Optimization of Turning Process Parameters using Particle Swarm and Ant Colony Optimization Techniques

A Project Report

Submitted by

V.Sri Ambicai - 71205402008

in partial fulfillment for the award of the degree
of

Master of Engineering
in
CAD/CAM

DEPARTMENT OF MECHANICAL ENGINEERING
KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE – 641 006

ANNA UNIVERSITY :: CHENNAI 600 025

JUNE– 2007

ANNA UNIVERSITY :: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report entitled "**Optimization of Turning Process Parameters using Particle Swarm and Ant Colony Optimization Techniques**" is the bonafide work of

Ms. V.Sri Ambicai -Register No. 71205402008

Who carried out the project work under my supervision

Signature of the Head of the Department

Signature of the supervisor
(Prof.T.Kannan)

Internal Examiner 4/7/02

4/7/7
External Examiner

DEPARTMENT OF MECHANICAL ENGINEERING
KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE - 641 006

Contemporary Approaches in Design and Manufacturing

CADAM 07

CADAM '07

A.C.College of Engineering and Technology
Karaikudi - 04, Tamil Nadu.

26th & 27th April 2007

Certificate of Recognition

This is to certify that ~~Mr.~~ / Ms. /Mrs. V. SRI AMBICAI has presented

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

r entitled OPTIMIZATION OF TURNING PROCESS PARAMETERS USING INTELLIGENT TECHNIQUE in the National Conference

1 '07 organized by Department of Mechanical Engineering, A.C. College of Engineering & Technology,

ikudi - 04.

uthors :

Syed Abu Thaheer
Co-Ordinator

Prof. K.R.Viswanathan
Chairman

Dr.R.Sundararajan
Principal

REVEAL THE UNREVEALED

COIMBATORE - 641 006. TAMILNADU - INDIA

ISO 9001

DEPARTMENT OF MECHANICAL ENGINEERING

National Conference

on

"ADVANCES IN MECHANICAL SCIENCES (AIMS 2007)"

Certificate

certify that ~~Dr.~~ / ~~Mr.~~ / Ms. V. SRI AMBICAI, P G Scholar *has*

and presented a paper titled DEVELOPMENT OF MATHEMATICAL MODELS TO PREDICT TOOL WEAR IN CNC TURNING PROCESS

TE Sponsored National Conference on "ADVANCES IN MECHANICAL SCIENCES (AIMS 2007)"

- 24 March 2007.

C. Velmurugan
karan & Mr.C.Velmurugan
anizing Secretaries

N. Gunasekaran
Dr.N.Gunasekaran
Convener & HOD$_{i/c}$

Dr.Joseph V.Thanikal
Principal

ABSTRACT

The ultimate aim of turning is to remove workpiece material in a single cut rather than a lengthy grinding operation in order to reduce processing time, production cost, surface roughness and setup time and to remain competitive. The process of turning offers many potential benefits compared to conventional grinding operation. In this work, tool wear, tool life, quality of surface turned, and amount of material removed are also predicted. Similarly, three process parameters (cutting speed, feed, and depth of cut) were identified for optimization subjected to realistic process constraints. This optimization problem formulated as a multi-objective, multi-variable, and non-linear programming problems. Several conventional techniques had been suggested in the literature for solving this problem. But these techniques are not robust and take lot of time to find the global optimum and are difficult to understand and implement. In order to overcome the difficulties with conventional techniques, the new techniques called Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) Techniques are implemented in this work. PSO is a simple and powerful technique based on the concept of social interaction to problem solving. In PSO, a swarm search of 'n' individuals communicate either directly or indirectly with one another for getting the search direction. In ACO studies, an artificial system that take inspiration from the behavior of real ant colonies and which is used to solve discrete optimization problems. The proposed algorithm starts with 20 particles (solutions) and search for the new ones by updating the velocities. Maximum of 500 iterations were performed and the solutions were obtained. Program has been written using C language. The solution obtained by this procedure was found to be superior. The computational effort is very less and easy to implement.

ஆய்வுச் சுருக்கம்.

கடைசலின் உச்சகட்ட நோக்கம் வொர்க் பீஸ் மெட்டீரியலை ஒரே தடவை வெட்டி நீக்குதல் ஆகும். இதனால் உற்பத்தி விலை மற்றும் சொரசொரப்பான பரப்பு ஆகியவற்றை கிரைண்டிங் முறையைவிட குறைக்க இயலும்.

கடைசல் முறை கிரைண்டிங் முறையைவிட பல முக்கிய பலன்களை பெற்றுள்ளது. இந்த ஆய்வில் டூல் தேய்மானம், டூல் ஆயுள் பொருளின் மேற்பரப்பு தரம் மற்றும், நீக்கும் மெட்டீரியலின் அளவு அனைத்தும் மதிப்பிடப்பட்டுள்ளது. அதே மாதிரி மூன்று பெராமீட்டர்கள் ஆப்டிமைசேஷனுக்காக பரிசோதிக்கப்பட்டது. இந்த ஆப்டிமைசேஷன் ஆய்வில் மல்டி-ஆப்ஜெக்டிவ், மல்டி-வேரியபில் மற்றும் நான்-லீனியர் புரோகிராடிங் உருவாக்கப்பட்டது.

பல ஆப்டிமைசேஷன் மாற்று முறைகள் இலக்கியத்தில் கூறப்பட்டுள்ளது, ஆனால் இம்மாற்று முறைகள் எல்லாம் புரிந்து கொள்வதற்கும், நடைமுறைப்படுத்துவதற்கும் கடினம் மற்றும் விடையை கண்டுபிடிக்க அதிக கால அளவை எடுத்துக் கொள்ளும்.

இப்பிரச்சனையினை வெல்ல புதிய பார்டிகில் ஸ்வார்ம் ஆப்டிமைசேஷன் மற்றும் ஆன்ட் காலனி ஆப்டிமைசேஷன் ஆகிய முறைகள் இந்த ஆய்வில் உபயோகிக்கப்பட்டுள்ளது. பார்டிகில் ஸ்வார்ம் ஆப்டிமைசேஷன் என்பது எளிய மற்றும் வலிமையான முறையாகும். இது சமூகத் தொடர்பு என்ற கோட்பாட்டின் அடிப்படையில் செயல்படுதல்லாகும். இம் முறையில் ஒரு ஸ்வார்ம் ஆனது தேடுதல் திசையை அரிய ‘n’ ஸ்வார்ம்களை நேரடியாகவோ, மறைமுகமாகவோ தொடர்பு கொள்ளும். ஆன்ட் காலனி ஆப்டிமைசேஷன் முறையில் ஒரு தற்காலிக அமைப்பானது ஆன்ட் காலனீஸின் நடவடிக்கையின் மூலம் உள்ளுணர்வி கொண்டு மேற்கூறிய ஆப்டிமைசேஷன் பிரச்சனைகளை நீக்க கூடியது. மேற்கூறிய அல்காரிதம் இருபது சொல்யூசன் ஆரம்பித்து புதிய ஆண்டுகளை தேடி சேகரித்துக் கொள்கிறது. அதிகபட்சமாக ஐநூறு துகள்கள் வரை கிடைக்கும். ‘சி’ லேங்வேஜ் பயன்படுத்தி புரோக்ராம் எழுதப்பட்டுள்ளது. இம்முறையினால் ஏற்படும் விளைவு (விடை) உயர்வானது. இம்முறைகளை நடைமுறைப்படுத்துவது எளிது, மற்றும் கணணியின் பங்கு மிகவும் குறைவு.

ACKNOWLEDGEMENT

The author grateful to her guide **Prof.T.Kannan**, Assistant Professor of Mechanical Engineering, Kumaraguru College of Technology, Coimbatore for his excellent, utmost motivation, valuable advice, untiring support, timely suggestion, constant encouragement, enthusiasm, relentless patience, and inspiration throughout the study, holding in all the places.

The Author is also grateful to **Dr. N. Gunasekaran** , Professor of Mechanical Engineering for his valuable suggestions and timely help towards her project.

The Author expresses her deep gratitude to **Dr.JosephV.Thanikal**, Principal, Kumaraguru College of Technology , Coimbatore for patronizing her, besides providing all assistance.

The Author owes her immense gratitude and thanks to **Dr.P.Palanisamy**, Assistant Professor of Mechanical Engineering, Kumaraguru College of Technology, Coimbatore for his valuable guidance, encouragement and support during the entire course of this project.

The Author also expresses her sincere thanks to **Dr.R.Saravanan**, Head of the Department of Mechatronics Engineering for his valuable suggestions and timely help towards her project.

The Author also expresses her sincere thanks to the Faculty and Staff members, those who helped in carrying out the project work.

CONTENTS

Title		Page No.
Certificate		ii
Abstract		iii
Acknowledgement		v
Contents		vii
List of Tables		xi
List of Figures / Photos		xix
List of Symbols & Abbreviations		xxi
CHAPTER-1 INTRODUCTION		
1.1.	Turning Process	2
1.2.	Design of Experiments	2
1.3.	Particle Swarm Optimization	2
1.4.	Ant Colony Optimization	2
1.5.	Importance of Project	3
CHAPTER-2 LITERATURE SURVEY		
2.1.	Introduction	5
2.2.	Selection of Machining Parameters	5
2.3.	About Researchers Explanation	5
CHAPTER-3 OPTIMIZATION TECHNIQUES		
3.1.	Introduction	9
3.2.	Need for Optimization	9
3.3.	Types of Optimization Techniques	9
3.4.	Types of Optimization Problems	10
3.5.	Optimization Techniques – An Overview	11
CHAPTER-4 DESIGN OF EXPERIMENTS		
4.1.	Introduction	16

4.2. Purpose of Experimentation 16

4.3. Design of Experiments 16

4.4. Design of Experiment Process 17

4.5. Response Surface Methodology 18

4.6. Regression Equation 19

4.7. Objective Function 19

4.8. Constraints 19

4.8.1. Surface Roughness 19

4.8.2. Tool Life 19

4.8.3. Cutting Forces 20

4.9. Chosen Input Parameters 21

4.10. Chosen Output Parameters 21

4.11. Experimental Details & Specifications 21

4.11.1 Aluminum 25

4.12. Experimental Work 25

4.13. Experimental Design Procedure 25

4.13.1. Identification of Factors and Responses 25

4.13.2. Finding the Limits of the Process Variables 26

4.13.3. Development of Design Matrix 26

4.13.4. Conducting the Experiments as Per Design Matrix 28

4.13.5. Recording the Responses 28

4.13.6. Development of Mathematical Model 28

4.13.7. Checking the Adequacy of the Developed Models 31

4.13.8. Conducting the Conformity Tests 31

CHAPTER-5 PARTICLE SWARM OPTIMIZATION

5.1. Introduction 33

5.2. Background of Artificial Life 33

5.2.1. Bird Flocking 34

5.3. Particle Swarm Optimization Technique 34

5.4. Algorithm of PSO 35

5.5. PSO Parameters Control 35

5.6. Particle Swarm Algorithm 37

5.7. Objective Function 39

5.8. Formulas Used 39

5.9. Results of PSO 40

CHAPTER-6 ANT COLONY OPTIMIZATION

6.1. Introduction 42

6.2. Ant Colony Algorithm 43

6.2.1. Schemes of the Ant Colony Algorithm 48

6.2.2. Distribution of Ants 48

6.2.3. Global Search 48

6.2.3.1 Crossover 48

6.2.3.2 Mutation 50

6.2.3.3 Trial Diffusion 50

6.2.4. Local Search 50

6.2.5. Algorithm 51

6.2.6. Results of ACO 53

CHAPTER-7 CONCLUSION 55

REFERENCES

LIST OF TABLES

Table	Title	Page No.
3.1	Summary of Machining Optimization Techniques	13
4.1	Limits of Parameters	26
4.2	Design Matrix and the Observed Values of Machining Parameters	27
4.3	Calculated Regression Coefficients	29
4.4	Design Matrix and the Predicted Values of Machining Parameters	30
5.1	Sample Results of PSO	40

LIST OF FIGURES

Figure	Title	Page No.
4.1	Cutting Forces Acting on a Tool	20
4.2	PSG CNC 110 Lathe	22
4.3	Turned Components	22
4.4	Experimental Setup	23
5.1	Flowchart of PSO Algorithm	38
5.2	PSO - Cost Curve	40
6.1	Concept of Ant Colony Algorithm	43
6.2	Representation of Superior Solutions and Inferior Solutions	48
6.3	A Representation of N-G Parents & G Child	49
6.4	Distribution of Ants for Local & Global Search	49
6.5	Flowchart of ACO Algorithm	52
6.6	ACO – Cost Curve	53

LIST OF SYMBOLS

Y - Dependent Variable

x - Independent Variable

c - Regression Residual

X_1 - Cutting Speed

X_2 - Feed

X_3 - Depth of Cut

v[] - Particle Velocity

present[] - Current Particle

pbest[] - Best Solution among Each Particle

gbest[] - Global Best

rand( ) - Random Number between (0,1)

ω - Inertia Weights

c1 & c2 - Learning Factors

Ra - Surface Roughness

Fc - Cutting Force

P - Power

T - Tool Life

Ra_{min} - Target Value for the Surface roughness

Fc_{min} - Target Value for the Cutting Force

P_{min} - Target Value for the Power

T_{min} - Target Value for the Tool Life

R - Maximum Step Size

T - Ratio of Current Iteration Number to Total Number of Iterations

b - Positive Parameter

i - Region Index

ρ - Evaporation Rate

τ_i (k) - Pheromone Trial

τ_i (t) - Trial Associated with Solution at Time t

w1 to w4 - Weights to each Response

LIST OF ABBREVIATIONS

DNA	Di Nucleic Acid
PSO	Particle Swarm Optimization
ACO	Ant Colony Optimization
DOE	Design of Experiments
RA	Regression Analysis
CNN	Computational Neural Networks
ANOVA	Analysis of Variance
COF	Combined Objective Function
CO	Combinational Optimization

CHAPTER 1

INTRODUCTION

1.1 TURNING PROCESS

Turning produces solids of revolution that can be tightly toleranced because of the specialized nature of the operation. Turning is performed on a machine called a lathe in which the tool is stationary and the part is rotated. Workpieces, large and small can be machined in one setup through multi-tasking capabilities on single- and multiple-spindle machines. For larger workpieces, workhorse engine lathes, vertical turning centers, and a new generation of Computer Numerical Control lathes deliver the required power and accuracy. Automation is provided by bar feeders, robots, and pallet delivery systems.

