Diameter (mm)</strong>
<input type=\"text\" name=\"finish_diam\" size=30><br>
<P><strong>Work Speed (mm/s)</strong>
<input type=\"text\" name=\"work_speed\" size=30><br>
<P><strong>Max Roughness Ra(micro meter)</strong>
<input type=\"text\" name=\"roughness\" size=30><br>
<P><strong>Width (mm)</strong>
<input type=\"text\" name=\"width\" size=30><br>
<P><strong>Size Tolerance (micro meter)</strong>
<input type=\"text\" name=\"size_tolerance\" size=30><br>
<p><input type=\"submit\" name=\"submit\" value=\"OK\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
} else if ($_POST[op]=="view")          {
//check for the required fields
```

```
        if (($_POST[sel_matgroup]== "") || ($_POST[sel_material]== "") || ($_POST[hardness]== "") ||
($_POST[roughness]=="") || ($_POST[width]== "")) {
                header ("Location: newpart.php");
                exit;
        }
//***********************Assign index for material group(primary index) *****
    if ($_POST[sel_matgroup]== 1) {
            $matgroup_index="LS";
            //echo "materila group is $matgroup_index<br>";
    } else if ($_POST[sel_matgroup]== 4) {
            $matgroup_index="HS";
            //echo "materila group is $matgroup_index<br>";
    } else if ($_POST[sel_matgroup]== 2) {
             $matgroup_index="TS";
            //echo "materila group is $matgroup_index<br>";
    } else if ($_POST[sel_matgroup]== 3) {
            $matgroup_index="CI";
            //echo "materila group is $matgroup_index<br>";
    } else {
            //echo "the material is not available";
    }

 //***********************Assign an index for the hardness (primary index)*****************
    if ($_POST[hardness]== "<50RC") {
          $hardness_index="S";
                    }else if ($_POST[hardness]=="<50-58RC"){
           $hardness_index="M";
  }else if ($_POST[hardness]==">58RC"){
           $hardness_index="H";
           }
 //**************Assign an index for the roughness (primary index*****************
    if (($_POST[roughness]>=1.0) && ($_POST[roughness]<= 1.60)){
            $roughness_index=1;
    } else if (($_POST[roughness]>=0.70)&&($_POST[roughness]<= 0.99)){
            $roughness_index=2;
    } else if (($_POST[roughness]>=0.40)&&($_POST[roughness]<= 0.69)){
            $roughness_index=3;
    } else if (($_POST[roughness]>=0.20)&&($_POST[roughness]<= 0.39)){
            $roughness_index=4;
             //echo"roughness index $roughness_index";
    } else if (($_POST[roughness]>=0.10)&&($_POST[roughness]<= 0.19)){
            $roughness_index=5 ;
    } else{
            echo "the roughness value is not accepted";
    }
    if ($_POST[wheel_sel]=="By System") {

          $get_case= "SELECT * FROM `casesrecord` WHERE `matgroup_index` = '$matgroup_index' AND
`roughness_index` = '$roughness_index' AND `hardness_index`= '$hardness_index'";  $get_cases_res=
mysql_query($get_case) or die (mysql_error());
  while ($arr=mysql_fetch_array($get_cases_res)) {
        $caseid=$arr['caserec_id'];
        $vindex=$arr['v_index'];
        $insert_copycases="INSERT INTO copy_casea SELECT * FROM casesrecord WHERE `caserec_id`=
$caseid";
        mysql_query($insert_copycases) or die(mysql_error());
        }
            if (mysql_num_rows($get_cases_res)<1){
                    //that means no matced cases. Then the rule base system will be lunced.
                    echo "<p>the rule based system will be lunched</p>";
                    header ("Location: RuleBased.php");
                exit;
  }
      //Calculate Similarity value
      $get_matched_cases= "select * from copy_casea ";
```

```
$get_matched_cases_res= mysql_query($get_matched_cases) or die(mysql_error());
//go throgh each case to calculate the similarity value
while ($result = mysql_fetch_array($get_matched_cases_res)){
        $matchedcase=$result[case_id];
        $matchedmaterial=$result[material_id];
        $matchedroughness=$result[roughness];
        $matchedwork_speed=$result[work_speed];
                if ($_POST[sel_material]==$matchedmaterial){
                $sim_material=1;
                $sim_roughness= 1-((($_POST[roughness]-
$matchedroughness)/0.3)*($_POST[roughness]-$matchedroughness)/0.3);
                $sim_work_speed=(1-($_POST[work_speed]-
$matchedwork_speed)/500)*($_POST[work_speed]-$matchedwork_speed)/500;
                $sim= (1+ ($sim_roughness* 0.6)+ ($sim_work_speed*0.4))/2
                $update_sim= "UPDATE copy_casea set sim=$sim where case_id=$matchedcase";
                mysql_query($update_sim) or die (mysql_error());
                        }
        if ($_POST[sel_material]!=$matchedmaterial){
                $sim_roughness= (1-(($_POST[roughness]-
$matchedroughness)/0.3)*($_POST[roughness]-$matchedroughness)/0.3);
                $sim_work_speed=(1-($_POST[work_speed]-
$matchedwork_speed)/500)*($_POST[work_speed]-$matchedwork_speed)/500;
                 $sim= ($sim_roughness* 0.6)+ ($sim_work_speed*0.4);
  $update_sim= "UPDATE copy_casea set sim=$sim where case_id=$matchedcase";
                mysql_query($update_sim) or die (mysql_error());
                }
        }
 //end of system if statment
    }
        if ($_POST[wheel_sel]=="By User") {

       $get_cases="SELECT * FROM `casesrecord` WHERE `matgroup_index` = '$matgroup_index' AND
`roughness_index` = '$roughness_index' AND `hardness_index`= '$hardness_index' AND `abrasivetype_index` =
'$_POST[abrasive_type]'
    AND `bondtype_index`='$_POST[bond_type]'
    AND`grits_index`='$_POST[grits]'";
       $get_cases_res= mysql_query($get_cases) or die (mysql_error());

       //we have to make sure that there is at least one matched case...
        if (mysql_num_rows($get_cases_res)<1){
                //that means no matced cases. Then the rule base system will be lunced.
                echo "the rule based system will lunched";
        }
       while ($arr=mysql_fetch_array($get_cases_res)) {
        $caseid=$arr['caserec_id'];
        $vindex=$arr['v_index'];
  $insert_copycases="INSERT INTO copy_casea SELECT * FROM casesrecord WHERE `caserec_id`=
$caseid";
       mysql_query($insert_copycases) or die(mysql_error());
                }
//calculate the similarity value
 $get_matched_cases= " select * from copy_casea";
$get_matched_cases_res= mysql_query($get_matched_cases) or die(mysql_error());
while ($result = mysql_fetch_array($get_matched_cases_res)){
        $matchedcase=$result[case_id];
        $matchedmaterial=$result[material];
        $matchedroughness=$result[roughness];
        $matchedwork_speed=$result[work_speed];
        $matchedwheel_speed=$result[wheel_speed];
        $matchedwheel_diam=$result[wheel_diameter];
                if ($_POST[sel_material]==$matchedmaterial){
                $sim_material=1;
                $sim_roughness= 1-(($_POST[roughness]-
$matchedroughness)/0.3)*(($_POST[roughness]-$matchedroughness)/0.3);
```

```
                                $sim_work_speed=(1-($_POST[work_speed]-
$matchedwork_speed)/500)*(($_POST[work_speed]-$matchedwork_speed)/500); $sim_wheel_speed=(1-
($_POST[wheel_speed]-$matchedwheel_speed)/15)*(($_POST[wheel_speed]*$matchedwheel_speed)/15);
$sim_wheel_diam= 1-(($_POST[wheel_diameter]-$matchedwheel_speed)/100)*(($_POST[wheel_diameter]-
$matchedwheel_speed)/100);
 $sim= (1+ ($sim_roughness* 0.6)+
($sim_work_speed*0.4)+(0.4*$sim_wheel_speed)+(0.4*$sim_wheel_diam))/2.8;
$update_sim= "UPDATE copy_casea set sim=$sim where case_id=$matchedcase";
                                mysql_query($update_sim) or die (mysql_error());
                        }
                     if ($_POST[sel_material]!=$matchedmaterial){
  $sim_roughness= (1-(($_POST[roughness]-$matchedroughness)/0.3)*($_POST[roughness]-
$matchedroughness)/0.3);
$sim_work_speed=(1-($_POST[work_speed]-$matchedwork_speed)/500)*(($_POST[work_speed]-
$matchedwork_speed)/500);
$sim_wheel_speed=(1-($_POST[wheel_speed]-
$matchedwheel_speed)/15)*(($_POST[wheel_speed]*$matchedwheel_speed)/15);
$sim_wheel_diam= 1-(($_POST[wheel_diameter]-$matchedwheel_speed)/100)*(($_POST[wheel_diameter]-
$matchedwheel_speed)/100);
$sim= ($sim_roughness* 0.6)+ ($sim_work_speed*0.4)+(0.4*$sim_wheel_speed)+(0.4*$sim_wheel_diam)/1.8;
$update_sim= "UPDATE copy_casea set sim=$sim where case_id=$matchedcase";
mysql_query($update_sim) or die (mysql_error());
                     }
                  }
             }
$get_max_sim="SELECT * FROM copy_casea WHERE sim= (Select Max(sim) FROM copy_casea)";
$get_max_sim_res= mysql_query($get_max_sim) or die(mysql_error());
//match the problem definition and must be modified.
                While ($modify_result = mysql_fetch_array($get_max_sim_res))      {
                               $max_caseid=$modify_result[case_id];
                               $max_feedrate=$modify_result[feed_rate];
                               $max_workspeed=$modify_result[work_speed];
                               $max_dressinglead=$modify_result[dressing_lead];

                               $max_wheelspeed=$modify_result[wheel_speed];
                               $max_roughness=$modify_result[roughness];
                               $max_workdiameter=$modify_result[start_diameter];
                               $max_wheeldiameter=$modify_result[wheel_diameter];
                               if ($_POST[wheel_sel]=="By System"){

$eqdiameter=(($_POST[start_diam]*$max_wheeldiameter)/($_POST[start_diam]+$max_wheeldiameter));
$reco_feedrate=
($max_feedrate*$max_workdiameter*$max_wheelspeed)/($max_wheelspeed*$_POST[start_diam]);

$reco_dressinglead=($max_dressinglead*$_POST[roughness]*$_POST[roughness])/(($max_roughness)*($max_r
oughness));
                               $max_eqdiam=
(($max_workdiameter*$max_wheeldiameter)/($max_workdiameter+$max_wheeldiameter));
                               $reco_workspeed=
($_POST[work_speed]*($max_wheelspeed*$eqdiameter)/($max_eqdiam*$max_wheelspeed));
$eqdiameter=sprintf("%.4f",$eqdiameter);
                               $reco_feedrate=sprintf("%.4f",$reco_feedrate);
                               $reco_dressinglead=sprintf("%.4f",$reco_dressinglead);
                               $max_eqdiam=sprintf("%.4f",$max_eqdiam);
                               $reco_workspeed= sprintf("%.4f",$reco_workspeed);}
                      if ($_POST[wheel_sel]=="By User"){
$eqdiameter=(($_POST[start_diam]*$_POST[w_diameter])/($_POST[start_diam]+$_POST[w_diameter]));
$reco_feedrate=
($max_feedrate*$max_workdiameter*$_POST[w_speed])/($max_wheelspeed*$_POST[start_diam]);

