A Framework for Multi-objective Optimization and Multi-criteria Decision Making for Design of Electrical Drives

A Framework for Multi-objective Optimization and Multi-criteria Decision Making for Design of Electrical Drives

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. Prof.dr.ir,J.T.Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op donderdag 2 oktober 2008 om 13.30 uur
door

Praveen Kumar

Master of Technology, IIT Delhi
geboren te Vill. Barkali, India

Dit proefschrift is goedgekeurd door de promotor:
Prof Dr. J.A.Fereira
Co-promotor:
Dr.ir P.Bauer

Samenstelling promotiecommissie:

Rector Magnificus, voorzitter instellingen
Prof. Dr. J.A.Fereira, Technische Universiteit Delft, promotor
Dr. P.Bauer, Technische Universiteit Delft, co-promotor
Prof. P.P.J.van den Bosch, Technische Universiteit Eindhoven
Prof. F.B.J. Leferink, Universiteit Twente
Prof. L.van der Sluis, Technische Universiteit Delft
Prof. B.Koren, Technische Universiteit Delft
Dr. P.J.van Duijsen, Simulation Research

ISBN ....

Table of contents

Chapter 1

Introduction

1.1 Introduction

All the products that we use and see in our daily lives are the results of one or the other engineering disciplines. The optimal design of these products is a multidisciplinary task and usually it is an iterative task performed by a team of skilled and experienced engineers. Therefore *design* is widely considered to be the central or distinguishing activity of engineering [1]. The *engineering design* is a constantly evolving discipline and design engineers are constantly trying to identify means of producing a better product in a shorter period of time. Design of a product classified as a complex system poses substantial challenges to both analysis and design [2]. Broadly speaking, in *engineering design* one attempts to improve or optimise several objectives, frequently competing and conflicting with each other, subject to satisfying a set of design and physical constraints. The problem solution involves two primary elements: formulation of the problem as a mathematical model that is suitable for optimisation and then defining a procedure for finding optimal solution of the problem once it has been formulated. To combine the mathematical model of the system to be designed with the optimisation algorithm for concrete engineering problems, it is necessary to clearly demarcate the boundaries of the engineering system to be designed and optimised. The development of the appropriate model of the system is a very challenging task because the models of the system to be designed have to represent the reality as closely as possible and also should not be computationally intensive. Invariably a detailed model of the system tends to give more accurate results but is computationally intensive whereas a simplified model does not represent the system so accurately but is computationally inexpensive and is suitable for optimisation algorithms. Besides modelling and optimisation methods, the other important aspect of engineering design is decision making because there are parts of the design process that require human or un-quantifiable judgement that is not suited for automation such as manufacturability.

A systematic design methodology is required to design and optimise the engineering systems and this is the main focus of this thesis. In this thesis a new design methodology is presented in order to systematise the process of design of engineering systems using modelling, optimisation and decision making techniques.

1.2 Engineering System Design

Phadke [3] gives the following definition of engineering design and its objectives: "*The objective of engineering design, a major part of research and development (R&D) is to produce drawings, specifications, and other relevant information needed to manufacture products that meet customer requirements*".

According to Pahl & Beitz [4]: *"The main task of engineers is to apply their scientific and engineering knowledge to the solutions of the technical problems, and then to optimise those solutions within the requirements and constraints set by material, technological, economical, legal, environmental and human-related considerations. Problems become concrete tasks after the clarification and definition of the problems which engineers have to solve to create new technical products (artefacts)"*.

Several definitions of *engineering systems* exist in the literature [4-7]. A system is a set of interrelated components intended to achieve a common objective. The system is also characterised by an interface with the surrounding environment thus creating inputs and outputs to the system. The design of *engineering systems* arises in engineering design projects that require the consideration of several disciplinary analyses [8-10], for example design of electrical drives involves expertise in motor design, power electronics, control systems, mechanical engineering, material science, etc. The process of design can be broadly classified into following steps [4]:

1. **Conceptualising**: In this stage different solution principles are investigated
2. **Embodying:** The solution principles obtained in step 1 above are engineered by determining the general arrangement and preliminary shapes and materials of all the components pertaining to the system under consideration
3. **Detailing:** The production and operational details are laid out
4. **Computing:** This includes drawing and information collection. These occur during all phases of the design process

A common model for *engineering system* design is ***phase type*** model [4, 7, 11]. The phase model is a top-down iterative process. The various steps involved in ***phase type*** model are:

1. **Conceptual Design:** In this step the essential problems are identified and suitable working principles are sought for. Having identified the problems and the working principles the basic solution path is laid down.

