A STEPPER MOTOR DESIGN OPTIMIZATION USING GENETIC ALGORITHMS

Oleh

Wong Chin Wei

Disertasi ini dikemukakan kepada
UNIVERSITI SAINS MALAYSIA

Sebagai memenuhi sebahagian daripada syarat keperluan
untuk ijazah dengan kepujian

SARJANA MUDA KEJURUTERAAN (KEJURUTERAAN ELEKTRIK)

Pusat Pengajian Kejuruteraan
Elektrik dan Elektronik
Universiti Sains Malaysia

Mac 2005

ABSTRACT

In this project, the design optimization of a stepper motor is presented. In general, the area of study can be divided into motor principles and construction, design methods, and digital control experiments. Theory is taught in classroom lectures, whereas control methods are learned primarily in laboratory situations. Instruction on motor design, however, is usually limited to the study of motor construction, with practically no laboratory time spent on the actual fabrication of motors. The production process, including material processing and winding, would take up too much time and expense. There is a need to fill this void in the area of small-motor design, and develop a program using Genetic Algorithms (GAs) as an approach to achieve optimization. The aim of optimum design in this project is to minimize the volume, weight and cost of stepper motor while keeping constraint variable at the desired value. In order to achieve the optimum design, Genetic Algorithms (GAs) approach has been applied. GAs approach is selected because it is a powerful and broadly applicable stochastic search and optimization techniques that works for many problems that are very difficult to solve by conventional methods. The design optimization procedure of a stepper motor is described in this project. A C++ program has been successfully developed based on the GAs by using the GAs library. This GAs library is a C++ library that contains tools and built-in components for using GAs to minimize the fitness function. In this project, the program that has been developed is run to get the optimization result with Microsoft Visual C++. In order to obtain better results from the program, some of the parameters have to be changed. These include GA parameter that is number of generation and size of population and penalty factor. From the result, it is shown that the objective function is achieved while keeping other constraint function at desired value. This project and successful results have proved the suitability of GA for design optimization of electrical equipment. It is shown that GA can be used to solve complex problems within a short period.

ACKNOWLEDGEMENT

The completion of this thesis is not possible without the support and encouragement from my well wishers. It is my great pleasure to express my sincere gratitude to my supervisor, Dr. K.S. Rama Rao, Department of Electrical and Electronic Engineering, Universiti Sains Malaysia, for his knowledgeable insight, professionalism, advice, and patience. Without his kindness and help, this project could not be completed successfully. I really appreciate it.

I want to thank my parents for their love, education, support, encouragement, and prayers. Their support gave me spiritual strength in pursuing my ambition and completion of this thesis. Life has no taste to me without my parents and my family.

Finally, I want to thank all my friends and laboratory assistants who assisted me upon request and were very helpful in their suggestions.

CONTENTS

PAGE

CHAPTER 3 OPTIMIZATION APPROACH OF GENETIC ALGORITHMS

CHAPTER 4 DESIGN METHODOLOGY

CHAPTER 1

INTRODUCTION

.

1.1 Introduction to stepper motor

Stepping motor has gained an important place in modern technology and is used widely in office automation and factory automation application. Stepping motors can be viewed as electric motors without commutator. Stepper motors consist of a rotating shaft, called the rotor, and electromagnets on the stationary portion that surrounds the motor, called the stator. All of the commutation must be handled externally by the motor controller, and typically, the motors and controllers are designed so that the motor may be held in any fixed position as well as being rotated one way or the other. To move the rotor the electric magnets on the motor are activated in the right order. Every change in this process moves the motor one step. The order in which those electromagnets are activated determines the rotation direction. The stepper's resolution is based on the steps. In the stepper system, the driver advances one step, and the stepper motor follows. Each step covers a specific range of "swing". In a nutshell, a stepper (with or without gear-train) is a set of "preset" positions that can move to. Any position that's not part of the "presets" is unattainable by that motor or motor-and-gear-train combination, and can only be reached as an approximation.

There is virtually no conceivable failure within the stepper drive module that could cause the motor to run away. Stepper motors are simple to drive and control in an open-loop configuration. They provide excellent torque at low speeds, up to 5 times the continuous torque of a brush motor of the same frame size or double the torque of the equivalent brushless motor. This often eliminates the need for a gearbox. A stepper-driven system is inherently stiff, with known limits to the dynamic position error.

Stepper motors have the following benefits:

• Low cost

• Ruggedness

• Simplicity in construction
• High reliability
• No maintenance
• Wide acceptance
• No tweaking to stabilize
• No feedback components are needed
• They work in just about any environment
• Inherently more failsafe than servo motors.

Stepper motors have their limitations. They are available in limited power and their rotation speed is limited**.** The energy efficiency of stepper motors is low and stepper motor systems have tendency to have resonances which needs to be avoided. Stepper motors have characteristic holding torque (ability to hold the position) and pullout torque (ability to move to the next position). Because of open-loop nature of stepper motor controlling, they are not very good to be used with varying loads. It is possible for a stepper motor to loose steps if it is loaded too much. The stepper motors typically have a rated voltage at what they can work without overheating. Operating the motor at this voltage limits the maximum speed and torque at high speed. The current limiting can be done by using power resistors or a chopper drive to keep current at the desired level.

