CLOUD COMPUTING BASED MACHINING OPTIMIZATION

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

by

M.Chandrasekaran

(Roll No. 07610308)

Department of Mechanical Engineering

Indian Institute of Technology Guwahati

Guwahati–781039

INDIA

January 2014

Department of Mechanical Engineering,
Indian Institute of Technology Guwahati,
Guwahati-781 039,
INDIA

Date 1.1.14

CERTIFICATE

It is certified that the work contained in the thesis entitled "**Cloud Computing Based Machining Optimization**" submitted by **Mr. M.Chandrasekaran** to the Indian Institute of Technology Guwahati for the award of the degree of Doctor of Philosophy has been carried out under our supervision in the Department of Mechanical Engineering, Indian Institute of Technology Guwahati. This work has not been submitted elsewhere for the award of any other degree or diploma.

Dr. U. S. Dixit
Professor
Department of Mechanical Engineering,
Indian Institute of Technology Guwahati,
Guwahati – 781 039, Assam
INDIA

Dr. M. Muralidhar
Professor
Department of Mechanical Engineering,
NERIST, Nirjuli – 791 109,
Arunachalpradesh
INDIA

Dedicated to.....

My parents

My teachers

&

My family

Acknowledgement

I would like to express my most sincere gratitude to my supervisors Professor U.S. Dixit and Prof. M. Muralidhar for their gratitude, support and encouragement. I am grateful to Prof. Dixit as without his support and advice, it would not have been possible for me to continue my research work. He has helped me at various difficult phases of my work with his valuable guidance and moral support. I would like to thank my doctoral committee members, Prof. S. K. Kakoty, Prof. U.K.Saha and Prof. A.K.Das for their valuable suggestions and encouragement during the period of my research work. I am also grateful to the past and present heads of the mechanical engineering department, Prof. U.S. Dixit, Prof. D. Chakraborty and Prof. P. Mahanta for extending various facilities during the tenure of my doctoral programme. I am thankful to Prof. Kalyan Kumar Ex. Director, NERIST for granting me study leave for my PhD course work. I am also thankful to present Prof. Dipankar Pal, Director NERIST and Dr. Asis Giri, HOD (ME), NERIST for their continuous support.

I also gratefully acknowledge the financial assistance provided by the All India Council for Technical Education through project AICTE: 8023/RID/BOIII/NCP (21) 2007-2008, project identification number at IIT Guwahati being ME/P/USD/4 for the research work. I also acknowledge the financial assistance provided from TEQIP fund of NERIST.

I am very much thankful to Ms. Astha Kohli, summer trainee at IITG for her invaluable help during my PhD work. I am also thankful to Dr. D. K. Sarma, Workshop Superintendent, IITG for his help and support. I wish to express my sincere thanks to Mr. Khaklary for the assistance during the experimental work. I am thankful to my faculty colleagues/friends Dr. C. M. Krishna, Mr. Sachindra Mahto, Late Shri S.Ray for their invaluable help. I also express my sincere thanks to Prof. S.Ravi, Physis department of IIT Guwhati for his help. My sincere thanks also go to Dr. Rathnakar Das, Mr. Vinod Yadav, Mr. Seikh Mustafa Kamal, Mr. Rosang

Pongen, Mr. A.Singh, Mr.Sai Prasad Ojha, Mr. S.Tamang and Mr. Shiv Kumar Sharma for their help in different aspects during my thesis work.

I shall always be grateful to my parents (Late Shri.M. Muthumari and Mrs. M.Gruvammal), my uncle Shri R. Subbaraman for their encouragement and wishes. Last but not least, I thank the members of my family for their constant, encouragement and patience for the completion of this thesis work. Special thanks go to my wife Mrs. Kamala, my beloved daughter, C.Nivetha and son C.Vignesh.

Finally, I bow my head to the God Almighty in deepest gratitude and seek blessings.

January 2014 M.Chandrasekaran

IIT Guwahati

Abstract

In the era of information technology the application of computer and web technology help to connect different manufacturers and customers globally on wire. Internet or web based manufacturing and very recently the concept of cloud computing and cloud manufacturing has started gaining popularity. The cloud computing is a concept of providing computing services and data on demand. The word 'cloud' (also phrased as 'the cloud') stands for 'the internet', but the phrase 'cloud computing' means much more than 'internet based computing'. Several researchers have applied web technologies for various applications such as mechanical design, production scheduling, process planning, *etc*. The application of web technology in the area of machining optimization and information sharing is the need of hour. In this work the feasibility of using cloud computing for the optimization of machining processes is explored.

The optimization of machining process aims to obtain optimum cutting parameters for economic manufacturing of the product. Researchers have used a number of traditional and non-traditional optimization techniques for obtaining optimum parameters in single and multipass machining processes. The results of conventional offline optimization have many limitations and need to be fine tuned in the shop floor depending on the rigidity of machine tool, deviation in the properties of material and a number of random causes. In this thesis, a heuristic based online optimization methodology suitable for a cloud based system has been developed. It optimizes the process during production of jobs. The optimization strategy does not require any *a priori* machining models *viz*., tool life and surface roughness prediction models. The methodology is demonstrated for optimization of common machining processes using virtual machines that replace actual machining experiments by simulated results.

A web enabled cloud computing based optimization and database (CCBOD) system is developed, in which the cloud supports machining database and computing facilities. The developed CCBOD system employ an online optimization methodology that suggests cutting conditions for shop floor machine tool operator, accepts machining feedback, and optimizes the machining process online. The clients/users that connect to CCBOD *via* web browsers can avail the facility. The real machining data provided by different clients are stored in database server. The MySQL relational database system is used to store information in web server. The main server keeps the repository of data and carries out optimization. The system works on private cloud and is available at IIT Guwahati intranet.

For web implementation of the cloud based optimization system hypertext markup language (HTML), cascading style sheets (CSS), and hypertext pre-processor (PHP) scripts have been written for clients to interact with the system. The web enabled feature help for accessing and sharing information among the users. The system also provides facility for retrieval of optimum cutting parameters using PHP scripts and MySQL queries. The result can be viewed at client's screen. MySQL database system used at server side stores all information in "machining database" at main server and is managed by the administrator. The user can get the information by fetching data from the database and user also sends the feedback of the same through HTML and PHP pages. For easy shop floor implementation of proposed cloud based online optimization, an interactive based menu driven system is developed. A number of user interactive screens that accepts machining informations from the user and displays the cutting conditions or instructions to the user.

A fuzzy set approach to optimization also has been proposed in this thesis. It consists of finding the optima with the help of IF-THEN rules. As the procedure is based on linguistic subdivision of the domain, a range is obtained in which the objective function value does not change drastically. The methodology is validated with standard test functions and also applied to machining optimization problem for optimizing the process parameters both by offline and online procedure. Machining experiments were carried out to demonstrate online optimization of machining process using fuzzy set based optimization procedure. It is applied for optimization

of finish turning process of cold rolled steel bars using TiN coated carbide tool. The optimization objective is to minimize the cost satisfying desired surface roughness. The optimization strategy is expected to be efficient for the problem with a large number of design variables and suitable for a cloud based system. It also acknowledges the presence of statistical variation in the process.

An artificial intelligence (AI) based online learning methodology to optimize cutting parameters also has been proposed. The online learning procedure is demonstrated through numerical experiments. The results show that prediction system keeps becoming more and more intelligent by machining a variety of different jobs. The data sets obtained through simulation results are stored and effectively used for training a neural network of artificial intelligence (AI) system.

The application of these techniques in a web based cloud computing environment will be highly beneficial to manufacturing industries in the area of machining and optimization. It is envisaged that the information technology and soft computing based optimization techniques will dominate manufacturing process optimization in future.

Contents

List of Figures

List of Tables

Nomenclature

c_1, c_2	Learning factors
C	Machining cost per piece
C'	Tool life constant
C_F	Cost for finish passes
C_L	Cost for loading and unloading
C_R	Cost for roughing passes
C_T	Cost per piece function for toll life T
C_0	Operating cost
C_1	Constant for tool life in equation (5.10)
d	Depth of cut
d_F	Depth of cut for finishing passes
d_{min}, d_{max}	Minimum and maximum depth of cut
d_R	Depth of cut for roughing passes
D	Diameter of the work piece
D_0, D_f	Initial and final diameter of work piece
D_T	Distance between the two datasets
D_{Tmax}	Maximum possible value of D_T
f	Feed or feed rate
f_{min}, f_{max}	Minimum and maximum feed
f'	Feed/tooth

F'	Cutting force
f_F	Feed rate for finishing passes
f_{opt}	Optimum feed rate
f_R	Feed rate for roughing passes
g, w, n	Exponents for different tool and work material combination
L	Machining length
m	Number of passes
n	Number of data in equation (4.1)
N	Spindle speed
p', q', r'	Tool life exponents
Q	Contact proportion of cutting edges with work piece per evolution
r	Random number between 0 and 1
R	Tool nose radius
R_{a}	Surface roughness
R_a^t	Target surface roughness
R_a^*	Maximum allowable surface roughness
$\hat{R}_a$	Actual (achieved) surface roughness
t_{c}	Tool change time
t_{s}	Tool reset time/pass
T	Tool life
T_F	Tool life for finishing passes
T_L	Loading and unloading time
T_{p}	Machining time per component

T_R	Tool life for roughing passes
T_{tF}	Machining time for finish pass
T_{tR}	Total machining time for roughing passes
T_{ts}	Total tool resetting time
v	Cutting speed
v_F	Cutting speed for finishing passes
v_{opt}	Optimum cutting speed
v_R	Cutting speed for roughing passes
$v_{\mathrm{min}}, v_{\mathrm{max}}$	Minimum and maximum cutting speed
w	Inertia weight
z	Number of tooth

Greek letters

β	Contraction coefficient
DE	Difference in energy level
ε	Termination parameter
η	Expansion co-efficient
μ	Population mean
σ	Standard deviation

Acronyms

ACC	Adaptive control constraints
ACO	Adaptive control optimization
AE	Acoustic emission
AI	Artificial intelligence
AMD	Advanced micro devices
ANFIS	Adaptive neuro-fuzzy inference system
ANN	Artificial neural network
ART	Adaptive resonance theory
BGA	Binary coded genetic algorithm
BHN	Brinell Hardness
BP	Back propagation
CBN	Carbon boron nitride
CCBOD	Cloud computing based optimization and database
CLA	Center line average
CNC	Computer numeric control
CSS	Cascading style sheets
DaaS	Data storage as a service, design as a service
DFD	Data flow diagram
EaaS	Experimentation as a service
FAN	Fuzzy adaptive neural networks

FBFNs	Fuzzy basis function neural networks
FBNN	Fuzzy basis function network
FNN	Fuzzy neural network
GA	Genetic algorithm
GP	Geometric programming
GSA	Genetic simulated annealing
GUI	Graphical user interface
HRC	Rockwell Hardness
HSS	High speed steel
HTML	Hypertext mark up language
IaaS	Infrastructure as a service
IP	Internet protocol
IT	Information technology
MB	Mega byte
MBPNND	Modified back propagation neural network with delta bar delta
MFGaaS	Manufacturing as a service
MLP	Multi-layer perceptron
MRR	Material removal rate
MySQL	My structured query language
NC	Numeric control
NN	Neural network
PaaS	Platform as a service
PHP	Hypertext pre-processor
PSO	Particle swarm optimization

QFD	Quality function deployment
RAM	Random access memory
RBF	Radial basis function
RBFN	Radial basis function neural network
RGA	Real coded genetic algorithm
RMS	Root mean square
SA	Simulated annealing
SaaS	Software as a service
SOM	Self-organizing map
SQP	Sequential quadratic programming
SVMs	Support vector machining
TCMS	Tool condition monitoring system
URL	Uniform resource location

Chapter 1

Introduction

This is the era of information technology. Computers, information technology, and web applications have been found in every domain of society and manufacturing industries are no exception. The application of computer and web technology connects different manufacturers and customers globally on wire. The information distributed at different locations can be accessed and shared by users anywhere in the world using web tools such as web browsers (Yang and Xue, 2003). Cloud computing is a recent concept that has evolved from the web based technologies, which provides on-demand computing services and data. Several researchers have applied web technologies for various applications such as mechanical design, production scheduling, process planning, *etc*. The application of web technology in the area of machining optimization and information sharing is the need of hour.

Machining optimization uses a number of traditional and non-traditional optimization techniques for obtaining optimum process parameters. Due to high complexity involved in machining, the development of predictive modelling is a challenging task, but it is necessary for proper optimization of the process. Most of the researchers employ empirical relations or experimental based predictive modelling for evaluation the parameters *viz*., surface roughness, tool life, cutting force, *etc*. in the optimization process. The optimum process parameters obtained by this offline technique deviate from real optimal parameters.

In view of its importance, the present thesis proposes online machining optimization based on cloud computing. A generalised heuristic based online optimization technique, the implementation of online optimization among different clients using cloud computing feature, an application of fuzzy set theory as an optimization tool, the experimental validation of fuzzy set based online machining optimization, an artificial intelligence (AI) based online learning procedure for obtaining optimum machining parameters are reported in this thesis. The keywords of this thesis are online machining optimization, cloud computing, fuzzy set based optimization, and online learning. These are briefly introduced in Sections 1.1–1.4. Organization of the thesis is presented in the last section of this chapter.

1.1 Machining Optimization

Process modelling and parameter optimization are two important issues in manufacturing. The modelling of machining process establishes the relationship between process parameters (*i.e.*, cutting velocity, tool/cutter feed rate, depth of cut, *etc.*) and output responses such as job surface roughness, cutting force, cutting temperature, tool flank wear, *etc*. The process optimization obtains optimum cutting parameters aiming economy in machining process. Optimization of machining process is performed in two ways— (i) offline optimization procedure and (ii) online optimization procedure.

For optimizing the process the problem is formulated for the desired objective(s) subjected to set of constraints. The general form of a single-objective optimization problem is

$$\begin{aligned} &\textit{Minimize } f(\boldsymbol{x}), \\ &\quad \textit{subject to a list of constraints.} \end{aligned} \tag{1.1}$$

where $\boldsymbol{x}$ is a vector containing the process variables. The maximization of a function is equivalent to the minimization of the negative of the function. The problem is solved for obtaining optimum process parameters and it is implemented during machining process. For modelling and optimization of machining processes, many researchers have opted traditional optimization techniques considering practical constraints. Due to complexity and uncertainty of the machining processes, soft computing techniques are being preferred to physics-based models for predicting the performance of the machining processes and optimizing them (Aggarwal and Singh, 2005; Mukherjee and Ray, 2006). Major soft computing tools applied for this purpose are neural networks, fuzzy sets, genetic algorithms, simulated annealing, ant colony optimization and particle swarm optimization. All these techniques solve the problems offline employing conventional predictive models and optimum parameters are obtained. Since the performance of machining process changes with time and other factors, the results of offline optimization need to be fine tuned online during implementation.

In online optimization the process parameters are optimized during production of jobs by suitably selecting the cutting conditions. Optimization with online monitoring of machining process have been developed that uses adaptive control strategy such as adaptive control optimization (ACO) and adaptive control constraints (ACC). It provides the capability to adjust operating parameters online, based on measurement of appropriate process characteristics form real machining environment. The procedure requires costly process-sensors and acceptable interfacing circuitry in its application. Therefore, there is a need for economic online optimization method which can cope up with real time machining environment and can be directly implemented in the shop floor.

The research papers found in the area of machining optimization focus mainly on effectiveness of the techniques in term of 'accuracy of the result' and 'computational time' required for optimization. There is hardly any paper that details out its application in shop floor. The present work focuses on development of online optimization of machining process which can be directly implemented during production of components and *a priori* machining models are not required.

1.2 Cloud Computing Based Optimization

The process optimization and information sharing using web technologies is a new concept. Cloud manufacturing aims to share manufacturing resources to make 'manufacturing as service' environment. Similarly, the cloud computing provides computing services and sharing of information among the different users/manufacturer. The National Institute of Standards and Technology defines cloud computing as "a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management efforts or service provider interactions." The cloud infrastructure can be owned by a private organization, a group of organizations, or the public. In this work the feasibility of using cloud computing for the optimization of machining processes is explored. Figure 1.1 shows the concept of cloud computing system for optimization of machining process and information sharing.

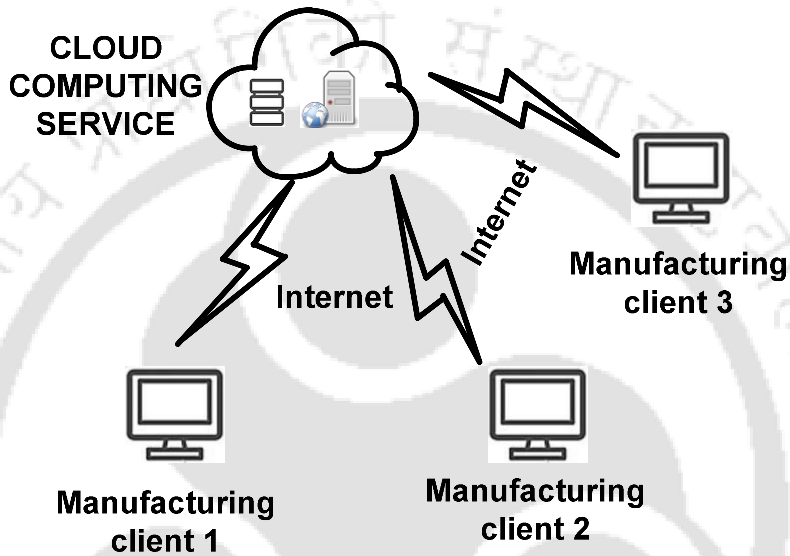


Figure 1.1. Concept of cloud computing based optimization

The system consists of the web server, a machining optimization scheme for the use of various manufacturing clients, exchange of machining informations *via* internet, cloud computing services, *etc*. The main server receives machining information from various clients, keeps the repository of data and carries out optimization. The optimized process parameters are provided to various clients on demand. The clients can use the optimized data, fine-tune them if necessary and send the feedback to the server. The informations are processed using probability based approach at the server side.

1.3 Fuzzy Set Based Optimization

In 1965, Lotfi Zadeh put forward the idea of fuzzy sets (Zadeh, 1965). The fuzzy set theory is a tool for "computing with language". The technique is found quite effective in converting subjective knowledge/opinion of the skilled operator into a mathematical framework (Zadeh, 1973). Many linguistic terms can be converted into a fuzzy set. For example, the "low feed" can be represented by a fuzzy set in which the feed values more than an upper threshold value can be assigned a membership grade 1 and those lower than a lower threshold value can be assigned a membership grade 0. Between lower and upper threshold, the feed values can have a gradual variation of membership grades from 0 to 1. Once the linguistic variables have been converted into fuzzy sets, set theoretic operations on them can be carried out.

In the literature, predominantly the fuzzy set theory has been used in three ways. In the first type of applications, fuzzy set theoretic operations help to arrive at certain decision. For example, if we know the membership grades of certain solutions in two conflicting objectives, then the optimum solution can be chosen as the one providing the highest membership grade in the intersection set of two objectives. The second type of application makes use of fuzzy arithmetic, which deals with fuzzy numbers. A fuzzy number is a generalization of an interval number, in which various intervals may have different membership grades. For example, the most likely estimate of friction may be assigned a membership grade 1 and the lower and upper estimate may be assigned a membership grade 0.5. With these three points, a triangle can be constructed to represent friction as a fuzzy number. Fuzzy arithmetic computations are useful when the input parameter values, for example friction and material properties, are not known precisely. In the third type of application, fuzzy logic is used for making inferences based on the input values. The fuzzy set-based prediction system takes input data and carries out "fuzzification". In the fuzzification process, the input data undergo some translation in the form of linguistic terms such as "low feed", "average cutting speed", "high depth of cut", "very high cutting force", *etc*. The translated data are sent to an inference engine, which applies a set of predefined IF-THEN rules. The output of inference system in linguistic form will go through defuzzification process, which converts it to numerical data.

Large number of publication is found in modelling and control applications of fuzzy sets. In the area of optimization, researchers used the fuzzy set theory that can deal with the situations of fuzziness in the objective function and/or constraints. For solution of the problem, traditional linear/non-linear programming methods or soft computing based methods like genetic algorithm are employed. However, the researchers have not used the fuzzy set theory as an optimization tool. In the present work, the potential of fuzzy set theory is demonstrated as a general optimization tool and applied for machining optimization problems.

1.4 Online Learning

The modern automated manufacturing relies on computer based learning schemes that are able to learn the complex machining process. In machining process, the prediction of surface roughness, cutting force, and tool life is a challenging task, but is necessary for proper optimization of the process. For obtaining the knowledge of tool life as a function of cutting parameters a number of tool life tests is to be conducted. It is not economical to conduct tool life test for different work piece–tool combinations. In recent years researchers have used soft computing based modelling techniques *viz*., neural network, fuzzy logic and combination of both (neuro-fuzzy) for developing predictive models. Soft computing techniques are tolerant of imprecision, uncertainty and partial truth. However, for model development, it requires a number of data for training. The experimentation consumes a lot of time and may not be sufficient for online optimization.

The present thesis focuses the development of online learning methodology for optimizing the machining process. The proposed scheme consists of three modules *viz*., AI module, virtual simulation module (that will be replaced by actual machine in the real shop floor) and an optimization module. The developed online learning algorithm keeps on becoming more and more intelligent by machining variety of different jobs. The data sets obtained through online machining are stored and effectively used for training neural network (NN) based artificial intelligence (AI) system. A well-trained network could predict better guess and takes less time for prediction. The cutting parameters required for producing maximum allowable surface finish at minimum cost is obtained. With the help of numerical experiments, the effectiveness of proposed methodology is demonstrated.

1.5 Highlights of the Thesis

The primary objective of the present thesis work is to develop an online optimization methodology based on cloud computing. A heuristic based online optimization methodology has been developed first. The methodology is applied for optimization of finish turning and end milling process. The optimization model does not use *a priori* machining models. Experimental validation of the methodology has been performed. A number of numerical problems is solved using semi virtual machine designed for the purpose. For easy implementation of the proposed optimization procedure in the shop floor, a user friendly interactive machining system has been developed. It helps to suggest cutting conditions for shop floor machine tool operator, accepts feedback, and optimizes the machining process during the production of components. Source code is written in MATLAB® with graphical user interface (GUI) facilitate the user to interact with the system easier.

A web system called 'Cloud Computing Based Optimization and Database (CCBOD)' has been developed, in which the cloud supports machining database and computing facilities for optimization of machining process. For web implementation of the system HTML, CSS, and PHP scripts have been written for clients/users to interact with the system. The web-enabled feature helps the users to get connected with the system *via* web browsers.

A new optimization tool based on fuzzy set theory has been attempted in this thesis. The methodology obtains the optimum using IF-THEN rule base. The methodology is effective for the problem where an approximate solution is required. In this thesis the procedure is applied for machining optimization. It employs a fuzzy set based optimization methodology applied for optimization of machining processes. The methodology has been validated using standard test function as well as optimization problems in offline and online modes. It also acknowledges the presence of statistical variation in the process. The online optimization methodology has been demonstrated experimentally by conducting machining experiments for optimizing finish turning process. A methodology for optimization with online learning has been also proposed. The online learning procedure is demonstrated through simulations.

1.6 Organization of the Thesis

The thesis is organized as follows:

- The first chapter introduces the problem and the highlights of the thesis are described.
- Chapter 2 presents a review of literature on modelling and optimization of conventional machining processes, web based machining and cloud computing. The challenging issues, scope and detailed objectives of the thesis have been described.
- Chapter 3 first discusses the basic frame work of the cloud computing system for optimization of machining process. Next, the different components of proposed cloud computing based optimization and database (CCBOD) and its development are presented. The system information flow, obtaining optimum process parameters from cloud, online fine tuning and sending back the information is discussed. Finally the web implementation of the system with hypertext mark up language (HTML), cascading style sheets (CSS), and hypertext pre-processor (PHP) scripts to enable the clients/users interact with the system has been discussed.
- In Chapter 4, the heuristic based online optimization for cloud computing system is discussed. The design of virtual machine to simulate the experiments is presented. The proposed online optimization is applied for optimization of both finish and multipass machining processes. Simplex search is used as optimization tool.
- In order to make the optimization procedure more efficient a new optimization tool based on fuzzy set theory is presented in Chapter 5. The methodology is demonstrated with standard test functions. It is also applied to offline and online machining optimization problems. The experimental demonstration of online optimization using fuzzy set based optimization is presented. Advantages and limitations of the online optimization method are highlighted.

- Chapter 6 presents the development of menu driven-interactive online optimization. The method for obtaining optimum parameters based on probabilistic approach has been discussed.
- In Chapter 7, the application of soft computing for online learning in a cloud computing based machining optimization system is presented. The various components of the system *viz*., AI module, virtual simulation module, and optimization module are described with online learning procedure. A number of illustrative examples are presented to demonstrate the methodology.
- Conclusions and scope for future work are presented in Chapter 8, followed by references and appendices.

Chapter 2

Literature Review and Detailed Objectives

2.1 Introduction

Machining, forming, casting and joining are the four basic manufacturing processes. Machining is the most popular among all these processes. It is the process of shaping and sizing the raw material by removing the excess material in the form of chips. Optimization of machining process is widely researched in the field of manufacturing. Selection of right combination of cutting parameters (i.e., cutting speed, feed and depth of cut) to satisfy one or more economic objectives such as maximization of production rate, minimization of cost, maximization of surface finish, *etc.* is important for metal cutting industries.

Machining optimization started around 60 years back with the pioneering work of Gilbert (1950). It has attracted the attention of a number of researchers in view of its significant contribution to the overall cost of the product (Merchant, 1998). From the era of conventional machine tools to the present era of CNC machine tools, the prediction of cutting behavior of processes and optimization of machining parameters have been hot areas of research. In a review of metal cutting analyses in 1956, Finnie (1956) pointed out—"Despite the large number of attempts, past and present, to analyze metal cutting, a basic relationship between the various variables is still lacking." This remark is valid till today, even after about a half century. Nevertheless, the efforts to model machining process are still going on, as the proper understanding of the machining process has a large bearing on the economics of machining. With the advent of capital intensive CNC machine tools, this need has strengthened.

A number of researchers have presented various traditional and non traditional methods for optimizing the process parameters in single and multi pass machining processes (Aggarwal and Singh, 2005; Mukherjee and Ray, 2006). The traditional methods of optimization such as graphical techniques, constrained optimization strategy, dynamic programming, branch and bound algorithm, *etc*. are found to be ineffective since they either result in local minima, or take long time to converge on a reasonable result. In recent past, the researchers have used soft computing techniques as they are being preferred to physics-based models. Major soft computing tools applied for this purpose are neural network (NN), fuzzy sets, genetic algorithm (GA), simulated annealing (SA), ant colony optimization (ACO) and particle swarm optimization (PSO). The application of these tools in modelling and optimization of machining processes is reviewed here.

Apart from machining optimization, the application of computer and information technology plays an important role in manufacturing industries. Starting from the late 1990s, web- based manufacturing is steadily gaining popularity. The computer system and information technology (IT) are commonly used for data processing, storage and information access through internet. With the help of World Wide Web, information distributed at different locations can be accessed and shared by users anywhere in the world using web tools such as web browsers (Yang and Xue, 2003). The web technology connects different manufacturer and customers globally on wire. Several researchers have applied web technologies for various applications such as mechanical design, quality modelling, production scheduling, process planning, *etc*. The application of web technologies in the area of machining optimization and information sharing has now brought the attention of researchers.

This chapter presents a review of available literature in the area of machining performance prediction and optimization of different machining process. The application of web technologies in manufacturing is also presented. Section 2.2 presents an overview of modern soft computing tools used in modelling and optimization of machining processes. The literature related to the application of soft computing tools for modelling and optimization of different machining processes *viz*., turning, milling, grinding and drilling is presented in Section 2.3. The review of literatures on web based machining and cloud computing in manufacturing is presented in Section 2.4. The challenging issues are discussed in Section 2.5. The detailed scope and objectives of the present work is presented in Section 2.6

2.2 An Overview of Soft Computing Techniques

Soft computing is an approach to computing which parallels the remarkable ability of the human mind to reason and learn in an environment of uncertainty and imprecision. In an attempt to find out reasonably useful solutions, soft computing methods (Dixit and Dixit, 2008; Deb and Dixit, 2008) acknowledge the presence of imprecision and uncertainty present in machining. Soft computing techniques such as fuzzy sets, NN, GA, SA, ACO and PSO have received a lot of attention of researchers due to their potentials to deal with highly non-linear, multi dimensional and ill-behaved complex engineering problems. A brief overview of various soft computing techniques is presented here.

2.2.1 Fuzzy Set Theory

The idea of fuzzy sets is first proposed by Lotfi Zadeh (Zadeh, 1965). It was briefly described in Section 1.3 of Chapter 1. There are a plethora of application of fuzzy sets in the area of modelling and some in the area of optimization. Fuzzy logic modelling is performed in four basic steps: fuzzification of the input variables, rule evaluation, aggregation of the rule outputs and finally defuzzification (Dixit and Dixit, 2008). Mamdani and Sugeno type fuzzy inference process are commonly used by the researchers. The Sugeno-type models are computationally more efficient compared to Mamdani-type models, but rules of Mamdani type models provide better physical feel. Number of research papers have found in the area of modelling of machining processes.

2.2.2 Neural Networks

Neural networks are systems that can acquire, store and utilize knowledge gained from experience. An artificial neural network (ANN) is capable of learning from an experimental data set to describe the non-linear and interaction effects with great success. It consists of an input layer used to present data to the network, output layer to produce ANN's response and one or more hidden layers in between. The input and output layers are exposed to the environment and hidden layers do not have any contact with the environment. ANNs are characterized by their topology, weight vectors and activation function that are used in hidden and output layers of the network. A neural network is trained with a number of data and tested with other set of data to arrive at an optimum topology and weights. Once trained, the neural networks can be used for prediction.

A multi layer perceptron (MLP) is a feed forward neural network with one or more hidden layers. A feed forward neural network has sequence of layers consisting of a number of neurons in each layer. The output of one layer becomes input to neurons in the succeeding layer. The radial basis function (RBF) neural network consists of three layers: an input layer, a single hidden layer with non-linear processing neurons and an output layer. During training process, the network adjusts its weights to minimize the errors the error between the predicted and desired outputs. Back propagation algorithm is most common algorithm for adjusting the weights. A brief background of neural networks is provided in (Dixit and Dixit, 2008).

2.2.3 Genetic Algorithm

Genetic algorithm (GA) mimics the process of natural evolution by incorporating the "survival of the fittest" philosophy (Goldberg, 1989). In GA, a point in search space is represented by binary or decimal numbers, known as string or chromosome. Each chromosome is assigned a fitness value that indicates how closely it satisfies the desired objective. A set of chromosomes is called population. A population is operated by three fundamental operations *viz*., reproduction (to replace the population with large number of good strings having high fitness values), crossover (for producing new chromosomes by combining the various pairs of chromosomes in the population) and mutation (for slight random modification of chromosomes). A sequence of these operations constitute one generation. The process repeats till the system converges to the required accuracy after many generations. The genetic algorithms have been found very powerful in finding out the global minima. Further, these algorithms do not require the derivatives of the objectives and constraints functions.

2.2.4 Simulated Annealing

Simulated annealing (SA) mimics the cooling process of metal during annealing to achieve the minimization of function values. The algorithm begins with an initial point, $\boldsymbol{x}_1$ and a large number corresponding to a high temperature T. A second point $\boldsymbol{x}_2$ is created near the first point using a Gaussian distribution with first point as a mean. The difference in the function values at these points is considered analogues to the difference in energy level (ΔE). For a minimization process, if the second point has smaller function value, then it replaces the first point; otherwise it replaces the first point with a probability $\exp(-\Delta E/T)$ (Deb, 1995). The algorithm is terminated when a sufficiently small temperature is obtained or no significant improvement in the function value is observed.

2.2.5 Ant Colony Optimization

The ACO algorithm is a kind of natural algorithm inspired by foraging behavior of real ants. Researchers are fascinated by seeing the ability of near-blind ants in establishing the shortest route from their nest to the food source and back. These ants secrete a substance, called pheromone and use its trails as medium of communicating information (Dorigo, 1996). The probability of the trail being followed by other ants is enhanced by further deposition of pheromone by other ants moving on that path. This co-operative behavior of ants inspired the new computational paradigm for optimizing real life systems, which is suited for solving large scale problems (Socha and Dorigo, 2008).

There are different variants of ant colony optimization algorithm. In essence, these algorithms carry out three operations: (1) ant based solution construction, (2) pheromone update and (3) daemon actions. In ant based solution construction, solutions representing artificial ants are constructed. Solutions are chosen probabilistically based on pheromone level. Thus, this operation forces the algorithm to search in the area of better solutions. The aim of pheromone update is to increase the pheromone values associated with good or promising solutions and decrease those that are associated with bad ones. Usually this is achieved by increasing the pheromone levels associated with chosen good solutions and by decreasing the pheromone values through pheromone evaporation, which basically reduces the pheromone level. Daemon actions are used to implement centralized actions which cannot be performed by a single ant. For example, the global information can be collected to decide whether it is useful or not to deposit additional pheromone.

Initially, the ant colony optimization was used for combinatorial problems. Nowadays, it is also being used for solving continuous optimization problems. The positive feedback accounts for rapid discovery of good solutions. The distributed computation prevents premature convergence of the problem.

2.2.6 Particle Swarm Optimization

Particle swarm optimization is a population based stochastic optimization technique developed by Kennedy and Eberhart in 1995 and is inspired by the social behavior of bird flocking or fish schooling (Kennedy and Eberhart, 1995). In PSO each solution in search space is analogous to a bird and generally called 'particle'. The system is initialized with population of random particles (called swarm) and search for the optimum continues by updating generations. The fitness value of each particle is evaluated by objective function to be optimized. Each particle remembers the coordinates of the best solution ($\boldsymbol{p}$) achieved so far. The coordinates of current global best ($\boldsymbol{p}_\mathbf{g}$) are also stored. The coordinates of the particle are updated according to the following relations:

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \ , \qquad (2.1)$$

and

$$v_{id}^{k+1} = \chi\left\{ w v_{id}^{k} + c_1 r_{id1}(p_{id}^{k} - x_{id}^{k}) + c_2 r_{id2}(p_{gd}^{k} - x_{gd}^{k}) \right\}, \qquad (2.2)$$

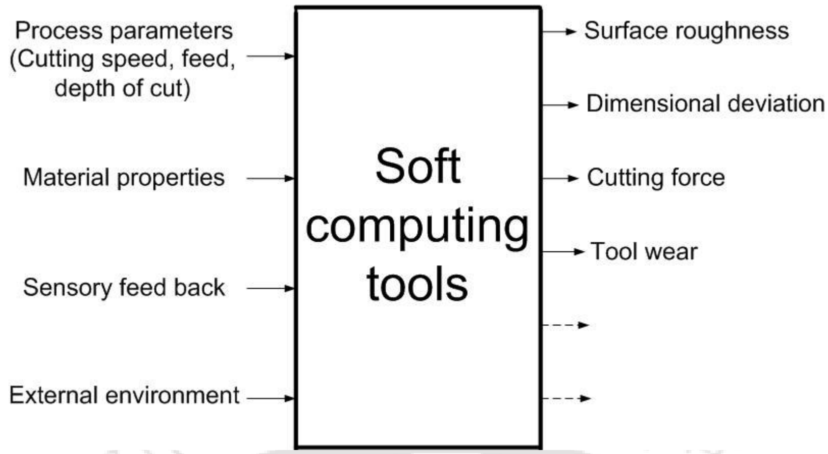
where $\boldsymbol{x}$ is particle coordinate, $\boldsymbol{v}$ is particle velocity, χ is a constriction factor, i denotes the particle or solution, g denotes global, d denotes dimension (varying from 1 to number of decision variables), k is the iteration number, w is the inertia weight, c_1 is a cognitive parameter, c_2 is a social parameter and r stands for random number between 0 and 1.

PSO is simple to implement and needs less parameters to adjust. It converges with less number of iterations. In many cases, it has been reported to be more efficient than GA. Unlike in GA and other evolutionary programming methods PSO does not employ selection operation.

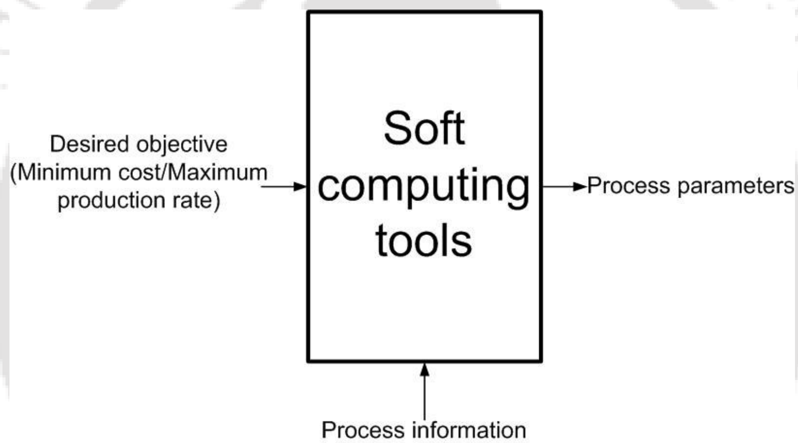
2.3 Soft Computing Based Modelling and Optimization of Machining Processes

Machining processes are complex, nonlinear, multivariate in nature. Modelling of machining process establishes the relationship between process parameters such as cutting speed, feed, depth of cut, *etc*. and performance measures such as surface roughness, dimensional deviation, cutting force, tool wear, cutting temperature, *etc*. The process optimization is for obtaining optimum cutting parameters aiming an economy of the machining process. The prediction of performance measure is a challenging task but it is necessary for proper optimization of the process. Soft computing techniques are more preferred to physics-based models for predicting the performance of the machining processes and for optimizing them (Dixit and Dixit, 2008; Deb and Dixit, 2008; Zarei *et al*., 2009; Wang *et al*., 2005; Zang *et al*., 2006). Figures 2.1 (a) and 2.1(b) schematically depict the application of soft computing techniques for these tasks.

In this thesis the review the literatures on major machining processes *viz*., turning, milling, grinding and drilling is presented so as to have complete picture in the area of machining performance prediction and optimization.



(a) For performance prediction



(b) For process optimization

Figure 2.1. Two applications of soft computing in machining

2.3.1 Turning Process

In cylindrical turning processes, a single point cutting tool moves along the axis of a rotating work piece. The peripheral speed of the work piece called cutting speed, movement of the tool along the axis of job for one revolution of job called feed and radial depth of cut of the tool are the process parameters. These parameters may be optimized for obtaining the minimum cost of machining and minimum production time. However, for optimization, performance of the process is to be predicted. The previous research work related to optimization of turning process as well as various performance measures such as surface finish, dimensional deviation, tool life, tool wear and cutting force has been reviewed in the following sub-sections.

2.3.1.1 Surface finish and dimensional deviation

Two main attributes of quality of turned job are surface finish and dimensional deviation. Surface finish is defined as the degree of smoothness of a part's surface after it has been manufactured. Surface finish is the result of the surface roughness, waviness and flaws remaining on the part. Dimensional deviation is defined as the radial difference between the set depth of cut and the obtained depth of cut. Researchers studied the effect of number of factors such as feed rate, cutting speed, depth of cut, work material characteristics, unstable built up edge, tool nose radius, tool angles, stability of material, tool and work piece set-up, use of cutting fluids, radial vibration, tool material *etc*. on surface finish. For modelling of machining processes, researchers used four main methods *viz.,* multiple-regression, mathematical modelling based on physics of process, fuzzy set theory and neural network. The review in this paper will focus on last two techniques.

An early work for the prediction of machining performance using neural networks is by Rangawala and Dornfeld (1989). The authors utilized a feed forward network model to predict the cutting performance in turning process. The network is trained using a number of patterns of input variables *viz*., cutting speed, feed rate and depth of cut and the output variables *viz.,* cutting force, power, temperature and surface finish. Rangawala and Dornfeld (1989) and many later researchers used MLP neural network for machining performance prediction. An MLP neural network consists of input layer, output layer and one or more hidden layers. Each layer consists of a number of neurons. Neurons in the hidden and output layers get input from the neurons of the preceding layers. Neurons in the input and hidden layers emit the output to be received by the neurons of the succeeding layer.

Azouzi and Guillot (1997) proposed neural network model to predict surface finish and dimensional deviation of the job based on the feedback from various sensors. They developed several network models to find out the influence of feedbacks from a number of sensors. The author observed that feed, depth of cut, radial force and feed force provide the best combination to build a model for on-line prediction of surface roughness and dimensional deviation.

Several researchers have compared the effectiveness of the neural network model with statistical regression models. Chryssolouris and Guillot (1990) observed the superiority of neural network model compared to regression model. On the other hand, Feng and Wang (2003) found multiple regression analysis and neural networks equally effective in predicting surface roughness for a finish turning process.

Risbood *et al.* (2003) predicted the surface roughness for the dry and wet turning of mild steels using carbide and high-speed steel tools. For each tool-job combination and cutting condition (dry or wet) different networks were used. The acceleration of radial vibration was taken as an input for on line prediction of surface roughness. The authors also developed a neural network model for predicting the dimensional deviation of the job. For this purpose, the radial cutting force, the radial acceleration of vibration, cutting speed, feed, depth of cut, length to diameter ratio of the job and position of the tool was taken as input parameter. Pal and Chakraborty (2005) predicted surface roughness by taking main cutting force, feed force, cutting speed, feed and depth of cut as input parameters of the network. Ozel and Karpat (2005) predicted surface roughness by developing two different network models. One model was offline with process parameters, tool and job information as input, whilst the other was an online model with cutting forces as an additional input. It was observed that the model with cutting forces as additional input yielded better results.

The major limitation to the use of neural networks is that it requires a large set of experimental data. Considering this limitation, Kohli and Dixit (2005) proposed an MLP neural network based methodology to predict surface roughness in turning process that uses small sized training and testing data sets. Their model predicts lower, most likely and upper estimate of the surface roughness for a cutting condition based on the algorithm developed by Ishbuchi and Tanaka (1991).

The above-mentioned researchers have essentially employed MLP neural networks and differ in terms of the network-training methodology and input parameters. Of late, there have been some applications of radial basis function (RBF) neural network as a substitute for MLP neural networks. Compared to MLP neural networks, RBF neural networks can be trained faster although requiring more training data. Sonar *et al.* (2006) studied the performance of radial basis function network (RBF) for predicting lower, most likely and upper estimates of surface roughness in turning process. They observed the performance of RBF neural network slightly inferior to MLP neural networks. Basak *et al.* (2007) used RBF models to predict surface roughness in finish hard turning process of AISI D2 cold rolled steel with mixed ceramic tools. For better modelling of an RBF network, assistance of multiple-linear regression was taken. Authors observed that in RBF neural network training, the spread parameter, which is essentially the zone of influence of a neuron, plays a significant role. Authors have proposed a strategy for the optimal selection of this parameter. Sarma and Dixit (2007) employed the MLP and RBF sequentially for predicting surface roughness of cast iron work piece with a ceramic tool. First, an MLP network was trained with limited training data. Afterwards, the trained MLP network was used to generate a large training data set for the RBF network. The experimental data set were used for testing of the RBF network. Considering that data may be noisy, authors have proposed a strategy to slightly perturb the data for better prediction.

Apart from neural networks, fuzzy set theory has been used for turning performance prediction in general and surface roughness in particular. A fuzzy based methodology has been proposed by Fang and Jawahir (1994) to assess total machining performance encompassing surface finish, tool-wear rate, dimensional accuracy, cutting power and chip breakability. The authors quantified the effect of major influencing factors on total machining performance by fuzzy-set method and developed a series of fuzzy-set models to give quantitative assessments for the given set of input conditions.

Abburi and Dixit (2006) developed a knowledge base system with the help of neural network and fuzzy set theory to predict surface roughness in turning process. The neural network is trained with experimental data. The trained network generates a huge data set that is fed to a fuzzy set based rule generation module. A large number of IF-THEN rules are generated that are reduced by using Boolean operations. This rule base module is used for predicting surface roughness for given process variables as well as for the inverse prediction of process variables for a given surface roughness. The performance of the developed knowledge based system was found satisfactory based on the validation with shop floor data.

A number of authors have used the combination of two or more soft computing tools as an effective strategy for the prediction of surface roughness. Jiao *et al.* (2005) used a neuro-fuzzy approach for the prediction of dimensional deviation. The developed fuzzy adaptive neural networks (FAN) are capable of providing both learning ability of a neural network and tolerance of imprecision, uncertainty and vagueness in the machining process. First, an approximate model representing the influence of machining parameters on dimensional deviation is established. This model is then improved by learning with the given training data. The authors showed the superiority of their model over a classical regression model. Nandi and Pratihar (2004) employed a fuzzy basis function network (FBNN) for predicting the surface roughness in ultra-precision turning. The parameters of the network were optimized with a genetic algorithm code.

2.3.1.2 Tool life and tool wear

Tool life and tool wear play a major role in the economic aspects of metal cutting operations. Most of the time, tool life is considered as a time lapse between two successive regrinds of tool when operating under specified cutting conditions. In an early work on tool wear estimation using neural networks, Ezugwu *et al.* (1995) predicted tool life and failure mode in machining of grey cast iron with ceramic cutting tools. The tool failure based on average and maximum flank wear, crater depth, notch depth, surface roughness and catastrophic failure of the tool have been considered. The experimental data have been used to train the MLP neural network using back propagation algorithm. However, the authors had limitation of having just 25 data. With these data, they could predict the correct tool life (within 20%) in 58.3% cases and tool failure mode in 87.5% cases. Dutta *et al.* (2000) studied the application of neural network with different learning schemes for faster processing of data which is a major criterion in on-line tool condition monitoring. Speed, feed, depth of cut and three components of cutting forces were inputs of the neural networks and flank wear was output of the network. The authors found that the modified back propagation algorithm converge faster than standard back propagation algorithm.

Tool life in hot machining of high magnesium steel has been modeled by Tosun and Ozler (2002) using an MLP neural network. The turning was carried out at four temperatures— room temperature, 200 °C, 400 °C and 600 °C. The neural network model predicted the tool life with better accuracy as compared to a regression model.

Ojha and Dixit (2005) proposed an economic and quicker method of tool life estimation and predicted low, most likely and higher estimates of tool life using neural networks. In their approach the tool life is estimated by fitting a best-fit line for the data falling in steady wear zone and finding the time till tool fails by extrapolation. For predicting the lower and upper estimates of the tool life, they used the backpropagation algorithm of Ishbuchi and Tanaka (1991), as was earlier done by Kohli and Dixit (2005). The neural network model was found superior to the multiple regressions model. Quiza *et al.* (2008) carried out an experimental investigation on tool wear prediction on ceramic cutting tools used for turning hardened cold rolled tool steel. They predicted tool wear with the help of neural network and regression models. The neural network model was found superior to the regression model.

Soft computing optimization techniques *viz*., genetic algorithm, particle swarm optimization and simulated annealing were used for optimizing neural network model parameters. For tool life estimation, Natarajan *et al.* (2006) employed a neural network model that was optimized by particle swarm optimization (PSO). The use of PSO resulted in reduction of training time by 50%.

Tool wear monitoring has been other widely investigated research topic. Tool wear monitoring is of two types: direct and indirect. Direct methods measure the actual values of size of wear parameters with optical instruments, whilst indirect methods measure parameters such as cutting forces or vibrations that are correlated with tool wear. A number of tool wear monitoring schemes have been proposed that employ vibrations, ultrasonic, torque, power, velocity and temperature sensors and sensor fusion. Sick (2002) has reviewed a number of research papers dealing with on-line and indirect tool wear monitoring in turning using artificial neural networks.

Mainly, vibrations, acoustic-emission, torque, power, velocity and temperature sensors were employed for obtaining the feedback for indirect estimation of tool wear. Some of the representative work is as follows. Das *et al.* (1996) developed a back propagation neural network model for the reliable on-line tool condition monitoring based on cutting force measurement. The ratio of cutting force components have been found as a better indicator of the tool wear.

Silva *et al.* (1998) developed neural network based online tool condition monitoring for turning process using signals from five sensors. They used two types of neural network learning algorithms— adaptive resonance theory (ART) and self-organising map (SOM) to classify statistical and frequency domain features of the sensor signals. The ART2 creates and classifies tool wear with less number of sampled data and has the ability to respond immediately. The SOM is a method of mapping a high-dimensional input space on to a one or two–dimensional output space by using an unsupervised neural model. The authors found that the NN with the SOM perform better than the ART2 in classifying unseen sampled data. Nadgir and Ozel (2000) employed a neural network to model tool condition monitoring system (TCMS). On line cutting forces were measured by a piezoelectric tool dynamometer and used as inputs to the network. Data obtained from several machining test with use of different cutting speed, feed and a constant depth of cut were used to train the network. Chungchoo and Saini (2002) proposed an on-line fuzzy neural network (FNN) to predict tool wear. Cutting forces, acoustic emission signals, skew and kurtosis of force bands and total energy of forces were taken as input parameters of the neural network.

2.3.1.3 Cutting force

Cutting force is one of the important characteristic variables to be monitored during machining process. Tool breakage, tool-wear and work piece deflection are mainly due to abnormal cutting force developed during machining process. To predict and monitor cutting forces various models were proposed using soft computing techniques.

Khanchustambham and Zhang (1992) used neural network to predict cutting force as well as surface finish during machining of ceramic material. Feed, depth of cut and spindle speed are used as input parameters for the network. The network is trained by cutting force signal and measured surface finish for on-line monitoring of turning process. Lee *et al.* (1995) used feed forward neural network to predict cutting force components. The network is trained using undeformed chip thickness, chip width, cutting speed and tool rake angle as input parameters. The authors found the predicted results are in good agreement with experimental data. Luong and Spedding (1995) developed neural network model to find cutting conditions for a given work material and required depth of cut to predict the cutting forces, surface roughness and tool life. They trained the network using data from Machining Data Handbook and have shown that the neural network establishes a correlation from empirical data. Szecsi (1999) used neural network model for cutting force estimation in turning process. Process parameters, tool geometry, work piece material and flank wear were taken as input parameters of the network. The author used 3200 training and 1500 testing data to train the network and found very good prediction accuracy. However, there is no discussion about the statistical variation of the cutting forces. Lin *et al.* (2003) used radial basis function neural network and multiple regression analysis to predict machining forces-tool wear relationship in machining of aluminum metal matrix composites. Besides process parameters, feed and cutting forces were used to estimate tool wear. The authors obtain better correlation of tool wear with feed force data than with cutting force.