$reco_dressinglead=($max_dressinglead*$_POST[roughness]*$_POST[roughness])/(($max_roughness)*($max_r
oughness));
 $max_eqdiam= (($max_workdiameter*$max_wheeldiameter)/($max_workdiameter+$max_wheeldiameter));
$reco_workspeed=
($_POST[work_speed]*($max_wheelspeed*$eqdiameter)/($max_eqdiam*$_POST[w_speed]));
```

```
$eqdiameter=sprintf("%.4f",$eqdiameter);
$reco_feedrate=sprintf("%.4f",$reco_feedrate);
$reco_dressinglead=sprintf("%.4f",$reco_dressinglead);
$max_eqdiam=sprintf("%.4f",$max_eqdiam);
$reco_workspeed=sprintf("%.4f",$reco_workspeed);
      }
}
//the recommended case is located and modified..
//to display the resul
$get_macase="select
cases.case_id,cases.machine_id,cases.coolant_id,cases.wheel_id,cases.part_id,cases.dresser_id
            from cases,copy_casea
            where cases.case_id=copy_casea.case_id
            and copy_casea.case_id=$max_caseid";
$get_macase_res=mysql_query($get_macase) or die(mysql_error());
while ($rec_det=mysql_fetch_array($get_macase_res)){
$caseid=$rec_det[case_id];
$machine=$rec_det[machine_id];
        $partid=$rec_det[part_id];
        $materialgroup=$_POST[sel_matgroup];
        $wheel=$rec_det[wheel_id];
        $dresser=$rec_det[dresser_id];
        $coolant=$rec_det[coolant_id];
        }
$get_reco="select * from copy_casea where case_id=$max_caseid";
$get_reco_res= mysql_query($get_reco) or die(mysql_error());
$display_block = "<h1>The Recommended Case</h1>";
while ($res_det=mysql_fetch_array($get_reco_res)){
        $index=$res_det[v_index];
        $wheelspeed=$res_det[wheel_speed];
        $workspeed=$reco_workspeed;
        $roughness=$_POST[roughness];
        $feedrate=$reco_feedrate;
        $material=$_POST[sel_material];
        $hardness=$_POST[hardness];
        $startdiam=$_POST[start_diam];
        $finishdiam=$res_det[finish_diameter];
        $width=$_POST[width];
        $wheeldiam=$res_det[wheel_diameter];
        $eqdiam=$max_eqdiam;
        $dressingdepth=$res_det[dressing_depth];
        $dressinglead=$reco_dressinglead;
        $bondtype_index=$res_det[bondtype_index];
        $abrasivetype_index=$res_det[abrasivetype_index];
        $grits_index=$res_det[grits_index];
        $display_block .= "<li> the case id $caseid <br>
                <li>Part number $partid <br>
                <li>Index
$matgroup_index$hardness_index$_POST[abrasive_type]$_POST[bond_type]$_POST[grits]$roughness_index<b
r>
                <li>Machine $machine <br>
                <li>Wheel Speed $wheelspeed <br>
                <li>Work Speed $reco_workspeed<br>
                <li>Roughness $roughness<br>
                <li>Feed Rate $feedrate <br>
                <li>Coolant $coolant<br>
                <li>Material Group $materialgroup <br>
                <li>Material $material<br>
                <li>Hardness $hardness<br>
                <li>Start Diameter $startdiam<br>
                <li>Finish Diameter $finishdiam<br>
                <li>Width $width<br>
                <li>Wheel $wheel<br>
                <li>Wheel Diameter $wheeldiam<br>
                <li>Equivelent Diameter $eqdiam<br>
```

```
                                                     <li>Dresser $dresser <br>
                                                     <li>Dressing Depth $dressingdepth <br>
                                                     <li>Dressing Lead $dressinglead <br>";     }

                           $display_block .= "</u>";
//save if the user happy with the result
$display_block .= "</u>";
                          if ($_POST[wheel_sel]=="By User"){
echo "well done2";
              }
if ($_POST[wheel_sel]=="By System"){
        echo "well done";}
//mysql_query($insert_new_case) or die(mysql_error());
                                      //delete the records in the temporary table
                                       $del_copycases="Delete from copy_casea";
                     mysql_query($del_copycases) or die(mysql_error());
}
 ?>
 <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTM>
<?php include('header.php');
#script5.8- add_new_grindingcase.php
$page_title='Managing Grinding Cases';
include('headder_manage_grinding.html');
session_unregister('error') ;
session_unregister('notify');
$showForm = 1;
?>
<?php
//this code will allow user to input the case varaible
$str="<font color='#ff0000'> * </font>";
if ($_POST[op]=="view")      {
            //if the user click add button
if ($_POST['submit']){
//*** validate part number
//$part_id = trim($_POST['part_id']);
If (($_POST[sel_part])=="")
{
$_SESSION['error'] .="Part number should be selected <br>";
}
else
{
//accept part number entry and sanitize it
$part_id = mysql_real_escape_string(stripslashes($part_id));
}
//validate hardness field
$hardness = trim($_POST['hardness']);
If (empty($hardness) || (!(ctype_digit($hardness))))
{
$_SESSION['error'] .="Hardness filed is either empty or Enter only NUMERIC characters <br>";
}
else
{
//accept hardness entry and sanitize it
$hardness = mysql_real_escape_string(stripslashes($hardness));
}
//**validate start diameter filed
$start_diam = trim($_POST['start_diam']);
If (empty($start_diam) || (!(is_numeric($start_diam))))
```

```
{
$_SESSION['error'] .="Start Diameter filed is either empty or Enter only NUMERIC characters <br>";
}
else
{
//accept start diameter entry and sanitize it
$start_diam = mysql_real_escape_string(stripslashes($start_diam));
}
//*** validate work speed
$work_speed = trim($_POST['work_speed']);
If (empty($work_speed) || (!(is_numeric($work_speed))))
{
$_SESSION['error'] .="Work speed filed is either empty or Enter only NUMERIC characters<br>";
}
else
{
//accept work speed entry and sanitize it
$work_speed = mysql_real_escape_string(stripslashes($work_speed));
}
//*** validate wheel name
//$w_diameter = trim($_POST['w_diameter']);
If (empty($w_diameter))
{
$_SESSION['error'] .="Wheel filed is not selected !<br>";
}
//*** validate wheel speed
$w_speed = trim($_POST['w_speed']);
If (empty($w_speed) || (!(ctype_digit($w_speed))))
{
$_SESSION['error'] .="Wheel speed filed is either empty or Enter only NUMERIC characters<br>";
}
else
{
//accept wheel speed entry and sanitize it
$w_speed = mysql_real_escape_string(stripslashes($w_speed));
}
//*** validate wheel diameter
$w_diameter = trim($_POST['w_diameter']);
If (empty($w_diameter) || (!(is_numeric($w_diameter))))
{
$_SESSION['error'] .="Wheel diameter filed is either empty or Enter only NUMERIC characters<br>";
}
else
{
//accept work diameter entry and sanitize it
$w_diameter = mysql_real_escape_string(stripslashes($w_diameter));

}
//*******wheel width**********
$width=trim($_POST[width]);

 if (!(empty($width))){
       If (!(is_numeric($width))){
          $_SESSION['error'] .="<br> For Width filed: Enter only NUMERIC characters<br>";
        } else{
                //accept roughness entry and sanitize it
       $width = mysql_real_escape_string(stripslashes($width));

       }
   }

//*** dressing depth
$d_depth = trim($_POST['d_depth']);
If (empty($d_depth) || (!(is_numeric($d_depth))))
{
```

```
$_SESSION['error'] .="Dressing depth filed is either empty or Enter only NUMERIC characters<br>";
}
else
{
//accept dressing entry and sanitize it
$d_depth = mysql_real_escape_string(stripslashes($d_depth));

}
 //*** validate  dressing lead entry
$d_lead = trim($_POST['d_lead']);
If (empty($d_lead) || (!(is_numeric($d_lead))))
{
$_SESSION['error'] .="Dressing lead filed is either empty or Enter only NUMERIC characters<br>";
}
else
{
//accept work diameter entry and sanitize it
$d_lead = mysql_real_escape_string(stripslashes($d_lead));

}
 //*** validate feed rate entry
$feed_rate = trim($_POST['feed_rate']);
If (!(empty($feed_rate))){
      If (!(is_numeric($feed_rate)))
      {
     $_SESSION['error'] .="<br> For Feed rate filed: Enter only NUMERIC characters<br>";
       }
       else
     {
       //accept work diameter entry and sanitize it
        $feed_rate = mysql_real_escape_string(stripslashes($feed_rate));

        }
   }
 //*** validate depth of cut
$depth_cut = trim($_POST['depth_cut']);

If (!(empty ($depth_cut))){
          If (!(is_numeric($depth_cut)))
              {
            $_SESSION['error'] .="<br> For Depth of cut filed: Enter only NUMERIC characters<br>";
               }
           else
           {
      //accept work diameter entry and sanitize it
       $depth_cut = mysql_real_escape_string(stripslashes($depth_cut));

          }
      }

 //********validate roughness field
 $roughness= trim($_POST[roughness]);
 If (!(empty($roughness))){
        If (!(is_numeric($roughness)))
        {
        $_SESSION['error'] .="<br> For Roughness filed: Enter only NUMERIC characters<br>";
        } else{
               //accept roughness entry and sanitize it
       $roughness = mysql_real_escape_string(stripslashes($roughness));

       }
  }

 //***********  validate volumetric reomval *****
 $volum_remo=trim($_POST[volum_remo]);
```

```
 If (!(empty($volum_remo))){
          If (!(is_numeric($volum_remo))){
           $_SESSION['error'] .="<br> For Volumetric Removal Rate filed: Enter only NUMERIC characters<br>";
        } else{
                //accept roughness entry and sanitize it
       $volum_remo = mysql_real_escape_string(stripslashes($volum_remo));

       }
   }
 //**********size tolerance**********
 $size_tolerance=trim($_POST[size_tolerance]);
 if (!(empty($size_tolerance))){
       If (!(is_numeric($size_tolerance))){
           $_SESSION['error'] .="<br> For Size Tolerance filed: Enter only NUMERIC characters<br>";
        } else{
                //accept roughness entry and sanitize it
       $size_tolerance = mysql_real_escape_string(stripslashes($size_tolerance));

       }
   }

 //*********roundness
$round=trim($_POST[roundness]);
 if (!(empty($round))){
       If (!(is_numeric($round))){
           $_SESSION['error'] .="<br> For Roundness filed: Enter only NUMERIC characters<br>";
        } else{
                //accept roughness entry and sanitize it
       $round = mysql_real_escape_string(stripslashes($round));

