OPTIMIZATION OF FCAW PROCESS PARAMETERS USING GENETIC ALGORITHM

A PROJECT REPORT

Submitted By

N.A.PRABU- 71206409008

In partial fulfillment for the award of the degree of

MASTER OF ENGINEERING

in

INDUSTRIAL ENGINEERING

DEPARTMENT OF MECHANICAL ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE – 641 006

ANNA UNIVERSITY :: CHENNAI 600 025

JUNE – 2008

ANNA UNIVERSITY :: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report entitled **"Optimization of FCAW Process Parameters Using Genetic Algorithm"** is the bonafide work of

Mr. N. A. Prabu - Register No. 71206409008

who carried out the project work under my supervision.

C Sivasundaram
Signature of the HOD

17/6/2008
Signature of the Supervisor

Internal Examiner

External Examiner

Department of Mechanical Engineering

KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE – 641 006

International Conference on

FRONTIERS IN DESIGN AND MANUFACTURING ENGINEERING

ICDM - '08

CERTIFICATE

This is to certify that N. A. Prabu has presented a paper in the International Conference on Frontiers in Design and Manufacturing Engineering (ICDM-'08) held at Karunya University, Coimbatore during Feburary 1-2, 2008

Secretary ICDM - '08

Registrar

Paul P. Appasamy
Vice-Chancellor

School of Mechanical Sciences, Karunya University, Coimbatore, India

ABSTRACT

Flux cored arc welding is used in most of the manufacturing industries. In flux cored arc welding process, it is very essential to optimize the process parameters to achieve the desired weld bead characteristics. In this project, objective function of maximizing bead width and height of reinforcement and minimizing the dilution and penetration were considered. Four process parameters welding current, welding speed, nozzle-to-plate distance and welding gun angle were identified for optimization.

To overcome the difficulties with conventional techniques, new non-traditional technique called Genetic Algorithm was implemented in this work.

MATLAB simulation software was used to solve the problem using Genetic Algorithm. Maximum of 100 iterations were performed and the solution was obtained. The computational efforts are very less and easy to implement.

ஆய்வு சுருக்கம்

பெரும்பாலான தொழிற்சாலைகளில் ப்ளக்ஸ் கோர்டு ஆர்க் வெல்டிங் முறையை பயன்படுத்தி வருகின்றனர். இந்த ப்ளக்ஸ் கோர்டு ஆர்க் வெல்டிங் முறையை கொண்டு வெல்ட் பீடின் தன்மையை அரிய சரியான உகப்பு நிலையை அறிவது முக்கியமானது. இந்த ஆய்வில் பீட் வித் மற்றும் ரீன்போர்ஸ்மெண்ட்டை அதிகரித்தல் அதே போன்று உகப்புநிலை மற்றும் பெனட்ரேசனை குறைத்தலை நோக்கமாக எடுத்து கொள்ளப்பட்டது. வெல்டிங்காக உபயோகிக்கப்படும் மின், வேகம், தகடிற்கும் வெல்டிங் செய்யும் கருவிக்கும் இடைபட்ட தூரம் மற்றும் கோணத்தை உகப்பாக்க இவ்வாய்வு மேற்கொள்ளபட்டது.

இதனை சரி செய்ய இவ்வாய்வில் மரபு நெறிமுறை உகப்பாக்கம் அமலாக்கப்பட்டது.

கணிப்பொறியின் மூலம் மென்பொருளின் உதவி கொண்டு தீர்வு கண்டுபிடிக்கப்பட்டது.

ACKNOWLEDGEMENT

With utmost gratitude to **Kumaraguru College of Technology**, I express my gratitude to my guide **Dr. T. Kannan**, Associate Professor, Department of Mechanical Engineering, for his guidance in carryout this project.

I take immense pleasure in expressing my sincere gratitude to the **Dr. C. Sivanandan**, Dean and Head of Department of Mechanical Engineering for his great support and enthusiasm.

I am thankful to **Dr. Joseph V. Thanikal**, Principal for giving me an opportunity to carry out this project successfully.

My sincere thanks to **Mr. P. S. Sivasakthivel**, Project Co-ordinator, Industrial Engineering, for his extreme co-operation and support.

My sincere thanks to **Mr. R. Rajasekar**, Software Engineer, Robert Bosch, Coimbatore.

I am very grateful to my father **Mr. S. Annamalai**, mother **Mrs. A. Malarvizhi**, for extending their help directly or indirectly to complete this project successfully.

- N. A. Prabu

CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

SYMBOL	ABBREVIATION
GA	Genetic Algorithm
LP	Linear Programming
FCAW	Flux Cored Arc Welding
I	Welding Current
S	Welding Speed
N	Nozzle-To-Plate Distance
T	Welding Gun Angle
GMAW	Gas Metal Arc Welding
W	Bead Width
P	Average Depth of Penetration
R	Height of Reinforcement
D	Percentage Dilution

CHAPTER 1

INTRODUCTION

1.1. OPTIMIZATION

Optimization is the process of obtaining the best result under given circumstances. In design, construction and maintenance of any engineering system, engineers/managers have to take many technological and managerial decisions at several stages, figure 1.1 shows the steps involved in the optimization process [11].

The ultimate goal of all such decisions is to either minimize the effort required or maximize the desired benefit.

Mechanical engineers design mechanical equipments like pumps, turbines and heat transfer equipment for maximum efficiency and mechanical components like linkages, cams, and gears, machine tools for the purpose of achieving either a minimum manufacturing cost or a maximum component life.

Production engineers are interested in designing optimum schedules of various machining operations to minimize the idle time of machines and the overall job completion time.

1.2. TYPES OF OPTIMIZATION TECHNIQUES

The following are the types of optimization.

i. Traditional optimization technique and
ii. Non- Traditional optimization technique

1.3. TRADITIONAL AND NON-TRADITIONAL OPTIMIZATION TECHNIQUES

Traditional techniques for optimization include linear programming, random search method, geometric programming, dynamic programming and

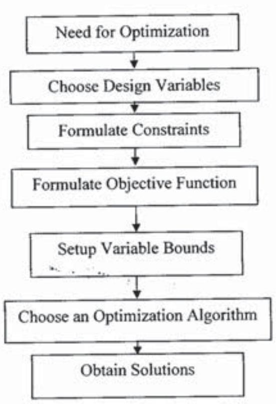


Figure 1.1 Steps involved in the Optimization Process [4]

1.3.1. Linear Programming

Linear programming is an optimization method applicable for the solution of problems in which the objective function and the constraints appear as linear functions of the decision variables. The constraint equations in a linear programming problem may be in the form of equalities or inequalities. The linear programming type of optimization problem was first recognized in the 1930's by economist while developing methods for the optimal allocation of resources.

Linear programming is considered a revolutionary development that permits us to make optimal decisions in complex situations. At least four Noble Prizes were awarded for contributions related to linear programming.

1.3.2. Random Search Method

The random search method described for unconstrained minimization can be used, with minor modifications, to solve a constrained optimization problem. The basic procedure can be described by the following steps:

i. Generate a trial design vector using one random number for each design variable.

ii. Verify whether the constraints are satisfied at the trial design vector. Usually, the equality constraints are considered satisfied whenever their magnitudes lie within a specified tolerance. If any constraint is violated, continue generating new trial vectors until a trial vector that satisfies all the constraints is found.

iii. If all the constraints are satisfied, retain the current trial vector as the best design if it gives a reduced objective function value compared to the previous best available design. Otherwise, discard the current feasible trial vector and proceed to step 1 to generate a new trial design vector.

iv. The best design available at the end of generating a specified maximum number of trial design vectors is taken as the solution of the constrained optimization problem.