In high speed turning, the correlations between various cutting parameters play an important role while model building. Ezugwu *et al.* (2005) used an ANN approach to model the correlation between five process parameters *viz.,* speed, feed rate, depth of cut, cutting time and coolant pressure with seven performance parameters *viz.*, tangential force, feed force, spindle motor power consumption, surface roughness, average flank wear, maximum flank wear and nose wear. The developed model agrees well with experimental data and can be used to analyze and predict the relationship between process and performance parameters.

Hao *et al.* (2006) proposed multi-layer perceptron neural network model for predicting cutting force in self-propelled rotary tool in turning. Cutting speed, feed rate, depth of cut and tool inclination angle were input parameters and thrust force, radial force and main cutting force are output parameters of the network. The authors applied hybrid of genetic algorithm (GA) and Back propagation (BP) algorithm to improve performance of neural network model.

Lin *et al.* (2001) predicted surface roughness and cutting force using abductive neural network during turning of high carbon steel with carbide inserts. Abductive networks are composed of a number of polynomial functional nodes organized into several layers. Unlike general neural network, abductive neural network generate optimal network architecture automatically and take less iterations in training. The network is trained with cutting speed, feed and depth of cut as input parameters. Predicted results are found to be more accurate compared to regression analysis. Li *et al.* (2000) used neuro-fuzzy techniques to estimate feed force by measuring motor current using current sensor in CNC turning centre. Motor current and feed rate was used as input parameters. The authors found that the estimated force was within an error of 5%.

2.3.1.4 Process optimization

The machining optimization problem is highly non-linear and possesses multiple solutions. In multi-objective optimization, cutting parameters is of great concern in manufacturing environment. Researchers considered various input (cutting) parameters like cutting speed, feed rate, depth of cut, cutting time, coolant pressure *etc.* and output (process) parameters like tangential force, axial force, radial force, feed force, spindle power consumption, surface roughness, tool life, average and maximum flank wear and nose wear *etc.* for modelling. Optimization of single-pass turning has been attempted in early works. However, in general, a turning operation involves a number of rough cuts and a final finish cut. In manufacturing industries, multi-pass turning is widely used than single-pass turning. The highest possible metal removal is aimed in rough passes, where surface finish is not an important consideration. However, in finish turning process surface finish is the most important consideration. Researchers have used soft computing optimization techniques *viz.*, fuzzy logic, neural network, simulated annealing, genetic algorithm, ant colony optimization and particle swarm optimization to optimize both single and multi-pass turning problem.

Karpat and Ozel (2005) developed a multi-objective optimization model for single pass turning to model surface roughness and tool wear. They used PSO based neural network optimization scheme to optimize finish hard turning processes using CBN tools. NN model predicts surface roughness and tool wear during machining and PSO is used to obtain optimum cutting speed, feed rate and tool geometry. The authors found that PSO takes less number of iterations to reach the optimum conditions.

Neural network used in fuzzy decision environment have been reported in literature. Wang (1993) used feed forward neural network using manufacturer's fuzzy preferences to determine the optimum cutting parameters by solving the multi-objective problem with the help of a neural network model. The objectives considered were productivity, operation cost and cutting quality. Lee *et al.* (1999) presented fuzzy non-linear programming model to optimize cutting conditions for a turning process. Neural network is modeled to predict cutting speed and machinability and the output of the optimization model is applied as inputs to neural network. Hashmi *et al.* (1999) developed a fuzzy logic model for selection of cutting conditions for machining

GAs are very suitable for solving multi-objective problems. Based on representation of design variables they are of two types: binary coded genetic algorithm (BGA) and real coded genetic algorithm (RGA). In machining optimization, RGAs are more precise, more consistent and lead to faster convergence. Abburi and Dixit (2007) developed an optimization methodology which is a combination of a real-coded genetic algorithm (RGA) and sequential quadratic programming (SQP) to obtain Pareto-optimal solutions to minimize production cost. The major advantage of the methodology is that various Pareto-optimal solutions are generated without the knowledge of the cost data. The optimization is carried out with equal depths of cut for roughing passes and the authors found that RGA combined with SQP is very efficient in reaching up to global optimum. Kim *et al.* (2008) also explored the applicability of real coded genetic algorithm (RGA) in machining optimization. In their work, RGA has been compared to SA, continuous SA, GA and generalized reduced gradient method.

Chen and Tasi (1996) used simulated annealing (SA) to optimize multi pass turning to obtain minimum unit production cost and found it effective. Chen and Tasi followed by many researchers applied SA approach to solve the optimization problem of multi pass turning process. Baykasoglu and Dereli (2002) have used SA to optimize cutting conditions in their heuristic model. However, they did not take surface finish into consideration. Onwubolu and Kumalo (2001) applied BGA to minimize unit production cost and concluded that BGA significantly performs better than SA. However, Chen and Chen (2003) have shown that Onwubolu and Kumalo have incorrectly handled the machining model and showed that the GA provides no better solution than SA.

The distribution of total depth of stock among different rough cuts and final finish pass is an important task in multi-pass turning optimization. Wang and Jawahir (2005) proposed GA based methodology for the allocation of total depth of cut in multi pass turning. A novel feature of this work is the consideration of tool wear in the optimization procedure.

In recent literature, newer optimization techniques such as ant colony optimization (ACO) and particle swarm optimization (PSO) are used to optimize multi pass machining. Srinivas *et al.* (2009) optimize multi pass turning process to minimize total production cost using PSO in their mathematical model. They used six cutting constraints *viz.,* variable bounds, tool life, cutting force, power, stable cutting region and chip-tool interface temperature. It is found that PSO provides optimal feasible solutions within a reasonable computational time. Vijayakumar *et al.* (2003) used ACO to minimize unit production cost subjected to various practical constraints and found that the approach performs better than SA and GA. However, it has been contradicted by Wang (2007).

Ojha *et al.* (2009) used neural networks, fuzzy sets and genetic algorithms based soft computing methodology to optimize parameters in multi pass turning operation. Neural network has been used for prediction of surface finish and tool life. In view of uncertainty, surface roughness is quantified by using fuzzy number. An equal depth of cut for roughing passes along with a single finish pass strategy, similar to earlier work of Yeo (1995) has been considered in optimization model. The optimization model has been applied for determining the optimum cutting parameters for two cases *viz.,* minimization of production cost and maximization of production rate.

2.3.2 Milling Process

Milling is a multipoint tool cutting process in which the cutter rotates at some speed while the work feeds past the cutter. The peripheral speed of the cutter called cutting speed, movement of the work piece under the work piece per unit time called feed rate or table feed, depth of cut in the direction along the cutter axis called axial depth of cut, depth of cut normal to the cutter axis called radial depth of cut and number of passes are process parameters. These parameters may be optimized for obtaining the minimum cost of machining and minimum production time. Review of research work related with optimization of milling process as well as prediction of various milling performance measures such as surface roughness, tool wear and cutting force presented in the following sub-sections.

2.3.2.1 Surface roughness

Surface roughness has been an important factor in predicting the performance measure of any machining process. In milling, it is influenced by machining parameters such as speed, feed and depth of cut, tool diameter, radial rake angle, nose radius, work piece material, tool material, machine vibration, *etc*. Researchers have attempted to predict surface roughness using an adaptive neuro-fuzzy inference system (ANFIS), genetic programming and fuzzy logic, mostly on end milling operation. GA and PSO were used as optimization techniques.

Lo (2003) used ANFIS to predict the surface roughness in end milling process. Spindle speed, feed rate and depth of cut were considered as input parameters. The ANFIS was modeled using triangular and trapezoidal membership functions. The average error of prediction of surface roughness for triangular membership function was found lower, around 4%. Ho *et al.* (2009) also proposed ANFIS to predict surface roughness on aluminum alloy work piece milled with HSS tool. They used a hybrid Taguchi genetic learning algorithm in the ANFIS to determine the most suitable membership functions and simultaneously find the optimum parameters by directly minimizing surface roughness error. The results show that their approach using Gaussian membership function outperforms the ANFIS method given in the MATLAB® tool box and other reported work.

Brezocnic *et al.* (2004) proposed a genetic programming (GP) approach to predict surface roughness in end milling process. The genetic programming is an evolutionary computation method that was first introduced by Koza (1992) in the year 1992. It aims to find out computer programs (called as chromosomes) whose size and structure dynamically changes during simulated evolution that best solve the problem. Cutting parameters *viz.,* spindle speed, feed and depth of cut as well as vibration between tool and work piece were used to predict the surface roughness and the authors found that the model that involves all these variables accurately predict the surface roughness.

Reddy and Rao (2005) developed an empirical surface roughness model for end milling of medium carbon steel, whose parameters were optimized using GA. Oktem *et al.* (2005) determined the optimum cutting conditions for minimum surface roughness in milling of mould surfaces. The surface roughness was modeled by response surface method and GA was used for optimizing the cutting conditions. Reddy and Rao (2006) used genetic algorithm to optimize tool geometry *viz.,* radial rake angle and nose radius and cutting conditions *viz.,* cutting speed and feed-rate to obtain desired surface quality in dry end milling process.

Prakasvudhisarn *et al.* (2009) proposed an approach to determine the optimum cutting condition for desired surface roughness in end milling. The approach consists of two parts: machine learning technique called support vector machining (SVMs) to predict surface roughness and particle swarm optimization technique for parameters optimization. The authors found that PSO shows consistent near-optimal solution with little effort.

Chen and Savage (2001) used fuzzy-net based model to predict surface roughness under different tool and work piece combination for end milling process. Speed, feed and depth of cut, vibration, tool diameter, tool material and work piece material are used as input variables for fuzzy system. The authors found that the predicted surface roughness is within an error of 10%**.** Iqbal *et al.* (2007) developed a fuzzy expert system for parameter optimization that includes prediction of tool life and surface finish in hard-milling (high speed milling of steel having 45 HRC hardness) process.

2.3.2.2 Tool wear and tool condition monitoring

During milling process, cutter wear reduces surface finish of the work piece and increases cutting forces, power consumption, *etc*. Unfortunately there is no direct way of online measuring of tool wear. In indirect method of estimation sensors are used to extract features from cutting zone and tool wear is estimated. Neural network, fuzzy logic and genetic algorithm are used to predict wear and monitoring tool condition online in face and end milling processes. Ghosh *et al.* (2007) developed a neural network-based sensor fusion model to estimate tool wear during CNC milling process. Signals in the form of cutting forces, spindle motor current and sound pressure level were used as inputs for neural network. The authors have proposed newer methods such as feature space filtering, prediction space filtering, *etc*. to improve prediction accuracy and found that the prediction is satisfactory in a real-time error-prone environment.

Chen and Black (1997) proposed a fuzzy-nets tool-breakage detection system for monitoring tool breakage in end milling operations. The developed system has self learning capability and generates fuzzy rule base from the experimental data. However, for the generation of fuzzy rules from given input-output data pairs they have used large data sets compared to that used by Kohli and Dixit (2005) in fitting neural networks for predicting surface roughness in turning process. The proposed system has ability to detect tool breakage online, approaching real time basis. Dutta *et al.* (2000) also proposed a fuzzy controlled back propagation neural network model for predicting the tool wear in face milling process. The convergence speed, prediction accuracy and total time of system development make this approach an attractive technique suitable for online tool condition monitoring. Fuzzy logic based in-process tool wear monitoring system has been proposed by Susanto and Chen (2003). Cutting parameters *viz.,* feed and depth of cut and maximum resultant cutting force were used as variables to predict flank wear of the cutter. The authors found that the system effectively monitors the wear condition on the tool during cutting process with an average error of 8.7%. Tansel *et al.* (2005) proposed tool monitoring system using genetic algorithm to monitor micro-end-milling operations that is able to estimate wear and local damages of the cutting edges of a tool. Dutta *et al.* (2006) predicted the wear of the tungsten carbide inserts using neural network during face milling of steel. They proposed a new approach called modified back propagation neural network with delta bar delta (MBPNND) learning to enhance the convergence speed and prediction accuracy of the network. The authors found that MBPNND is efficient compared to three other approaches *viz.,* back propagation neural network, fuzzy back propagation neural network, modified back propagation neural network.

Ching-kao and Lu (2007) used grey-fuzzy logic approach to predict optimal cutting conditions for improved tool life and metal removal rate during side milling of SUS304 stainless steel. Grey-fuzzy logic that combines grey relational analysis and a fuzzy logic is useful to deal poor, incomplete and uncertain information. The approach converts multi response variable into single response grey-fuzzy reasoning grade and simplifies optimization procedure. The result shows improved performance on side milling process with heavy cutting.

2.3.2.3 Cutting force

The performance of milling process is monitored through measurable output parameters mainly by cutting forces and controlling input cutting parameters to suit particular cutting conditions. Neural networks were used to predict cutting forces during machining. Tandon and El-Mounayari (2001) employed multi-layer perceptron network for modelling the cutting forces in end-milling process. The model was limited to one tool-work material combination (high speed steel tool and aluminum work piece) and the authors used 96 data for training and testing the network. However, the model does not take into account the tool wear. Zuperl and Cus (2004) proposed feed-forwarded neural network model with 10 input neurons to estimate cutting forces in ball-end milling operation. Cutting speed, feed rate, axial and radial depth of cut, cutter diameter, wok material type and its hardness, type of insert, tool wear and cutting fluid are used as input parameter of the neural network to predict three cutting force components.

Radhakrishnan and Nandan (2005) predicted cutting force model using regression and neural networks. A regression model is used to filter out abnormal data points and the filtered data were used in neural network for better prediction. Briceno *et al.* (2002) compared a multi-layer perceptron neural network with radial basis function network for prediction of machining forces in milling. The radial basis function network was found to be superior to multi-layer perceptron neural network. Considering the statistical variation, the neural networks were used for predicting minimum, maximum, mean and standard deviation of the cutting forces. Zuperl *et al.* (2006) also found the radial basis function neural networks as superior to the multi-layer perceptron in modelling of machining forces in ball-end milling. Aykut *et al.* (2007) used artificial neural network to predict cutting forces and studied machinablity in face milling process**.** Cutting speed, feed rate and depth of cut were used as input parameters of the network to predict three cutting forces and they found the estimated force within an error of 10%.

2.3.2.4 Process optimization

Most of the work in machining optimization has been focused on turning problems. In milling, the process is modeled with input parameters like cutting speed, feed rate, radial depth of cut, axial depth of cut, number of passes, cutter diameter, helix angle of the cutter, work piece hardness, milling orientation, cutting time, *etc.* and performance parameters like tangential force, axial force, radial force, feed force, spindle power consumption, arbor defection, surface roughness, tool life, average and maximum flank wear and nose wear *etc.* Soft computing optimization techniques such as neural network, genetic algorithm and simulated annealing are mostly employed to optimize both single and multi pass milling problem.

Tandon *et al.* (2002) used an artificial neural network to predict the cutting forces which in turn was used to optimize both feed and speed using particle swarm optimization algorithm for NC pocket milling process. They observed that the approach reduces machining time up to 35%.

Shunmugam *et al.* (2000) used genetic algorithm to optimize minimum production cost in face milling operation. The machining parameters *viz.*, number of passes, depth of cut in each pass, speed and feed are obtained. The authors found that the proposed optimization method yields better results to those reported in literatures. Cus *et al.* (2006) proposed an on-line monitoring and optimization of cutting process to optimize production cost and time for a desired surface finish in ball end milling process. The cutting forces measured by sensors were analysed and optimum cutting parameters is obtained by using genetic algorithm. The results show that the method is effective and can be integrated into real time manufacturing environment. Sreeram *et al.* (2006) also proposed binary coded genetic algorithm to optimize cutting parameters for maximum tool life as well as minimum production cost in micro end milling process. Depth of cut along with other cutting parameters (cutting speed and feed rate) has been considered as decision variables. The authors found improved results compared to tool supplier's data.

Genetic simulated annealing (GSA) algorithm, which is a hybrid of GA and SA is used by Wang *et al.* (2004) to determine the optimum machining parameters for plain milling process. GSA being hybrid algorithm exploits the strength of SA and GA and overcome their weakness in optimizing feed rate and speed for an objective of minimum total production time. Number of constraints such as allowable cutting speed, feed rate, horse power, arbor strength and arbor deflection are considered. The results show that GSA is more efficient than GA and geometric programming (GP).

Apart from economic objective of optimization, soft computing techniques have also been used for tool path planning in milling complex part geometry. Suh and Shin (1996) used neural network model for generation of tool path during rough cutting of pocket milling. The algorithm is validated through computer simulation as well as real machining.

2.3.3 Grinding Process

Grinding is a finishing process, widely used in manufacturing of components requiring fine tolerances and good surface finish. In basic cylindrical grinding process the wheel speed, work speed, work feed rate and radial depth of cut are the process parameters. The peripheral speed of the grinding wheel is called wheel speed and the peripheral speed of the work against wheel rotation called work speed. During machining process the work supported on the table travels across width of the wheel. The process parameters slightly vary for surface grinding and other grinding process. These parameters may be optimized for obtaining the minimum cost of machining and minimum production time. However, for optimization, performance of the process such as surface finish, wheel wear, grinding burn, grinding force and vibrations has to be predicted. The review of the literature in this area is presented in the following sub-sections.

2.3.3.1 Surface finish

In grinding process, surface roughness is one of the important factors in assessing the quality of the ground component. Operating parameters such as wheel speed, work speed, feed and depth of cut, work material properties, grinding wheel composition, coolant application, machine vibration, *etc*. are the variables that affect the surface roughness in the grinding process. Many of these variables are non-linear, inter-dependent and difficult to quantify. In recent decades, due to complexity of the grinding process soft computing techniques were used to estimate surface roughness that takes many variables into account and covers a wide range of cutting conditions.

Ali and Zhang (1999) proposed a fuzzy logic model for surface roughness estimation of work piece produced by surface grinding operation. They used 16 variables which are influential for surface roughness and found that the method is simple, effective and superior in modelling non-linearity. Nandi and Pratihar (2004) developed a genetic-fuzzy system for prediction of surface finish as well as power required in grinding process. They used GA to optimize fuzzy knowledge base offline and compared the predicted results with experimental data.

Kim (2002) developed a neuro-fuzzy model to optimize cycle time in plunge grinding process. He used grinding power, peak value of power spectrum and time constant as inputs for neural network to estimate work piece surface roughness. From the results obtained, he found that the model is more reliable than regression model. Samhouri and Surgenor (2005) proposed an adaptive neuro-fuzzy inference system (ANFIS) to predict surface roughness in grinding process. The power spectral density parameters of piezoelectric accelerometer were used as inputs to ANFIS and the authors found prediction accuracy as 91%.

In 1992, Wang and Mandel (1992) first introduced fuzzy basis function which has the capability of combining both numerical data and linguistic information. Thereafter, many researchers used fuzzy basis function network employing single or multiple variables. Nandi and Banerjee (2005) used fuzzy basis function neural network to predict surface roughness and corresponding power requirement in cylindrical plunge grinding process. Wheel speed, work speed and feed rate were considered as input variables and, power requirement and surface roughness as output variables of the network architecture. The fuzzy rule base was designed automatically using a genetic algorithm and from the results it was concluded that the model predicts better than mathematical models.

2.3.3.2 Wheel wear and grinding burn

Estimation of life of grinding wheel is important in grinding process. Generally, surface roughness, chatter marks and burn marks, *etc*. on work piece surface are considered as an indication for wheel life limit in grinding process. Deivanathan *et al.* (1999) used neural network to predict wheel life in cylindrical plunge grinding process. The occurrence of burn marks on the work piece surface was adopted as criterion for wheel life. Work speed, work diameter, infeed and power was taken as inputs of feed forward neural network to estimate 'time to burn' as output parameter. Wang *et al.* (2001) proposed a radial basis function neural network approach to detect grinding burn on work piece from acoustic emission (AE) signals. The neural network is trained with two sets of data, one being mean and standard of frequency of band power, the kurtosis and skew of AE signals and other being autoregressive coefficients. The authors found good prediction in both the cases. Ali and Zhang (2004) developed a fuzzy-rule based model for predicting burns in surface grinding of steel. They used 37 absolute and eight relative rules to predict the grinding conditions. The method was found suitable for many types of steels grinding condition. Liu *et al.* (2005) used fuzzy pattern recognition technique to predict grinding burn. They used acoustic signals which were obtained by wavelet pocket transform and optimized by fuzzy clustering.

Lezanski (2001) used neural network and fuzzy logic to monitor the grinding wheel condition using multiple sensors. Two grinding parameters *viz.,* grinding engagement and coolant volume rate and 14 sensory signal feature related to vibration, grinding forces, vibration-power spectrum, acoustic emission RMS and acoustic emission-power spectrum RMS, were used as inputs to the neural network. The author found that neuro-fuzzy system performs lower than neural network approach.

2.3.3.3 Grinding force and vibrations

Grinding forces influence the efficiency and productivity of the process since it affects the mechanism of metal removal, wheel wear, *etc*. Neural network approach is used to estimate grinding force in various grinding processes. Fuh and Wang (1997) used back propagation neural network to model grinding force in creep feed grinding process. Work speed, wheel speed, wheel diameter, depth of cut and grinding manner (up or down) are used as input parameters of the neural network to estimate vertical force and horizontal force.

Kawak and Ha (2004) proposed a neural network approach to predict chatter vibration and grinding burn. Data from AE sensor and power meter *viz.,* RMS peak, FFT peak, static power and dynamic power were used as input parameters to predict status of grinding condition as normal, burning, or chatter vibration. The authors found that the learned neural network with power and AE parameter provides the diagnosis of the grinding process better.

2.3.3.4 Process optimization

The optimization of grinding process is an important task due to accurate and economical means of shaping the parts into final product with required surface finish and high dimensional accuracy. Problems are formulated to maximize production rate and minimize production cost subject to constraints such as burns, wheel wear and machine stiffness and so on.

Liao and Chen (1994) used back propagation neural network (BPNN) model to optimize creep feed grinding process and found better result than the regression method. The model considers five input variables, *viz.,* bond type, mesh size, concentration of abrasive particles, work speed and depth of cut and three output parameters, *viz.,* surface finish, normal grinding force per unit width and grinding power per unit width. Govindasamy *et al.* (2005) developed a dynamic neural network model-based control strategy to minimize defects in grinding of aluminum disks. They found that the approach reduces thickness defects of the component.

Brinksmeier *et al.* (1998) proposed NN and fuzzy set based model to optimize the grinding processes. They evaluated the grinding process in terms of geometric quantities (such as dimension, shape, waviness and surface integrity) and product quality i.e., surface roughness. Lee and Shin (2000) proposed fuzzy basis function neural networks (FBFNs) for modelling of grinding processes to find the optimum process conditions. The model was applied for two grinding optimization problems *viz.,* creep feed grinding and surface grinding process and it was found that the NN based algorithm outperforms traditional optimization techniques in surface grinding process.

Saravanan and Sachithanandam (2001) proposed binary coded genetic algorithm (BGA) based optimization procedure to optimize cutting conditions, *viz.*, wheel speed, work speed, lead of dressing and depth of dressing in multi-objective surface grinding problem. They found that GA performs better than traditional quadratic programming. Gopal and Rao (2003) also developed a BGA based optimization procedure to optimize grinding conditions, *viz.,* depth of cut, work speed, grit size, density, using a multi-objective function model. From the results obtained they found that the material removal rate and cost of grinding are influenced more by surface roughness constraint than by that of surface damage. Baskar *et al.* (2004) also considered multi–objective surface grinding problem to optimize cutting conditions. They used ACO based optimization and found that it outperforms quadratic programming technique and GA.

2.3.4 Drilling Process

Drilling is the process of making a cylindrical hole in a solid work piece using a cutting tool called drill. The peripheral speed of the drill is called cutting speed. The movement of the drill along the axis of the hole for one revolution is called feed and radius of the drill is called as depth of cut. These three parameters *viz.*, cutting speed, feed and depth of cut are the process parameters. The special feature of drilling is that the cutting speed varies along the cutting edge, from almost zero near the center of the drill to the circumferential speed of the drill at its outer radius. These parameters may be optimized for obtaining the minimum cost of machining and minimum production time. However, for optimization, performance of the drilling process has to be predicted. Soft computing has been applied for machining performance prediction and optimization. The review of literatures in this area is presented here.

2.3.4.1 Surface finish and dimensional deviation

In industrial modern precision assembly system, “hole quality” in term of surface roughness and roundness is an important factor and is achieved by reaming process. The reaming is the process of accurately sizing and finishing the previously produced hole. Mathews and Shunmugam (1999) used ANN approach for predicting hole quality in reaming process. Acoustic emission, cutting force and vibration sensor data were used as input parameters and surface finish, roundness error and residual stress as output parameters of the neural network. The result shows that ANN predictions using multisensor data were much closer to the experimental values and this approach is found to be an effective approach.

The effect of drilling a hole on specialized materials like composites has been addressed by few researchers. In drilling of fiber-reinforced composites delamination and surface finish are two major concerns. Taso and Hocheng (2008) used RBF neural network with three drilling factors, *viz*., spindle speed, feed rate and drill diameter to predict thrust force and surface finish of candlestick drill in drilling of composite materials. The predicted results show that the errors in thrust force and surface finish prediction are below 2% and 4%, respectively and the procedure is more accurate compared to regression analysis.

Nandi and Davim (2009) used fuzzy logic rules to predict the performance of drilling process with minimum quantity lubricant in aluminum alloy work piece. The comparison of model prediction with experimental results shows that the fuzzy rule base model with Takagi–Sugeno–Kang type of fuzzy rules provides better prediction of surface roughness, cutting power and specific cutting force.

2.3.4.2 Tool life and tool wear

The useful life of drill and its operating conditions largely control the economics of the machining operations. Cutting parameters such as speed, feed, cutting force, *etc*. influences flank wear of the drill. Fuzzy logic and neural network were used to estimate the drill life, drill wear and monitor the drill condition.

Biglari and Fang (1995) used real-time fuzzy logic control for maximizing drill life in a small-hole drilling (3 mm diameter) process. Experiments were conducted under five different drill wear conditions—initial, normal wear, acceptable wear, severe wear and drill failure to record thrust force, torque and radial force, which developed 53 fuzzy rules. The methodology for online monitoring of drill wear is used for controlling drill feed rate for maximum tool life.

Lin and Ting (1996) predicted drill wear using neural network and regression models. Average thrust force and torque, spindle speed, feed rate and drill diameter were used as input parameters and average flank wear was the only output parameter of the neural network. The authors found that the neural networks with two hidden layers learn faster and can more accurately estimate tool wear than the networks with one hidden layer.

Liu *et al*. (2000) used polynomial network for in-process prediction of corner wear on the drill. The polynomial network is composed of a number of functional nodes having self-organizing feature with an ability to construct the relationship between input and output variables. It has greater prediction accuracy and have fewer internal network connections, compared to backpropagation network. Thrust force or torque, cutting speed, feed rate and drill diameter were used as input parameters. The authors found that the use of thrust force in the model provides predictions within an error of 10%. Abu Mahfouz (2003) compared several architectures of feedforward BPNN for tool condition monitoring of twist drill wear. The network is trained using vibration data and five drill wear conditions, *viz*., chisel wear, crater wear, flank wear, edge fracture and corner wear, which were artificially introduced in the network for prediction of drill wear. Fully connected networks were found to be better than grouped network and the vibration signals are promising data for tool condition monitoring.

Sanjay *et al*. (2005) proposed BPNN to detect drill wear on 8 mm HSS drill while machining mild steel. The network is trained using spindle speed, feed, drill size, machining time, torque and thrust force as input parameters. The authors found that the three layered network with the hidden layer having two and ten neurons is the best layered network and predicted values are accurate compared to regression analysis for all the combination of cutting speed and feed. Panda *et al*. (2006) also used BPNN for predicting flank wear on HSS twist drill while drilling mild steel work piece. The network is trained with spindle speed, feed rate, drill diameter, thrust force, torque and chip thickness as input parameters. The authors found that the inclusion of chip thickness reduces mean square training error and takes less number of iteration. Patra *et al*. (2007) have considered spindle motor current signal (RMS current) in addition to process parameters, used as input parameters for predicting drill wear using BPNN. They found that the predicted values provide better accuracy compared to a regression model. Garg *et al*. (2007) compared BPNN with RBFN for prediction of flank wear in drilling process. Chip thickness was used as additional parameter to train the networks. From the results, the authors found that RBFN requires large number of training patterns and large network architecture to achieve same level of desired accuracy as the BPNN in machining copper and mild steel work piece with HSS drill bits.

Choi *et al*. (2008) used neural network to predict incipient stage of drill failure so as to prevent any damage in the drilling process. Time and frequency domains of feed motor current were taken as input parameters and drill wear state, *viz*., 0.1 mm for normal state and 0.9 mm for drill failure state, as output parameters of the neural network. The authors found that the proposed algorithm predicted the drill breakage accurately for different cutting conditions and machine tool types.

Khajavi and Komanduri (1993) used BPNN to predict drill wear employing multiple sensors. The signals from four sensors, *viz*., thrust, torque and strain in two directions, were used. It is found that the change in area under power spectral density plots shows good correlation with corner drill wear. The authors concluded that one sensor signal would be adequate for drill wear estimation.

Liu and Chen (1998) developed a BPNN model for online detection of drill wear for drilling stainless steel work piece with HSS drills. They trained the network with drill size, feed rate, spindle speed and eight features of thrust and torque signals, *viz*., average thrust, average torque, peak thrust, peak torque, RMS thrust, RMS torque, area under thrust vs time curve and area under torque vs time curve as input parameters and two stage drill wear state—useable and failed as output parameters of the network. The authors found that the developed system shows high robustness and usefulness for complex production environments like flexible manufacturing system. Li and Tso (1999) proposed fuzzy logic model for online drill wear state monitoring using spindle motor and feed motor current signals. They classified drill wear into three categories, *viz*., small (0.2 mm), normal (0.5 mm) and severe (0.8 mm), to ensure tool replacement at the proper time and found that flank wear can be accurately predicted by using feed motor current.

2.3.4.3 Drilling force

In drilling process thrust force and its control is a major concern. The thrust force and torque are influenced mainly by work piece hardness, drill point angle, drill diameter and feed rate. The various soft computing techniques were used to estimate the thrust force and control the drilling process particularly in drilling of composite laminates where occurrence of delamination at exit and entrance planes is the main problem.

Stone and Krishnamurthy (1996) used neural network to model the relationship between feed rate and the thrust force during drilling of fiber-reinforced composite materials using diamond tipped drill. They developed a neural network based controller that minimizes the problem of delamination or crack growth during the drilling process. The authors compared the proposed method with experimental results and found that thrust force controlled drilling process is advantageous over conventional constant feed drilling process. The similar problem of delamination during drilling is modeled by Chung and Tomizuka (2001) using fuzzy logic. They proposed drill thrust force control strategy by adjusting drill feed rate to reduce occurrences of delamination in drilling composite materials.

Karri (1999) used RBF neural network to predict thrust and torque during process. Tool geometry and operating conditions *viz.,* spindle speed, feed and drill diameter were used as input parameters of the network. The author tested network over a range of process variable and found the predicted values within an error of 10%.

Sheng and Tomizuka (2006) developed an intelligent control system using neural network and fuzzy logic to control thrust force in drilling process. Drill head position information is included in neural network model and fuzzy logic is used to deal with gain variation due to drill wear. The proposed model was compared with simulation and experimental results. It was found that the method worked well over a wide operating range.

2.3.4.4 Process optimization

In conventional drilling process actual cutting takes very less time compared to other non-productive time. Therefore, the process optimization to minimize cutting time is of less concern among the researchers compared to other metal cutting process. A few researchers have attempted to determine correlation among cutting parameters.

Lee *et al.* (1998) developed an abductive neural network model for predicting drill life, metal removal rate, thrust force and torque. Cutting speed, feed rate and drill diameter were taken as input parameters of the network and simulated annealing algorithm is used to optimize the process parameters considering tool cost and productivity.

Hashmi *et al.* (2000) developed fuzzy logic model for selection of cutting speed to drill three different work materials *viz.*, medium carbon steel, low carbon alloy steel and medium carbon free machining steel. The predicted drilling speeds for different work material hardness shows good co-relation with machining data handbook.

Ghaiebi and Solimanpur (2007) used an ant algorithm to minimize tool air-time and tool switching time in a multiple hole making process employing several tools. The authors found that the proposed method is effective and efficient compared to traditional dynamic programming. Zhu and Zhang (2007) proposed a PSO algorithm for solving the problem of drilling path optimization in holes machining in CNC machining centre. The proposed algorithm obtains the global optimization solution with reduced computational time.

2.3.5 Discussion on the Review of Soft computing Based Modelling and Optimization

From the review of the research papers, it is clear that soft computing techniques have been applied for prediction of surface roughness, dimensional deviation, tool life, tool wear and cutting forces. A number of soft computing techniques have been used for the optimization of machining processes. The main objectives in the optimization of turning, milling and grinding have been minimization of cost of machining and maximization of production rate. Multi-objective problems also have been solved. In drilling process, maximization of drill life and optimization of drill path have been the main objectives.

For the prediction of performance parameters, MLP neural networks with single hidden layer have been widely used. Some authors (Lin and Ting, 1999) found that MLP neural networks with two hidden layers are superior to those with single hidden layer.

However, there is no general agreement for it. Neural network training algorithms require some adjustable parameters. Some skill is needed in choosing the proper value of the parameter for faster training and better accuracy. Some authors (Dutta *et al.,* 2000; Dutta *et al.,* 2006) have modified the standard back propagation algorithm for improving the training. There are a few papers that paid attention to develop a systematic procedure for deciding the size of training and testing data sets and for improving the quality of the data (Kohli and Dixit, 2005; Ghosh *et al.,* 2007). A number of authors have used radial basis function networks and observed their superiority in terms of network training (Sonar *et al.,* 2006; Basak *et al.,* 2007; Taso and Hochang, 2008; Karri, 1999).

Compared to neural networks, there is lesser number of applications of fuzzy sets for the prediction of machining performance. Neural network models have been found superior to fuzzy set based model in learning. However, the knowledge captured by them is not transparent. On the other hand, fuzzy set based models provide a rule base that can be interpreted. To take the advantages of both, Abburi and Dixit (2009) developed a knowledge base system, in which the neural network was used for learning and fuzzy set theory was used for rule generation and inference. There have been some applications of neuro-fuzzy systems (Nandi and

Pratihar, 2004; Chungchoo and Saini, 2002; Li *et al.,* 2000; Lo, 2003; Chen and Savage, 2001; Kim, 2002, Samhouri and Surgenor, 2005). For optimization, GA and SA have been applied by a number of researchers (Abburi and Dixit, 2007; Kim *et al.,* 2008; Chen and Tasi, 1996; Baykasoglu and Dereli, 2002; Onwubolu and Kumalo, 2001; Chen and Chen, 2003; Wang and Jawahir, 2005; Ojha *et al.,* 2005; Oktem *et al*., 2005; Shunmugam *et al.,* 2000; Wang *et al.,* 2004; Saravanan and Sachithanandam, 2001). There have been a few application of neural networks (Wang, 1993; Tandon, *et al*., 2002, Liao and Chen, 1994; Govindasamy, 2005; Brinksmeier *et al.,* 1998), fuzzy sets (Lee, *et al.,* 1999, Hashmi *et al.,* 1999; Hashmi *et al.,* 2000), PSO (Srinivas *et al.,* 2009; Zhu and Zhang, 2007) and ACO (Vijayakumar *et al.,* 2003; Ghaiebi and Solimanpur, 2007) in optimization.

MLP neural networks, RBF neural networks and fuzzy sets have been used mostly for performance prediction of machining processes. The applicability of a particular soft computing technique is not dependent on the type of machining process, but is dependent on the amount of data/information available, training time required and the transparency in prediction needed. Table 2.1 shows the relative merits and demerits of these three techniques based on the information available in the literature and the experience of the author and supervisors of this thesis. These techniques do not provide a unique solution and their performance is dependent on the skill of model-developer, atleast to some extent. However, Table 2.1 fairly summarizes the general observation

Considering the relative merits and demerits, it is natural that many authors are prompted to use the techniques that are combinations of these techniques. The best strategy can be to first develop a neural network based model and use it to build a fuzzy set based model to attain transparency and improved performance as was done in (Abburi and Dixit, 2006). If enough training data is available, then RBF neural network should be given preference to MLP neural network. In absence of enough data, a strategy suggested in (Sarma and Dixit, 2007) may be adopted, where first an MLP neural network is trained with limited data, which is used to generate a huge amount of data for training an RBF neural network with real data reserved for testing.

Table 2.1. Comparison of MLP neural network, RBF neural network and fuzzy sets for performance prediction in machining.

Technique	Merits	Demerits
MLP neural network	• Can learn from limited dataset. • Provide enough accuracy if trained properly.	• Training process requires skill and is time consuming. • Very poor in extrapolation.
RBF neural network	• Training is easier and faster	• Requires more training data than MLP. • Accuracy slightly inferior to MLP
Fuzzy sets	• Can easily incorporate expert knowledge. • Transparent compared to neural networks. • Reasonable performance in extrapolation.	• Need huge amount of data for automatic rule generation. • Some skill is needed for assigning membership grades to subjective information.

Soft computing based optimization techniques like GA, SA, ACO and PSO are suitable for optimization. The best strategy is to use neural networks and fuzzy set theory for performance prediction and GA/SA/ACO/PSO for optimization. The optimization techniques can be used for adjusting the internal parameters of predictive systems, for example, for adjusting the weights in neural networks and membership functions in fuzzy sets. Among the four optimization techniques mentioned in this review, GA has been widely used and has matured as a robust technique of metal cutting optimization. The real parameter GA that does not require the decision variables to be converted as binary numbers is very effective for solving multi-objective optimization problem. ACO and PSO have been applied for machining optimization recently. Due to simplicity in its execution, the PSO may emerge out to be a viable alternative to GA.

Following issues have not received enough attention and need to be addressed:

1. **Optimization with online learning:** In optimizing the machining process, the knowledge of tool life as a function of cutting parameters is an essential requirement. It is not economical to conduct tool life test for different work piece-tool combinations. Also, the neural network or fuzzy-based systems need a number of data for training. The experiments consume a lot of time and may not be sufficient for online optimization. A scheme can be developed that can carry out the optimization with online learning. The approach may be helpful for predicting optimized process parameters in real time machining environment economically
2. **Coping with statistical variation:** In machining, the performance variables possess statistical variation. A reliable model building requires a number of replicate experiments. The built model should be able to predict the probability distribution of the performance parameters. There have been very few attempts in this direction. Some authors have developed neural network models that can predict the confidence intervals (Chryssolouris *et al.,* 1996; Shao *et al.,* 1997) of the prediction. However, it has not been applied to machining. In references (Kohli and Dixit, 2005; Sonar *et al.,* 2006), the most likely, upper and lower estimates of the performance variables are predicted.
3. **Data filtration:** The prediction accuracy is dependent on the accuracy of the experimental data. The data may invariably contain noise. Therefore, it is necessary to filter it for removing the outliers. Only a few researchers have paid attention to this aspect (Kohli and Dixit, 2005; Radakrishnan and Nandan, 2005).
4. **Hybrid computing:** The combination of hard (physics based) and soft computing has not been provided proper attention in metal cutting arena. Such attempts have been successfully employed in other areas of manufacturing, for example in Geerdes and Alvardo (2008). Finite element analysis can be combined with soft computing tools to obtain more accuracy and have less dependency on experimental data.

5. **Incorporating the time as a variable in prediction and optimization:** Most of the researchers have predicted surface roughness and cutting force as a function of cutting process parameters only. The performance of the machining process gets changed with time due to tool wear. Therefore, there is a need to model these parameters with time as a variable. Similarly, in optimization also, the time effect has to be considered. For example, the effect of tool wear has been considered to optimize multi-pass turning operations in (Wang and Jawahir, 2005).
6. **Research on hardware side:** Effectiveness of soft computing tools for online prediction is dependent on the use of proper sensors. In many cases, combination of many sensors called sensor fusion is helpful. The development of proper sensors should go side by side with the development of software tools. At the same time, efficient techniques for data acquisition, data filtration and feature extraction should be developed.

2.3.6 Concluding Remarks for Soft Computing Based Modelling and Optimization of Machining Processes

The major observations from the review of literature in the area of performance modelling and optimization are as follows:

1. Neural network models have been effectively employed for predicting the surface roughness of machined components in turning and drilling. However, most of the models do not predict the surface roughness as a function of time, concentrating on the time zone when surface finish changes only slightly with time.
2. Most of the authors have used MLP neural networks. Some authors have employed RBF neural networks due to their ability of getting trained much faster. RBF neural networks require more training data and provide slightly inferior accuracy. However, it is only an experimental observation and in the context of machining, no mathematical proof has been provided to support this observation.

3. Fuzzy sets and combination of fuzzy sets and neural networks have been used for predicting the surface roughness in turning, milling and grinding. Fuzzy set based methods are especially advantageous when the expert knowledge is available.
4. Neural networks have also been used for surface finish prediction in reaming and drilling of composite materials.
5. Soft computing tools have been used for estimating the tool wear and tool life. However, the results are not as impressive as in the case of surface roughness prediction. This is due to highly statistical nature of tool life and tool wear and difficulty in identifying a measurable parameter with which the tool wear can be well-correlated. Most of the authors have used cutting force components as input parameters in their soft computing based models. Several other signals such as acoustic emission, vibrations and temperatures have also been tried. It is observed that instead of raw data from sensors, features extracted from the signals are more effective in modelling the tool wear and tool life.
6. Whereas surface roughness and tool wear are the major indicators of machining performance, modelling of cutting force gains importance for its use in estimating the tool condition, particularly the tool breakage detection. Modelling of cutting force may also be used for estimating the power in machining. However, minimization of cutting power is not as significant objective as the minimization of machining cost and time.
7. Soft computing optimization methods like GA have been used in machining area for two purposes— (i) for optimizing the internal parameters of neural networks, fuzzy sets and neuro-fuzzy systems and (ii) for machining optimization. Application of GA for machining has attained maturity. PSO, a relatively newer technique, may emerge as a better alternative to GA.

The best strategy is to use a combination of fuzzy and neural network for performance prediction and soft computing optimization tool like GA or PSO for optimization. However some important issues need urgent attention. These are (i) acquisition of data in an economical and efficient way, (ii) filtering of noisy data and

(iii) extracting the statistical feature of the data. If these issues are addressed the industrial application of these techniques will become an easy and fruitful task.

2.4 Web Based Machining and Cloud Computing Applications in Manufacturing

With the advancement in technological development and communication tools the concept of virtual enterprises, web based manufacturing, collaborative design and distance manufacturing have become reality. The manufacturing industries are now evolving towards digitalisation, networking and globalisation (Rahman, *et al*., 1999; Lal *et al*., 2007). In a web based manufacturing environment the manufacturing industries use computer and web technology for connecting different manufacturer and customers globally on wire.

Starting from the late 1990s, web-based manufacturing is steadily gaining popularity. With the help of World Wide Web, information distributed at different locations can be accessed and shared by users anywhere in the world using web tools such as web browsers. Yang and Xue (2003) carried out a comprehensive review of recent research on developing web-based manufacturing systems. Several researchers have applied web technologies for various applications such as mechanical design, quality modelling, production scheduling, process planning, *etc*.

Chen *et al.* (1998) developed an Integrated Concurrent Engineering Design for Mechanical Parts system. In this system the user can access product information from different domains of computerized tools by interacting with the system through graphical user interface (GUI) interactive screens. Also the design can be evaluated based on criteria of design for manufacturing and design for assembly. Huang and Mak (2001) developed a web-based quality function deployment (QFD) modelling system. It is a three-tier architecture system. The first tier connects QFD application clients with web server. The QFD web server and QFD application server are two kinds of middle tier and QFD database server is third tier. The application server is responsible for dealing with database server for synchronization and computational activities.

Li *et al.* (2001) developed a web-based system for supply chain management employing heuristic method for collaborative decision making. Web-based production scheduling introduced by Jia *et al.* (2002) obtains the optimal schedule considering different requirements and manufacturing resources distributed at different locations. Wang and Jawahir (2004) developed web-based, user interactive expert system for optimization of single-pass milling for the selection of optimum cutting conditions using genetic algorithms. The system allows user to access relevant databases and calculate results according to user's request/ input and generate HTML documents to send back corresponding information. Wu and Liao (2005) developed an internet-based machining parameter optimization and management system using genetic algorithm (GA) for high speed machining process. Thakur and Pande (2006) developed an internet-based system for feature modelling and process planning of sheet metal components commonly manufactured by blanking, shearing and bending processes. In the area of process optimization and information sharing using web technologies, less numbers of research papers are available. Zheng *et al.* (2008) developed a web-based machining parameter selection system for agile turning which provides user appropriate selection of machine tools, cutting tools and cutting parameters for reduction in product life cycle cost and increase in product quality.

Recently, cloud computing has evolved from the web-based technologies. The cloud computing is a concept of providing computing services and data on demand. There are four types of models *viz*., private, community, public, or hybrid is used for getting/sharing the information *via* internet cloud. Very recently, the concept of cloud computing and cloud manufacturing has started gaining popularity. Rimal *et al*. (2011) classified architectural requirements into cloud providers, the enterprises that use the cloud, and cloud users. They also provided the guidelines to software architects and cloud computing application developers for creating architectures.

Marston *et al*. (2011) discusses on various issues that will affect the different stakeholders of cloud computing and also provided key recommendations for the cloud providers. Similar to cloud computing, cloud manufacturing aims to share manufacturing resources to make "manufacturing as service" environment. Xu (2012) discusses the essential requirements of a cloud computing system useful for software architects and developers to design cloud-based applications. They also presented different layers viz., manufacturing resource layer, virtual service layer, global service layer and application layer of cloud manufacturing system framework. Considering the availability of cloud computing infrastructure, it is possible to develop reliable predictive models and carry out the offline optimization in a cloud. The offline results supplied by the cloud can be fine-tuned at the client side and feedback can be provided to the cloud. Compared to other types of clouds, a private cloud is much secure. It can also interact with outside world as per the requirement.

2.5 Summary and Gaps in Literature

In the literature, soft computing techniques are highly preferred by the researchers for performance modelling and optimization of machining process. The main objectives in the optimization of turning, milling and grinding have been minimization of cost of machining and maximization of production rate. Multi-objective problems also have been solved. In drilling process, maximization of drill life and optimization of drill path have been the main objectives.

A number of optimization techniques have been proposed by the researchers to various machining processes. These are comparable by the accuracy of the result obtained and computational time needed. All these problems are of offline type and need *a priori* machining models. During optimization researchers employ empirical relations or conduct offline experiments to develop predictive models for predicting surface roughness, tool life, cutting force, *etc*. Therefore the application of these techniques in shop floor involves a number of constraints and has many limitations in its implementation.

Recently, the use of computer and internet technologies has become popular in the area of process optimization and information sharing. The applications were found only in the area of production scheduling and process planning. Very recently, the concept of cloud computing and cloud manufacturing aims to share manufacturing resources to make “manufacturing as service” environment. There are only a few publications are available in this area. Many issues about the cloud manufacturing are still confusing and need more attention. Therefore the challenging issues are as follows:

- In the manufacturing sector, the application of web technology have been found in the area such as quality modeling, production scheduling, process planning, *etc*. However in machining optimization only a few attempts were made. A web based cloud computing system connecting various clients facilitate sharing machining related information and avail computing services under a common platform. The feasibility of developing a cloud computing system for optimization of machining process needs to be explored.
- The optimized process parameters using conventional and soft computing techniques have a number of limitations in its implementation. There is a need to develop an optimization methodology that will optimize the process online during the production of components. Optimizing the process based on real machining environment provides reliable optimum parameters. It can be directly implemented in the shop floor at different locations *via* web based system.
- The fuzzy set theory is used mainly in modelling and control problems. Its application as general optimization tool, particularly for optimizing machining problem will be very advantageous. It has the ability to handle uncertain and imprecision data. It can facilitate incorporation of expert knowledge in the optimization process. The development of fuzzy set based optimization can emerge as viable alternative to other optimization techniques. An attempt to develop fuzzy set based online optimization methodology applicable for cloud based optimization system is essential.
- With the development of modern soft computing techniques the application of artificial intelligence (AI) have been found in many applications. The AI based machining optimization learns the complex machining process. It obtains optimum process parameters during the job production process predicting better guess as more and more jobs are machined. An attempt in this direction applicable to cloud based system is felt essential in this thesis.

2.6 Scope and Detailed Objectives of the Present work

Based on the literature survey and to fill the gaps in the literatures the following scope and objectives are identified:

1. **Development of cloud computing frame work:** A web enabled cloud computing system provides computing services and data on demand connecting different manufacturers globally on wire. The application of cloud computing in the area of machining optimization is explored in this work. The major objective of this thesis is to develop a cloud computing frame work suitable for machining optimization in which clouds support machining database and computing facilities.

2. **Development of online machining optimization methodology:** The optimization of machining process employing offline techniques has many limitations in its implementation. Therefore a new machining optimization methodology, which can hope up with real time machining environment suitable for cloud computing system is felt very essential.

   (a) Development of heuristic based procedure: In this work, a heuristic based online optimization methodology is aimed to be developed. The methodology does not use *a priori* machining models and optimizes the process online during job production.

   The proposed optimization technique is demonstrated with virtual simulation. A 'virtual machine' is developed, which replaces machine tool available in the shop floor. Coding is written in MATLAB® for the development of virtual machine and optimization algorithm.