       }
   }
//***********check for required parameters are there the case will be saved**************
if (($_POST[sel_matgroup]== "")|| ($_POST[sel_material]== "") ) {
                         $_SESSION['error'] .=" The materil or materila group is not selected !";
}else
   {
             //***********check for feed rate first***********
    $cal_roughness=0;
    $cal_feed_rate=0;
    $cal_depth_cut=0;

  if (($_POST[feed_rate]=="") && ($_POST[width]!="")  && ($_POST[volum_remo]!="") &&
($_POST[start_diam]!="")){
                         $feed_rate= $_POST[volum_remo]/(3.14*$_POST[start_diam]);
      $cal_feed_rate=1;
       $_SESSION['notify'] .= "Feed rate has been calulated . The value of feed rate is $feed_rate mm, inorder to
add the case key inn the depth of cut value <br>";
} elseif (($_POST[feed_rate]=="") && ($_POST[depth_cut]!="")  && ($_POST[work_speed]!="") &&
($_POST[start_diam]!="")){
              //multiplay by 1000 to transfer m to mm for work speed.
       $feed_rate= 1000*$_POST[depth_cut]*$_POST[work_speed]/(3.14*$_POST[start_diam]);
       $cal_feed_rate=1;
       $_SESSION['notify'] .= "Feed rate has been calulated . The value of feed rate is $feed_rate mm.  <br>";

      }
elseif (($_POST[feed_rate]=="")){
                         $_SESSION['error'] .= " feed rate could not be calulated <br>";
                                      }
//*************check for the depth of cut value,if it is not available then caculate it*******
if (($_POST[depth_cut]=="") && ($_POST[work_speed]!="") && ($_POST[feed_rate]!="") &&
($_POST[start_diam]!="") ){
$depth_cut=3.14*$_POST[start_diam]*($_POST[feed_rate]/($_POST[work_speed]*1000));
       $cal_depth_cut=1;
                         $_SESSION['notify'] .= "Depth of cut is caculated which is $depth_cut mm <br>";
```

```
}      elseif (($_POST[depth_cut]=="")&&($_POST[feed_rate]=="") && ($feed_rate!="")){
            //depth of cut could be cacluated using caculated feed rate
          $depth_cut=3.14*$_POST[start_diam]*$feed_rate/(1000*$_POST[work_speed]);
$cal_depth_cut=1;
$_SESSION['notify'] .= "Depth of cut is caculated which is $depth_cut mm. <br>";

}elseif (($_POST[depth_cut]=="") && ($_POST[feed_rate]=="")){
            //messsage to the user that the case could not be added!!
$_SESSION['error'] .="In order to save the new case,please enter at least work speed, feed rate and start
diameter or volumetric reomval rate and work speed to calculate the depth of cut <br>";
}
//***********check for roughness value****************

if (($_POST[roughness]=="")){
 //check for the surface roughness value if it is missing then it could be calculated using GP Model
                  if (($_POST[sel_matgroup]==4) && ($_POST[sel_material]==5) &&
($_POST[hardness]>57)){
                        //Ra could be cacluated using GP model for this material for chen data
                              //Shin data the roughness will be calculated using the GP general equation
                                          $he=$_POST[work_speed]*$_POST[depth_cut]/$_POST[w_speed];

                                          $x1=$_POST[d_depth]/4.79*(0.2963+$he-$_POST[d_depth]);
                                          $x2=0.72-$x1;
                                          $x3= pow($x2,0.7866);
                                          $x4=0.2963*$x3;
                                          $x5=pow($_POST[d_depth],$x4);
                                          $roughness=$x5-$x1;
              $cal_roughness=1;
                              $_SESSION['error'] .= "Ra can be caculated using avilable GP model ,roughness is
$roughness micro m. The value can be added after an experiment to avoid biased values    <br> ";
                              // header ("Location: add_new_grindingcase.php");
                                                //            exit;
                              } else if (($_POST[sel_matgroup]==1) && ($_POST[sel_material]==89) &&
($_POST[hardness]<51)){
                                          //the roughness could be calculated using Shin model
                                          $he=1000*$_POST[depth_cut]*$_POST[work_speed]/$_POST[w_speed];
                                          $roughness=1.64*pow($_POST[d_depth],-
0.021)*pow($_POST[d_lead],0.385)*pow($he,0.284);
            $cal_roughness=1;
            $_SESSION['error'] .= "The roughness could be calculated using Shin model, It is equal $roughness
micro m. The value can be added after an experiment to avoid biased values  <br>";

                              } else if (($_POST[sel_matgroup]==4) && ($_POST[sel_material]==1) &&
($_POST[hardness]>51)){
                                          //Ebberal data the roughness will be calculated using the GP general
equation
                                          $he=$_POST[work_speed]*$_POST[depth_cut]/$_POST[w_speed];

                                          $x1=$_POST[d_depth]/4.79*(0.2963+$he-$_POST[d_depth]);
                                          $x2=0.72-$x1;
                                          $x3=pow($x2,0.7866);
                                          $x4=0.2963*$x3;
                                          $x5=pow($_POST[d_depth],$x4);
                                          $roughness=$x5-$x1;
              $cal_roughness=1;
              $_SESSION['notify'] .= "Ebberal data the roughness will be calculated using the GP general
equation $roughness <br>";
                                          }else
 //if the material is not their then the system could calculate the roughness using the GP using MataLab and the
case will not be saved
                                          $_SESSION['error'] .= "Ra could be calculted using Genetic Programming
GP (MatLasb extension) <br>";
}
              }
                  $wheel=$_POST['sel_wheel'];
```

```
			$matg=$_POST['sel_matgroup'];
			$coolant=$_POST['sel_coolant'];
			$machine=$_POST['sel_machine'];
			$dresser=$_POST['sel_dresser'];
			$material=$_POST['sel_material'];
			$partid=$_POST['part_id'];
	$condtype=$_POST['cond'];
	$part_id=$_POST['sel_part'];
	$com=$_POST['com'];
  //		$_SESSION['notify'] .= "wheel $wheel coolant $coolant machine $machine material $material dresser
$dresser roughness $roughness";
//insert into cases table first
if (!$_SESSION[error]){
$matg_index="SELECT `material_Group_id`,`mg_index` FROM `material_group` WHERE
`material_Group_id`='$matg'";
$get_matg_index= mysql_query($matg_index) or die(mysql_error());
while ($recs=mysql_fetch_array($get_matg_index)){
		$id=$recs['material_Group_id'];
}
		$mgindex=$recs['mg_index'];
$add_case="INSERT INTO `sengaa2`.`cases` (`case_id`, `v_index`, `wheel_id`, `coolant_id`,
`dresser_id`,`material_id`, `material_group_id`, `part_id`, `machine_id`,`condition_type`,`comment`)
	VALUES
(NULL,'$mgindex','$wheel','$coolant','$dresser','$material','$matg','$part_id','$machine','$condtype','$com');";

mysql_query($add_case) or die (mysql_error());
//the new case id should be known before adding the rest data into the caserecord tabel
$caseid=mysql_insert_id();
$_SESSION[case_id]=$caseid;
echo "new case id is $_SESSION[case_id] ";
}
 //***********************find the index value for the hardness (primary index)*****************
    if ($_POST[hardness] <= 50) {
		$hardness_index="S";
		//$_SESSION['notify'] .= "hardness index $hardness_index <br>";

  }else if ($_POST[hardness] > 50 && ($_POST[hardness]) <= 58){
		 $hardness_index="M";
		 }else if ($_POST[hardness] > 58){
		 $hardness_index="H";

 //******************find the index value for the roughness (primary index*****************  if
(($roughness>=1.00) && ($roughness<= 1.60)){
		  $roughness_index=1;
    } else if (($roughness>=0.70)&&($roughness<= 0.99)){
		  $roughness_index=2;
    } else if (($roughness>=0.40)&&($roughness<= 0.69)){
		  $roughness_index=3;
    } else if (($roughness>=0.20)&&($roughness<= 0.39)){
		  $roughness_index=4;
    } else if (($roughness>=0.10)&&($roughness<= 0.19)){
		  $roughness_index=5 ;
    }
		// $_SESSION['notify'] .= "roughness index $roughness_index <br>";
//*****find out the material group index by running an query******************************
$matg_index="SELECT `material_Group_id`,`mg_index` FROM `material_group` WHERE
`material_Group_id`='$matg'";
$get_matg_index= mysql_query($matg_index) or die(mysql_error());
while ($recs=mysql_fetch_array($get_matg_index)){
		$id=$recs['material_Group_id'];
		$mgindex=$recs['mg_index'];
		}
//******find out the wheel index******************************
$wheel=$_POST['sel_wheel'];
```

```
$wheel_index="SELECT `wheel_id`,'wheel_name',`abrasive_type_index`,`bond_index`,`grit_size_index` FROM
`wheel` WHERE `wheel_id`='$wheel'";
$get_wheel_index= mysql_query($wheel_index) or die(mysql_error());
while ($recs=mysql_fetch_array($get_wheel_index)){
		$wab_index=$recs['abrasive_type_index'];
		$wb_index=$recs['bond_index'];
		$wgs_index=$recs['grit_size_index'];
		}
 //echo "wheel abrasive index $wab_index wheel bond index $wb_index wheel grits size $wgs_index";
if (!$_SESSION[error]){
$add_caserec="INSERT INTO `sengaa2`.`casesrecord` (`caserec_id`, `case_id`, `wheel_speed`, `width`,
`start_diameter`, `finish_diameter`,`depth_cut`, `size_tolerance`,
`roundness`, `wheel_diameter`, `work_speed`,`feed_rate`, `dressing_depth`, `dressing_lead`, `dressing_speed`,
`spark_out`, `roughness`,
 `hardness`, `material_id`, `equivelent_diameter`, `sim`, `matgroup_index`, `roughness_index`, `hardness_index`,
`abrasivetype_index`,
 `grits_index`, `bondtype_index`, `emp_id`, `Date`, `cal_roughness`, `cal_feed_rate`,`cal_depth_cut`)
VALUES('','$caseid','$w_speed','$width','$start_diam','$_POST[finish_diam]','$depth_cut','$size_tolerance',
 '$round','$w_diameter','$work_speed','$feed_rate','$d_depth','$d_lead','$d_speed','','$roughness',

'$hardness','$material','','','$mgindex','$roughness_index','$hardness_index','$wab_index','$wgs_index','$wb_index',
'','','$cal_roughness','$cal_feed_rate','$cal_depth_cut');";
  $_SESSION[width]=$width;
  $_SESSION[start_diam]=$start_diam;
  $_SESSION[finish_diam]=$_POST[finish_diam];
  $_SESSION[hardness]=$hardness;
  $_SESSION[material]=$material;
  $_SESSION[mgindex]=$mgindex;
  $_SESSION[roughness_index]=$roughness_index;
  $_SESSION[hardness_index]=$hardness_index;
  $_SESSION[wab_index]=$wab_index;
  $_SESSION[wgs_index]=$wgs_index;
  $_SESSION[wb_index]=$wb_index;
mysql_query($add_caserec) or die (mysql_error());
$_SESSION['notify'].="<p><b> A new case is added </b></p>";
$_SESSION['notify'].="<p><b> In order to add more recordes in this case <a
href=\"add_new_grindingrec.php\">click here</a> </b></p>";
$_SESSION['notify'].="<p><b> In order to add a new Grinding Case<a href=\"add_new_grindingcase.php\">click
here</a></b></p>";
$showForm = 0;
}
}
}
print_r($_POST) ;
echo "<br><Br>";
print_r($_SESSION);
$get_matgroup="SELECT `material_Group_id` as id, `name`, `mg_index` FROM `material_group` ORDER By
`name`";
$get_matgroup_res=mysql_query($get_matgroup)or die(mysql_error());
$get_material="SELECT `material_id`,`name`FROM `material` `name`
order by `name`";
$get_material_res=mysql_query($get_material) or die(mysql_error());