Stepper motors have the following disadvantages:
• Resonance effects and relatively long settling times
• Rough performance at low speed unless a micro step drive is used
• Liability to undetected position loss as a result of operating open-loop
• They consume current regardless of load conditions and therefore tend to run hot
• Losses at speed are relatively high and can cause excessive heating, and they are frequently noisy (especially at high speeds).
• They can exhibit lag-lead oscillation, which is difficult to damp. There is a limit to their available size, and positioning accuracy relies on the mechanics. Many of these drawbacks can be overcome by the use of a closed-loop control scheme

A wide variety of electrical motors are in use and the stepping motor can be classified into several types according to machine structure and principle of operation. There are three main stepper motor types:

- Variable Reluctance (V.R.) Motors
- Permanent Magnet (P.M.) Motors
- Hybrid Motors

1.2 Optimization

Studies on design, construction of stepper motor are less compared with the control drive of stepper motor. Although there are a large number of publications describing the simulation of a stepping motor system, relatively few deal with the motor design. The designing of a stepper motor is a complex problem and need a lot of study. A lot of factors have to be considered like number of poles, dimensions of the motor, number of turns and conductor cross-sectional areas for the armature windings and so on. Besides that, certain performance criteria, for example, the efficiency and the torque of the motor have to be taking into account. Almost all the variables in the problem are interdependent and thus no unique solution is available. To realize a best design, optimization techniques are ideally used.

Selecting suitable design variables, an objective function, geometric and performance constraints are to be defined for the optimization problem. For a stepper motor, the objective function and constraints are highly nonlinear. The optimal design of such motor needs a nonlinear programming method. GA approach used in this project is suitable to obtain an optimal design of stepper motor. With the help of penalty factor, the constrained optimization problem is converted into an unconstrained optimization problem by forming an augmented objective function.

1.3 Genetic Algorithms Approach

Genetic algorithms are now widely applied in science and engineering as adaptive algorithms for solving practical problems. Certain classes of problems are particularly suited to being tackled using a GA based approach. The general acceptance is that GAs is particularly suited to multidimensional global search problems where the search space potentially contains multiple local minima. Unlike other search methods, correlation between the search variables is not generally a problem. The basic GA does not require extensive knowledge of the search space, such as likely solution bounds or functional derivatives. A task for which simple GAs are not suited is rapid local optimization; however, coupling the GA with other search techniques to overcome this problem is trivial.

GAs were first introduced by John Holland for the formal investigation of the mechanisms of natural adaptation but the algorithms have been since modified to solve computational search problems. Modern GAs deviate greatly from the original form proposed by Holland, but their linage is clear. There is no single firm definition for a genetic algorithm, and the computational system is highly simplified compared to the actual situation in nature.

1.3.1 Genetic Optimizations

In the past few years, many studies have been carried out in the design optimization of electrical devices. Such a task is often a computationally hard problem since the relationships between the geometrical and electrical parameters that generally speaking represent the design variables and the performance of the device to be optimized are usually nonlinear and highly complex.

The task of optimization is minimizing or maximizing an objective function related to the motor torque, the efficiency, the cost of material or some other motor figures of merit., preserving a set of given specifications related to the motor performance. By emulating the mechanism of natural selection and natural genetics, this technique tends to reproduce motors with improved performance without surpassing the imposed limits on the materials stresses. This is achieved by selecting the best motors in each population of them and trying to discharge the worst ones.

1.4 Objective

The main aim of this project is to develop a design optimization procedure of a stepper motor using a programming technique. To arrive at an optimal design minimizing the cost of volume or weight of the stepper motor, a program is developed considering certain performance criteria of the motor. First a set of design variables are specified and then objective and constraint functions are expressed in terms of the design variables. A non-linear programming GA method is then used, as the design equations are non-linear. The design optimization problem of stepper motor is written in programmer language C++ and solved by Microsoft visual C++ 6.0. There are three important things to be considered before using the GA method for solving this optimization problem, which are selection of design variables, design constraints and the objective function.

1.5 Organization of the project report

This report is divided into six main chapters

I. Introduction
II. Design consideration of stepper motor
III. Optimization approach of Genetic Algorithms
IV. Design methodology
V. Results and analysis
VI. Conclusion

Each chapter is also divided into some sub-sections for convenience organized to present useful information in this chapter.

In 1 Chapter-Introduction, the overall picture of this project is presented based on how this project is carried out. A brief introduction of stepper motor is given and the reason why this project is selected also stated. Genetic Algorithms, a method to obtain optimal design is also briefly discussed.

In Chapter 2-Design Consideration of Stepper Motor, the concept and theory about the design of stepper motor is explained. The geometry of motor, winding connection, materials used and other performance of motor is discussed. Important equations that are used in the design are stated.

In Chapter-3 Optimization Approach of Genetic Algorithms, history of genetic algorithms is presented. The concept of GA and how it is implemented for design optimization is discussed in detail.

In Chapter-4 the Design Methodology of this project is presented. The objective functions and constraint equations are derived in forms of the specified design variables. All equations are then written programming language C++.

Chapter 5, Results and Analysis, deals with the optimization procedure that takes the form of minimization of a particular objective function of the stepper motor. For a selected set of GA parameters, changing the number of generation, optimal design values and minimum objective functions are computed and tabulated. The results are analyzed and graphs are plotted.

In the last chapter, the results which are obtained from the program an graphs are briefly discussed. The conclusions of the project are presented at the end of this chapter.