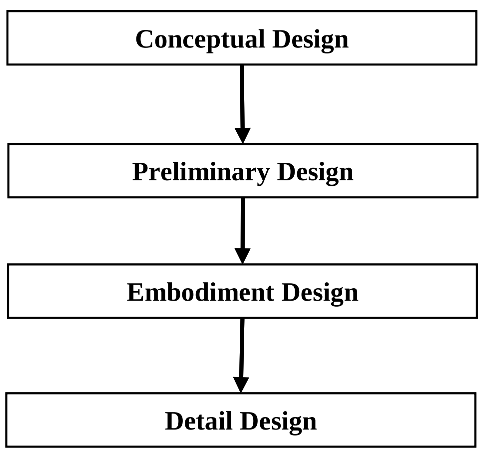
2. **Preliminary Design:** The preliminary design is obtained by refining the conceptual design and ranking them against the design specifications and choosing the best preliminary design [12].

3. **Embodiment Design:** Here the preliminary design obtained in step2 is elaborated, taking into consideration technical and economic criteria, to the point where subsequent detail design can lead directly to production [4].

4. **Detailed Design:** In this part of the design process the embodiment design is refined and final layout, forms, dimensions and surface properties of all the individual components, the definitive selection of materials and a final scrutiny of the production methods, operating procedure and cost are determined [4].

The ***phase type*** design process in its most general framework is shown in Figure 1.1. However, the design process is seldom straightforward and in a majority of cases it is highly iterative [7, 13-15] and a large number of iterations are required before the final design is achieved. An iterative ***phase type*** design process as proposed by Roozenburg and Eekels [14] is outlined in Figure 1.2. According to them, the iterative part consists of analysis, synthesis, simulation, evaluation and decision. For each provisional design the expected properties are compared to the criteria. If the design does not meet the criteria it is modified and evaluated again in the search for the best possible design. Hence it can be seen that design is essentially an optimisation process, as stated already by Simon [1].

The classification of engineering design into four steps in ***phase type*** design is not unique. Some authors make distinction between embodiment design and detailed design [4], some others make difference between conceptual and preliminary designs. On the more general level, design process consists of a loop shown in Figure 1.3 [16].