1.3.3. Geometric Programming

Geometric Programming is a relatively new method of solving a class of non linear programming problems. It was developed by Duffin, Peterson and Zener. It is used to minimize functions that are in the form of polynomials subject to constraints of the same type. It differs from other optimization techniques in the emphasis it places on the relative magnitudes of the terms of the objective function rather than the variable. Instead of finding optimal values of the design variables first, geometric programming first finds the optimal value of the objective function. This feature is especially advantageous in situations where the optimal value of the objective function may be all that is of interest. In such cases, calculation of the optimum design vectors can be omitted. Another advantage of geometric programming is that it often reduces a complicated optimization problem to one involving a set of simultaneous linear algebraic equations. The major disadvantage of the method is that it requires the objective function and the

1.3.4. Dynamic Programming

Dynamic programming is a mathematical technique well suited for the optimization of multistage decision problems. This technique was developed by Richards Bellman in the early 1950s.

The dynamic programming technique, when applicable, represents of decomposes a multistage decision problem as a sequence of single-stage decision problems. Thus an N-variable problem is represented as a sequence of N single-variable problems that are solved successively. In most cases, these N sub-problems are easier to solve than the original problem. The decomposition to N sub-problems is done in such a manner that the optimal solution of the original N-variable problem can be obtained from the optimal solutions of the N one-dimensional problems. It is important to note that the particular optimization technique used for the optimization of the N single- variable problems is irrelevant. It may range from a simple enumeration process to a differential calculus or a nonlinear programming technique.

The dynamic programming technique suffers from a major drawback, known as the curse of dimensionality. However, despite this disadvantage, it is very suitable for the solution of a wide range of complex problems in several areas of decision making.

1.3.5. Integer Programming

When all the variables are constrained to take only integer values in an optimization problem, it is called an integer programming problem. When the variables are restricted to take only discrete values, the problem is called a discrete programming problem. When some variables only are restricted to take integer values, the optimization problem is called a mixed-integer programming problem. When all the design variables of an optimization problem are allowed to take on values of either zero or one, the problem is called zero-one programming problem.

Non-traditional techniques for optimization include fuzzy logic, search technique, genetic algorithm, Taguchi technique and response surface methodology [1].

1.3.6. Fuzzy Logic

Fuzzy logic has great capability to capture human commonsense reasoning, decision-making and other aspects of human cognition. It overcomes the limitation of classical logical systems, which impose inherent restrictions on representation of imprecise concepts. Vagueness in the coefficients and constraints may be naturally modeled by fuzzy logic. Modelling by fuzzy logic opens up a new way to optimize cutting conditions and also tool selection.

1.3.7. Genetic Algorithm (GA)

These are the algorithms based on mechanics of natural selection and natural genetics, which are more robust and more likely to locate global optimum. It is because of this feature that GA goes through solution space starting from a group of points and not from a single point. The cutting conditions are encoded as genes by binary encoding to apply GA in optimization of machining parameters. A set of genes is combined together to form chromosomes, used to perform the basic mechanisms in GA, such as crossover and mutation.

Crossover is the operation to exchange some part of two chromosomes to generate new offspring, which is important when exploring the whole search space rapidly. Mutation is applied after crossover to provide a small randomness to the new chromosomes. To evaluate each individual or chromosome, the encoded cutting conditions are decoded from the chromosomes and are used to predict machining performance measures. Fitness or objective function is a function needed in the optimization process and selection of next generation in genetic algorithm. Optimum results of cutting conditions are obtained by comparison of values of objective functions among all individuals after a number of iterations. Besides weighting factors and constraints, suitable parameters of GA are required to operate efficiently. GA optimization methodology is based on GA parameters along with relevant objective functions and set of machining performance constraints are imposed on GA optimization methodology to provide optimum cutting conditions.

1.3.8. Scatter Search Technique

This technique originates from strategies for combining rules and surrogate constraints. Scatter Search is completely generalized and problem-independent since it has no restrictive assumptions about objective function, parameter set and constraints set. It can be easily modified to optimize machining operation under various economic criteria and numerous practical constraints. It can be extended as an on-line quality control strategy for optimizing machining parameters based on signals from sensors.

1.3.9. Taguchi Technique

Genichi Taguchi is a Japanese engineer who has been active in the improvement of Japan's industrial products and processes since the late 1940's. He has developed both the philosophy and methodology for process of products quality improvement that depends heavily on statistical concepts and tools, especially statistically designed experiments. Many Japanese firms have achieved great success by applying his methods. Wu (1982) has reported that thousands of engineers have performed tens of thousands of experiments based on his teachings. Sullivan (1987) reports that Taguchi has received some of Japan's most prestigious awards for quality achievement, including the Deming prize. In 1986, Taguchi received the most prestigious prize from the International Technology major contribution has involved combining engineering and statistical methods to achieve rapid improvements in cost and quality by optimizing product design and manufacturing processes.

1.3.10. 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 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. Lastly, techniques for proper interpretation of results are devised.

1.4. ADVANTAGES OF NON-TRADITIONAL OPTIMIZATION TECHNIQUES

The advantages of Non-traditional optimization techniques are as follows: [4]

i. A population of points is used for starting the procedure instead of a single design point.

ii. GA uses only the values of the objective function. The derivatives are not used in the search procedure.

iii. Search method is naturally applicable for solving discrete and integer programming problems. For continuous design variables, the string length can be varied to achieve any desired resolution.

iv. The objective function value corresponding to a design vector plays the role of fitness in natural genetics.

v. In every new generation, a new set of strings is produced by using randomized parents selection and crossover from the old generation.

1.5. PROCESS OPTIMIZATION

When optimization is based on the processes that the product undergoes is called Process optimization.

The ability to control a process does not guarantee optimal control [14]. Optimal process control can be a difficult task due to several reasons:

i. Complex correlations between process variables might make it necessary to consider many parameters simultaneously during process adjustments.

ii. Several process levels might exist, all with different optimal variable settings.

iii. Changes in raw material and process conditions require continuous adjustments of variable settings.

iv. Several quality parameters might need to be optimized simultaneously.

A comprehensive and successful process optimization should thus entail:

i. A dynamic optimization goal that should consist of a cost efficient weighted combination of the interesting process outputs (production variations, production cost, product qualities and emission levels). A dynamic goal also means that it should be possible to automatically change the optimization goal as the process levels change.

ii. Handling of any process complexity with possibilities to successfully carryout the optimization, whether output from process models can be used or not.

iii. Handling of long term process changes with possibilities to continuously carryout optimization regardless of seasonal changes or changes in raw material.

This project work is mainly concerned with process parameters such as welding current (I), welding speed (S), nozzle-to-plate distance (N) and welding gun angle (T) of Flux Cored Arc Welding (FCAW). Hence it comes under process optimization.

1.6. OBJECTIVES OF OPTIMIZATION

Following are the objectives of optimization

ii. To decrease the fatigue of the worker who is on the shop floor.

iii. To increase productivity of the organization gradually.

iv. To satisfy the employees in the organization.

v. Procurement of material will be very less because of the higher productivity.

1.7. TYPES OF SOLUTIONS [7]

i. A solution to an optimization problem specifies the values of the decision variables, and also the value of the objective function.

ii. A feasible solution satisfies all constraints.

iii. An optimal solution is feasible and provides the best objective function value. There may be multiple optimal solutions for a given problem.

iv. A near optimal solution is feasible and provides a superior objective function value, but not necessarily the best.