CHAPTER 2

DESIGN CONSIDERATIONS OF STEPPER MOTOR

2.1 Introduction

Step motors are electromagnetic actuators that convert digital pulse inputs to analog output motion. These motors differ from the conventional motors in that, when energized by a voltage or current pulse train, the step motor moves in an incremental fashion. Each pulse advances the motor shaft one step increment.

Step motors are used in many types of control systems in the industry. By far the highest-volume users of step motors are found in the computer industry. Such peripheral equipment as printers, tape drives, capstan drives and disk drives for computer memories all have step motors in them. In addition, step motors are used in numerical control system, machine tool controls, process control system, robots, clocks and watches and numerous other applications.

In the last chapter, type of stepper motors that most widely used in industry are listed out. The most basic and simple construction is the variable reluctance motor type and is selected in this project for design optimization.

After a type of motor is selected, the next consideration is to select parameters of motor, which yields the specified rated voltage and current. Based on selected parameters, all the related equations are formed. Important performance parameters of motor are also included.

This chapter first will explore the fundamentals of stepper motor- principle of operation and features of motor that effect design considerations. Stepper motor design deals with:

1) Selection of geometry of motor – shape/geometry
2) Consideration of tooth geometry
3) All losses that may occur i.e. iron losses and copper losses
4) Winding coil connection

2.2 Fundamental of stepper motor

In order to get a whole picture of stepper motor, principle operation, features of stepper motor, aspect of design and various losses that involved is investigate.

2.2.1 Principle operation

The variable-reluctance stepping motor or VR motor for short may be considered to be the most basic type of stepping motor. The cross-section of a motor is illustrated in Figure 2.1 to facilitate the explanation of the basic principles. This is a three-phase motor having six stator teeth. Any two opposing stator teeth which are at 180^o from each other belong to the same phase; that is the coil on each opposing tooth is connected in series or parallel. The rotor has four teeth. Both the stator and rotor materials must have high permeability and be capable of allowing a high magnetic flux to pass through them even if a low magneto motive force is applied. Current to each phase is controlled in the ON/OFF mode by their respective switches. If current is applied to the coils of phase 1, (Ph1), the magnetic flux will occur as shown in Figure2.2. The rotor will then be positioned so that the stator teeth 1 and 1' and any two of the rotor teeth are aligned. Thus when the rotor teeth and stator teeth are in alignment, the magnetic reluctance is minimized and this state provides a rest or equilibrium position. When Ph1 is turned off and Ph2 is turned on (refer to Figure 2.3), the motor reluctance seen from the DC power supply will be suddenly increased just after the switching takes place. The rotor will then move through a step angle of 30^o counter-clockwise so as to minimize the reluctance. This motion through a step angle at each switching of excitation is called a step. After completing a rotor-tooth-pitch rotation in three steps, the rotor will apparently return to its original position. This is illustrated in Figure 2.4

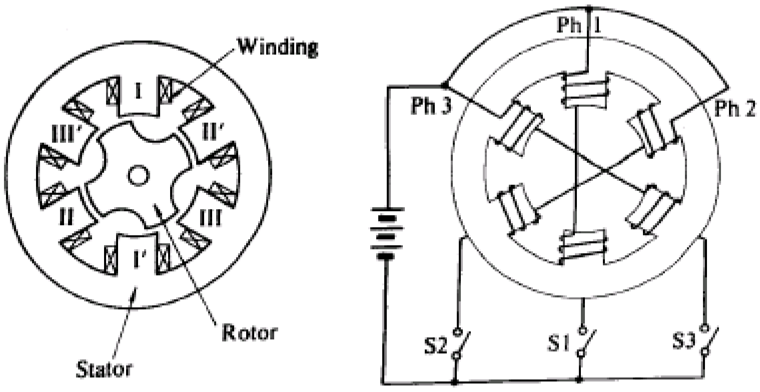

Figure 2.1 Cross section model of a three phase VR stepping motor and winding arrangement