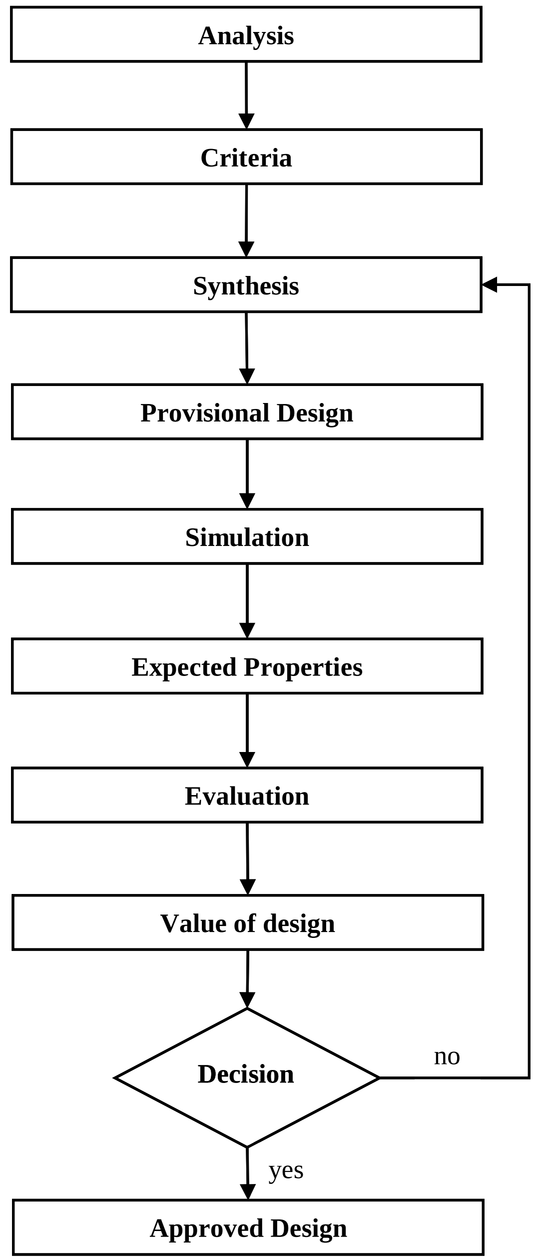


Figure 1.1: Simplified model of design process

Figure 1.2: Roozenburg's design cycle

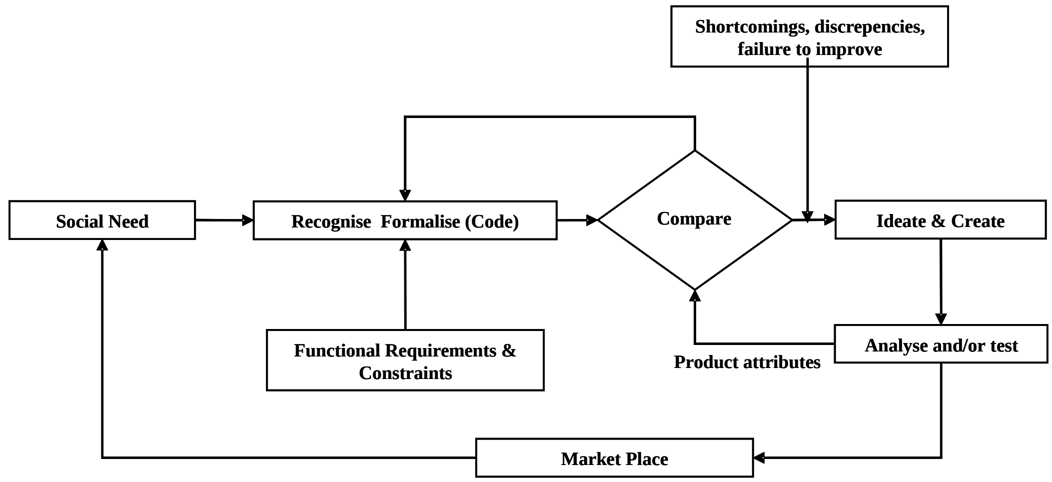


Figure 1.3: Design loop of Bahrami and Dagle

The interdisciplinary nature of *engineering systems* design poses challenges associated with computational burdens. There is a need to modify design methods that can model different degrees of collaboration and help to resolve the conflicts between different disciplines. In some cases simplifying assumptions can be made with reasonable accuracy. In other situations the interactions between different disciplines themselves may produce changes in the system's response. In such circumstances it is necessary to consider multiple disciplines in order to accurately evaluate the performance of a system. For example in the design of permanent magnet brushless direct current (BLDC) motor drives if the motor is designed purely on the basis of its magnetic circuit then it is possible that the motor may not deliver the required torque when connected to a voltage source inverter (VSI) due to high electrical time constant. Hence in order to obtain a proper design the interaction between the BLDC motor and VSI must be taken into consideration.

In general due to the interdisciplinary nature of *engineering systems*, the design process is characterised by the following ingredients [17]:

1. each discipline contributes to describe the overall problem
2. each discipline represents an independent problem with its own formulation. This formulation and its solution rely on theoretical results (e.g. optimality conditions, sensitivity analysis, convergence analysis) and solution techniques

3. the independent formulation of different disciplines requires a suitable unification, to form an overall *engineering system* formulation, of simulation and modelling techniques.

1.3 Challenges in engineering system design

Some of the challenges involved in design of engineering systems are [18, 19]:

1. Identification of functions that are objectives and constraints. The difference between them is blurred and some functions will move from objectives to constraints or vice versa. Some constraints are hard (equality type), some not; some will change or disappear while the others may be introduced as the problem knowledge base expands.
2. In many cases the variable ranges are also fuzzy and flexible and there is a requirement for exploration outside of the default regions. The reason is that the real bounds and limits are not always known from the very beginning and could be rather artificial limitations.
3. A set of results is required which the design engineer can analyse off-line, i.e. the design engineer should be able to import these results to some other programmes or to consult some database or persons for different aspects of given solutions.

Furthermore the multidisciplinary nature of the engineering system requires considering the design objectives and constraints from different disciplines concurrently in order to reduce the design time. The consideration of objectives and constraints from different disciplines requires developing the model of the entire system that can be very complicated requiring extensive computing time. To design such a system it is necessary to perform optimisation based on certain objectives and subject to certain constraints. The optimisation of the engineering system based of complex and computationally costly models may not be feasible because the multiobjective optimisation requires the model to be executed a large number of times. Moreover proper choice of constraints and objectives is crucial to the outcome of the optimisation. Often the objectives and constraints are not very clearly defined in the early stages of the design and complex models of the system will make it even more difficult to identify independent objectives and constraints.

1.4 Aim of the thesis

The primary aim of this thesis is to present a *design methodology* of engineering systems that supports the involvement of modelling and simulation of the system, multiobjective optimisation and multicriteria decision making. The first aim is to present a framework where modelling and optimisation is employed to accelerate and improve the design of complex engineering systems.