1.8. CLASSIFICATION OF OPTIMIZATION PROBLEMS [7]

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 equation: 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 indicates 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 of real valued, and deterministic or stochastic.

In this thesis optimization of flux cored arc welding (FCAW) is attempted using an optimization technique called Genetic Algorithm (GA).

Chapter 2

Literature Survey

CHAPTER 2

LITERATURE SURVEY

2.1. LITERATURE SURVEY

Following are the overview of the relevant work done earlier related to the problem identified and the methodology to be adopted to solve the chosen problem for this work. It gives the description of literature reviewed from various research papers published in journals, proceedings of various conferences and books.

Murugan, et al. (2004), has used GA to optimize the process parameters to achieve minimum dilution, maximum reinforcement, minimum penetration and maximum bead width with the view of economizing on material.

Correia, et al. (2004), had done a work using GA as a method to decide near-optimal settings of a GMAW welding process. The problem was to choose the near-best values of three control variables (welding voltage, wire feed rate and welding speed) based on four quality responses (deposition efficiency, bead width, depth of penetration and reinforcement), inside a previous delimited experimental region. The search for the near-optimal was carried out step by step, with the GA predicting the next experimental based on the previous, and without the knowledge of the modeling equations between the inputs and outputs of the GWAW process. The GAs was able to locate the near-optimum conditions with a relatively small number of experiments.

Kannan, et al. (2005), has done the work on combined objective function of maximizing the bead penetration, minimizing the dilution, reinforcement and width was considered. Four SAW process parameters (voltage, wire feed rate, welding speed and nozzle-to-plate distance) were identified for optimization subjected to realistic process constraints. Several conventional techniques had been suggested in the literature for solving this problem. But these techniques are understand and implement. In order to overcome the difficulties with conventional technique called particle swarm optimization was implemented in this work.

Kannan, et al. (2005), has solved the problem of selecting optimum combination of input process parameters for achieving the required clad quality by optimizing the process parameters. In this paper an experimental study and analysis of various input parameters and important clad quality parameters in duplex stainless steel cladding of low carbon structural steel plates deposited by flux cored arc welding. The experiments were conducted based on the four-factor five levels central composite rotatable design with full replications techniques and mathematical models were developed using multiple regression method.

Sathiya, et al. (2005), had proposed a method to decide near optimal settings of the welding process parameters in friction welding of austenitic stainless steel by using a Genetic Algorithm. This method tries to find near optimal settings of the welding process parameters through experiments without a model between the inputs and output variable. It has an advantage of being able to carryout search without modifying the design space, which includes some irregular points. The method suggested in this study is used to determine the welding process parameters by which the desired tensile strength can be obtained in friction welding. The output variable is the tensile strength.

Aman Aggarwal, et al. (2005), had made an attempt to review the literature on optimizing machining parameters in turning processes. Various conventional techniques employed for machining optimization include geometric programming, geometric plus linear programming, goal programming, sequential unconstrained minimization technique, dynamic programming etc. The latest techniques for optimization include fuzzy logic, scatter search technique, genetic algorithm, and Taguchi technique and response surface methodology.

Kannan, et al. (2006) has conducted experiments to study and analyze the effects of various FCAW process parameters on important clad quality parameters in duplex stainless steel cladding of low carbon structural steel plates. The experiments were conducted using the four factor five level central composite

developed using multiple regression method. The effects of the input process parameters on clad quality parameters on clad quality parameters have been presented in graphical form , which helps in selecting welding process parameters to achieve the desired clad quality quickly.

Kumar, et al. (2006), has developed a Seven feed forward neural networks were for gas metal arc fillet welding, one each for predicting penetration, leg length, throat, weld pool length, cooling time between 800°C and 500°C, maximum velocity and peak temperature in the weld pool.

Mishra, (2007) has used genetic algorithm to determine a population of solutions by minimizing an objective function that represents the difference between the calculated and the desired values of weld pool penetration and width. The use of a neural network in place of a heat transfer and fluid flow model significantly expedites the computational task.

Kumar, et al. (2007) shown that the various combinations of welding variables necessary to achieve a target gas metal arc fillet weld geometry can be systematically and quickly computed by a real-number-based genetic algorithm and a neural network that has been trained with the results of a heat transfer and fluid flow model.

Crina Grosan et al. (2007) Evolutionary Computation has become an important problem solving methodology among many researchers. The population based collective learning process; self adaptation and robustness are some of the key features of evolutionary algorithms when compared to other global optimization techniques. Even though evolutionary computation has been widely accepted for solving several important practical applications in engineering, business, commerce etc., yet in practice sometimes they deliver only marginal performance. Inappropriate selections of various parameters, representation, etc. are frequently blamed. There is little reason to expect that one can find a uniformly best algorithm for solving all optimization problems.

Chapter 3

Process Selection and Data

CHAPTER 3

PROCESS SELECTION AND DATA COLLECTION FOR OPTIMIZATION

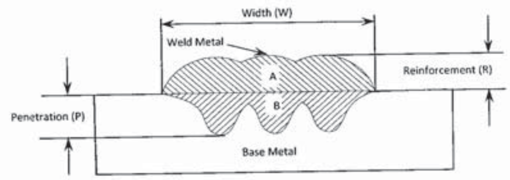
3.1. INTRODUCTION

Weld cladding is an excellent way to impart properties to the surface of a substrate that are not available from that of base metal. Typical base metal components that are weld cladded include the internal surfaces of carbon and low-alloy steel pressure vessels used in chemical, fertilizer, food processing and petrochemical plants. The biggest difference between welding a joint and cladding is dilution. It is the amount of base metal melted divided by the sum of the filler added and base metal melted.

Dilution reduces the alloying elements and increases the carbon content in clad layer which reduces corrosion resistance properties, and causes other metallurgical problems [6]. The composition and properties of cladding are strongly influenced by the dilution obtained. Control of dilution is very important in cladding, where low dilution is typically desirable.

Various welding processes employed for weld cladding are Shielded Metal Arc Welding, Submerged Metal Arc Welding, Gas Tungsten Arc Welding, Plasma Arc Welding, Gas Metal Arc Welding, Flux Cored Arc Welding (FCAW), Electroslag Welding, Oxy-Acetylene Welding and Explosive Welding. In this work, FCAW process was selected for optimization due to the following features,

i. High speed deposition rate and increased productivity
ii. Smooth welding characteristics and weld finish
iii. Lower cost for the shielding gas
iv. Simple and more cost effective post weld cleaning
v. Reliable and consistent weld quality



Percentage Dilution (D) = [B/(A+B)]x100

Figure 3.1. weld Bead Geometry

3.2. DATA COLLECTION

Kannan and Murugan had chosen process parameters for the study were welding current (I), welding speed (S), nozzle-to-plate distance (N), and welding torch angle (T) in Materials Processing Technology, April 2006, pp 230-239. The chosen response variables were weld bead width (W), average depth of penetration (P), average height of reinforcement (R), and percentage dilution (D). The chosen levels of the selected process parameters with their units and notations are given in Table 3.1.

Table 3.1. Welding process parameters and their levels [6]

Parameter	Unit	Notation	Factor Levels				
			-2	-1	0	+1	+2
Welding Current	A	I	200	225	250	275	300
Welding Speed	cm/min	S	20	30	40	50	60
Nozzle-to-Plate Distance	mm	N	22	24	26	28	30
Welding Gun Angle	degree	T	20	15	10	05	00

The following data were collected from their paper for this work.