$get_wheelname="SELECT `wheel_id`,`wheel_name` FROM `wheel`";
$get_wheelname_res= mysql_query($get_wheelname) or die (mysql_error());

$get_machinename="SELECT `machine_id`,`name` FROM `machine`";
$get_machinename_res= mysql_query($get_machinename) or die(mysql_error());

$get_dressername="SELECT `dresser_id`,`name`FROM `dresser`";
$get_dressername_res=mysql_query($get_dressername) or die(mysql_error());


$get_coolant="SELECT `coolant_id`,`name` FROM `coolant`";
$get_coolant_res=mysql_query($get_coolant) or die(mysql_error());
```

```
$get_part="SELECT `part_id` FROM `part_details`";
$get_part_res= mysql_query($get_part) or die(mysql_error());
if (!isset($_SESSION['error']) && $showForm!=1 ){
 $display_block .="<p><font color='#006600'>". $_SESSION['notify']."</font> </p>";
}else{
$display_block="<h1>User Input</h1>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\" name='myform' >
<p><font color='#FF0000'> ". $_SESSION['error']." </font></p>";
$display_block .="
<P><strong>Part Number</strong>
<select name=\"sel_part\" value="".$_POST[sel_part]."">
<option value=\"\" >......Select One.... </option>";
while ($recs=mysql_fetch_array($get_part_res)){
		$id=$recs['part_id'];
		$display_name=$recs['part_id'];
     if ($id==$_POST[sel_part]) {
		$display_block .= "<option value=\"$id\" selected >";}
    else{
    $display_block .= "<option value=\"$id\">";}
		$display_block .= "$display_name</option>";
		}
$display_block.="
</select>
<P>(if the part number is not listed, add the part using add new part link on the left hand side)<br>
<P><strong>Grinding Condition</strong><br>

  <input type=\"radio\" name=\"cond\" value=\"R\">Roughing<br>
   <input type=\"radio\" name=\"cond\" value=\"F\">Finishing<br>



<h2>Material Details</h2>
<strong>   Material*</strong>
<select name=\"sel_material\" onChange=\"getInfo();\">
<option value=\"\" >...Select One... </option>";

while ($recs=mysql_fetch_array($get_material_res))  {
		$id=$recs['material_id'];
		$display_name=$recs['name'];

    if ($id==$_POST[sel_material]) {
		$display_block .= "<option value=\"$id\" selected >";}
    else{
    $display_block .= "<option value=\"$id\">";}
		$display_block .= "$display_name</option>";
}
$display_block.= "
</select>
<P>(if the Material is not listed, add a new material using add new material link on the left hand side)<br>
<P><strong> Material Group*</strong>
<select name=\"sel_matgroup\">
<option value=\"\" >..Select One...</option>";
while ($recs=mysql_fetch_array($get_matgroup_res))  {
		$id= $recs['id'];
		$display_name=$recs['name'];
    if ($id==$_POST[sel_matgroup]) {
		$display_block .= "<option value=\"$id\" selected >";}
    else{
    $display_block .= "<option value=\"$id\">";}
		$display_block .= "$display_name</option>";
}
$display_block.= "
</select>
```

```
<P>(if the Material group is not listed, add a new material group using add new material group link on the left
hand side)<br>
<P><strong>Hardness*</strong>
<input type=\"text\" name=\"hardness\" size=15 value="'.$_POST[hardness].'"><br>
<div id='mydiv'></div>
<P><strong>Start Diameter*</strong>
<input type=\"text\" name=\"start_diam\" size=15 value="'.$_POST[start_diam].'"> mm
<strong>Finish Diameter </strong>
<input type=\"text\" name=\"finish_diam\" size=15 value="'.$_POST[finish_diam].'"> mm<br><br>
<P><strong>Work Speed*</strong>
<input type=\"text\" name=\"work_speed\" size=15 value="'.$_POST[work_speed].'"> m/s<P><h2>Machine
Details</h2>
<P><strong>Machine Name</strong>
<select name=\"sel_machine\" value="'.$_POST[sel_machine].'">
<option value=\"\" >......Select One.... </option>";
while ($recs=mysql_fetch_array($get_machinename_res)){
		$id=$recs['machine_id'];
		$display_name=$recs['name'];

     if ($id==$_POST[sel_machine]) {
		$display_block .= "<option value=\"$id\" selected >";}
    else{
    $display_block .= "<option value=\"$id\">";}
		$display_block .= "$display_name</option>";

		}
$display_block.="
</select>
<P>(if the machine is not listed, add a machine using add new machine link on the left hand side)<br>
<P><h2>Wheel Details</h2>
<P><strong>Wheel Name*</strong>
<select name=\"sel_wheel\" value="'.$_POST[sel_wheel].'">
<option value=\"\" >......Select One.... </option>";
while ($recs=mysql_fetch_array($get_wheelname_res)){
		$id=$recs['wheel_id'];
		$display_name=$recs['wheel_name'];

    if ($id==$_POST[sel_wheel]) {
		$display_block .= "<option value=\"$id\" selected >";}
    else{
    $display_block .= "<option value=\"$id\">";}
		$display_block .= "$display_name</option>";
		}
$display_block.="
</select>
<P>(if the wheel is not listed, add a wheel using add new wheel link on the left hand side)<br>
<P><strong>Wheel Speed*</strong>
<input type=\"text\" name=\"w_speed\" size=15 value="'.$_POST[w_speed].'"> m/s
<strong>    Wheel Diameter*</strong>
<input type=\"text\" name=\"w_diameter\" size=15 value="'.$_POST[w_diameter].'"> mm <br><br>
<strong> Wheel Width</strong>
<input type=\"text\" name=\"width\" size=15 value="'.$_POST[width].'"> mm<br>
<P><h2>Coolant</h2><br>
<select name=\"sel_coolant\">
<option value='' >....Select One... </option>";
while ($recs=mysql_fetch_array($get_coolant_res)){
		$id=$recs['coolant_id'];
    $display_name= $recs['name'];
     if ($id==$_POST[sel_coolant]) {
		$display_block .= "<option value=\"$id\" selected >";}
    else{
    $display_block .= "<option value=\"$id\">";}
		$display_block .= "$display_name</option>";
}
$display_block.="
```

```
</select>
 <P>(if the Coolant is not listed, add a coolant using add new coolant link on the left hand side)<br>
<P><h2>Dresser</h2>
<P><strong>Dresser Name</strong>
<select name=\"sel_dresser\">
<option  value='' >....Select One... </option>";
while ($recs=mysql_fetch_array($get_dressername_res)){
		$id=$recs['dresser_id'];
		$display_name=$recs['name'];

    if ($id==$_POST[sel_dresser]) {
		$display_block .= "<option value=\"$id\" selected >";}
   else{
   $display_block .= "<option value=\"$id\">";}
		$display_block .= "$display_name</option>";
		}
$display_block.="
</select>
<P>(if the dresser is not listed, add a dresser using add new part link on the left hand isde)<br>
<P><strong>Dressing Depth* </strong>
<input type=\"text\" name='d_depth' size=20 value='".$_POST[d_depth]."'> mm <strong>Dressing Lead *
</strong><input type=\"text\" name=\"d_lead\" size=20 value='".$_POST[d_lead]."'> mm/rev<br>
<P><strong>Dressing Speed </strong>
<input type=\"text\" name=\"d_speed\" size=20 value='".$_POST[d_speed]."'> mm/s
<h2>Control Parameters</h2><br>
<strong>Depth of Cut</strong>
<input type=\"text\" name=\"depth_cut\" size=15 value='".$_POST[depth_cut]."'> mm <br>
<P><strong>Feed Rate</strong>
<input type=\"text\" name=\"feed_rate\" size=15 value='".$_POST[feed_rate]."'> mm/s
<P><strong>Roughness</strong>
<input type=\"text\" name=\"roughness\" size=15 value='".$_POST[roughness]."'> micro-m
<P><strong>Volumetric Removal Rate</strong>
<input type=\"text\" name=\"volum_remo\" size=15 value='".$_POST[volum_remo]."'>  mm3/mm.s
<P><strong>Size Tolerance </strong>
<input type=\"text\" name=\"size_tolerance\" size=15 value='".$_POST[size_tolerance]."'> micro-m
<P><strong> Roundness</strong>
<input type=\"text\" name='roundness' size=15 value='".$_POST[roundness]."'>
<P><strong> Comments</strong>
<input type=\"text\" name='com' size=50 value='".$_POST[com]."'>
<p><input type=\"submit\" name=\"submit\" value=\"Add\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
 }
 ?>
<HTML>
<HEAD>
<TITLE>User Input</TITLE>
		<script language="JavaScript" src="main.js"></script>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTML>
<?php
$page_title='Community of Practice';
include('headdercop.html');
?>
<p>Community of practice tool is considered to be an effective tool to share and exchange knowledge.</p>
<p>From the left hand side links, user can choose different options to share and exchange their knowledge</p>
<li>exchange questions and answers</li>
<li>share best practice and storyteller </li>
<li>call for events and conferences</li>
<li>pass web-links</li>
 <?php
```

```
 include('footer.html');
 ?>
<?php include('header.php');
$page_title='Discussion';
include ('headdercop.html');
?>
<?php
//run a query to get all of the categories
$catsql="SELECT * FROM categories";
$catresult=mysql_query($catsql);
echo "<table cellspacing=0>";
while ($catrow=mysql_fetch_assoc($catresult)) {
		echo "<tr class='head'><td colspan=2>";
		echo "<h2>".$catrow['name']."</h2></td>";
		//after wach category has been displayed check if the current category has got any forums
		echo "<tr>";
		$forumsql="SELECT * FROM forum WHERE cat_id = " . $catrow['cat_id'] .";";
		$forumresult=mysql_query($forumsql);
		$forumnumrows=mysql_num_rows($forumresult);
		if ($forumnumrows==0){
				echo "<tr><td>No forums!</td></tr>";
				}
		else{
				while ($forumrow=mysql_fetch_assoc($forumresult)){
						echo "<tr>";
						echo "<td>";
		//to add each one to the table
		echo "<strong><a
		href='viewforum.php?id="
		.$forumrow['forum_id']."'>" .
		$forumrow['name'] . "</a></strong>";
		echo "<br/><i>".$forumrow['description']. "</i>";
		echo "</td>";
		echo "</tr>";
						}
				}
		}
echo "</table>";
?>
<?php include('header.php');

include("copconfig.php");


if(isset($_GET['id'])==TRUE){
   if(is_numeric($_GET['id'])==FALSE){
		header("Location: " . $config_basedir);
   }
				$validforum=$_GET['id'];
		}
		else{
				header("Location: " .$config_basedir);
				}
		require("copheader.php");
		//echo "valid forum $validforum";
$forumsql= "SELECT * FROM `forum` WHERE `forum_id`=$validforum";
$forumresult=mysql_query($forumsql) or die(mysql_error());
$forumrow=mysql_fetch_assoc($forumresult);
echo "<h2>" .$forumrow['name']."</h2>";
//the next page is a breadcrumb trial
echo "<a href='index.php'>" . $config_forumname . "forums</a><br /><br />";
//add a link to allow user to add new topic to this forum
echo "[<a href='newtopic.php?id=" .$validforum ."'> New Topic </a>]";
echo "<br /><br />";
$topicsql= "SELECT MAX(message.date) AS maxdate, topic.topic_id AS topicid,topic.*,emp.*
```