1.2 DESIGN OF EXPERIMENTS

Design of experiments is a collection of procedures used to create a set of design samples. Statistical design of experiments involves the process of planning and designing an experiment so that appropriate data can be collected and then analyzed by statistical methods. Application areas include economic parameter analysis in an open market environment, in beach nourishment, in Di Nucleic Acid (DNA) sequence assembly, in biotechnology and health informatics, in management, energy and the sciences etc,.

1.3 PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization is similar to evolutionary computation techniques in that a population of potential solutions to the optimal problem under consideration is used to probe the search space. Each potential solution is assigned a randomized velocity, and the potential solutions called particles correspond to individuals. Application areas of Particle Swarm Optimization include water quality, in power system operations, in stock markets, in intensity-modulated radiotherapy planning, in sensor networks, in signal detection and blind extraction etc,.

1.4 ANT COLONY OPTIMIZATION

Ant Colony Optimization (ACO) is a paradigm for designing metaheuristic algorithms for combinatorial optimization problems. The ant colony optimization algorithm is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs. They are inspired by the behavior of ants in finding paths from the colony to food. Application areas include routing in telecommunication networks, in vehicle routing problems, routing in wireless sensor networks, in 2D and 3D hydrophobic polar protein folding problem, in software test data generation etc,.

1.5 IMPORTANCE OF THE PROJECT

Optimum machining parameters are of great concern in manufacturing environments, where economy of machining operation plays a key role in competitiveness in the global market. Optimization analysis of the machining process is usually based on minimizing production cost, minimizing production rate, or obtaining the finest possible surface finish by using the empirical relationships between tool life and the operating parameters. For solving machining optimization problems, various conventional techniques had been used so far that have created problems when applied to turning process. To overcome the above problems, Particle swarm and Ant colony optimization techniques are used in this work. Moreover, PSO and ACO converge to the global optimal solutions at a faster rate.

CHAPTER 2

LITERATURE SURVEY

2.1 INTRODUCTION

Recently different methods have been reported in the literature to optimize the machining parameters using various non-conventional methods. A number of researchers have dealt with the optimization of machining parameters. The turning operation was considered by many researchers and starting from graphical methods to geometric programming methods have been used to determine the optimum speed and feed.

2.2 SELECTION OF MACHINING PARAMETERS

These include - A simplified approach to optimum selection of machining parameters, machining economics and industrial data manuals, optimization of the constrained machining economics problem by geometric programming, a probabilistic approach to determination of the optimum cutting conditions and machining parameters as cutting speed, feed and depth of cut with constrains as surface roughness, cutting force, power, tool life and tool wear.

2.3 ABOUT RESEARCHERS EXPLANATION

Taylor (1907) showed that an optimum or economic cutting speed exists which could maximize material removal rate. Considerable efforts are still in progress on the use of handbook-based conservative cutting conditions and cutting tool selection at the process planning level. Russian handbooks as well in the American handbook, textbook most authors have not included discussions on the more modern tools, new work materials and tool coatings. Taylor (1907) and his famous tool life equation, different analytical and experimental approaches for the optimization of machining parameters have been investigated.

Gilbert (1950) studied the optimization of machining parameters in turning with respect to maximum production rate and minimum production cost as criteria. Armarego and Brown (1969) investigated unconstrained machine-parameter optimization using differential calculus. Brewer and Rueda (1963) carried out simplified optimum analysis for non-ferrous materials. Brewer (1966) suggested the use of Lagrangian multipliers for optimization of the constrained problem of unit cost, with cutting power as the main constraint. While our previous research focused on tolerance study (Feng et al. 2001, Feng and Kusiak 2000, and Feng and Kusiak 1997), this one attempts to develop empirical models with some data mining techniques, such as regression analysis (RA) and computational neural networks (CNN), to help the selection of cutting parameters and the improvement of surface roughness.

Gopalakrishnan and Khayyal (1991) described the design and development of an analytical tool for the selection of machine parameters in turning. Geometric programming was used as a basic methodology to determine values for feed rate and cutting speed that minimize the total cost of machining SAE1045 steel with cemented carbide tools of ISO P-10 grade. Surface finish and machine power were taken as the constraints while optimizing cutting speed & feed rate for a given depth of cut.

Agapiou (1992) formulated single-pass and multi-pass machining operations. Production cost and total time were taken as objectives and a weighting factor was assigned to prioritize the two objectives in the objective function. He optimized the number of passes, depth of cut, cutting speed and feed rate in his model, through a multi-stage solution process called dynamic programming. Several physical constraints were considered and applied in his model. In his solution methodology, every cutting pass is independent of the previous pass; hence the optimality for each pass is not reached simultaneously.

Cochran and Cox (1962) quoted Box and Wilson as having proposed response surface methodology for the optimization of experiments. Lambert and Taraman (1973) developed an adequate mathematical model for the cutting force acting on a carbide tool while machining SAE1018 cold-rolled steel in a turning operation and then utilized the model in the selection of the levels of the machining variables of cutting speed, feed rate, and depth of cut such that the rate of metal-removal could be at the highest possible value without violating some given force restriction.

Bhattacharya (1970) optimized the unit cost for turning, subject to the constraints of surface roughness and cutting power by the use of Lagrange's method. Walvekar and Lambert (1970) discussed the use of geometric programming to selection of machining variables. They optimized cutting speed and feed rate to yield minimum production cost.

Petropoulos (1973) investigated optimal selection of machining rate variables that is cutting speed and feed rate, by geometric programming. Sundaram (1978) applied a goal programming technique in metal cutting for selecting levels of machining parameters in a fine turning operation on AISI4140 steel using cemented tungsten carbide tools. Ermer and Kromodiharajo (1981) developed a multi-step mathematical model to solve a constrained multi-pass machining problem. They concluded that in some cases with certain constant total depth of cut, multi-pass machining was more economical than single-pass machining, if depth of cut for each pass was properly allocated. They used high-speed steels (HSS) cutting tools to machine carbon steel.

Hinduja (1985) described a procedure to calculate the optimum cutting conditions for turning operations with minimum cost or maximum production rate as the objective function. Tsai (1986) studied the relationship between multi-pass and single-pass machining. He presented the concept of a break-even point that is there is always a point, a certain value of depth of cut, at which single pass and double pass machining are equally effective. Taraman (1974) investigated multi-machining output multi-independent variable turning research by response surface methodology.

Hassan and Suliman (1990) presented mathematical models for the prediction of surface roughness, tool vibration, power consumption and cutting time, when turning medium carbon steel using tungsten carbide tools under dry condition. El Baradie (1993) presented a study of a surface roughness model for turning grey cast iron (154BHN) using tipped carbide tools under dry conditions and for a constant depth of cut (d=1.00mm). Li and Mathew gave the classification for various direct and indirect methods for on-line measurement of tool wear particularly during turning operation, which includes the tool wear and failure monitoring techniques for turning. The variations in the hardness of material and case depth are the other parameters affecting surface finish and tool wear is included in an experimental study of the impact of turning parameters on surface roughness. Dawson and Kurfess stated that the experimental and theoretical roughness values match very well, except at low feed values. Chang-Xue et al. explained the various parameters affecting surface roughness of the turned surface. The method of decreasing machining time and reducing the number of machines required in hard turning, compared to conventional grinding was described by Konig et al. Negishi et al. studied the maximum tool life period of carbide tools. Kopac et al. presented the analysis of machining parameters in the finished turning process.

CHAPTER 3

OPTIMIZATION TECHNIQUES

3.1 INTRODUCTION

Optimization analysis of the machining process is usually based on minimizing production cost, maximizing production rate, or obtaining the finest possible surface finish by using the empirical relationships between tool life and the operating parameters.

3.2 NEED FOR OPTIMIZATION

Optimization algorithms are becoming increasingly popular in engineering design activities, primarily because of the availability and affordability of high-speed computer. They are extensively used in those engineering problems where the emphasis is on maximizing or minimizing a certain goal. For example, optimization is routinely used in aerospace deign activities to minimize the overall weight of the aircraft. Thus the minimization of the weight of the aircraft components is of major concern to aerospace designers. Chemical engineers, in the other hand, are interested in designing and operating a process plant for an optimum rate of production. Mechanical engineers design mechanical components for the purpose of achieving either a minimum manufacturing cost or a maximum component life.

Production engineers are interested in designing optimum schedule of the various machining operations to minimize the ideal time of machines and the overall job completion time. Civil engineers are involved in designing buildings, bridges, dams and other structures in order to achieve a minimum overall cost or maximizing safety or both. Electrical engineers are interested in designing communication networks so as to achieve minimum time for communication from one node to another.

All the above-mentioned task either minimization or maximization (collectively known as optimization) of an objective requires knowledge about the working principles of different optimization methods.

3.3 TYPES OF OPTIMIZATION TECHNIQUES

The types of optimization techniques are given below :

- Single or multi variable optimization
- Single or multi objective optimization
- Constrained or unconstrained optimization

- Linear or non-linear optimization
- Non-traditional optimization algorithms
  - Genetic algorithm
  - Particles swarm optimization
  - Neural networks
  - Simulated annealing
  - Fuzzy logic

3.4 TYPES OF OPTIMIZATION PROBLEMS

Existence of constraints:

An optimization problem can be classified as a constrained or an unconstrained one, depending upon the presence or not of constraints.

Nature of the equations:

Optimization problems can be classified as linear, quadratic, polynomial, non-linear depending upon the nature of the objective functions and the constraints. This classification is important, because computational methods are usually selected on the basis of such a classification, i.e. the nature of the involved functions dictates the type of solution procedure.

Admissible values of the design variables:

Depending upon the values permitted for the design variables, optimization problems can be classified as integer or real valued, and deterministic or stochastic.

Applications in turning optimization problems

Hard turning machining exhibits a unique behavior, which is different than conventional turning operations. Application of hard turning technology can be improved by utilizing advanced optimization algorithms which helps manufacturers to make educated decisions when faced with multiple objectives to be satisfied. Finish hard turning, using cubic boron nitride (CBN) tools, allows manufacturers to simplify their processes and still achieve the desired surface roughness, which can compete with grinding operations. Surface roughness is mainly a result of process parameters such as tool geometry (i.e., nose radius, edge geometry, rake angle etc.) and cutting conditions (feed rate, cutting speed, depth of cut etc.). In finish hard turning, tool wear becomes an additional parameter affecting surface quality of finished parts. Performance of CBN cutting tools is highly dependent on cutting conditions such as cutting speed, feed, feed-rate, and depth of cut. Cutting speed and depth of cut have a particularly significant influence on tool life. It has been observed that decreasing feed rate helps obtain a good surface finish but increases machining time. High cutting speeds may help reduce the surface roughness, but since tool life at high cutting speeds is just a few couple minutes this solution is not applicable. In some cases surface roughness is improved with increasing tool wear; therefore, attention should be paid to the relation between tool wear and surface roughness. It is therefore crucial to obtain a group of optimum conditions, which may serve different purposes under different circumstances.

3.5 OPTIMIZATION TECHNIQUES – AN OVERVIEW

Most traditional optimization methods used in industrial engineering problems can be divided into two broad classes: Direct search method and Gradient search methods. In which direct search method requires only the objective function values and gradient search method requires gradient information either exactly or numerically. One common characteristic of most of this method is that they all work by point-by-point basic. An algorithm starts with an initial point (usually supplied by the user) and depending on the transition rules used in the algorithm a new point is determined. Essentially, algorithms vary according to the transition rule used to update a point.

Among the direct search method, pattern search method and conjugate direction method have been extensively used. In pattern search methods at every iteration a search direction is related according to a combination of exploratory search locally and a pattern search regulated by some heuristics rules. Often this method gets terminated prematurely and degenerates to a sequence of exploratory moves. In conjugate direction methods, a set of conjugate directions are generated using the history of a previous few iterations. Even though this method has been very popular, the common problem with this method is that often the search directions become independent and occasional restarts are necessary. Moreover this algorithm has a convergence proof of well-behaved, unimodel functions. Box's direct search method is different from these methods in that the algorithm works with a number of points instead of a single point.

The algorithm resumes with an evenly distributed set of points. At every iteration a new set of points is created around the best point of the previous iteration. Since no information about the rejected points is used in choosing new points in subsequent iterations, the method is slow and inefficient; but the waiting time to obtain the global solution may be too large to make the search useful in real world problems. Simplex search method uses a simplex point to create a new simplex according to some rules depending on the objective function values at all points of the simplex. The essential idea is to generate the whole space cannot be spanned, the simplex search is blind and cannot general find the global solution.

Besides, some random search techniques are also used extensively especially in problems where no knowledge about the problem is known or where the search is large or where none of the traditional methods has worked. These methods are also used to find a feasible starting point especially if the number of constraints is large.

It is to be mentioned here that this discussion is not to say that these traditional are useless, infact they have been extensively used in many engineering optimization problems. The suggestion here is that if the solutions obtained by some traditional methods are satisfactory, there is no problem. But if the solutions obtained are not satisfactory or some known methods can not be applied, then the user either has to learn and use some other optimization methods suitable to solve that problem.