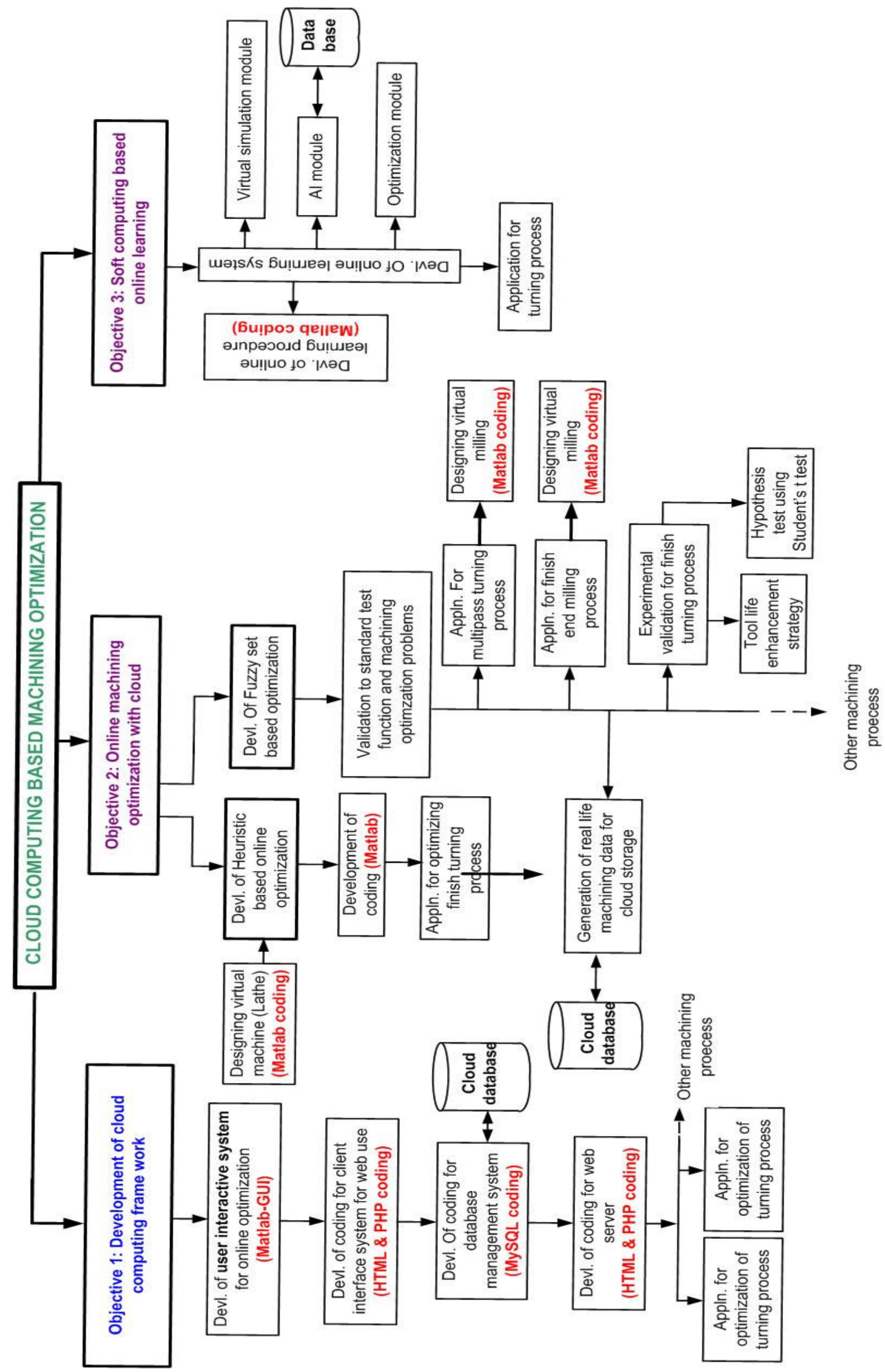
   (b) Development of an efficient fuzzy set based machining optimization strategy: A number of machining optimization techniques based on recent soft computing method is developed by the researchers. All these problems are of offline type and need *a priori* machining models. Therefore in this thesis, an efficient fuzzy set based online machining optimization strategy is developed. The procedure acknowledges the presence of statistical variation in the process and also efficient for the problem with a large number of design variables.

3. **Development of online learning system in a cloud computing based machining optimization:** The application of soft computing for online learning of cloud based machining optimization is the final objective of this thesis. The different modules of the AI system and an online learning

procedure are aimed to be developed. The online learning procedure is demonstrated with the help of numerical experiments.

To fulfil above objectives the research plan is drafted and is shown in Figure 2.2

Figure 2.2. Research plan for development of cloud computing based machining optimization system



Chapter 3

Cloud Computing Frame Work for Optimization

3.1 Introduction

Cloud computing is a new paradigm for hosting and delivering services over internet. The information distributed at different locations can be accessed and shared by users anywhere in the world using web tools such as web browsers. The 'cloud' in the system supports machining database and computing facilities. The system is used for sharing the machining related information among the users and provides optimized parameter. The basic frame work of the cloud computing system for optimization of machining process is described in this Chapter. The background of cloud computing is presented in Section 3.2. The optimization of machining process based on offline and online techniques is described in Section 3.3. The basics on development of interactive machining system used for online optimization of machining processes are presented in Section 3.4. The development of cloud computing system for optimization is discussed in Section 3.5. Finally, a summary is presented in Section 3.6.

3.2 A Background of Cloud Computing

With the rapid development of computer system and information technology the data processing, storage and information sharing through internet have become easier. It makes the computing resources cheaper, more powerful and made available ubiquitously than ever before. The new technological trend that has enabled the realization as computing model is called 'Cloud computing'. The cloud computing is a concept of providing computing services and data on demand. There are many definitions given for cloud computing. The National Institute of Standards and Technology (Mell and Grance, 2009) defines cloud computing as follows:

> **"Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (*e.g.*, networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management efforts or service provider interactions**."

Cloud computing is becoming more and more popular today. Large companies are gaining their popularity as they share valuable resources in a cost effective way. It facilitates the user to carryout their computational tasks and retrieve data from the virtualized cloud.

In the history of cloud computing, in the year 1961, John McCarthy first proposed the idea of computation being delivered as a public utility. He suggested in a speech at MIT (Massachusetts Institute of Technology) that computing can be sold like a utility, like electricity or water. In the year 1999, Salesforce.com started delivering applications to users through website and this way the dream of computing sold as utility started in reality. In 2002 Amazon started Amazon Web Services, providing services like storage, computation and even human intelligence. The turning point in the evolution of cloud computing came with the arrival of browser based cloud enterprise applications. Google Apps, Microsoft Windows Azure, and companies like Oracle and HP have joined in the main stream of cloud computing (Blaisdell, 2011).

In cloud computing various computing resources *viz*., hardware, networks, storage facilities, computing services, and interfaces can be delivered as a service *via* internet. Cloud providers host and manage the underlying infrastructure and offer cloud services to consumers/users. Cloud services include the delivery of software, infrastructure, and storage over the internet (either as a separate components or in a complete platform) based on user demand. It allows user to run the applications and store data online. Cloud computing providers offer services that can be mainly grouped into three categories: infrastructure as a service (*IaaS*), software as a service (*SaaS*), and platform as a service (*PaaS*) (Tao *et al*., 2011; Dillon *et al*., 2010). These are briefly described as follows:

- Infrastructure as a service (*IaaS*): In this model, the infrastructure facilities such as storage and computing capabilities (high performance servers) are provided as a service. These services are offered on a pay-as-you-go/use basis. The consumers of *IaaS* are usually engineers and mangers, who need access to these computing resources.

- Software as a service (*SaaS*): In this model, the application and software are offered as a service. The application running on the cloud can be accessed by the user on payable basis. The user does not require installing and running the application on the computer. In the traditional system user buys licensed software and gains ownership for its maintenance and installation whereas *SaaS* is cost effective and gives significant advantage to the user. The examples are the applications such as e-mail, Facebook, and Twitter as people use everyday.

- Platform as a service (*PaaS*): In this model the platform is provided as a service. This includes the cloud infrastructure such as network, servers and operating systems. These are used for the development and deployment of applications. The main difference between *SaaS* and *PaaS* is that *SaaS* only hosts completed cloud applications whereas *PaaS* offers a development platform that hosts both completed and in-progress cloud applications. This requires *PaaS* to possess development infrastructure including programming environment, tools, and configuration management in addition to supporting application hosting environment. Microsoft windows Azure Platform Services, Google Applications, and Amazon's Relational Database Services are some of examples of *PaaS*.

However the cloud services are not limited. Depending on the applications, the cloud providers have specific services such as Data storage as a Service (DaaS), Design as a Service (DaaS), Manufacturing as a Service (MFGaaS), Experimentation as a Service (EaaS), and so on.

3.2.1 Cloud Deployment Models

The cloud infrastructure can be owned by a private organization, a group of organizations, or the public. The following are four cloud deployment models defined by the cloud community:

- Private cloud: "The cloud infrastructure is operated solely for an organization. It may be operated by the organization or a third party and may exist on premise or off premise."
- Community cloud: "The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns. It may be managed by the organizations or a third party and may exist on premise or off premise."
- Public cloud. "The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services."
- Hybrid cloud. "The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability."

3.2.2 Cloud Security

In cloud computing data security is one of the main issues. All data belonging to user are stored on an external server. When the cloud computing runs on the internet, it is vulnerable to some security threats from hackers. Usually, the cloud provider should assure that their infrastructure is secure and that their clients' data and applications are protected. The customer must ensure that the cloud provider has taken proper security measures to protect the information. Researchers have analyzed security aspects of the cloud and suggested suitable measures to counter security threats. Rosenthal *et al.* (2010) suggested selective application of cloud such that highly confidential data are not put on the web, while other type of data is placed on the web cloud. Jamil and Zaki (2011) suggested forming private cloud for selective data, which can interact with other types of clouds through a secure bridge with firewall.

Mostly, security threats are related with public cloud system. Some of the security measures include (i) adding authentication layers, (ii) streamlining logging and monitoring, (iii) identifying/selecting right application for the public cloud and (iv) evaluating the security measure time to time. Zissis and Lekkas (2012) proposed introducing a trusted third party, tasked with assuring specific security issues within a cloud environment.

3.2.3 Advantages and Disadvantages of Cloud Computing

There are many advantages of using cloud computing. The computing resources and data provided through the cloud will benefit all types of business, if used properly. The following are some of the benefits of cloud computing (Jamil and Zaki, 2011):

- Cloud computing is cost effective to use, maintain and upgrade. The cost of traditional desktop software and operating system increases with number of licenses needed. On the other hand cloud computing is available at cheaper rate with an option as one-time-payment, pay-as-you-go/use, and so on.

- Cloud computing can be accessed from anywhere through internet/intranet connection and it is highly automated.

- Backup and recovery of stored information is easier compared to any other physical storage.

- The user is no longer tied up with any specific computer system so as to use software of particular version and updating is not necessary.

There are some limitations of the cloud computing. Payment of fee for using cloud computing applications is needed. For storing data proper care needs to be taken in order to avoid replication of data stored across multiple machines. The internet connectivity should always be made available for cloud applications, failing which, the accessing own documents would not be possible sometimes. When user system interfaces with cloud for data transfer, the documents need to be sent back and forth between the systems. It requires high speed internet connection for speed operation.

3.3 Machining Optimization

The success of the manufacturing process depends upon the selection of appropriate process parameters. It is very important for metal cutting industries since it has significant contribution to the overall cost of the product. Modelling and optimization of process parameters is usually a difficult task since it needs to consider the number of aspects. It requires (i) knowledge of manufacturing process, (ii) empirical equations to develop realistic constraints, (iii) specification of machine capabilities, (iv) development of an effective optimization criterion, and (v) knowledge of mathematical and numerical optimization techniques. Optimization aims to obtain optimal combination of cutting parameters (*i.e.*, cutting speed, feed, and depth of cut) for one or more economic objectives satisfying the constraints. It is one of the most widely researched topics in the area of machining process (Aggarwal and Singh, 2005; Mukherjee and Ray, 2006). Maximization of production rate, minimization of cost, maximization of surface finish, *etc*., are some of the objectives. Bounds of process parameters, cutting force, tool life, *etc*., are various constraints for optimization (Dixit and Dixit, 2008; Rao, 2011). The typical problem formulation is to either minimize or maximize the objective function subjected to a set of constraints. Different process variables are the constrained to the maximum and minimum limits. The formulation is as follows:

$$\text{Minimize/Maximize } Z=f\,(v, f, d) \tag{3.1}$$

Subject to,

$$R_{a_{\min}} \le R_a \le R_{a_{\max}} \quad \text{(Surface roughness constraint)} \tag{3.2}$$

$$F_{\min} \le F \le F_{\max} \quad \text{(Cutting force constraint)} \tag{3.3}$$

$$T_{\min} \le T \le T_{\max} \quad \text{(Tool life constraint)} \tag{3.4}$$

The variable bounds are

$$v_{\min} \le v \le v_{\max}, \tag{3.5}$$

$$f_{\min} \le f \le f_{\max}, \tag{3.6}$$

$$d_{\min} \le d \le d_{\max}, \tag{3.7}$$

where v, f, d, F, T, and R_a are cutting speed, feed, depth of cut, cutting force, tool life, and surface roughness, respectively. The machining optimization problem is highly nonlinear and complex to solve. Many traditional and non traditional optimization techniques were used for finding the optimal cutting parameters for single or multi pass machining problems.

Optimization of single pass machining has been attempted in early works. In manufacturing industries, multipass machining is widely used than single-pass machining. It involves a number of rough cuts and a final finish cut. The highest possible metal removal is aimed in rough passes, where surface finish is not an important consideration. However, in finish process, surface finish is the most important consideration. Among the conventional machining processes, in this work the attention has been paid to two commonly used processes—turning and milling. Turning is used for producing axisymmetric components and milling for producing flat or curved surfaces and prismatic shapes. These processes are performed by conventional and computer numerical control (CNC) machine tools. The traditional methods such as graphical techniques, constrained optimization strategy, dynamic programming, branch and bound algorithm, *etc*., either result in local minima, or take long time to converge (Lambert and Walvekar, 1978; Shin and Joo, 1992; Gupta, *et al*., 1995). In recent past, the researchers have used soft computing techniques as they are being preferred to physics-based models in predicting the performance of machining processes with the use of neural network (NN) and fuzzy logic (FL) and optimizing them with the use of genetic algorithm (GA), simulated annealing (SA), particle swarm optimization (PSO), and ant colony optimization (ACO) (Wang and Jawahir, 2005; Vijayakumar, *et al*., 2003; Ojha, *et al*., 2009).

The optimization problem is formulated and parameters are optimized using suitable optimization techniques. The prediction of surface roughness, tool life, cutting force, *etc*., for a particular cutting condition is obtained by evaluation. Due to high complexity involved in machining, the modelling and evaluation of these parameters is difficult but it is necessary for proper optimization of the process.

Based on optimization procedure it is broadly classified into two categories *viz.*, offline optimization and online optimization.

3.3.1 Offline Optimization

In offline optimization, the formulated optimization problem is solved with appropriate optimization techniques in offline mode. The obtained optimum parameters are used for machining the components. During optimization the different parameters such as surface roughness, tool life, and cutting force are obtained using empirical relations or with the use of predictive modelling. Most of the researchers have taken either the empirical surface roughness relation or the ideal surface roughness formula based on geometric consideration as given by

$$R_a = \frac{f^2}{32R}, \tag{3.8}$$

where R_a is the centre line average (CLA) surface roughness value in µm, f is the feed in mm/rev and R is the tool nose radius in mm. This assumes that only nose portion of the tool is making contact with workpiece and it is not valid when the feed is more than nose radius. For the determination of tool life most of the researchers have used extended Taylor's tool life formula,

$$vT^p f^q d^r = C' \tag{3.9}$$

where v is the cutting speed in m/min, T is the tool life in minutes, d is the depth of cut in mm and p, q, r and C' are the constants for a particular tool and work material combination. However, homogeneity of tool and work material and stability of machine tool have significant effects on tool life. Similarly, cutting force and tool wear are also predicted as functions of cutting process parameters.

As a result of this the optimal cutting parameters obtained by offline optimization need to be fine tuned online. Although the research on the optimization of machining process started since about 60 years back with the pioneering work of Gilbert (1950), there is hardly any paper that demonstrates implementation of optimization techniques in the shop floor. The main reasons are as follows:

- For solving the optimization problem offline, the functional relationship between the tool life and cutting process parameters must be known. Often this information is not available due to requirement of extensive tests.

- In the constrained optimization, dependence of process parameters on tool life, cutting force, cutting power *etc*. is usually not available.
- The results of offline optimization need to be fine tuned online depending on the rigidity of machine tool, deviation in the properties of material and a number of random causes.

Therefore, there is a need for appropriate optimization technique that optimizes the process by its direct implementation in the shop floor.

3.3.2 Online Optimization

Online optimization is *a posteriori* optimization. The selection of machining parameters is based on the feedback obtained from the machining process. There are some differences in the perception of word "online". Some researchers consider online optimization as an automatic and continuous adjustment of the cutting parameters without human intervention and process interruption for obtaining the best possible machining performance. In a broader sense the word "online optimization" mean the optimization by a human or machine operator in real-time (*i.e.*, during actual production) by taking the shop floor feedback either through sensors or human sense organs (Dixit *et al*., 2012). Thus, online optimization need not be automatic. It is different from offline optimization in the sense that offline optimization uses machining process models, like cutting force model, tool wear model, surface roughness model, *etc*. based on *a priori* knowledge gathered from offline experiments.

In the present work, a generalized heuristic based online optimization of machining process is developed. The scheme first suggests an initial cutting condition and machining is performed. Based on machining feed back, the cutting condition is modified to improve the machining performance. The procedure is continued till optimum cutting parameters are obtained. The scheme does not employ any empirical relations or predictive modelling for the evaluation of cutting force (F), surface roughness (R_a), and tool life (T). The proposed heuristic based online optimization scheme cope up with real time machining environment and can be directly implemented in the shop floor. The heuristic procedure for the online optimization is described in detail in Chapter 4. The proposed methodology has potential to consider statistical variation in prediction and optimization of the machining process, which is not considered in most of the offline optimization techniques available in the literature. Figure 3.1 shows proposed scheme of online optimization.

The strategy mimics the behaviour of a common machine tool operator in the shop floor. The most common practice followed in a machine shop is as follows. The operator makes a suitable guess for the process parameters, carries out the machining and fine tunes the process parameters by observing the machining performance. In the process, operator's knowledge about the process gets strengthened that helps him/her in arriving at the better guess in the next situation. The proposed methodology is suitable for any work material and tool material combination.

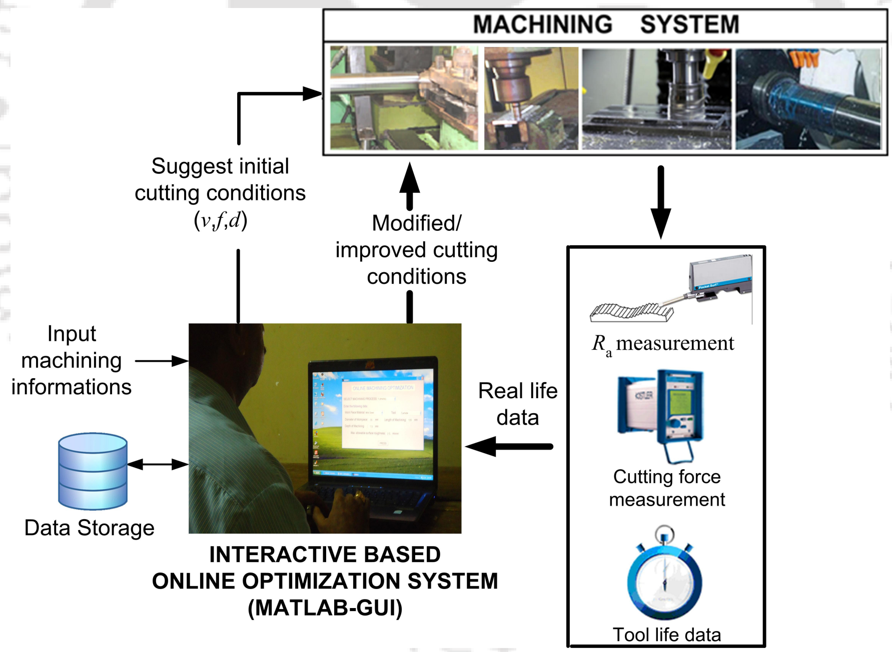


Figure 3.1. Online optimization scheme

The following are the benefits of online machining optimization strategy:

- In the proposed optimization process the surface roughness and tool life are directly obtained from machining process.
- The optimization strategy does not require any offline experimentation.

- The data generated during online machining optimization can be used for online learning of the process.
- The actual surface roughness and tool life can be used for building predictive models.

The shop floor implementation of the proposed online optimization procedure is made easier with the development of an interactive machining system. The details for development of an interactive menu are presented in the next section.

3.4 Development of an Interactive Menu

The online optimization of machining process is implemented through the development of interactive menu (screen) design. It facilitates the machine tool operator/user to easily interact with the system. A number of interactive screens are designed, through which the user inputs machining informations and obtains cutting conditions. The operator performs the machining process as per instructions displayed on the screen and inputs the machining feed back such as actual surface roughness, tool life, and cutting force into the system. Based on the inputted real life machining information the system provides improved cutting conditions suggested by online optimization algorithm. Thus the process is optimized through interactive session between the operator and the system. The optimum cutting parameters are evaluated and displayed on the system screen. The operator does not require performing any computation for the same.

The interactive screens are designed for inputting the machining information in the appropriate windows and push buttons are provided for the operator to proceed from one screen to the next. MATLAB® (R2010 a) Graphical User Interface (GUI) is used to develop the interactive screen system.

3.4.1 GUI: A Pictorial Interface Program

A graphical user interface of MATLAB® is a pictorial interface program. It provides the user to develop an easy interactive menu driven system. The environment contains static text, edit text, pop-up menu, push buttons, *etc.* which is very familiar to the user and can be operated with mouse click. In this work, GUI of MATLAB® (R2010a) is used. GUI is created using a tool called 'guide'. This tool allows displaying GUI layout and permitting for selecting and aligning the GUI components to be placed on it. Once the components are in place, its properties such as name, colour, size, font, text to display, *etc*. are edited.

When 'guide' is saved it creates working program including skeleton functions that can be modified whenever required. Figure 3.2 shows the layout editor showing different elements of GUI.

The use of different elements of GUI in menu development is as follows:

- The large white area with grid lines is the layout area, where a programmer can layout the GUI components.

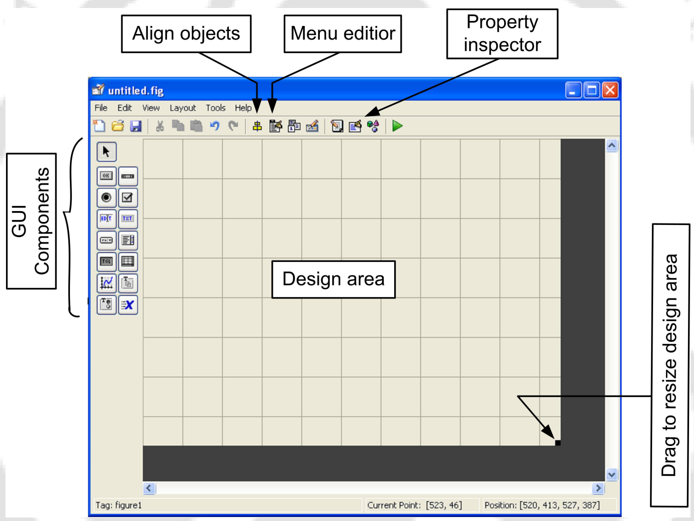


Figure 3.2. Layout editor window of MATLAB® GUI

- The layout editor window has a palate of GUI components along the left side of the layout area. A user can create any number of GUI components by first clicking on the desired components (icons), and then dragging its outline in the layout area.

- The top of the window has a toolbar with a series of useful *tools* that allow the user to distribute and align GUI components, modify the properties of GUI components, add menus to GUIs, and so on.

The basic steps required to create the interactive screen is as follows:

Step 1: First decide the different GUI components needed to create an interactive screen.

Step 2: Layout the components on a figure using 'guide' tool. Adjust the size, alignment and spacing of the components on the figure using 'property inspector'.

Step 3: Finally save the figure to a file. When the figure is saved, two files will be created with same file name, but one having extension as *.fig* and other as *.m*. The M-file contains the code to load the figure and display GUI screen.

Step 4: Extend the M-file code to implement the behavior associated with each callback function.

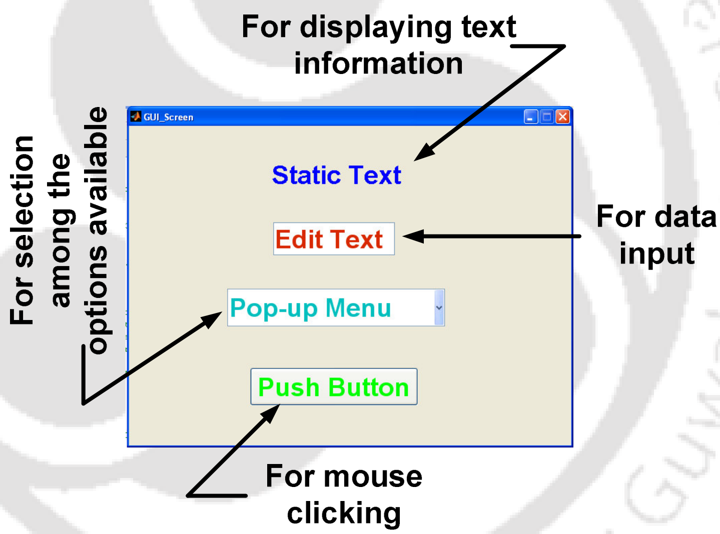


Figure 3.3. GUI screen with different components

Figure 3.3 shows a GUI screen consisting of various buttons (MATLAB® GUI components) used for different purpose of data entry/display. 'Static text' is used for displaying instruction to the user and 'edit text' is used for acceptance of data such as work material diameter, machining length, *etc*. 'Pop-up menu' help the user for quick selection of work material and tool material used for machining. 'Push button' to facilitate mouse click allows the user to move on to previous or next screen in the system.

3.4.2 Development of an Interactive Screen for Online Machining Optimization

In order to develop interactive screens in GUI, the required informations that are needed to be displayed or accepted is created using the appropriate GUI elements along with its desirable properties. Figure 3.4 shows the GUI figure of an interactive screen in development stage.

The complete form of interactive screens is presented in subsequent section. The development of this figure in GUI creates the corresponding program code and it is as follows:

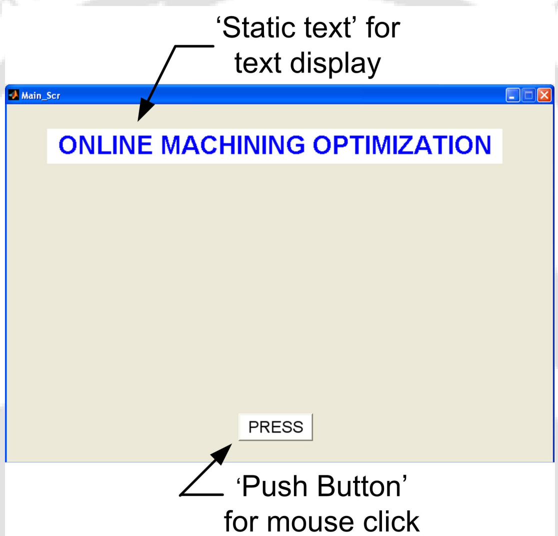


Figure 3.4. An interactive screen prepared in GUI

```
function varargout = Main_Scr(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Main_Scr_OpeningFcn, ...
                   'gui_OutputFcn',  @Main_Scr_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
```

```
else
    gui_mainfcn(gui_State, varargin{:});
end
function Main_Scr_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = Main_Scr_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
function pushbutton1_Callback(hObject, eventdata, handles)
```

In this work the developed interactive machining scheme is operated in a user friendly manner. The scheme is developed with menu driven screens in which the machine tool operator/user interacts with the system for inputting and obtaining information and so as to optimize the process online.

3.5 Cloud Computing Based Optimization and Database (CCBOD)

The cloud computing based optimization system is a web system having a web server integrated with database that connects different clients *via* internet/Intranet for accessing and sharing informations. The cloud supports machining database and computing facilities. The physical setup of CCBOD is shown in Figure 3.5.

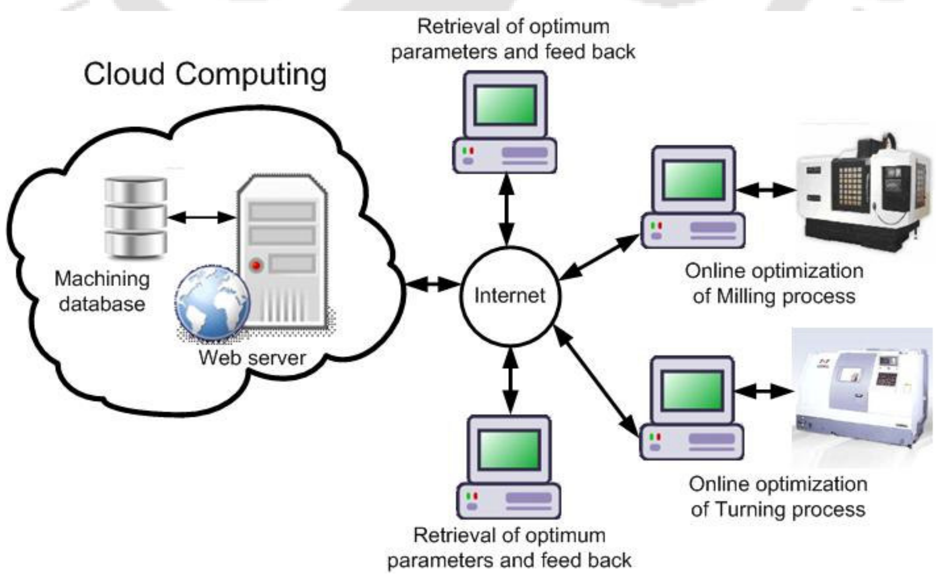


Figure 3.5. Structure of the CCBOD system

The server which hosts the CCBOD system has been configured to run in Windows environment and use ‘Apache’ web server. The Apache is an open-source web server platform developed by Apache Software Foundation (Apache guide, 2011). The web server, the online optimization system, the machining database, retrieval and sharing of information and the clients are various components of CCBOD. Being the central link, the client interfaces web server *via* internet/intranet by typing the internet protocol (IP) address. Every web server has an IP address known as domain name. Entering the uniform resource location (URL), the client’s internet browser sends request to the web server and the server then fetches the page and sends it to the client browser to establish remote link *via* internet. Figure 3.6 shows the home/main page of the proposed CCBOD developed in this thesis.

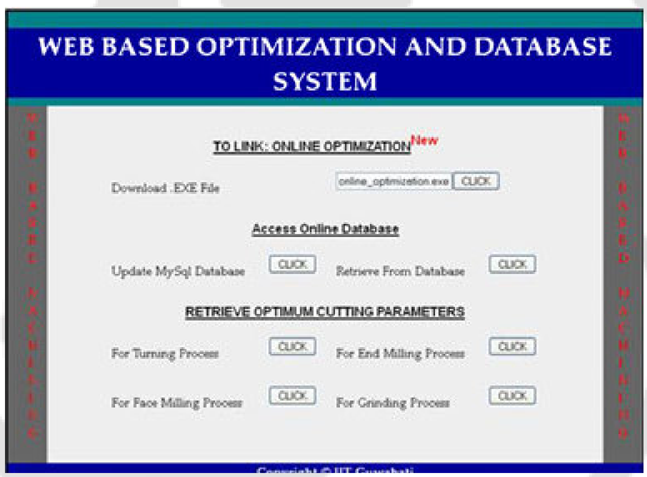


Figure 3.6. Home page of CCBOD system

3.5.1 Communication Structure of CCBOD

The developed cloud based system used for machining optimization is based on the internet. The system has ‘client-server’ architecture. In this, both the client programs and the server (administrator) programs are preserved in the server side. However, the client can automatically download informations connecting *via* web browser. The system uses HTML, CSS and PHP scripts for clients/users to interact with the system. Figure 3.7 visualize the communication structure, describing the different feature of the CCBOD system.

The user-interactive online optimization of multipass machining scheme is coded in MATLAB® with GUI. User can browse the system for implementation at their location and submit feedback data. The informations are stored in a database server, which can be accessed by the user. The various features of developed CCBOD system are as follows:

- The web-enabled feature helps for accessing and sharing information among the users located at anywhere in the globe *via* web browsers.

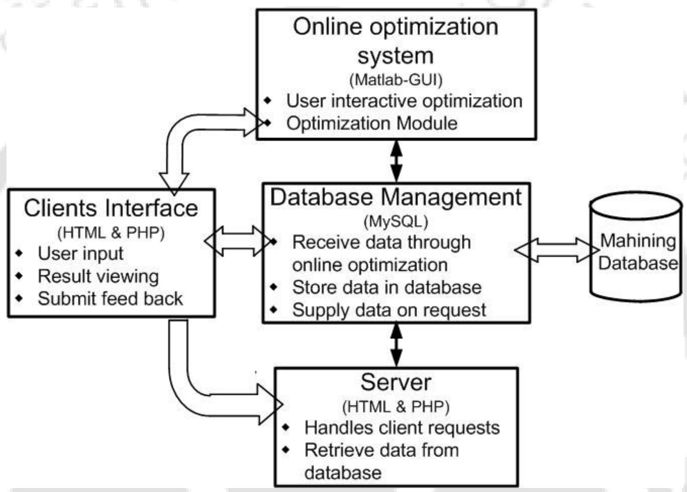


Figure 3.7. Communication structure of CCBOD system

- The proposed online optimization scheme suggests cutting conditions for shop floor machine tool operator, accepts feedback, and optimizes the machining process during the production of components.

- The cloud supports machining database and computing facilities. The database enriched with real machining datasets obtained by online machining optimization. The user can get data as well as computational service on demand.

- The main server also provides the optimized process parameters on demand to various clients. The clients can use the optimized data and fine-tune them if necessary.

- The system has advantage of open access, applicable to optimization of different process, and simplified interactive access and provides opportunity for sharing the process data information among different manufacturers.

3.5.2 Components of CCBOD

The different components and its functions of the proposed CCBOD system are discussed here.

3.5.2.1 Web Server

Web server is a computer program that runs on a computer system to deliver web pages on request. In the present work, an intranet web server with domain name 202.140.80.14 is used. The main/home page of CCBOD is named as home.html. If http://202.140.80.14/home.html is entered in URL of the client's browser, it sends a request to the web server. The server then fetches the home page and sends it to the client's browser. The server which hosts the CCBOD has been configured to run in Windows environment and use 'Apache' web server

3.5.2.2 Online Machining Optimization System

Online machining optimization is the core part of the CCBOD. It is an interactive computer program coded in MATLAB® (R2010a) with GUI and is used to perform online optimization of different machining process. While accessing CCBOD system the user can download 'online_optimization.exe' file in their system. It is used for optimizing the machining at their location. Figure 3.8 shows the flowchart describing how to use online optimization system of CCBOD.

The user communicates with the system through interactive screens designed for inputting and accepting the informations. Initially, the user enters the system and inputs machining informations such as process, work material type, tool material, work/cutter diameter, machining length, depth of machining, and maximum allowable surface roughness. The system suggests initial cutting conditions to be selected.

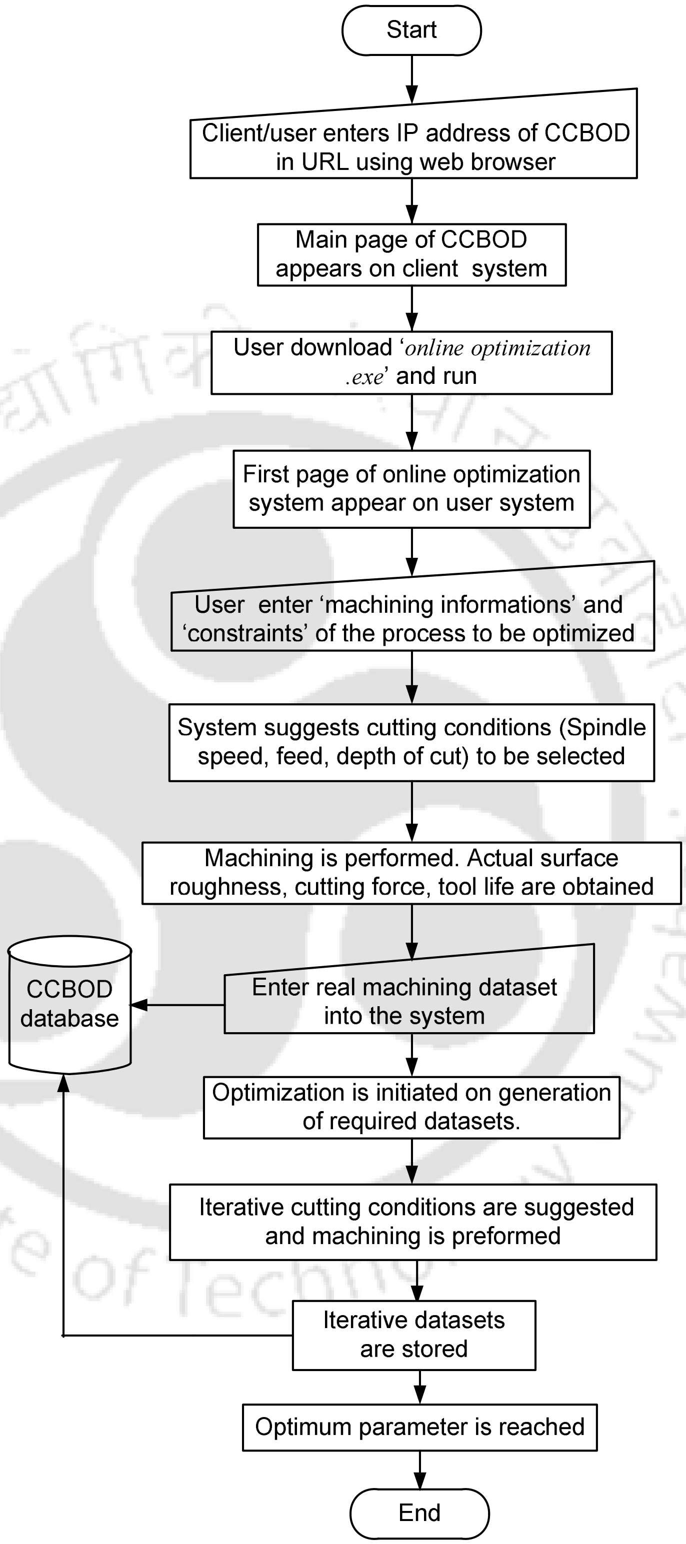


Figure 3.8. Flowchart for using online optimization system of CCBOD

The user implements it and the machining feed back *viz*., actual surface roughness, cutting force and tool life from real machining environment is entered into the system. Based on feedback received the system suggests improved cutting conditions and thereby the machining process is optimized online. The developed interactive based online optimization system is user friendly. The user does not require performing any computation and being aware of the heuristic algorithm for selection of different cutting conditions.

The user communicates with the system through interactive screens designed for inputting and accepting the informations. Initially, the user enters the system and inputs machining informations such as process, work material type, tool material, work/cutter diameter, machining length, depth of machining, and maximum allowable surface roughness. The system suggests initial cutting conditions to be selected. The user implements it and the machining feed back *viz*., actual surface roughness, cutting force and tool life from real machining environment is inputted into the system. Based on feedback received the system suggests improved cutting conditions and thereby the machining process is optimized online. The developed interactive based online optimization system is user friendly. The user does not require to perform any computation and to be aware of the heuristic algorithm for selection of different cutting conditions.

3.5.2.3. Machining Database

The database consists of organized data structures that store the informations which can be accessed as and when required. The online machining information of different machining processes contributed by the clients is stored process wise in a separate database. CCBOD stores this information and maintains the same on server side. Four databases one for each process *viz*., turning, end milling, face milling, and grinding are created for storing information such as cutting parameters and corresponding performance measures *i.e.*, actual surface roughness, cutting force, tool life, and function value. In this work, MySQL database system is used for storing data on the web.

MySQL is an acronym for My Structured Query Language (SQL) and it is a relational database system used to store information in the web applications. It is free and open software that works on different platforms such as Microsoft windows, Linux, Unix Ware, Solaris, *etc*. It allows creating a relational database structure on a web-server in order to store data or automate the procedures. The data is stored as database objects called ‘tables’. The structure of the table consists of different field variables, its type, and extra features such as auto-increment and primary key used for indexing the datasets. A row consisting number of fields termed as ‘record’ and a database file have many such records. The CCBOD system database maintains eight tables (also known as files):

(i) Four tables *viz*., db_turning, db_endmilling, db_facemilling and db_grinding for storing online machining informations

(ii) Four tables *viz*., fb_turning, fb_endmilling, fb_facemilling, and fb_grinding for storing feed back information.

The clients contribute this information to the database, which is maintained by the server. These records can be indexed with ‘prime_key’. It is one of the fields of database structure.

3.5.2.4 Retrieval of Optimum Cutting Parameters and Feedback

The web-based implementation allows the users to access database information. For retrieval and display of information at client screen HTML coding are written in the PHP scripting language. PHP is an open source of HTML scripting language commonly used for web development. It is used in this work as it can run on many different operating systems and web browsers. It is used to fetch the information from the respective database for display on the browser screen. It supports many data bases *viz*., MySQL, Sybase, Solid, Postgre SQL, *etc*. A PHP file has a .php extension and normally contains HTML tags and PHP scripting code. The code starts and ends with HTML tags as <*html*> and <*html*> respectively. Similarly PHP script begins with <*?php* and ends with *?*> and each line of PHP code ends with semicolon (*;*). The PHP uses two basic statement *viz*., *echo* and *print* to output text. The following is an example PHP script that sends the text "WEB BASED OPTIMIZATION AND DATABASE SYSTEM” back to the browser.

```
<html>
<body>
<?php
echo "WEB BASED OPTIMIZATON AND DATABASE SYSTEM";
?>
</body>
</html>
```

The variables are also used in PHP to hold numerical values or words. It is identified with *$* sign such as *$data*, *$db*, *etc*. The conditional execution of certain coding uses logical statements such as *if...else*, *if…elseif……else*, and *switch* statements. For example, *if…else* statement executes some code if logical condition is true/satisfied and executes another code if the condition is false.

```
<html>
<body>
<?php
if($result=mysqli_query($db,"select sno,v,f,d,Ra,T,FUN from
table1"))
      {
      while($data1=mysqli_fetch_object($result))
      {
      if(($data1->Ra<0.96*$ra)&&($data1->Ra>0.64*$ra))
      {
      if($min>$data1->Ra)
      {
      $min=$data1->Ra;
      $s=$data1->sno;
      }
      }
      }
      }?>
</body>
</html>
```

The above coding is used to fetch data from MySQL database to retrieve optimum cutting parameters (v, f, d) that have minimum function value (*FUN*) satisfying the limits of allowable surface roughness value (R_a).

3.5.2.5 Clients

The client and server are two parts of the CCBOD system. In a web based system, these are two distinct machines. A client or user is one who connects with the system from remote location for seeking information. The user is browser based thin client. They generally do not require any software installation and it is available across platforms. However MATLAB® (R2010 a) is required for the operation of online optimization system. The minimal hardware requirement includes a local computer with internet connectivity having web browser for accessing CCBOD system. The users input data and obtain outputs through keyboard and monitor display, respectively. The operational speed depends upon bandwidth available. The following are hardware requirement for client system.

- **Processor**: Minimum 1.0 GHz Intel or AMD (Advanced Micro Devices) 32 bit Processor
- **Memory**: 1 GB minimum
- **Disk**: 4 GB minimum (recommended)
- **Operating System** : Windows XP Professional, Windows Vista, or Windows 7
- **Monitor**: 17” at 1024X768 resolution
- **Mouse**: PS/2 or USB Mouse
- **Network Connectivity:** 56 kb Modem (for dial-up) or Ethernet Network Adapter (for cable model, DSL, *etc.*)
- **Wireless Network Connectivity:**: Wi-Fi Certified 802.11a/b/g/n Wireless Network Adapter (laptop only)
- **External Ports**: USB 2.0
- **Web Browser**: (Mozilla Firefox recommended)

3.5.3 System Information Flow

The data/information flow in CCBOD is in two different levels. Figure 3.9 shows data flow diagram (DFD) at zero level. It depicts that clients/users interact with CCBOD. Initially user enters the URL of the website to enter into the system and accesses the information being transacted by the developed system.

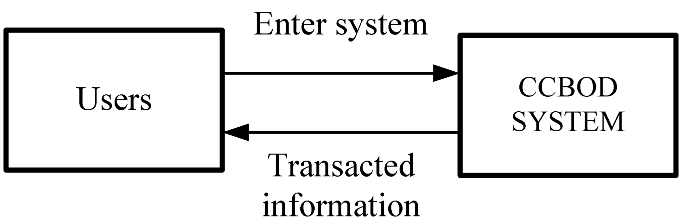


Figure 3.9. DFD level 0

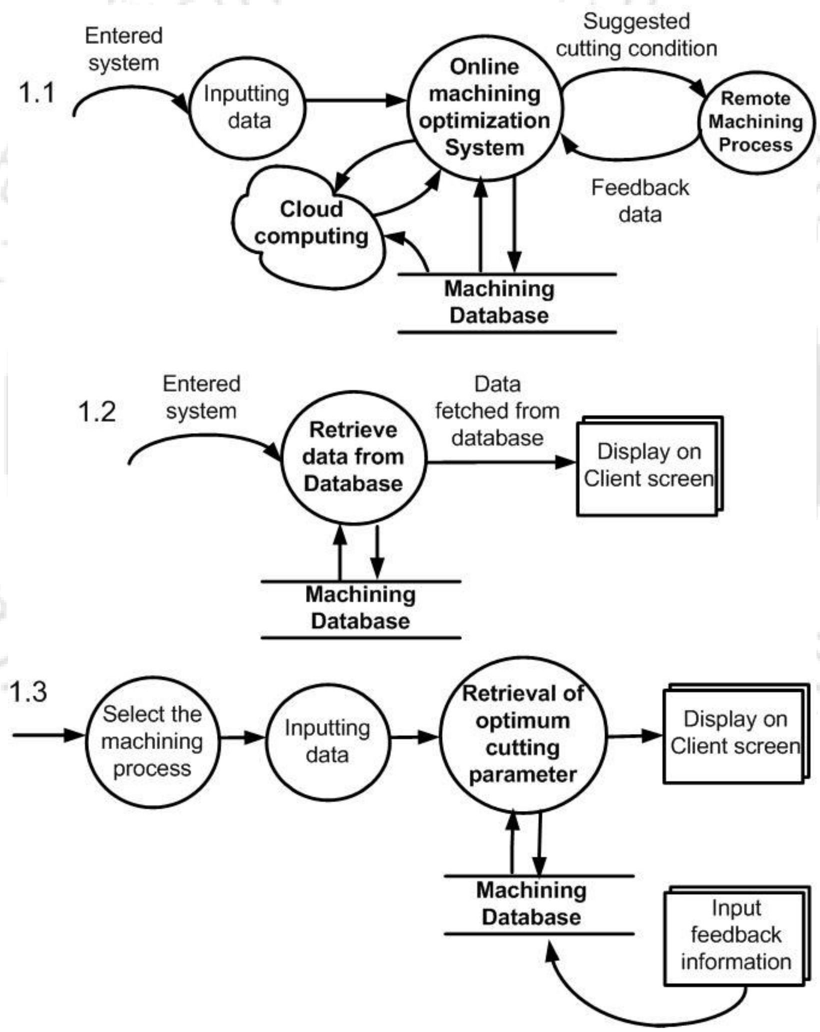


Figure 3.10. DFD level 1

The next level is level 1. Figure 3.10 shows the data flow diagram of the level 1 and it has three main functions. They are i) online machining optimization system,

ii) retrieval of database information, and iii) retrieval of optimum cutting parameters and user feedback about the same. In online machining optimization the user enters the system and inputs machining informations of the process to be optimized. The system suggests initial cutting conditions (cutting speed, feed, and depth of cut) to be used. The user implements the data on machining and obtains actual output data *viz*., actual surface roughness and tool life. The system accepts the machining feed back and suggests improved cutting conditions, thus optimizing process online through interactive session between user and the system. The procedure is similar to the common practice followed by machine tool operator who fine tunes the cutting parameters based on his experience as well as observation of machining performance. The data obtained by real machining process is stored in a database called 'machining database'.

The second function is retrieval of information from database. On specifying the name of the process the real machining information stored on particular database can be retrieved by the user. The information is displayed on the user screen and can be downloaded and saved into the file. The third function is retrieval of optimum cutting parameters. On selecting the process name (*i.e*., turning, end milling, face milling, and grinding) and entering data (*i.e*., total depth of stock to be removed, maximum allowable surface roughness), the system will fetch the optimum process parameters for rough and finish machining processes. The information will be displayed on user screen and user can implement the data on machining. The system also accepts the machining feed back and stores in separate database tables. The machining data obtained from different users/machine tool operators stored in the database helps to develop predictive model and understand probabilistic nature of the machining process. Based on data received from different clients the CCBOD machining database update its informations through cloud computing feature performed at server.

3.5.4 Obtaining the Optimized Parameters from Cloud

The CCBOD system has a feature for retrieval of the data values fetched from respective database and displayed at client's system. The system also accepts client's inquiry and provides updated database information through server calculation and display optimum cutting parameters for specific machining process. The HTML page will be displayed based on user option on the main page for inputting the information such as total depth of stock to be removed and maximum allowable surface roughness. The web server executes the entered information and retrieves data and submits the result to the browser in the form of HTML, which appears in the client's system screen. The database is maintained by administrator for necessary updating and deletion of the database information. It is carried out at web server using structured query language (SQL) statement.

Figure 3.11 shows the coordinated working pages and HTML frame sheet of the developed system. The system manages the process through the front-end screens made up of HTML, cascading style sheets (CSS), Javascript and PHP codings. This provides an interface between the database and users to perform various functions.