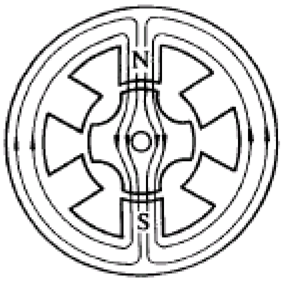

Figure 2.2 Equilibrium position with phase 1 excited

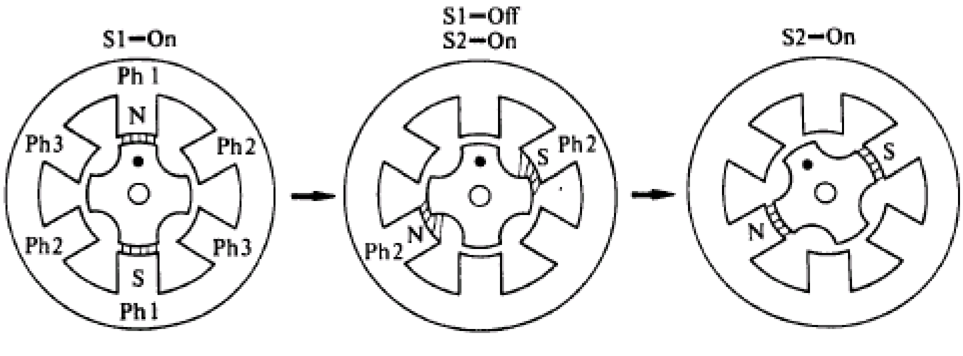

Figure 2.3 How a step motion proceed when excitation is switched from Ph1 to Ph2

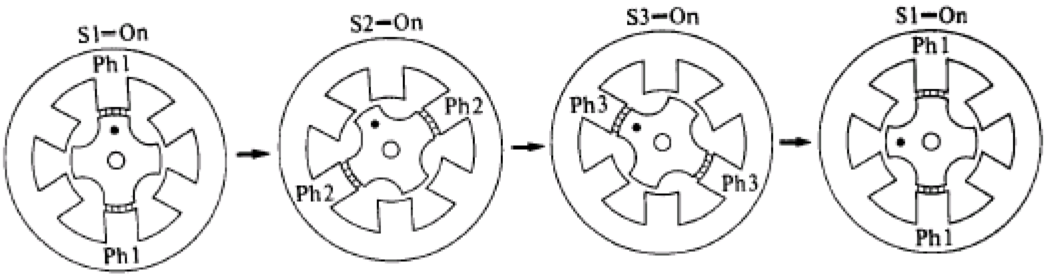

Figure 2.4 Step motion as switching sequence proceeds in a three phase VR motor

2.2.2 Features of stepper motor

In this section, features of open-loop control of stepper motor and some technical terms will discuss in detail.

2.2.2.1 Small step angle

A stepping motor rotates through a fixed angle for each pulse. As explained earlier, the rated value of this is called the step angle and expressed in degrees. Decreasing the step angle increases the resolution of positioning. One feature of stepping motor is that they can be made to realize a small step angle. Number of steps per revolution, denoted by S and step angle θ_s is related to S as follows

$$\theta_s = \frac{360}{S} \tag{2.1}$$

S is related to the number of teeth on the rotor, P_r and the number of phases q as

$$S = q.P_r \tag{2.2}$$

2.2.2.2 High holding and restoring torque

Stepping motors are designed so that a high static torque is generated. This enables the motor to start and stop quickly and exhibit a strong restoring torque when a displacement from a rest position occurs due to a load torque. Usually the terms holding torque and detent torque in relationship with static torque. The definitions for these are as follows

1. Holding torque. Defined as the maximum static torque that can be applied to the shaft of a motor excited with a rated current in a specific mode without causing continuous rotation.
2. Detent torque. Defined as the maximum static torque that can be applied to the shaft of an unexcited motor without causing continuous rotation.

In general, the larger the holding torque, the smaller the position error due to load. Ideally, the holding torque is independent of which phases are excited.

2.2.2.3 Noncumulative positioning error

Accuracy in positioning is an important factor which determines the quality of a stepping motor. Stepping motors are designed so that they rotate through a predetermined step angle in response to a pulse signal and come to rest at a precise position. Since the accuracy at no load depends on the physical accuracy of the rotor and stator, the positioning error is not accumulative.

With regards to the positions at which the rotor stops moving we have two concepts:

1. Rest position or equilibrium position. Defined as the position at which an excited motor comes to rest at no-load.
2. Detent position. Defined as the position at which a motor having a permanent magnet in its rotor comes to rest without excited at no-load.

There are two concepts of positioning error as follow

1. Step position error. Defined as the largest positive or negative static angular position error (compared with the rated step angle) which can occur when the rotor moves from one rest position to the next.
2. Positional accuracy. Defined as the largest angular position error of a rest position with respect to the nominal cumulative step angle which can occur during a full revolution of the rotor when moving from a reference rest position.

(Takashi Kenjo and Akira Sugawara, 1994)

2.3 Aspect of design

Some of the aspects of design have to take in account in order to carry out design optimization.

2.3.1 Geometry

Geometry of motor is an important consideration in designing. The first thing in design is to decide the geometry of motor. The type of motor that has been selected to design in this project is variable reluctance motor. The geometry is shown in Figure 2.5. It can be seen that it is double salient pole structure with 12 poles in stator and 8 poles in rotor. The stepper motor can be divided into several parts: stator, rotor, winding and shaft.