Table 3.2. Welding Conditions (Natural Scale) and their Weld Bead Geometry dimensions [7]

Trial No.	Process Parameters				Weld Bead Geometry			
	I (A)	S (cm/min)	N (mm)	T °	W (mm)	P (mm)	R (mm)	D (%)
1	225	30	24	15	29.50	0.61	4.97	07.86
2	275	30	24	15	36.62	0.73	5.00	12.10
3	225	50	24	15	24.20	0.63	4.23	11.35
4	275	50	24	15	28.00	0.77	4.27	11.98
5	225	30	28	15	30.00	0.57	4.80	06.54
6	275	30	28	15	34.98	0.67	4.95	08.82
7	225	50	28	15	25.59	0.58	4.18	09.69
8	275	50	28	15	29.51	0.70	4.24	11.16
9	225	30	24	05	28.34	0.73	5.32	08.97
10	275	30	24	05	34.50	0.97	5.10	13.75
11	225	50	24	05	24.00	1.00	4.20	18.52
12	275	50	24	05	27.80	1.20	4.34	20.58
13	225	30	28	05	29.26	0.60	5.25	07.46
14	275	30	28	05	34.80	0.80	5.22	09.14
15	225	50	28	05	25.30	0.97	3.57	18.00
16	275	50	28	05	27.70	1.00	4.21	14.80
17	200	40	26	10	20.15	0.40	3.98	05.86
18	300	40	26	10	31.00	1.07	4.90	16.48
19	250	20	26	10	39.53	0.70	5.68	05.31
20	250	60	26	10	23.10	1.00	3.63	17.35
21	250	40	22	10	25.10	0.83	4.81	11.71
22	250	40	30	10	28.00	0.63	4.32	09.01
23	250	40	26	20	30.20	0.56	4.17	10.54

Trial No.	Process Parameters				Weld Bead Geometry			
	I (A)	S (cm/min)	N (mm)	T °	W (mm)	P (mm)	R (mm)	D (%)
25	250	40	26	10	27.88	0.70	4.55	10.33
26	250	40	26	10	29.42	0.83	4.35	13.60
27	250	40	26	10	28.00	0.77	4.48	10.73
28	250	40	26	10	27.90	0.87	4.50	11.71
29	250	40	26	10	29.20	0.83	4.32	13.76
30	250	40	26	10	27.80	0.79	4.58	10.99
31	250	40	26	10	27.80	0.80	4.57	10.67

W – Width, P – Penetration, R – Reinforcement, D - %Dilution

The following equations are taken from the paper [6] for writing the M file.

Bead width (W) (mm) = $27.225+2.494I-3.244S+0.415N-0.610T-0.303I^2+1.066S^2+0.316T^2-0.616IS$ (3.1)

Average Depth of Penetration (P) (mm) = $0.764+0104I+0.074S-0.048N+0.110T+0.021S^2+0.061ST$ (3.2)

Average height of reinforcement (R) (mm) = $4.535+0.128I-0.475S+0.054N+0.052T+0.053S^2-0.052SN$ (3.3)

Percentage Dilution (D) = $11.702+1.466I+2.730S-1.037N+1.608T-0.751IS-0.593IN+1.482ST$ (3.4)

Chapter 4

Optimization Of Flux Cored Arc Welding Using Genetic Algorithmic Approach

CHAPTER 4

OPTIMIZATION OF FLUX CORED ARC WELDING USING GENETIC ALGORITHMIC APPROACH

4.1. INTRODUCTION

The idea of applying the biological principle of natural evolution to artificial systems, introduced more than three decades ago, has seen impressive growth in the past few years. The basic concept of Genetic Algorithm is to encode a potential solution to a problem as a series of parameters. A single set of parameter value is treated as the genome of an individual solution. An initial population of individuals is generated at random or statistically.

Every evolutionary step, known as a generation, the individuals in the current population are decoded (evaluated) according to some predefined quality criterion, referred to as fitness function. The chromosomes with the highest population fitness function. The chromosomes with the highest population fitness score are selected for mating. The genes of the two parents are allowed to exchange to produce offsprings. These children then replace their parents in the next generation. Thus, the old population is discarded and the new population becomes the current population. The current population is checked for acceptability of solution. The iteration is stopped after the completion of maximal number of generations or on the attainment of the best result.

4.2. BASIC DESCRIPTION OF GENETIC ALGORITHM

The Genetic Algorithms are inspired by Darwin's theory about evolution [8]. Algorithm is started with a set of solutions from one population are taken and used to form a new population. This is motivated by a hope, that the new population will be better than the old one. Solutions which are selected to form new solutions (offsprings) are selected according to their fitness. The more suitable they are, the more chances they have to reproduce. This is repeated until some conditions (for example, number of population or improvement of the best solution) are satisfied.

4.3. OUTLINE OF BASIC GENETIC ALGORITHM CYCLE

The Genetic Algorithm cycle used in this study is illustrated in Figure 4.1. The various steps involved are briefly described as given below.

4.3.1. Start

Random populations of 'n' chromosomes (suitable solutions for the problem) are generated.

4.3.2. Fitness

The fitness function of each chromosome in the population is evaluated.

4.3.3. New Population

A new population is created by repeating following steps.

4.3.3.1. Selection

Two parent chromosomes are selected from the population according to their fitness, better the fitness, bigger the chance to be selected.

4.3.3.2. Cross-over

The parents are crossed over to form a new offspring with a cross-over probability.

4.3.3.3. Childless

If no cross-over is performed, offspring is an exact copy of parents.

4.3.3.4. Mutation

New offsprings are mutated with a mutation probability.

4.3.3.5. Accepting

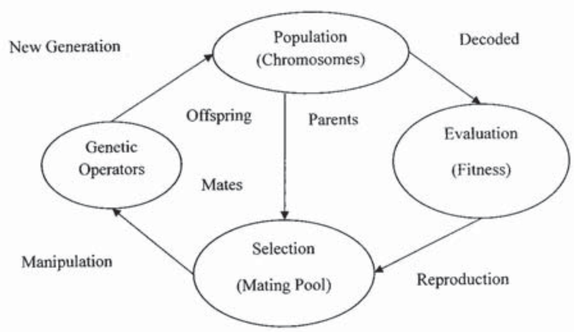


Figure 4.1. Genetic Algorithm Cycle [8]

4.3.3.6. Replace

Newly generated population is used for a further run of algorithm, that is, individuals from old population are killed and replaced by the new ones.

4.3.3.7. Test

The generation is stopped, if the end condition is satisfied and returns the best solution in current population.

4.3.3.8. Loop

If the termination criteria are not met, the loop is repeated from the fitness step again as reported above.

Chapter 5

Optimization Using Genetic Algorithm Tool In MATLAB

CHAPTER 5

OPTIMIZATION USING GENETIC ALGORITHM TOOL IN MATLAB

5.1. OPTIMIZATION PROCEDURE

The FCAW optimization procedure using genetic algorithm shown in figure 5.1 is used in this study. Here, initial population means the lower bounds of the optimization problems, and each possible solution is called an individual. In this work, a possible solution is formed by values of the welding current, welding speed, nozzle-to-plate distance and welding gun angle. The following section will give the clear cut procedure of using the MATLAB genetic algorithm tool options for the problem that has been chosen [13].

5.1.1. Fitness Function

Fitness function is the objective function that is to minimize. Specify the function as a function handle of the form @objfun, where objfun.m is an M-file that returns a scalar. In this study percentage dilution is taken as objective function.

5.1.2. Number of variables

The numbers of independent variables for the fitness function. Here four numbers of variables were taken for simulation.