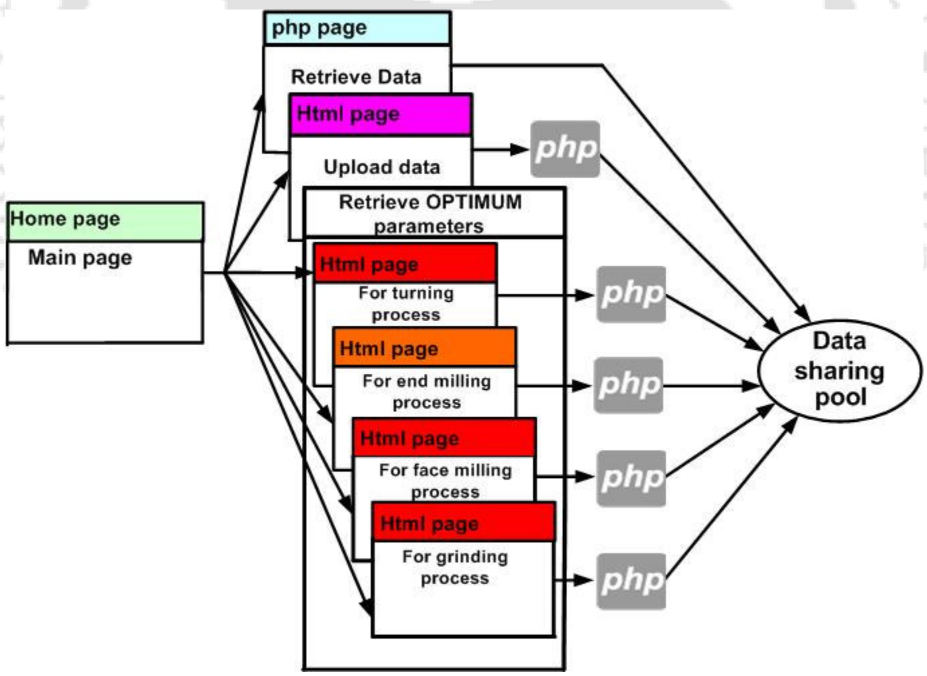


Figure 3.11. Functional working pages of CCBOD system

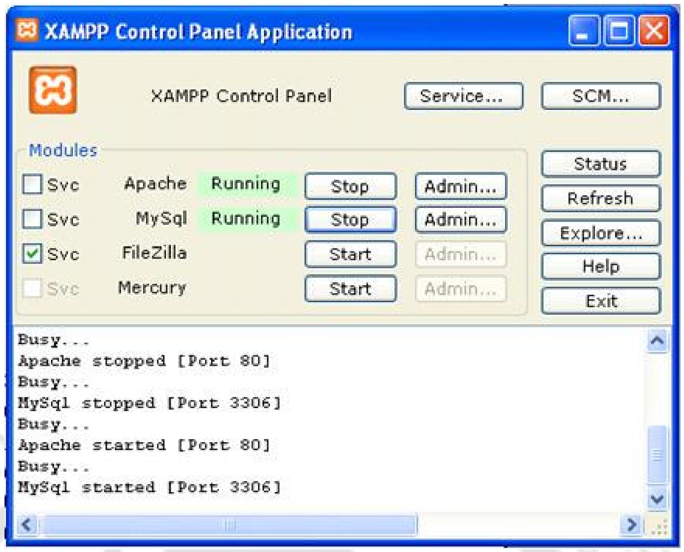


Figure 3.12. XAMPP control panel

The server uses XAMPP, which is a cross-platform software consisting of the Apache web server, MySQL, PHP and Perl. Figure 3.12 shows typical screen shot of the XAMPP control panel application in the process.

3.5.5 Probabilistic Approach for Obtaining Optimum Parameter

The optimum process parameters are evaluated by the system for total depth of stock to be removed and maximum allowable surface roughness. It uses real machining data sets provided by different clients through online optimization. These informations are stored in CCBOD database and different clients may access/retrieve the optimum process parameters. Also the clients may implement these parameters for machining and submit the feed back informations such as actual surface roughness and tool life. Consider the feed back received for tool life from different clients. Based on feed back the optimization parameters have to be found with probabilistic input.

In this work a probabilistic approach of evaluating optimum process parameter is proposed. Let the tool life is the function of process parameters as $T(f, v)$. For the given tool life function as $T(f, v)$, the cost per piece is $C(f, v)$, a function of feed f and cutting speed v. Consider n $(= n_1+n_2)$ number of machine tool operators predicts tool life for optimum cutting parameters.

The tool life function predicted by n_1 machines = $T_1(f, v)$

The probability of tool life function $p_1 = n_1/n$

The tool life function predicted by n_2 machines = $T_2(f, v)$

The probability of tool life function $p_2 = n_2/n$

The cost per piece by n_1 machines that follows tool life function as $T_1(f, v) = C_1(f, v)$

The cost per piece by n_2 machines that follows tool life function as $T_2(f, v) = C_2(f, v)$

Thus, based on the mathematical expectation, the expression for the cost per piece is given as

$$C(f,v) = p_1 C_1(f,v) + p_2 C_2(f,v) \tag{3.10}$$

The above expression can be minimized to obtain the minimum cost of machining.

Once sufficient amount of data is available, the discrete probabilities can be replaced by the probability distribution of tool life. Let $p(f, v)$ be the probability distribution of tool life and $C_T\left(f,v,T(f,v)\right)$ denotes the cost per piece function for a tool life T. Based on mathematical expectation, the cost per piece is given as

$$C(f,v) = \int_{T_{\min}(f,v)}^{T_{\max}(f,v)} C_T\left(f,v,T(f,v)\right) p(f,v)\mathrm{d}T \tag{3.11}$$

where $T_{\min}(f, v)$ and $T_{\max}(f, v)$ are the minimum at a maximum tool life at process parameters f and v. The above expression can be minimized by any optimization method. In this work, because of scarcity of data and to preserve simplicity, the discrete model has been used. The approach is demonstrated with examples in Section 5.5.

3.5.6 Online Fine Tuning and Sending Feedback

The adoption of cloud computing feature helps to optimize different machining processes. An online optimization methodology is accessed *via* web/internet or intranet technologies and applied to turning, end milling, face milling and grinding processes. The machining feedback is sent to the server through clouds. The real machining data provided by different clients/users are stored in database server. The information is fine tuned and provides useful data to users through cloud services.

The clients at different locations access online optimization system through web browsers and operate at their locations for optimization of machining process.

The presentation layer, *viz.*, GUI facilitates easy interaction with the system by the machine tool operator. On entering basic informations such as name of the machining process, tool and work material, work or cutter diameter, length of machining, total depth of stock to be removed, maximum allowable cutting force, maximum allowable surface roughness and limits of cutting parameters, the system suggests the suitable initial cutting conditions for rough machining. The result of machining process such as actual cutting force and surface roughness are fed back into the system. Based on this information improved cutting conditions are suggested and the process is optimized for roughing as well as finishing process during production of components.

In optimizing multipass machining process the cutting parameters for rough and finish machining process are optimized. On rough machining at selected cutting condition, if the actual cutting force is within the maximum allowable value, machining is continued to produce components and tool life data is obtained. During finish machining the process is continued if the selected cutting condition satisfies cutting force constraint and the actual surface roughness falls within the limits. The data cloud obtained for different cutting conditions and from different machine tool operators helps to develop predictive tool life model which is used for optimization. Considering tool life variation due to its probabilistic nature the production cost per piece is evaluated. The system suggests improved cutting conditions and obtains feedback. The process is optimized at the server using suitable optimization technique. The user can access the information for necessary implementation at their location.

3.6 Summary

In this Chapter, an overview of the proposed cloud computing based optimization is discussed. Brief background of cloud computing system describing various deployment models, cloud security and advantages and disadvantages are presented. Next, the background of machining optimization for optimizing the process parameters employing offline and online procedure is discussed. The basics for development of menu based interactive system for implementation of online optimization system are described. In an interactive machining the machine tool operator/user interacts with a menu driven system for optimizing the machining process in online and also send their feedbacks.

Next, the developed web based CCBOD system is described. The web implementation and overall communication structure among various components is presented. The different components of CCBOD *viz.* web server, the online optimization system, the machining database, retrieval and sharing of information and the clients are described. The developed CCBOD is implemented with client-server architecture. The graphical representation of data flow diagram of CCBOD is also presented. The system has a feature for retrieval of the data values fetched from respective database and displayed at client's system. The different working pages and HTML frame sheet of the developed system is presented. It provides an interface to users through the front-end screens made up of HTML, cascading style sheets (CSS), Javascript and PHP codings.

Based on feed back the optimization parameters received from different clients have found with probabilistic input. Therefore a new probabilistic based approach is proposed for evaluating the optimum process parameters. The methodology of online fine tuning of process parameters and sending feedback is discussed. The real machining data provided by different clients/users are stored in database server. The information is fine tuned and provides useful data to users through cloud services.

Chapter 4

Heuristic Based Online Optimization with Cloud

4.1 Introduction

The optimization of machining process aims to obtain optimum cutting parameters for economic manufacturing of the product. Researchers have used a number of traditional and non-traditional optimization techniques for obtaining optimum parameters in single and multipass machining processes. The results of conventional offline optimization have many limitations and need to be fine tuned in the shop floor depending on the rigidity of machine tool, deviation in the properties of material and a number of random causes. In this thesis, a heuristic based online optimization methodology has been developed that optimizes the process during production of jobs. The heuristic knowledge can efficiently control non-linear optimization procedure for obtaining optimum cutting parameters. The strategy does not require *a priori* machining models. It obtains reliable optimum cutting parameters by its direct implementation. The procedure may be implemented in any remotely connected machine tool *via* internet for optimizing machining process.

A virtual machine is designed to test the effectiveness of the optimization model and minimize the number of physical experiments. For given values of input parameters the virtual machine produces the actual outputs. The developed online optimization procedure is demonstrated for optimizing finish turning process. A number of test problems are solved. The procedure is also applied for optimizing multipass turning process. The optimum cutting velocity and feed are obtained for roughing and finishing processes. Illustrative examples are presented.

In this Chapter, Section 4.2 describes the design of a virtual machine. In Section 4.3, a heuristic online machining optimization for cloud computing system is presented. The optimization procedure is demonstrated for optimization of finish and multipass turning process. Illustrative examples for finish and multipass turning process are presented in Section 4.4. The summary of the Chapter is presented in Section 4.5

4.2 Design of Virtual Machine

A virtual machine is a simulation of a real machine. It is used to carry out the machining experiments through simulated results. This reduces the number of physical experiments by saving time needed for conducting experiments as well as material cost. For given inputs the virtual machine produces the actual outputs. In this work, semi-virtual lathe is designed to test the effectiveness of the optimization model. It consists of three modules: (i) Surface roughness prediction module, (ii) Tool life prediction module and (iii) Cutting force prediction module. Figure 4.1 shows block diagram of a semi-virtual lathe.

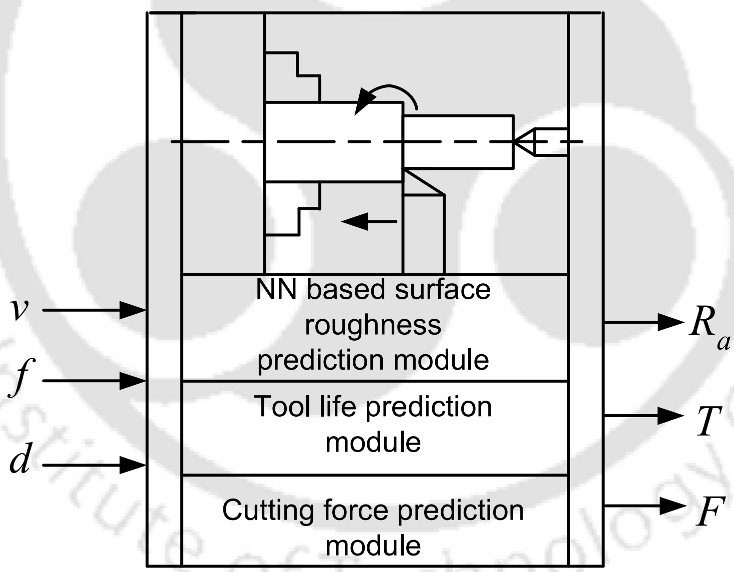


Figure 4.1. Semi-virtual lathe

For a given process parameters *viz.*, cutting speed (v), feed (f) and depth of cut (d), the semi-virtual lathe provides centre line average (CLA) value of surface roughness (R_a), tool life (T) and main cutting force (F). The surface roughness module is developed using neural network (NN) modelling. The tool life and cutting force prediction modules use empirical relation for evaluation of tool life and cutting force. These are discussed in detail in the following subsections.

4.2.1 Surface Roughness Prediction Module Based on NN

Surface finish is the one of the main attribute of the job quality. The surface quality of the manufactured products is mainly accessed by surface roughness value which is expressed as the irregularities of machined surface quantified by CLA value of surface roughness. It is represented by 'R_a'. For prediction of surface roughness researchers found neural network modelling as more promising technique because of its ability to model complex processes (Rangwala and Dornfeld, 1989; Azouzi and Guillot, 1998; Kohli and Dixit, 2005). It provides reasonable accuracy in prediction and requires lesser computational time.

In this work, multilayer perceptron (MLP) network model that works on backpropagation algorithm is used as a part of virtual machine for the prediction of surface roughness. The experimental data are selected from the work of Kohli and Dixit (2005) for dry turning of steel using carbide tools. The cutting speed, feed and depth of cut are considered as three independent variables on which the surface roughness depends. Therefore, three input neurons of a neural network are cutting speed (v), feed (f) and depth of cut (d). The network required 21 data sets for training and 9 for testing as shown in Table 4.1 and 4.2 respectively. The selected data sets are in the range of cutting parameters as cutting speed: 140–240 m/min; feed: 0.04– 0.32 mm/rev; and depth of cut: 0.01–1.0 mm.

The output layer has a single neuron corresponding to the predicted value of surface roughness. For computation of weights, '*logsig*' activation function is used for neurons in hidden layer, whereas '*purelin*' is used for output neurons. For training the network, the '*trainlm*' function of MATLAB® 7.1 that works on back propagation algorithm is used. The performance of the network prediction is assessed by using root mean square fractional error as given by

$$RMS_{err}^{f} = \sqrt{\frac{\sum (R_a - \hat{R}_a)^2}{nR_a^2}} \;, \tag{4.1}$$

where R_a is the measured CLA value of surface roughness $\hat{R}_a$ is the predicted value.

Table 4.1. Training data set in dry turning by Carbide tool

Serial number	Cutting speed, v (m/min)	Feed, f (mm/rev)	Depth of cut, d (mm)	Surface roughness, R_a (μm)
1	235.84	0.04	0.1	2.50
2	234.56	0.08	0.1	2.29
3	197.92	0.16	0.1	2.15
4	188.52	0.28	0.1	3.81
5	226.87	0.32	0.1	3.28
6	236.51	0.04	0.3	1.90
7	150.33	0.08	0.3	2.27
8	197.92	0.12	0.3	2.22
9	221.67	0.24	0.3	2.50
10	212.76	0.32	0.3	3.26
11	140.27	0.24	0.3	2.84
12	203.36	0.08	0.6	2.23
13	227.61	0.12	0.6	2.64
14	197.42	0.12	0.6	2.87
15	221.67	0.16	0.6	2.33
16	215.72	0.24	0.6	1.85
17	179.61	0.24	0.6	2.02
18	203.86	0.28	0.6	2.79
19	236.48	0.04	1.0	2.43
20	212.27	0.12	1.0	2.35
21	238.41	0.32	1.0	2.43

The root mean square (RMS) error is calculated separately for both training and testing for various network topologies by varying number of hidden layers and number of neurons in each hidden layers. A neural network topology that has least effective error (*i.e.*, maximum of training and testing RMS fractional error) is considered as best network topology. In this work, the neural network topology with one hidden layer with five neurons has the least effective error and is considered as best network architecture. The training performance of the network shows the percentage of error to be within 20% limits and maximum error recorded is 17.5%. The network is validated using 18 data sets and results obtained are within 20% limits. Figure 4.2 shows the architecture of the proposed NN model for prediction of surface roughness.

Table 4.2. Testing data set in dry turning by Carbide tool

Serial number	Cutting speed, *v* (m/min)	Feed, *f* (mm/rev)	Depth of cut, *d* (mm)	Surface roughness, R_a (µm)
1	201.88	0.08	0.1	2.59
2	157.37	0.28	0.1	4.42
3	184.06	0.30	0.1	4.70
4	233.55	0.08	0.3	1.97
5	144.83	0.16	0.3	2.52
6	223.55	0.08	0.6	2.41
7	146.93	0.04	0.6	3.04
8	197.92	0.32	0.6	2.62
9	223.67	0.08	1.0	2.65

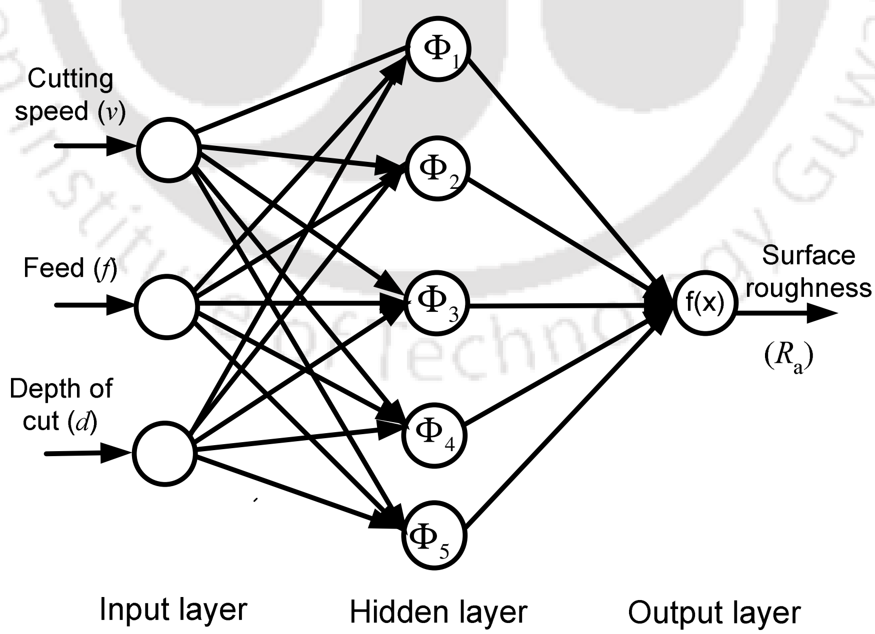


Figure 4.2. A typical MLP neural network architecture

4.2.2 Tool Life Prediction Module

In this module the evaluation of tool life is obtained by using empirical relation based on Taylor's extended tool life equation and it is given as

$$vT^{0.2} f^{0.15} d^{0.15} = 273, \tag{4.2}$$

where v is in m/min, tool life T is in minute, f is in mm/rev and d is in mm. The module doest not employ NN model since the NN modelling requires number of experimental data sets. Conducting offline experiments for tool life tests are expensive and therefore an empirical relation is used. Note that this equation is used to develop a black box module of tool life. This equation is not used anywhere in the optimization procedure, assuming that tool life can be found only experimentally.

4.2.3 Cutting Force Prediction Module

In this module the evaluation of main cutting force is obtained by using empirical relation (Ojha *et al.*, 2009) and it is given as

$$F' = 858.33 f^{0.248} d^{1.6819}, \tag{4.3}$$

where F' is main cutting force in Newton, f is feed in mm/rev and d is depth of cut in mm. In this module an empirical relation is preferred than NN model since it minimize the experimental effort.

4.3 An Online Machining Optimization for Cloud Computing System

The selection of optimum process parameters is an important step in machining, considering economic aspects. Researchers have investigated single and multipass machining problems mostly for minimizing production cost or time. Apart from various optimization algorithms proposed, the implementation of optimal cutting conditions during machining is very important. However, shop floor applications of optimization methods are limited due to non-availability of required information about the process readily. The approach for online optimization of machining process has not received attention of the researchers.

The most common practice followed in a machine shop is as follows. The operator makes a suitable guess for the process parameters, carries out the machining and fine tunes the process parameters by observing the machining performance. In the process, operator's knowledge about the process gets strengthened that helps him/her in arriving at the better guess in the next situation. Very recently, the concept of cloud computing and cloud manufacturing has started gaining popularity (Marston *et al*., 2011; Xu, 2012). Similar to cloud computing, cloud manufacturing aims to share manufacturing resources to make "manufacturing as service" environment. In this work a cloud computing based online optimization of machining system is developed. The methodology mimics the behaviour of an operator for obtaining optimum process parameters. For optimizing the finish turning process the machining parameters are first selected so as to keep the surface finish of the component produced well within the specified limits. The modification of cutting condition in machining of subsequent job is carried out in order to optimize the process. Minimizing the machining cost is considered as optimization objective and a well-known Simplex search method (Rao, 1999) is used as optimization technique. The heuristic algorithm for online optimization of machining process is presented here.

4.3.1 A Heuristic Algorithm

In the proposed scheme the optimization is carried out during actual machining by selecting the cutting conditions suitably. The scheme selects initial cutting condition and modifies it suitably based on feed back information from real machining environment. These data sets are used to optimize the cutting condition. The algorithm procedure is described in the context of turning process.

For example, consider the optimization of finish turning process for obtaining the maximum allowable surface roughness (R_a^*). Let $\hat{R}_a$ be the actual (achieved) surface roughness. Often researchers consider the surface roughness constraints as $\hat{R}_a \leq R_a^*$. However, the actual surface roughness is desired between upper and lower limits of the surface roughness. It is required to obtain definite tribological characteristics, heat transfer, *etc*. (Ojha *et al*., 2009). In this work, the actual surface roughness is expected to produce between the lower and upper limits and is estimated as follows.

Let the target surface roughness be taken as 80% of maximum allowable surface roughness, *i.e.*, $R_a^t = 0.8R^*$. With the inherent randomness in metal cutting process Risbood *et al.* (2003) considered a maximum of 20% error as reasonable in prediction of surface roughness. Incorporating some statistical variation in surface roughness, let $\hat{R}_a$ should be at 20% deviation from that of target surface roughness *i.e.*, $0.8R_a^t < \hat{R}_a < 1.2R_a^t$. The online optimization strategy begins with selection of initial cutting condition suitably so that there is no loss /rejection of components due to surface roughness in early production. In a finish turning process, as the depth of cut remains same the optimization function is a two variables (v, f) problem. In Simplex search (Appendix A) the function is optimized in an interactive manner with (n+1) initial points (cutting conditions) in an n-dimensional space. For the present problem three initial machining conditions are generated. Other two machining conditions are selected with 10% deviation, being not much away from the previous cutting condition. With these three set of cutting conditions and tool life data the function values are evaluated and optimization is carried out for obtaining optimum process parameters. The online optimization strategy consists of the following steps:

Step 1: Select the first set of cutting condition.

Set initial cutting velocity, v_1= maximum available cutting velocity, v_{max}.

Set initial feed f_1= minimum allowable feed, f_{min}.

Set i = 0;

Step 2: Perform machining with selected values of cutting conditions for a given tool-work piece combination.

Step 3: Measure the surface roughness of the finished component. Depending on the value of the surface roughness, three cases may arise:

Case 1: If $0.8R_a^t < \hat{R}_a < 1.2R_a^t$, continue machining at this setting till tool fails and note down the tool life. Increment i by 1.

If 'i = 1', go to Step 4 for computing second set of conditions.

If 'i = 2', go to Step 5 for computing third set of conditions.

If 'i = 3', go to Step 6 for initiating optimization method.

Case 2: If $\hat{R}_a > 1.2R_a^t$, give a message "Surface roughness cannot be achieved with this set of tool-work piece combination". Since the process is initiated with $v_{max.}$ and f_{min} the achieved surface roughness is expected to be the minimum. But, if the achieved surface roughness value exceeds the upper limit, which is 96% of desired surface roughness value the process will soon be out of control. Therefore machining should be stopped.

Case 3: If $0.8R_a^t > \hat{R}_a$, increase the feed using the following equation:

$$(f_1)_{\text{new}} = \left(\frac{R_a^t}{\hat{R}_a}\right)^{1.5} f_1, \tag{4.4}$$

where f_1 is feed in mm/rev. Go to Step 2.

The exponent 1.5 has been chosen so that the new feed will produce the components within the control limit. The exponent 1.5 is decided based on the experimental observations.

Step 4: Select second set of cutting condition such that v_2= 0.9 v_1 and f_2= 1.1 $f_{1.}$ Go to Step 2.

Step 5: Select third set of cutting condition such that v_3= 0.9 v_2 and f_3= f_2. Go to Step 2.

Step 6: The three set of cutting conditions obtained from the previous steps is used to evaluate the function values may be used to initiate Simplex search. to determine optimum cutting velocity and feed values. The flow chart of the online optimization procedure is shown in Figure 4.3.

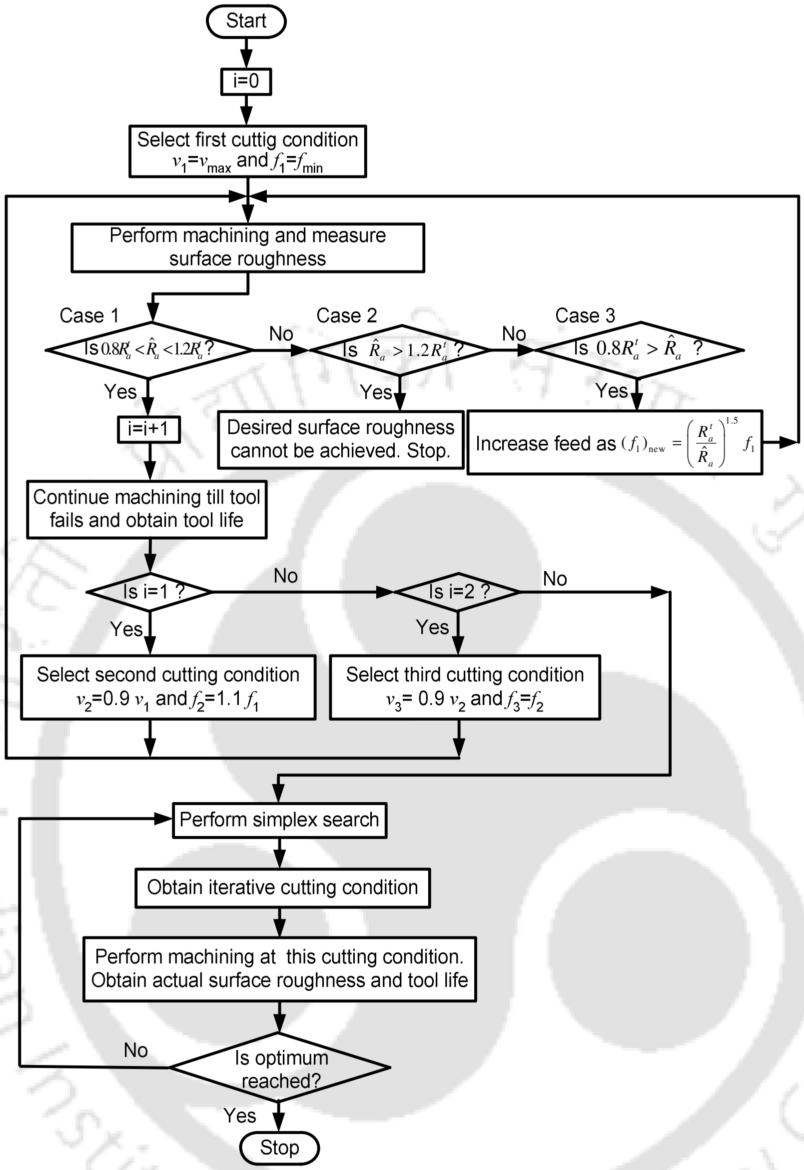


Figure 4.3. Flowchart for online optimization finish machining

The proposed scheme of online machining optimization obtains reliable optimum cutting parameters by its direct implementation**.** It has potential to consider statistical variation of surface roughness in prediction and optimization of the machining process. The actual surface roughness ($\hat{R}_a$) is aimed between upper and lower limits of the target surface roughness (R_a^t) value instead of maximum allowable value of surface roughness (R_a^*). It is not considered in most of the offline optimization techniques available in the literature. The online optimization procedure does not employ any empirical relation for evaluation of tool life and surface roughness and it obtains reliable optimum cutting parameters by its direct implementation. Further it generates reliable data sets during the process of optimization and which can be used for many other useful purposes.

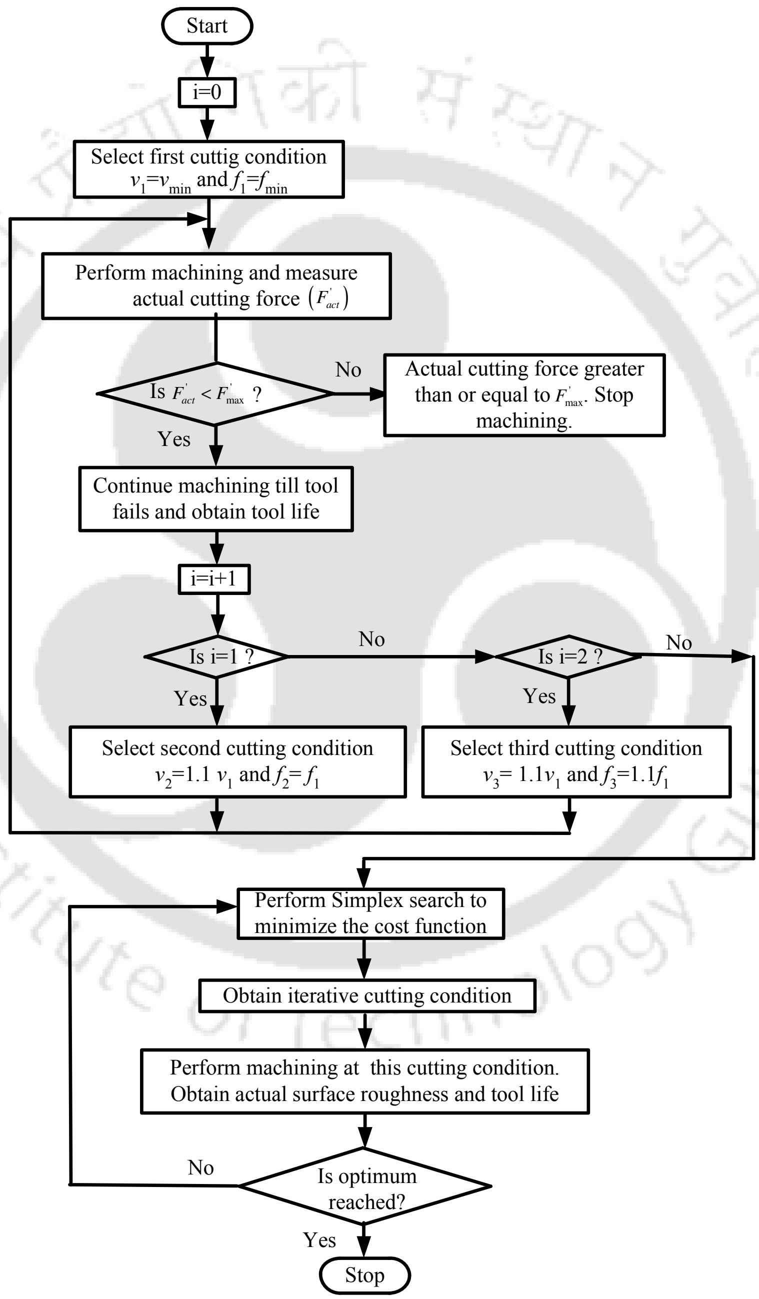


Figure 4.4. Flowchart for online optimization rough machining

Figure 4.4 shows the methodology for online optimization of rough machining process. Three initial cutting conditions *viz*., ($v_1 = v_{min}$, $f_1 = f_{min}$), ($v_2 = 1.1v_1$, $f_2 = f_1$) and ($v_3 = 1.1v_1$, $f_3 = f_1$) are considered and machining is carried out. If actual cutting force obtained is less than F_{max} machining is continued till tool fails. Tool life is obtained and function values are evaluated. Simplex search is now initiated to obtain optimum cutting parameter (v_{opt}, f_{opt}) for rough machining.

4.3.2 Objective Function for Finish Machining

In finish turning process two cutting parameters *i.e.*, cutting speed and feed are optimized while depth of cut remains constant. The optimization objective is to minimize the machining cost satisfying the maximum allowable surface roughness.

In finish turning, the machining time (T_p) for producing a component (Kilic, 1987) is given by

$$T_p = \frac{\pi LD}{fv}\left(1 + \frac{t_c}{T}\right), \tag{4.5}$$

where L is the machining length in m, D is the diameter of the work piece in mm, t_c is the tool change time in minute and T is the tool life for particular cutting condition in minute. The tool life is a function of cutting velocity, feed and depth of cut.

If C_0 is the operating cost in \$/min and C_{tc} is the tool change cost in \$, then machining cost per piece (C) is given by

$$C = C_o T_p + C_{tc}\left(\frac{\pi LD}{fvT}\right) \tag{4.6}$$

Substituting the value of T_p from Eq. (4.7)

$$C = C_0 \frac{\pi LD}{fv}\left(1 + \frac{t_c}{T}\right) + C_{tc}\left(\frac{\pi LD}{fvT}\right)$$

$$= C_0 \frac{\pi LD}{fv}\left(1 + \frac{t_c^*}{T}\right), \tag{4.7}$$

where $t_c^* = t_c + C_{tc}/C_0$. With the constant values of costs, length and diameter, the objective function to minimize machining cost is expressed by the function F as

$$\text{Minimize } F = \frac{1}{fv}\left(1+\frac{t_c^*}{T}\right). \tag{4.8}$$

The cost function given in Eq. (4.8) is optimized satisfying the desired surface roughness value. The function reduces to two-variables (f and v) problem and it has the units of min/mm^2. Simplex search may be used to compare the values of the objective function.

4.3.3 Objective Function for Multipass Machining

The proposed heuristic online optimization is also applied for multipass machining process. In manufacturing industries, multipass machining is widely used than single pass. In a multipass machining process the total depth of stock is removed by number of roughing passes and a final finish pass. The highest possible metal removal is aimed in rough passes, where surface finish is not an important consideration. Cutting force developed should be within maximum allowable value. In finish turning process, surface finish is the most important consideration.

The optimization objective is to minimize the cost. Total machining time for producing a component (T_p) is sum of (i) total machining time for roughing passes (T_{tR}), (ii) machining time for finish pass (T_{tF}) and (iii) loading and unloading time (T_L) and (iv) total tool reset time (T_{ts}). It is given by

$$T_p = T_{tR} + T_{tF} + T_L + T_{ts} \tag{4.9}$$

Equal depth of cut is considered for all rough passes. The total machining time for roughing and finishing pass is given by (Ojha *et al.*, 2009)

$$T_{tR} = \left(1+\frac{t_c}{T_R}\right)\frac{\pi L}{v_R f_R}(mD_0 - m(m-1)d_R) \tag{4.10}$$

$$T_{tF} = \frac{\pi L}{v_F f_F}(D_f + 2d_F), \tag{4.11}$$

where D_0 and D_f are initial and final diameter of work piece in mm, v_R, f_R, d_R, m and T_R are cutting speed in m/min, feed in mm, depth of cut/pass in mm, number of passes, tool life in minute for roughing passes and v_F, f_F, d_F and T_F are cutting speed in m/min, feed in mm, depth of cut in mm, tool life in minute for finishing pass. The total tool resetting time is given by

$$T_{ts} = (m+1)t_s, \tag{4.12}$$

where t_s is the tool reset time/pass. Hence, total machining time per component is obtained as

$$T_{\mathrm{p}} = \left(1+\frac{t_c}{T_R}\right)\frac{\pi L}{v_R f_R}(mD_0 - m(m-1)d_R) + \left(1+\frac{t_c}{T_F}\right)\frac{\pi L}{v_F f_F}(D_f + 2d_F) + T_L + T_{ts}. \tag{4.13}$$

The total production cost is given by

$$C = C_0 T_P + C_{tc}\left(\frac{T_{tR}}{T_R} + \frac{T_{tF}}{T_F}\right), \tag{4.14}$$

where C_0 is the operating cost in \$/min and C_{tc} is the tool change cost in \$/cutting edge. Substituting the value of T_p in Eq. (4.14) the total production cost being the summation of cost for roughing passes, cost for finishing pass and cost involved for loading and unloading is given by

$$C = C_R + C_F + C_L$$

$$= \left[C_0\frac{\pi L}{v_R f_R}(mD_0 - m(m-1)d_R) + (C_0 t_c + C_{tc})\frac{\pi L}{v_R f_R T_R}(mD_0 - m(m-1)d_R) + C_0(m+1)t_s\right]$$

$$+\left[C_0\frac{\pi L}{v_F f_F}(D_f + 2d_F) + (C_0 t_c + C_{tc})\frac{\pi L(D_f + 2d_F)}{v_F f_F T_F}\right] + C_0 T_L. \tag{4.15}$$

The cutting and feed for roughing passes (v_R and f_R) are obtained by minimizing the function C_R, given as

$$C_R = \left[C_0\frac{\pi L}{v_R f_R}(mD_0 - m(m-1)d_R) + (C_0 t_c + C_{tc})\frac{\pi L}{v_R f_R T_R}(mD_0 - m(m-1)d_R) + C_0(m+1)t_s\right]. \tag{4.16}$$

Finish turning parameters (v_F and f_F) are obtained by minimizing the function C_F, given as

$$C_F = \left[C_0 \frac{\pi L}{v_F f_F}(D_f + 2d_F) + (C_0 t_c + C_{tc}) \frac{\pi L(D_f + 2d_F)}{v_F f_F T_F} \right]. \qquad (4.17)$$

4.4 Illustrative Examples: Online Optimization of Machining Process

In manufacturing industries, multipass machining is widely used than single-pass or finish machining. The highest possible metal removal is aimed in rough passes, where surface finish is not an important consideration. However, in finish machining process, surface finish is the most important consideration. The proposed online procedure is explained with illustrative examples for finish as well as multipass turning process.

4.4.1 Optimization of Finish Turning

In this section, the proposed methodology is explained with the help of a suitable example. Let us assume that optimum process parameters for a finish turning process are to be found for a depth of cut of 0.3 mm and allowable surface roughness value (R_a^*) is 3.25 µm. The target surface roughness (R_a^t) is evaluated as 2.60 µm. The actual surface roughness may lie between 2.08 and 3.12 µm. The algorithm initializes the cutting conditions, v_1 (= v_{max}) as 238 m/min and f_1 (= f_{min}) as 0.04 mm/rev. The machining of component is performed and surface roughness is measured as explained in Step 2 and Step 3 of the methodology in Section 4.3.1. First set of cutting condition reaches when v =238 m/min and f = 0.12 mm; the CLA surface roughness was 2.26 µm. Now machining is continued till tool fails. The tool life obtained is 24 min. The remaining steps of the methodology are followed to obtain other two cutting conditions. The other two cutting conditions are (214, 0.13) and (193, 0.13) and tool lives are 38 and 64.5 min respectively. Table 4.3 shows the three set of cutting conditions and surface roughness values obtained in this process.

For all three initial cutting conditions the function values are evaluated using Eq. (4.8). With the operating cost, C_0 as \$0.5/min, the tool change cost, C_{tc} as \$2.5/cutting edge and tool change time, t_c as 1.5 min (Gupta *et al.*, 1995; Ojha *et al.*, 2009), $t_c^* = 1.5 + \frac{2.5}{0.5} = 6.5\,\text{min}$. The function values are

$$F_{(238,0.12)} = \frac{1}{238\times10^3\times0.12}\left(1+\frac{6.5}{24}\right) = 4.45\times10^{-5}\ \text{min/mm}^2$$

$$F_{(214,0.13)} = \frac{1}{214\times10^3\times0.13}\left(1+\frac{6.5}{38}\right) = 4.21\times10^{-5}\ \text{min/mm}^2$$

$$F_{(193,0.13)} = \frac{1}{193\times10^3\times0.13}\left(1+\frac{10}{64.5}\right) = 4.34\times10^{-5}\ \text{min/mm}^2.$$

With the help of three cutting conditions and corresponding function values as $(238, 0.12, 4.45\times10^{-5})$, $(214, 0.13, 4.21\times10^{-5})$ and $(193, 0.13, 4.34\times10^{-5})$, the process is minimized using Simplex search algorithm.

Table 4.3. Results of different cutting conditions

S.No.	Cutting parameters (*v* *d* *f*) (m/min mm mm/rev)	Surface roughness obtained($\hat{R}_a$)(µm)	Job accepted /rejected	Remarks
1	238 0.3 0.04	2.05	Accepted	Only one job is machined
2	238 0.3 0.06	2.03	Accepted	–do–
3	238 0.3 0.08	1.98	Accepted	–do–
4	238 0.3 0.12	2.26	Accepted	First set of cutting condition reached, machine a number of jobs till tool fails.
5	214 0.3 0.13	2.21	Accepted	Second set of cutting condition reached, machine a number of jobs till tool fails.
6	193 0.3 0.13	2.14	Accepted	Third set of cutting condition reached, machine a number of jobs till tool fails.

The optimum process parameters are obtained after six iterations as v_{opt} = 188 m/min and f_{opt} = 0.24 mm. The function value reached is 2.69×10^{-5} min/mm^2. At this optimum condition, the surface roughness obtained is 2.54 µm which is well within the specified limits. Iteration wise results are presented in Table 4.4. It may also be noted that the values of surface roughness obtained in all iterations are within the specified limits. By machining continuously at every cutting condition till tool fails, tool life data is also obtained through online machining and it does not require any separate off-line experimentation.

Table 4.4. Results of simplex search

New points	New point obtained through Simplex search			Surface roughness obtained ($\hat{R}_a$) (μm)	Function value (F) ($\times 10^{-5}$ min/mm^2)
	(v (m/min	d mm	f) mm/rev)		
1	186	0.3	0.14	2.22	4.18
2	215	0.3	0.12	2.10	4.06
3	178	0.3	0.15	2.29	4.03
4	207	0.3	0.16	2.06	3.52
5	156	0.3	0.20	2.04	3.36
6	188	0.3	0.24	2.54	2.53

Taking D as 30 mm and L as 100 mm, the cost of machining at optimum cutting condition is evaluated as

$$C_{(188,0.24)} = \frac{0.5\times\pi\times0.03\times100}{188\times0.24}\left(1+\frac{6.5}{46}\right) = \$0.1191.$$

A number of other problems having depth of cut varying from 0.1 mm to 1.0 mm for different values of maximum allowable surface roughness are tested. The optimum process parameters (v_{opt} and f_{opt}) and function values ($F_{\min}$) calculated are given in Table 4.5. The surface roughness values for all the cases are within the desired limits and the cutting conditions do not deviate significantly from the optimum values.

Table 4.5. Results of different problems

Problem number	Data set (d (mm	R_a^*) µm)	Simplex optimum (v_{opt} (m/min	f_{opt} mm/rev	$\hat{R}_a$ µm	F_{min}) $\times 10^{-5}$ min/mm^2)	Number of iterations
1	(0.1	3.25)	173	0.18	2.33	3.32	11
2	(0.3	3.00)	172	0.19	2.38	3.29	12
3	(0.4	3.25)	175	0.22	2.34	2.89	7
4	(0.5	3.60)	188	0.28	3.40	2.34	11
5	(0.6	3.75)	208	0.31	2.72	2.28	10
6	(0.7	3.00)	221	0.06	2.08	9.12	5
7	(0.8	3.75)	152	0.28	3.48	2.62	12
8	(0.9	2.50)	194	0.13	2.31	4.90	7
9	(0.9	3.75)	152	0.24	2.46	3.04	12
10	(1.0	3.50)	194	0.23	2.94	2.40	9

The economic analysis of the illustrated example is presented here. For machining 30 mm diameter job for 100 mm long the cost of machining at different cutting condition is evaluated. The results are given in Table 4.6. In the proposed methodology, for the illustrated example in Section 4.4.1 the optimum cutting condition is reached after producing 1739 components. However, it may vary depending upon the produced surface roughness and tool life under selected cutting condition. The surface roughness of the machined jobs is well within the maximum specified limit of 3.12 µm and there is no rejection. In the present case, the optimum cutting condition (v_{opt} and f_{opt}) obtained is (188 m/min and 0.24 mm). The machining cost for producing one component at optimum cutting condition (C_{opt}) is \$0.1191 (=₹ 7.15; 1\$=₹ 60).

In practice, conducting one tool life test consumes about 20 kg material and 20 hrs of machining (Hegginbotham and Pandey, 1967). The cost involved to conduct one tool life test is approximately \$91.03 (=₹ 5462) (Appendix B).

The cost involved in machining first component at optimum is the sum of the tool life estimation cost and optimum cost of machining (*i.e.*, \$91.03+\$0.1191 =

$91.15). In online machining, the cost involved in machining first job is $0.584. Therefore, it can be said that saving is $90.56. Similarly the saving is evaluated for as machining of jobs goes on and is reported in column number 6 of Table 4.6. In batch production of components machining under various tool-work piece combination/or change of job, the proposed methodology is highly suitable.

Table 4.6. Economic analysis based on Simplex optimization of finish machining

Cutting condit-ion	Data set obtained (v (m/min)	d mm	f mm/rev	T min	$\hat{R}_a$ µm)	Machin-ing time, T_p (min)	Number of jobs turned, $n = T/T_p$	Total machining cost ($) (nC)	Saving ($)
Machining at guess Conditions	238	0.3	0.04	54.3	2.05	0.990	01	0.584	90.56
	238	0.3	0.06	40.4	2.03	0.660	01	0.383	90.30
	238	0.3	0.08	32.6	1.98	0.495	01	0.267	90.16
	238	0.3	0.12	24.1	2.26	0.330	72	15.09	83.64
	214	0.3	0.13	38.5	2.21	0.339	113	22.37	74.73
	193	0.3	0.13	64.5	2.14	0.376	171	35.35	59.65
Machining during Simplex iterations	186	0.3	0.14	73.4	2.22	0.3619	202	39.79	43.92
	174	0.3	0.15	97.3	2.29	0.3611	269	51.81	24.24
	215	0.3	0.14	35.6	2.14	0.3131	113	20.92	16.78
	211	0.3	0.16	35.4	2.08	0.2792	126	20.81	10.98
	156	0.3	0.20	135.4	2.40	0.3021	448	70.92	-6.59
	188	0.3	0.24	46.5	2.54	0.2089	222	26.44	-6.59
						Total	1739		

In conclusion, if a very large number of jobs are to be produced, it is better to conduct the tool life test first and then machining should be carried out at the optimum machining conditions. However, when the batch size is smaller than a threshold value, the proposed algorithm will provide economical solution. The threshold value of batch size can be easily estimated, as the rough estimate of the minimum possible cost of machining can always be made (for example by assuming a very large value of tool life). Also, it is emphasized that the purpose of proposing the present algorithm does not mean that existing knowledge from machining handbooks or shop floor information should not be utilized. This methodology can be considered more as a fine tuning method. In order to show the effectiveness of the methodology, here deliberately large ranges of process parameters have been chosen assuming that little is *a priori* information known. In actual practice, fine tuning will not take much time, with some information already available.

4.4.2 Optimization of Multipass Turning

Consider a multipass turning process to be optimized for obtaining optimum process parameters. Let total depth of stock to be removed is 3.0 mm and allowable surface roughness value (R_a^*) is 3.25 μm. The maximum allowable cutting force (F'_{max}) is 1000 N. The variable bounds are $140 \leq v\,(\text{m / min}) \leq 238$, $0.04 \leq f\,(\text{mm/rev}) \leq 0.32$, $0.1 \leq d\,(\text{mm}) \leq 1.0$.

The online optimization of roughing process starts with selection of initial cutting condition suitably so that actual cutting force is within limit. The depth of cut in all roughing process is taken as equal. The number of passes is decided such that the maximum depth of cut constraint is not violated. In the present problem depth of cut for finishing and two roughing passes comes out as 1.0 mm. As the optimization function is a two variable (v, f) problem, three initial machining conditions are generated and optimized using Simplex search method. There are many non-conventional optimization methods proposed in the literature. However in this work, simplex search method is used to optimize the parameters. To initiate optimization, three cutting conditions are selected as (v_1, f_1), (1.1v_1, f_1) and (1.1v_1, 1.1f_1).

For the present problem initial cutting condition v_1 (= v_{min}) as 140 m/min and f_1 (= f_{min}) 0.04 mm/rev is chosen. Roughing depth of cut/pass (d_R) is 1.0. Number of rough passes (m) is two. The cutting force is obtained as 386.3 N, which is within limit. The machining is continued with first set of cutting condition (v_1 = 140 m/min, f_1 = 0.04 mm/rev, d = 1 mm/pass) till tool fails and obtained tool life 315.2 minute.

Substituting L= 100 mm, D_0=30 mm, C_0= 0.5 \$/min, C_{tc}= 2.5 \$/min, t_c=1.5 min/edge, t_s= 0.51min/pass, Eq. (4.16) rough machining cost is evaluated and it is

$$C_R = \left[\frac{0.5\times\pi\times0.1\times58}{140\times0.04} + (0.5\times1.5+2.5)\frac{\pi\times0.1\times58}{140\times0.04\times315.2} + 0.5\times3\times0.51\right] = \$2.42.$$

The other two cutting conditions are (154, 0.04) and (154, 0.05). The corresponding cost is \$2.28 and \$1.98. The rough machining cost is optimized using Simplex search (Rao, 1999) and optimum is found. The iterative cutting conditions and roughing costs are shown in Table 4.7. The cutting force obtained is within the limit of 1000 N. The optimum process parameters are obtained as v_{opt} = 196.0 m/min and f_{opt} = 0.08 mm/rev. The optimum cost for roughing/piece is \$ 1.43. The proposed scheme obtains reliable optimum cutting parameters by its direct implementation without deviating cutting force constraint.