The second aim is to support the formulation of the optimisation problem by the selection of optimisation parameters, selection and formulation of the objectives and constraints. The design of complex engineering system is multiobjective in nature and hence the design problem is formulated as a multiobjective optimisation problem. Therefore, the other aim is to develop a reliable multiobjective optimisation algorithm.

Since multiobjective optimisation algorithm gives a set of feasible solutions, hence it becomes important to implement decision-making process to reduce the number of feasible solutions. To achieve this, multi-criteria decision making is presented and its application to engineering design problems is demonstrated in this thesis.

As an example, to demonstrate the application of different steps of the proposed **P**rogressive **D**esign **M**ethodology (PDM) to real engineering design problem, design of a BLDC motor drive is considered in this thesis. It will be shown using the example of BLDC motor drive that application of PDM to engineering problems leads to optimal solutions in a structured way.

1.5 Contribution

The main contributions of this thesis are:

1. A new design methodology for engineering systems is presented.
2. In order to achieve efficient multiobjective optimisation a new genetic algorithm, **N**on-dominated Sorting **B**iologically motivated **G**enetic **A**lgorithm (NBGA), is developed and presented
3. Detailed analytical models of BLDC motors and VSI are developed and presented. These models are developed to design the BLDC drive based on PDM.

1.6 Thesis Layout

This thesis is divided into two parts. The first part, chapters 2-5, is devoted to development of proposed design methodology. In the second part, chapters 6-9, of the thesis implementation issues of PDM are presented using an example of the design of a permanent magnet brushless direct current (BLDC) motor drive.

In chapter 2 the general framework of the proposed design methodology is presented. This methodology is called Progressive Design Methodology (PDM). The primary goal of the proposed methodology is to simplify and shorten the design process. In this chapter a framework is presented in which modelling, multiobjective optimisation and multi criteria decision making techniques are used to design an engineering system. Various steps of the proposed methodology are presented

One of the main aspects of PDM is multiobjective optimisation. In chapter 3 a formal definition of multiobjective optimisation problem is presented. A survey of different types of algorithms available for solving multiobjective optimisation problems is also given in chapter 3.

In chapter 4 a new multiobjective optimisation algorithm, Non-dominated Sorting Biologically Motivated Genetic Algorithm (NBGA) based on biological mutation operators is presented. The main purpose behind this approach was to improve the efficiency of genetic algorithms and to find widely distributed Pareto optimal solutions. This algorithm was tested on some benchmark test functions and compared with other GAs. It was observed that the introduction of these mutations does improve the genetic algorithms in terms of convergence and quality of solutions.

The various issues about modelling and the suitability of models for PDM are discussed in chapter 5. An important aspect in the success of PDM is the model of the system to be designed. At different stages of PDM different analyses need to be performed and hence different types of the model of the system to be designed are required.

In this thesis the application of PDM is explained using the example of design of a permanent magnet brushless DC (BLDC) motor drive. In order to design the drive the magnetic model and the dynamic model of the BLDC motor drive is required. These models have been developed keeping in view the issues discussed in chapter 5. The models of the motor that are used in the PDM are simple analytical magnetic model, detailed magnetic model and the dynamic model.

The details of the magnetic model of the BLDC motor are presented in chapter 6. In this chapter an analytical model for determining instantaneous air gap field density distribution is developed. This instantaneous field distribution can be further used to determine the cogging torque, induced back emf and iron losses in the motor. The advantage of analytical models is that they can be used for optimisation of BLDC motor as they are fast and can be conveniently integrated into PDM framework.

The dynamic model of the BLDC drive driven by a voltage source inverter (VSI) is developed in chapter 7. The model presented is valid for any shape of back emf and for both 120° and 180° modes of inverter operation. This model provides a rapid means of determining the drive performance in the initial design stages.

In chapter 8 the importance of system boundaries is highlighted. This is shown by a case study that demonstrates how misleading the design of a BLDC motor based purely on magnetic circuit could be.

In chapter 9 another case study is presented where all steps of PDM are applied to develop an optimal design of a BLDC motor. Finally, conclusions and future direction of research work are given in chapter 10.

1.7 References

[1] H. A. Simon, The Sciences of the Artificial, 3 ed: MIT Press, 1996.

[2] J. T. Allison, "Complex System Optimization: A Review of Analytical Target Cascading, Collaborative Optimization, and Other Formulations," in Mechanical Engineering: The University of Michigan, 2004.

[3] M. S. Phadke, Quality Engineering Using Robust Design. Prentice Hall International, 1989.

[4] G. Pahl and W. Beitz, Engineering Design- A Systematic Approach: Springer-Verlag, London, 1996.