```
          FROM message,topic,emp WHERE message.topic_id=topic.topic_id AND topic.emp_id=emp.emp_id
          AND topic.forum_id= ".$validforum ." GROUP BY
           message.topic_id
           ORDER BY maxdate DESC; ";
$topicresult=mysql_query($topicsql)or die(mysql_error());
$topicnumrows=mysql_num_rows($topicresult);
//if the $topicnumrows contains 0,there are no topics
if ($topicnumrows==0) {
              echo "<table width='300px'><tr><td> No Topics!</td></tr></table>";
}
//if there is topic. it will be presented in a table as well
else {
              echo "<table class='forum'>";
              echo "<tr>";
              echo "<th>Topic</th>";
              echo "<th>Author      </th>";
              echo "<th>        Date Posted  </th>";
              echo "</tr>";
              }
//a query will be run to count the number of messages for the topic in the current row
 while ($topicrow=mysql_fetch_assoc($topicresult)){
                $msgsql="SELECT message_id FROM message WHERE topic_id=". $topicrow['topic_id'];
                $msgresult=mysql_query($msgsql)or die(mysql_error());
                echo "<tr>";
                echo "<td>";
                echo "<strong><a href='viewmessages.php?id="
                . $topicrow['topic_id']. "'>"
                . $topicrow['subject']. "</a></strong></td>";
                  $msgnumrows=mysql_num_rows($msgresult);
                echo "<td>" .$msgnumrows . "</td>";
                 echo "<td>" .$topicrow['last_name']. $topicrow['first_name']. "</td>";
                 echo "<td>" .date("D jS F Y g.iA", strtotime ($topicrow['date']))
                 ."</td>";
                 echo "<tr>";
                  }
               echo "</table>";
               ?>
<?php include('header.php');
include("copconfig.php");
if(isset($_GET['id'])==TRUE){
               if(is_numeric($_GET['id'])==FALSE){
                 $error=1;
       }
       if($error== 1){
                  $config_basedir="http://localhost/touploadGrinding/index.php";
                  header("Location: " .$config_basedir);
         }
   else{
                            $validtopic=$_GET['id'];
                  }
}
               else {
                              $config_basedir="http://localhost/touploadGrinding/index.php";
               header("Location: " .$config_basedir);
                              }
               require("copheader.php");
              //add the name of the topic and the breadcrumb trial at the top of the page
              $topicsql="SELECT topic.subject,topic.forum_id,forum.name FROM topic,forum
                     WHERE topic.topic_id=forum.forum_id AND topic.topic_id= $validtopic ";
              $topicresult=mysql_query($topicsql) or die(mysql_error());
              $topicrow=mysql_fetch_assoc($topicresult);


              //add the subject of the topic
              echo "<h2>" . $topicrow['subject'] . "<h2>";
```

```
            //add the link to viewforum
            echo "<a href='copindex.php'>" .$config_forumname
            ."forum</a> -> <a href='viewforum.php?id="
            .$topicrow['forum_id']. "'>" . $topicrow['name']
            ."</a><br /><br/>";
$threadsql="SELECT message.*, emp.emp_id ,emp.last_name AS lname, emp.first_name AS fname FROM
message,emp WHERE
          message.emp_id= emp.emp_id and message.topic_id= $validtopic
          ORDER BY message.date";
$threadresult=mysql_query($threadsql) or die(mysql_error());
echo"<table>";
while ($threadrow=mysql_fetch_assoc($threadresult)){
            echo "<tr><td><strong>Posted by <i>"
            .$threadrow['fname'] . $threadrow['lname']."</i> on"
            .date ("D jS F Y g.iA", strtotime($threadrow['date']))
            . "- <i> ".$threadrow['subject']
            . "</i></strong></td></tr>";
            echo "<tr><td>" .$threadrow['body'] . " </td></tr>";
            echo "<tr></tr>";

            $_SESSION['topicID']=$validtopic;
            echo "<tr><td> [<a href='reply.php?id=" .$validtopic .
             "'>reply</a>]</td></tr>";
 }
             echo "</table>";
             ?>
<?php
session_start();
require("copconfig.php");
require("functions.php");
$conn=mysql_connect("localhost","root","password")
or die(mysql_error());
mysql_select_db("sengaa2",$conn) or die(mysql_error());
$topic=$_SESSION['topicID'];
if(isset($_GET['id'])== TRUE){
  if(is_numeric($_GET['id'])== FALSE){
               $error=1;
            }
  if($error==1){
      header("Location: " . $config_basedir);
  }
                        else{
                                    $validtopic=$_GET['id'];
                        }
  } else {
            header("Location: " . $config_basedir);
            }

if($_POST['submit']){

            if ((!$_POST['subject']) ||(!$_POST['body'])){
                        require("headdercop.html");
                        echo"Subject or Body nessage is Empty!Go back to fill them";
                        exit;
                        }
            require("headdercop.html");
            $conn=mysql_connect("localhost","root","password")
       or die(mysql_error());
          mysql_select_db("sengaa2",$conn) or die(mysql_error());

          $emp= $_SESSION['USERID'];
    $messagesql=" INSERT INTO message(date,emp_id,topic_id,subject,body,children)
             VALUES(NOW(),'$emp','$topic','$_POST[subject]','$_POST[body]','Yes')";
 mysql_query($messagesql) or die( mysql_error());
         echo "Reply is added";
```

```
          //header("Location:" . "viewmessages.php?id=" .$validtopc);
  }
 else{
                   require("headdercop.html");
  ?>
<form action="
<?php echo "reply.php";?>" method="post">
<table>
<tr>
<td>Subject</td>
<td><input type="text" name="subject" ></td>
</tr>
<tr>
  <td>Body</td>
  <td><textarea name="body" rows="10" cols="50"></textarea></td>
  </tr>
  <tr>
  <td></td>
  <td><input type="submit" name="submit" value="Post!"></td>
  </tr>
  </table>
  </form>
<?php
}
?>
<?php include('header.php');

require("copconfig.php");
require("functions.php");
$emp= $_SESSION['USERID'];
//run quick query to check if any fourm exist
$forchecksql="SELECT * FROM `forum`";
$forcheckresult=mysql_query($forchecksql) or die(mysql_error());
$forchecknumrows=mysql_num_rows($forcheckresult);
//if there is no forum exist the page will be redirected
if ($forchecknumrows==0){
              header("Location: " .$copconfig_basedir);
              }
//validate get variable
if(isset($_GET['id'])==TRUE){
  $validforum=$_GET['id'];
}
              else {
                            $vaildforum=0;
                            }

//check if the user is logged in, and if not, deny access
if(isset($_SESSION['USERNAME'])== FALSE){
  header("Location: " .$copconfig_basedir. "/login.php?ref=newpost&id="
    .$validforum);
 }

 if($_POST['submit']){
                  if($validforum==0){
                                //if the page was not passedvaildforum variable
                             $topicsql="INSERT INTO topic(date,emp_id,forum_id,subject)
                              VALUES(NOW(),'$emp','$_POST[forum]','$_POST[subject]')";


                  }
                  else{

         //if the page passed vaildforum variable

              $topicsql="INSERT INTO topic(date,emp_id,forum_id,subject)
```

```
                              VALUES(NOW(),'$emp','$validforum','$_POST[subject]')";
  }
//if checks to see if $validforum is equal
mysql_query($topicsql) or die(mysql_error());
$topicid=mysql_insert_id();
$messagesql=" INSERT INTO message(date,emp_id,topic_id,subject,body)
          VALUES(NOW(),'$emp','$topicid','$_POST[subject]','$_POST[body]')";

          mysql_query($messagesql) or die(mysql_error());

          include ('headdercop.html');
          echo "A new topic is added";
       }

    else{
                    require("copheader.php");
                    if($validforum !=0){
                                $namesql="SELECT name FROM forum ORDER BY name";
                                $nameresult=mysql_query($namesql);
                                $namerow=mysql_fetch_assoc($nameresult);
                                echo "<h2>Post new message to the ".$namerow['name']."
                                forum</h2>";
                    }
                    else{
                                echo "<h2>Post a new message</h2>";
                                }
              ?>
<form action="<?php echo "newtopic.php";?>" method="post">
 <table>
 <?php
  if ($validforum == 0){
              $forumsql="SELECT * FROM forum ORDER BY name";
              $forumresult=mysql_query($forumsql) or die(mysql_error());
?>
<tr>
<td>Forum</td>
  <td>
  <select name="forum">
  <?php
  while($forumrow=mysql_fetch_assoc($forumresult)){
              echo "<option value='" .$forumrow['forum_id']."'>".
              $forumrow['name']."</option>";
  }
  ?>
  </select>
  </td>
  </tr>
  <?php
}
?>
  <tr>
<td>Employee Name</td>
  <td>
  <select name="emp">
  <?php
   $empsql="SELECT `emp_id`,`first_name`,`last_name` FROM `emp` ORDER BY 'last_name'";
              $empresult=mysql_query($empsql) or die(mysql_error());
  while($emprow=mysql_fetch_assoc($empresult)){
              echo "<option value='" .$emprow['forum_id']."'>".
              $emprow['last_name']." ".
              $emprow['first_name']."</option>";
  }
  ?>
  </select>
  </td>
```

```
  </tr>

<tr>
<td>Subject</td>
<td><input type="text" name="subject" ></td>
</tr>
<tr>
  <td>Body</td>
  <td><textarea name="body" rows="10" cols="50"></textarea></td>
  </tr>
  <tr>
  <td></td>
  <td><input type="submit" name="submit" value="Post!"></td>
  </tr>
  </table>
  </form>
  <?php
}
?>
<?php
#script5.8- index.php
$page_title='knowledge warehouse(old part)';
include('headder.html');
?>
<?php
<form method=\"GET\" action=\"search1.php\" >
<P><strong> Enter Symptoms of the problem </strong>
  <input type= \"text\" name=\"q\">
  <input type= \"submit\" name=\"Submit\" value=\"Search\" >
</form>
?>
 <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTML>
<?php include('header.php');