Table 4.7. Iterative results of optimization for roughing process

New points	New point obtained through Simplex search			Actual cutting force (F) (N)	Roughing cost per piece (C_R) (\$)
	(v_R (m/min	d_R mm	f_R) mm/rev)		
1	182.0	1.0	0.06	427.2	1.67
2	192.0	1.0	0.07	443.8	1.52
3	196.0	1.0	0.08	458.8	1.43

In optimization of finish turning process the initial cutting condition are selected as v_1 (=v_{max}) and f_1 (=f_{min}) and the constant depth of cut is equal to d_{max}. For the present problem v_1, f_1 and d_1 are 238 m/min, 0.04 mm/rev and 1 mm respectively. The allowable surface roughness value is (R_a^*) 3.25 µm. Considering some statistical variation the actual surface roughness ($\hat{R}_a$) is considered between the lower and upper limits and may lie between 2.08 and 3.12 µm. The machining of component is performed and the obtained cutting force and CLA surface roughness are 386.3 N and 2.12 µm respectively. If surface roughness falls below lower limit, feed is increased suitably with v remaining same. Now machining is continued till tool fails and obtained tool life is 22.2 minute. Substituting L= 100 mm, D_f=24 mm, C_0= 0.5 \$/min, C_{tc}= 2.5 \$/min, t_c=1.5 min/edge and t_s= 0.51min/pass in Eq. (4.17) finish machining cost is evaluated and it is

$$C_F = \left[\frac{0.5 \times \pi \times 0.1 \times 26}{238 \times 0.04} + (0.5 \times 1.5 + 2.5) \frac{\pi \times 0.1 \times 26}{238 \times 0.04 \times 22.2} \right] = \$0.56.$$

The other two cutting conditions are considered as (v_2=0.9v_1, f_2=1.1f_1) and (v_3=0.9v_2, f_3=1.1f_1). While machining is performed at (214, 0.05) and (193, 0.05) the cutting force and surface roughness obtained are within the limit. The optimum process parameters are obtained as v_{opt} = 224 m/min and f_{opt} = 0.09 mm/rev. The optimum cost for finishing/piece is \$ 0.28.

Table 4.8. Iterative results of optimization for finishing process

New points	New point obtained through Simplex search			Surface roughness obtained ($\hat{R}_a$)(µm)	Cost per piece (C_F) (\$)
	(v_F (m/min	d_F mm	f_F) mm/rev)		
1	168	1.0	0.06	2.56	0.43
2	184	1.0	0.07	2.52	0.36
3	227	1.0	0.08	2.63	0.31
4	224	1.0	0.09	3.03	0.28

The cost involved for loading and unloading the component, C_T= 0.5x0.75= \$ 0.375. Thus the total production cost per component is \$2.09.The proposed scheme obtains reliable optimum cutting parameters by its direct implementation without causing any loss/rejection of components due to surface finish.

4.5 Summary

In this Chapter, a heuristic online optimization suitable for cloud computing system is developed. Unlike offline optimization procedure, the proposed method optimizes the machining process during production of components. The method does not require *a priori* machining models and no offline experiments is needed. The methodology is demonstrated for optimizing finish and multipass machining process. The process is optimized for obtaining optimum cutting velocity and feed. Optimization objective is to minimize the machining cost satisfying the desired surface roughness. The effectiveness of the model is tested on virtual machine. In this work, the semi virtual lathe consisting of three modules *viz.*, surface roughness prediction module, tool life module and cutting force module is designed. For given values of inputs parameters, the virtual machine produces the actual outputs. The machining experiments are carried out through simulated results. This reduces number of physical experiments by saving time needed for conducting experiments as well as material cost. The online optimization is applied for multipass machining process. The optimization procedure and illustrative examples are presented.

The proposed scheme of online machining optimization obtains reliable optimum cutting parameters by its direct implementation. The procedure may be implemented over intranet/internet in a cloud computing system. The online optimization procedure generates real machining data sets which can be used for developing predictive modelling. The datasets can also be used to develop an artificial intelligence-based online learning methodology for optimizing cutting parameters.

Chapter 5

Fuzzy Set Approach to Optimization

5.1 Introduction

For optimizing the machining processes researchers have been using various traditional and non-traditional optimization techniques. Since the publication of first paper on the fuzzy set theory (Zadeh, 1965), it has been applied to a number of engineering problems (Bojadzieve and Bojadzieve, 1995; Dixit and Dixit, 2008). The applications can be broadly classified into three categories— (i) decision making with the help of fuzzy set operations, (ii) use of fuzzy arithmetic wherein physical variables are considered as fuzzy numbers and (iii) use of fuzzy logic in modelling and control problems. The fuzzy set theory has also been applied in the area of optimization (Rao, 1987; Luhandjula, 1989; Ekel *et al.*, 1998; Liu, 2002; Dixit *et al.*, 2002).

Rao (1987) proposed a computational approach to solving a multi-objective fuzzy optimization problem consisting of fuzzy constraints and fuzzy objective functions. The approach solves fuzzy optimization problem using conventional single-objective programming techniques. Luhandjula (1989) reviewed fuzzy optimization problems by classifying them into two parts: flexible programming and mathematical programming problems with fuzzy parameters. Flexible programming is capable of tolerating some flexibility in the formulation of goals and constraints of the mathematical program. Although the objective function and constraints in the problem are in crisp forms, they are specified/satisfied in a fuzzy form. For example, "Find $\boldsymbol{x}\in\boldsymbol{X}$ such that $f(\boldsymbol{x})$ is maintained to a lower level while $g_i(\boldsymbol{x})\leq b_i$ (i=1,...,m) are met as well as possible" is a flexible programming problem. In mathematical programming problems with fuzzy parameters, some or all coefficients of the problems are fuzzy numbers.

The Luus and Jaakola (LJ) (1973) proposed an optimization procedure for solving nonlinear optimization problems, which uses randomly chosen test points in the search space that is decreased in size as iterations proceed, in search of optimum. In this work, a fuzzy set approach to optimization has been proposed. It is similar to Luus and Jaakola algorithm, but here instead of probalilistic approach, fuzzy set approach is adopted. A similar method is grid search method, in which the search space is divided into number of grids. The objective function values at all the grid points are evaluated and the lowest function value is obtained (Rao., 2011). The proposed fuzzy set based optimization method can be considered as generalised and fuzzy form of grid based method. Further discussion is presented in Section 5.5.

It consists of finding the optima with the help of IF-THEN rules. The method is found to be effective in solving the engineering problems, where usually an approximate solution is desired. In case the optimization problems having large number of variables the optimization procedure is modified suitably. Thus, the algorithm does not suffer from 'curse of dimensionality' and found suitable for optimization the problem with several decision variables.

The efficiency of the optimization algorithm is tested with two standard test problems and results are compared. The methodology is also applied for both offline and online machining optimization problems. The experimental demonstration of the proposed strategy is presented for finish turning of cold rolled steel with TiN coated carbide tool. Online optimization strategy combined with cloud computing helps to share the informations on a cloud. In this Chapter, Section 5.2 describes the methodology of fuzzy set based optimization procedure. In Section 5.3 some test functions are solved and compared with its standard result. The application of fuzzy set based optimization in machining optimization problems is presented in Section 5.4. It is applied for both offline and online machining optimization. The merits and demerits are presented in Section 5.5. The experimental demonstration of the fuzzy based online optimization strategy is presented in Section 5.6. A note on tool life enhancement strategy is described in Section 5.7. The summary of the Chapter is presented in Section 5.8.

5.2 Methodology

The algorithm based on fuzzy set based optimization methodology solving general case of optimization problem is demonstrated here. Figure 5.1 shows the general procedure and steps of fuzzy set based optimization.

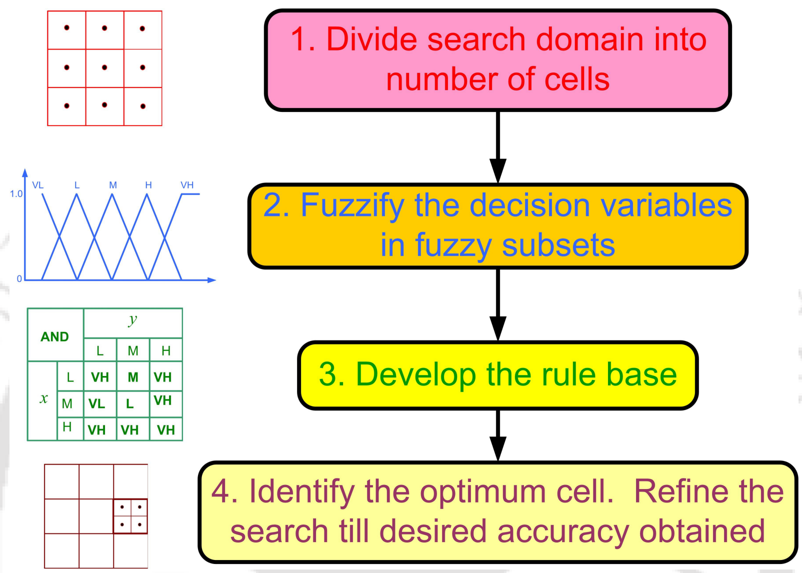


Figure 5.1. Procedure and steps of fuzzy set based optimization

Step 1. Dividing the search domain: The search domain is divided into number of cells. Identify the centroid of the cells.

Step 2. Fuzzifying the decision variables: Fuzzify decision variables into a number of fuzzy subsets. Each fuzzy subset of a variable has membership grade 1 at the centroid of the cell and 0 at the centroid of the adjacent cells. The membership functions for the fuzzy subsets can be linear or non-linear.

Step 3. Developing the fuzzy rules: If the fuzzy rules are available for each cell based on physical consideration, then go to Step 3. Else, generate the fuzzy rules. For this purpose, the following method may be employed. Obtain the function values at the centroid of each cell. Based on the function values, decide the minimum ($X_{\min}$) and maximum ($X_{\max}$) values. Fuzzify the output variable into n number of overlapping fuzzy sets. For simplicity, linear membership functions may be chosen. First and last fuzzy subset membership functions can be taken as half trapezoidal and the remaining membership functions may be triangular. Suppose the variable bounds for a variable are X_{min} and X_{max}, then for i-th fuzzy subset of the fuzzified variable, the value at the vertex corresponding to the membership grade 1 is given by

$$X_{min} + \frac{X_{max} - X_{min}}{n-1}(i-1)\,. \tag{5.1}$$

Similarly, the right and left limits of the fuzzy subset (vertices corresponding to 0 membership grades) are given by

$$\text{right limit} = X_{min} + \frac{X_{max} - X_{min}}{n-1}i\,, \tag{5.2}$$

$$\text{left limit} = X_{min} + \frac{X_{max} - X_{min}}{n-1}(i-2) \tag{5.3}$$

Note that the first fuzzy subset does not have a left vertex corresponding to membership grade of 0 and last fuzzy subset does not have a right vertex corresponding to membership grade of 0. Now, for each cell, the consequent part of the rule is the fuzzy subset of the output at the centroid of the cell and the strength of the rule is the membership grade of fuzzy subset.

Step 4. Identifying the optimum cell and refining the search:

Based on the rule base, desired objective and constraints identify the cells where the optimum may lie. Refine the search (beginning with Step 1) by choosing the cell having the highest strength of the rule amongst the identified cells. Depending on the result, either move on to the next identified cell or refine further or stop.

5.3 Test Problems

The proposed fuzzy set based optimization tested with two standard test problems.

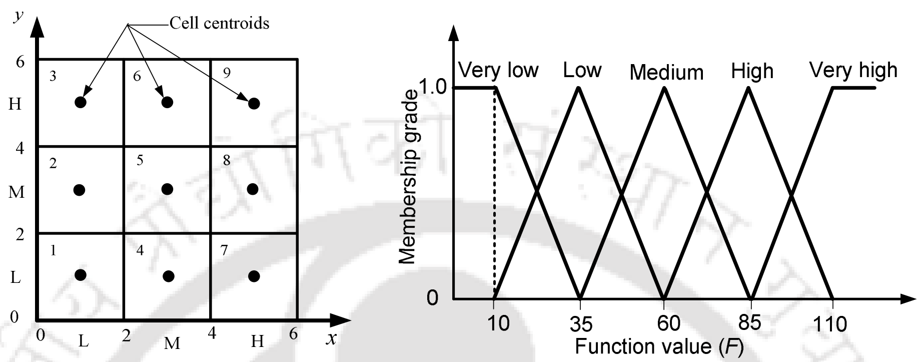
5.3.1 Test Function 1

Consider the following optimization problem.

$$\begin{aligned}&\text{Minimize } F = \left(x^2 + y - 7\right)^2 + (x + y^2 - 11)^2\\&\text{Subject to } 0 \le x \le 6;\ 0 \le y \le 6\end{aligned} \tag{5.4}$$

The objective function is a Himmelbau's function. The function is to be minimized. The search domain is two dimensional space having $0 \le x \le 6$ and $0 \le y \le 6$. The following steps are followed for solving the problems.

1. The search domain is divided into 9 cells as shown in Figure 5.2 (a); each decision variables x and y gets divided into low, medium, and high cells. The centroid of the each cell is identified and the function values are evaluated. Table 5.1 shows the function values at the centroid of each cell.



(a) (b)

Figure 5.2. Fuzzy optimization (test function 1) (a) linguistic division of search domain into 9 cells (b) fuzzification of function values

Table 5.1. Function values at cell centroids (test function 1)

Cell number	Decision variables		Function value
	x	y	F
1	1 (Low)	1 (Low)	106
2	1 (Low)	3 (Medium)	10
3	1 (Low)	5 (High)	226
4	3 (Medium)	1 (Low)	58
5	3 (Medium)	3 (Medium)	26
6	3 (Medium)	5 (High)	338
7	5 (High)	1 (Low)	386
8	5 (High)	3 (Medium)	450
9	5 (High)	5(High)	890

2. The obtained function values are in the range from 10 to 890. As the goal of the problem is the minimization of function values, $F_{\min}$ = 10 and $F_{\max}$= 110 is

chosen. The output variable is fuzzified into 5 fuzzy subsets using Eqs. (5.1–5.3) as shown in Figure 5.2 (b).

3. Table 5.2 depicts the rule base along with the strength of the rules. As the goal is to obtain very low value of objective function, the Rule 2 is fired.

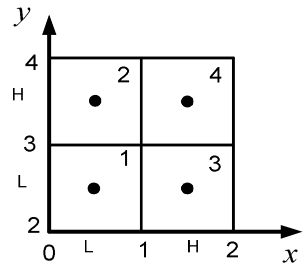
Rule 2: If x is low and y is medium F is very low

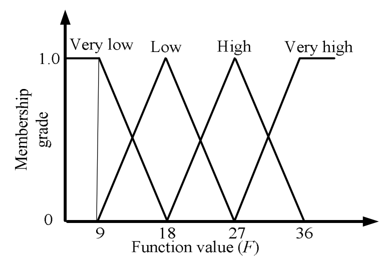
Table 5.2. Rule base with strength of rules (test function 1)

x	y	F	Strength
Low	Low	High	0.15
Low	**Medium**	**Very low**	**1.0**
Low	High	Very high	1.0
Medium	Low	Low	0.08
Medium	Medium	Very Low	0.36
Medium	High	Very high	1.0
High	Low	Very high	1.0
High	Medium	Very high	1.0
High	High	Very high	1.0

4. The search domain now gets reduced to $0 \le x \le 2$ and $2 \le y \le 4$. Further search is refined in this zone.

   The domain is now divided into 4 cells with the decision variables getting divided into 'low' and 'high'. Since the search domain now becomes small it is divided into lesser number of cells. The function values corresponding to cell centroids are obtained and it ranges from 9 to 36. The output variable is fuzzified into 4 fuzzy sub sets. Figure 5.3 shows the division of domain and fuzzification of the output variable and Table 5.3 depicts the cell centroid and rule base of the reduced search domain. From the rule base, the rule 4: "If x is high and y is high, F is very low" having the membership grade 1.0 is fired. The search domain now gets reduced to $1 \le x \le 2$ and $3 \le y \le 4$.





(a) (b)

Figure 5.3. Search domain: $0 \le x \le 2$ and $2 \le y \le 4$ (a) division of domain into 4 cells (b) fuzzification of function values.

Table 5.3. Cell centroids with rule base of the domain $0 \le x \le 2$ and $2 \le y \le 4$

Cell number	Decision variables		Function value	Strength
	x	y	F	
1	0.5 (Low)	2.5 (Low)	36 (Very high)	1.0
2	0.5 (Low)	3.5 (High)	14 (Very low)	0.44
3	1.5 (High)	2.5 (Low)	16 (Very low)	0.22
4	**1.5 (High)**	**3.5 (Low)**	**9 (Very low)**	**1.0**

The procedure may be continued, till the cell size becomes sufficiently small. Figure 5.4 shows the reducing size of the search domain for different iterations. The method provides optimum zone as $1.75 \le x \le 2$ and $3 \le y \le 3.25$ after 4 iterations with small variation in function value as $0 \le F \le 2.5$. The optimum of the problem is (2, 3) and the function value is 0.

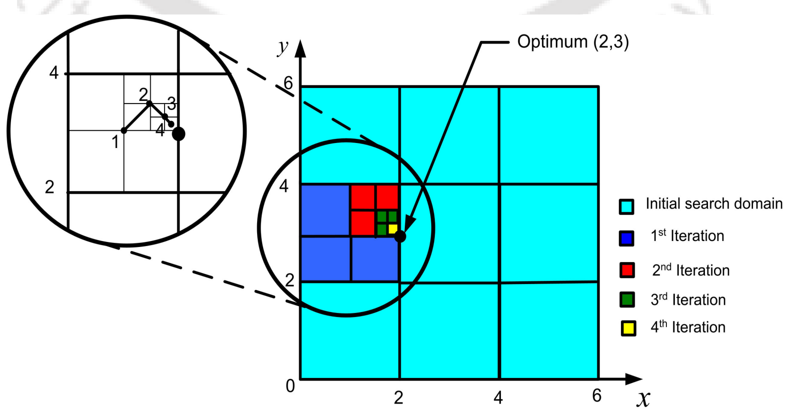


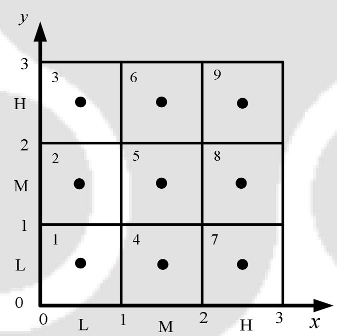
Figure 5.4. Reducing sizes of search domain and iterative points (test function 1)

5.3.2 Test Function 2

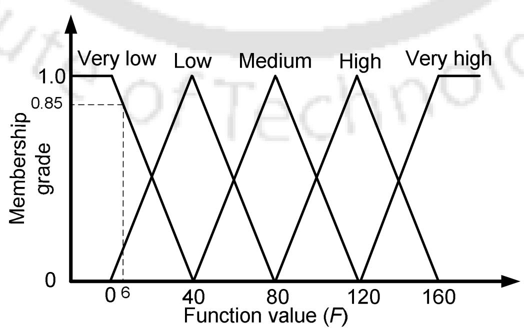
Consider the following optimization problem.

$$\begin{aligned} &\text{Minimize } F = 10\left(y + x^2\right)^2 + (1-x)^2 \\ &\text{Subject to } 0 \le x \le 3;\ 0 \le y \le 3 \end{aligned} \tag{5.5}$$

The search domain is $0 \le x \le 3$ and $0 \le y \le 3$. The two variables non-linear function is solved by fuzzy set based optimization and compared with result. The initial search domain is divided into 9 cells with decision variables gets divided into 'low', 'medium' and 'high' as shown in Figure 5.5 (a). The function values are evaluated at each cell centroids. Table 5.4 shows the function values at the centroid of each cell. It is observed that function values range from 6 to 768. As the goal of the problem is minimization of function values, $F_{\min}$ = 0 and $F_{\max}$= 160 is chosen. The output variable is fuzzified into 5 fuzzy subsets as shown in Figure 5.5 (b). Table 5.5 depicts the rule base along with the strength of the rules. Rule 1 has the highest membership strength and it is fired.



(a)



(b)

Figure 5.5. Fuzzy optimization (test function 2) (a) Division of search domain into 9 cells (b) fuzzification of function values.

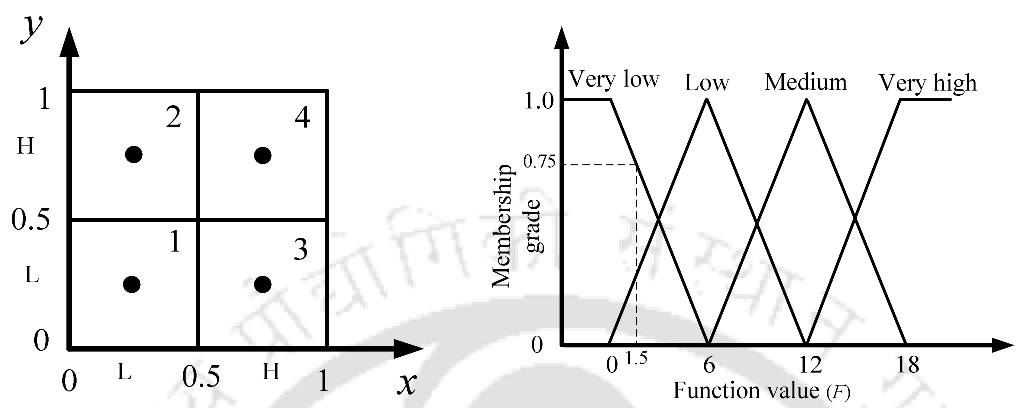
Table 5.4. Function values at cell centroids (test function 2)

Cell number	Decision variables		Function value (F)
	x	y	
1	0.5(low)	0.5(low)	6.0
2	0.5(low)	1.5(medium)	31.0
3	0.5(low)	2.5(high)	76.0
4	1.5(medium)	0.5(low)	76.0
5	1.5(medium)	1.5(medium)	141.0
6	1.5(medium)	2.5(high)	226.0
7	2.5(high)	0.5(low)	458.0
8	2.5(high)	1.5(medium)	603.0
9	2.5(high)	2.5(high)	768.0

Table 5.5. Rule base with strength of rules (test function 2)

x	y	F	Strength
Low	**Low**	**Very low**	**0.85**
Low	Medium	Very low	0.23
Low	High	Low	0.1
Medium	Low	Low	0.1
Medium	Medium	High	0.48
Medium	High	Very high	1.0
High	Low	Very high	1.0
High	Medium	Very high	1.0
High	High	Very high	1.0

The search domain now gets reduced to $0 \le x \le 1$ and $0 \le y \le 1$. The search domain is refined further. The domain is divided into 4 cells with the decision variables gets divided into 'low', 'high'. The output variable is fuzzified into 4 fuzzy sub sets. Figure 5.6 (a) shows the division of 4 cells and Figure 5.6 (b) shows fuzzification of the output variable. Table 5.6 depicts the cell centroid and rule base of the reduced search domain. On firing rule 1, the search domain now gets reduced to $0 \le x \le 0.5$ and $0 \le y \le 0.5$.



(a)

(b)

Figure 5.6. Search domain: $0 \le x \le 1$ and $0 \le y \le 1$ (a) division of domain into 4 cells (b) fuzzification of function values

Table 5.6. Cell centroids with rule base of the domain $0 \le x \le 1$ and $0 \le y \le 1$

Cell number	Decision variables		Function value (F)	Strength
	x	y		
1	**0.25 (low)**	**0.25 (low)**	**1.5 (very low)**	**0.75**
2	0.25 (low)	0.75 (high)	7.2 (low)	0.8
3	0.75 (high)	0.25 (low)	6.7 (low)	0.88
4	0.75 (high)	0.75 (low)	17.3 (high)	0.12

The procedure is continued till the cell size becomes sufficiently small. Figure 5.7 shows the reducing size of the search domain for different iterations. The method provides optimum zone as $0.25 \le x \le 0.375$ and $0 \le y \le 0.125$ after 4 iterations with small variation in function value as $0.56 \le F \le 1.1$. The optimum of the problem is (0.32, 0) and the function value is 0.567.

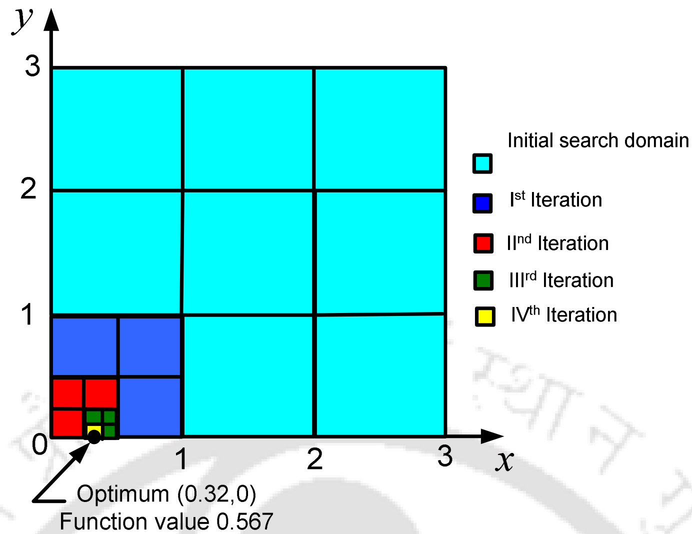


Figure 5.7. Reducing sizes of search domain and the optimum zone (test function 2).

5.4 Application for Machining Optimization

For the objective of present thesis the proposed fuzzy set based optimization is applied for optimization of machining process. The problem is highly nonlinear in nature. The optimization technique is used for obtaining optimum process parameters. It is applied for both offline and online machining optimization problem.

5.4.1 Offline Machining Optimization

Consider an optimization of multipass turning process. The optimization model is based on the analysis presented by Ojha *et al.* (2009). The total production time per component is given by Eq. (4.13) and the machining/production cost of a component is given by Eq. (4.14).

The optimization model is solved using the proposed fuzzy optimization method with an illustrative example. A standard example is chosen from reference Ojha *et al.* (2009). It is concerned with the multipass turning of a workpiece to reduce its diameter by 12 mm. A Taylor's tool life equation has been provided, which indicates that cutting speed has the maximum influence on tool life followed by feed and depth of cut. Surface finish is important in the finishing pass but not in the roughing pass. There is a limit of cutting force and cutting power. With this information, the following rules are used.

- Try to take maximum possible depth of cut in each pass.
- Obtain the feed based on maximum possible force in roughing pass and desired surface roughness in the finishing pass.
- Once the feed and depth of cut are decided, choose the cutting speed based on the maximum power.

With these rules, the following values of process parameters are obtained: depth of cut for roughing pass = 3 mm, depth of cut for finishing pass = 3 mm, feed for roughing pass = 0.56 mm/rev, feed for finishing pass = 0.3 mm/rev, cutting speed for roughing pass = 130 m/min and cutting speed for finishing pass = 209 m/min. The cost of machining at these parameters may not be the minimum due to lesser tool life.

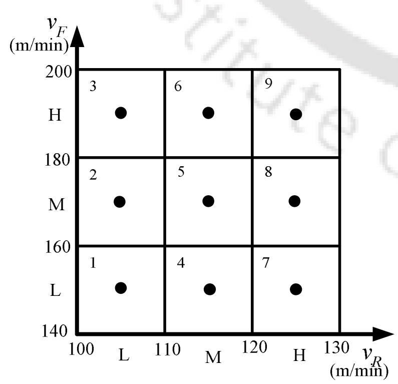
As the cutting speed has the most influence on tool life, it is decided to search for roughing speed (v_R) in the range 100–130 m/min and finishing speed (v_F) in the range 140–200 m/min. Eq (4.14) is simplified as two variables function and the optimization problem is

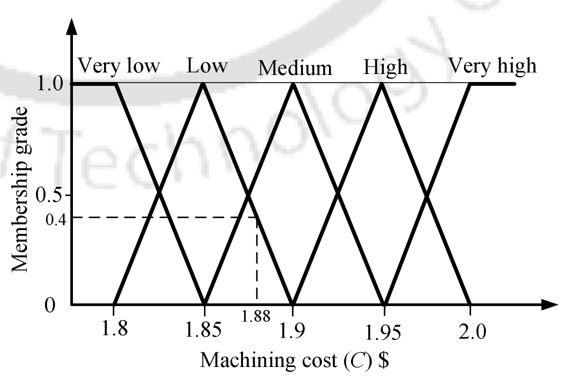
$$Minimize\_C = 0.5\left(\frac{84.15}{v_R} + 1.738\times10^{-10}{v_R}^4 + \frac{138.23}{v_F} + 9.599\times10^{-11}{v_F}^4 + 1.77\right)$$

$$+2.5\left(1.159\times10^{-10}{v_R}^4 + 6.399\times10^{-11}{v_F}^4\right) \quad (5.6)$$

$$\text{subjected to:} 100 \le v_R \le 130 \text{ and } 140 \le v_F \le 200 \quad (5.7)$$

The search domain is initially divided into 9 cells as shown in Figure 5.8 (a); each decision variables are divided into 'low',' medium', and 'high'. Table 5.7 shows function values at the centroid of each cell in the search domain.





(a) (b)

Figure 5.8. Fuzzy optimization of turning process (offline – turning) (a) linguistic division of search domain (b) fuzzification of function values

Table 5.7. Function values at cell centroids (offline optimization)

Cell number	Rough cutting speed (v_R) (m/min)	Finish cutting speed (v_F) (m/min)	Machining cost (C) ($)
1	105 (low)	150 (low)	1.90
2	105 (low)	170 (medium)	1.91
3	105 (low)	190 (high)	1.97
4	115 (medium)	150 (low)	1.88
5	115 (medium)	170 (medium)	1.90
6	115 (medium)	190 (high)	1.95
7	125 (high)	150 (low)	1.88
8	125 (high)	170 (medium)	1.89
9	125 (high)	190 (high)	1.95

It is observed that the function values are in the range as $C_{\min}$ = 1.88 and $C_{\max}$ = 1.95. The output variable is fuzzified using triangular membership function with 5 fuzzy subsets *viz.* very low, low, medium, high, very high using Eqs. (5.1–5.3) and is shown in Figure 5.8 (b). Table 5.8 depicts the rule base along with the strength of the rules in the search domain. As the goal is to obtain very low value of production cost per component, the fourth or seventh rule may be fired. Firing of seventh rule the search domain gets reduced to $120 \le v_R \le 130$ and $140 \le v_F \le 160$, and this refined search domain is now divided into 4 cells as shown in Figure 5.9 (a).

Table 5.8. Rule base with strength of the rules (offline optimization)

v_R	v_F	C	Strength
Low	low	medium	1.0
Low	medium	medium	0.8
Low	high	high	0.6
medium	low	low	0.4
medium	medium	medium	1.0
medium	high	high	1.0
High	low	low	0.4
High	medium	low	0.2
High	high	high	1.0

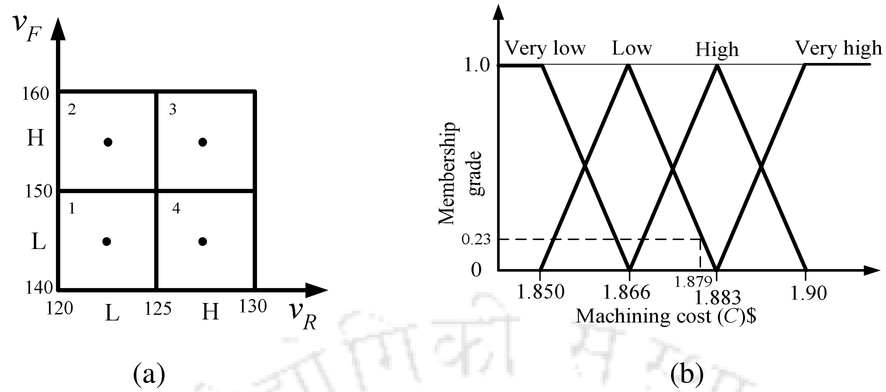


Figure 5.9. Search domain: $120 \le v_R \le 130$ and $140 \le v_F \le 160$ (a) division of domain into 4 cells (b) fuzzification of output variable

Each decision variable now gets divided into 'low' and 'high'. The output variable is fuzzified into four fuzzy subsets *viz.* very low, low, high, very high as shown in Figure 5.9(b). Table 5.9 shows the function values at the cell centroids and corresponding rule base with its strength.

Table 5.9. Rule base for refined search domain (offline optimization)

Cell centroid (v_R ,v_F)	Machining cost (C)	Rule base (v_R–v_F–C)	Strength
122.5,145	1.882	Low–Low–Low	0.06
122.5, 155	1.879	Low–High–Low	0.23
127.5,145	1.883	High–Low–Low	0.00
127.5,155	1.881	High–High–Low	0.15

For obtaining minimum machining cost rule two is fired and the search space confined to the domain: $120 \le v_R \le 125$ and $150 \le v_F \le 160$. Figure 5.10 (a) depicts various co-ordinate positions in the fuzzy domain and Figure 5.10 (b) shows the corresponding machining costs. The cost of machining per component in this range comes out to \$1.88 (or ₹113; 1\$ = ₹60) per piece, as obtained by previous researchers by different methods.

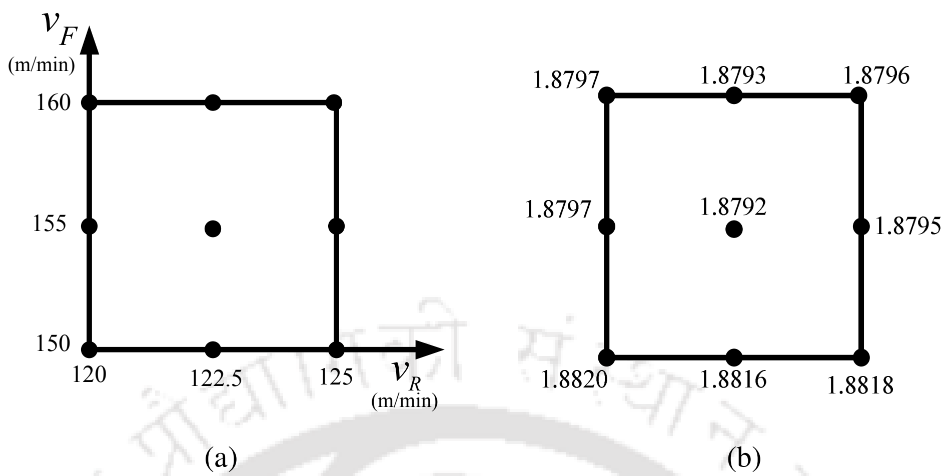


Figure 5.10. Optimum fuzzy domain (offline optimization) (a) co-ordinates of decision variables (b) corresponding machining cost

5.4.2 Online Machining Optimization

In online optimization strategy employing fuzzy set based optimization the function values at cell centroids are evaluated by obtaining tool life directly from the machining process. Consider an optimization of online finish machining process to obtain optimum cutting speed (*v*) and (*f*) for the maximum allowable surface roughness (R_a^*). The depth of cut (*d*) during finish machining process is kept constant. Let the target surface roughness (R_a^t) is 80% of R_a^*. Considering some statistical variation in surface roughness the actual surface roughness ($\hat{R}_a$) produced is kept between the lower and upper limit with 20% variation from the target surface roughness and it is $0.8R_a^t \le \hat{R}_a \le 1.2R_a^t$. The demonstrated test problems on fuzzy set based optimization do not have any constraints. However in online optimization strategy the surface roughness must be satisfied and the problem become constrained optimization. The search domain is bounded as $v_{\min} \le v \le v_{\max}$ and $f_{\min} \le f \le f_{\max}$. The cutting parameters are optimized to minimize the cost. The illustrative examples are presented for turning and end milling processes

5.4.2.1 Example 1: An Application to Turning Process

Consider a finish turning process having depth of cut of 0.5 mm to obtain maximum allowable surface roughness value (R_a^*) of 2.5 µm. The target surface roughness (R_a^t) is evaluated as 2.0 µm. The actual surface roughness may lie between 1.60 and 2.40 µm. The search domain is in the range 140–240 m/min for cutting speed and 0.04–0.32 mm/rev for feed. The linguistic sub division of search domain is shown in Figure 5.11(a). Machining is performed at each of cell centroids. For conducting machining experiments the semi-virtual lathe designed in Section 4.2 is used.

If the actual surface roughness ($\hat{R}_a$) obtained is within the limit, continue machining till tool fails due to surface roughness and obtain tool life. The objective of optimization is to minimize the machining cost of a component satisfying the surface roughness. The objective function is given by the Eq. (4.8). The results are shown in Table 5.10.

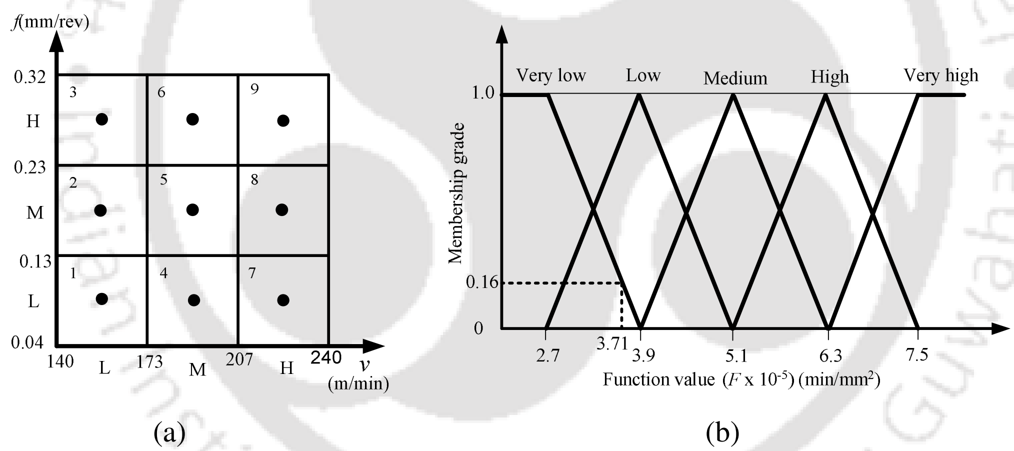


Figure 5.11. Fuzzy optimization (offline–turning) (a) linguistic division of search domain (b) fuzzification of function values

Choosing F_{min}=2.7x10^{-5} and F_{max}=7.5x10^{-5} min/mm^2 the output variable is fuzzified into 5 fuzzy subsets as shown in Figure 5.11(b). Table 5.11 depicts the rule base along with strength of the rules. As the problem objective is to minimize the function value, the rule 5 having highest membership function is fired. The search domain is reduced to $173 \le v \le 207$ and $0.13 \le f \le 0.23$.

Table 5.10. Function values at centroids of each cell (online optimization)

Cell num -ber	Cutting condition		Surface roughness obtained ($\hat{R}_a$) (μm)	Is $\hat{R}_a$ within accepted range?	Tool life (T) (min)	Function value (F) (×10^{-5} min/mm^2)
	v (m/min)	f (mm/rev)				
1	156.7 (low)	0.09 (low)	2.79	No	-	-
2	156.7 (low)	0.18 (medium)	2.38	Yes	97.7	3.91
3	156.7 (low)	0.27 (high)	3.69	No	-	-
4	190 (medium)	0.09 (low)	2.59	Yes	-	-
5	190 (medium)	0.18 (medium)	2.10	No	37.3	3.71
6	190 (medium)	0.27 (high)	3.33	No	-	-
7	223 (high)	0.09 (low)	2.20	Yes	28.1	6.96
8	223 (high)	0.18 (medium)	1.83	Yes	16.7	3.99
9	223 (high)	0.27 (high)	2.67	No	-	-

Table 5.11. Rule base with strength of rules (online optimization)

Cell number	Cutting speed (v)	Feed (f)	Function value (F)	Membership grade
1	Low	Low	-	-
2	Low	Medium	Low	0.99
3	Low	High	-	-
4	Medium	Low	-	-
5	**Medium**	**Medium**	**Very Low**	**0.16**
6	Medium	High	-	-
7	High	Low	High	0.45
8	High	Medium	Low	0.93
9	High	High	-	-

The new search is now initiated in the identified search domain, dividing it into 4 cells with the fuzzy sub-set being ‘low’ and ‘high’. This provides new domain:

$173 \le v \le 190$ and $0.18 \le f \le 0.23$. In the obtained fuzzy domain the surface roughness produced are in the range between $2.08 \le \hat{R}_a \le 2.31$, which is well within limits and the function value has no significant variation, which is in the range of $3.3\times10^{-5} \le F \le 3.7\times10^{-5}$.

Thus, with the given range of cutting speed and feed rate, the fuzzy set based optimization provides the following solution:

$$173 \le v \le 190, \quad 0.18 \le f \le 0.23. \qquad (5.8)$$

The cost of machining (C) comes out in the range of $\$0.14 \le C \le \$0.17$ per piece. With small variation in cost the fuzzy set based optimization provides multiple numbers of solutions. The optimum solution is obtained in eight iterations such that four iterations are performed each in fuzzy machining 1 and fuzzy machining 2. Figure 5.12 shows the position of optimum zone in the entire search space with the black dot showing the iterative cutting conditions.

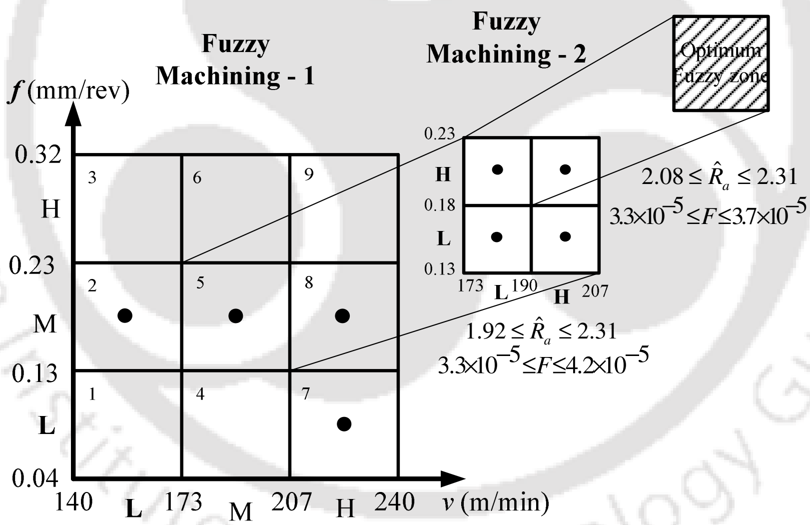


Figure 5.12. Optimum fuzzy domain (online optimization)

A number of other problems having depth of cut varying from 0.1 mm to 1.2 mm for different values of maximum allowable surface roughness are tested and the results are shown in Table 5.12. The function value and surface roughness produced in optimum fuzzy domain do not vary significantly

Table 5.12. Results of different problem

Problem number	Data set (d R_a^*) (mm µm)	Fuzzy optimum (v_{opt} f_{opt}) (m/min mm/rev)	Surface roughness Range ($\hat{R}_a$)(µm)	Function value range (F) ($\times 10^{-5}$min/mm^2)	Number of iterations
1	(0.1 3.00)	$224 \le v \le 240; 0.18 \le f \le 0.23$	1.95–2.55	2.4–2.9	9
2	(0.2 2.75)	$190 \le v \le 207; 0.18 \le f \le 0.23$	2.36–2.44	2.6–3.2	10
3	(0.3 3.25)	$223 \le v \le 240; 0.18 \le f \le 0.23$	2.69–2.97	2.3–2.7	9
4	(0.4 3.75)	$190 \le v \le 207; 0.28 \le f \le 0.32$	3.12–3.58	2.3–2.5	7
5	(0.5 2.50)	$173 \le v \le 190; 0.18 \le f \le 0.23$	2.08–2.31	3.1–3.7	9
6	(0.6 3.00)	$207 \le v \le 224; 0.28 \le f \le 0.32$	2.61–2.73	2.6–2.7	10
7	(0.7 4.00)	$157 \le v \le 173; 0.28 \le f \le 0.32$	3.23–3.62	2.4–2.7	7
8	(0.8 2.60)	$173 \le v \le 190; 0.18 \le f \le 0.23$	1.71–2.34	3.2–3.9	8
9	(0.9 4.00)	$157 \le v \le 173; 0.28 \le f \le 0.32$	2.65–3.09	2.5–2.8	7
10	(1.0 3.50)	$157 \le v \le 173; 0.28 \le f \le 0.32$	2.52–2.77	2.5–2.9	11

5.4.2.2 Example 2: An Application to End milling Process

The online optimization is tested for optimization of end milling process. In optimization of finish end milling process the process parameters *i.e.,* cutting speed and feed are optimized to minimize the cost, as depth of cut remains constant in the finish machining process. Figure 5.13 shows schematic view of end milling process showing different process parameters.

In end milling operations the production time per component can be determined by

$$T_{\mathrm{p}} = \frac{\pi LD}{1000vf'z}\left(1+\frac{t_{\mathrm{c}}}{T}\right), \tag{5.9}$$

where L is the machining length, D is the cutter diameter, v is cutting speed, f' is feed per tooth and z is number of tooth, t_c is the tool/cutter change time, and T is the tool life for particular cutting condition.

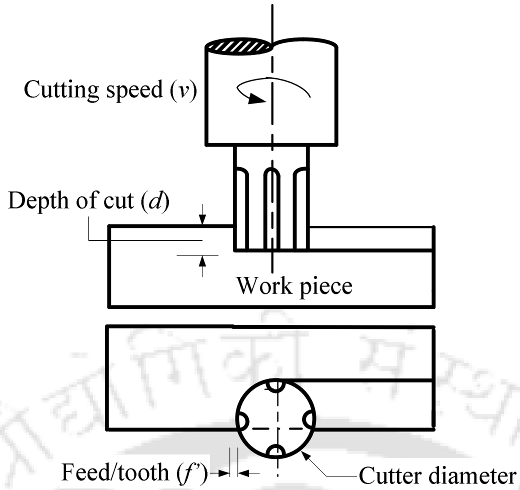


Figure 5.13. End milling process

The tool life is a function of cutting velocity, feed and depth of cut. Tolouei-Rad and Bidhendi (1997) proposed an empirical relation to evaluate tool life T for end milling process as

$$T = \frac{60}{Q}\left(\frac{C_2\left(0.2d/f\right)^g}{\left(df\right)^w v}\right)^{1/n}, \tag{5.10}$$

where T is the tool life in minutes, v is the cutting speed in m/min, C_2 is the constant for 60 min tool life when the area of cut is 1 mm^2, f is the feed in mm/rev$(= f' \times z)$, d is the depth of cut in mm, and g, w, and n are exponents for different tool and work material combination. The factor Q is contact proportion of cutting edges with work piece per revolution.

If C_0 is the operating cost in \$/min, and C_{tc} is the tool change cost in \$/cutting edge, then machining cost per job is given by

$$C = C_0 T_p + zC_{tc}\left(\frac{\pi DL}{1000vf'zT}\right), \tag{5.11}$$

Substituting the value of T_p from Eq. (5.9) in Eq. (5.11),

$$C = C_0\frac{\pi LD}{1000vf'z}\left(1+\frac{t_c}{T}\right) + zC_{tc}\left(\frac{\pi DL}{1000vf'zT}\right). \tag{5.12}$$

With the constant values of costs, length and diameter, the objective function to minimize machining cost is expressed by the function F in Eq. (4.8), in which $t_c^* = t_c + zC_{tc}/C_0$. Considering t_c = 5 min, the equation reduces to two-variables (f and v)

problem. This function is optimized using fuzzy optimization method, thus optimum cutting speed and feed per tooth are obtained satisfying the maximum allowable surface roughness within its control limit.

Consider a finish end milling of 6061 aluminium alloy using HSS end mill cutter. The cutter diameter is 19 mm. Optimum cutting parameters to be obtained for machining 0.8 mm depth of cut to obtain maximum allowable surface roughness value (R_a^*) of 3.25 μm. The target surface roughness (R_a^t) is evaluated as 2.6 μm. The actual surface roughness may lie between 2.08 and 3.12 μm. The search domain is in the range 45–90 m/min for cutting speed and 0.03–0.23 mm/tooth for feed. The linguistic sub division of search domain is shown in Figure 5.14 (a). Machining is performed at each of cell centroids. For conducting machining experiments the semi-virtual milling (Appendix C) is used using experimental datasets from Lou and Chen (1999). If the actual surface roughness ($\hat{R}_a$) is satisfied, continue machining till tool fails due to surface roughness and obtain tool life time. The objective of optimization is to minimize the machining cost of a component satisfying the surface roughness. The objective function is given by the Eq. (5.13). The results are shown in Table 5.13.

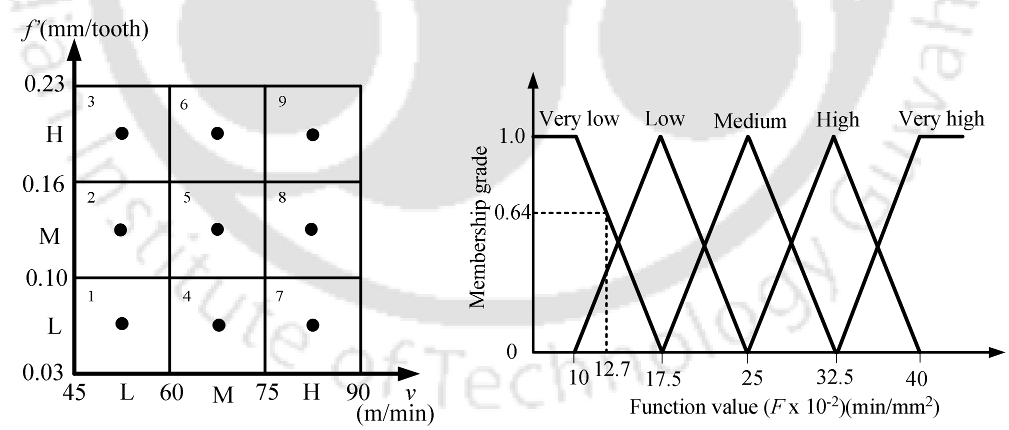


Figure 5.14. Fuzzy optimization (online–end milling) (a) linguistic division of search domain (b) fuzzification of function values

Table 5.13. Function values at cell centroids (online–end milling)

Cell No.	Cutting condition (v, f')	$\hat{R}_a$ obtained	$\hat{R}_a$ satisfied?	Function value (F) (min/mm^2)
1	(52.5, 0.06)	2.08	No	-
2	(52.5, 0.13)	3.26	No	-
3	(52.5, 0.20)	4.43	No	-
4	(67.5, 0.06)	2.28	Yes	0.237
5	(67.5, 0.13)	3.13	Yes	0.127
6	(67.5, 0.20)	4.78	No	-
7	(82.5, 0.06)	2.22	Yes	0.200
8	(82.5, 0.13)	3.09	Yes	0.135
9	(82.5, 0.20)	4.92	No	-

Choosing $F_{\min}$= 0.1 and $F_{\max}$= 0.4 the output variable is fuzzified into 5 fuzzy subsets as shown in Figure 5.14 (b). Table 5.14 depicts the rule base along with strength of the rules. As the problem objective is to minimize the function value, the rule 5 having higher strength of membership grade is fired. The search is now refined in the corresponding cell number *i.e.*, cell 5.