5.1.3. Plot Functions

Plot functions enable to plot various aspects of the genetic algorithm as it is executing. Each one will draw in a separate axis on the display window. The stop button on the window is used to interrupt a running process. The following section shows the options in the gatool command window.

5.1.3.1. Plot interval

Plot interval specifies the number of generations between successive

5.1.3.2. Best Fitness Plot

Best fitness plot, plots the best function value in each generation versus iteration number. The best fitness value will show the optimized value of the objective function. In this work it shows the optimized value of percentage dilution, as shown in the figure 6.1.

5.1.3.3. Expectation Plot

Expectation plot will plot the expected number of children versus the raw scores at each generation.

5.1.3.4. Score Diversity Plot

Score diversity plot will plot a histogram of the scores at each generation.

5.1.3.5. Stopping Plot

Stopping plot will plot the stopping criteria levels.

5.1.3.6. Best Individual Plot

Best individual plot, plots the vector entries of the individual with the best fitness function value in each generation. Here it plots the histogram of welding current, welding speed, nozzle-to-plate distance and welding gun angle which is indicated as 1, 2, 3 and 4 respectively in the figure 6.2.

5.1.3.7. Genealogy Plot

Genealogy plot plots the genealogy of individuals. Lines from one generation to the next are color-coded as follows:

i. Red lines indicate mutation children.
ii. Blue lines indicate crossover children.
iii. Black lines indicate elite individuals.

Genealogy plot is shown in the figure 6.3.

5.1.3.8. Scores Plot

5.1.3.9. Distance Plot

Distance plot will plot the average distance between individuals at each generation.

5.1.3.10. Range Plot

Range plot will plot the minimum, maximum, and mean fitness function values in each generation.

5.1.3.11. Selection Plot

Selection plot will plot a histogram of the parents. This shows you which parents are contributing to each generation.

5.1.3.12. Custom function

Custom function enables to use own plot function.

5.1.4. Population Options

Population options specify options for the population of the genetic algorithm as shown in the table 5.2.

5.1.4.1. Population type

Population type specifies the type of the input to the fitness function. Double vector is taken as the population type.

5.1.4.2. Population size

Population size specifies how many individuals there are in each generation. If population size to be a vector of length greater than 1, the algorithm creates multiple subpopulations. Each entry of the vector specifies the size of a subpopulation. For this work population size is set as five.

5.1.4.3. Creation function

Creation function specifies the function that creates the initial population. The default creation function uniform creates a random initial population with a

5.1.4.4. Initial population

Initial population enables to specify an initial population for the genetic algorithm. An initial population not specified, for this work the algorithm will create one using the creation function.

5.1.4.5. Initial scores

Initial scores enable to specify scores for initial population. If initial scores not specified, the algorithm will compute the scores using the fitness function. For this work initial score was not specified.

5.1.4.6. Initial range

Initial range specifies lower and upper bounds for the entries of the vectors in the initial population. Initial range can be specified as a matrix with 2 rows and initial length columns. The first row contains lower bounds for the entries of the vectors in the initial population, while the second row contains upper bounds. If initial range is specified as a 2-by-1 matrix, the two scalars are expanded to constant vectors of length initial length. Here the initial range is specified as [-2, -2, -2, -2; 2, 2, 2, 2] which is the coded values of the input.

5.1.5. Fitness Scaling Options

The scaling function converts raw fitness scores returned by the fitness function to values in a range that is suitable for the selection function.

5.1.5.1. Scaling function

Scaling function specifies the function that performs the scaling. The following functions are the various options:

i. Rank will scale the raw scores based on the rank of each individual, rather than its score. The rank of an individual is its position in the sorted scores. The rank of the fittest individual is 1, the next fittest is 2 and so on. Rank fitness scaling removes the effect of the spread of the raw scores.

ii. Proportional will make the expectation proportional to the raw fitness score. This strategy has weaknesses when raw scores are not in a "good" range.

iii. Top will scale the individuals with the highest fitness values equally. If this option selected, then specify as quantity, the number of fittest individuals that produce offspring. Quantity must be an integer between 1 and population size or a fraction between 0 and 1 specifying a fraction of the population size. Each of these individuals has an equal probability of reproducing. The rest have probability 0 of reproducing. The expectation has the form [0 1/n 1/n 0 0 1/n 0 0 1/n ...].

iv. Shift linear - The function will scale the raw scores so that the expectation of the fittest individual is equal to a constant, which should be specified as maximum survival rate, multiplied by the average score.

v. Custom will enables to write own scaling function.

Among these options rank is chosen as the scaling function for this work.

5.1.6. Selection Options

The selection function will choose parents for the next generation based on their scaled values from the fitness scaling function.

Selection option is chosen from the following selection functions:

i. Stochastic uniform will layout a line in which each parent corresponds to a section of the line of length proportional to its expectation. The algorithm will move along the line in steps of equal size, one step for each parent. At each step, the algorithm will allocate a parent from the section it lands on. The first step is a uniform random number less than the step size.

ii. Remainder will assign parents deterministically from the integer part of each individual's scaled value and then uses roulette selection on the

iii. Uniform will select parents at random from a uniform distribution using the expectations and number of parents. This results in an undirected search. Uniform selection is not a useful search strategy, but can used to test the genetic algorithm.

iv. Roulette simulates a roulette wheel with the area of each segment proportional to its expectation. The algorithm then uses a random number to select one of the sections with a probability equal to its area.

v. Tournament - The function will select each parent by choosing individuals at random, and then choosing the best individual out of that set to be a parent.

vi. Custom will enables to write own selection function.

Hence roulette is chosen for the study.

5.1.7. Reproduction Options

Reproduction options will determine how the genetic algorithm creates children at each new generation.

5.1.7.1. Elite count

Elite count will specify the number of individuals that are guaranteed to survive to the next generation. Hence elite count must be a positive integer less than or equal to population size.

5.1.7.2. Crossover fraction

Crossover fraction will specify the fraction of the next generation, other than elite individuals, that are produced by crossover. The remaining individuals, other than elite individuals, in the next generation are produced by mutation. So crossover fraction should be a fraction between 0 and 1, either by entering the fraction in the text box or moving the slider.

5.1.8. Mutation Options

Mutation functions will make small random changes in the individuals in the population, which provide genetic diversity and enable the GA to search a broader space. Specify the function that performs the mutation in the mutation function field. The option is chosen from the following functions:

i. Uniform is a two-step process. First, the algorithm will select a fraction of the vector entries of an individual for mutation, where each entry has a probability of mutation rate of being mutated. In the second step, the algorithm will replace each selected entry by a random number selected uniformly from the range for that entry.

ii. Gaussian will add a random number to each vector entry of an individual. This random number is taken from a Gaussian distribution centered on zero. The variance of this distribution can be controlled with two parameters. The scale parameter determines the variance at the first generation. The shrink parameter controls how variance shrinks as generations go by. If the shrink parameter is 0, the variance is constant. If the shrink parameter is 1, the variance shrinks to 0 linearly as the last generation is reached.

iii. Custom will enables to write own mutation function.

Uniform is the option for the study.

5.1.9. Crossover Options

Crossover combines two individuals, or parents, to form a new individual, or child, for the next generation.