The latest techniques for optimization include fuzzy logic, scatter search technique particle swarm optimization technique, genetic algorithm, taguchi technique, ant colony optimization technique and response surface methodology.

Fuzzy logic given in Table 3.1, has great capability to capture human commonsense reasoning, decision-making and other aspects of human cognition. Koska (1997) shows that it overcomes the limitations of classic logical systems, which impose inherent restrictions on representation of imprecise concepts.

Genetic algorithm given in Table 3.1, based on mechanics of natural selection and natural genetics, which are more robust and more likely to locate global optimum. Scatter search technique given in Table 3.1, originates from strategies for combining decision rules and surrogate constraints. Genichi Taguchi is a Japanese engineer who has developed both the philosophy and methodology for process or product quality improvement that depends heavily on statistical concepts and tools, especially statistically designed experiments.

TABLE 3.1 SUMMARY OF MACHINING OPTIMIZATION TECHNIQUES

Technique	References	Tools Used	Remarks
Lagrange's method	Brewer (1966);Bhattacharya et al. (1970)	Lagrange's multiplier	Used for constrained optimization
Geometric programming	Walvekar and Lambert (1970); Petropouluos (1973); Gopalakrishnan and Khayyal (1991)	Theory is based on the arthmetic-geometric mean inequality	Optimization technique developed for solving a class of nonlinear optimization problem especially found in engineering design and manufacture
Dynamic programming	Agapiou (1992)	A collection of algorithms used to compute optimal policies given a perfect model of environment	Solving sequential or multi-stage decision problems by solving a series of single variable problems

Fuzzy logic	Kosko (1997); Klir 7 Yuan (1998)	Fuzzy interface engine & fuzzification-defuzzification module	Based on a machining model which works on human common-sense reasoning, decision-making and other concepts of human cognition
Genetic algorithm	Kuo (2002); Wang (2004)	A CGI (common gate-way interface) program	Based on a machining model developed from theoretical analysis, experimental database and numerical methods
Scatter search	Chen (2003)	A program designed by Laguna and Marti in C code	A generalized optimization methodology for machining problems that has no restrictive assumptions about objective function, parameter set and constraint set
Taguchi technique	Pignatiello (1993); Tsui (1999); Singh and Kumar (2003,2004,2005)	Design of experiments, Orthogonal arrays, ANOVA	Based on actual experimental work and determination of optimum conditions using statistical tools
Response surface methodology	Taraman (1974); Hassan and Suliman (1990); bardie (1993); Noordin (2004)	Design expert software (DX6)	Based on a machining model developed by mathematical and statistical techniques

CHAPTER 4

DESIGN OF EXPERIMENTS

4.1 INTRODUCTION

Design of experiments (DOE) is a statistical technique used to study multiple variables simultaneously. Sir R.A Fisher in England introduces DOE in the early 1920s. His primary goal was to determine the optimum water, rain, sunshine, fertilizer, and soil condition needed to produce the best crop. Using the DOE technique Fisher was to able to lay out all combination of the factors included in the experimental study. The conditions were created using a matrix, which allowed each factor an equal number of test conditions. Methods for analyzing the results of such experiments were also introduced. When the number of combinations possible became too large, schemes were devised to carry out a fraction of total possibilities such that all factors would be evenly present. Fisher devised the first methods that made it possible to analyze the effect of more than one factor at a time . DOE is highly effective wherever and whenever it is suspected that the performance of a part or process is controlled by more than one factor.

4.2 PURPOSE OF EXPERIMENTATION

The purpose of experimentation should be to understand how to reduce and control variation of a product or process; subsequently, decisions must be made concerning which parameters affect the performance of a product or process. The loss function quantifies the need to understand which design factors influence the average and variation of a performance characteristic of a product or process. By properly adjusting the average and reducing variation, the product process, the process or Product losses and minimized.

4.3 DESIGN OF EXPERIMENTS

A designed experiment is the simultaneous evaluation of two or more factors (parameters) for their ability to affect the resultant average or variability of particular product or process characteristics. It is important to note that this is an iterative process; the first round through the DOE process will many times lead to subsequent rounds of experimentation. The beginning round, often referred to as a screening experiment, is used to find the few important, influential factors out of the many possible factors involved with a product or process design. This experiment is typically a small experiment with many factors at two levels. Later rounds of experiments typically involve few factors at more than two levels to determine conditions of further improvement.

4.4 DESIGN OF EXPERIMENTS PROCESS

The DOE process is divided into three main phases, which encompass all experimentation approaches. The three phases are

(1) Planning phase,

(2) Conducting phase, and

(3) Analysis phase.

The planning phase is by far the most important phase for the experiment to provide the expected information. The planning phase is when factors and levels are selected and, therefore, is the most important stage of experimentation. Also, the correct selection of factors and levels is non-statistical in nature and is more dependent upon product and process expertise.

The second most important phase is the conducting phase, when test results are actually collected. If experiments are well planned and conducted, the analysis is actually much easier and more likely to yield positive information about factors and levels.

The analysis phase is when the positive or negative information concerning the selected factors and levels is generated based on the previous two phases. The analysis phase is least important in terms of whether the experiment will successfully yield positive results. This phase, however, is the most statistical in nature of the three phases of the DOE by a wide margin. Because of the heavier involvement of statistics, the analysis phase is typically the least understood by the product or process expert.

The major steps to complete an effective designed experiment are listed in the following text. The planning phase includes steps 1 through 9, the conducting phase is step 10, and the analysis phase includes steps 11 and 12.

1. State the problem(s) or area(s) of concern.
2. State the objective(s) of the experiment.
3. Select the Quality characteristic(s) and measurement system(s).
4. Select the factors that may influence the selected quality characteristics.
5. Identify limits of factors

6. Select levels for the factors.
7. Select the appropriate design
8. Select interactions that may influence the selected quality characteristics or go back to step 4 (iterative steps).
9. Assign factors to design and locate interactions.
10. Conduct tests described by trials in design.
11. Analyze and interpret results of the experimental trials.
12. Conduct confirmation experiment.

These steps are fundamentally the same regardless of whether one is designing a Taguchi-based experiment or a classical design. All designed experiments require that a certain number of combinations of factors and levels be tested to observe the results of those test conditions. Two or more passes through the process are often utilized; earlier rounds of experimentation provide a growth of knowledge and a basis for later rounds of experimentation.

4.5 RESPONSE SURFACE METHODOLOGY

Experimentation and making inferences are the twin features of general scientific methodology. Statistics as a scientific discipline is mainly designed to achieve these objectives. Planning of experiments is particularly very useful in deriving clear and accurate conclusions from the experimental observations, on the basis of which inferences can be made in the best possible manner. The methodology for making inferences has three main aspects. First, it establishes methods for drawing inferences from observations when these are not exact but subject to variation, because inferences are not exact but probabilistic in nature. Second, it specifies methods for collection of data appropriately, so that assumptions for the application of appropriate statistical methods to them are satisfied. The advantages of design of experiments are as follows:

- Number of trials is reduced
- Optimum values of parameters can be determined
- Assessment of experimental error can be made
- Qualitative estimation of parameters can be made
- Inference regarding the effect of parameters on the characteristics of the process can be made

4.6 REGRESSION EQUATION

A statistical technique used to explain or predict the behavior of a dependent variable. A regression equation takes the form of

$$Y=a+bx+c$$

where, Y - dependent variable

x - independent variable

c - regression residual

a & b – constants

4.7 OBJECTIVE FUNCTION

The purpose is to investigate the optimal cutting parameters for minimizing machining time of the turning operation while maintaining material removal rate. The main parameters in machining affecting machining time are cutting speed, feed and depth of cut. The optimal cutting parameters are subjected to an objective function of minimum machining time with the feasible range of cutting parameters.

4.8 CONSTRAINTS

4.8.1 Surface Roughness

Surface roughness has received serious attentions for many years. It has been an important design feature and quality measure in many situations such as parts subject to fatigue loads, precision fits, fastener holes and esthetic requirements. Furthermore, surface roughness in addition to tolerances imposes one of the most critical constraints for cutting parameter selection in manufacturing process planning.

4.8.2 Tool life

Cutting tool computer programs for analyzing general two-dimensional cutting tool geometries has been developed over a wide range of operating conditions. These programs can be used to predict chip shape and form, cutting forces, tool pressure distribution, and temperatures in the work piece, chip, and tool. This information can be used for further tool analysis, such as calculating tool wear rates, tool stresses, and a tool's chip breaking potential. Tool designers can use these programs to achieve optimal cutting efficiency through the design of proper cutting tool geometries and tool materials. Manufacturing engineers can use the programs to select the best cutting tool for a particular cutting operation. The ultimate result is improved cutting efficiency, work piece quality, and tool performance.

Predicting tool performance

Using the cutting tool programs, a more methodical approach can be taken for designing new high performance cutting tools than trial-and-error approaches used in the past. An engineer can explore the effects of tool geometry changes on cutting tool performance. Tools can be redesigned to achieve lower temperatures, higher cutting speeds, and reduced tool forces, thereby improving cutting efficiency and because a tool can be evaluated before it is fabricated, less reliance is placed on prototype building. Several important variables can be included in a simulation, including workpiece and tool thermal properties, cutting speed, feed rate or depth of cut, and frictional effects.

4.8.3 Cutting forces

There are three cutting forces which are acting on a single point tool and shown in figure1. The F_x is the feed force which is acting on the X direction, the F_y is cutting force acting of on Y direction and F_z is the radial force acting on the Z direction. The vibration will be more in the direction of cutting force F_y than that in the radial direction. Shintani et al with the increase of feed or depth of cut, vibration increase the tool wear when machining at feed of 0.125 mm/ rev. The cutting force F_y was low and almost equal to F_x and F_z and gives better results at higher cutting speed. During machining at 0.16 mm / rev feed force F_x was high and shows increasing trend. Ulvi Seker and Hasan Hasirci stated that cutting forces remained at about 20 % when machining austempered ductile irons and considerable improvement in surface quality.

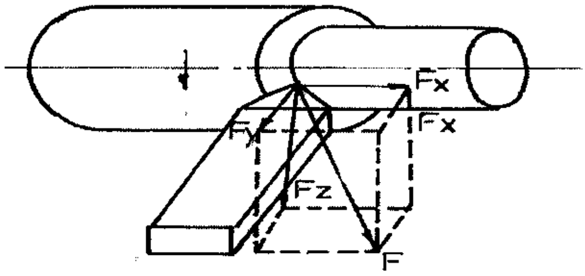

FIGURE 4.1 CUTTING FORCES ACTING ON A TOOL

4.9 CHOSEN INPUT PARAMETERS

The following input parameters were chosen for this study:

- Cutting speed
- Feed
- Depth of cut

4.9.1 Feeds ,Speeds & Depth of cut

For all metal-cutting processes, "speeds and feeds" are important parameters. The colloquial term "speeds and feeds" refers to the speed, feed, and depth of cut of a metal-cutting process. To describe these parameters, we will be using the turning process. The figure below shows the important geometry. The speed is the cutting speed, which is a measure of the part cut surface speed relative to the tool. Speed is a velocity unit, which is typically listed in terms of feet/min, inches/min, meters/second, or meters/min. Feed is the amount of material removed for each revolution or per pass of the tool over the workpiece. Feed is measured in units of length/revolution, length/pass, length/tooth, length/time, or other appropriate unit for the particular process. The depth of cut, DOC represents the third parameter for metal cutting. For turning, DOC is the depth that the tool is plunged into the surface. The DOC is half of the difference in the diameters D_a and D_b, the initial and final diameters, respectively.

4.10 CHOSEN OUTPUT PARAMETERS

The following output parameters were chosen for this study:

- Surface roughness
- Tool life
- Cutting force
- Power

4.11 EXPERIMENTAL DETAILS & SPECIFICATIONS

• Machine	- PSG CNC 110 lathe
• Work piece material	- Aluminum
• Tool material	- Insert carbide tip 4035