Table 5.14. Rule base with strength of rules (Online–end milling)

Cell number	Cutting speed – feed	Function value (F)	Membership grade
1	Low – Low	-	-
2	Low – Medium	-	-
3	Low – High	-	-
4	Medium – Low	Low	0.17
5	**Medium – Medium**	**Very Low**	**0.64**
6	Medium – High	-	-
7	High – Low	Low	0.67
8	High – Medium	Very Low	0.53
9	High – High	-	-

Thus, the search domain is reduced to $60 \le v \le 75$ and $0.10 \le f' \le 0.16$. The new search is now initiated in the identified search domain, dividing it into 4 cells with the fuzzy sub-set being 'low' and 'high'. This provides new domain: $67.5 \le v \le 75$ and $0.10 \le f' \le 0.13$. In the obtained fuzzy domain the surface roughness produced are within the limit and the function value produced in the range has no significant variation. The optimum parameter is obtained in six iterations. Figure 5.15 shows the position of optimum zone in the entire search space with the black dot showing the iterative cutting conditions.

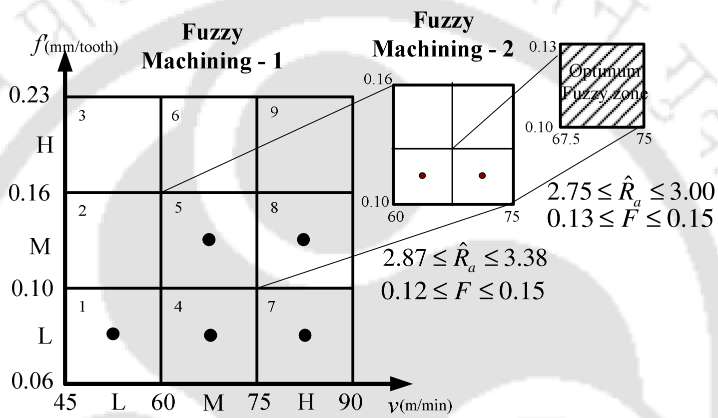


Figure 5.15. Fuzzy optimum domain (Online–end milling)

In the present case, the optimum cutting condition (v_{opt} and f'_{opt}) corresponds to centroids of fuzzy domain is: 71.3 m/min and 0.13 mm/tooth**.** The result is validated using exhaustive search method. The machining cost for producing one component at optimum cutting condition is calculated as \$0.2028 (₹ 12.20).

5.5 Advantages and Limitations of Fuzzy Set Based Optimization

The fuzzy set based optimization procedure proposed in this thesis has some similarity with fuzzy inference system and some traditional optimization methods such as grid search method and response surface methodology. If the problem contains a large number of decision variables, the procedure will take enormous amount of computational time. It is felt that if the rules could be generated by experts or some efficient technique for the rule generation is employed, this method can emerge as a viable alternative to other optimization methods. This optimization procedure has the following advantages over the other methods.

- The procedure provides number of solutions having feasibility for alternative selection of optimum cutting condition.
- As the procedure is based on linguistic subdivision of the domain, a range is obtained in which the objective function value does not change drastically. Thus, one can get a feel of sensitivity in this procedure.
- An expert knowledge can easily be incorporated and the procedure can be made interactive.
- When the information contains uncertainty and imprecision, the procedure is much more effective than other methods.
- If the function values have to be obtained by online experimentation, the procedure gains much importance, as in this procedure most of the search is confined to better regions. Therefore, during the process of search also, one is not very far away from the optimum.
- The information such as actual surface roughness produced, tool life, function value and the cost of machining are obtained at iterative cutting conditions and could be used for online learning of the machining process.

Also there are some limitations of the method too. If the function changes abruptly or a very high accuracy in decision variables is required, the procedure will take more time/iteration compared to many available optimization methods. The method is dependent on the efficiency of the rule generation module and appropriate fuzzification of the variables.

5.6 Experimental Validation of Fuzzy Set Approach to Online Machining Optimization

The experimental demonstration of fuzzy set approach to online machining optimization is presented here. The strategy does not require *a priori* machining models. It also acknowledges the presence of statistical variation in the process. The optimization procedure is described in Section 5.2. For simplicity of experimentation instead of machining at all cell centroids, the search is initiated in 'high-high' cell. Machining is carried out and surface roughness is measured. If the cutting condition satisfies surface roughness constraint, continue machining till tool fails. Evaluate machining cost/piece. Initiate the search in other cells by reducing '*v*' with '*f*' remaining same (*i.e.*, scanning horizontally) and identify minimum cost cell. The search is initiated further by varying '*f*' with '*v*' corresponding to minimum cost cell. Figure 5.16 shows the procedure to find minimum cost zone. Here, the minimum cost is found by combining two one-dimensional searches (along horizontal and vertical lines). The cell that obtains minimum cost/piece satisfying surface roughness is the optimum cost zone. The cell centroid is the optimum cutting condition.

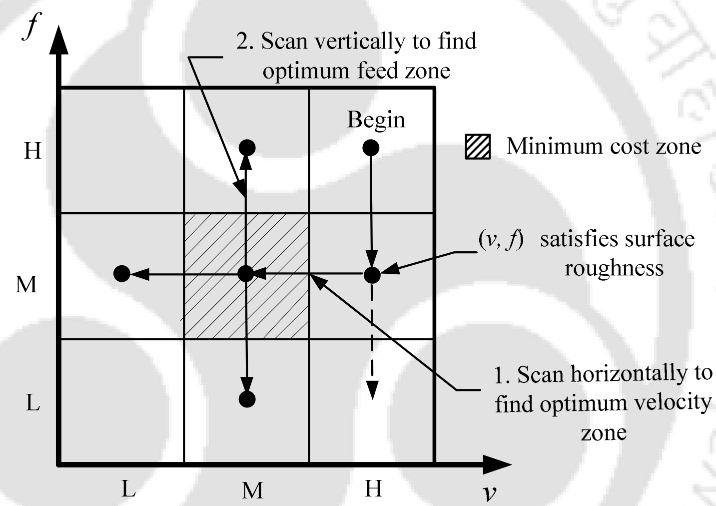


Figure 5.16. Procedure to find minimum cost zone

In case of virtual machine the prediction of surface roughness for particular cutting condition (v, f) is deterministic in nature. However it needs to be tested for randomness. Therefore, in experimental validation the machining is performed again at the optimum cutting condition and the result may differ due to randomness. Estimate new machining cost/piece and find out (average cost/piece)$_{opt}$. Based on hypothesis testing, find out the other points at which machining can be carried out without significant change in machining cost. The experimental demonstration is presented in the following subsection.

5.6.1 Experimentation

Consider that the optimum process parameters for a finish turning process are to be found for a depth of cut of 0.2 mm and allowable surface roughness value (R_a^*) of 2.5 μm. The target surface roughness (R_a^t) is 2.0 μm. Considering some statistical variation in surface roughness, let actual surface roughness ($\hat{R}_a$) should be such that $0.8R_a^t < \hat{R}_a < 1.2R_a^t$, where R_a^t is the target surface roughness and is taken as 80% of R_a^*. Thus, the actual surface roughness may lie between 1.6 and 2.4 μm. Consider the search domain with the range of cutting parameters 140–240 m/min and 0.04–0.28 mm/rev is considered for cutting velocity and feed respectively. The search domain is divided into a number of cells with linguistic sub division being low, medium, and high. The range of cutting velocity is divided equally into three fuzzified zone such as low (140–173 m/min), medium (173–207 m/min) and high (207–240 m/min). Machining is initialized at 'high-high' cell with centroid cutting condition as v=223 m/min and f=0.24 mm/rev.

Machining is carried out on HMT make NH-26 lathe. The work material is cold rolled steel bars and TiN coated carbide tool is used as cutting tool. The spindle was driven with three-phase 11 kW induction motor having 23 speeds ranges between 40 and 2040 rpm and 27 different feeds ranging from 0.04 to 2.24 mm/rev. Figure 5.17 shows experimental setup showing workpiece and cutting tool. The cutting tool used for the experiment was square shaped and having TiN coated carbide designated as SNMG 12 04 08-PM of make: Sandvik. The ASA tool signature is designated by (-9)-(-5)-7-7-30-1-0.8. The compatible tool holder was PTGNR 2020 K16 of make: WIDAX. The work material diameter was 35 mm and length was 100 mm.

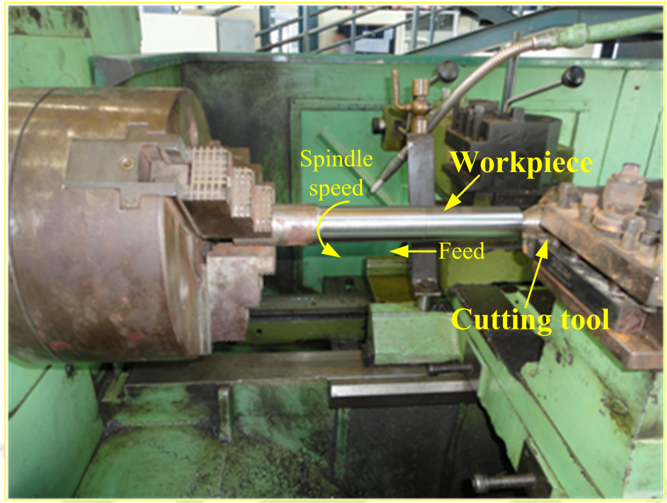


Figure 5.17. Experimental setup

The CLA surface roughness values of the turned component is measured using Pocket Surf (Mahr, GMBH). The measuring range is 0.03–6.35 μm. The surface roughness evaluation length in each case is taken as 2.4 mm. Figure 5.18 shows online surface roughness measurement of workpiece with Pocket Surf instrument.

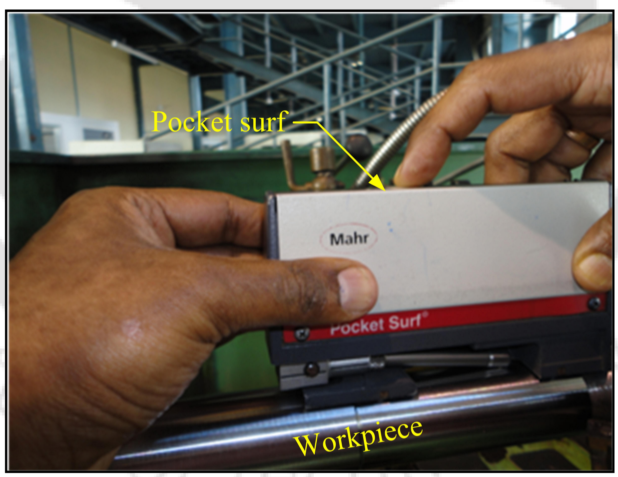


Figure 5.18. Surface roughness measurement

The surface roughness obtained at cutting condition v=223 m/min and f=0.24 mm/rev, is 3.12 μm. Since it does not satisfy the limit, the machining is now performed by reducing feed, moving to 'high-medium' cell *i.e.*, at (223 m/min, 0.2 mm/rev) and continued further till desired surface finish is obtained. The surface roughness obtained at 'high-low' cell (223 m/min, 0.08 mm/rev) is 2.14 μm that satisfies the limit. Now, machine a number of jobs till tool fails due to surface roughness. While machining at this cutting condition (*i.e.*, 223 m/min, 0.08 mm/rev) the actual surface roughness ($\hat{R}_a$) obtained are given in Table 5.15.

Table 5.15. Measurement of actual surface roughness and tool life at cutting speed of 223 m/min and feed of 0.08 mm/rev

Job diameter (D): 35 mm

Length of machining (L): 100 mm

Maximum allowable surface roughness (R_a^*) = 2.5 µm

Limit of surface roughness: 1.6–2.4 µm

Experiment/Job number	Actual surface roughness obtained ($\hat{R}_a$) (µm)	Is actual surface roughness within limit?
Job 1	2.14	Yes
Job 2	2.10	Yes
Job 3	2.21	Yes
Job 4	2.26	Yes
Job 5	2.32	Yes
Job 6	2.34	Yes
Job 7	2.42	No

Number of jobs produced: 06

The tool life is based on surface roughness criterion is given by the period that tool and work piece are in contact during machining the job. The calculation is as follows:

$$\text{Machining time/job} = t_m = \frac{L}{fN} = \frac{\pi DL}{vf} = \frac{\pi \times 35 \times 100}{1000 \times 223 \times 0.08} = 0.62 \text{ min.}$$

Tool life (T) = t_m × number of jobs produced = 0.62 × 6 = 3.72 min. In similar way actual surface roughness and tool life were evaluated for different cutting conditions. Taking operating cost (C_0) is 0.07 \$/min, tool change cost (C_{tc}) is \$2.5 and tool change time (t_c) is 1 min, the machining cost/piece is evaluated using Eq. (4.7) is \$0.47.

Now machining is carried out at other cell centroid conditions that satisfy surface roughness by reducing *v* while keeping *f* constant (*i.e.*, scanning horizontally); machining cost is evaluated. The cost of machining is minimum at 'medium-low' cell (centroid cutting condition: 190 m/min, 0.08 mm/rev) and it is \$0.42. Further search is initiated by varying *f* with *v*=190 m/min (*i.e.*, scanning vertically). The cost of machining is minimum at 'medium-medium' cell, providing the rule: "If *v* is medium and *f* is medium then machining cost in minimum". The machining cost/ piece is \$0.29 at centroid cutting condition of 190 m/min and 0.16 mm/rev. The results of different cutting conditions are presented in Table 5.16.

Table 5.16. Results of different cutting conditions (1st iteration)

Cell num-ber	Centroid cutting conditions		Surface roughness obtained ($\hat{R}_a$) (μm)	Accepted or not?	Cost of machining (*C*) (\$)
	v(m/min)	*f* (mm/rev)			
1	223 (High)	0.24 (High)	3.12	No	-
2	223 (High)	0.16 (Medium)	2.80	No	-
3	223 (High)	0.08 (Low)	2.14	Yes	0.47
4	190 (Medium)	0.08 (Low)	2.20	Yes	0.42
5	157 (Low)	0.08 (Low)	2.25	Yes	0.49
6	**190 (Medium)**	**0.16 (Medium)**	**2.38**	Yes	**0.29**
7	190 (Medium)	0.24 (High)	3.14	No	-

Figure 5.19 shows the division of search domain into 9 cells. The black dots show selected machining condition and shaded portion (Cell 6) shows minimum cost zone obtained in first iteration. The new search is now initiated in the identified (medium-medium) cell.

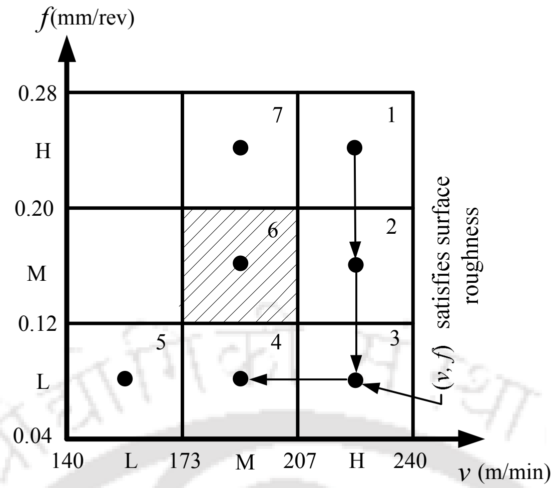


Figure 5.19. The minimum cost zone (1st iteration)

The search domain is now reduced to $173 \le v \le 207$ and $0.12 \le f \le 0.2$ and second iteration is initiated. The new domain is divided into 4 cells with the linguistic sub division being 'low' and 'high' as shown in Figure 5.20. Machining at 'high-high' cell with cutting condition as (199 m/min, 0.18 mm/rev) obtains surface roughness value of 2.28 μm and tool life is found to be 4.0 min. The constrained machining cost of second iteration is $0.19.

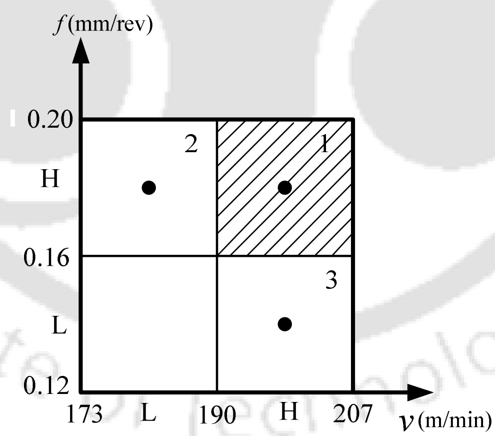


Figure 5.20. The minimum cost zone (2nd iteration)

The results of different cutting conditions in search of minimum machining cost/piece are presented in Table 5.17. The minimum cost zone of second iteration (*i.e.*, optimum zone) is $190 \le v \le 207$ and $0.16 \le f \le 0.2$. The centroid cutting condition (199 m/min, 0.18 mm/rev) is the optimum.

Table 5.17. Results of different cutting conditions (2nd iteration)

Cell No.	Centroid cutting conditions		Surface roughness obtained ($\hat{R}_a$) (μm)	Accepted or not?	Cost of machining (C) ($)
	v(m/min)	f (mm/rev)			
1	**199 (High)**	**0.18 (High)**	**2.28**	Yes	**0.19** (Minimum)
2	182 (low)	0.18 (High)	2.32	Yes	0.24
3	199 (High)	0.14 (Low)	2.20	Yes	0.21

Machining second time at optimum cutting condition, the tool fails after 14 components and tool life is found as 4.3 min; total machining cost/piece is \$0.20. The average cost/piece at optimum cutting condition is \$0.195**.** In the optimum zone, experiments are carried out at a number of points varying both cutting velocity and feed. In this work, due to availability of discrete feed values in the machine, only cutting velocity is varied here. Figure 5.21 shows the various points in which machining are carried out at optimum zone. At each cutting condition machining is carried out twice, till tool fails and average cost/piece is evaluated. The results are shown in Table 5.18.

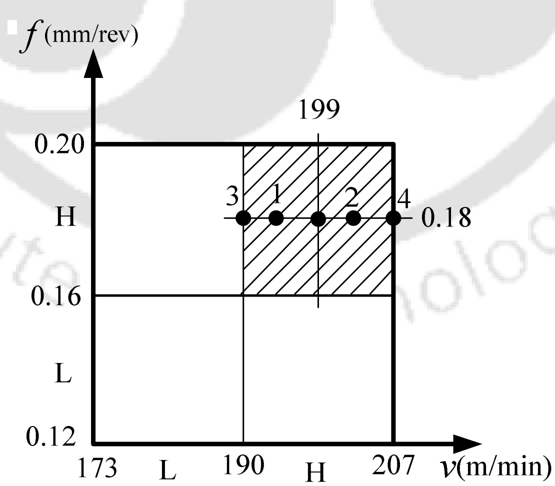


Figure 5.21. Machining at optimum zone

Table 5.18. Results of different cutting condition

Serial number	Cutting conditions (v, f) (m/min, mm/rev)	Surface roughness obtained ($\hat{R}_a$) (µm)	Tool life Obtained (T) (min)	Average cost/piece (C) ($)
1	195, 0.18	2.28	5.0	0.185
		2.30	4.6	
2	203, 0.18	2.26	4.0	0.23
		2.28	3.6	
3	190, 0.18	2.34	4.0	0.21
		2.32	3.8	
4	207, 0.18	2.24	2.9	0.265
		2.20	3.1	

The minimum average cost/piece is at point 1 (*i.e.*, $0.185). It is less than the average cost/piece at optimum cutting condition (*i.e.*, 199 m/min, 0.18 mm/rev), which is $0.195. The minimum average cost point is chosen as 'true optimum' and is corresponding to point 1. The cutting condition is (195 m/min, 0.18 mm/rev). Hypothesis testing is carried out to assess the repeatability of the results.

5.6.2 Test of Significance using Student's *t*-test

For small sample size the test of significance is carried out based on Student's *t*-test. Replicate experiments are carried out at 'true optimum' cutting conditions to evaluate tool lives and total machining cost. Machining five times the tool lives are 4.3, 5.0, 4.3, 4.3, and 4.7 min; the total machining cost/piece are $0.21, $0.18, $0.21, $0.21, and $0.195 respectively. The five data sets are considered as a sample. The mean cost is $0.20 and standard deviation is $0.01. The results are shown in Table 5.19.

Table 5.19. Results of replicated experiments

Cutting condition (v, f): (195 m/min,0.18 mm/rev) Length of machining (L): 100 mm; Limit of surface roughness: 1.6–2.4 μm			
Experiment number	Number of jobs turned	Tool life (T) (min)	Total machining cost/piece (C) ($)
Replicate 1	14	4.3	0.21
Replicate 2	16	5.0	0.18
Replicate 3	14	4.3	0.21
Replicate 4	14	4.3	0.21
Replicate 5	15	4.7	0.195
Sample size (n): 5 ; Mean cost (μ): $0.20 Standard deviation (σ): $0.01			

The Student's t-test is used to compare the significance between the cutting conditions. The t-statistic for the test of significance is given by

$$t = \frac{\bar{x} - \mu}{\sigma / \sqrt{n}}, \tag{5.13}$$

where μ is the population mean. At 95% confidence level, for 2 tailed and 4 degree of freedom the t_{critical} is 2.776. This gives $\bar{x}$ in the range of 0.19 and 0.21. Hence, all cutting conditions in which cost/piece varies from $0.19 to $0.21 are not significantly different from optimum cutting condition and machining can be performed at these points. The data obtained can be stored to train a neural network.

5.7 A Note on Tool life Enhancement Strategy with Cloud Support

In finish turning, tool life is evaluated based on two different criterion, maximum surface roughness criterion and maximum flank wear criterion. If the tool fails due to maximum flank wear, it cannot be reused. However, if tool fails based on the maximum surface roughness criterion, then there is a possibility to enhance its life by operating at changing cutting condition. Merdol and Altintas (2008) proposed online tuning of cutting conditions to maximize metal removal rate satisfying the machining constraints. The preprocess optimization provides allowable depth and width of cut satisfying chatter constraint and post process optimization tunes only feed rate. They have tested the algorithm in machining a helicopter gear box cover.

Sarma and Dixit (2009) presented environment-friendly strategies for efficient utilization of cutting tools in finish turning process. They conducted a number of experiments on the dry and air-cooled turning of steel. The tool consumed in air cooled turning is used for dry turning. The proper selection of reduced feed rate improves the surface finish value and it is decided without increasing the cost. They also tested the proposed strategy in the shop floor. Let f_{opt} and v_{opt} are the optimum feed rate and cutting velocity and T be the tool life at optimum cutting condition. After tool fails the feed is reduced to f_1 and the tool life obtained is T_1.

The cost/piece corresponds to optimum cutting condition (C_{opt}) is obtained by using Eq. (4.7). While reusing the tool operating at reduced feed rate the cost /piece is given by

$$C_1 = C_0 \frac{\pi LD}{f_1 v_{opt}} \left(1 + \frac{t_{\mathrm{c}}}{T_1}\right). \tag{5.15}$$

The reusing of the tool will be economical if $C_1 < C_{opt}$. The required condition for economical machining is

$$C_0 \frac{\pi LD}{f_1 v_{opt}} \left(1 + \frac{t_{\mathrm{c}}}{T_1}\right) < C_0 \frac{\pi LD}{f_{opt} v_{opt}} \left(1 + \frac{t_{\mathrm{c}}^*}{T}\right) \tag{5.16}$$

or

$$f_1 > \frac{f_{opt}\left[1 + \frac{t_c}{T_1}\right]}{1 + \frac{t_c^*}{T}} \tag{5.17}$$

Since t_{c} is very small fraction of the tool life, then the value of the reduced feed rate (f_1) that follows the Eq. (5.17) will be economical.

$$f_1 > \frac{f_{opt}}{1 + \frac{C_{tc}/C_0}{T}} \tag{5.18}$$

Here at optimum cutting condition (195 m/min, 0.18 mm/rev) tool fails at 4.3 min. Taking C_0= \$0.07/min, C_{tc}= \$2.5, f_{opt}=0.18 mm/sec, T=4.3 min, we get f_l>0.02 mm. Therefore, the reduced feed rate should be above 0.02 mm/rev but below the optimum value. It is safe to operate at lower value at which the surface roughness is expected to be well within the limit. The tool reused at feed 0.04 mm/rev produces 16 components and tool life is found as 22.4 min. The actual surface roughness of the components is in the range of 1.62–2.36 µm. The cost/piece is \$0.26. Alternatively, if some one operates at any other feed rate in the range, the cost/piece will be different. Over long period of time, number of data is generated, which can be stored on cloud. Once enough datasets are obtained, optimum feed in the range can be decided.

5.8 Summary

In this Chapter, a fuzzy set based optimization methodology has been proposed that optimizes general engineering problems. The optimization procedure is demonstrated and tested with two standard test problems and the results are compared. The application of fuzzy set based optimization methodology is demonstrated with number of examples. The online optimization process generates a lot of real life data which can be used to developing predictive models and online learning of the machining process in a cloud computing based optimization system. The experimental validation of online machining optimization based on fuzzy set approach optimization procedure is also presented. The machining experiments are carried out on cold rolled steel work piece using TiN coated carbide tool. The strategy also acknowledges the presence of statistical variation in the process. Student's t-test is performed to compare the significance between the cutting conditions. The proposed optimization strategy is expected to be efficient for the problem with a large number of design variables. The strategy of reusing the failed tool by modified cutting conditions is also implemented on the shop floor.

Chapter 6

Development of Interactive Machining System for Cloud Computing

6.1 Introduction

The optimum selection of machining parameters and its successful implementation in shop floor is very important for manufacturing industries. Recently, cloud computing evolved from web based technologies is gaining popularity now. The proposed online optimization strategy can be combined with web or cloud computing based machining system. The manufacturers at different locations will be using the methodology and gives machining feed back to the system. Therefore, it is essential to develop the technique for easy implementation of the procedure in the shop floor.

In literature, Wang and Jawahir (2004) have developed a web-based, user interactive expert system for optimization of single-pass milling process for obtaining optimum cutting conditions using genetic algorithm. The system allows the user to access relevant data base. According to user's request/input the results are evaluated and HTML documents are generated to send the corresponding information to the user. Wu and Liao (2005) developed an internet-based machining parameter optimization and management system using genetic algorithm for high speed machining process. Zheng *et al*. (2008) developed a web-based machining parameter selection system for agile turning which provides user appropriate selection of machine tools, cutting tools, and cutting parameters for reduction in product life cycle cost and increase in product quality.

In this thesis, an interactive menu driven system is developed for easy shop floor implementation of the proposed heuristic based online optimization procedure. A number of user interactive screens are designed and is coded in MATLAB®

(R2010a) Graphical User Interface (GUI). The screen accepts machining informations from the user as well as displays the cutting conditions or instructions. The machine tool operator will follow the instruction and operate the machine tool accordingly. Thus, the user does not require to aware of the heuristic algorithmic procedure for optimizing the machining process. In this chapter, the development of interactive screens for the implementation of online optimization procedure is presented in Section 6.2. Section 6.3 describes the interactive menu driven system developed for finish machining process in a tabular form. Also, typical screenshots are presented. The illustrative examples for the interactive menu driven system for online optimization of multipass turning and end milling processes are presented in Section 6.4. The summary of the Chapter is presented in Section 6.5.

6.2 Methodology

Based on online heuristic procedure the machining informations are transacted between the user and the system. The interactive menus are suitably designed for easy interaction by the user. The different menus are interlinked so that the user is able to receive and input the required details. The presentation layer *i.e.*, namely GUI of MATLAB® (R2010a) is used in this work. The basics of the menu development are discussed in Section 3.4 of Chapter 3. The scheme for online optimization of multi pass machining process is attempted here.

The user enters basic informations about the process to be optimized. The input informations include: (i) tool and work material, (ii) workpiece or cutter diameter, (iii) length of machining, (iv) total depth of stock to be removed, (v) maximum allowable cutting force, (vi) maximum allowable surface roughness and (vii) limits of cutting parameters. On receiving the informations, the system suggests initial cutting conditions for machining. The user implements the same and initiate online optimization process. The result of machining process such as actual cutting force and surface roughness are entered into the system. Based on this improved cutting conditions are suggested. The process optimization is to minimize the cost which is of two variables problem. Three initial cutting conditions are required to initiate the optimization process. Three set of initial cutting conditions ((v_1, f_1), (v_2, f_2) and (v_3, f_3)) that satisfy the constraints are identified. The corresponding machining cost is evaluated optimization is initiated. The parameters are optimized for roughing as well as finishing process while producing the components.

'Pop-up' windows are used for entry of workpiece and tool material selection. It facilitates easy selection of the appropriate material from the available list through pull down menu, using mouse click. 'Edit text' windows are used for entry of data such as work material diameter, length, maximum allowable surface roughness, constraint values of the process parameters, *etc*. Once all the entries are completed 'push button' switch with label 'PRESS' is used by the user to continue with the present menu layer in the online optimization process. The details of the interactive menu screen and screenshot picture are presented in the subsequent section.

6.3 An Operator Interactive Machining System

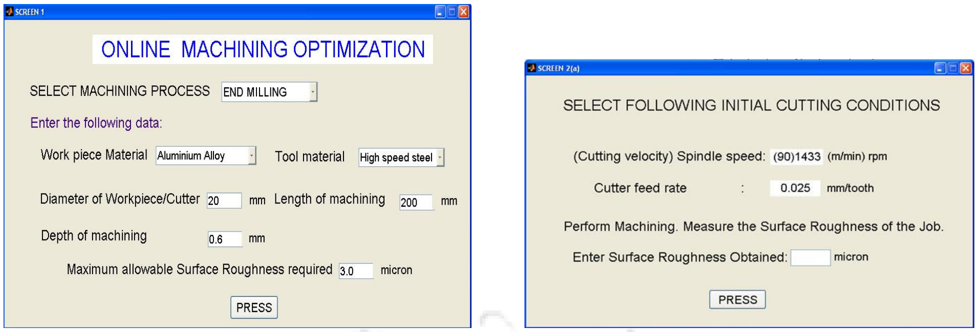
The interactive machining system is designed for easy shop floor implementation of heuristic based online optimization of the cloud computing system. Table 6.1 shows the description of the menu development for online optimization of finish machining process. The input/output information provided into/by the system with appropriate details and screen number is described. The screens are suitably designed to follow algorithmic procedure used for optimization.

The screen 1 accepts the information from the operator and screen 2(a) display initial cutting condition to be used for machining. Based on selection of cutting speed the required spindle speed is displayed for easy implementation of cutting condition by the operator. The actual surface roughness obtained is entered into the system in the screen. The system verifies the surface roughness and if it is acceptable within the limits the screen 2(c) displays the instruction for record of the tool life. Otherwise, the cutting condition with modified feed is displayed in screen 2(b) for further machining. Similarly three set of cutting conditions with actual surface roughness and tool life are obtained. The system evaluates the function values and initiates simplex optimization. The iterative cutting conditions are displayed and feed back is accepted in the screen 5. The optimum cutting parameters are displayed finally in screen 6. Screen 7 displays the report generation and the real machining data sets are stored in the cloud database. The typical screen snapshots of the developed system are shown in Figure 6.1.

Table 6.1. Details of interactive screen for online finish machining process

Serial number	Description	Data recorded/displayed	Screen details
1	Accept the details of machining process.	Work material, tool material, required surface roughness, work diameter, depth of machining and machining length	SCREEN 1
2	Display first set of cutting condition to perform machining and obtain actual surface roughness	Display cutting speed (v_1) and feed (f_1). Record actual surface roughness ($\hat{R}_a$) obtained.	SCREEN 2(a)
3	If $0.8R_a^t > \hat{R}_a$, display modified cutting condition. Record $\hat{R}_a$ of the cutting condition. (It continues till $0.8R_a^t < \hat{R}_a < 1.2R_a^t$ achieved).	Display the modified feed.	SCREEN 2(b) . . . .
4	Instruct to machine continuously and record tool life.	Record tool life (T_1).	SCREEN 2(c)
5	If $\hat{R}_a > 1.2R_a^t$, Display message.	Display "Surface roughness cannot be achieved with this set of work tool combination".	SCREEN 2(d)

6	Display second set of cutting condition and record $\hat{R}_a$ of the cutting condition.	Display cutting speed (v_2) and feed (f_2). Record actual surface roughness ($\hat{R}_a$) obtained.	SCREEN 3(a)
7	On satisfying desired limit, display instruction for machining continuously and record tool life.	Record tool life (T_2).	SCREEN 3(b)
8	Display third set of cutting condition and record $\hat{R}_a$ of the cutting condition	Display cutting speed (v_3) and feed (f_3). Record actual surface roughness ($\hat{R}_a$) obtained.	SCREEN 4(a)
9	On satisfying desired limit display the instruction to machine continuously and accept tool life.	Record tool life (T_3).	SCREEN 4(b)
Simplex optimization is performed			
10	Display iterative cutting conditions to be operated and record its response.	Display v and f. Record $\hat{R}_a$ and T.	SCREEN 5
11	Display optimum cutting condition and record its response.	Display v_{opt} and f_{opt}. Record $\hat{R}_a$ and T at optimum condition.	SCREEN 6
12	Generate report. Store it in a database.	Record real machining data sets (v, f, $\hat{R}_a$, T, F_{min}), optimum data set (v_{opt}, f_{opt}, $\hat{R}_a$ T, F_{min}) and machining time/job (T_m).	SCREEN 7



(a) Screen 1 **(b)** Screen 2

Figure 6.1. Snapshots of the interactive screen

This interactive method of online machining methodology optimizes the cutting condition in subsequent machining of component. The surface roughness of the job produced is well within limit. The report generation screen which displays the real machining data sets *viz.*, cutting speed (v), feed (f), depth of cut (d), actual surface roughness obtained ($\hat{R}_a$), tool life (T), function value (F) and machining time. Similarly, the screen details for rough machining process also developed considering the different constraints employed.

6.4 Examples

In this section the web implemented interactive machining system for online optimization of multi pass machining process is demonstrated. The optimization of multi pass machining process constitute: (i) optimization of roughing passes and (ii) optimization of finishing pass. The process parameters *i.e.*, cutting speed and feed for roughing and finishing passes are optimized to minimize the machining cost. In this section the illustrative examples for multi pass machining of two common machining processes *viz.*, turning and milling process is presented.

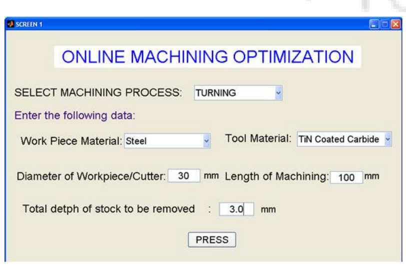
6.4.1 Interactive Based Online Optimization of Multipass Turning Process

Consider a multipass turning process to be optimized for obtaining optimum process parameters. Let total depth of stock to be removed is 3.0 mm and allowable surface roughness value (R_a^*) is 3.25 μm. The maximum allowable cutting force is 1000 N. The variable bounds of the process are $140 \le v\left(\mathrm{m/min}\right) \le 238$,

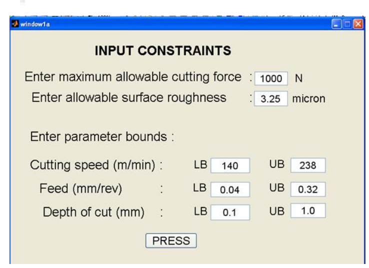
$0.04 \le f\,(\text{mm/rev}) \le 0.32$, $0.1 \le d\,(\text{mm}) \le 1.0$. The input information for the optimization problem is shown in Table 6.2. Figure 6.2 shows snapshots of the input screens. For optimizing the problem actual machining has not been carried out. Instead a code was written for making a semi virtual lathe and the design is presented in Section 4.2. Actual machining experiments are replaced by simulated results.

Table 6.2. Input details

Description	Typical Input data
Machining details:	
Process	Turning
Workpiece material	Steel (0.4 % C)
Tool material	TiN coated Carbide
Diameter of workpiece	30 mm
Length of machining	100 mm
Maximum depth of stock to be removed	3.0 mm
Constraints details:	
Allowable cutting force	1000 N
Allowable surface roughness	3.25 μm
Cutting speed (Lower bound)	140 m/min
Cutting speed (Upper bound)	238 m/min
Feed (Lower bound)	0.04 mm/rev
Feed (Upper bound)	0.32 mm/rev
Depth of cut (Lower bound)	0.1 mm
Depth of cut (Upper bound)	1.0 mm



(a) Machining details

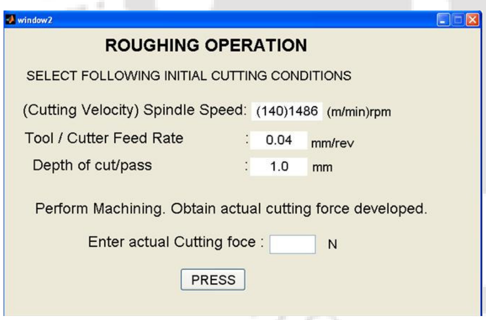


(b) Constraints details

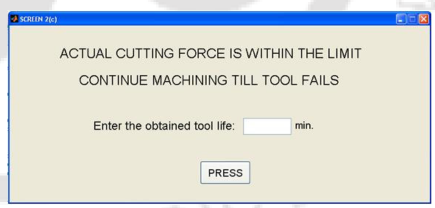
Figure 6.2. Snapshots of screen for input details

The online optimization roughing process starts with selection of initial cutting condition suitably so that actual cutting force is within limit. The depth of cut in all roughing process is taken as equal. The number of passes is decided such that the maximum depth of cut constraint is not violated. In the present problem depth of cut for finishing and two roughing passes comes out as 1.0 mm.

The formulation of optimization problem is presented in Section 4.6.1. The cutting speed and feed of roughing passes are optimized using Eq. 4.18. As the optimization function is a two variable (v, f) problem, three initial machining conditions are generated and is optimized using simplex search method. The three cutting conditions are selected as (v_1, f_1), (1.1v_1, f_1) and (1.1v_1, 1.1f_1). For the present problem initial cutting condition is v_1 (= v_{min}) as 140 m/min and f_1 (= f_{min}) as 0.04 mm/rev is chosen. The cutting force is obtained as 386.3 N, which is within limit. The machining is continued with first set of cutting condition (v = 140 m/min, f = 0.04 mm/rev, d = 1 mm/pass) till tool fails. The obtained tool life 315.2 minute is entered in the system. The snapshots of the interactive screens are as shown in Figure 6.3. The display of spindle speed along with cutting speed facilitates operator for quick setting of cutting parameters.



(a) Display of initial cutting conditions



(b) Tool life entry

Figure 6.3 Snapshots of the screen during roughing process.

As different machine tool operators employ this methodology, the real tool life data are supplied to the clouds. These data are fine tuned at the server side by adoption of cloud computing feature and help to optimize the machining process. The feedback is provided to the users *via* cloud service. The use of cloud computing combines multiple machining data and provide comprehensive result to the users.

As an example consider tool life data of 12 machine tool operators, in which 3 machines predicted the tool life that follow Taylor's extended tool life function as

$$vT^{0.2}f^{0.15}d^{0.15} = 273 \tag{6.1}$$

and the other 9 machines follow tool life function as

$$vT^{0.2}f^{0.18}d^{0.12} = 276\,. \tag{6.2}$$

For equal depth of cut in all roughing passes the tool life become function of T(f,v). The rough machining cost per piece is obtained from Eq. 4.18 of Section 4.6.1 of Chapter 4. Let C_{R1} and C_{R2} be the rough machining cost per piece obtained using tool life data with 3 machines and 9 machines respectively. Now substituting m=2 passes and d_R=1.0 mm/pass Eq. 4.18 is obtained as

$$C_{R1} = \left[C_0 \frac{\pi L}{v_R f_R}(2D_0 - 2) + (C_0 t_c + C_{tc}) \frac{\pi L v_R^4 f_R^{-0.25}}{273^5}(2D_0 - 2) + 3C_0 t_s \right] \tag{6.3}$$

and

$$C_{R2} = \left[C_0 \frac{\pi L}{v_R f_R}(2D_0 - 2) + (C_0 t_c + C_{tc}) \frac{\pi L v_R^4 f_R^{-0.1}}{276^5}(2D_0 - 2) + 3C_0 t_s \right], \tag{6.4}$$

where C_0 is the operating cost in \$/min, and C_{tc} is the tool change cost in \$.

Based on probabilities of tool life functions of 0.25 and 0.75 the roughing cost per piece is

$$\begin{aligned} C_R = 0.25 &\left[C_0 \frac{\pi L}{v_R f_R}(2D_0 - 2) + (C_0 t_c + C_{tc}) \frac{\pi L v_R^4 f_R^{-0.25}}{273^5}(2D_0 - 2) + 3\,C_0 t_s \right] \\ +0.75 &\left[C_0 \frac{\pi L}{v_R f_R}(2D_0 - 2) + (C_0 t_c + C_{tc}) \frac{\pi L v_R^4 f_R^{-0.1}}{276^5}(2D_0 - 2) + 3C_0 t_s \right]. \end{aligned} \tag{6.5}$$

Substituting L=100 mm, D_0=30 mm, C_0= 0.5 \$/min, C_{tc}= 2.5 \$/min, t_c=1.5 min, t_s= 0.51min/pass, Eq. (6.5) is simplified as

$$C_R = \frac{9.10}{v_R f_R} + \frac{14.79 v_R^4 f_R^{-0.25}}{273^5} + \frac{44.39 v_R^4 f_R^{-0.1}}{276^5} + 0.765. \tag{6.6}$$

To optimize the function, three set of cutting conditions are guessed as (140.0, 0.04), (154.0, 0.04) and (154, 0.05) respectively. The evaluated costs are \$2.41, \$2.28 and \$1.98. Simplex search optimization (Rao, 1996) is used for optimization. Normally, it takes 3 iterations to reach optimum. The iterative cutting conditions and roughing costs are shown in Table 6.3. The cutting force obtained is within the limit of 1000 N. The optimum process parameters are obtained as v_{opt} = 142.0 m/min and f_{opt} = 0.14 mm/rev. The optimum cost for roughing/piece is $1.24.

Table 6.3. Iterative results of optimization for roughing process

Iteration number	Suggested iterative cutting conditions (v_R (m/min)	d_R mm	f_R) mm/rev)	Actual cutting force (F_{act}) (N)	Roughing cost per piece (C_R) ($)
1	173.0	1.0	0.06	427.2	1.69
2	154.0	1.0	0.08	458.8	1.53
3	142.0	1.0	0.14	527.1	1.24

The proposed scheme obtains reliable optimum cutting parameters by its direct implementation without deviating cutting force constraint. Figure 6.4 shows the final display screen for roughing process showing the real machining data clouds stored in database of the system.

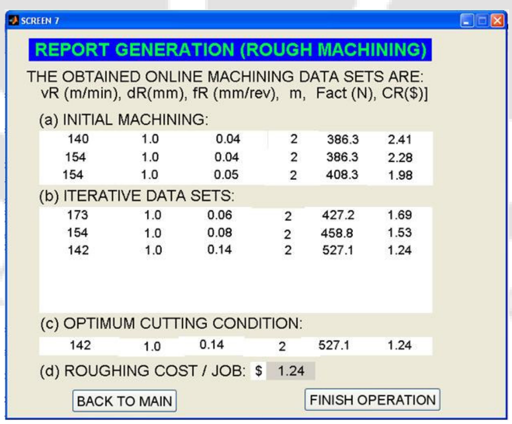


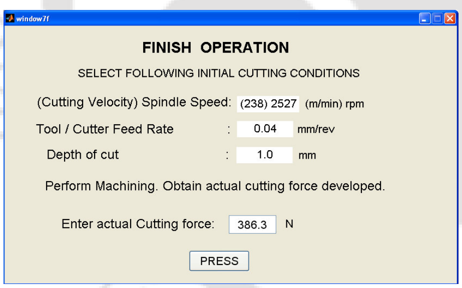
Figure 6.4. Final display for roughing process

In optimization of finish turning process the initial cutting condition are selected as v_1 (=v_{max}), and f_1 (= f_{min}) and the constant depth of cut is equal to d_{max}. For the present problem v_1, f_1, and d_1 are 238 m/min, 0.04 mm/rev, and 1 mm respectively. The allowable surface roughness value is (R_a^*) 3.25 μm. Considering some statistical variation the actual surface roughness ($\hat{R}_a$) is considered between the lower and upper limits such that $0.8R_a^t < \hat{R}_a < 1.2R_a^t$ where R_a^t may be taken as $0.8R^*$. If $0.8R_a^t > \hat{R}_a$, the feed is increased using the following relation:

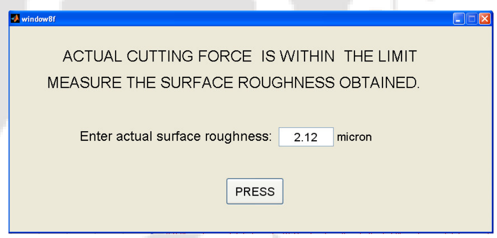
$$(f_1)_{\text{new}} = \left(\frac{R_a^t}{\hat{R}_a}\right)^{1.5} f_1, \tag{6.7}$$

where f_1 is feed in mm/rev. The exponent 1.5 has been chosen based on some experimental study. If $\hat{R}_a > 1.2R_a^t$, the system displays "Surface roughness cannot be achieved with this set of work tool combination".

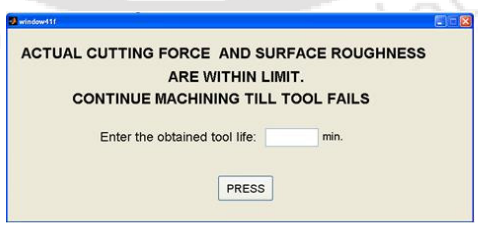
The actual surface roughness ($\hat{R}_a$) may lie between 2.08 and 3.12 μm. The machining of component is performed and the obtained cutting force and CLA surface roughness are 386.3 N and 2.12 μm respectively. If surface roughness falls below lower limit, feed is increased suitably with v remaining same. Now machining is continued till tool fails and obtained tool life data is entered in the system. The snapshots of the interactive screens during finish machining process are shown in Figure 6.5.



(a) Display of initial cutting condition



(b) Surface roughness entry



(c) Tool life entry

Figure 6.5. Snapshots of the screen during finish machining process

Based on probabilities of tool life functions of 0.25 and 0.75 the finishing cost per piece is

$$C_F = 0.25\, C_{F1} + 0.75\, C_{F2} \tag{6.8}$$

where C_{F1} and C_{F2} are finishing costs corresponding to tool lives that follows Eq. (6.1) and Eq. (6.2) respectively. Substituting L=300 mm, D_f=24 mm, C_0=0.5 \$/min, C_{tc}= 2.5 \$/min, t_c=1.5 min, t_s= 0.51min/pass, Eq. (6.10) is simplified as

$$C_F = \frac{4.08}{v_F f_F} + \frac{6.63 v_F^4 f_F^{-0.25}}{273^5} + \frac{19.9 v_F^4 f_F^{-0.1}}{276^5} \tag{6.9}$$

To optimize the function (Eq. 5.28) three set of cutting conditions are guessed as (238, 0.04), (214, 0.04) and (214, 0.05) respectively. The evaluated costs are \$0.51, \$0.53 and \$0.44. Simplex search optimization takes 4 iterations to reach optimum. The iterative cutting conditions and evaluated costs are shown in Table 6.4. The surface roughness obtained is within the limit of 2.08 and 3.12 μm. The optimum process parameters for finish machining are obtained as v_{opt} = 218.4 m/min and f_{opt} = 0.10 mm/rev. The optimum cost for finishing/piece is \$0.24. The job loading/unloading cost= $C_0 \times T_L$=0.5×0.75= \$0.375. Thus, total production cost /piece is \$1.86.

Table 6.4. Iterative results of optimization for finishing process

Iteration number	Suggested iterative cutting conditions			Surface roughness obtained ($\hat{R}_a$)(μm)	Finishing cost per piece (C_F) (\$)
	(v_F (m/min	d_F mm	f_F) mm/rev)		
1	148.2	1.0	0.06	3.02	0.47
2	196.5	1.0	0.08	2.52	0.30
3	163.0	1.0	0.12	2.98	0.28
4	218.4	1.0	0.10	2.56	0.24

The proposed scheme obtains reliable optimum cutting parameters by its direct implementation without causing any loss/rejection of components due to surface finish. With the availability of more and more real life machining data the system database enriches with information and helps to develop predictive models for machining optimization. It is also fine tuned at server side based on feedback information provided by the user.