It specifies the function that performs the crossover in the crossover function field. From the following functions the crossover option is chosen:

i. Single point will choose a random integer n between 1 and number of variables, and selects the vector entries numbered less than or equal to n

second parent, and concatenates these entries to form the child. For example,

p1 = [a b c d e f g h]
p2 = [1 2 3 4 5 6 7 8]
crossover point (at random) = 3
child = [a b c 4 5 6 7 8]

ii. Two point will select two random integers m and n between 1 and number of variables. The algorithm selects genes numbered less than or equal to m from the first parent, will select genes numbered from m+1 to n from the second parent, and selects genes numbered greater than n from the first parent. The algorithm then concatenates these genes to form a single gene. For example,

p1 = [a b c d e f g h]
p2 = [1 2 3 4 5 6 7 8]
crossover points (at random) = 3,6
child = [a b c 4 5 6 g h]

iii. Scattered will create a random binary vector. It then selects the genes where the vector is a 1 from the first parent, and the genes where the vector is a 0 from the second parent, and combines the genes to form the child. For example,

p1 = [a b c d e f g h]
p2 = [1 2 3 4 5 6 7 8]
random crossover vector = [1 1 0 0 1 0 0 0]
child = [a b 3 4 e 6 7 8]

iv. Intermediate creates children by a weighted average of the parents. Intermediate crossover is controlled by a single parameter ratio:

child1 = parent1 + rand * ratio * (parent2 - parent1)

If ratio is in a larger range, say 1.1 then children can be generated outside the hypercube. Ratio can be a scalar or a vector of length number of variables. If ratio is a scalar, then all of the children will lie on the line between the parents. If ratio is a vector then children can be any point within the hypercube.

v. Heuristic will create children that lie on the line containing the two parents, a small distance away from the parent with the better fitness value in the direction away from the parent with the worse fitness value.

vi. Custom will enable to write own crossover function.

So intermediate is used for this work.

5.1.10. Migration Options

Migration is the movement of individuals between subpopulations, which the algorithm creates if population size is to be a vector of length greater than 1. Every so often, the best individuals from one subpopulation replace the worst individuals in another subpopulation. Controlling migration occurs by the following three parameters.

5.1.10.1. Direction

Migration can take place in one direction or two.

i. If direction is set as forward, migration takes place toward the last subpopulation. That is the nth subpopulation migrates into the (n+1)'th subpopulation.

ii. If direction is set as both. the nth subpopulation migrates into both the (n-1) th and the (n+1) th subpopulation.

Migration wraps at the ends of the subpopulations. That is, the last subpopulation migrates into the first. and the first may migrate into the last. To prevent wrapping, a subpopulation of size zero is specified.

5.1.10.2. Fraction

Fraction controls how many individuals move between subpopulations. Fraction is the fraction of the smaller of the two subpopulations that moves. If individuals migrate from a subpopulation of 50 individuals into a population of 100 individuals and fraction is 0.1, 5 individuals (0.1 * 50) migrate. Individuals that migrate from one subpopulation to another are copied. They are not removed from the source subpopulation.

Default migration fraction 0.2 is taken for the work.

5.1.10.3. Interval

Interval controls how many generations pass between migrations. For this work migration interval is set as 20, hence migration between subpopulations takes place every 20 generations.

5.1.11. Hybrid Function Options

Hybrid function enables to specify another minimization function that runs after the genetic algorithm terminates. The choices are

i. None
ii. fminsearch
iii. patternsearch
iv. fminunc

None is set as hybrid function option. Because no other function will run after GA terminates.

5.1.12. Stopping Criteria Options

Stopping criteria will determine what causes the algorithm to terminate.

5.1.12.1. Generations

Generations specify the maximum number of iterations the genetic

5.1.12.2. Time limit

Time limit will specify the maximum time in seconds the genetic algorithm runs before stopping.

5.1.12.3. Fitness limit

If the best fitness value is less than or equal to the value of fitness limit, the algorithm stops.

5.1.12.4. Stall generations

If there is no improvement in the best fitness value for the number of generations specified by stall generations, the algorithm stops.

5.1.12.5. Stall time limit

If there is no improvement in the best fitness value for an interval of time in seconds specified by stall time limit, the algorithm stops.

5.1.13. Output Function Options

5.1.13.1. History to new window

The iterative history of the algorithm outputs to a separate window.

5.1.13.2. Interval

Interval specifies the number of generations between successive outputs.

5.1.13.3. Custom

Custom enables to write own output function.

5.1.14. Display to Command Window Options

5.1.14.1. Level of display

Level of display specifies the amount of information displayed in the MATLAB command window while running the genetic algorithm. Option is chosen from the following options:

i. Off - Only the final answer is displayed.

ii. Iterative - Information is displayed for each iteration.

iii. Diagnose - Information is displayed if the algorithm fails to converge. In addition, options that are changed from the defaults are listed.

iv. Final - The outcome of the pattern search (successful or unsuccessful), the reason for stopping, and the final point are displayed.

Off is chosen for the work.

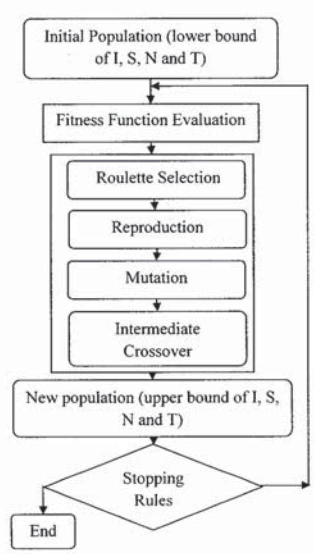


Figure 5.1. Flowchart for Genetic Algorithm

5.2. SIMULATION PROCEDURE

The aim of this study is to find the optimum adjusts for the welding closest possible of the cited values. And it is assumed that the near optimal point is within the following experimental region proposed by Kannan and Murugan and is shown in the Table 5.2.

Table 5.1. GA search ranges [5]

Parameters	Range
Welding current(I)	200-300 A
Welding speed (S)	20-60 cm/min
Nozzle-to-plate distance (N)	22-30 mm
Welding gun angle (T)	0-20°

When the MATLAB command window is opened, M-file has been created and saved as the file name dot m. Then, in the MATLAB command window to open GA tool, type gatool and press enter. When GA toolbox is opened, enter the fitness function as @file name (same file name where the M-file has been saved), number of variables that is used for the fitness function and select the plots required. Following Table 5.2 show the options used for the study

In GA, the population size, crossover rate and mutation rate are important factors in the performance of the algorithms. A large population size or a higher crossover rate allows exploration of the solution space and reduces the chances of settling for poor solution. However, if they are too large or high, it results in wasted computation time exploring unpromising regions of the solution space.

About mutation rate, if it is too high, there will be much random perturbation, and the offspring will loose the good information of the parents. The 1% value is within the typical range for the mutation rate. The crossover rate is 90% i.e., 90% of the pairs as crossed, whereas the remaining 10% are added to the next generation without crossover. The chosen type of the crossover was single, which means that a new individual is formed when the parent genes are swapped

Table 5.2. Options of GA computation

Population type	Double Vector
Population size	5
Creation Function	Uniform
Initial range	[-2,-2,-2,-2;2,2,2,2]
Fitness scaling function	Rank
Selection function	Roulette
Reproduction elite count	2
Crossover fraction	0.9
Mutation function	Uniform
Mutation rate	0.01
Crossover function	Intermediate
Migration Direction	Forward
Migration Fraction	0.2
Migration Interval	20
Hybrid Function	none
Number of generations	100
Stall Generations	50
Stall time limit	20

5.3. SELECTION OF OBJECTIVE FUNCTIONS AND CONSTRAINTS

The objective function selected for optimization was percentage dilution. The response variables bead width, average depth of penetration and average height of reinforcement were given as constraints in their equation form. In optimization, generally the constraints with their upper bounds should be given in

application, it is always desirable to have maximum weld bead width and reinforcement with minimum penetration. The process parameters and their notations used in writing the M-file using MATLAB software are given below.

x(1) = Welding current (I)
x(2) = Welding speed (S)
x(3) = Nozzle-to-plate distance (N)
x(4) = Welding torch angle (T)

5.4. OPTIMIZATION OF THE FUNCTION

The purpose is optimization of weld bead geometry parameters with their limits as constraints. The model is a nonlinear equation with constraints. The constrained minimum of a scalar function of several functions of several variables at an initial estimate, which is referred as "constrained nonlinear optimization" is mathematically stated as follows

Minimize f(x)
Subject to g(x1, x2, x3…xn) <0

The limits of the constraints bead width; penetration and reinforcement were established by data obtained from past experience with a view that they should provide a sound and defect-free weld bead along with a feasible solution to the objective function.