[5] B. Blanchard and W. Fabrycky, Systems Engineering and Analysis: Prentice Hall, New Jersey, 1990.

[6] M. Bruns, Systemtechnik Methoden zur Interdiziplinären Systementwicklung: Springer-Verlag, Berlin, 1991.

[7] W. Hubka and E. Eder, Design Science: Springer-Verlag, London, 1996.

[8] J. S. Sobieszczanski and R. T. Haftka, "Multidisciplinary Design Optimization," Structural Optimization, vol. 14, pp. 1-23, 1997.

[9] K. Lewis and F. Mistree, "Collaboration, Sequential and Isolated Decision Design," ASME journal of Mechanical Design, vol. 120, pp. 643-652, 1998.

[10] R. J. Balling and J. S. Sobieszczanski, "Optimization of Coupled Systems: A Critical Overview of Approaches," AIAA, vol. 34, pp. 6-17, 1996.

[11] S. Pugh, Total Design: Addison-Wesly, 1991.

[12] C. L. Dym, Engineering Design: A Synthesis of Views: Cambridge University Press, 1994.

[13] R. P. Smith and S. D. Eppinger, "Identifying Controlling Features of Engineering Design Iteration," Management Science, vol. 43, pp. 276-293, 1997.

[14] N. Roozenburz and J. Eekels, Product Design: Fundamentals and Methods: John Wiley & Sons Inc, 1995.

[15] N. Cross, Engineering Design Method: John Wiley, 1994.

[16] A. Bahrami and C. H. Dagli, Design Science, 1994.

[17] E. F. Campana, G. Fasano, and D. Peri, "Issues on Nonlinear Programming for Multidisciplinary Design Optimization (MDO) in Ship Design Framework," presented at 8th Numerical Towing Tank Symposium (NuTTS 05), Vatna, Bulgaria, 2-4 October 2005.

[18] I. C. Parmee, "Exploring the design potential of evolutionary/adaptive search and other computational intelligence technologies," ACDM, pp. 27-42, 1998.

[19] I. C. Parmee, "Strategies for the integration of evolutionary/adaptive search with engineering design process," in Evolutionary Algorithms in Engineering Applications, D. Dasgupta and Z. Michalewicz, Eds.: Springer Verlag, 1997.

Chapter 2

Progressive Design Methodology

2.1 Introduction

The design of complex engineering systems involves many objectives and constraints and requires application of knowledge from several disciplines (multidisciplinary) of engineering [1-3]. The multidisciplinary nature of complex systems design presents challenges associated with modelling, simulation, computation time and integration of models from different disciplines. In order to simplify the design problems, assumptions based on the designer's understanding of the system are introduced. The ability and the experience of the designer usually lead to good but not necessarily an optimum design. Hence there is a need to introduce formal mathematical optimisation techniques, in design methodologies, to offer an organised and structured way to tackle design problems.

A review of different methods for design and optimisation of complex systems is given in [4-8]. The increase in complexity of systems as well as the number of design parameters needed to be co-ordinated with each other in an optimal way have led to the necessity of using mathematical modelling of systems and application of optimisation techniques. In this situation the designer focuses on working out an adequate mathematical model and the analysis of the results obtained while the optimisation algorithms choose the optimal parameters for the system being designed. Marczyk [9] presented stochastic simulation using the Monte Carlo technique as an alternative to traditional optimisation. In recent years probabilistic design analysis and optimisation methods have also been developed [10-12] to account for uncertainty and randomness through stochastic simulation and probabilistic analysis. Much work has been proposed to achieve high-fidelity design optimisation at reduced computational cost. Booker et. al. [13] developed a direct search Surrogate Based Optimisation (SBO) framework that converges to an objective function subject only to bounds on the design variables and it does not require derivative evaluation. Audet et. al. [14] extended that framework to handle general non-linear constraints using a filter for step acceptance [15].

The primary shortcoming of many existing design methodologies is that they tend to be hard coded, that is they are discipline or problem specific and have limited capabilities when it comes to incorporation of new technologies. There appears to be a need for a new methodology that can exploit different tools, strategies and techniques which strive to simplify the design cycle of engineering systems. The other drawback of the existing methodologies is that the designer needs extensive knowledge of the process itself. In order to overcome these problems a new design methodology, **P**rogressive **D**esign **M**ethodology (PDM), has been proposed. In the following sections the details of PDM are laid down.