#script search.php
$page_title='Searhch Problem';
include('headdercop.html');
?>

<?php

  // Get the search variable from URL

  $var = @$_GET['q'] ;
  $trimmed = trim($var); //trim whitespace from the stored variable

// rows to return
$limit=10;

// check for an empty string and display a message.
if ($trimmed == "")
  {
  echo "<p>Please enter a key word for the search</p>";
  exit;
  }
// check for a search parameter
if (!isset($var))
  {
```

```
  echo "<p>We dont seem to have a search parameter!</p>";
  exit;
  }
// Build SQL Query
$query = "SELECT *  FROM `topic` WHERE `subject` LIKE '%".$trimmed."%'";
 //order by 'date'"; // EDIT HERE and specify your table and field names for the SQL query
 $numresults=mysql_query($query);
 $numrows=mysql_num_rows($numresults);
// If we have no results, offer a google search as an alternative
if ($numrows == 0)
  {
  echo "<h2>Results</h2>";
  echo "<p>Sorry, your search: &quot;" . $trimmed . "&quot; returned zero results</p>";

// google
 echo "<p><a href=\"http://www.google.com/search?q="
  . $trimmed . "\" target=\"_blank\" title=\"Look up
  " . $trimmed . " on Google\">Click here</a> to try the
  search on google</p>";
  }

// next determine if s has been passed to script, if not use 0
  if (empty($s)) {
  $s=0;
  }
// get results
  $query .= " limit $s,$limit";
  $topicresult = mysql_query($query) or die("Couldn't execute query");
// display what the person searched for
echo "<h2>You searched for: " . $var . "&quot;</h2>";
// begin to show results set
echo "<h3> Results </h3>";
$count = 1 + $s ;

// now you can display the results returned
//ass
echo "<table class='forum'>";
		echo "<tr>";
		echo "<th>Topic</th>";
		//echo "<th>Author	</th>";
		echo "<th>	Date Posted  </th>";
		echo "</tr>";
//a query will be run to count the number of messages for the topic in the current row
 while ($topicrow=mysql_fetch_assoc($topicresult)){
		 $msgsql="SELECT message_id FROM message WHERE topic_id=". $topicrow['topic_id'];
		 $msgresult=mysql_query($msgsql)or die(mysql_error());
		 echo "<tr>";
		 echo "<td>";
		 echo "<strong><a href='viewmessages1.php?id="
		 . $topicrow['topic_id']. "'>"
		 . $topicrow['subject']. "</a></strong></td>";

		 $msgnumrows=mysql_num_rows($msgresult);

		  echo "<td>" .$msgnumrows . "</td>";
		  //echo "<td>" .$topicrow['last_name']. $topicrow['first_name']. "</td>";
		  echo "<td>" .date("D jS F Y g.iA", strtotime ($topicrow['date']))
		  ."</td>";
		  echo "<tr>";
   }
 echo "</table>";
?>
<?php include('header.php');
#script5.8- index.php
$page_title='Download File';
```

```
include('headdercop.html');

?>
<html>
<head>
<title>Download File From MySQL</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body>
<?php
//include 'library/config.php';
//include 'library/opendb.php';

$query = "SELECT upload_id, name,keywrd,title FROM upload";
$result = mysql_query($query) or die('Error, query failed');
if(mysql_num_rows($result) == 0)
{
echo "Database is empty <br>";
}
else
{
while(list($upload_id, $name,$keywrd,$title) = mysql_fetch_array($result))
{

echo "<strong>$title </strong> (Keywords: $keywrd )<a href='downloading.php?id=".$upload_id."'>".
urlencode($name)."</a> <br><br>";
}
}
//include 'library/closedb.php';
?>
</body>
</html>

<?php
if(isset($_GET['id']))
{
// if id is set then get the file with the id from database
//include 'library/config.php';
//include 'library/opendb.php';
$id    = $_GET['id'];
$sql = "SELECT name, type, size, content " .
        "FROM upload WHERE upload_id = '$id'";
//$sql = "SELECT bin_data, filetype, filename, filesize FROM tbl_Files WHERE id_files=$id_files"; $result =
@mysql_query($sql, $conn);
  $data = @mysql_result($result, 0, "content");
  $name = @mysql_result($result, 0, "name");
  $size = @mysql_result($result, 0, "size");
  $type = @mysql_result($result, 0, "type");
  header("Content-type: $type");
  header("Content-length: $size");
  header("Content-Disposition: attachment; filename=$name");
  header("Content-Description: PHP Generated Data");
  echo $data;
}
?>
HTML>
<head>
  <meta http-equiv="content-type" content="text/html; charset=is-8859-1"/>
  <title><?php echo $page_title; ?>
  </title>
  <style type="text/css" media="screen">
```

```
{background-color:#ffffff;}

  .content {
		background-color: #f5f5f5;
		padding-top: 10px; padding-right: 10px; padding-bottom: 10px; padding-left:10px;
		marging-top:10px; margin-right:10px; margin-bottom:10px;margin-left:10px;

		}
   a.navlink:link{ font-size:16px ;color:#003366;
   text-decoration: none;
		}
  a.navlink:visited { font-size:16px; color:#003366;
  text-decoration: none;
		}
    a.navlink:hover { font-size:16px ;color:#ccccc;
   text-decoration: none;
		}
  td{
		font-family: verdana, Arial,
		Helvetica, sans-serif; font-size:13px;
		vertical-align:top;
		}
 .title{
		font-size:28px;font-weight:normal;color:#ffffff;
		margin-top:5px; margin-bottom:5px;
		paddong-top:5px; padding-bottom:5px; padding-left:20px;
		}
 </style>
 </head>
 <body>
 <table width="90%" border="0" cellspacing="10"
 cellpadding="0" align="center">
<tr>
 <td colspan="2" bgcolor='#003366'>
 <p class="title"> Knowledge WareHouse for Grinding  Technology </p></td></tr>
 <tr>
 <td valign="top" nowrap="nowrap">
<b><a href="index.php" class="navlink">Home</a><br/><br/>
 <a href="copindex.php" class="navlink">Discussion Forum</a><br/><br/>
 <a href="newtopic.php" class="navlink">Post New Topic</a><br/><br/>
 <a href="cop_searchform.html" class="navlink">Search CoP Topics </a><br/><br/>
   <a href="form_uploadfile.php" class="navlink">UpLoad File</a><br/><br/>
  <a href="download_file.php" class="navlink">DownLoad File</a><br/><br/>
   <a href="doc_searchform.html" class="navlink">Search for File</a><br/><br/>
 </td>
 <td valign="top" class="content">
 <!--Script 12.1- headder.html -->
<h1>Search for Documents</h1><br/>
<br/>
<br/>
<h2> Enter the Key word for search</h2><br/>
<br/>
<form name="form" action="search_doc.php" method="get">
  <input type="text" name="q">
  <input type="submit" name="Submit" value="Search" />
</form>
</BODY>
</HTML>
<?php  include('header.php');
#script5.8- index.php
$page_title='Grinding Problems and Remedies';
include('headder_pro.html');
?>

<?php
```

```
//this code will enable users to search for problem solutions and browse solutions for different problems
//run a query to get all of the categories
$classsql="SELECT * FROM `prob_class`";
$classresult=mysql_query($classsql);
echo "<table cellspacing=0>";
while ($classrow=mysql_fetch_assoc($classresult)) {
        echo "<tr class='head'><td colspan=2>";
        echo "<strong>".$classrow['classification']."</strong></td>";
        //after wach category has been displayed check if the current category has got any forums
        echo "<tr>";
        $forumsql="SELECT * FROM `problem_solution` where class_id= " . $classrow['class_id'] .";";
        $forumresult=mysql_query($forumsql);
        $forumnumrows=mysql_num_rows($forumresult);
        if ($forumnumrows==0){
                echo "<tr><td>No forums!</td></tr>";
                }
        else{
                while ($forumrow=mysql_fetch_assoc($forumresult)){
                        echo "<tr>";
                        echo "<td>";
        echo "<strong><a
        href='viewsolution.php?id="
        .$forumrow['class_id']."'>" .
        $forumrow['sym'] . "</a></strong>";
        echo "</td>";
        echo "</tr>";
                        }
                }
        }
echo "</table>";
?>
<?php include('header.php');
//include("headder.html");
 include('headder_pro.html');

if(isset($_GET['id'])==TRUE){
                $validprob=$_GET['id'];
                $classsql="SELECT * FROM `prob_class` where class_id= '$validprob'";
   $classresult=mysql_query($classsql);
   while ($classrow=mysql_fetch_assoc($classresult)){
     echo "<strong>".$classrow['classification']."</strong>";
   }
   //to create a table
        echo "<table cellpadding='10'>";
        echo"<tr>";
        echo"<td></td>";
        echo"<td><strong>Symptoms</strong></td>";
        echo"<td><strong>Causes</strong></td>";
        echo"<td><strong>Remedy</strong></td>";
        echo"<td></td>";
        echo"</tr>";

        $solusql="SELECT * FROM `problem_solution` WHERE `class_id`= '$validprob' ";
        $soluresult=mysql_query($solusql);
        $solurows=mysql_num_rows($soluresult);
        if ($solurows==0){
                echo "<tr><td>Nothing Recorded!</td></tr>";
                }
        else{
        while ($solurows=mysql_fetch_assoc($soluresult)){
                        echo "<tr>";
                        echo "<td>";
        echo "<td>" .$solurows['sym']. "</td>" ;
        echo "<td>" .$solurows['cause']."</td>" ;
        echo "<td>" .$solurows['remedy'] . "</td>";
```

```
                echo "</tr>";

                                          }
}
}
 echo "</table>";
 ?>
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=is-8859-1"/>
  <title><?php echo $page_title; ?>
  </title>
        <style type="text/css" media="screen">
  body {background-color:#ffffff;}

  .content {
                background-color: #f5f5f5;
                padding-top: 10px; padding-right: 10px; padding-bottom: 10px; padding-left:10px;
                marging-top:10px; margin-right:10px; margin-bottom:10px;margin-left:10px;

                }
    a.navlink:link{ font-size:16px ;color:#003366;
    text-decoration: none;
                }
   a.navlink:visited { font-size:16px; color:#003366;
   text-decoration: none;
                }
     a.navlink:hover { font-size:16px ;color:#ccccc;
    text-decoration: none;
                }
   td{
              font-family: verdana, Arial,
              Helvetica, sans-serif; font-size:13px;
              vertical-align:top;

              }
  .title{
              font-size:28px;font-weight:normal;color:#ffffff;
              margin-top:5px; margin-bottom:5px;
              paddong-top:5px; padding-bottom:5px; padding-left:20px;
              }
 </style>
 </head>
 <body>
 <table width="90%" border="0" cellspacing="10"
 cellpadding="0" align="center">
 <tr>
 <td colspan="2" bgcolor='#003366'>
 <p class="title"> Knowledge WareHouse for Grinding  Technology <a href='logout.php' style="font-size:
12px;color: #FF0000;" >Log out</a> </p></td></tr>
 <tr>
 <td valign="top" nowrap="nowrap">
 <b><a href="index2.php" class="navlink">Home</a><br/><br/>