Several numerical methods are available for optimization of non linear equation with constraints. A Genetic Algorithm method is efficient and quickest one, and this method was used to determine the optimum percentage dilution. The step by step procedure of minimization of percentage dilution using the GA optimization tool box available in MATLAB software is given below.

Step 1: writing M-file function [f, g] =f(x)

f(1)=11.702+1.466*x(1)+2.73*x(2)-1.037*x(3)+1.608*x(4)-0.751*x(1)*x(2)-0.593* x(1)* x(3) +1.482*x(2)*x(4);Percentage dilution.

g(1)=27.775+2.494*x(1)-3.244*x(2)+0.45*x(3)-0.61*x(4)-0.303*x(1)^2+1.066* x(2)^2 + 0.316* x(4)^2 -0.616*x(1)*x(2)-39.53;Bead Width and its upper limit.

g(2)=20.15-27.775+2.494*x(1)-3.244*x(2)+0.45*x(3)-0.61*x(4)-0.303*x(1)^2 +1.066* x(2)^2 + 0.316*x(4)^2-0.616*x(1)*x(2); Bead Width and lower limit.

g(3)=0.764+0.104*x(1)+0.074*x(2)-0.048*x(3)+0.11*x(4)+0.021 *x(2)^2+0.061 *x(2)* x(4)-1.2; Penetration and its upper limit.

g(4)=0.4-0.764+0.104*x(1)+0.074*x(2)-0.048*x(3)+0.11*x(4)+0.021*x(2)^2+ 0.061* x(2)*x(4); Penetration and its lower limit.

g(5)=4.535+0.128*x(1)-0.475*x(2)+0.054*x(3)+0.052*x(4)+0.053*x(2)^2-0.052 *x(2)*x(3)-5.68; Reinforcement and its upper limit.

g(6)=3.63-4.535+0.128*x(1)-0.475*x(2)+0.054*x(3)+0.052*x(4)+0.053*x(2)^2-0.052* x(2)*x(3); Reinforcement and its lower limit.

g(7)=f-20.58; upper limit of percentage dilution.

g(8)=5.31-f; lower limit of percentage dilution.

Step 2: invoke an optimization routine

Select and type in the corresponding boxes as per the requirement as shown in the Table 5.2.

Step 3: Run the M-file.

Chapter 6

Results and Discussions

CHAPTER 6

RESULTS AND DISCUSSIONS

6.1. GENETIC ALGORITHMIC APPROACH

After running the M-file in MATLAB simulation software for the options that has been shown on table 5.2, following various optimum results have been obtained.

The optimum values of the process parameters were obtained at 52^{nd} iteration.

Corresponding input parameters are as follows.

x(1) = Welding current (I) = 261.005A
x(2) = Welding speed (S) = 26.28 cm/min
x(3) = Nozzle-to-plate distance (N) = 27.89198 mm
x(4) = Welding torch angle (T) = 11.86535 °

For these optimized process parameters, the values of the clad quality parameters are

Bead Width (W) = 36.66542 mm
Penetration (P) = 0.666234 mm
Reinforcement (R) = 5.441047 mm
Percentage Dilution (D) = 10.8667%

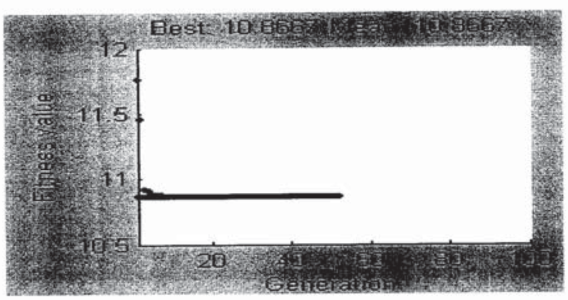


Figure 6.1. Best Fitness Plot

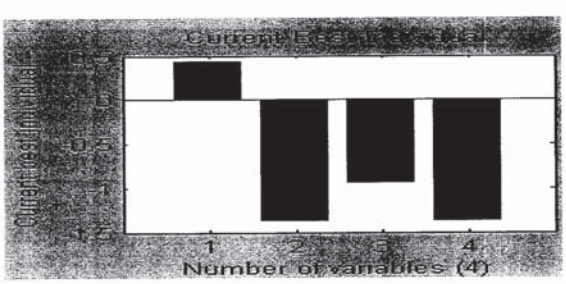


Figure 6.2. Best Individual Plot

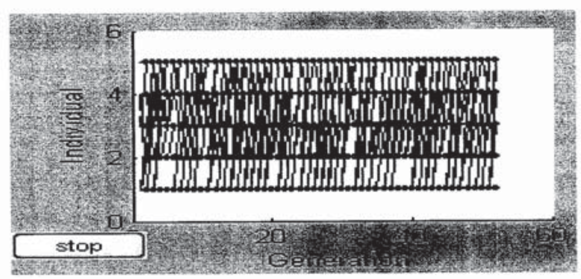


Figure 6.3. Genealogy Plot

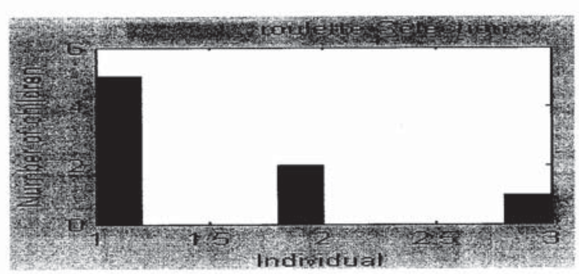


Figure 6.4. Roulette Selection Plot

The above figures 6.1., 6.2., 6.3. and 6.4. are the plots obtained when the optimized results was obtained.

Chapter 7

Conclusion

CHAPTER 7

CONCLUSION

The possibility of a FCAW optimization procedure using GA is investigated in this work; more specifically, the determination of the near optimal FCAW process parameters, welding current, welding speed, nozzle-to-plate distance and welding torch angle. The search for the optimization was based on the minimization of an objective function.

It was found that GA can be a powerful tool in experimental welding optimization, even when the experimenter does not have a model for the process.

However, the optimization by GA technique requires a good setting of its own parameters, such as population size, number of generations, etc. otherwise there is a risk of an insufficient sweeping of the search space. These results will solve lot of real time problems in the manufacturing industries.

Appendix 1

APPENDIX 1

Various optimum results obtained while simulating the gatool in MATLAB.7 software while computing different number of trials. Some of them are shown in the following screen shots.