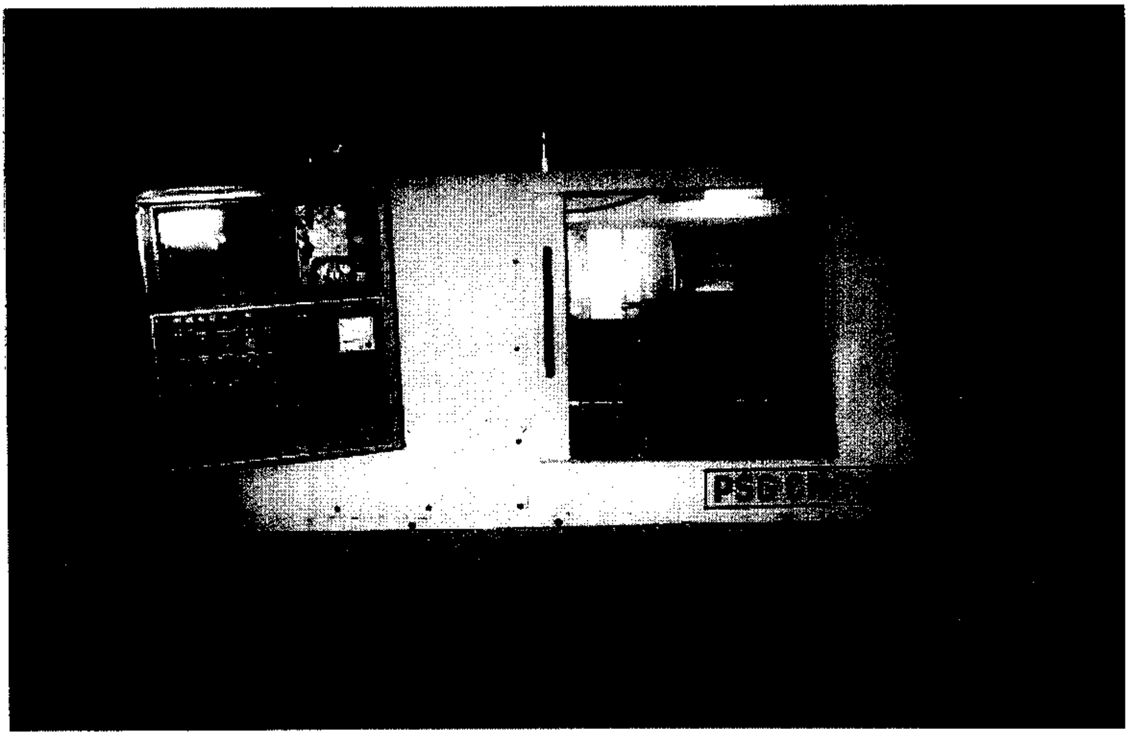

FIG 4.2 PSG CNC 110 LATHE

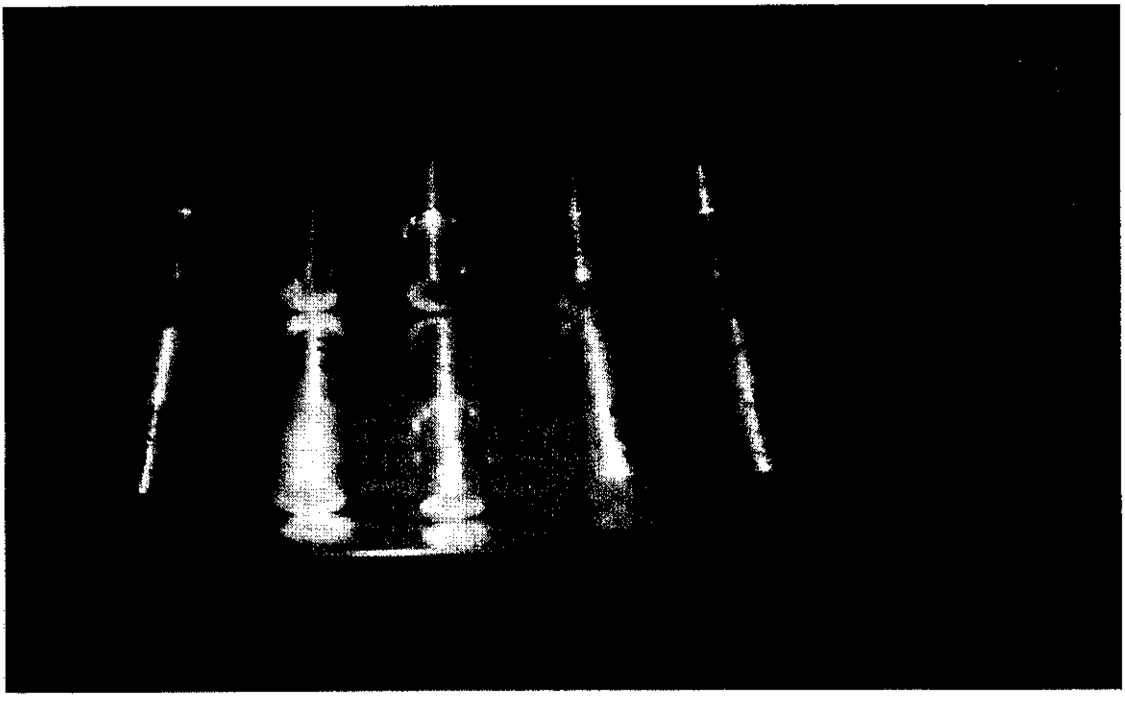

FIG 4.3 TURNED COMPONENTS

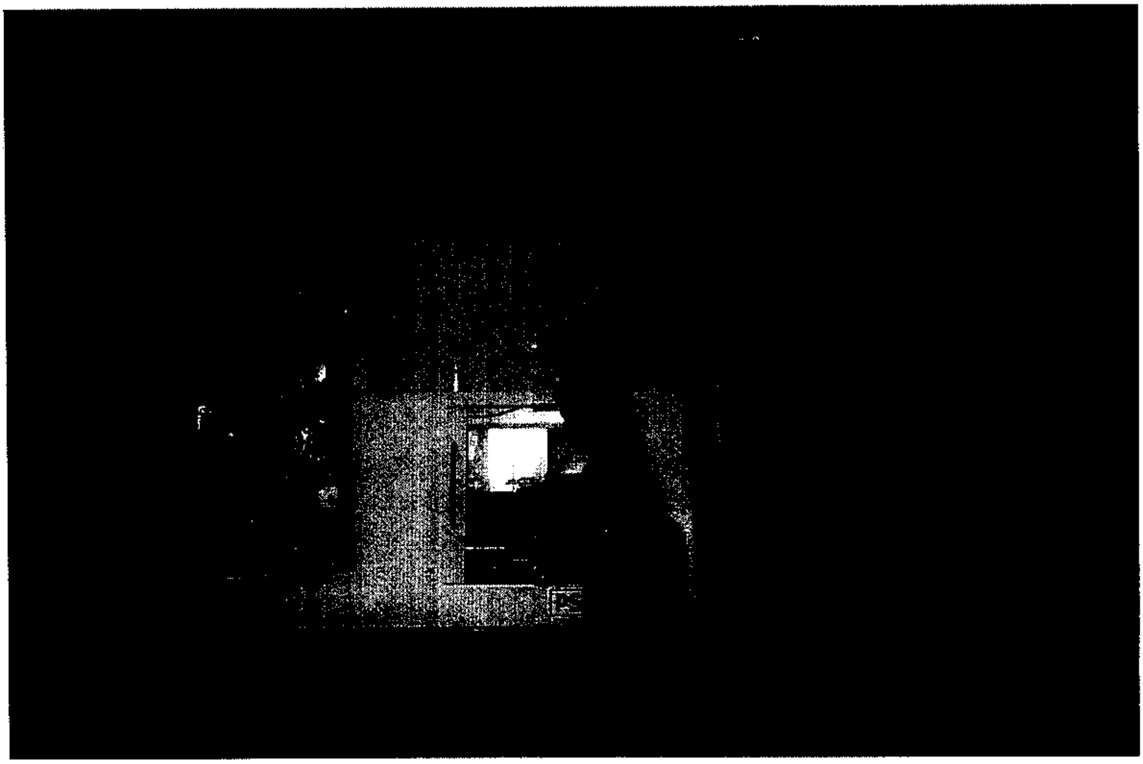

FIG 4.4 EXPERIMENTAL SETUP

CNC Lathe:

- Type of bed : Inclined 45° to vertical
- Swing over bed covers : 215 mm
- Maximum turning diameter of chucking job : 110 mm
- Maximum turning length (between centers) : 210 mm
- Maximum boring length : 90 mm

Spindle:

- Spindle nose : A2 – 3”
- Hole through spindle : 22 mm
- Spindle bore taper : MT4
- Spindle speed range infinitely variable : 40 – 3000 rpm

Fixed speeds – 4	720, 960, 1440&1920 rpm

Axes Travel:

Feed range x and z axis	1 – 3750 mm/min
Rapid traverse rate X and z axis	5 m/min
Threading pitch	0.25 – 16 mm
Cross slide stroke	80 mm

Tailstock:

Tailstock spindle diameter	50 mm
Tailstock spindle stroke	60 mm
Tailstock spindle taper	MT3

Turrent:

Number of tools	8(4 ID and 4 OD tools)
Turrent indexing positions	8
Turrent indexing accuracy	± 6 seconds
Turrent tool shank size	12 * 12 mm
Turrent actuation	electric
Maximum shank diameter of boring tool	20 mm
Indexing time	45° indexing:0.6 seconds 180° indexing:2.0 seconds

Power:

Power of spindle drive motor	2.2 KW continuous
Power of hydraulic motor	0.75 KW
Power of coolant pump motor	0.1 KW

Accuracies:

Positioning accuracy	± 0.010 mm
Repeatability	± 0.010 mm
Programming resolution	± 0.001 mm
Feedback resolution	± 0.001 mm

4.11.1 Aluminum

Aluminum is the world's most abundant metal and is the third most common element comprising 8% of the earth's crust. The versatility of aluminum makes it the most widely used metal after steel._Bauxite is converted to aluminum oxide (alumina) via the Bayer Process. The alumina is then converted to aluminum metal using electrolytic cells and the Hall-Heroult Process. Worldwide demand for aluminum is around 29 million tons per year. About 22 million tons is new aluminum and 7 million tons is recycled aluminum scrap. The use of recycled aluminum is economically and environmentally compelling. It takes 14,000 kWh to produce 1 tonne of new aluminum. Conversely it takes only 5% of this to remelt and recycle one tonne of aluminum. There is no difference in quality between virgin and recycled aluminum alloys. Pure aluminum is soft, ductile, corrosion resistant and has a high electrical conductivity. It is widely used for foil and conductor cables, but alloying with other elements is necessary to provide the higher strengths needed for other applications. Aluminum is one of the lightest engineering metals, having a strength to weight ratio superior to steel.By utilizing various combinations of its advantageous properties such as strength, lightness, corrosion resistance, recyclability and formability, aluminum is being employed in an ever-increasing number of applications. This array of products ranges from structural materials through to thin packaging foils.

4.12 EXPERIMENTAL WORK

The experiments were conducted using PSG CNC 110 Lathe. Aluminum test pieces of size 100 mm length and 35 mm diameter were turned using a tungsten carbide tip - 4035 tool.

4.13 EXPERIMENTAL DESIGN PROCEDURE

The experimental design procedure used for this study is briefly explained below.

4.13.1 Identification of Factors and Responses

The chosen factors were cutting speed (m/min), feed (mm/rev), and depth of cut (mm). The chosen response was surface roughness, cutting force, tool life and power.

4.13.2 Finding the limits of the process variables

The working ranges of all selected factors are fixed by conducting trial runs. This was carried out by varying one of the factors while keeping rest of them at constant values [6]. The upper limit of a factor was coded as +1.68 and the lower limit was coded as -1.68. The chosen levels of the selected process parameters with their units and notations are given in Table 4.1

TABLE 4.1. LIMITS OF PARAMETERS

PARAMETERS	UNIT	LIMITS				
		-1.68	-1	0	+1	+1.68
Cutting speed	m/min	75	87.5	162.5	237.5	400
Feed	mm/rev	0.025	0.0375	0.0625	0.0875	0.15
Depth of cut	mm	0.25	0.375	0.625	0.875	1.5

4.13.3 Development of design matrix

The design matrix chosen to conduct the experiment was a central composite rotatable design. The design matrix comprises a full replication of 2^3 (=8) factorial design plus 6 center points and 6 star points [7] which is shown in Table 4.2.

TABLE 4.2. DESIGN MATRIX AND THE OBSERVED VALUES OF MACHINING PARAMETERS

Trial No.	Cutting Speed (m/min)	Feed (mm/rev)	Depth of Cut (mm)	Surface Roughness (µm)	Cutting Force (N)	Tool Life (mins)	Power required for turning (KW)
1	-1	-1	-1	1.1	30.57	16138.0	0.037
2	1	-1	-1	0.76	22.97	509.800	0.101
3	-1	1	-1	1.62	66.48	8948.30	0.087
4	1	1	-1	1.28	49.97	282.670	0.237
5	-1	-1	1	1.25	77.63	10929.1	0.087
6	1	-1	1	0.91	58.35	345.250	0.237
7	-1	1	1	1.76	168.85	6059.90	0.204
8	1	1	1	1.43	126.60	191.430	0.554
9	-1.68	0	0	1.82	89.52	15241.6	0.089
10	1.68	0	0	1.02	55.46	46.5100	0.476
11	0	-1.68	0	0.86	30.97	1986.81	0.077
12	0	1.68	0	1.98	160.15	570.900	0.464
13	0	0	-1.68	1.17	26.19	1600.45	0.077
14	0	0	1.68	1.5	187.98	701.920	0.464
15	0	0	0	1.3	71.76	1050.00	0.193
16	0	0	0	1.3	71.76	1050.00	0.193
17	0	0	0	1.3	71.76	1050.00	0.193
18	0	0	0	1.3	71.76	1050.00	0.193
19	0	0	0	1.3	71.76	1050.00	0.193
20	0	0	0	1.3	71.76	1050.00	0.193

4.13.4 Conducting the experiments as per the design matrix

The experiments were conducted at the CAM Lab in Kumaraguru College of Technology, Coimbatore. In this work, twenty deposits were made using machining condition corresponding to each combination of parameters shown in Table 4.2 at random.

4.13.5 Recording the responses

The responses, surface roughness, tool life, cutting force and power were measured as shown in Table 4.2.