2.2 Progressive Design Methodology

A design method is a scheme for organising reasoning steps and domain knowledge to construct a solution [16]. Design methodologies are concerned with the question of how to design whereas the design process is concerned with the question of what to design. A good design methodology has following characteristics [17]:

1. Takes less time and causes fewer failures
2. Produces better design
3. Works for a wide range of design requirements
4. Integrates different disciplines
5. Consumes less resources: time, money, expertise
6. Requires less information

An ideal condition in the design of an engineering system will be if all the objectives and constraints can be expressed by a simple model. However in practical design problems this is seldom the case due to the complexity of the system. Hence a trade-off has to be made between the complexity of the model and time to compute the model. A complex model will enable us to represent all the objectives and constraints of the system but will be computationally intensive. On the other hand a simple model will be computationally inexpensive but will limit the scope of objectives and constraints that can be expressed. In order to overcome this problem PDM consists of three main phases:

1. Synthesis phase of PDM
2. Intermediate analysis phase of PDM

3. Final design phase of PDM

Since in the first step (*synthesis phase*) of PDM the detailed knowledge is unavailable hence the optimisation process is exhaustive. If complex models are used in this phase then the computational burden will be overwhelming. In order to facilitate the initial optimisation process only those objectives and constraints are considered that can be expressed by simple mathematical models of the system. In the *synthesis phase* a set of feasible solutions (Pareto Optimal Solutions) is obtained. The principle of Pareto optimality and ways to determine the Pareto optimal solutions is presented in chapter 3. The Figure 2.1 illustrates a set of Pareto Optimal Solutions for a problem where two objectives (f_1 and f_2) are simultaneously minimised. The set of feasible solutions is obtained by using *multi objective optimisation*. Hence the engineering design problem is a *multi objective optimisation problem (MOOP)*. The primary purpose of the *synthesis phase* is to develop simple models of the system and translate the problem as to a multi-objective optimisation problem. The details of the *synthesis phase* are explained in section 2.3.

The most important task in engineering design problems, besides developing suitable mathematical models, is to generate various design alternatives and then to make preliminary decision to select a design or a set of designs that meets a set of criterion. Hence the engineering design problem is also a *multicriteria decision making* (*MCDM*) problem as well. In the conceptual stages of design, the design engineer faces the greatest uncertainty in the product attributes and requirements (e.g., dimensions, features, materials and performance). The evolution of design is greatly affected by decisions made during the conceptual stage and these decisions have a considerable impact on overall cost.

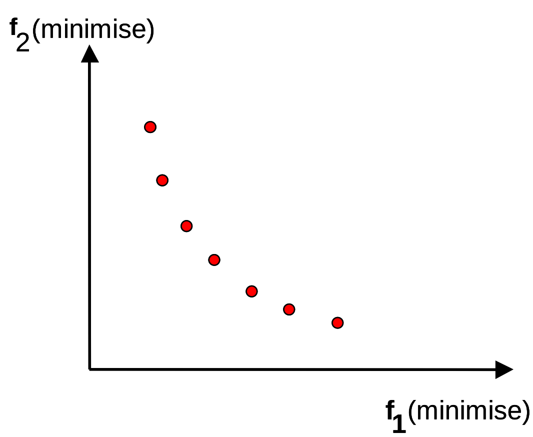


Figure 2.1: A set of Pareto optimal solutions for an optimisation problem with two objectives

In the *intermediate analysis phase multicriteria decision making* process is carried out. This step is a screening process where the set of solutions obtained from the *synthesis phase* is subjected to the process of screening. In order to achieve the screening additional constraints are taken into consideration. The constraints considered here are those that cannot be expressed explicitly in mathematical terms. The details of the *intermediate analysis phase* are given in section 2.4 of the present chapter.

In the *final design phase* detail model of the system is developed. After having executed the *synthesis phase* a better understanding of the system is obtained and it is possible to develop a detail model of the system. In this phase all the objectives and constraints that could not be considered in the *synthesis phase* are taken into consideration. In this phase exhaustive optimisation is not carried out, rather fine tuning of the variables is performed in order to satisfy all the objectives and constraints. The outline of the *final design phase* is given in section 2.5.

2.3 Synthesis Phase of PDM

In the ***synthesis phase*** the requirements of the system to be designed are identified. Based on these requirements system boundaries are defined and performance criterion/criteria are determined. The next step is to determine the independent design variables that will be changed during the optimisation process. The various steps involved in the ***synthesis phase*** are:

1. **System requirements analysis**
2. **Definition of system boundaries**
3. **Determination of performance criterion/criteria**
4. **Selection of variables and sensitivity analysis**
5. **Development of system model**
6. **Deciding on the optimisation strategy**

The implementation of the above steps is shown in Figure 2.2. From Figure 2.2 it can be seen that the six steps involved in the *synthesis phase* are not executed in purely sequential manner. After the sensitivity analysis has been done and a set of independent design variables (IDV) has been identified, the designer has to decide if the set of IDV obtained is