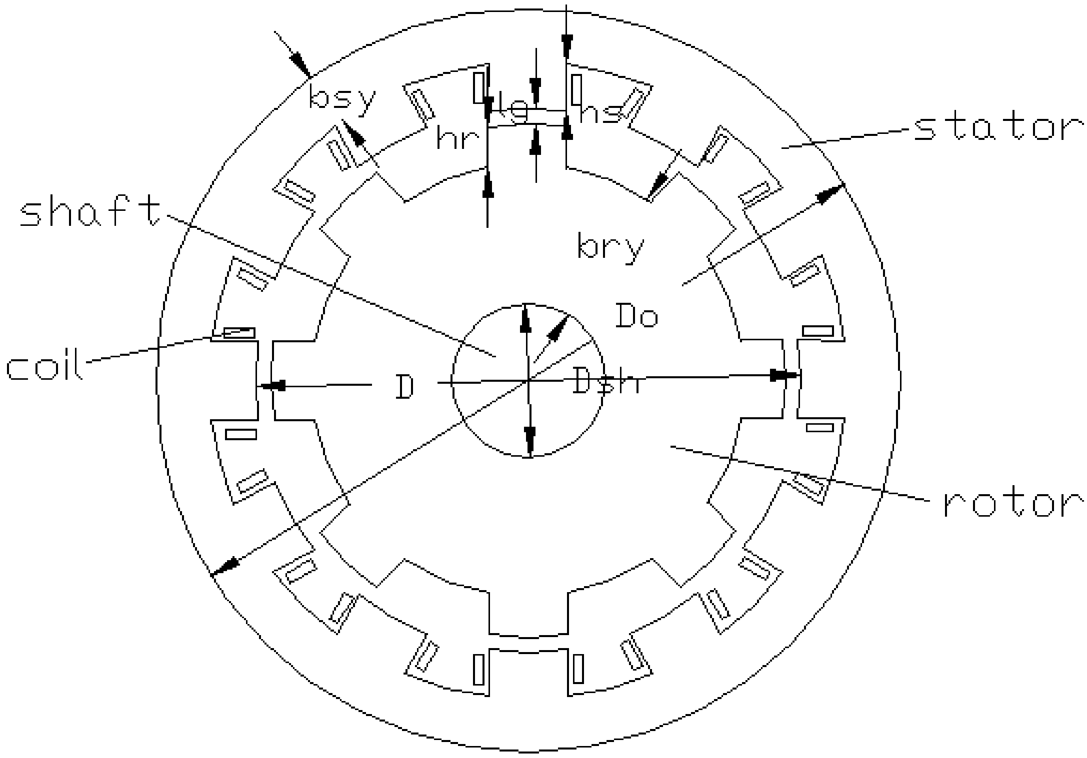

Figure 2.5 Cross section view of variable reluctance stepper motor

Different parts of motor need different material. It is because the characteristic of material will affect the performance of motor. Considering all the factors, the material used in stator and rotor is silicon steel, low carbon steel in shaft and copper in coil.

2.3.2 Tooth structure

The tooth structure of various stepping motor can be divided into three basic types. In the first type, as shown in Figure 2.6, there are the same number of teeth on the stator and rotor. In this structure all teeth are excited and de-energized at the same time. In the second type is illustrated in Figure 2.7 in which the numbers of teeth are different on the stator and rotor. In this machine not all of the teeth are excited at the same time. In the third type is illustrated in Figure 2.8. The stator teeth are arranged in groups on poles while rotor teeth are distributed homogeneously on the periphery.

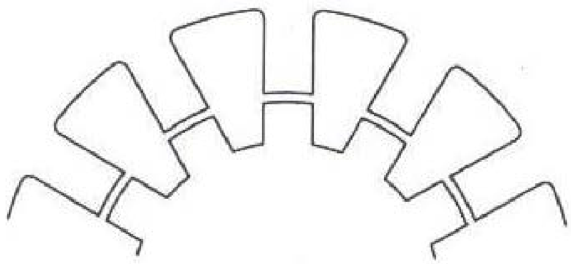

Figure 2.6 Toothed structures having the same tooth pitch on rotor and stator

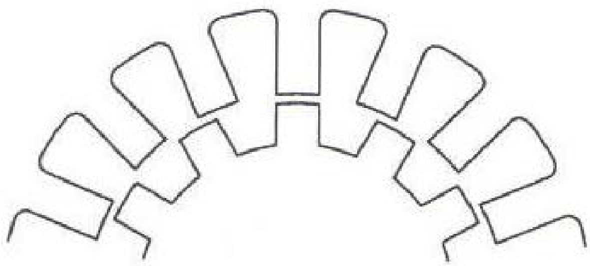

Figure 2.7 Toothed structure having different tooth pitches on rotor and stator

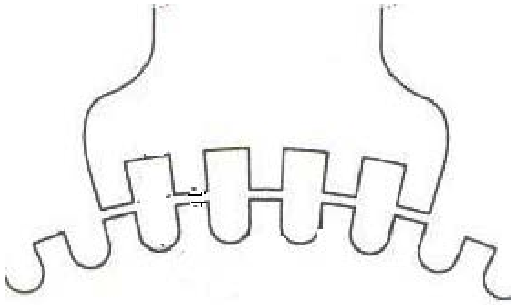

Figure 2.8 Toothed structures having a group of teeth at stator and rotor teeth are distributed on periphery.

(Takashi Kenjo and Akira Sugawara, 1994)

2.3.3 Winding connection

There are three alternative methods of connecting winding and is shown in Figure 2.9. Although the rated pole winding current depends only on the acceptable temperature rise, the corresponding rated phase current also depends on the interconnection as shown in Table 2.1. The rated phase voltage is the voltage which must be applied at the phase terminals to circulate the rated current in the windings. For the series connection the phase current is low and the voltage high compared to the parallel connection but there is no difference in the power supplied to the phase. (Paul Acarnley, 2003)

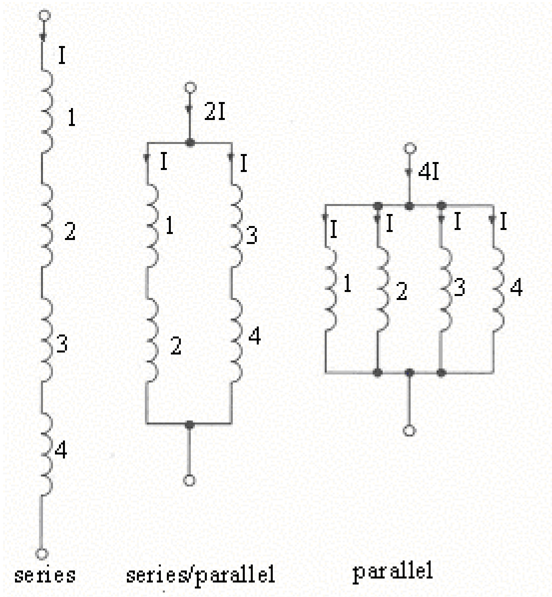

Figure 2.9 Interconnection of pole windings

Table 2.1 Effect of winding connection on ratings

Connection	Rated current	Resistance	Rated voltage	Power
Series	I	$4r$	$4rI$	$4rI^2$
Series/parallel	$2I$	r	$2rI$	$4rI^2$
Parallel	$4I$	$r/4$	rI	$4rI^2$