4.13.6 Development of a mathematical model

The response function representing any of the machining parameters can be expressed using the equation 4.1

$$Y = f(X_1, X_2, X_3) \qquad \ldots\ldots\ldots\ldots\ldots\ldots (4.1)$$

where

Y	=	Response or yield
X_1	=	Cutting Speed (v) in m/min
X_2	=	Feed (f) in mm/rev
X_3	=	Depth of cut (d) in mm

The second order response surface model for the four selected factors is given by the equation 4.2

$$Y = ß_0 + \sum_{i=1}^{4} ß_i X_i + \sum_{i=1}^{4} ß_{ii} X_i^2 + \sum_{i<j} ß_{ij} X_i X_j \qquad \ldots\ldots\ldots\ldots\ldots\ldots (4.2)$$

The second order response surface model [equation 4.3] could be expressed as follows

$$Y = ß_0 + ß_1 v + ß_2 f + ß_3 d + ß_{11} v^2 + ß_{22} f^2 + ß_{33} d^2 + ß_{12} vf + ß_{13} vd + ß_{23} fd \ldots\ldots\ldots (4.3)$$

Where $ß_0$ is the free term of the regression equation, the coefficients $ß_1$, $ß_2$ and $ß_3$ are linear terms, the coefficients $ß_{11}$, $ß_{22}$ and $ß_{33}$ are the quadratic terms, and the coefficients $ß_{12}$, $ß_{13}$ and $ß_{23}$ are the interaction terms. The coefficients were calculated using QA six sigma software (DOE-PCIV) and the same was verified using the software SYSTAT 7.0

which is shown in Table 4.3. After determining the coefficients, the mathematical model were developed and given below:

TABLE 4.3 CALCULATED REGRESSION COEFFICIENTS

Surface Roughness	Cutting Force	Tool Life	Power
1.304	72.353	985.788	0.196
-0.197	-10.470	-4857.10	0.100
0.290	32.203	-1086.00	0.093
0.084	39.081	-722.810	0.093
0.014	-3.5080	2743.38	0.013
0.014	4.6650	488.140	0.009
-0.016	8.7490	442.905	0.009
0.001	-3.9850	1459.74	0.036
0.001	-4.6780	980.193	0.036
-0.001	12.070	299.230	0.031

Surface Roughness:

$$Ra\ (\mu m) = 1.304 - 0.197v + 0.29f + 0.084d + 0.014v^2 + 0.014f^2 - 0.016d^2 + 0.001vd + 0.001fd - 0.001vf$$

Cutting Force :

$$Fc\ (N) = 72.353 - 10.47v + 32.203f + 39.081d - 3.508v^2 + 4.665f^2 + 8.749d^2 - 3.985vf - 4.678vd + 12.07fd$$

Tool life:

$$T\ (mins) = 985.788 - 4857.1v - 1086f - 722.81d + 2743.38v^2 + 488.14f^2 + 442.905d^2 + 1459.74vf + 980.193vd + 299.23fd$$

Power :

$$P\ (KW) = 0.196 + 0.1v + 0.093f + 0.093d + 0.013v^2 + 0.009f^2 + 0.009d^2 + 0.036vf + 0.036vd + 0.031fd$$

These mathematical models have been used to predict the values of the output parameters which is shown in Table 4.4.

TABLE 4.4 DESIGN MATRIX AND THE PREDICTED VALUES OF MACHINING PARAMETERS

Trial No.	Cutting Speed (m/min)	Feed (mm/rev)	Depth of Cut (mm)	Surface Roughness (μm)	Tool Wear (mm)	Cutting Force (N)	Tool Life (mins)	Power required for turning (KW)
1	-1	-1	-1	1.14	0.057	24.852	14065.3	0.044
2	1	-1	-1	0.74	0.087	21.238	528.000	0.100
3	-1	1	-1	1.72	0.097	73.088	8375.31	0.096
4	1	1	-1	1.32	0.127	53.534	1258.00	0.296
5	-1	-1	1	1.30	0.091	88.230	10060.8	0.096
6	1	-1	1	0.91	0.121	65.904	652.356	0.296
7	-1	1	1	1.88	0.119	184.74	5567.75	0.272
8	1	1	1	1.49	0.149	146.48	733.465	0.616
9	-1.68	0	0	1.67	0.089	80.041	16888.6	0.064
10	1.68	0	0	1.01	0.139	44.862	568.822	0.400
11	0	-1.68	0	0.85	0.083	31.418	4188.03	0.065
12	0	1.68	0	1.83	0.140	139.62	538.997	0.377
13	0	0	-1.68	1.11	0.065	31.390	3450.17	0.065
14	0	0	1.68	1.39	0.112	162.70	1021.51	0.377
15	0	0	0	1.30	0.092	72.353	985.788	0.196
16	0	0	0	1.30	0.092	72.353	985.788	0.196
17	0	0	0	1.30	0.092	72.353	985.788	0.196
18	0	0	0	1.30	0.092	72.353	985.788	0.196
19	0	0	0	1.30	0.092	72.353	985.788	0.196
20	0	0	0	1.30	0.092	72.353	985.788	0.196

4.13.7 Checking the adequacy of the developed models

The adequacies of the developed models were tested using the analysis of variance (ANOVA) technique. As per this technique if the calculated F-ratio values for the developed models do not exceed the standard tabulated values for a desired level of confidence (95%) and the calculated R-ratio values of the developed model exceed the standard tabulated values for a desired level of confidence (95%), then the models are said to be adequate within the confidence limit. The conditions were satisfied for the developed model.

4.13.8 Conducting the conformity test

Confirmation tests were conducted in the same experimental setup to confirm the results of the experiment and demonstrate the reliability of the predicted values. The conformity tests show the accuracy of the models developed, which is above 95%.

CHAPTER 5

PARTICLE SWARM OPTIMIZATION

5.1 INTRODUCTION

Dr. Russell C. Eberhart first introduced particle Swarm Optimization and Dr.James Kennedy in 1995. As described by Eberhart and Kennedy, the PSO algorithm is an adaptive algorithm based on a social-psychological metaphor; a population of individuals (referred to as particles) adapts by returning stochastically toward previously successful regions. Particle Swarm has two primary operators:

Velocity update

Position update

During each generation each particle is accelerated toward the particles previous best position and the global best position. At each iteration a new velocity value for each particle is calculated based on its current velocity, the distance from its previous best position, and the distance from the global best position. The new velocity value is then used to calculate the next position of the particle in the search space. This process is then iterated a set number of times or until a minimum error is achieved.

5.2 BACKGROUND OF ARTIFICIAL LIFE

The term "Artificial Life" (Alive) is used to describe research into human-made systems that possess some of the essential properties of life. Alive includes two-folded research topic:

i. Alive studies how computational techniques can help when studying biological phenomena.

ii. ALife studies how biological techniques can help out with computational problems

The focus of particles swarm optimization is on second life. Actually, there are already lots of computational techniques inspired by biological systems. For example, artificial neural network is a simplified model of human brain; genetic algorithm is inspired by the human evolution.

Another type of biological system - social system, more specifically, the collective behaviors of simple individuals interacting with their environment and each other. Someone called it as swarm intelligence. There are two popular swarm inspired methods in computational intelligence areas: Ant colony optimization and particle swarm optimization. ACO was inspired by the behaviors of ants and has many successful applications in discrete optimization problems.

5.2.1 Bird Flocking

As stated before, PSO simulates the behaviors of bird flocking, suppose the following scenario: a group of birds are randomly searching food in an area. There is only one piece of food in the area being searched. All the birds do not know where the food is. But they know how far the food is in each iteration, so what's the best strategy to find the food? The effective one is to follow the bird, which is nearest to the food.

In PSO, each single solution is a "bird" in the search space. We call it "particle". All of particles have fitness values, which are evaluated by the fitness function to be optimized, and have velocities, which direct the flying of the particles. The particles are "flown" through the problem space by following the current optimum particles by Murthy et al. (2003).

5.3 PARTICLE SWARM OPTIMIZATION TECHNIQUE

The particle swarm concept originated as a simulation of simplified social system. The original intent was to graphically simulate the choreography of bird of a bird block or fish school. However, it was found that particle swarm model can be used as an optimizer.

As stated before, PSO simulates the behaviors of bird flocking. Suppose the following scenario: a group of birds are randomly searching food in an area. There is only one piece of food in the area being searched. All the birds do not know where the food is. But they know how far the food is in each iteration. So what's the best strategy to find the food? The effective one is to follow the bird which is nearest to the food.

PSO learned from the scenario and used it to solve the optimization problems. In PSO, each single solution is a "bird" in the search space. We call it "particle". All of particles have fitness values which are evaluated by the fitness function to be optimized, and have velocities which direct the flying of the particles. The particles fly through the problem space by following the current optimum particles.

PSO is initialized with a group of random particles (solutions) and then searches for optima by updating generations. In every iteration, each particle is updated by following two "best" values. The first one is the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called pbest. Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the population. This best value is a global best and called gbest. When a particle takes part of the population as its topological neighbors, the best value is a local best and is called lbest.

5.4 ALGORITHM OF PARTICLES SWARM OPTIMIZATION

Most of evolutionary techniques have the following procedure:

i. Random generation of an initial population

ii. Reckoning of a fitness value for each subject. It will directly depend on the distance to the optimum.

iii. Reproduction of the population based on fitness values.

iv. If requirements are met, then stop. Otherwise go back to 2.

From the procedure, we can learn that PSO shares many common points with GA. Both algorithms start with a group of a randomly generated population, both have fitness values to evaluate the population. Both update the population and search for the optimum with random techniques. Both systems do not guarantee success.

However, PSO does not have genetic operators like crossover and mutation. Particles update themselves with the internal velocity. They also have memory, which is important to the algorithm.

5.5 PSO PARAMETERS CONTROL

There are two key steps when applying PSO to optimization problems: the representation of the solution and the fitness function. One of the advantages of PSO is that PSO take real numbers as particles. It is not like GA, which needs to change to binary encoding, or special genetic operators have to be used. Then we can use the standard procedure to find the optimum. The searching is a repeat process, and the stop criteria are that the maximum iteration number is reached or the minimum error condition is satisfied. There are not many parameter need to be tuned in PSO. Here is a list of the parameters and their typical values.

The number of particles:

The typical range is 20 - 40. Actually for most of the problems 10 particles is large enough to get good results. For some difficult or special problems, one can try 100 or 200 particles as well.

Dimension of particles:

It is determined by the problem to be optimized.

Range of particles:

It is also determined by the problem to be optimized, you can specify different ranges for different dimension of particles.

Vmax:

It determines the maximum change one particle can take during one iteration. Usually we set the range of the particle as the Vmax.

Learning factors:

C1 and C2 usually equal to 2. However, other settings were also used in different literatures. But usually C1 equals to C2 and ranges from [0, 4]

The stop condition:

The maximum number of iterations the PSO execute and the minimum error requirement. Global version vs. local version: we introduced two versions of PSO. global and local version. global version is faster but might converge to local optimum for some problems. local version is a little bit slower but not easy to be trapped into local optimum. One can use global version to get quick result and use local version to refine the search.

5.6. PARTICLE SWARM ALGORITHM

Step 1:

The random particles (X) and velocity vector (V) are initialized and the optima for updating generations are searched.

Step 2:

For each particle, the fitness is evaluated. If the fitness value is better than the best fitness value (pbest), then current value new pbest is set as shown in fig.5.1.

Step 3:

The particle with the best fitness value of all the particles, the global best gbest is chosen as shown in fig.5.1

Step 4:

For each particle, the particle velocity and particle position are calculated by the equations,

$$v[] = \omega * v[] + C1 * rand() * (pbest[] - present[]) + C2 * rand * (gbest[] - present[]) \qquad \ldots\ldots\ldots\ldots\ldots(5.1)$$

$$present[] = present[] + v[] \qquad \ldots\ldots\ldots\ldots\ldots\ldots(5.2)$$

where, v[ ] = Particle velocity

persent[ ] = Current particle (solution)

pbest[ ] = Best solution among the each particle

gbest[ ] = Best among defined as stated before.

rand ( ) = Random numbers between (0,1)

ω = Inertia Weights. Usually 0.8 or 0.9

C_1, C_2 are learning factors. Usually $C_1 = C_2 = 2$.

Step 5:

If the sum of the accelerations would cause the velocity on that dimension to exceed Vmax, which is the parameter specified by the user, then the velocity on that dimensions is limited to Vmax.

Step 6:

Termination criteria are maximum number of iterations or minimum error conditions.

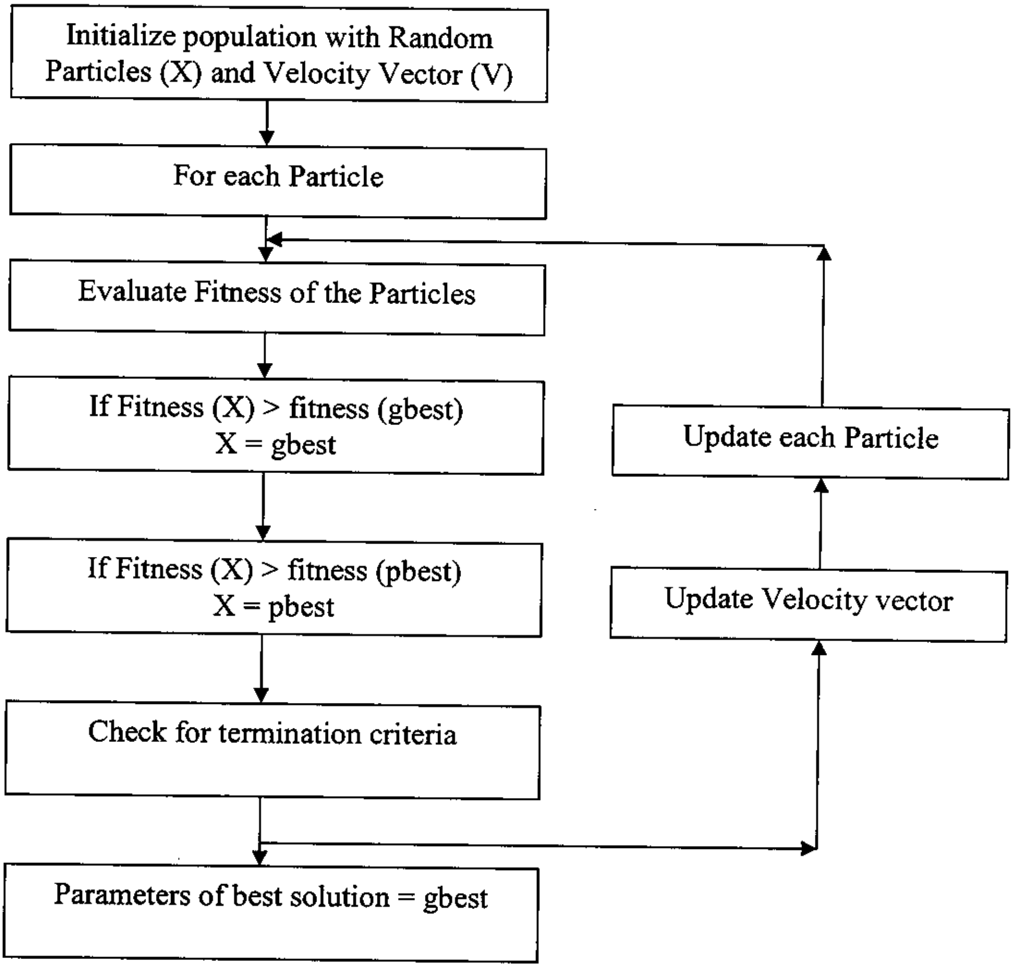
Initialize population with Random Particles (X) and Velocity Vector (V)
For each Particle
Evaluate Fitness of the Particles
If Fitness (X) > fitness (gbest) X = gbest
If Fitness (X) > fitness (pbest) X = pbest
Check for termination criteria
Parameters of best solution = gbest
Update each Particle
Update Velocity vector

FIG 5.1 FLOW CHART OF PSO ALGORITHM

5.7 OBJECTIVE FUNCTION

The optimization problem for the turning operations can be formulated as a multi-objective, multi-variable, and nonlinear optimization problem with multi-constraints. In order to overcome the large differences in numerical values between sub objectives, each sub objective is normalized. The following resultant weighted objective function is to be minimized:

$COF=w1\ (Ra/Ra_{min}) + w2\ (Fc/Fc_{min}) + w3\ (P/P_{min}) - w4\ (T/T_{max})$ ……(5.3)

Where:

COF = Value of combined objective function

Ra = Surface roughness

Fc = Cutting force

P = Power

T = Tool life

Ra_{min} = Target value for the surface roughness

Fc_{min} = Target value for the cutting force

P_{min} = Target value for power

T_{max} = Target value for tool life

w1to w4 – Weights that give different status (importance) to each response.