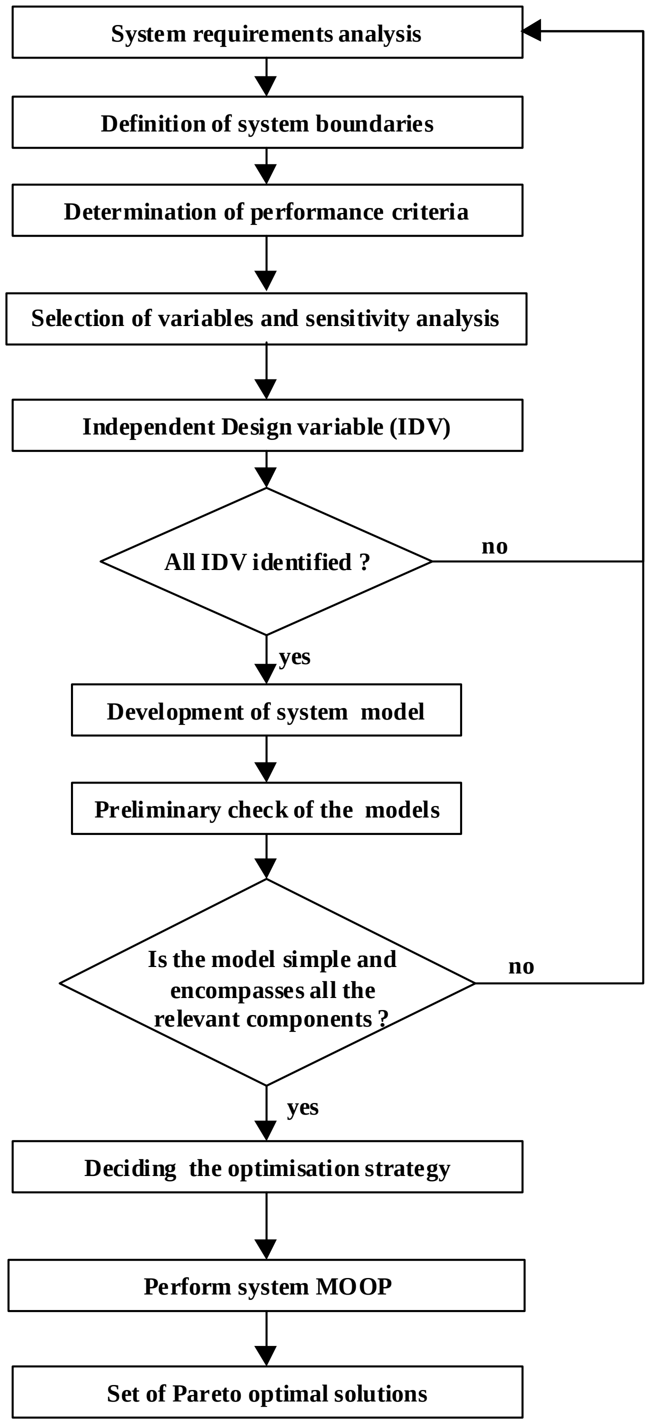


Figure 2.2: Steps in the synthesis phase of Progressive Design Methodology (PDM)

appropriate to proceed with the modelling process. The decision about the appropriateness of the set of IDV can be made based on previous experience or discussions with other experts. If the set of IDV is not sufficient then it is prudent to go back to system requirement analysis and perform the loop again. This loop can be repeated until a satisfactory set of IDV is identified. Similarly after the model of the system to be designed (target system) is developed, it is important to check if the model includes the system boundaries and the set of IDV. In reality the selection of variables and the development of the model has to be done iteratively since both depend on each other. The choice of variables has influence on modelling and the modelling process itself will influence of the variables needed. The details of each of the above steps are given in the following subsections.

2.3.1 System Requirements Analysis

The requirements of the system to be designed are analysed in this phase. The purpose of system requirement analysis is to develop a clear and detailed understanding of the needs that the system has to full fill. Hence this phase can be a challenging task since the requirements form the basis for all subsequent steps in the design process. The quality of the final product is highly dependent on the effectiveness of the requirement identification. The primary goal of this phase is to develop a detailed functional specification defining the full set of system capabilities to be implemented.