  <a href="search_prob.html" class="navlink">Search for solutions</a><br/><br/>

 <a href="problem_solution.php" class="navlink">Grinding Problems & Remedies</a><br/><br/>



</td>
<td valign="top" class="content">
 <!--Script 12.1- headder.html -->
 <h1>Search for Casues and Solutions</h1><br/>
<br/>
```

```
<br/>
<h2> Enter the symptoms</h2><br/>
<br/>
<form name="form" action="search1.php" method="get">
  <input type="text" name="q">
  <input type="submit" name="Submit" value="Search" />
</form>
<?php include('header.php');

#script search.php
$page_title='Searhch Problem';
include('headder_pro.html');
?>



<?php

  // Get the search variable from URL

  $var = @$_GET['q'] ;
  $trimmed = trim($var); //trim whitespace from the stored variable

// rows to return
$limit=10;

// check for an empty string and display a message.
if ($trimmed == "")
  {
  echo "<p>Please enter a search...</p>";
  exit;
  }

// check for a search parameter
if (!isset($var))
  {
  echo "<p>We dont seem to have a search parameter!</p>";
  exit;
  }

// Build SQL Query
$query = "SELECT * FROM `problem_solution` WHERE `sym` LIKE '%".$trimmed."%'";
 //order by 'date'"; // EDIT HERE and specify your table and field names for the SQL query

 $numresults=mysql_query($query);
 $numrows=mysql_num_rows($numresults);

// If we have no results, offer a google search as an alternative
if ($numrows == 0)
  {
  echo "<h4>Results</h4>";
  echo "<p>Sorry, your search: &quot;" . $trimmed . "&quot; returned zero results</p>";
// google
 echo "<p><a href=\"http://www.google.com/search?q="
  . $trimmed . "\" target=\"_blank\" title=\"Look up
  " . $trimmed . " on Google\">Click here</a> to try the
  search on google</p>";
  }
// next determine if s has been passed to script, if not use 0
  if (empty($s)) {
  $s=0;
  }
// get results
  $query .= " limit $s,$limit";
  $result = mysql_query($query) or die("Couldn't execute query");
```

```
// display what the person searched for
echo "<p>You searched for: " . $var . "&quot;</p>";
// begin to show results set
echo "<p> Results </p>";
$count = 1 + $s ;

// now you can display the results returned
  while ($row= mysql_fetch_array($result)) {
  $title = $row["sym"];
  $casues=$row["cause"];
  $remedy=$row["remedy"];
  echo "$count.) <strong>Result:</strong> <br/>";
  echo" <strong> Symptoms are</strong>  $title <br/><br/>" ;
  echo "<strong>The causes could be</strong> $casues <br/><br/>";
   echo "<strong>The remedy is </strong>  $remedy <br/><br/>" ;
  $count++ ;
  }
$currPage = (($s/$limit) + 1);
//break before paging
  echo "<br />";
 // next we need to do the links to other results
  if ($s>=1) { // bypass PREV link if s is 0
  $prevs=($s-$limit);
  print " <a href=\"$PHP_SELF?s=$prevs&q=$var\">&lt;&lt;
  Prev 10</a>  ";
  }
// calculate number of pages needing links
  $pages=intval($numrows/$limit);
// $pages now contains int of pages needed unless there is a remainder from division
  if ($numrows%$limit) {
  // has remainder so add one page
  $pages++;
  }
// check to see if last page
  if (!((($s+$limit)/$limit)==$pages) && $pages!=1) {
  // not last page so give NEXT link
  $news=$s+$limit;
  echo " <a href=\"$PHP_SELF?s=$news&q=$var\">Next 10 &gt;&gt;</a>";
  }
$a = $s + ($limit) ;
  if ($a > $numrows) { $a = $numrows ; }
  $b = $s + 1 ;
  echo "<p>Showing results $b to $a of $numrows</p>";
  ?>
<?php include('header.php');
#script5.8- index.php
$page_title='Add New Wheel';
include('headder_manage_grinding.html');
?>
<?php
//this code will allow user to input the case varaible
if ($_POST[op] != "view") {
$display_block="<h1>Add New Coolant</h1>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\">
<P><strong>Coolant Name  </strong>
<input type=\"text\" name=\"coolant_name\" size=20><br>
<P><strong>Description</strong>
<input type=\"text\" name=\"desc\" size=30><br>
<P><strong>Ratio</strong>
<input type=\"text\" name=\"ratio\" size=30><br>

<P><strong>Country</strong>
<input type=\"text\" name=\"country\" size=30><br>
<p><input type=\"submit\" name=\"submit\" value=\"Add\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
```

```
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
} else if ($_POST[op]=="view")          {
//if the user click add button
if ($_POST['submit']){
$mistakes=array();
            //check for the required fields that are coolant name and description
      $coolant=trim($_POST['coolant_name']);
            if (!$_POST['coolant_name']) {

                        $mistakes[]= 'Please eneter coolant name';
                        }     else{
      $coolant = mysql_real_escape_string(stripslashes($coolant));
      }
 $des= trim($_POST[desc]);
      $des=mysql_real_escape_string(stripslashes($des));
      $country=trim($_POST[country]);
      $country=mysql_real_escape_string(stripslashes($country));
      $ratio=trim($_POST[ratio]);
      $ratio=mysql_real_escape_string(stripslashes($ratio));
  if (sizeof($mistakes) > 0)
{
echo "<ul>";
foreach ($mistakes as $errors)
{
echo "<li>$errors</li>";
}
echo "</ul>";
echo '<br />';
echo "Press back button to CORRECT the entry";

}   else {
//create the query to insert the values into coolant table
$add_coolant="INSERT INTO `sengaa2`.`coolant` (`coolant_id`, `name`, `description`, `ratio`, `country`)
           VALUES (NULL, '$coolant', '$_desc', '$ratio', '$country');";
mysql_query($add_coolant) or die (mysql_error());
$msg="<p> A new coolant is added</p>";
echo $msg; }
}
}
 ?>
 <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block;" ?>
</BODY>
</HTML>
<?php include('header.php');
#script5.8- index.php
$page_title='Add New Wheel';
include('headder_manage_grinding.html');

?>
<?php
//this code will allow user to input the case varaible



if ($_POST[op] != "view") {

$display_block="<h1>Add New Dresser</h1>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\">
<P><strong>Dresser Name   </strong>
```

```
<input type=\"text\" name=\"dresser_name\" size=20><br>
<P><strong>Type</strong>
<input type=\"text\" name=\"type\" size=30><br>
<P><strong>Size</strong>
<input type=\"text\" name=\"size\" size=30><br>
<P><strong>Country</strong>
<input type=\"text\" name=\"country\" size=30><br>
<p><input type=\"submit\" name=\"submit\" value=\"Add\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
}
else if ($_POST[op]=="view")        {
//if the user click add button
if ($_POST['submit']){
$mistakes=array();
        //check for the required fields that are dresser name and type
  $dresser_name=trim($_POST['dresser_name']);
        if (!$_POST['dresser_name']){

                $mistakes[]='Please eneter at least the dresser name!';
                }    else{
      $dresser_name=mysql_real_escape_string(stripslashes($dresser_name));
    }
   $type= trim($_POST[type]);
    $type=mysql_real_escape_string(stripslashes($type));
    $size=trim($_POST[size]);
    $size=mysql_real_escape_string(stripslashes($size));
    $country=trim($_POST[country]);
    $country=mysql_real_escape_string(stripslashes($country));
if (sizeof($mistakes) > 0)
{
echo "<ul>";
foreach ($mistakes as $errors)
{
echo "<li>$errors</li>";
}
echo "</ul>";
echo '<br />';
echo "Press back button to CORRECT the entry";

}  else {
//create the query to insert the values into dresser table

          $add_coolant="INSERT INTO `sengaa2`.`dresser` (`dresser_id`, `name`, `type`, `size`, `country`)
       VALUES (NULL, '$dresser_name', '$type', '$size', '$country');";
mysql_query($add_coolant) or die (mysql_error());
$msg="<p><h2> A new Dresser is added</h2></p>";
echo $msg;
        }
  }
}
 ?>
 <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTML>
<?php include('header.php');
#script5.8- index.php
$page_title='Add New Wheel';
include('headder_manage_grinding.html');
```

```
?>
<?php
//this code will allow user to input the case varaible
if ($_POST[op] != "view") {
$display_block="<h1>Add New Machine</h1>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\">
<P><strong>Machine Name  </strong>
<input type=\"text\" name=\"machine_name\" size=20><br>
<P><strong>Description</strong>
<input type=\"text\" name=\"desc\" size=30><br>
<P><strong>Manufactured By</strong>
<input type=\"text\" name=\"country\" size=30><br>
<p><input type=\"submit\" name=\"submit\" value=\"Add\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
}
 else if ($_POST[op]=="view")             {
//if the user click add button
if ($_POST['submit']){
$mistakes=array();
        $machine=trim($_POST['machine_name']);
             //check for the required fields that are machine name and description
             if ((!$_POST['machine_name'])){
        $mistakes[]= 'Please eneter the machine name';
        } else {
        $machine = mysql_real_escape_string(stripslashes($machine));

                        //header ("add_new_machine.php");
                        //exit;
                        }
if (sizeof($mistakes) > 0)
{
echo "<ul>";
foreach ($mistakes as $errors)
{
echo "<li>$errors</li>";
}
echo "</ul>";
echo '<br />';
echo "Press back button to CORRECT the entry";
} else{
//create the query to insert the values into machine table
$add_coolant="INSERT INTO `sengaa2`.`machine` (`machine_id`, `name`, `description`, `manufactued_by`)
            VALUES (NULL, '$machine', '$_POST[desc]', '$_POST[country]');";
mysql_query($add_coolant) or die (mysql_error());
$msg="<p><h2> A new machine is added</h2></p>";
echo $msg;
                        }
}
}
 ?>
  <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTML>
<?php include('header.php');
#script5.8- index.php
$page_title='Add New Wheel';
include('headder_manage_grinding.html');
```

```
?>


<?php
//this code will allow user to input the case varaible



if ($_POST[op] != "view") {

$get_matgroup="SELECT `material_group_id` as id, `name`, `mg_index` FROM `material_group` order by
`name`";
$get_matgroup_res=mysql_query($get_matgroup)or die(mysql_error());

$display_block="<h1>Add New Material</h1>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\">


<P><strong>Material Name  </strong>
<input type=\"text\" name=\"material_name\" size=20><br>

<P><strong> Material Group</strong>
<select name=\"sel_matgroup\">
<option value=\"\" >..Select One...</option>";

while ($recs=mysql_fetch_array($get_matgroup_res))  {
        $id= $recs['id'];
        $display_name=$recs['name'];