5.8 FORMULAE USED

PSO is initialized with a group of random particles (solutions) and then searches for optima by updating generations. In every iteration, each particle is updated by two best values. The first one is the best solution (fitness) it has achieved so far. The value is called Pbest. Another best value obtained so far by any particle in the population. This best value is called Gbest After finding the two best values, the particle updates its velocity and position with the following equations.

V1=V1+C1*rand*(Pbest- present)+C2*rand*(Gbest-present)…………………………(5.4)

Present1= present + V1………………………………………………………(5.5)

V1 = particle velocity,

Present = current particle (solution),

Pbest and Gbest = defined as stated before.

Rand = random number between (0-1),

C1,C2 = learning factors. Usually C1=C2=2.

5.9 RESULTS OF PSO

The software for the optimal allocation of total stock, minimization of total production cost using PSO has been implemented in C language. The present work is an optimization problem with constraints. The objective function is the total production cost, which consists of different passes. In the present work involving turning, the optimum results could be obtained with the population size of 10 and 500 generations. The optimum cost (Rs 46.194) is obtained .The sample results obtained using PSO are presented in the following Table 5.1.

TABLE 5.1 SAMPLE RESULTS OF PSO

Cutting speed (m/min)	Feed (mm/rev)	Depth of cut (mm)	Cost (Rs)
350	0.117	1.685	45.947433
227	0.106	1.631	46.412426
318	0.093	1.415	46.224903

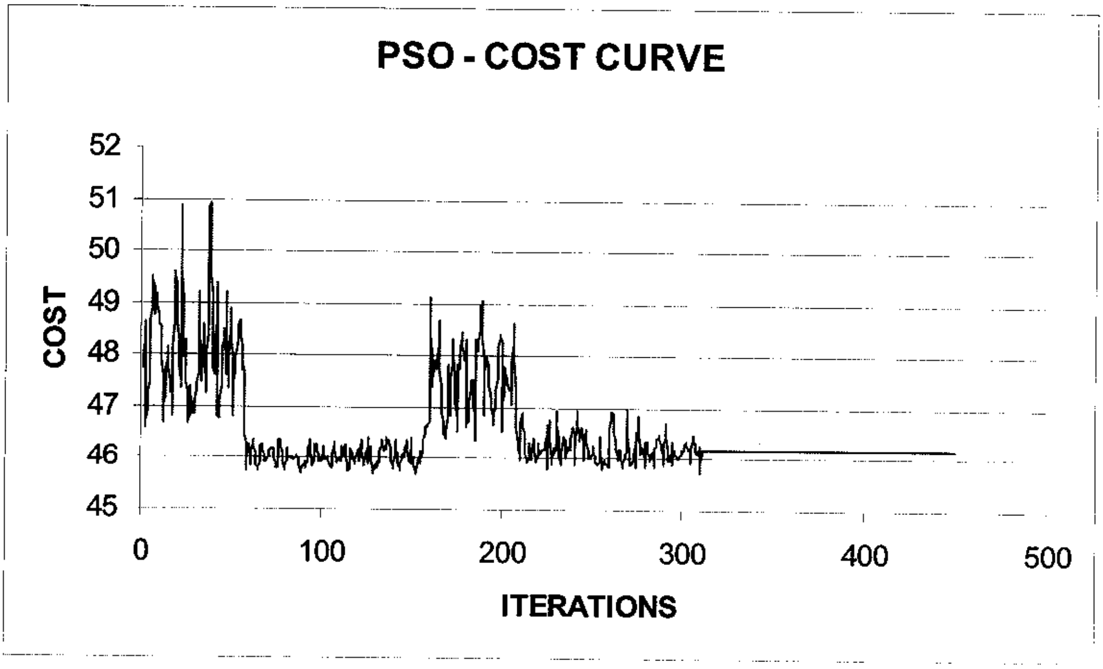

FIG.5.2. COST CURVE

From fig.5.2, the cost is continuously decreased up to the optimum value

CHAPTER 6

ANT COLONY OPTIMIZATION

6.1 INTRODUCTION

Ant Colony Optimization (ACO) studies artificial systems that take inspiration from the behavior of real ant colonies and which are used to solve discrete optimization problems. Ant Colony Optimization is a class of algorithms, whose first member, called Ant System, was initially proposed by Colorni, Dorigo and Maniezzo The main underlying idea, loosely inspired by the behavior of real ants, is that of a parallel search over several constructive computational threads based on local problem data and on a dynamic memory structure containing information on the quality of previously obtained result. The collective behavior emerging from the interaction of the different search threads has proved effective in solving combinatorial optimization (CO) problems.

Ant colony optimization is a kind of non traditional optimization technique in which the main idea underlying is that of a parallelizing search over several constructive computational threads , all based on a dynamic memory structure incorporating information on the effectiveness of previously obtained results and in which the behavior of each single agent is inspired by the behavior of real ants. Researchers are also fascinated by seeing the ability of the almost blind ants to establish the shortest route from their nests to the food source and back. These ants secrete a substance called "pheromone" and use its trials as a medium for communicating information among each other .the probability of the trial being followed by other ants is reinforced by increased trial deposition of others following this trial.

This cooperative search behavior of real ants inspired the new computational paradigm for optimizing real life systems and it is suited for solving large scale optimization problem. ACO has also been applied to other optimization problems like the quadratic assignment problem. More recently, a modified ACO was presented as an effective global optimization procedure by introducing bi-level search procedure called local and global search. The important aspect in ACO is that the artificial ants select the solution. They move with the selection probability proportional to the pheromone trial.

A set of computational concurrent and asynchronous agents (a colony of ants) moves through states of the problem corresponding to partial solutions of the problem to solve. They move by applying a stochastic local decision policy based on two parameters, called *trails* and *attractiveness*. By moving, each ant incrementally constructs a solution to the problem.

When an ant completes a solution, or during the construction phase, the ant evaluates the solution and modifies the trail value on the components used in its solution. This pheromone information will direct the search of the future ants.

Furthermore, an ACO algorithm includes two more mechanisms:

Trail evaporation:

Trail evaporation decreases all trail values over time, in order to avoid unlimited accumulation of trails over some component.

Daemon actions:

Daemon actions can be used to implement centralized actions that cannot be performed by single ants, such as the invocation of a local optimization procedure, or the update of global information to be used to decide whether to bias the search process from a non-local perspective.

6.2 ANT COLONY ALGORITHM

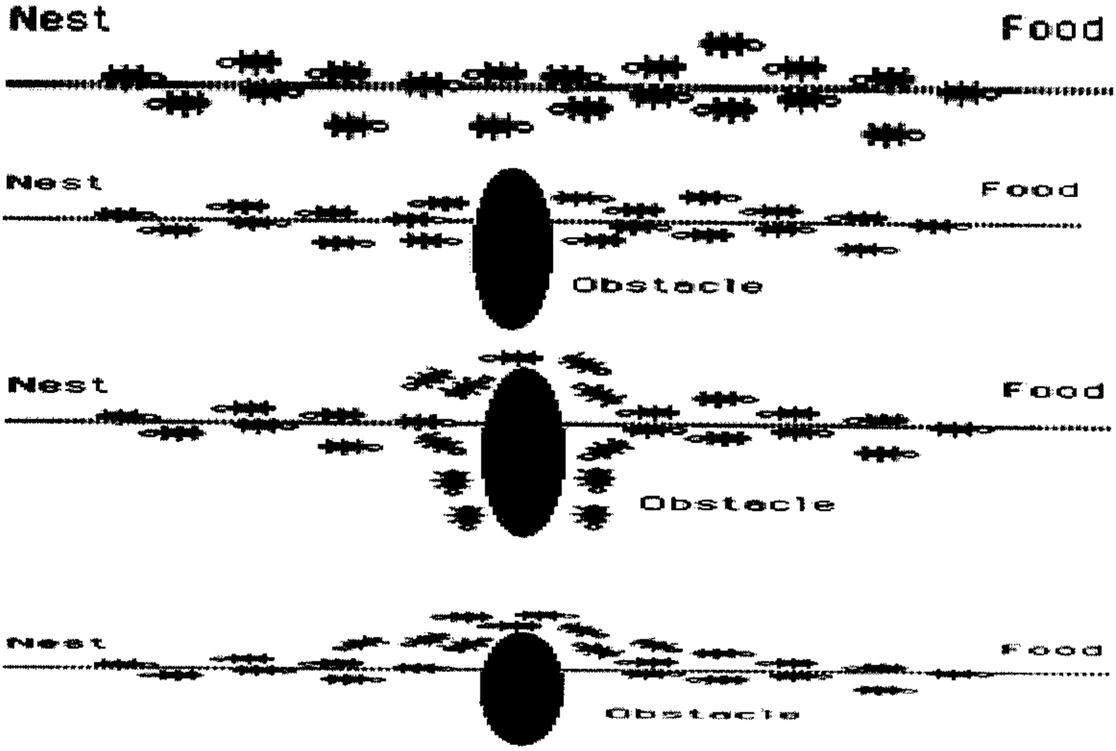

FIG.6.1 CONCEPT OF ANT COLONY ALGORITHM

Consider the following figure in which ants are moving on a straight line which connects a food source to the nest. It is well-known that the main means used by ants to form and maintain the line is a pheromone trail. Ants deposit a certain amount of pheromone while walking, and each ant probabilistically prefers to follow a direction rich in pheromone rather than a poorer one. This elementary behavior of real ants can be used to explain how they can find the shortest path which reconnects a broken line after the sudden appearance of an unexpected obstacle has interrupted the initial path as shown in fig.6.1. In fact, once the obstacle has appeared, those ants which are just in front of the obstacle cannot continue to follow the pheromone trail and therefore they have to choose between turning right or left. In this situation we can expect half the ants to choose to turn right and the other half to turn left. The very same situation can be found on the other side of the obstacle. It is interesting to note that those ants which choose, by chance, the shorter path around the obstacle will more rapidly reconstitute the interrupted pheromone trail compared to those which choose the longer path. Hence, the shorter path will receive a higher amount of pheromone in the time unit and this will in turn cause a higher number of ants to choose the shorter path. Due to this positive feedback (autocatalytic) process, very soon all the ants will choose the shorter path. The most interesting aspect of this autocatalytic process is that finding the shortest path around the obstacle seems to be an emergent property of the interaction between the obstacle shape and ants distributed behavior: Although all ants move at approximately the same speed and deposit a pheromone trail at approximately the same rate, it is a fact that it takes longer to contour obstacles on their longer side than on their shorter side which makes the pheromone trail accumulate quicker on the shorter side. It is the ants preference for higher pheromone trail levels which makes this accumulation still quicker on the shorter path.

Ant Colony Algorithms are typically use to solve minimum cost problems. There are two working modes for the ants: either forwards or backwards. Pheromones are only deposited in backward mode. The ants memory allows them to retrace the path it has followed while searching for the destination node as shown in fig.6.1. Before moving backward on their memorized path, they eliminate any loops from it. While moving backwards, the ants leave pheromones on the arcs they traversed. The ants evaluate the cost of the paths they have traversed. The shorter paths will receive a greater deposit of pheromones. An evaporation rule will be tied with the pheromones, which will reduce the chance for poor quality solutions.