6.4.2 Interactive Based Online Optimization of Multipass End milling Process

Consider the optimization of multipass end milling process. Aluminium alloy workpiece is machined with 20 mm diameter high speed steel (HSS) end mill cutter. The cutter has 4 number of tooth (z). Length of machining is 100 mm. Total depth of stock to be removed is 5 mm and allowable surface roughness value (R_a^*) is 2.75 μm. Maximum allowable cutting force is 500 N. The machining is performed on semi virtual milling (Appendix B) which consists of the following modules: (i) surface roughness prediction module based on NN, (ii) tangential cutting force prediction module based on empirical relation, and (iii) tool life module based on empirical relation. NN module is developed using an experimental data from the work of Lou and Chen (1999) for end milling of aluminum alloy using HSS end mill cutter. The tangential cutting force is evaluated from empirical relation obtained from experimental data presented in (Wan *et al.*, 2012). Tool life is evaluated by the relation proposed by Tolouei-Rad and Bidhendi (1997) and it is given as

$$T = \frac{60}{Q}\left(\frac{C_1\left(\frac{0.2d}{f'z}\right)^g}{(df'z)^w v}\right)^{1/n} \tag{6.10}$$

where T is the tool life in minutes, v is the cutting speed in m/min, C_1 is the constant for 60 minutes of tool life when the area of cut is 1 mm^2, f' is the feed in mm/tooth, z is number of tooth, d is the depth of cut in mm, and g, w, and n are exponents for different tool and work material combination. The factor Q is contact proportion of cutting edges with work piece per revolution.

The ranges of cutting parameters are as follows:

Cutting speed (v): 45–90 m/min

Feed (f') : 0.025–0.23 mm/tooth

Depth of cut (d) : 0.25–1.27 mm

The optimization problem is to determine optimum cutting velocity and feed/tooth through online machining process for minimizing total production cost based on cloud computing feature. The total production cost is given by

$$C = C_R + C_F + C_L = \left[C_0 \frac{\pi LDm}{1000 v_R f_R' z} + (C_0 t_c + z C_{tc}) \frac{\pi LDm}{1000 v_R f_R' z T_R} + C_0 (m+1) t_s \right] + \left[C_0 \frac{\pi LD}{1000 v_F f_F' z} + (C_0 t_c + z C_{tc}) \frac{\pi LD}{1000 v_F f_F' z T_F} \right] + C_0 T_L \qquad (6.11)$$

The online optimization of roughing process starts with selection of initial cutting condition suitably so that actual cutting force is within limit. The machining is initialized with v_1 (= v_{min}) as 45 m/min and f_1 (= f'_{min}) as 0.025 mm/tooth. The radial depth of cut (d) is 1.24 mm and number of passes are 3. The obtained cutting force is 106 N, which is within limit. Thus machining is continued with first set of cutting condition (v_1=45 m/min, f_1=0.025 mm/tooth) till tool fails and obtained tool life data is entered in the system.

Consider the tool life provided by 8 machine tool operators. Two machines predicted tool life that follows tool life function as

$$T_1 = \frac{60}{0.29} \left(\frac{33.98 \left(\frac{0.2d}{f'z} \right)^{0.14}}{(df'z)^{0.35} v} \right)^{1/0.15} \qquad (6.12)$$

The tool life predicted by other 6 machines follows tool life function as

$$T_2 = \frac{60}{0.29} \left(\frac{33.98 \left(\frac{0.2d}{f'z} \right)^{0.16}}{(df'z)^{0.38} v} \right)^{1/0.15} \qquad (6.13)$$

Considering the probability of tool life function of 0.25 and 0.75 the rough machining cost per piece is given by

$$C_R = 0.25\, C_{R1} + 0.75\, C_{R2} \qquad (6.14)$$

With the constant values of costs, length and diameter, the objective function to minimize roughing cost is expressed as

$$C_R = \frac{2.355}{v_R f_R'} + 2.115 \times 10^{-9} v_R^{5.67} f_R'^{2.2.7} + 3.606 \times 10^{-10} v_R^{5.67} f_R'^{2.6} + 1.02\,. \qquad (6.15)$$

The optimization is initialized with three set of cutting conditions as (45, 0.025), (49.5, 0.025), and (49.5, 0.028). Optimum cutting parameters v_{opt}=53.4 m/min and

$f'_{z(opt)}$=0.12 mm/tooth is obtained in 4 iterations. The optimum cost for roughing/piece is \$1.51.

The optimization of finish milling process considers the initial cutting condition as v_1 (= v_{max}), and f_1 (= f'_{min}) with depth of cut equal to d_{max}. The initial cutting parameters are 90 m/min, 0.025 mm/tooth, and 1.27 mm. The machining of component is performed and the obtained cutting force and CLA surface roughness are 106 N and 1.43 μm respectively. The limit of the actual surface roughness value ($\hat{R}_a$) is between 1.76 and 2.64 μm. Now the feed is increased following Eq. (6.7) with cutting velocity remaining the same. This increases the surface roughness value and thus first set of cutting condition is obtained as 90 m/min and 0.042 mm/tooth. The CLA surface roughness is 2.26μm. Considering the probabilities of tool life function as 0.25 and 0.75 the finish machining cost per piece is given by

$$C_F = 0.25C_{F_1} + 0.75C_{F_2} = \frac{0.785}{v_F f'_F} + 7.2\times10^{-10} v_F^{5.67} f_F'^{\,2.27} + 4.36\times10^{-10} v_F^{5.67} f_F'^{\,2.6} \tag{6.16}$$

The optimization is initialized to minimize the function Eq. (5.35) with three set of cutting conditions as (90, 0.042), (81, 0.052) and (73, 0.052). Optimum cutting parameters v_{opt}=63.2 m/min and $f'_{z(opt)}$ = 0.11 mm/tooth is obtained in 3 iterations. The optimum cost for finishing/piece is \$0.21. Total production cost/piece is \$2.10. The cutting force and surface roughness obtained at optimum cutting condition are 395 N and 2.46 μm respectively. Figure 6.6 shows display of optimum cutting conditions for finish milling process.

The iterative cutting conditions are provided by the system in a user friendly interactive GUI screen based on optimization source code written in MATLAB® (R2010a) runs at Pentium-4 with 512 MB RAM. In the proposed scheme optimization is carried out during machining by selecting cutting conditions suitably.

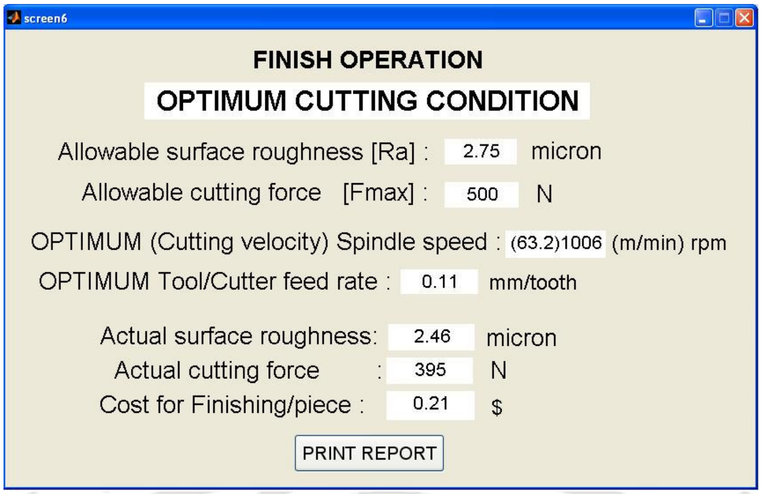


Figure 6.6. Display of optimum cutting parameters for finish milling process

6.5 Summary

In this chapter the development of menu driven interactive machining system for implementation of online machining methodology is presented. The menu based user interactive system is coded in MATLAB® (R2010a) with GUI. The client at different location may download the .EXE file which is kept in CCBOD system and can easily implement at their location for optimizing different machining process. The interactive screens, transacted information, screen numbers are detailed out. The screenshot of the developed GUI screen also presented. The optimization procedure for multi pass machining process is discussed. Examples for optimization of turning and end milling process are presented with screen snapshots. The cloud computing system that uses the probabilistic based approach for optimizing machining parameters based on machining feedback received from various clients is discussed. The developed menu based system has simplified interactive access, easy to use and provides opportunity for sharing the process data information among different manufacturers.

Chapter 7

Application of Soft Computing for Online Learning in a Cloud Computing Based Machining Optimization

7.1 Introduction

Automated manufacturing relies on computer based learning schemes that are able to learn the complex machining process. Researchers employ an artificial intelligence with soft computing tools for learning the complex machining process. Kumar and Dixit (2008) developed an online learning system using finite element method (FEM) for determining the proper traverse speed required to form a sheet in laser forming process. Their work focuses on reducing computational time involved in FEM module by combining the artificial intelligence (AI) module that learns with experience.

In cloud computing based machining optimization the actual experimental data set are stored in server database sent by various clients through clouds. It can be used for online learning and optimizes the process predicting better cutting condition. In this thesis soft computing techniques based online learning of machining process is developed. The algorithm keeps on becoming more and more intelligent by machining variety of different jobs. The proposed online learning methodology consists of three modules:

(i) Artificial Intelligent (AI) module: It consists of NN modelled with multi layer perceptron (MLP) network combined with an optimization algorithm used for predicting the machining parameters. The network is continuously updated and thus it improves its prediction accuracy with time.

(ii) Machining module or virtual simulation module: The machining system implements the dictated cutting condition and obtains the real machining information. In this work, the machining system is substituted by a 'virtual machine' consisting of two modules for predicting 'surface roughness' and 'tool life' for the given set of cutting conditions.

(iii) Optimization module: The module consists of an optimization algorithm that optimizes the process and develops successive cutting conditions for online machining.

The methodology is demonstrated using optimization of finish turning process. It obtains optimum 'cutting speed' and 'feed' for minimizing machining cost. The machining data sets are stored and effectively used for training a neural network of AI system. In this chapter, the scheme of proposed soft computing based online learning system is presented in Section 7.2. The various modules of the system are discussed in detail. Section 7.3 demonstrates the online learning procedure that improves the predictive capability of AI module. Section 7.4 demonstrates the procedure with illustrative examples for optimizing finish turning process. The summary of the chapter is presented in Section 7.5.

7.2 Optimization with Continuous Learning

Optimization of process parameters is an important step in the machining process to achieve economy in machining. Usually in shop floor the skilled operator decides the process parameters by the rule of thumb based on his experience (intelligence) gained from the process. With this back ground an artificial intelligence based online learning methodology is developed in this work.

The present scheme consists of three modules *viz*., AI module, virtual simulation module (that will be replaced by actual machine in the real shop floor) and an optimization module. The prominent features of these modules are discussed in the following subsections. The methodology is as follows. Firstly a highly approximate neural network model combined with optimization algorithm is kept in AI module. In an optimization of finish turning process, for achieving the desired surface finish AI module predicts the required cutting velocity and feed, which serves as initial guess for virtual simulation module. While machining at this condition the surface roughness and tool life are obtained and the condition is optimized for minimizing the cost in optimization module. The optimization module dictates successive cutting conditions till optimum is reached and is implemented in the machining module.

The information obtained during cost optimization process is sent to the AI module. The neural network is continuously updated with this information and thus it improves its prediction accuracy with the time as the varieties of jobs are produced. The optimum cutting condition is determined accurately without consuming much time and resources at the shop floor. Figure 7.1 shows the block diagram of the proposed scheme.

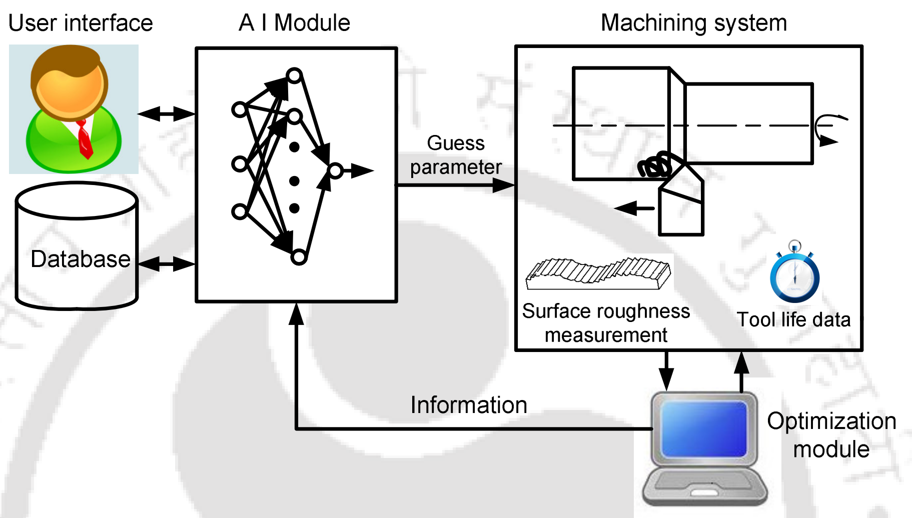


Figure 7.1. Proposed scheme of online learning

7.2.1 Development of AI Module

An AI module is developed using neural network (NN) modelled with multi layer perceptron (MLP) network combined with an optimization algorithm for predicting guess cutting parameters (v and f). Let R_a^* be maximum allowable surface roughness of the job required to be produced while the depth of cut (d) of the work piece remains constant in a finish turning process. The two separate networks are modelled with three input variables: cutting speed (v), feed (f) and depth of cut (d) to predict surface roughness and tool life. The initial training of networks uses all factorial combinations at three levels– low, medium, and high, resulting in $3^3 = 27$ data sets. Initially there is no data available to train the networks. In a particular data group if only a few data were missing, they can be taken either as intervals by the range of its possible values or by estimates from rough-set theory. Ishibuchi *et al.*, (1994) and Hong *et al.*, (2002) have also used similar approaches to identify missing data. In the present case since all data were missing, the following strategy is adopted to obtain the possible output values. The range of input parameters is 140– 240 m/min for cutting velocity, 0.04–0.32 mm for feed, and 0.1–1.0 mm for depth of cut. The range of output variables is taken as 1.9–4.7 μm for surface roughness and 20–220 minute for tool life. With the increase of cutting speed, surface roughness decreases and with the increase of feed and depth of cut, surface roughness increases. The tool life reduces as the cutting speed, feed, and depth of cut increases. These rules form the basis of creating hypothetical data. Table 1 shows the data sets used to train the networks. The trained network is used to find cutting parameters (v and f) to produce the surface roughness value as well as tool life. For the particular value of depth of machining (d) to be carried out the cutting parameters (v and f) are selected from hypothetical datasets. If the predicted value of surface roughness satisfies then the cost function is optimized employing simplex search. The optimized cutting parameters are considered as initial guess predicted by AI module.

7.2.2 Machining System or Virtual Simulation Module

This module is the actual machining system used for production of components. In this work, instead of an actual machine tool, a virtual simulation is carried out using semi virtual lathe presented in Section 4.2. It consists of two modules *viz*., surface roughness (R_a) prediction module based on NN trained with real experimental data sets and tool life evaluation module using an empirical relation. The neural network prediction module is trained with real experimental data set obtained by dry turning of steel using carbide tool from work of Kohli and Dixit (2005). It is used for the prediction of center line average (CLA) value of surface roughness. The tool life prediction module uses an empirical equation based on Taylor's extended tool life relation for evaluating the tool life at particular cutting condition.

The guess cutting parameters are used as initial machining parameters for producing the component. On performing the machining process as per guess cutting condition and with the confirmation of actual surface roughness within control limits, the machining is continued till tool fails and tool life data is obtained based on real time environment and is fed to the optimization module.

Table 7.1. Data set for training NN module

Cutting speed (m/min)	Depth of cut (mm)	Feed (mm/rev)	Surface roughness (μm)	Tool life (min)
140	0.1	0.04	3.10	220
140	0.1	0.18	3.30	206
140	0.1	0.32	3.50	192
140	0.4	0.04	3.70	177
140	0.4	0.18	3.90	163
140	0.4	0.32	4.10	149
140	1.0	0.04	4.30	134
140	1.0	0.18	4.50	120
140	1.0	0.32	4.70	106
190	0.1	0.04	2.50	177
190	0.1	0.18	2.70	163
190	0.1	0.32	2.90	149
190	0.4	0.04	3.10	134
190	0.4	0.18	3.30	120
190	0.4	0.32	3.50	106
190	1.0	0.04	3.70	92
190	1.0	0.18	3.90	77
190	1.0	0.32	4.10	63
240	0.1	0.04	1.90	134
240	0.1	0.18	2.10	120
240	0.1	0.32	2.30	106
240	0.4	0.04	2.50	92
240	0.4	0.18	2.70	77
240	0.4	0.32	2.90	63
240	1.0	0.04	3.10	49
240	1.0	0.18	3.30	34
240	1.0	0.32	3.30	20

7.2.3 Optimization Module

The optimization objective is to minimize the cost of machining. In the proposed scheme of optimization with online learning, the machining is carried out on virtual lathe for the guess cutting parameters. With the confirmation of actual surface roughness within control limits, the machining is continued till tool fails and tool life data is obtained based on real time environment. The computation of machining cost is carried out as follows.

For a finish turning process, if L is the machining length, D is the diameter of the work piece, t_c is the tool change time, T is the tool life for particular cutting condition, C_0 is the operating cost in \$/min, and C_{tc} is the tool change cost in \$ then machining cost per job is given by

$$C_{\mathrm{m}} = C_0 \frac{\pi LD}{fv}\left(1+\frac{t_{\mathrm{c}}}{T}\right)+C_{\mathrm{tc}}\left(\frac{\pi LD}{fvT}\right). \tag{7.1}$$

With the constant values of costs, length and diameter, the objective function to minimize machining cost is expressed by the function F as

$$\text{Minimize } F = \frac{1}{fv}\left(1+\frac{t_{\mathrm{c}}^{*}}{T}\right), \tag{7.2}$$

where $t_{\mathrm{c}}^{*} = t_c + C_{tc}/C_0$. Considering t_c = 1.5 min, the above equation reduces to two-variables (f and v) problem.

The optimization module optimizes the cost function F to obtain optimum cutting parameters. Simplex search (Rao, 1999) is used to optimize the function. The algorithm requires less number of function evaluations (*i.e.*, (n+1) evaluations in n dimensional space) for each iteration and is found suitable for online machining optimization. In the present case, three cutting conditions (v_1, f_1, T_1, F_1), (v_2, f_2, T_2, F_2), and (v_3, f_3, T_3, F_3) are selected with v_1 and f_1 being guess parameters, v_2=0.9v_1, f_2=1.1f_1, v_3=0.9v_2 and f_3=f_2, to optimize the function F as given in Eq. (7.2). Different cutting conditions obtained during each iteration are simulated in virtual simulation module and data obtained is used in optimization module. The source code for the optimization module is written in MATLAB® version 7.1 and computational time takes less than a second in Pentium-4 with RAM 512.

7. 3 Online Learning Procedure

In this section, online learning procedure of AI module is presented. The optimization process first generates the initial guess parameter in AI module. It is used in machining system and machining feed back is submitted to optimization module. The optimization module suggests further cutting conditions and machining is carried out in the virtual simulation module. This procedure continues till optimum cutting parameters are obtained and the surface finish is kept within limits. $0.8R_a^t < \hat{R}_a < 1.2R_a^t$ is considered as control limits where $\hat{R}_a$ and R_a^t are actual surface finish produced and target surface roughness (0.8 times of R_a^*) respectively. This strategy accounts for some statistical variation in surface roughness.

In this process, the simulation is carried out n number of times before it converge to optimum cutting condition, hence n simulation data sets will be generated. These data sets are compared with data sets of neural networks of the AI module. If (v_1, f_1, d_1) and (v_2, f_2, d_2) are two data sets with each variables normalized between 0.1 to 0.9, then the two data sets are considered to be similar if the deviation is less than or equal to 10%, *i.e*.

$$D_T \le 0.1 \times D_{T_{\max}} \tag{7.3}$$

where D_T is the distance between the two datasets and it is given by

$$D_T = \sqrt{(v_1 - v_2)^2 + (f_1 - f_2)^2 + (d_1 - d_2)^2}\,. \tag{7.4}$$

$D_{T_{\max}}$ is the maximum possible value of D_T. Since all values are normailsed between 0.1 and 0.9, $D_{T_{\max}}$ is given by

$$D_{T_{\max}} = \sqrt{(0.8^2 + 0.8^2 + 0.8^2)} = 1.385 \tag{7.5}$$

If any of the simulated data set is similar to a dataset in the NN module, *i.e.*, $D_T \le 0.1385$ then it will replace the initial random data set, otherwise it gets appended to the data base of AI module for the use when it needed. This helps in improving the accuracy of network prediction. Figure 7.2 shows the flow chart of the present methodology.

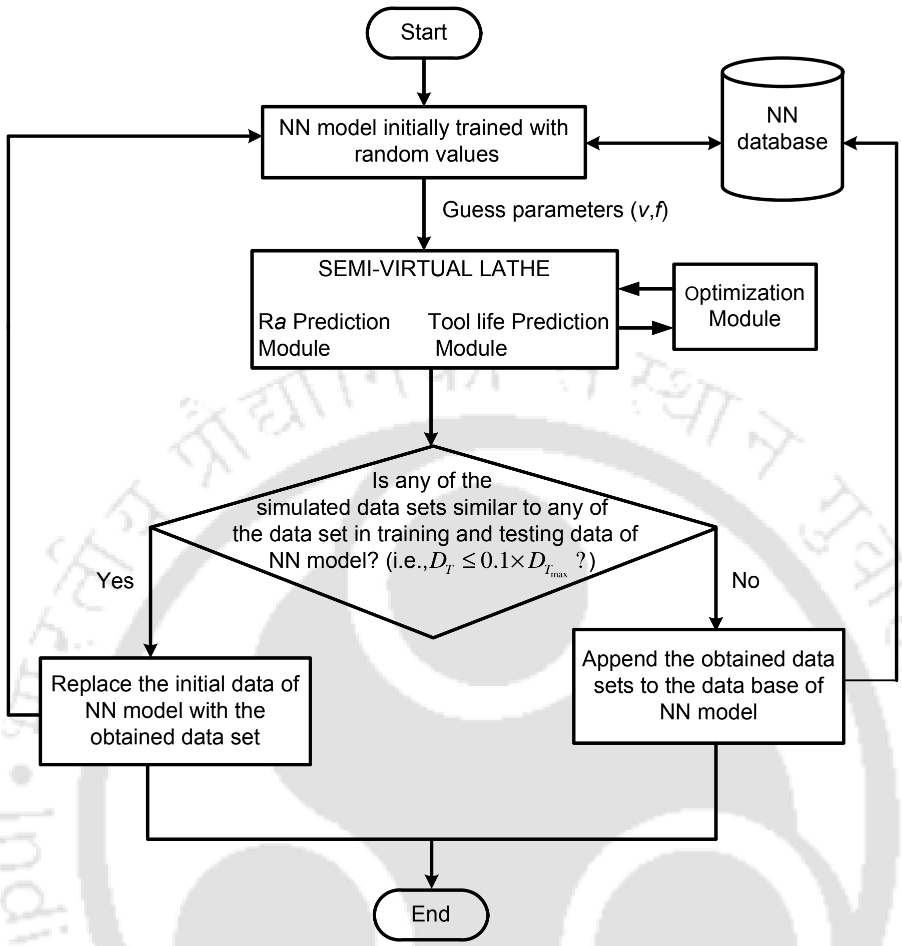


Figure 7.2. Flow chart of the online learning methodology

7.4 Optimization of Finish Turning Process through Online Learning

The proposed soft computing based online learning methodology is demonstrated for optimization of finish turning process. The cutting parameters *viz.*, cutting speed and feed are optimized for obtaining minimum cost subjected to maximum allowable surface finish.

7.4.1 Illustrative Example 1

Consider an optimization of finish turning process. The process is optimized for minimizing the cost satisfying the maximum allowable surface roughness value. A number of jobs are machined for different values of maximum allowable surface finish with varying depth of cut.

7.4.1.1 Job Number 1

Let optimum parameters are to be obtained for a finish turning process with maximum allowable surface finish (R_a^*) as 3.5 μm at constant depth of cut of 0.4 mm. Considering the inherent randomness in metal cutting process, a maximum of 20% error may be taken in prediction of surface roughness (Risbood *et al.*, 2003). Therefore the target surface roughness value is taken as 80% of maximum allowable surface roughness, $R_a^t = 0.8 \times 3.5 = 2.8$ μm. With the statistical variation in surface roughness, the process is said to under control if actual surface roughness value ($\hat{R}_a$) falls within $\pm 20\%$ deviation from that of targeted surface roughness value (*i.e.*, in between 2.24 and 3.36 μm). The work diameter and length of machining are 30 mm and 100 mm respectively.

Initially an appropriate value of cutting parameters (*v*, *f*) satisfying surface roughness is to be selected. There are a number of different combinations of cutting parameters are available. To identify the optimum cutting parameters that minimize the cost satisfying the surface roughness limits, a simplex search optimization is employed. For two variables problem three set of initial cutting conditions are needed for optimization. At first, roughly the cutting parameter (*v*, *f*): (190, 0.15) is selected as first cutting condition and other two cutting conditions as (171, 0.20) and (156, 0.20) are selected with 10% deviation so that it is not far away from the first cutting condition. The NN model of AI system predicts the surface roughness ($\hat{R}_a$) and tool life (*T*) and cost function values (*F*) are evaluated. The values obtained are (*v*, *f*, $\hat{R}_a$, *T*, *F*): (190, 0.15, 2.98, 118, 3.7×10^{-5}), (171, 0.17, 3.16, 126, 3.62×10^{-5}), and (156, 0.17, 3.22, 131, 3.96×10^{-5}). It is optimized and AI module obtain the guess parameters, satisfying the surface roughness and it is (197, 0.18). The first component is machined at this cutting condition. The surface roughness is measured and machining is continued, tool life is obtained and cost function is evaluated. Machining is continued with other cutting conditions provided by optimization module in successive iteration process till it reaches optimum cutting condition. Table 7.2 shows different cutting conditions with actual values of surface roughness, tool life and the corresponding function value.

Table 7.2. Different simulation runs (Illustrative example: Job number 1)

Piece (simulation) number	v (m/min)	f (mm)	$\hat{R}_a$ (μm)	T (min)	F (10^{-5} min/mm^2)
1	160	0.15	2.37	119	4.52
2	152	0.20	2.41	168	3.70
3	178	0.20	2.32	57	3.31
4	182	0.22	2.29	49	3.15
5	199	0.28	3.32	25	2.51

The optimum cutting condition satisfying the surface roughness limits is reached in five iterations. Thus, we obtain five simulated data sets. Each of the simulated data sets was compared with datasets of AI module. The data sets are normalised and the distances between each pair were evaluated. Normalising of data between 0.1 and 0.9 is done through linear mapping using the following relation:

$$x_n = x_{n\min} + \frac{x_{n\max} - x_{n\min}}{x_{\max} - x_{\min}}(x - x_{\min}), \tag{7.6}$$

where x_n is the normalised value, x_{max} and x_{min} are the maximum and minimum values of a parameter and x_{nmax} and x_{nmin} are the normalised maximum and minimum values. For example the data set (v, d, f): (152, 0.4, 0.2) is normalised as follows.

$$x_{(152)} = 0.1 + \frac{0.9-0.1}{240-140}(152-140) = 0.196 \tag{7.7}$$

$$x_{(0.4)} = 0.1 + \frac{0.9-0.1}{1.0-0.1}(0.4-0.1) = 0.366 \tag{7.8}$$

$$x_{(0.2)} = 0.1 + \frac{0.9-0.1}{0.32-0.04}(0.2-0.04) = 0.557 \tag{7.9}$$

While comparing the distance between (152, 0.4, 0.2) and (140, 0.4, 0.18) it is found that the distance is less than 10% of maximum distance between the datasets ($D_{T_{\max}}$) (*i.e.*, 0.1385).

$$D_{T_{\max}} = \sqrt{(0.196-0.1)^2 + (0.336-0.366)^2 + (0.557-0.5)^2} = 0.112 \le 0.1385 \tag{7.10}$$

In five simulated datasets the distance of two data sets *viz.* (152, 0.4, 0.20) and (199, 0.4, 0.28) were less than 10% of $D_{T_{\max}}$ (*i.e.*, 0.1385) and therefore it is inducted into

AI module and remaining data are stored in the database. The further training of neural network enhances its prediction capability due to induction of real machining datasets in NN data sets of AI module.

7.4.1.2 Job Number 2

Let the second job is to obtain optimum parameters for achieving maximum allowable surface finish (R_a^*) is 3.0 μm at constant depth of cut of 0.4 mm. The target surface roughness value is 2.4 μm; the actual surface roughness to control the process is in between 1.92 and 2.40 μm. The initial guess parameters predicted by AI module is (228, 0.14) and it is implemented in the machining system. The actual surface roughness obtained is 2.33 μm and tool life is 42 min. The problem is optimized in 4 iterations and optimum parameter is (179, 0.20). Table 7.3 shows iterative results. Two simulated data were found similar and are inducted into NN datasets of AI module.

Table 7.3. Different simulation runs (Illustrative example 1: Job number 2)

Piece (simulation) number	v (m/min)	f (mm)	$\hat{R}_a$ (μm)	T (min)	F (10^{-5} min/mm^2)
1	196	0.16	2.28	89	3.61
2	180	0.18	2.36	100	3.50
3	189	0.19	2.32	43	3.20
4	179	0.20	2.39	55	3.12

The optimum cutting condition that satisfies the surface roughness limits is reached after four iterations. On comparing, all four simulated data sets were found similar and are inducted to NN data sets of AI module. The NN of AI module is now trained again and is used for future prediction.

7.4.1.3 Job Number 3

The second job is machined at 0.1 mm depth of cut. The optimum parameters are to be obtained for achieving maximum allowable surface finish (R_a^*) as 3.00 μm. The target surface roughness value (R_a^t) is 2.4 μm. The actual surface roughness ($\hat{R}_a$) limit is

1.92–2.88 μm. The AI module predicts guess cutting parameters (*v* and *f*) as 181 m/min and 0.18 mm. The machining system uses this parameter for producing the component. The actual surface roughness and tool life obtained are 2.43 μm and 159 min respectively. Production is continued further with iterative cutting conditions dictated by optimization module, till optimum is reached. Table 7.4 shows different iterative results and corresponding values of surface roughness, tool life and the function value.

Table 7.4. Different simulation runs (Illustrative example 1: Job number 3)

Piece (simulation) number	v (m/min)	f (mm)	$\hat{R}_a$ (μm)	T (min)	F (10^{-5} min/mm^2)
1	198	0.18	2.38	101	2.99
2	189	0.19	2.41	123	2.93
3	178	0.21	2.46	154	2.79
4	164	0.23	2.85	216	2.73

Four iterative runs are required for obtaining optimum cutting conditions. On comparison, two simulated data sets *viz.* (189, 0.1, 0.19) and (178, 0.1, 0.21) are found similar and are inducted to NN data sets of AI module and the other data stored in the data base. The further training of NN model enhances its prediction capability.

Similarly, number of problems having depth of cut varying from 0.1 mm to 1.0 mm for different values of maximum allowable surface roughness are solved. Table 7.5 shows various jobs machined during the process. The bar chart shown in Figure 7.3 presents the number of virtual simulation runs required for different jobs. From the figure it is evident that number of virtual simulation runs keeps reducing as more and more jobs are produced. In the present case, once shop floor has handled 7 different jobs, only one simulation is required and thus AI module said to provide correct prediction. As the AI module gets trained well it provides exact cutting parameters in the initial guess and thus machining can be performed at optimum cutting condition even at first job and no loss is incurred.

Table 7.5. Various jobs machined in online learning process (Illustrative example 1)

Piece (simulation) Number	Data set (d	R_a^*)	Initial guess parameters (v	f)	Number of simulation runs	Optimum cutting parameters (v_{opt}	f_{opt})	Actual surface roughness, Function value ($\hat{R}_a$	F_{min})
1	0.4	3.50	197	0.12	5	199	0.28	3.32	2.51
2	0.4	2.75	191	0.16	4	164	0.23	2.85	2.73
3	0.1	3.00	181	0.18	4	184	0.22	2.74	2.84
4	0.7	3.25	154	0.21	3	169	0.26	2.69	2.85
5	0.8	3.75	168	0.20	3	152	0.28	3.47	2.62
6	0.2	4.25	195	0.24	2	172	0.29	3.38	2.28
7	0.5	3.50	191	0.19	2	203	0.31	2.97	2.46
8	1.0	3.25	219	0.22	1	220	0.22	3.10	4.32
9	0.1	3.25	174	0.24	1	174	0.24	2.63	2.54
10	0.9	3.50	164	0.30	1	164	0.30	3.02	2.63
11	0.6	3.75	180	0.26	1	180	0.26	3.00	2.80

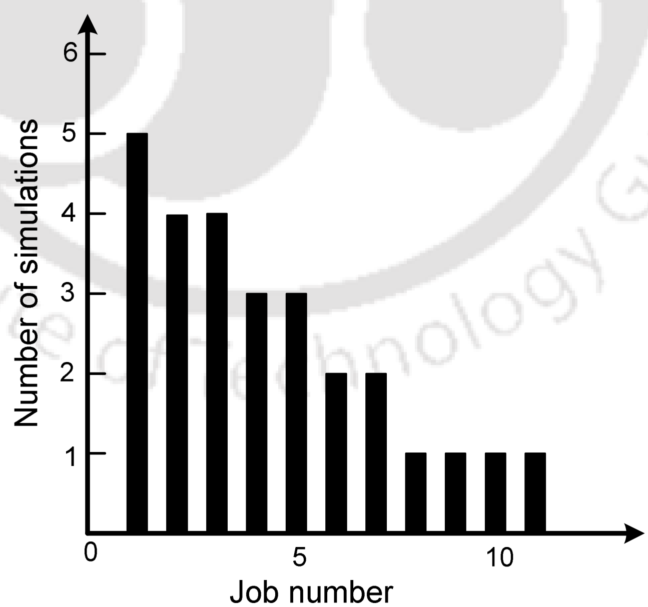


Figure 7.3. Bar chart showing the reduction in simulation runs with different jobs produced (Illustrative example 1)

7.4.2 Illustrative Example 2

The online learning procedure is tested for second set of components. Another set of jobs with varying depth of cut for different values of maximum allowable surface roughness are solved.

7.4.2.1 Job number 1

Let the first job is to obtain optimum parameters for achieving maximum allowable surface finish (R_a^*) as 3.25 µm at constant depth of cut of 0.4 mm. The target surface roughness value (R_a^t) is 2.6 µm. The actual surface roughness ($\hat{R}_a$) limit is 2.08–3.12 µm. The AI module predicts guess cutting parameters (*v* and *f*) as 178 m/min and 0.14 mm. The machining system uses this parameter for producing the component. The actual surface roughness and tool life obtained are 2.39 µm and 74 min respectively. Production is continued further with iterative cutting conditions dictated by optimization module, till optimum is reached. Table 7.6 shows different iterative results and corresponding values of surface roughness, tool life and the function value.

Table 7.6. Different simulation runs (Illustrative example 2: Job number 1)

Piece (simulation) number	v (m/min)	f (mm)	$\hat{R}_a$ (µm)	T (min)	F (10^{-5} min/mm^2)
1	157	0.17	2.78	119	3.95
2	135	0.19	2.86	234	3.99
3	144	0.20	2.72	163	3.61
4	141	0.24	2.63	158	3.08
5	166	0.24	2.87	70	2.92

The optimum cutting condition is reached after four iterations. On comparing three simulated data sets *viz*. (157, 0.3, 0.17); (135, 0.3, 0.19) and (144, 0.3, 0.20) are inducted to NN data sets of AI module and the other data stored in the data base.

7.4.2.2 Job number 2

The second job is machined for achieving maximum allowable surface finish (R_a^*) of 3.5 μm and depth of cut is 0.5 mm. The process is said to under control if actual surface roughness ($\hat{R}_a$) is in between 2.24 and 3.36 μm. AI module predicts initial guess parameters as (160, 0.13). It is implemented in the machining system. The actual surface roughness and tool life obtained is 2.59 μm and 112 min respectively. The optimum parameter reached in 5 iterations and is (171, 0.24). Table 7.7 shows the iterative result. Two simulated data sets (191, 0.5, 0.16) and (171, 0.5, 0.24) were found similar and are inducted into NN datasets of AI module.

Table 7.7. Different simulation runs (Illustrative example 2: Job number 2)

Piece (simulation) number	v (m/min)	f (mm)	$\hat{R}_a$ (μm)	T (min)	F (10^{-5} min/mm^2)
1	146	0.12	2.87	189	4.89
2	147	0.16	2.40	147	4.44
3	191	0.16	2.23	40	3.81
4	168	0.19	2.33	66	3.44
5	171	0.24	2.41	51	2.75

Table 7.8 shows various jobs machined during online learning process. A number of problems having depth of cut varying from 0.1 mm to 1.0 mm for different values of maximum allowable surface roughness are solved.

Figure 7.4 shows the bar chart of virtual simulation runs required for the production of different jobs. The number of simulation runs keeps on reducing as varieties of jobs are produced. In this example it is found that once shop floor handled 7 different jobs, the AI module predicts exact cutting parameters in the initial guess as it require one simulation runs. Since 8th job onwards the AI module predicts optimum parameters and machining is performed at optimum cutting condition.

Table 7.8. Various jobs machined in online learning process (Illustrative example 2)

Job number.	Data set (d	R_a^*)	Initial guess parameters (v	f)	Number of simulation runs	Optimum cutting parameters (v_{opt}	f_{opt})	Actual surface roughness, Function value ($\hat{R}_a$	$F_{\min}$)
1	0.4	3.00	178	0.14	5	166	0.24	2.87	2.92
2	0.5	3.00	160	0.13	5	171	0.24	2.84	2.75
3	0.2	2.50	162	0.16	4	181	0.23	2.38	2.60
4	0.6	3.50	187	0.18	3	208	0.26	2.65	2.62
5	0.8	3.25	154	0.15	3	147	0.20	3.02	3.65
6	0.4	2.50	168	0.18	3	176	0.20	2.34	3.15
7	0.3	3.25	187	0.20	2	164	0.24	2.85	2.72
8	0.7	3.00	145	0.26	1	145	0.26	2.57	2.86
9	1.0	3.75	210	0.21	1	210	0.21	3.12	2.40
10	0.5	3.00	171	0.24	1	171	0.24	2.84	2.75

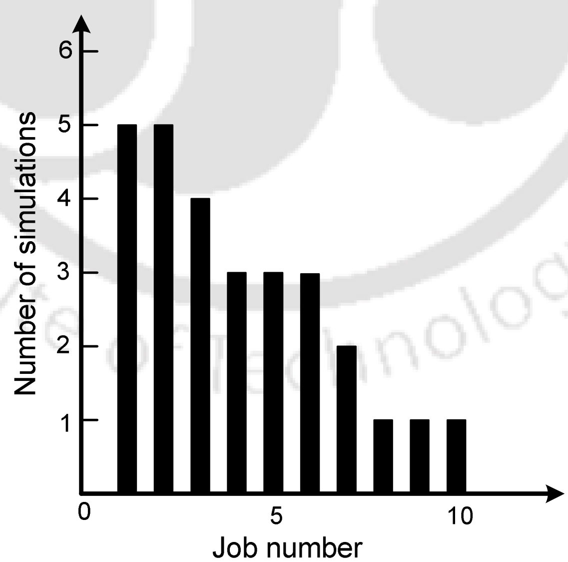


Figure 7.4. Bar chart showing the reduction in simulation runs with different jobs produced (Illustrative example 2)

7.4.3 Illustrative example 3

A number of different jobs having depth of cut from 0.1 to 1.0 for achieving the different values maximum allowable surface roughness are solved. The optimum cutting parameters are optimized for minimizing the cost.

7.4.3.1 Job number 1

Consider the first job for obtaining optimum parameters for finish turning process. The maximum allowable surface finish (R_a^*) is 3.50 µm and depth of cut is 0.1 mm. The target surface roughness value (R_a^t) is 2.8 µm. The actual surface roughness ($\hat{R}_a$) limit is 2.40–3.60 µm. The AI module predicts guess cutting parameters (*v* and *f*) as 190 m/min and 0.16 mm. The machining system uses this parameter for producing the component. The actual surface roughness and tool life obtained are 2.13 µm and 136 min respectively. Production is continued further with iterative cutting conditions dictated by optimization module, till optimum is reached. Table 7.9 shows different iterative results and corresponding values of surface roughness, tool life and the function value.

Table 7.9. Different simulation runs (Illustrative example 3: Job number 1)

Piece (simulation) number	v (m/min)	f (mm)	$\hat{R}_a$ (µm)	T (min)	F (10^{-5} min/mm^2)
1	175	0.21	2.40	167	3.12
2	206	0.19	2.33	80	2.76
3	224	0.20	2.15	51	2.5
4	216	0.22	2.39	56	2.35
5	225	0.25	3.41	42	2.05

The optimum cutting condition is reached after five iterations. On comparing two simulated data sets *viz.* (175, 0.1, 0.21) and (206, 0.1, 0.19) found similar and are inducted to NN data sets of AI module and the other data stored in the data base.

7.4.3.2 Job number 2

Let the second job is to obtain optimum parameters for achieving maximum allowable surface finish (R_a^*) is 3.0 µm at constant depth of cut of 0.4 mm. The target surface roughness value is 2.4 µm; the actual surface roughness to control the process is in between 1.92 and 2.40 µm. The initial guess parameters predicted by AI module is (232, 0.15) and it is implemented in the machining system. The actual surface roughness obtained is 2.17 µm and tool life is 19 min. The problem is optimized in 4 iterations and optimum parameter is (177, 0.20). Table 7.10 shows iterative results. Two simulated data were found similar and are inducted into NN datasets of AI module.

Table 7.10. Different simulation runs (Illustrative example 3: Job number 2)

Piece (simulation) number	v (m/min)	f (mm)	$\hat{R}_a$ (µm)	T (min)	F (10^{-5} min/mm^2)
1	165	0.18	2.32	89	3.61
2	160	0.19	2.39	100	3.50
3	170	0.18	2.37	77	3.44
4	177	0.20	2.40	58	3.14

A number of different jobs having depth of cut varying from 0.1 mm to 1.0 mm for different values of maximum allowable surface roughness are solved. Table 7.11 shows the number of iterations required for various jobs to arrive at optimum.

Figure 7.5 shows the bar chart of the number of virtual simulation runs required for different jobs. As varieties of jobs are handled the number of virtual simulation runs keeps reducing. In the present case, once shop floor has handled 9 different jobs, only one simulation is required. Once AI module gets completely trained, it predicts optimum cutting parameters as initial guess and thus machining can be performed at optimum without any loss incurred.

Table 7.11. Various jobs machined in online learning process (Illustrative example 3)

Job number	Data set (d	R_a^*)	Initial guess parameters (v	f)	Number of simulation runs	Optimum cutting parameters (v_{opt}	f_{opt})	Actual surface roughness, Function value ($\hat{R}_a$	F_{min})
1	0.1	3.50	190	0.16	5	225	0.25	3.34	2.05
2	0.4	3.00	232	0.15	4	177	0.20	2.63	3.14
3	0.6	3.50	210	0.14	4	166	0.26	2.91	2.63
4	0.9	3.25	210	0.14	3	155	0.23	2.62	3.14
5	0.5	3.00	180	0.17	3	171	0.24	2.84	2.75
6	0.2	2.75	170	0.21	3	179	0.30	2.20	2.04
7	0.3	3.50	186	0.19	2	141	0.26	3.16	2.82
8	1.0	3.0	161	0.18	2	130	0.26	2.85	3.13
9	0.8	2.75	152	0.19	2	132	0.26	2.53	3.10
10	0.6	3.00	167	0.23	1	167	0.23	2.62	2.93
11	0.5	3.75	143	0.26	1	143	0.26	3.20	2.84
12	0.4	2.75	156	0.24	1	156	0.24	2.41	2.85
13	0.3	2.50	234	0.23	1	234	0.23	2.52	2.61

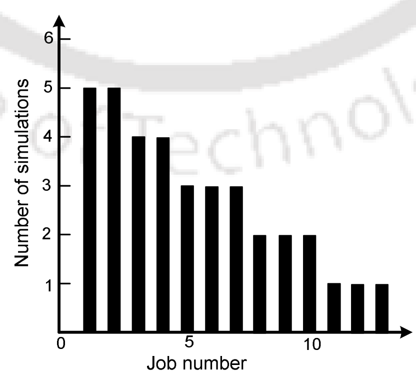


Figure 7.5. Bar chart showing the reduction in simulation runs with different jobs produced (Illustrative example 3)

7.5 Summary

A soft computing based online learning procedure is proposed in this work. The proposed scheme of online learning optimization methodology helps to optimize the metal cutting process and involves less cost compared to other offline optimization methods. The method is highly suitable to handle variety of different jobs in a batch production of jobs, which is the main requirement of today's market scenario. The data sets obtained through online machining is used to carry out optimization with online learning. An artificial intelligence (AI) based online learning methodology to optimize cutting parameters for finish turning process is presented. The virtual simulation module combined with the AI module learns with experience as more and more jobs are machined. The well trained AI module provides exact cutting parameters in the initial guess and thus machining can be performed at optimum cutting condition even at first job and no loss is incurred. The direct implementation at shop floor in different manufacturing environment is main advantage of this scheme. The neural network model of AI system learns from limited dataset and predicts the parameter with reasonable accuracy. The effective error of NN training is obtained within 20 %. Source code for training the network is written in MATLAB® version 7.1. It runs on Pentium–4 with 512 MB RAM is used. The computational time required is less than a second.

Chapter 8

Conclusions and Scope of Future Work

8.1 Conclusions

In this thesis, the feasibility of using cloud computing for the optimization of machining processes is explored. A web system called cloud computing based optimization and data base (CCBOD) that works on private cloud is developed. The adoption of cloud computing feature helps to optimize different machining processes. A generalized online optimization algorithm for optimization of machining process has been developed. The optimization strategy does not require any *a priori* machining models. It optimizes the process during production of components. Source code for the algorithm has been written in MATLAB®. For easy implementation of the proposed online optimization procedure a 'user interactive based menu driven' system has been coded in MATLAB® with GUI. The .EXE file named 'online_optimization.exe' is placed in CCBOD system, which can be downloaded at client system and the machine tool operator can implement the same for optimization of machining process at their location. The machining feedback is sent to the server through clouds. The real machining data provided by different clients are stored in database server. The server carries out optimization and optimized data can be provided to various clients. The clients who connect to CCBOD *via* web browsers can avail the facility. The system can be accessed through common web browsers such as Windows IE, Mozilla Firefox, Google Chrome, Safari, and Opera at client system.

For web implementation of the cloud based optimization system HTML, CSS, and PHP scripts have been written for clients/users to interact with the system. The web enabled feature help for accessing and sharing information among the users. The retrieval of optimum cutting parameters is obtained using PHP scripts and MySQL queries. The result can be viewed at client's screen. For storing the machining information MySQL database system is used at server side. It stores all information in "machining database" at main server and is managed by the administrator. The user can get the information by fetching data from the database. The user also sends the feedback of the same through HTML and PHP pages.

A fuzzy set based optimization methodology has been developed. It consists of finding the optima with the help of IF-THEN rules. The methodology is applied for a number of test functions and machining optimization problem to optimize the process parameters both by offline and online procedure. The proposed fuzzy set based optimization is experimentally demonstrated for online optimization machining process. It is applied for optimization of finish turning process of cold rolled steel bars using TiN coated carbide tool and is carried out in a lathe. The optimization objective is to minimize the cost satisfying desired surface roughness.

A soft computing based online learning procedure for optimizing the process has been developed. The online learning procedure is demonstrated through numerical experiments that optimizes the process predicting better cutting condition. The algorithm shows that it keeps becoming more and more intelligent by machining variety of different jobs. The data sets obtained through simulation results are stored and effectively used for training a neural network of artificial intelligence (AI) system.

The main contributions of the thesis are as follows:

1. The developed web enabled cloud computing based optimization and database (CCBOD) is employed for optimizing common machining processes. The different manufacturing clients/users can connect to CCBOD *via* web browsers for availing the facility. The functionality of the developed system is tested with virtual machine. Actual machining experiments are replaced by simulated results. The case examples show the feasibility of using the system for helping in agile manufacturing.
2. A new heuristic based online optimization methodology is proposed in the present work. The optimization model does not use any empirical relation and/or offline experimentation for evaluation of tool life and surface roughness in the optimization process. The methodology obtains reliable optimum cutting parameters by its direct implementation. In batch production of components machining under various tool-work piece combination/ or change of job, the proposed methodology is found suitable. The datasets obtained through online machining can also be used to develop

an artificial intelligence-based online learning methodology for optimizing cutting parameters.

3. An interactive based online optimization scheme working on cloud environment suggests cutting conditions for shop floor machine tool operator, accepts feedback, and optimizes the machining process during the production of components. The client can also retrieve optimum cutting parameters which can be viewed at clients' screen. The feedback of real life data (*i.e.*, actual surface roughness, tool life, cutting force) from different machine tool operator are supplied to the clouds. The developed private cloud uses a probability-based procedure for processing the information at the server side.
4. A fuzzy set based optimization methodology is developed. The procedure is based on linguistic sub-division of search domain and it obtains optimum zone in which the objective function value does not change drastically. This provides multiple numbers of solutions. The optimization procedure can be applied for optimizing both offline and online problems. It is experimentally demonstrated for obtaining optimum parameters for turning process. Student's *t*-test is performed to compare the significance between the cutting conditions in the optimum zone. The proposed optimization strategy is suitable for an agile manufacturing environment. The fuzzy set based online machining optimization is suitable when not much *a priori* information about the process is available in the shop floor.
5. Finally, a soft computing based online learning scheme has been developed and demonstrated for optimization of finish turning process. The method is highly suitable to handle variety of different jobs in a batch production system, which is the main requirement of today's market scenario. A well trained artificial intelligence (AI) module provides cutting parameters as the initial guess. Further, fine tuning can be carried out with a small amount of effort. The direct implementation at shop floor in different manufacturing environment is main advantage of this scheme. The methodology can also be implemented over a cloud.

8.2 Scope for future work

- In the present work, the cloud computing based optimization of conventional machining process has been developed. The work can be extended to other manufacturing process.
- The online measurement of process responses *viz*., surface roughness, cutting force and tool life is dependent on the use of sensors in the machining system. Research on hardware side for the development of proper sensors, efficient techniques for data acquisition, data filtration, and feature extraction should be developed.
- In the present work, an artificial intelligence based online learning methodology is proposed for finish turing process. It may be extended for multipass machining as well as to other machining process. The procedure may be integrated with cloud based machining optimization for complete automation of the system.
- A consortium of industries having machine shops can be formed for implementing the cloud models in this work. The feedback obtained from the consortium can be used to further refine the models.