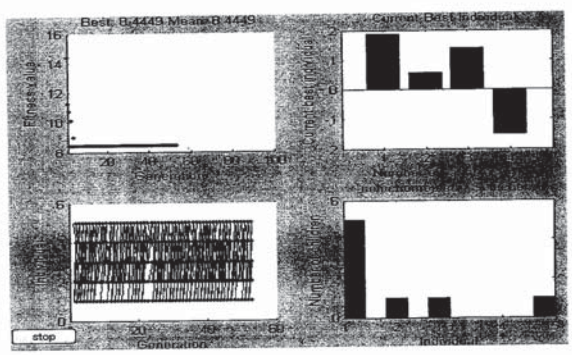


Figure A.1. Percentage Dilution 8.4449%

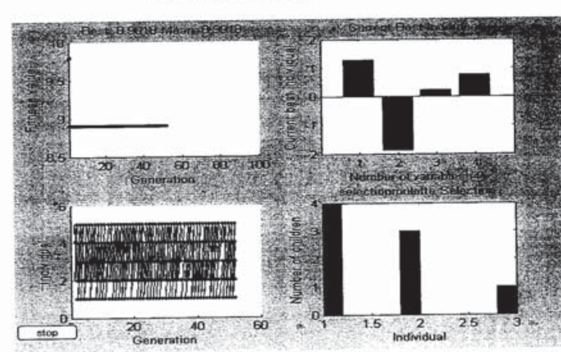


Figure A.2. Percentage Dilution 8.9018%

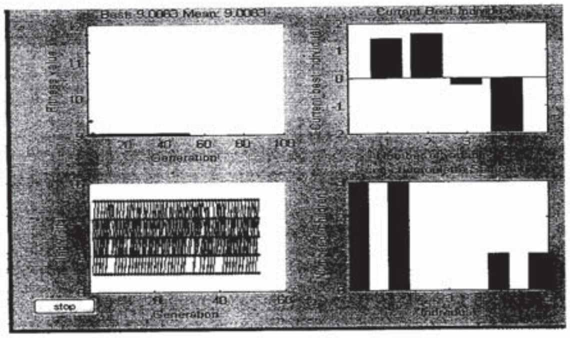


Figure A.3. Percentage Dilution 9.0083%

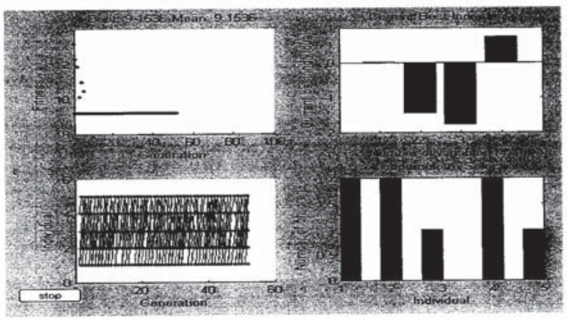


Figure A.4. Percentage Dilution 9.1536%

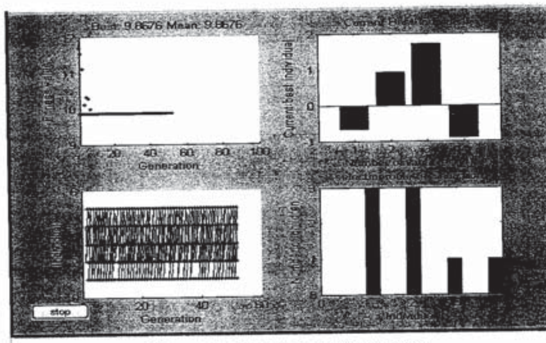


Figure A.5. Percentage Dilution 9.8676%

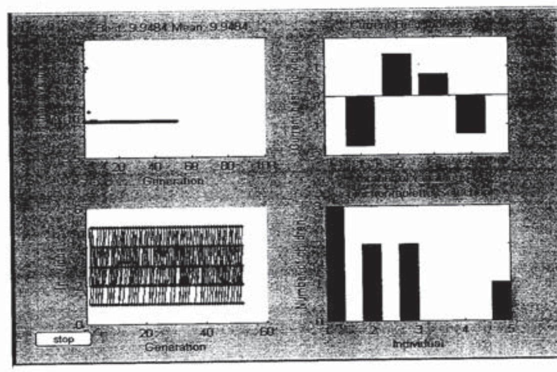


Figure A.6. Percentage Dilution 9.9484%

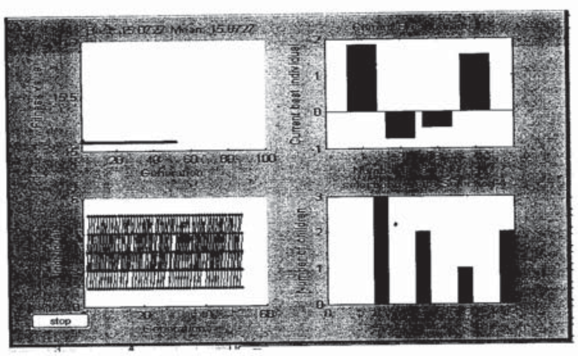


Figure A.7. Percentage Dilution 15.0727%

References

REFERENCES

1. Aman Aggarwal and Hari Singh (1995) "Optimization of Machining Techniques- A retrospective and Literature Review", Sadhana, Vol. 30, Part 6, pp 699-711.

2. Correia.D.S., Goncalves.C.V. Sebastiao.S.C.Junior and Ferraresi.V.A, (2004) 'GMAW Welding Optimization Using Genetic Algorithms' Journal of the Braz. Soc. Of Mech. Sci, & Eng., Vol.26, No.1 pp 28-33.

3. Dongcheol Kim, Sehun Rhee and Hyunsung Parks (2002) "Modelling and Optimization of a GMA Welding Process by Genetic Algorithm and Response Surface Methodology", International Journal of Production, Vol. 40, No.7 pp 1699-1711.

4. Kalyanmoy Deb (2006) 'Optimization for Engineering Design Algorithms and Examples' 9th printing, Prentice Hall India.

5. Kannan.T and Murugan.N. (2006) 'Effect of FCAW process parameters on duplex stainless steel clad quality', Journal of Materials Processing Technology, vol. 176, pp 230-239.

6. Kannan.T and Murugan.N.(2005) 'Optimization of FCAW process parameters in Duplex Stainless Steel Weld Cladding' Journal of Manufacturing Technology Today, Vol. No. 4, I. No. 4 pp 3-7.

7. Kannan.T, Saravanan.R and Ramesh.T. (2005) 'Optimization of SAW Process Parameters using Particle Swarm Optimization' Manufacturing Technology Today, Vol. No. 4, I.No. 11, pp 16-19.

8. Murugan.N and Palani.P.K.(2004) 'Optimization of Bead Geometry in Automatic Stainless Steel Cladding by MIG Welding using a Genetic Algorithm' IE(I) Journal-PR, Vol.84, pp 49-54.

9. Saravanan.R, Asokan.P and Sachithanandam.M. (2001) 'Comparative Analysis of Conventional and Non-Conventional Optimization Techniques for CNC Turning Process', International Journal of Advanced Manufacturing Technology 17:471-476.

10. Sathiya.P. Aravindan.S, Noorul Haq.A, Panneerselvam.K. (2005) 'Optimization of Process Parameters of Friction Welding by Genetic Algorithm' Manufacturing Technology Today, Vol. No. 4, I.No. 2, pp 9-15.

11. Singiresu S.Rao (1995) 'Engineering Optimization Theory and Practice' 3rd Edition, New Age Publication.

12. Wolfgang Banzhaf, Peter Nordin, Robert E. Keller and Frank D. Francone (1998) 'Genetic Programming An Introduction' Morgan Kaufmann Publishers.

13. www.mathworks.com/gatool

14. www.multisimplex.com/optimization.htm