The nature metaphor on which ant algorithms are based is that of ant colonies. One of the problems which confronted scientists was to understand how almost blind ants could establish the shortest route from their nests to the food source and back. Their results revealed that these ants used their pheromone trails as a medium for communicating among themselves. A moving ant lays various quantities of pheromone on its route. To apply the ant colony optimization methodology for continuous function optimization problems, the domain has to be divided first into a specific number of regions. The fitness of the regions are first evaluated and stored in descending fitness. A total of "*A*" number of ants explore these regions and updating of the region is done locally and globally with the local search mechanism respectively. Thus these ants are divided into "*G*" global ants and "*L*" local ants. In the local search, the local ants have the capability of selecting regions proportional to the current pheromone values of these regions. The age of the region is another important parameter in the CACO algorithm. The size of the ant movement in the local search depends on the current age. The search radius is maximum for zero age and minimum for maximum age, with the variation in radius set to follow a linear relationship. The global search creates "*G*" new regions by replacing the weaker portions of the existing one. The natural metaphor on which ant algorithms are based is that of ant colonies. Researchers are fascinated by seeing the ability of the almost blind ants to establish the shortest route from their nests to the food source and back. These ants secrete a substance called 'pheromone' and use its trails as a medium for communicating information among each other. The probability of the trail being followed by other ants is enhanced by increased trail deposition of others following this trail. This co-operative search behavior of real ants inspired the new computational paradigm for optimizing real life systems and it is suited to solving large-scale optimization problems. ACO has also been applied to other optimization problem like the traveling salesman problem, scheduling etc. More recently, the modified ACO is made as effective global optimization procedure by introducing bi-level search procedures called local and global search. The important aspect in ACO is that the artificial ants select the solution to which they move with a selection probability proportional to the pheromone trial.

Ants

ANTS is an extension of the some under defined elements of the general algorithm, such as the attractiveness function to use or the initialization of the trail distribution. This turns out to be a variation of the general ACO framework that makes the resulting algorithm similar in structure to tree search algorithms. In fact, the essential trait which distinguishes ANTS from a tree search algorithm is the lack of a complete backtracking mechanism, which is substituted by a probabilistic (*Non-deterministic*) choice of the state to move into and by an incomplete (*Approximate*) exploration of the search tree: this is the rationale behind the name ANTS, which is an acronym of *Approximated Nondeterministic Tree Search*. In the following, we will outline two distinctive elements of the ANTS algorithm within the ACO framework, namely the attractiveness function and the trail updating mechanism.

Attractiveness

The attractiveness of a move can be effectively estimated by means of lower bounds (upper bounds in the case of maximization problems) on the cost of the completion of a partial solution. In fact, if a state corresponds to a partial problem solution it is possible to compute a lower bound on the cost of a complete solution containing i. Therefore, for each feasible move i,y, it is possible to compute the lower bound on the cost of a complete solution containing y: the lower the bound the better the move. Since a large part of research in ACO is devoted to the identification of tight lower bounds for the different problems of interest, goodlower bounds are usually available.When the bound value becomes greater than the current upper bound, it is obvious that the considered move leads to a partial solution which cannot be completed into a solution better than the current best one. The move can therefore be discarded from further analysis. A further advantage of lower bounds is that in many cases the values of the decision variables, as appearing in the bound solution, can be used as an indication of whether each variable will appear in good solutions. This provides an effective way of initializing the trail values.

Trail update

A good trail updating mechanism avoids stagnation, the undesirable situation in which all ants repeatedly construct the same solutions making any further exploration in the search process impossible. Stagnation derives from an excessive trail level on the moves of one solution, and can be observed in advanced phases of the search process, if parameters are not well tuned to the problem. The trail updating procedure evaluates each solution against the last *k* solutions globally constructed by ANTS. One of the most difficult aspects to be considered in met heuristic algorithms is the trade-off between exploration and exploitation. To obtain good results, an agent should prefer actions that it has tried in the past and found to be effective in producing desirable solutions (exploitation); but to discover them, it has to try actions not previously selected (exploration). Neither exploration nor exploitation can be pursued exclusively without failing in the task: for this reason, the ANTS algorithm integrates the stagnation avoidance procedure to facilitate exploration with the probability definition mechanism based on attractiveness and trails to determine the desirability of moves.

Pheromone

In ACS once all ants have computed their tour (i.e. at the end of each iteration) As updates the pheromone trail using all the solutions produced by the ant colony. Each edge belonging to one of the computed solutions is modified by an amount of pheromone proportional to its solution value. At the end of this phase the pheromone of the entire system evaporates and the process of construction and update is iterated. On the contrary, in ACS only the best solution computed since the beginning of the computation is used to *globally update* the pheromone. As was the case in AS, global updating is intended to increase the attractiveness of promising route but ACS mechanism is more effective since it avoids long convergence time by directly concentrate the search in a neighborhood of the best tour found up to the current iteration of the algorithm. In ACS, the final evaporation phase is substituted by a *local updating* of the pheromone applied during the construction phase. Each time an ant moves from the current city to the next the pheromone associated to the edge is modified in the following way: ( ) *tij t* = *r* ×*t ij* (*t* - *1*)+(1- *r*) ×*t* 0 where 0 £ r £ 1 is a parameter (usually set at 0.9) and *t*0 is the initial pheromone value. *t*0 is defined as *t*0=(*n*·L*nn*)-1, where L*nn* is the tour length produced by the execution of one ACS iteration without the pheromone component (this is equivalent to a probabilistic nearest neighbor heuristic). The effect of local-updating is to make the desirability of edges change dynamically: every time an ant uses an edge this becomes slightly less desirable and only for the edges which never belonged to a global best tour the pheromone remains *t*0. An interesting property of these local and global updating mechanisms is that the pheromone *tij*(*t*) of each edge is inferior limited by *t*0.

6.2.1 Schemes of the ant colony algorithm

The 200 solutions are then sorted in ascending order with respect to the objective function. The regions pertaining to minimum production cost are referred to as superior solutions, while regions pertaining to the maximum production cost are referred to as inferior solutions. A typical representation of the superior and inferior solutions is shown in fig.6.2.

← N →

1	**2**	**3**	....	**120**	**121**	....	**190**	**200**

Superior solutions (min. cost) **inferior solutions (max. cost)**

FIG.6.2 REPRESENTATION OF SUPERIOR SOLUTIONS AND INFERIOR SOLUTIONS

6.2.2. Distribution of ants

The total numbers of ants, *A*, is 100, which is half of *N* and is distributed as 80 for global (*G*) and 20 for local search (*L*)

6.2.3 Global search

Using global search, global ants create G new regions by replacing the inferior solutions of the existing solutions. It consists of the following operations:

- crossover
- mutation
- trail diffusion.

6.2.3.1 Crossover

Parents are selected from superior solutions (*N* -*G*) (i.e., minimum cost region) and randomly selected from 90% of the inferior solutions for crossover. Then, the randomly selected n numbers are replaced by corresponding chosen parents, if the probability is equal to or greater than the crossover probability (*Cp*) 0.75. A representation of *N* -*G* parents (superior solutions) and *G* children (inferior solutions) is shown in Fig.6.3.

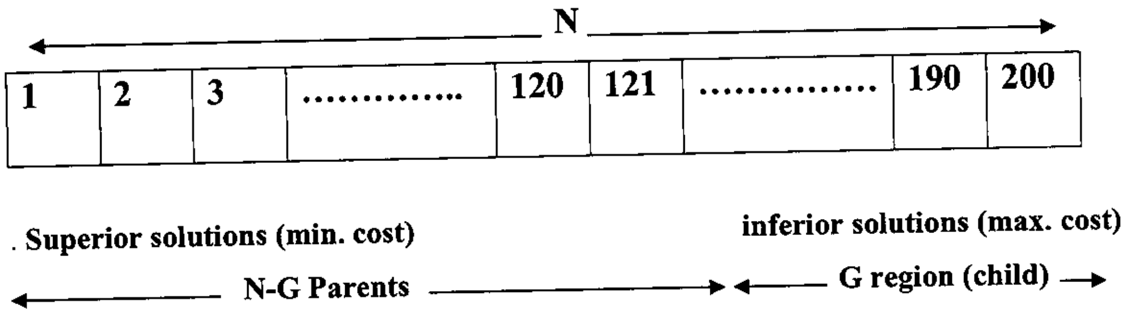

FIG. 6.3 A REPRESENTATION OF *N* -*G* PARENTS & *G* CHILD

The 90% of the solutions (randomly chosen) in the inferior solutions are replaced with randomly selected solutions from the superior solutions. The distribution of ants, as well as the selection of solutions is illustrated in fig.6.4.

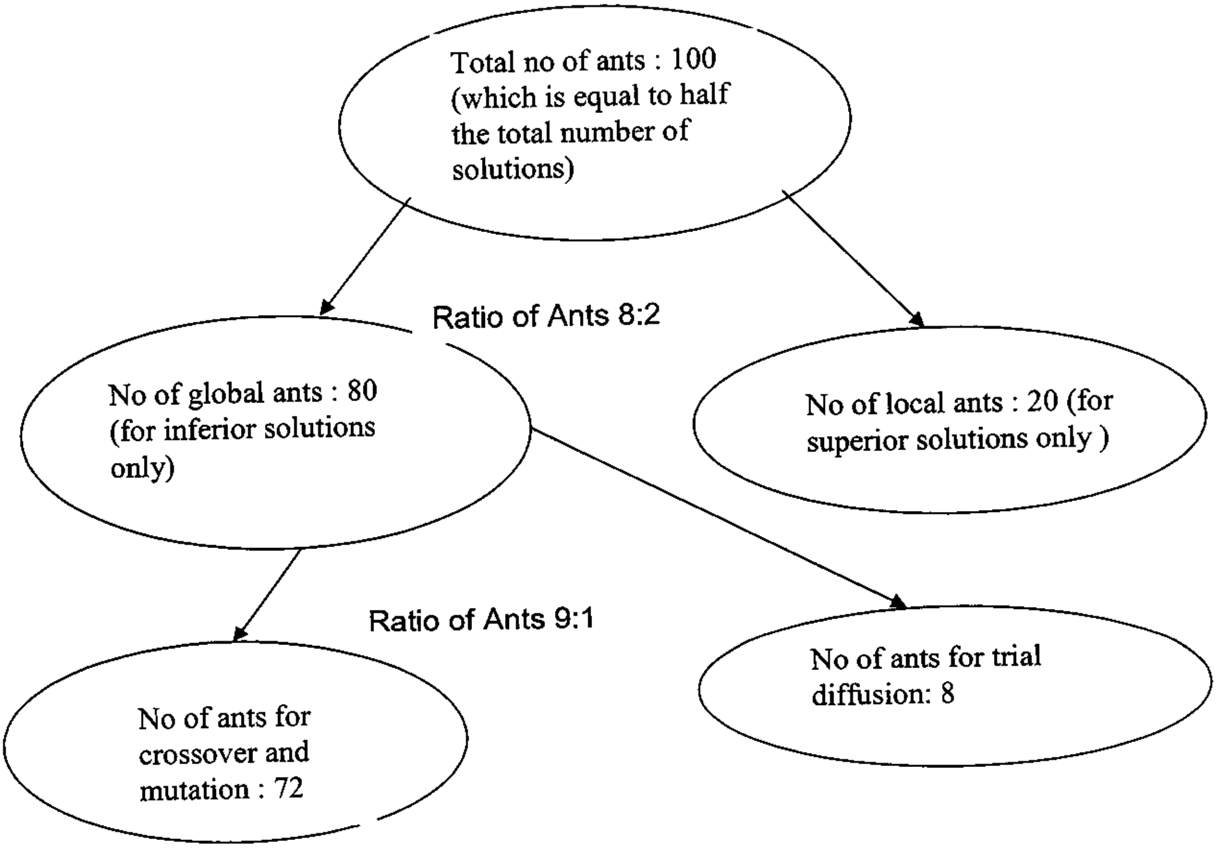
Total no of ants : 100 (which is equal to half the total number of solutions)
Ratio of Ants 8:2
No of global ants : 80 (for inferior solutions only)
No of local ants : 20 (for superior solutions only)
Ratio of Ants 9:1
No of ants for crossover and mutation : 72
No of ants for trial diffusion: 8

FIG. 6.4 DISTRIBUTION OF ANTS FOR LOCAL AND GLOBAL SEARCH

6.2.3.2 Mutation

After the random walk step, randomly adding or subtracting a value to each variable of the newly created solutions in the inferior region with a probability equal to a suitably defined mutation probability. The mutation size is reduced as per the relation

$$\Delta(T, R) = R(1 - r(1 - T)b) \qquad \text{.............. (6.1)}$$

where, r = random number from [0, 1],

R = the maximum step size,

T = the ratio of the current iteration number to that of the total number of iterations, and

b = positive parameter controlling the degree of nonlinearity. The value of b considered in this work is 10, which is arrived at by a trial basis.

6.2.3.3 Trial diffusion

Trail diffusion, which is another element in global search, is applied on the inferior solutions that were not considered during the random walk and mutation stages. Here, two parents are selected at random from the present parent superior solutions. The variables of the child's position vector can have either

1. the value of the corresponding variable from the first parent,
2. the corresponding value of the variable from the second parent,or
3. a combination arrived at from a weighted average of the above:

$$x(child) = (a).xi\ (parent1) + (1 - a).xi\ (parent2) \qquad \text{...........(6.2)}$$

where , a is a uniform random number in the range [0, 1].

The probability of selecting the third option is set equal to the mutation probability while allotting equal probability of selecting the first two steps. The trail value of the newly created child solutions is assigned a trail value lying between the values of the original parent solutions.

6.2.4 Local search

In the ACO algorithm, local (artificial) ants select a region i with a probability.

$$P_i(t) = \frac{t_i(t)}{\sum_i t_k(t)}$$

where, i is the region index and

$\tau_i\ (k)$ is the pheromone trail on region i at time t.

After selecting the region the ant moves through a short distance (finite random increment). The direction of movement is retained if the fitness value improvement is observed, otherwise it is reverted. Correspondingly the solution's position vector is updated and the pheromone trail value is improved based on the fitness value. The variables of this problem are velocity, feed rate and depth of cut, which can have any continuous value, subject to the limits available. In the continuous algorithm, the pheromone values are decreased after each iteration by:

$$\tau i\ (t+1) = \rho . \tau i\ (t) \qquad \ldots\ldots\ldots\ldots\ldots\ldots (6.3\ )$$

where , ρ is the evaporation rate which is assumed to be 0.2 on a trial basis and

$\tau_i\ (t)$ is the trail associated with solution at time t.