2.4 Losses

The relationship of input power, mechanical output, various losses and the parts which cause losses of stepper motor is shown in Figure 2.10. (K. Watanabe et. al, 1996)

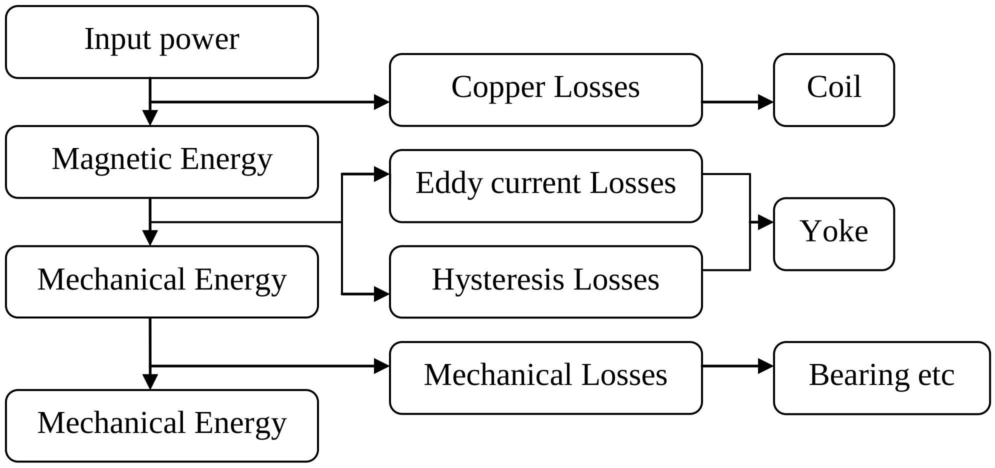

Figure 2.10 Input power, mechanical output and losses of stepper motor

2.4.1 Core losses

Ultimately, core losses contribute the most out of the total losses. Iron losses occurred in the stator and rotor both at yoke and pole which are further divided into hysteresis loss and eddy current loss. These are expressed as

$$P_{fe} = k_h.f.B^s + k_e.f^2B^2 \qquad (2.3)$$

2.4.1.1 Hysteresis losses

At high flux densities and relatively low frequencies, hysteresis losses are dominant. Hysteresis loss is the amount the magnetization of the ferrite material lags the magnetizing force because of molecular friction. The loss of energy due to hysteresis loss is proportional to the area of the static or low frequency B-H loop. The hysteresis loss is represented as

$$P_h = K_h\, B_{max}{}^s\, f \qquad (2.4)$$

the coefficient s is called the Steinmetz coefficient (after Charles Proteus Steinmetz, who discovered the relationship). It varies from about 1.5 to 2.5, depending on the material; 1.6 is often used. K_h is constant whose value depends on the ferromagnetic material.

2.4.1.2 Eddy current losses

At high frequencies, eddy current losses usually dominate. Eddy current loss is from a varying induction that produces electromotive forces, which cause a current to circulate within a magnetic material. The eddy current loss is commonly expressed in classical way as

$$P_e = K_e B_{max}^2 f^2 \tag{2.4}$$

where K_e is a constant whose value depends on the type of material and its lamination thickness. With the silicon content of 0.5% in the used silicon steel, the value of K_e is 0.004 and 0.0064 for core and teeth, respectively.

2.5 Copper losses

Motor windings have some internal resistance (R). This resistance is a function of wire diameter, the number of turns and the resistivity of the winding material. This resistance is ultimately what determines the maximum current that a winding should be subjected to. The power loss of a winding is given by

$$P_{cu} = I^2 R.q \tag{2.5}$$

2.6 Mechanical losses

The mechanical losses in a stepper motor are the losses associated with mechanical effects. Mechanical losses are mainly caused by bearing friction. Friction losses are losses caused by the friction of the bearings in the motor. The friction losses for a low power and tiny motor are very small. So, it is neglected for this design.
(P. C. Sen, 1997)

CHAPTER 3

OPTIMIZATION APPROACH OF GENETIC ALGORITHMS

3.1 Introduction

In chapter 1, the background and the idea of applying GA in solving nonlinear optimization problem is presented. In order to apply GA to the optimization problem, the library of genetic algorithms objects written in programming language C++ will be used and the library includes tools for using GA. The current version of Galib is available at (Matthew Wall,1996)

In this chapter, the methodology and how GA work in finding the optimal solution of a problem is presented. First of all, the history of Genetic Algorithms is presented. Then the overview of GA in searching the appropriate optimum solutions will be discussed.

3.2 Evolutionary algorithms and search types

According to (Heitkoetter, 1994): "Evolutionary algorithm is an umbrella term used to describe computer-based problem solving systems which use computational models of evolutionary processes as key elements in their design and implementation. A variety of evolutionary algorithms have been proposed. The major ones are:

- Genetic algorithms
- Evolution programming
- Evolution strategy
- Classifier system
- Genetic programming

They all share a common conceptual base of simulating the evolution of individual structures via processes of selection, mutation and reproduction. The processes depend on the perceived performance of the individual structures as defined by an environment. More precisely, evolutionary algorithms maintain a population of structures, that evolve according to rules of selection and other operators, that are referred to as "search operators", (or genetic operators), such as recombination and mutation. Each individual in the population receives a measure of it's fitness in the environment. Reproduction focuses attention on high fitness individuals, thus exploiting the available fitness information. Recombination and mutation perturb those individuals, providing general heuristics for exploration. Although simplistic from a biologist's viewpoint, these algorithms are sufficiently complex to provide robust (good performance across a variety of problem types) and powerful adaptive search mechanisms."