        $display_block .= "<option value=\"$id\">
        $display_name</option>";
}
$display_block.= "
</select>
<P><strong>Note</strong>
<input type=\"text\" name=\"desc\" size=30><br>
<p><input type=\"submit\" name=\"submit\" value=\"Add\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
} else if ($_POST[op]=="view")         {
 //if the user click add button
if ($_POST['submit']){
$mistakes=array();
        //check for the required fields that are machine name and description
  $material_name=trim($_POST['material_name'] );
        if ((!$_POST['material_name'])){
          $mistakes[]='Please Eneter Materila name';
                }    else{
     $material_name=mysql_real_escape_string(stripslashes($material_name));
    }
//***check the material group************
if ($_POST[sel_matgroup]==""){
$mistakes[]='Please select Material group!';
}
$desc=trim($_POST[desc]);
$desc=mysql_real_escape_string(stripslashes($desc));
//create the query to insert the values into machine table
if (sizeof($mistakes) > 0)
{
echo "<ul>";
foreach ($mistakes as $errors)
{
echo "<li>$errors</li>";
```

```
}
echo "</ul>";
echo '<br />';
echo "Press back button to CORRECT the entry";
} else {
$add_coolant="INSERT INTO `sengaa2`.`material` (`material_id`, `name`, `material_group_id`,`note`)
            VALUES (NULL, '$material_name', '$_POST[sel_matgroup]','$desc')";
mysql_query($add_coolant) or die (mysql_error());
$msg="<p><h2> A new material is added</h2></p>";
echo $msg;                          }
    }
}
 ?>
   <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTML>
<?php include('header.php');
#script5.8- index.php
$page_title='Add New Wheel';
include('headder_manage_grinding.html');
?>
<?php
//this code will allow user to input the case varaible
if ($_POST[op] != "view") {
$display_block="<h1>Add New Material Group</h1>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\">
<P><strong>Material Group Name  </strong>
<input type=\"text\" name=\"mg_name\" size=20><br>
<P><strong>Material Group Index</strong>
<input type=\"text\" name=\"mg_index\" size=15><br>
<P><strong>Density</strong>
<input type=\"text\" name=\"density\" size=15>
<strong>   Tensile</strong>
<input type=\"text\" name=\"tensile\" size=15>
<P><strong>   Material Tempreture</strong>
<input type=\"text\" name=\"temp\" size=15>
<strong>          Specific Energy</strong>
<input type=\"text\" name=\"specific_energy\" size=30><br>
<P><strong>    Thermal Conductivity</strong>
<input type=\"text\" name=\"ther_condu\" size=15>
<strong>Details</strong>
<input type=\"text\" name=\"details\" size=15>
<p><input type=\"submit\" name=\"submit\" value=\"Add\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
}
else if ($_POST[op]=="view")              {
//if the user click add button
if ($_POST['submit']){
$mistakes=array();
             //check for the required fields that are machine name and description
   $mg_name=trim($_POST['mg_name']);
             if ((!$_POST['mg_name'])){
             $mistakes[]= 'please enetr material group name';
                         }  else {
             $mg_name=mysql_real_escape_string(stripslashes($mg_name));
       }
          //******check material index
   $mg_index=trim($_POST[mg_index]);
```

```
    If (empty($mg_index) || (!(ctype_alpha($mg_index))))
{
$mistakes[] = 'Material group index is either empty or Enter only ALPHABET characters.';
}
else
{
//accept material index entry and sanitize it
$mg_index = mysql_real_escape_string(stripslashes($mg_index));
}
//**********check for dentisy*******
   $density=trim($_POST[density]);
     If (!(empty($density))){
       If (!(is_numeric($density))){
          $mistakes[] ='For Density filed: Enter only NUMERIC characters<br>';
        } else{
               //accept density entry and sanitize it
       $density=mysql_real_escape_string(stripslashes($density));
       }
   }
  $tensile=trim($_POST[tensile]);
    $tensile=mysql_real_escape_string(stripslashes($tensile));
    $temp=trim($_POST[temp]);
    $temp=mysql_real_escape_string(stripslashes($temp));
    $specific_energy=trim($_POST[specific_energy]);
    $specific_energy=mysql_real_escape_string(stripslashes($specific_energy));
    $ther_condu=trim($_POST[ther_condu]);
    $ther_condu=mysql_real_escape_string(stripslashes($ther_condu));
    $details=trim($_POST[details]);
    $details=mysql_real_escape_string(stripslashes($details));
//create the query to insert the values into machine table
      if (sizeof($mistakes) > 0)
{
echo "<ul>";
foreach ($mistakes as $errors)
{
echo "<li>$errors</li>";
}
echo "</ul>";
echo '<br />';
echo "Press back button to CORRECT the entry";
} else {
$add_matgroup="INSERT INTO `sengaa2`.`material_group`
(`material_Group_id`,`name`,`mg_index`,`density`,`tensile`,`melting_temperature`,`specific_energy`,`thermal_con
ductivety`,`Details` )
            VALUES (NULL, '$mg_name', '$mg_index',
'$density','$tensile','$temp','$specific_energy','$ther_condu','$details');";
mysql_query($add_matgroup) or die (mysql_error());
$msg="<p><h2> A new material group is added</h2></p>";
echo $msg;
              }
     }
}
?>
 <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTML>
<?php include('header.php');
#script5.8- index.php
$page_title='Add New Wheel';
include('headder_manage_grinding.html');
```

```
?>
<?php
//this code will allow user to input the case varaible
if ($_POST[op] != "view") {
$display_block="<h1>Add New Part</h1>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\">
<P><strong>Compostion</strong>
<input type=\"text\" name=\"compo\" size=20><br>
<P><strong>Chemical Properties</strong>
<input type=\"text\" name=\"chem_prop\" size=30><br>
<P><strong>Physical Properties</strong>
<input type=\"text\" name=\"phys_prop\" size=30><br>
<P><strong>Hardness </strong>
<input type=\"text\" name=\"hardness\" size=30><br>
<P><strong> Dimenssion</strong>
<input type=\"text\" name=\"dimn\" size=30><br>
<p><input type=\"submit\" name=\"submit\" value=\"Add\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
} else if ($_POST[op]=="view")          {
 //if the user click add button
if ($_POST['submit']){
$mistakes=array();
            //check for the required fields that are machine name and description
//****check for mateial compostioon
   $compo=trim($_POST['compo']);
            if ((!$_POST['compo'])){
                        $mistakes[]= 'Please enter the material compostion!';
                        }  else{
       $compo=mysql_real_escape_string(stripslashes($compo));
       }
            $hardness=trim($_POST[hardness]);
        If (!(empty($hardness))){
       If (!(ctype_digit($hardness))){
          $mistakes[] ='For Hardness: Enter only Integer NUMERIC characters<br>';
        } else{
              //accept hardness entry and sanitize it
       $hardness = mysql_real_escape_string(stripslashes($hardness));
       }
   }
     $chem_prop=trim($_POST[chem_prop]);
     $chem_prop=mysql_real_escape_string(stripslashes($chem_prop));
     $phys_prop=trim($_POST[phys_prop]);
     $phys_prop=mysql_real_escape_string(stripslashes($phys_prop));
     $dimn=trim($_POST[dimn]);
     $dimn=mysql_real_escape_string(stripslashes($dimn));
 if (sizeof($mistakes) > 0)
{
echo "<ul>";
foreach ($mistakes as $errors)
{
echo "<li>$errors</li>";
}
echo "</ul>";
echo '<br />';
echo "Press back button to CORRECT the entry";
}  else {
//create the query to insert the values into machine table



$add_coolant="INSERT INTO `sengaa2`.`part_details` (`part_id`, `composition`, `chemichal_properties`,
`physical_properties`, `hardness`, `dimenssion`)
```

```
VALUES (NULL, '$compo', '$chem_prop','phys_prop','$hardness','$dimn');";mysql_query($add_coolant) or die (mysql_error());
$msg="<p><h2> A new part is added is added</h2></p>";
echo $msg;
                }
  }
}
?>
  <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTML>
<?php include('header.php');
#script5.8- index.php
$page_title='Add New Wheel';
include('headder_manage_grinding.html');
?>
<?php
//this code will allow user to input the case varaible
if ($_POST[op] != "view") {
$display_block="<h1>Wheell Details</h1>
<form method=\"POST\" action=\"$_SERVER[PHP_SELF]\">
<P><strong>Wheel Name </strong>
<input type=\"text\" name=\"wheel_name\" size=20><br>
<P><strong>Wheel Grade</strong>
<input type=\"text\" name=\"wheel_grade\" size=30><br>
<P><strong>Bond Type</strong>
<input type=\"text\" name=\"bond_type\" size=30><br>
<P><strong>Bond Type Index</strong><br>
<input type=\"radio\" name=\"bond_type_ind\" value=\"V\" checked>V
<input type=\"radio\" name=\"bond_type_ind\" value=\"E\">E
<input type=\"radio\" name=\"bond_type_ind\" value=\"B\">B
<input type=\"radio\" name=\"bond_type_ind\" value=\"R\">R
<input type=\"radio\" name=\"bond_type_ind\" value=\"M\">M<br>

<P><strong>Abrasive Type</strong>
<input type=\"text\" name=\"bond_type\" size=30><br>
<P><strong>Abrasive Type Index</strong><br>
<input type=\"radio\" name=\"abrasive_type_ind\" value=\"A\" checked>A
<input type=\"radio\" name=\"abrasive_type_ind\" value=\"C\" >C
<input type=\"radio\" name=\"abrasive_type_ind\" value=\"B\" >B<br>
<P><strong>Grits Size</strong>
<input type=\"text\" name=\"grits\" size=30><br>
<P><strong>Grits Size Index</strong>
<input type=\"text\" name=\"grits_ind\" size=20><br>
<P><strong>Supplied By</strong>
<input type=\"text\" name=\"supply\" size=30><br>
<P><strong>Manufactured By</strong>
<input type=\"text\" name=\"manufact\" size=30><br>
<p><input type=\"submit\" name=\"submit\" value=\"OK\"></p>
<p><input type=\"submit\" name=\"Cancel\" value=\"Cancel\"></p>
<input type=\"hidden\" name=\"op\" value=\"view\">
</FORM>";
}
 else if ($_POST[op]=="view")           {
//if the user click add button
if ($_POST['submit']){
            //check for the required fields that are machine name and description
            if ((!$_POST['wheel_name'])){
            header ("add_new_wheel.php");
                        exit;
```

```
                    }
$add_wheel="INSERT INTO `sengaa2`.`wheel` (`wheel_id`, `wheel_name`, `grade`, `abrasive_type`,
`abrasive_type_index`, `bond`, `bond_index`, `grit_size`, `grit_size_index`, `supplied_by`, `manufactued_by` )
        VALUES (NULL, '$_POST[wheel_name]', '$_POST[wheel_grade]',
'$_POST[abrasive_type]','$_POST[abrasive_type_ind]','$_POST[bond_type]','$_POST[bond_type_ind]','$_POST[
grits]','$_POST[grits_ind]','$_POST[supply]','$_POST[manufact]');";
mysql_query($add_wheel) or die (mysql_error());
$msg="<p><h2> A new wheel is added</h2></p>";
echo $msg;                              }
}
?>
 <HTML>
<HEAD>
<TITLE>User Input</TITLE>
</HEAD>
<BODY>
<?php echo "$display_block" ?>
</BODY>
</HTML>
```