6.2.5. Algorithm

Step 1:

Initialization phase

The initial pheromone for each edge is set.

Step 2:

```
Repeat
    For each ant a starting node is selected randomly.
        Repeat
            The next node is moved according to the node
            transition rule, until a tour is completed.
    Using the pheromone updating rule, update the
    pheromone intensity for each edge, until the stopping
    criterion is satisfied.
```

Step 3:

Output the global best tour

The best tour reported by this algorithm is the attempt to

optimize the problem objective under a certain condition.

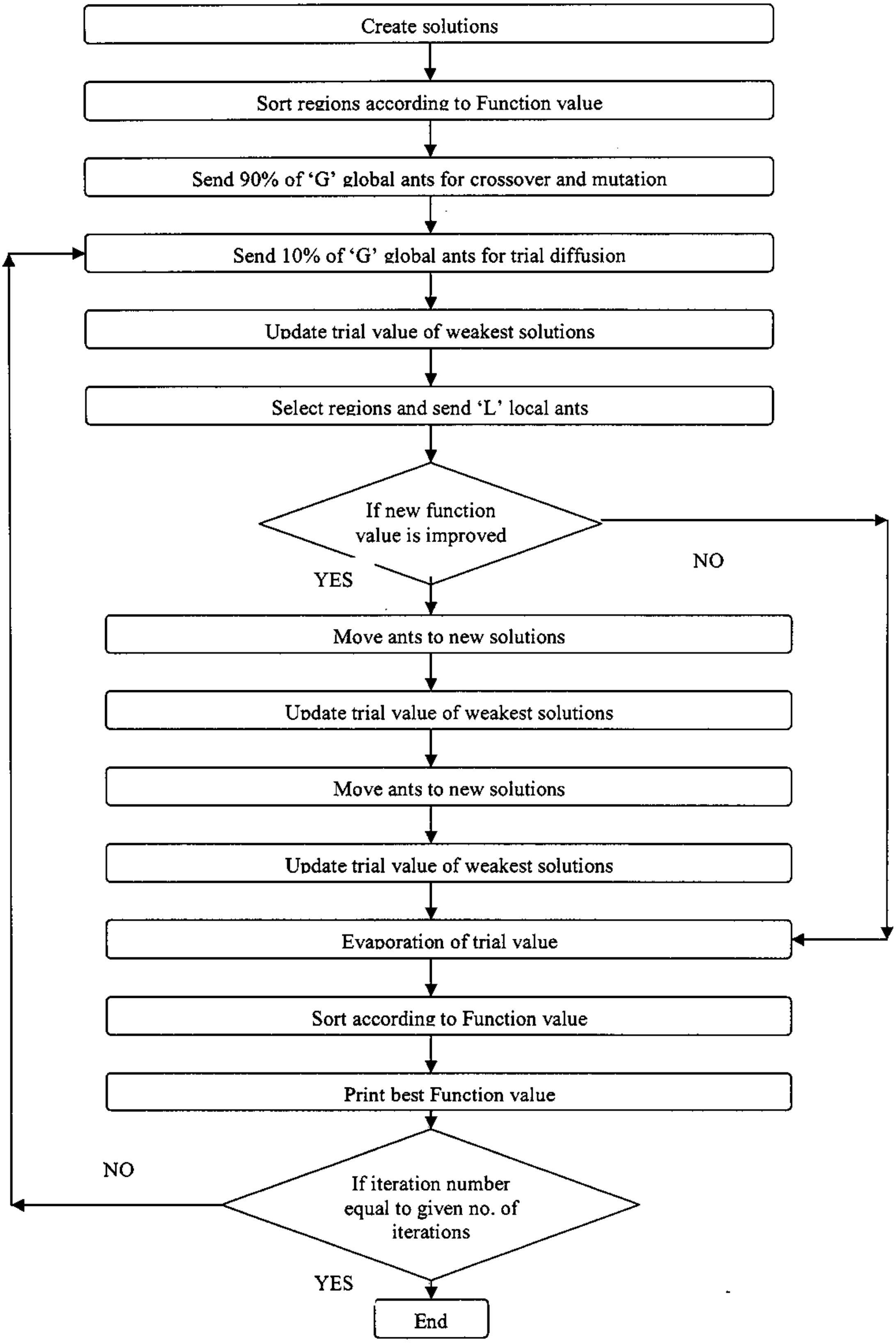
Create solutions
Sort regions according to Function value
Send 90% of 'G' global ants for crossover and mutation
Send 10% of 'G' global ants for trial diffusion
Update trial value of weakest solutions
Select regions and send 'L' local ants
If new function value is improved
NO
YES
Move ants to new solutions
Update trial value of weakest solutions
Move ants to new solutions
Update trial value of weakest solutions
Evaporation of trial value
Sort according to Function value
Print best Function value
NO
If iteration number equal to given no. of iterations
YES
End

FIG 6.5. FLOW CHART OF ACO ALGORITHM

6.2.6 Results of ACO

The software for the optimal allocation of total stock, minimization of total production cost using ACO has been implemented in C language. The present work is an optimization problem with constraints. The objective function is the total production cost, which consists of different passes. In the present work involving turning, the optimum results could be obtained with the population size of 10 and 500 generations. The optimum cost (Rs 46.4) is obtained .

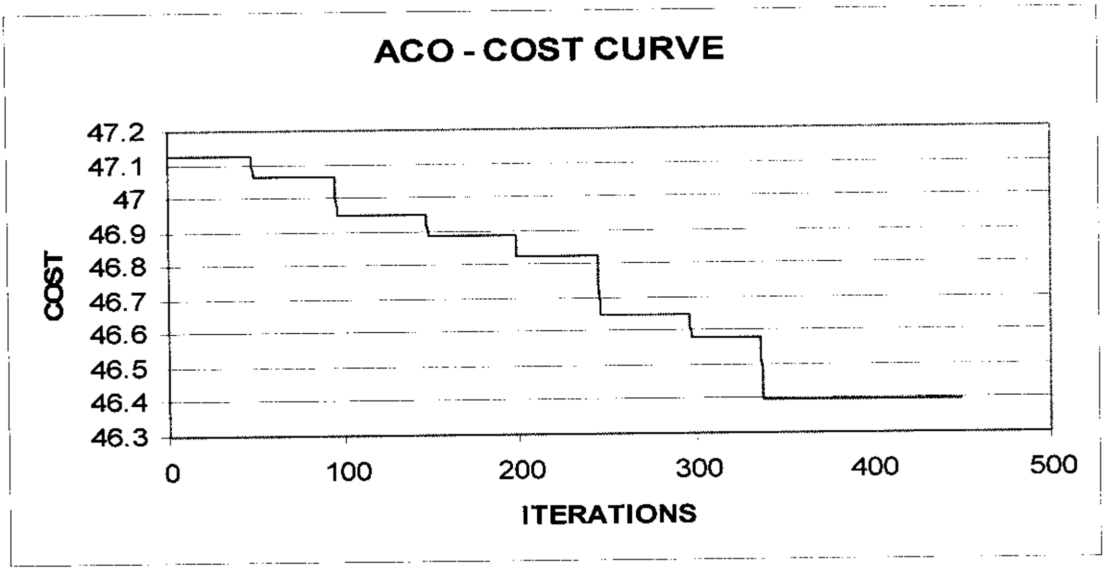

FIG. 6.6. COST CURVE

From fig.6.6, the cost is continuously decreased up to the optimum value

CHAPTER 7

CONCLUSIONS

The following conclusions were drawn from this work :

- This project presents a practical method of optimizing machining parameters to minimize production cost under the machining constraints.

- PSO was used to determine the optimal parameters such as speed , feed & depth of cut in each pass has yielded a minimum production cost.

- The working ranges of cutting speed, feed and depth of cut for turning of aluminum using tungsten carbide tipped tool – 4035 has been established.

- A five level three factor full factorial design matrix based on central composite rotatable technique was used for the development of mathematical models to predict the outputs.

- Based on the computational results presented herein, it may be concluded that the proposed nontraditional methods formulated present a significant enhancement in reducing unit production cost.

- It is also observed that the PSO algorithm can obtain a near optimal solution when compared to ACO in an extremely large solution space.

- The developed mathematical models can be used for optimizing the machining parameters using other intelligent optimization techniques.

REFERENCES

1. Agapiou S. (1992) "The Optimization of Machining Operations Based on Combined Criteria, Part:1 The use of Combined Objectives in Single Pass Operations," *ASME Journal of Engineering Industry,* 114, pp. 500-507.

2. Amman Aggarwal and Hari Singh (2005) "Optimization of machining techniques – A retrospective and literature review", Vol.30, pp 699-711.

3. Armarego E.J.A. Brown R.H. (1969) "The Machining of Metals," *Englewood Cliffs, National Journal: Prentice Hall.*

4. Asokan P, Baskar N, Babu K, Prabakaran G, and Saravanan R. (2005) "Optimization of surface grinding operations using particle swarm optimization technique", *Journal of Manufacturing Science and Engineering,* Vol.127, pp 885-892.

5. Bartarya G. and Choudhary S.K. (1999) "Role of temperature and surface finish in predicting tool wear and surface finish", *International Journal of Machine Tools and Manufacturing,* Vol.36, pp. 485 – 504.

6. Brewer R.C. Rueda R. (1963) "A Simplified Approach to the Optimum selection of Machining Parameters," *Engineering Digest* 24(9), pp. 133-150.

7. Choudhury S.K and Bartarya G. (2003) "Role of temperature and surface finish in predicting tool wear using neural network and design of experiments", *International Journal of Machine Tools & Manufacture,* Vol.43, pp 747-753.

8. ChristianBucher, Johannes Will and Jorg Riedal (2001), "Multidisciplinary non-linear optimization with optimizing structural language", *International Congress on FEM Technology.*

9. Cochran G. and Cox G.M. (1962) " Experimental Design," *New Delhi: Asia publishing house.*

10. Dawson T.G. and Kurfees T.R. (2000) "An Investigation of Tool Wear and Surface Quality in Hard Turning." *Transactions of 28th annual North American Manufacturing Research Institution of SME, University of Kentucky, Lexington,* May 24-26, pp. 215-220.

11. El Baradie M.A. (1993) "Surface Roughness Model for Turning Grey Cast Iron (154 BHN)," *Proceedings, Institute of Mechanical .Engineering* 207,pp. 43-50.

12. Ermer D. S. (1971) "Optimization of the Constrained Machining Economics Problem by Geometric Programming," *ASME J. Eng. Ind.,* 93, pp. 1067–1072.

13. Ermer D.S. and Kromordihardjo S. (1981) "Optimization of Multi- Pass Turning With Constraints," *Journal Engineering Industry* 103, pp. 462-468.

14. Gilbert W.W. (1950) "Economics of machining. In Machining – Theory and Practice" *Am. Soc. Met.,* pp. 476-480.

15. Gopalakrishnan B. and Faiz A. K. (1991) "Machining Parameter Selection for Turning With Constraints: An Analytical Approach Based on Geometric programming," *International Journal Prod. Res.,* 29, pp. 1897-1908.

16. GopalakrishnanB, and Khayyal F.A. (1991) "Machine Parameter Selection for Turning With Constraints: An Analytical Approach Based on Geometric Programming," *International Jornal Prod. Res.*29, pp. 1897-1908.

17. Hassan G.A, and Suliman S.M.A. (1990) "Experimental Modeling and Optimization of Turning Medium Carbon Steel," *International JournalProd.Res.28,*pp.1057-1065.

18. Hinduja S, Petty D.J, Teater M, Barrow G. (1985) " Calculation of Optimum Cutting Conditions for Turning Operations," *Proceedings, Institute Mechanical Engineering* 199(B2), pp. 81-92.

19. Kennedy J. et al. (1995) "Particle Swarm Optimization," *Proceedings of IEEE* International Conference on Neural Networks, IV, IEEE, Piscatway, National Journal, pp.1942–1948.

20. Lambert B.K. and Taraman K. (1973) "Development and utilization of a Mathematical Model of a Turning Operation," *International Journal Prod.Res.*11, pp. 69-81.

21. LIU Yi-jian, ZHANG Jian-ming, and WANG Shu-qing. (2005) "Parameter estimation of cutting tool temperature nonlinear model using PSO algorithm", *Journal of Zhejiang University SCIENCE*, pp 1009-3095.

22. Petropoulos P.G. (1973) "Optimal Selection of Machining Rate Variable by Geometric Programming," *International Journal Prod.Res.*11, pp. 305-314.

23. Sundara Vinayaga Murthy, Asokan P, and Baskar N. "Optimization of machining parameters for multi pass face milling operation using PSO technique", *Proceedings of the National Conference on Modeling and Analysis of Production System,* pp 19-25

24. Sundaram R.M. (1978) " An Application of Goal Programming Technique in Metal Cutting," *International Journal Prod.Res.*16, pp. 375-382.

25. Taraman K. (1974) "Multi-Machining Output-Multi Independent Variable Turning Research by Response Surface Methodology," *International Journal Prod.Res.*12,pp.232-245.

26. Taylor F. W. (1907) "On the art of cutting metals," *Trans. ASME 28,* pp. 31-35.

27. Tsai,P. (1986) "An Optimization Algorithm and Economic Analysis for a Constrained Machining model," *PhD thesis, West Virginia University.*

28. Walvekar A.G, and Lambert B.K. (1970) "An Application of Geometric Programming to Machining Variable Selection," *International Journal Prod.Res.8:3.*