The current literature identifies three main types of search methods (Goldberg, 1989):"

- Calculus-based
- Enumerative
- Random

3.2.1 Calculus-based

Calculus-based methods subdivide into two main classes: indirect and direct. Indirect methods seek local extreme by solving the usually non-linear set of equations resulting from setting the gradient of the objective function equal to zero, whereas direct methods seek local optima by hopping on the function and moving in a direction related to the local gradient. This is simply the notion of hill-climbing: to find the local best, climb the functions in the steepest permissible direction. While both of these calculus-based methods have been improved, extended, hashed and rehashed, they show lack of robustness because both methods are local in scope (the optima they seek are the best in the neighborhood of a current point) and they depend upon the existence of derivatives (well-defined slope values). This last problem can be overcome for certain applications using different techniques but implementations of these techniques are difficult.

3.2.2 Enumerative

Enumerative methods are search algorithms that start looking within a finite search space, or a discredited infinite search space, at objective function values at every point in the space, one at a time. Although the algorithm looks attractive due to its simplicity, many practical spaces are simply too large to search one at a time and still have a chance of using the information to some practical end: it lacks efficiency.

3.2.3 Random

Random search algorithms walk along randomly chosen points within the search space and simply choose the point with the best value of the objective function. They have achieved increasing popularity as they are easier to program than calculus-based methods. However, in the long run, random searches can be expected to do no better than enumerative schemes. Monte Carlo methods (random search for a limited period) are well-known random search algorithms.

Random search methods should be separated from randomized techniques. Randomized search does not necessarily imply directionless search. The genetic algorithm is an example of a search procedure that uses random choice as a tool to guide a highly exploitative search through a coding of a parameter space."

3.3 History of genetic algorithms

Genetic algorithms originated from the studies of cellular automata, conducted by John Holland and his colleagues at the University of Michigan. Holland’s book (Holland, 1975), published in 1975, is generally acknowledged as the beginning of the research of genetic algorithms. Until the early 1980s, the research in genetic algorithms was mainly theoretical with few real applications. This period is marked by ample work with fixed length binary representation in the domain of function optimization by, among others, De Jong and Hollstien. Hollstien's work provides a careful and detailed analysis of the effect that different selection and mating strategies have on the performance of a genetic algorithm. De Jong's work attempted to capture the features of the adaptive mechanisms in the family of genetic algorithms that constitute a robust search procedure.

From the early 1980s the community of genetic algorithms has experienced an abundance of applications which spread across a large range of disciplines. Each and every additional application gave a new perspective to the theory. Furthermore, in the process of improving performance as much as possible via tuning and specializing the genetic algorithm operators, new and important findings regarding the generality, robustness and applicability of genetic algorithms became available.

Following the last couple of years of furious development of genetic algorithms in the sciences, engineering and the business world, these algorithms in various guises have now been successfully applied to optimization problems, scheduling, data fitting and clustering, trend spotting and path finding. (Goldberg, 1989)

3.4 What are genetic algorithms

The genetic algorithm is a model of machine learning which derives its behaviors from a metaphor of the processes of evolution in nature. This is done by the creation within a machine of a population of individuals represented by chromosomes, in essence a set of character strings that are analogous to the base-4 chromosomes that we see in our own DNA. The individuals in the population then go through a process of evolution which is, according to Darwin, made up of the principles of mutation and selection; however, the modern biological evolution theory also knows crossover and isolation mechanisms to improve the adaptive ness of the living beings to their environments.

With genetic algorithms, elements or chunks of elements are swapped between individuals as if by sexual combination and reproduction (crossover), others are changed at random (mutation). New generations appear from clones of the current population, in proportion to their fitness: a single objective function of the parameters that returns a numerical value, to distinguish between good and bad solutions. Fitness is then used to apply selection pressure to the population in a ‘Darwin’ fashion (survival of the fittest).

GAs differ from more normal optimization and search procedures in four ways:

- GAs work with a coding of the parameter set, not the parameters themselves.
- GAs search from a population of points, not a single point.
- GAs use payoff (objective function) information, not derivatives or other auxiliary knowledge.
- GAs use probabilistic transition rules, not deterministic rules.

Genetic algorithms require the natural parameter set of the optimization problem to be coded as a finite-length string (analogous to chromosomes in biological systems) containing characters, features or detectors (analogous to genes), taken from some finite-length alphabet. Usually, the binary alphabet that consists of only 0 and 1 is taken. Each feature takes on different values (alleles) and may be located at different positions (loci). The total package of strings is called a structure or population (or, genotype in biological systems). A summary of the correspondence between natural and artificial terminology is given in Table 3.1.