References

- Abburi, N.R. and Dixit, U.S., (2006), A knowledge based system for the prediction of surface roughness in turning process, Robotics and Computer Integrated Manufacturing, **22**, pp. 363–372.

- Abburi, N.R. and Dixit, U.S., (2007), Multi-objective optimization of multi-pass turning processes, International Journal of Advanced Manufacturing Technology, **32**, pp. 902–910.

- Abouelatta, O.B. and Madl, J., (2001) Surface roughness prediction based on cutting parameters and tool vibrations in turning operations, Journal of Material Processing Technology, **118**, pp. 269–277

- Abu-Mahfouz, I., (2003), Drilling wear detection and classification using vibration signals and artificial neural network, International Journal of Machine Tools and Manufacture, **43**, pp.707–720.

- Aggarwal, A. and Singh, H., (2005), Optimization of machining techniques—A retrospective and literature review, Sadhana, **30**, pp. 699–711.

- Al-Marzoug, S.M. and Hodgson, R.J.M., (2006), Luus–Jaakola optimization procedure for multilayer optical coatings, Optics Communications, **265**, pp. 234–240.

- Ali, Y.M. and Zhang, L.C., (1999), Surface roughness prediction of ground components using a fuzzy logic approach, Journal of Materials Processing Technology, **89/90**, pp. 561–568.

- Ali, Y.M. and Zhang, L.C., (2004), A fuzzy model for predicting burns in surface grinding of steel, International Journal of Machine Tools and Manufacture, **44**, pp. 563–571.

- Apache web server: Complete guide (www.dedoimedo.com), 2011.

- Aygut, S., Golcu, M., Semiz, S. and Ergur, H.S., (2007), Modelling of cutting forces as function of cutting parameters for face milling of stellite 6 using an artificial neural network, Journal of Materials Processing Technology, **190**, pp.199–203.

- Azouzi, R. and Guillot, M., (1997), On-line prediction of surface finish and dimensional deviation in turning using neural network base sensor fusion, International Journal of Machine Tools and Manufacture, **37**, pp.1201–1217.

- Azouzi, R. and Guillot, M., (1998), On-line optimisation of turning process using an inverse process neuro controller, ASME Journal of Manufacturing Science Engineering, **120**, pp. 101–108.

- Basak, S., Dixit, U.S. and Davim, J.P., (2007), Application of radial basis function neural networks in optimization of hard turning of AISI D2 cold-worked steel with a ceramic tool, Proceedings of IMechE B Journal of Engineering Manufacture, **221**, pp. 987–998.

- Baskar, N., Saravanan, R., Asokan, P. and Prabhaharan, G., (2004), Ant colony algorithm approach for multi-objective optimization of surface grinding operations, International Journal of Advanced Manufacturing Technology, **23**, pp. 311–317.

- Baykasoglu, A. and Dereli, T., (2002), Novel algorithm approach to generate the 'number of passes' and ' depth of cuts' for the optimization routines of multi pass machining, International Journal of Production Research, **40**, pp.1549–1565.

- Biglari, F.R. and Fang, X.D., (1995), Real-time fuzzy logic control for maximizing the tool life of small –diameter drills, Fuzzy sets and systems, **72**, pp.91−101.

- Blaisdell, R., (2011), A brief history of cloud computing. (Online) Available from: http://www.rickscloud.com/a-brief-history-of-cloud-computing-2/

- Bo Liao, and Rein Luus., (2005), Comparison of the Luus–Jaakola optimization procedure and the genetic algorithm, Engineering Optimization, **37**, pp. 381–396.

- Bojadzieve, G. and Bojadzieve, M., (1995), Fuzzy sets, fuzzy logic, applications. Advances in Fuzzy Systems— Application and Theory, 5, World Scientific, Singapore.

- Brezocnik, M., Kovacic, M. and Ficko, M., (2004), Prediction of surface roughness with geometric programming, Journal of Materials Processing Technology, **157–158**, pp.28–36.

- Briceno, J.F., El-Mounayri, H. and Mukhopadhyay, S., (2002), Selecting an artificial neural network for efficient modeling and accurate simulation of the milling process, International Journal of Machine Tools and Manufacture, **42**, pp. 663−674.

- Brinksmeier, E., Tonshoff, H.K., Czenkusch, C. and Heinzel, C., (1998), Modelling and optimization of grinding processes, Journal of Intelligent Manufacturing, **9**, pp.303–314.

- Chen, J.C. and Black, J.T., (1997), A fuzzy-nets in process (FNIP) systems for tool-breakage monitoring in end-milling operations, International Journal of Machine Tools and Manufacture, **37**, pp. 783–800.

- Chen, J.C. and Savage, M., (2001), A Fuzzy-Net-Based Multilevel In-process Surface Roughness Recoginition System in Milling Operations, International Journal of Advanced Manufacturing Technology, **17**, pp. 670–676.

- Chen, K.H., Chen, S.J.G., Lin, L. and Changchien, S.W., (1998), An integrated graphical user interface (GUI) for concurrent engineering design of mechanical parts, Computer Integrated Manufacturing Systems, **11**, pp.91–112.

- Chen, M.C. and Chen, K.Y., (2003), Optimization of multi-pass turning operations with genetic algorithms: a note, International Journal of Production Research, **41**, pp.3385–3388.

- Chen, M.-C. and Tasi, D.-M., (1996), A simulated annealing approach for optimization of multi-pass turning operation, International Journal of Production Research, **34**, pp. 2803–2825.

- Ching-kao, C. and Lu, H.S., (2007), The optimal cutting –parameter selection of heavy cutting process in side milling for SUS304 stainless steel, The International Journal of Advanced Manufacturing Technology, **34**, pp. 440–447.

- Choi, Y.J., Park, M.S. and Chu, C.N., (2008), Prediction of drill failure using features extraction in time and frequency domains of feed motor current, International Journal of Machine Tools and Manufacture, **48**, pp.29–39.

- Chryssoluris, G. and Guillot, M., (1990), A comparison of statistical and AI approaches to the selection of process parameters in intelligent machining, ASME Journal of Engineering for Industry, **112**, pp.122–131.

- Chryssolouris, G., Lee, M., Ramsey, A., (1996), Confidence interval prediction for neural network models, IEEE Transactions on Neural Networks, **7**, pp. 229–232.

- Chung, B.M. and Tomizuka, M., (2001), Fuzzy logic modeling and control for drilling of composite laminates, 10th IEEE International Conference on Fuzzy Systems, Melbourne, pp.509–512.

- Chungchoo, C. and Saini, D., (2002), On-line tool wear estimation in CNC turning operations using fuzzy neural network model, International Journal of Machine Tools and Manufacture, **42**, pp. 29–40.

- Cus, F., Milfelner, M. and Balic, J., (2006), An intelligent system for monitoring and optimization of ball-end milling process, Journal of Materials Processing Technology, **175**, pp. 90–97.

- Das, S., Ray, R. and Chattopadhyay, A.B., (1996), Evaluation of wear of turning carbide inserts using neural networks, International Journal of Machine Tools and Manufacture **9**, pp.1639–1645.

- Deb, K., (1995), Optimization for engineering design: algorithms and examples. Prentice-Hall, New York.

- Deb, S. and Dixit, U.S., (2008), Intelligent machining: computational methods and optimization, In: Davim JP (ed) Machining: fundamentals and recent advances, Springer-Verlag, London.

- Deivanathan, R., Vijayaraghavan, L. and Krishnamurthy, R., (1999), In-process monitoring of grinding burn in the cylindrical grinding of steel, Journal of Materials Processing Technology, **91**, pp.37–42.

- Dixit, U.S., Robi, P.S. and Sarma, D.K., (2002), A systematic procedure for the design of a cold rolling mill, Journal of Materials Processing Technology, **121**, pp. 69–76.

- Dixit, P.M. and Dixit, U.S., (2008), Modeling of metal forming and machining processes: by finite element and soft computing methods, Springer-Verlag, London.

- Dixit, U.S., Sarma, D.K. and Davim, J.P., (2012), Environmentally friendly machining, Springer Briefs in Applied Sciences and Technology, DOI 10.1007/978-1-4614-2308-9_6.

- Dorigo, M., (1996), The ant system: optimization by a colony of cooperating agents. IEEE Transactions of Systems, Man and Cybernectics– Part B:Cybernetics, **26**, pp.29–41.

- Dutta, R.K., Paul, S. and Chattopadyay, A.B., (2000), Applicability of the modified back-propagation algorithm in tool condition monitoring for faster convergence, Journal of Materials Processing Technology, **98**, pp. 299̃309.

- Dutta, R.K., Paul, S., Chattopadyay, A.B., (2000), Fuzzy controlled back propagation algorithm for tool condition monitoring in face milling, International Journal of Production Research, **38**, pp. 2989–3010.

- Dutta, R.K., Paul, S. and Chattopadyay, A.B., (2006), The efficacy of back propagation neural network with delta bar delta in predicting the wear of carbide inserts in face milling, The International Journal of Advanced Manufacturing Technology, **31**, pp.434–442.

- Ekel, P., Pedrycz, W. and Schinzinger, R., (1998), A general approach to solving a wide class of fuzzy optimization problems, Fuzzy Sets and Systems, **97**, pp. 49–66.

- Ezugwu, E.O., Arthur, S.J. and Hines, E.L., (1995), Tool-wear prediction using artificial neural networks, Journal of Materials Processing Technology, **49**, pp. 255–264.

- Ezugwu, E.O., Fadare, D.A., Bonney, J., Da Silva R.B. and Sales, W.F., (2005), Modelling the correlation between cutting and process parameters in high-speed machining of Inconel 718 alloy using an artificial neural network, International Journal of Machine Tools and Manufacture, **45**, pp. 1375−1385.

- Fang, X. and Jawahir, I.S., (1994), Predicting total machining performance in finish turning using integrated fuzzy-set models of the machinability parameters, International Journal of Production Research, **32**, pp. 833−849.

- Feng, C-X. and Wang, X-F., (2003), Surface roughness prediction modelling: neural networks versus regression, IIE Transactions, **35**, pp.11−27.

- Finnie, I., (1956), Review of the metal-cutting analysis of the past hundred years. Mech Eng., **78**, pp. 715–721.

- Fuh, K.H. and Wang, S.B., (1997), Force modeling and forecasting in creep feed grinding using improved operation BP neural network, International Journal of Machine Tools and Manufacture, **37**, pp.1167–1178.

- Garg, S., Pal, S.K. and Chakraborty, D., (2007), Evaluation of the performance of back propagation and radial basis function neural networks in predicting the drill flank wear, Neural Computing and Application, **16**, pp.407–417.

- Geerdes, W.M. and Alvardo, M.A.T., (2008), An application of physics-based and artificial neural network-based hybrid temperature prediction scheme in a hot strip mill, Journal of Manufacturing Science and Engineering, **130**, pp.014501 (5 pages)

- Ghaiebi, H. and Solimanpur, M., (2007), An ant algorithm for optimization of hole-making operations, Computers and Industrial Engineering, **52**, pp. 308–319.

- Ghosh, N., Ravi, Y.B., Mukhopadyay, S., Paul, S., Mohanty, A.R. and Chattopadyay, A.B., (2007), Estimation of tool wear during CNC milling using neural network- based sensor fusion, Mechanical systems and signal processing, **21**, pp.466–479.

- Gilbert, W.W., (1950), Economics of machining. Machining Theory and Practice, American Society of Metals, pp.465–485.

- Goldberg, G.E., (1989), Genetic algorithms in search optimization and machine learning. Addison Wesley, Reading, MA.

- Gopal, A.V. and Rao, P.V., (2003), Selection of optimum conditions for maximum material removal rate with surface finish and damage as constraints in SiC grinding, International Journal of Machine Tools and Manufacture, **43**, pp.1327–1336.

- Govindasamy, J.J., McLoone, S.F., Irwin, G.W., French, J.J. and Doyle, R.P., (2005), Neural modeling, control and optimization of an industrial grinding process, Control Engineering Practice, **13**, pp.1243–1258.

- Gupta, R., Batra. J.L. and Lal, G.K., (1995), Determination of optimal subdivision of depth of cut in multipass turning with constraints, International Journal of Production Research, **33**, pp. 2555–2565.

- Hao, W., Zhu, X., Li, X. and Turagyenda, G., (2006), Prediction of cutting force for self-propelled cutting tool by artificial neural networks, Journal of Materials Processing Technology, **180**, pp.23–29.

- Hashmi, K., El Baradie, M.A. and Ryan, M., (1999), Fuzzy-logic based intelligent selection of machining parameters, Journal of Materials Processing Technology, **94**, pp. 94–111.

- Hasmi, K., Graham, I.D. and Mills, B., (2000), Fuzzy logic based data selection for the drilling process, Journal of Materials Processing Technology, **108**, pp.55–61.

- Hegginbotham, W.B. adn Pandey, P.C., (1967), A variable rate machining test for tool life evaluation, Proceedings of 8th International MTDR Conference, Manchester.

- Ho, W.H., Tasi, J.T., Lin. B.T. and Chou, J.H., (2009), Adaptive network-based fuzzy inference system for prediction of surface roughness in end milling process using hybrid Taguchi- genetic learning algorithm, Expert systems with applications, **36**, pp.3216–3222.

- Hong T, Tseng L, Wang S, (2002) Learning rules from incomplete training examples by rough sets. Expert Systems with Applications, **22**, pp. 285–293.

- Huang, G.Q. and Mak, K.L., (2001), Issues in the development and implementation of Web applications for product design and manufacture, International Journal of Computer Integrated Manufacturing, **14**, pp. 125–135.

- Iqbal, A., He, N., Li, L. and Dar, N.U., (2007), A fuzzy expert system for optimizing parameters and predicting performance measures in hard-milling process, Expert Systems with Applications, **32**, pp.1020–1027.

- Ishbuchi, H. and Tanaka, H., (1991), Regression analysis with interval model by neural networks, Proceedings of IEEE International Joint Conference on Neural networks, **2**, pp.1594–1599.

- Ishibuchi H., Miyazaki A, Tanaka H (1994) Neural Network based diagnosis systems for incomplete data missing inputs, Proceedings of ICNN'94, Orlando, pp. 3457–3460.

- Jamil, D. and Zaki, H., (2011), Cloud computing security, International Journal of Engineering Science Technology (IJEST), **3**, pp. 3478–3483.

- Jiao, Y., Pei, Z.S., Lei, S., Lee, E.S. and Fisher, G.R., (2005), Fuzzy adaptive networks in machining process modeling dimensional error prediction for turning operations, International Journal of Production Research, **43**, pp.2931–2948.

- Karpat, Y. and Ozel, T., (2005), Hard turning optimization using neural network modeling and swarm intelligence, Transactions of NAMRI/SME, **23**, pp.179–186.

- Karri, V., (1999), RBF neural network for thrust and torque predictions in drilling operations, 3rd International conference on computational intelligence and multimedia applications (ICCIMA'99), New Delhi, pp. 55–59.

- Kawak, J.S., Ha, M.K., (2004), Neural network approach for diagnosis of grinding operation by acoustic emission and power signals, Journal of Materials Processing Technology, **147**, pp.65–71.

- Kennedy, J. and Eberhart, R., (1995), Particle swarm optimization, In: Proceedings of the IEEE International Conference on Neural Networks (ICNN'95), Perth, Australia.

- Khajavi, A.N. and Komanduri, R., (1993), On multisensor approach in drill wear monitoring, Annals of the CIRP **42**, pp.71–74.

- Khanchustambham, R.G. and Zhang, G.M., (1992), A neural network approach to on-line monitoring of a turning process, Proceedings of IEEE International Joint Conference on Neural networks, **2**, pp.889–894.

- Kilic, S.E., (1985), Use of one-dimensional search method for the optimization of turning operations, Modeling, Simulation and Control B, **14**, pp. 39–63.

- Kim, G.H., (2002), Evaluation of pre-estimation model to the inprocess surface roughness for grinding operations, International Journal of the Korean Society of Precision Engineering, **3**, pp.24–30.

- Kim, S.S., Kim, I.H., Mani, V. and Kim, H.J., (2008), Real-coded genetic algorithm for machining condition optimization, International Journal of Advanced Manufacturing Technology, **38**, pp.884–895.

- Kohli, A. and Dixit, U.S., (2005), A neural-network-based methodology for the prediction of surface roughness in a turning process, International Journal of Advanced Manufacturing Technology, **25**, pp.118–129.

- Koza, J.R., (1992), Genetic programming, The MIT Press, Cambridge, MA.

- Lal, S.P. and Onwubolu, G.C., (2007), Three tiered web-based manufacturing system—Part 1:System development, Robotics and Computer-Integrated Manufacturing, **23**, pp. 138–151.

- Lambert, P.K. and Walvekar, A.G., (1978), Optimization of multipass machining operations, International Journal of Production Research, **16**, pp. 259–265.

- Lee, B.Y., Tarang, Y.S. and Ma, S.C., (1995), Modeling of the process damping force in chatter vibration, International Journal of Machine Tools and Manufacture, **35**, pp. 951–962.

- Lee, B.Y., Liu, H.S. and Tarang, Y.S., (1998), Modeling and optimization of drilling process, Journal of Materials Processing Technology, **74**, pp.149–157.

- Lee, Y.H., Yang, B.H. and Moon, K.S., (1999), An economic machining process model using fuzzy non-linear programming and neural network, International Journal of Production Research, **37**, pp.835–847.

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

- Lezanski, P., (2001), An intelligent system for grinding wheel condition monitoring, Journal of Materials Processing Technology, **109**, pp.258–263.

- Li, D., McKay, A., DePennington, A. and Barnes, C., (2001), A web-based tool and a heuristic method for cooperation of manufacturing supply chain decisions, Journal of Intelligent Manufacturing, **12**, pp. 433–453.

- Li, X. and Tso, S.K., (1999), Drill wear monitoring based on current signals, Wear, **231**, pp.172–178.

- Li, X. and Venuvinod, P.K., and Chen, M.K., (2000), Feed cutting force estimation from the current measurement with hybrid learning, International Journal of Advanced Manufacturing Technology, **16**, pp.859–862

- Liao, T.W. and Chen, L.J., (1994), A neural network approach for grinding processes: Modelling and Optimization, International Journal of Machine Tools and Manufacture, **34**, pp. 919–937.

- Lin, J.T., Bhattacharya, D. and Keeman, V., (2003), Multiple regression and neural networks analyses in composite machining, Composites science and technology, **63**, pp.539–548.

- Lin, S.C. and Ting, C.J., (1996), Drill wear monitoring using neural networks, International Journal of Machine tools and Manufacture, **36**, pp. 465–475.

- Lin, W.S., Lee, B.Y. and Wu, C,L., (2001), Modeling the surface roughness and cutting force for turning, Journal of Materials Processing Technology, **108**, pp. 286–293.

- Liu, B., (2002), Toward fuzzy optimization without mathematical ambiguity. Fuzzy Optimization and Decision Making, **1**, pp. 43–63.

- Liu, H.S., Lee, B.Y. and Tarang, Y.S., (2000), In-process prediction of corner wear in drilling operations, International Journal of Advanced Manufacturing Technology, **101**, pp.152–158.

- Liu, Q., Chen, X. and Gindy, N., (2005), Fuzzy pattern recognition of AE signals for grinding burn, International Journal of Machine Tools and Manufacture, **45**, pp. 811–818.

- Liu, T.I., and Chen, W,Y., (1998), Intelligent detection of drill wear. Mechanical systems and signal processing, **12**, pp. 863–873.

- Lo, S.P., (2003), An adaptive –network based fuzzy inference system for prediction of work piece surface roughness in end milling. Journal of Materials Processing Technology, **142**, pp. 665–675.

- Lou, S.J. and Chen, J.C., (1999), In-process surface roughness recognition (ISRR) system in end-milling operation, International Journal of Advanced Manufacturing Technology, **15**, pp. 200–209.

- Luhandjula, M.L., (1989), Fuzzy optimization: an appraisal. Fuzzy Sets and Systems, **30**, pp. 257–282.

- Luong, L.H.S. and Spedding, T.A., (1995), A neural network system for predicting machining behaviour, Journal of Materials Processing Technology, **52**, pp. 585–591.

- Luus, R. and Jaakola, T.H.I., (1973), Optimization by direct search and systematic reduction of the size of search region, AIChE Journal, **19**, pp.760–766.

- Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J. and Ghalsasi, A., (2011), Cloud computing — The business perspective, Decision Support Systems **51**, pp. 176–189.

- Mathews, P.G. and Shunmugam, M.S., (1999) Neural-network approach for predicting hole quality in reaming. International Journal of Machine Tools and Manufacture, 39, pp. 723–730.

- Mell, P. and Grance, T., (2009), Perspectives on cloud computing and standards, National Institute of Standards and Technology (NIST), Information Technology Laboratory.

- Merchant, M.E., (1998), Interpretative look on 20th century research on modeling of machining, Machining Science and Technology, **2**, pp.157–163.

- Merdol, S.D. and Altintas, T., (2008), Virtual simulation and Optimisation of milling applications—Part II: Optimisation and feed rate scheduling, ASME Journal of Manufacturing Science Engineering, **130**, pp. 0510051–05100510.

- Mukherjee, I. and Ray, P.K., (2006), A review of optimization techniques in metal cutting processes, Computers and Industrial Engineering, **50**, pp. 15–34.

- Nadgir, A. and Ozel, T., (2000), Neural network modeling of flank wear for tool condition monitoring in orthogonal cutting of hardened steel, 4th International Conference on Engineering Design and Automation. Florida, USA, pp. 1–6.

- Nandi, A.K. and Pratihar, D.K., (2004), An expert system based on FBFN using a GA to predict surface finish in ultra–precision turning, Journal of Materials Processing Technology, **155–156**, pp.1150–1156.

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

- Nandi, A.K. and Davim, J.P., (2009), A study of drilling performances with minimum quantity of lubricant using fuzzy logic rules. Mechatronics, 19, pp. 218–232.

- Nandhi, A.K. and Pratihar, D.K., (2004), Design of a genetic-fuzzy system to predict surface finish and power requirement in grinding. Fuzzy sets and systems, **148**, pp. 487–504.

- Natarajan, U., Saravanan, R. and Periasamy, V.M., (2006), Application of particle swarm optimization in artificial neural network for prediction of tool life, International Journal of Advanced Manufacturing Technology, **31**, pp. 1084–1088.

- Ojha, D.K. and Dixit, U.S., (2005), An economic and reliable tool life estimation procedure for turning, International Journal of Advanced Manufacturing Technology, **26**, pp. 726–732.

- Ojha, D.K., Dixit, U.S. and Davim, J.P., (2009), A soft computing based optimization of multi-pass turning processes, International Journal of Materials and Product Technology, **35**, pp.145–166.

- Oktem, H., Erzurumlu, T. and Kutaran, H., (2005), Applications of response surface methodology in the optimization of cutting conditions for surface roughness. Journal of Materials Processing Technology, **170**, pp. 11–16.

- Onwubolu, G.C. and Kumalo, T., (2001), Optimization of multi pass turning operations with genetic algorithm. International Journal of Production Research, 39(16):3727–3745.

- Ozel, T. and Karpet, Y., (2005), Predictive modeling of surface roughness and tool wear in hard turning using regression and neural networks, International Journal of Machine Tools Manufacture, **45**, pp. 467–479.

- Pal, S.K., Chakroborty, D, (2005), Surface roughness prediction in turning using artificial neural network, Neural Computing and Applications, **14**, pp.319–324.

- Panda, S.S., Singh, A.K., Chakraborty, D. and Pal, S.K., (2006), Drill wear monitoring using back propagation neural network, Journal of Materials Processing Technology, **172**, pp. 283–290.

- Patra, K., Pal, S.K. and Bhattacharyya, K., (2007), Artificial neural network based on prediction of drill flank wear motor current signals, International Journal of Soft Computing, **7**, pp. 927–935.

- Prakasvudhisarn, C., Kunnapapdeelert, S. and Yenradee, P., (2009), Optimal cutting condition determination for desired surface roughness in end milling, International Journal of Advanced Manufacturing Technology, **41**, pp. 440–451.

- Quiza, R., Figueira, L. and Davim, J.P., (2008), Comparing statistical models and artificial neural networks on predicting the tool wear in hard machining D2 AISI steel. International Journal of Advanced Manufacturing Technology, **37**, pp. 641–648.

- Radhakrishnan, T. and Nandan, U., (2005), Milling force prediction using regression and neural networks, Journal of Intelligent and Manufacture, **16**, pp. 93–102.

- Rahman, S.M., Sarker, R. and Bignall, B., (1999), Application of multimedia technology in manufacturing: a review, Computers in Industry, **38**, pp. 43–52.

- Rangwala, S.S. and Dornfeld, D.A., (1989), Learning and Optimization of Machining Operations Using Computing Abilities of Neural Networks, IEEE Trans Systems, Man and Cybernatics Society, **19**, pp. 299–314.

- Rao, P.V., (2011), Advanced Modeling and Optimization of Manufacturing Processes, Springer–Verlag, London.

- Rao, S.S., (1987), Multi-objective optimization of fuzzy structural systems, International Journal of Numerical Methods in Engineering, **24**, pp. 1157–1171.

- Rao, S.S., (1999), Engineering optimization: Theory and Practice, 4th Ed. John Wiley and sons, Inc.

- Reddy, N.S.K, and Rao, P.V., (2005), Selection of optimum geometry and cutting conditions using surface roughness prediction model for end milling, International Journal of Advanced Manufacturing Technology, **26**, pp. 1202–1210.

- Reddy, N.S.K. and Rao, P.V., (2006), Selection of an optimal parametric combination for achieving a better surface finish in dry milling using genetic algorithms, International Journal of Advanced Manufacturing Technology, **28**, pp. 463–473.

- Rimal, B.P., Jukan, A., Katsaros, D. and Goeleven, Y., (2011) Architectural Requirements for Cloud Computing Systems: An Enterprise Cloud Approach Journal of Grid Computing, **9**, pp. 3–26.

- Risbood, K.A., Dixit, U.S. and Sahasrabudhe, A.D., (2003), Prediction of surface roughness and dimensional deviation by measuring cutting forces and vibrations in turning process, Journal of Materials Processing Technology, **132**, pp. 203–214.

- Rosenthal, A., Mork, P., Li, M.H., Stanford, J., Koester, D. and Reynolds, P., (2010), Cloud computing: A new business paradigm for biomedical informatics, Journal of Bio Medical Informatics, **43**, pp. 342–353.

- Saho, R., Martin, R.B., Zhang, J. and Morris, A.J., (1997), Confidence bounds for neural network representations, Computers Chem Engng, **21**, pp. S1173–S1178.

- Samhouri, M.S. and Surgenor, B.W., (2005), Surface roughness in grinding: On-line prediction with adaptive neuro-fuzzy inference system, Transactions of NAMRI/SME 33:57–64.

- Sanjay, C., Neema, M.L. and Chin, C.W., (2005), Modeling of tool wear in drilling by statistical analysis and artificial neural network, Journal of Materials Processing Technology, **170**, pp. 494–500.

- Santosh kumar, G.R. and Dixit, U.S., (2008), Determination of traverse speed in the Laser forming by using FEM with online learning, Proceedings of 2nd International and 23rd AIMTDR Conference, IIT Madras, Chennai, December, pp.1035–1041.

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

- Sarma, D.K and Dixit, U.S., (2007), A Comparison of dry and air-cooled turning of grey cat iron with mixed oxide ceramic tool, Journal of Materials Processing Technology, **190**, pp. 160–172.

- Sarma, D.K. and Dixit, U.S., (2009), Environment-friendly strategies for efficient utilization of cutting tools in finish turning, ASME J Manufacturing Science Engineering, **131**, 064506 (5pages)

- Sheng Y, Tomizuka M (2006) Intelligent modeling of thrust force in drilling process. Journal of Dynamic System and Measurement Control, **128**, pp. 846–856.

- Shin, Y.C. and Joo, Y.S., (1992), Optimization of machining condition with practical constraints, International Journal of Production Research, **30**, pp. 2907–2919.

- Shunmugam, M.S., Bhaskara Reddy, S.V. and Narendran, T,T., (2000), Selection of optimal conditions in multi-pass face-milling using a genetic algorithm, International Journal of Machine Tools and Manufacture, **40**, pp. 401–414.

- Sick B (2002) On-line and indirect tool wear monitoring in turning with artificial neural net works: A review of more than a decade of research. Mechanical Systems and Signal Processing, **16**, pp. 487–546.

- Silva, R.G., Reuben, R.L., Baker, K.J. and Wilcox, S.J., (1998), Tool wear monitoring of turning operations by neural network and expert system classification of a feature set generated from multiple sensors, Mechanical Systems and Signal Processing, **12**, pp. 319–332.

- Socha, K. and Dorigo, M., (2008), Ant colony optimization for continuous domain, European Journal of Operation Research, **185**, pp. 1155–1173.

- Sonar, D.K., Dixit, U.S. and Ojha, D.K., (2006), The application of radial basis function neural network for predicting the surface roughness in a turning process, International Journal of Advanced Manufacturing Technology, **27**, pp. 661–666.

- Spendley, W., Hext, G.R. and Himsworth, F.R., (1962), Sequential application of simplex designs in optimisation and evolutionary operation, Technometrics, **4**, pp. 441–461.

- Sreeram, S., Senthilkumar, A., Rahman, M. and Zaman, M.T., (2006), Optimization of cutting parameters in micro end milling operations under dry cutting conditions using genetic algorithms, International Journal of Advanced Manufacturing Technology, **30**, 1030–1039.

- Srinivas, J., Giri, R. and Yang, S.H., (2009), Optimization of multi-pass turning using particle swarm intelligence, International Journal of Advanced Manufacturing Technology, **40**, pp.56–66.

- Stone, R. and Krishnamurthy, K., (1996), A neural network thrust force controller to minimize delaminating during drilling of graphite-epoxy laminates, International Journal of Machine Tools and Manufacture, **36**, pp. 985–1003.

- Suh, S.H. and Shin, Y.S., (1996), Neural network modeling for tool path planning of rough cut in complex pocket milling, Journal of Manufacturing Systems, **15**, pp. 295–304.

- Sundaram, R.M. and Lambert, B.K., (1981), Mathematical model to predict surface finish in fine turning of steel Part I, International Journal of Production Research, **19**, pp.547–564.

- Susanto, V. and Chen, J.C., (2003), Fuzzy logic based in-process tool wear monitoring system in face milling operations, International Journal of Advanced Manufacturing Technology, **21**, pp.186–192.

- Szecsi, T., (1999), Cutting force modeling using artificial neural networks. Journal of Materials Processing Technology, **92**, pp.:344–349.

- Tandon, V. and El-Mounayari, H., (2001), A novel artificial neural networks force model for end milling, The International Journal of Advanced Manufacturing Technology, **18**, pp. 693–700.

- Tandon, V., El-Mounayri, H. and Kishawy, H., (2002), NC end milling optimization using evolutionary computation, International Journal of Machine Tools and Manufacture, **42**, pp. 595–605

- Tansel, I.N., Bao, W.Y., Reen, N.S. and Kropas-Hughes, C.V., (2005), Genetic tool monitor (GTM) for micro-end-milling operations, International Journal of Advanced Manufacturing Technology, **45**, pp. 293–299.

- Tao, F., Zhang, L., Venkatesh, V.C., Luo, Y. and Cheng, Y., (2010), Cloud manufacturing: a computing and service oriented manufacturing model, Proc. IMechE Part B: Journal of Engineering Manufacture, **225**, pp. 1969–1976.

- Taso, C.C. and Hocheng, H., (2008), Evaluation of thrust force and surface roughness in drilling composite material using Taguchi analysis and neural network, Journal of Materials Processing Technology, **203**, pp.342–348.

- Tolouei-Rad, M. and Bidhendi, I.M., (1997), On the optimization of machining parameters for milling operations, International Journal of Machine Tools Manufacture, **37**, pp. 1–16.

- Tosun, N. and Ozler, L., (2002), A study of tool life in hot machining using artificial neural networks and regression analysis method, Journal of Materials Processing Technology, **124**, pp. 99–104.

- Vijayakumar, K., Prabhaharan G, Asokan P, Saravanan R (2003) Optimization of multi-pass operations using ant colony system. International Journal of Machine Tools and Manufacture, **43**, pp.1633−1639.

- Wan M, Lu MS, Zhang WH, Yang Y (2012) A new ternary-mechanism model for the prediction of cutting forces in flat end milling. International Journal of Machine Tools and Manufacture, **57**, pp. 34–45.

- Wang, J., (1993), A neural network approach to multiple-objective cutting parameter optimization based on fuzzy preference information, Computers and Industrial Engineering, **25**, pp. 389−392

- Wang, L.X. and Mandel, J.M., (1992), Fuzzy basis functions, universal approximation and orthogonal least squares learning, IEEE Trans. Neural Networks, **3**, pp. 807−814.

- Wang X, Jawahir IS (2004) Web-based optimization of milling operations for the selection of cutting conditions using genetic algorithms. Proc IME B Journal of Engineering Manufacture, **218**, pp. 647–655.

- Wang, X. and Jawahir, I.S., (2005), Optimization of multi-pass turning operations using genetic algorithms for the selection of cutting conditions and cutting tools with tool-wear effect, International Journal of Production Research, **43**, pp. 3543−3559.

- Wang, X.F. and Feng, C.X., (2002), Development of empirical models for surface roughness prediction in finish turning, International Journal of Advanced Manufacturing Technology, **20**, pp. 348–356.

- Wang, Y.C., (2007), A note on ‘Optimization of multi-pass turning operations using ant colony system’, International Journal of Machine Tools and Manufacture, **47**, pp. 2057–2059.

- Wang, Z., Willet, P., DeAguiar, P.R. and Webster, J., (2001), Neural network detection of grinding burn from acoustic emission, International Journal of Machine Tools and Manufacture, **41**, pp. 283–309.

- Wang, Z.G., Wong, Y.S. and Rahman, M., (2004), Optimisation of multi-pass milling using genetic algorithm and simulated annealing, International Journal of Advanced Manufacturing Technology, **24**, pp. 727–732.

- Wang, Z.G., Rahman. M., Wong, Y.S. and Sun. J., (2005), Optimization of multi-pass milling using parallel genetic algorithm and parallel genetic simulated annealing, International Journal of Machine Tools and Manufacture, **45**, pp.1726–1734.

- Wu., M.P. and Liao, W.H., (2005), Internet-based machining parameter optimization and management system for high-speed machining. Trans of Nanjing University of Aeronautics and Astronautics, **22**, pp. 42–46.

- Xu, X., (2012), From cloud computing to cloud manufacturing. Robotics and Computer Integrated Manufacturing, **28**, pp.75–86.

- Yan, D., Popplewell, N., Balkrishnan, S. and Kaye, J.E., (1996), On-line prediction of surface roughness in finish turning, Engineering Design Automation, **2**, pp.115–126.

- Yang, H. and Xue, D., (2003), Recent research on developing web-based manufacturing systems: a review, International Journal of Production Research, **41**, pp. 3601–3629.

- Yeo, S.H., (1995), A multi pass optimization strategy for CNC lathe operations, International Journal of Production Economics, **40**, pp. 209−218.
- Zang, J.Y., Liang, S.Y., Yao, J., Chen, J.M. and Hang, J.L., (2006), Evolutionary optimization of machining processes, Journal of Intelligent Manufacturing, **17**, pp. 203–215.
- Zadeh, L.A., (1965), Fuzzy sets. Information and Control, **8**, pp. 338–353.
- Zadeh, L.A., (1973), Outline of a new approach to the analysis of complex systems and decision processes. IEEE Transactions of Systems, Man and Cybernetics, **3**, pp. 28–44.
- Zhu, G.Y. and Zhang, W.B., (2007), Drilling path optimization algorithm with global convergence characteristics, International Journal of Production Research, **46**, pp.2299−2311.
- Zuperl, U. and Cus, F., (2004), Tool cutting force modeling in ball end milling using multilevel perceptron, Journal of Materials Processing Technology, **153−154**, pp. 268−275.
- Zuperl, U., Cus, F., Mursec, B. and Ploi, T., (2006), A generalized neural network model of ball end milling force system, Journal of Materials Processing Technology, **175**, pp. 98−108.
- Zarei, O., Fesanghary, M., Farshi, B., Jalili Saffar, R. and Razfar, M.R., (2009), Optimization of multi-pass face-milling via harmony search algorithm. Journal of Materials Processing Technology, **209**, pp.2386–2392.

- Zheng., L., Yang, X.M., Zhang, Z.H and Liu, T.I., (2008), A web-based machining parameter selection system for life cycle cost reduction and product quality enhancement, Computers and Industry, **59**, pp. 254–261.

- Zissis, D. and Lekkas, D., (2012), Addressing cloud computing security issues, Future generation computer system, 28, pp. 583–592.

Appendix

Appendix A

Simplex Search Algorithm

Simplex search algorithm is originally proposed by Spendley *et al*. (1962). It is used for optimizing non-linear functions. Simplex is a geometric figure formed by a set of (n+1) number of points in an n-dimensional space. The method is one of the most popular and efficient direct search methods. The initial points should be selected such that they should not form a zero-volume. For example, the chosen three initial points should not lie in a straight line for a two variables optimization problem.

In the optimization process, first the function value of all initial points are evaluated and worst simplex is found. The new simplex is obtained using set of rules that steer search away from the worst simplex. The search depends on the relative function values of the simplex points. Four different operations *viz*., reflection, expansion, contraction to inner space and contraction to outer space are performed. First the centroid of all points is found out and the new simplex is obtained away from worst simplex using refection operation. The other operations are performed depending up on the relative function values all the points. Figure 4.4 (a–d) shows different simplex operation performed during optimization process.

The simplex search follows the following algorithmic steps (Rao, 1999):

Step 1: Initialize expansion co-efficient (η>1), contraction co-efficient ($\beta \in (0, 1)$) and termination parameter (ε).

Step 2: Take (n+1) number of points such that it should not form zero volume, to create initial simplex and evaluate its function values

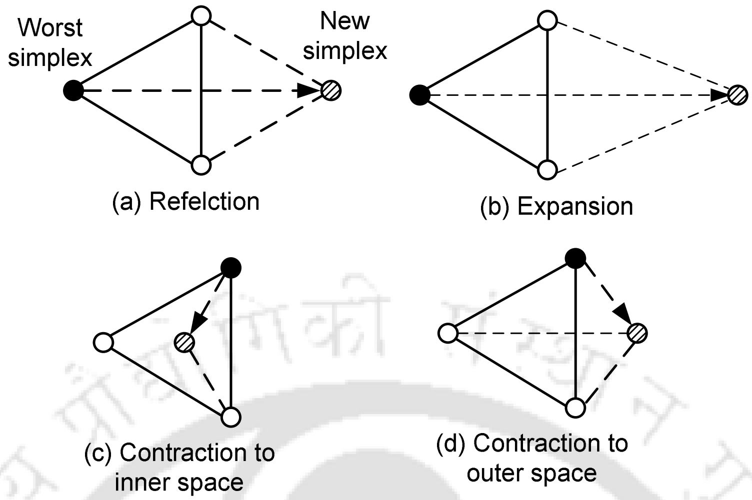


Figure A1. Simplex search operations

Step 3: Find the worst point (x_1), the best point (x_3) and the point next to worst point (x_2) based on the function values. Calculate centroids of the simplex

$$x_c = \frac{1}{n} \sum_{i=1, i \neq h}^{n+1} x_i \tag{A1}$$

Step 4: Calculate the reflected point, $x_r = 2x_c - x_1$. Set $x_{new} = x_r$.

i) If $f(x_r) < f(x_3)$, then $x_{new} = (1+\eta)x_c - \eta x_1$

ii) Else if $f(x_r) \geq f(x_1)$, then $x_{new} = (1-\beta)x_c + \beta x_1$

iii) Else if $f(x_2) < f(x_r) < f(x_1)$, then $x_{new} = (1+\beta)x_c - \beta x_1$

Calculate $f(x_{new})$ and replace x_1 by x_{new}.

Step 5: Terminate the process if $\sqrt{\sum_{i=1}^{n+1} \frac{(f(x_i) - f(x_c))^2}{n+1}} \leq \varepsilon$. (A2)

Else go to Step 3.

Appendix B

Cost Evaluation for Conducting Tool life test

One tool life test takes 20 hours of machining and consumes 20 kg of material. (Hegginbotham and Pandey, 1967)

Total cost of machining involves sum of: (i) tool cost, (ii) material cost and (iii) operating cost.

(i) Cost of cutting tool per edge

The coated carbide tool having 4 cutting edges is considered. The cost of the tool is ₹ 500.

So, the cost of the cutting tool per edge is ₹ 125.

(ii) Material cost

Work material is cold rolled steel (0.35% carbon) having 130 BHN.

The cost of one kg of material is ₹ 65.00

Total material cost for 20 kg = ₹ 1296.00

(iii) Operating cost for machining

It composed of cost of labour operator, cost of electricity, machine depreciation cost and machine maintenance cost.

a) Labour cost

Let the cost of one labour operator per day is ₹ 600.00.

The operator works 8 hrs per day.

So, wage of the labour operator per hour = ₹ 75.00.

b) Electricity cost

The lathe machine uses 11 kW electric motor. The cost of electric power per hour is ₹ 5.85. The cost of electricity per kW per hour = ₹ 64.00.

c) Machine depreciation cost

Considering the straight line depreciation method the depreciation cost is calculated by spreading the cost of the machine evenly over the entire life.

Let the cost of the lathe machine is ₹ 12,00,000.00

Life period of the machine is 10 years.

Total Depreciation cost = Cost of the fixed asset / Life period

= ₹ 12,00,000/10 years

= ₹ 1,20,000.00

Consider the machine works on 25 days in a months and 8 hours per day for each year.

The depreciation cost per hour = $\frac{1,20,000}{8 \times 25 \times 12}$ = ₹ 50.00

d) Machine maintenance cost

The maintenance includes engine oil, lubrication oil, grease, motor belts, swarf cleaning *etc*. Considering all these, the cost of maintenance of one machine per hour is ₹ 13.00.

The operating cost of the machine per hour

= labour cost + electricity cost + machine depreciation cost + machine maintenance cost.

= ₹ 75.00+ ₹ 64.00+ ₹ 50.00+ ₹ 13.00 = ₹ 202.00

Operating cost for 20 hours of machining = ₹ 4041.00

Total cost involved for conducting one tool life test

= cost of cutting tool per edge + material cost for 20 kg + operating cost for 20 hrs

= ₹ 125.00 + ₹1296.00 + ₹ 4041=₹ 5462.00 = $91.03 (₹ 60.00=$1)

Appendix C

Semi Virtual Milling

The semi virtual milling consists of three modules: (i) NN based surface roughness prediction module (ii) empirical relation based tool life prediction module and (iii) tangential cutting force prediction module based on empirical relation. For a given process parameters (v, f', d), the milling machine provides center line average (CLA) value of surface roughness (R_a), tool life (T) and tangential cutting force (F).

(i) NN module for Surface roughness prediction

For predicting surface roughness of finish milling process ANN model is developed. An experimental data from the work of Lou and Chen (1999) for end milling of 6061 aluminium alloy using HSS end mill cutter, is used in this work. The cutting speed, feed/tooth and depth of cut are three input neurons of a neural network on which the surface roughness depends. The data sets are in the range of cutting parameters as cutting speed (v): 44.9–89.7 m/min; feed (f'): 0.025–0.23 mm/tooth; and depth of cut (d): 0.254–1.27 mm.

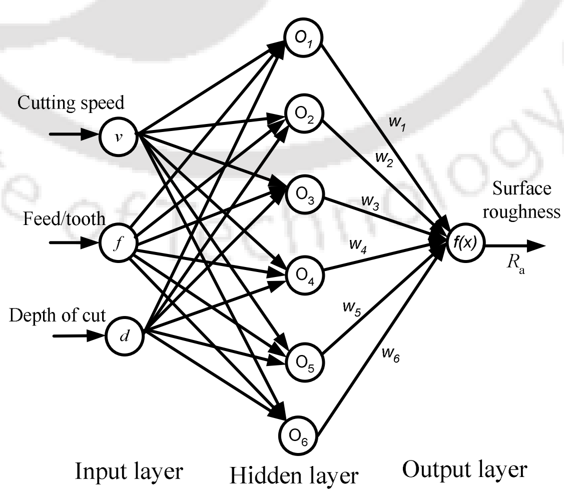


Figure C1. NN model for surface roughness prediction

Table C2. Testing data of NN model

Sl. No.	v (m/min)	f' ($x10^{-2}$ mm/tooth)	d (mm)	R_a (μm)
1	44.9	5.08	0.254	1.65
2	44.9	10.16	0.762	2.59
3	44.9	10.16	1.27	2.39
4	44.9	15.24	0.254	3.07
5	44.9	20.32	0.762	4.75
6	59.8	3.81	1.270	1.98
7	59.8	3.81	0.254	1.57
8	59.8	7.62	0.762	2.13
9	59.8	11.43	0.254	3.50
10	59.8	11.43	0.762	3.15
11	59.8	15.24	1.270	3.16
12	77.8	3.05	0.254	1.27
13	77.8	3.05	1.270	1.80
14	77.8	6.10	0.762	2.51
15	77.8	9.14	0.254	2.92
16	77.8	12.2	0.762	2.77
17	89.7	2.54	1.270	1.42
18	89.7	5.07	1.270	2.39
19	89.7	7.61	0.254	3.02
20	89.7	10.15	0.254	3.02
21	89.7	10.15	1.270	2.77
22	89.7	7.61	1.270	2.64
23	44.9	15.24	0.762	3.73

Table C1. Training data of NN model

Sl. No	v (m/min)	f' ($\times10^{-2}$ mm/tooth)	d (mm)	R_a (µm)
1	44.9	5.08	1.270	1.83
2	44.9	23.02	1.270	4.37
3	59.8	3.81	0.254	1.47
4	59.8	7.62	1.270	2.34
5	59.8	15.24	0.254	4.14
6	77.8	3.00	0.762	1.60
7	77.8	9.14	1.270	2.41
8	77.8	12.20	1.270	3.07
9	89.7	2.54	0.254	0.94
10	89.7	5.07	0.762	2.08

For computation of weights, 'logsig' activation function is used for neurons in hidden layer, whereas 'purelin' is used for output neurons. The network is trained with the 'trainlm' function of MATLAB that works on back propagation algorithm. The network performance is assessed by using root mean square fractional error. A neural network topology with one hidden layer and with six neurons as shown in Figure A1 having least effective error is selected as best network for this case.

The training performance of the network shows the percentage of error to be within ±20% limits and maximum error recorded is 15.6%. The network is validated using 23 data sets and results obtained are within ±20% limits.

(ii) Tool life prediction module

For the tool life evaluation of end mill cutter an empirical relation proposed by Tolouei-Rad and Bidhendi (2007) is used:

$$T = \frac{60}{0.29}\left(\frac{C\left(0.2d/f\right)^{0.14}}{\left(df\right)^{0.28}v}\right)^{1/0.15}, \tag{C1}$$

where T is the tool life in minutes, v is the cutting speed in m/min, C_2 is the constant for 60 min tool life when the area of cut is 1 mm^2, f is the feed in mm/rev$(= f' \times z)$, d is the depth of cut in mm, and g, w, and n are exponents for different tool and work material combination. The factor Q is contact proportion of cutting edges with work piece per revolution. However in online optimization using proposed methodology, the 'tool life' data is directly obtained in a real machining environment**.** Note that this equation is used to develop a black box module of tool life to simulate real machining.

(iii) Cutting force prediction module: The module evaluates tangential cutting force developed during end milling process. An empirical relation is used to evaluate the cutting force. An experimental data presented in (Wan *et al.*, 2012) is used to obtain the empirical relation. However during online optimization the cutting force is directly obtained in a real machining environment**.**

Publications from the Present Thesis

International Journals

1. M.Chandrasekaran, M.Muralidhar, C.M.Krishna and U.S.Dixit, Application of soft computing techniques in machining performance prediction and optimization: a literature review, International Journal of Advanced Manufacturing Technology (IJAMT), 46(5–8), pp. 445–464, 2010.
2. M.Chandrasekaran, M.Muralidhar and U.S.Dixit, An interactive online finish end milling process optimization, International Journal of Applied Engineering Research, 6(5), pp. 949–959, 2011.
3. M.Chandrasekaran, M.Muralidhar and U.S.Dixit, Online optimization of multipass machining based on cloud, International Journal of Advanced Manufacturing Technology (IJAMT), 65(1–4), pp. 239–250, 2013.
4. M.Chandrasekaran, M.Muralidhar and U.S.Dixit, Online optimization of a finish turning process: Strategy and experimental validation, International Journal of Advanced Manufacturing Technology (IJAMT), (Revised manuscript submitted).

Book Chapter

1. M.Chandrasekaran, M.Muralidhar, C.M.Krishna and U.S.Dixit, Online Machining Optimization with Continuous Learning. In: Davim JP (Ed) Computational Methods for Optimization Manufacturing Technology, IGI Global, Hershey, 2012.

International/National Conferences

1. M.Chandrasekaran, M.Muralidhar and U.S.Dixit, Optimization of Engineering problems by fuzzy set theory: An Application to Multipass Turning process, International Multi Conference on Intelligent Systems and Nanotechnology, Institute of Science and Technology Kalwad (ISTK) Jamuna Nagar, Haryana, India, February 26–28, 2010.

2. M.Chandrasekaran, M.Muralidhar and U.S.Dixit, Optimization of Finish Turning process with Online Learning, 3rd International and 24th All India Manufacturing Technology Design and Research (AIMTDR) conference, pp. 951–956, Andhra University, Visakhapatnam, Andhra Pradesh, India, December 13–15, 2010.

3. M.Chandrasekaran, M.Muralidhar and U.S.Dixit, Fuzzy set based online optimization of finish milling process, National Conference on Advanced Design and Manufacture (NCADM 2011), pp.132–137, Einstein College of Engineering, Thirunelveli, Tamilnadu, India, January 6–7, 2011.