Table 3.1 Comparison of Natural and GA Terminology

Natural	**Genetic Algorithms**
Chromosome	String
Gene	Feature, character or detector
Allele	Feature value
Locus	String position
Genotype	Structure or population
Phenotype	Parameter set, alternative solution, a decoded structure

3.5 GA Fundamental and technology

In nature all living organisms contain a set of genetic data, termed a "genome". Somehow, this genetic data encodes all of the physical characteristics of the organism, known as the "phenome". A particular set of genetic information is a "genotype", and likewise a particular set of physical characteristics, or "traits", is a "phenotype". There may, or may not, be a direct one-to-one mapping of genotypes to phenotypes. These physical characteristics determine how well suited to its environment a particular organism is. The suitability of a given organism to its environment is usually measured as its "fitness". Computationally, it is usual to evaluate the "fitness" of an organism directly, without considering any kind of phenome.

The idea of "Survival of the fittest", as introduced by Darwin is well known. Where in nature the "fitness" relates to the ability of the organism to survive and to reproduce, in genetic algorithms the "fitness" is the evaluated result of some "objective function". Organisms with a better "fitness" score are more likely to be selected for reproduction either through some mechanism of competition to mate, or as a result of the least fit organisms dying. In this way genes which encode beneficial characteristics are propagated through subsequent generations of the population at the expense of genes which encode detrimental characteristics.

The genome is stored in the form of DNA in nature. On computers there is more flexibility, but it is common that the main factor controlling whether a particular search problem is successfully resolved, or not, depends upon how the problem was encoded into the genome. In John Holland's original work, a single binary "bit string" was used. This had computational advantages in terms of efficiency, but with modern computer hardware the balance between raw efficiency and applicability to a given task has swung toward the latter. Many different methods for encoding the genetic information are in common use today; Tree encoding, real-valued arrays, permutations, Gray encoding and so on.

3.5.1 Fitness function

Putative solutions to the target problem are evaluated using "Fitness functions", otherwise known as "Objective functions". Based upon the result of such functions, evolutionary pressures may be applied to a set of solutions. The objective function will obviously be problem specific, but there are certain features which should be avoided for the effective application of a GA. Such unfavorable objective functions are discussed below, but often the problems may be alleviated by choosing a different encoding scheme, by normalizing the input parameters, or by rebasing the function. An advantage of GAs over many search or optimization algorithms is that derivatives of this function are not required. This fact ensures that GAs may be readily applied on fitness landscapes (or potential surfaces) which contain discontinuities or singularities without any special treatments.

3.5.2 Selection

GA selection operators perform the equivalent role to natural selection. The overall effect is to bias the gene set in following generations to those genes which belong to the most fit individuals in the current generation. There are numerous selection schemes described in the literature; "Roulette wheel" selection, tournament selection, random selection, stochastic sampling. These, in essence, mimic the processes involved in natural selection.

3.5.3 Crossover

Crossover is a two –step process that involves mating and swapping of partial strings. Each time the crossover operator takes action, two randomly selected strings from the mating pool are mated. Then in the case of simple crossover, a position along one string is selected at random and all binary digits following the position are swapped with the second string. The result is two entirely new strings that move on to the next generation. It can be shown at Figure 3.1.

3.5.4 Mutation

The exact purpose of the mutation operations depends upon who you talk to. Mutations enable the GA to maintain diversity whilst also introducing some random search behavior. As for crossover, many types of mutation operator may be conceived depending upon the details of the problem and the chromosomal representation of solutions to that problem.

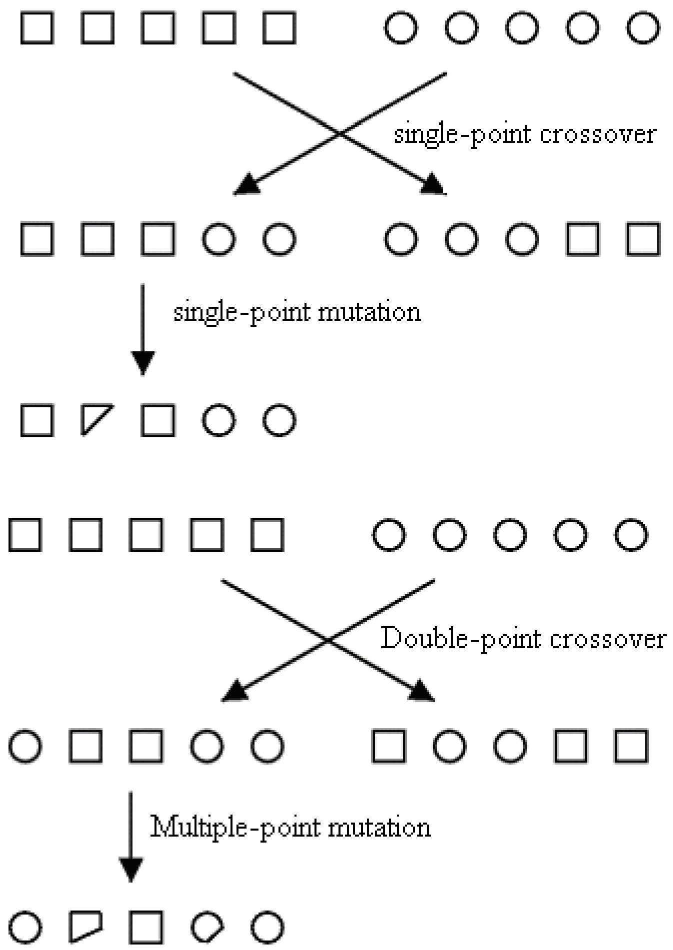

Figure 3.1 Diagram show how crossover and mutation occurred