2.3.2 Definition of System Boundaries

Before attempting to optimise a system, the boundaries of the system to be designed should be identified and clearly defined. The definition of the clear system boundaries helps in the process of approximating the real system [18]. Since an engineering system consists of many subsystems it may be necessary to expand the system boundaries to include those subsystems that have a strong influence on the operation of the system that is to be designed. As the boundaries of the system increases, i.e. more the number of subsystems to be included, the complexity of the model increases. Hence it is prudent to decompose the complex system into smaller subsystems that can be dealt with individually. However care must be exercised while decomposing the system as too much decomposition may result in misleading simplifications of the reality. For example a brushless direct current (BLDC) motor drive system consists of three major subsystems viz.

1. The BLDC motor
2. Voltage source inverter (VSI)
3. Feedback control

Usually a BLDC motor is designed for a rated load, i.e. the motor is required to deliver a specified amount of torque at specified speed for continuous operation at a specified input voltage. During design process the motor is the primary system under design. However, optimised design of the motor based only on the magnetic circuit may result in misleading results. It is possible that this optimised motor has a high electrical time constant and the VSI is not able to provide sufficient current resulting in lower torque at rated speed and given input voltage. Hence, for the successful design of the BLDC motor it is important to include the VSI in the system, i.e. the boundary of the system is expanded. Of course, it is a different matter that the model of the system that includes the BLDC motor and the VSI is more complicated but nevertheless is closer to the reality.

2.3.3 Determination of Performance Criterion/Criteria

Once the proper boundaries of the system have been defined, performance criterion/criteria are determined. The criterion/criteria form the basis on which the performance of the system is evaluated so that the best design can be identified. In engineering design problems different types of criteria can be classified as depicted in Figure 2.3 [18]:

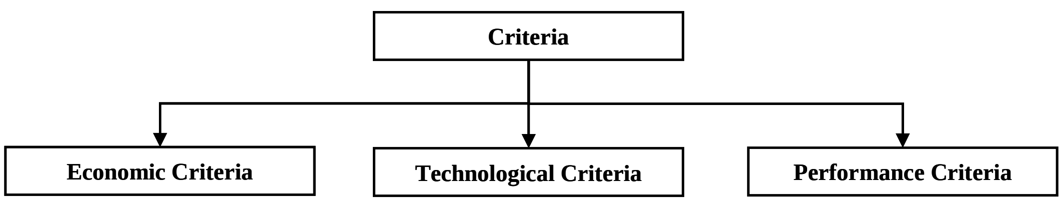


Figure 2.3: Classification of criterion

1. **Economic criterion/criteria:** In engineering system design problems the economic criterion involves total capital cost, annual cost, annual net profit, return on investment, cost-benefit ration or net present worth.
2. **Technological criterion/criteria:** The technological criterion involves production time, production rate, and manufacturability.

3. **Performance criterion/criteria:** Performance criterion is directly related to the performance of the engineering system such as torque, losses, speed, mass, etc.

In the synthesis phase of PDM the ***Performance criterion/criteria*** are taken into consideration because they can be expressed explicitly in the mathematical model of the system. The economic and technological criteria are suitable for ***Intermediate analysis*** and ***Final design phases*** of PDM because by then detailed knowledge about the engineering systems performance and dimensions are available.

2.3.4 Selection of Variables and Sensitivity Analysis

The next step is selection of variables that are adequate to characterise the possible candidate design. The design variables can be broadly classified as, Figure 2.4:

1. ***Engineering variables*****:** The engineering variables are specific to the system being designed. These are variables with which the designer deals.
2. ***Manufacturing variables*****:** These variables are specific to the manufacturing domain.
3. ***Price variables*****:** This variable is the price of the product or the system being designed.

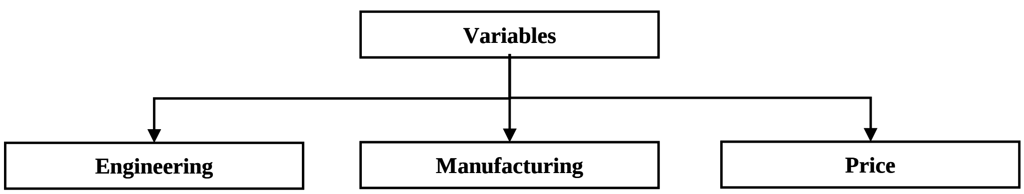


Figure 2.4: Classification of variables

In the *synthesis* phase of PDM *engineering* variables are considered. There are two factors to be taken into account while selecting the engineering variables. First it is important to include all the important variables that influence the operation of the system or affect the design. Second, it is important to consider the level of detail at which the model of the system is developed. While it is important to treat all the key *engineering* variables, it is equally important not to obscure the problem by the inclusion of a large number of finer details of secondary importance [18]. In order to select the proper set of variables, sensitivity analysis is performed. For sensitivity analysis all the *engineering* variables are considered and its influence on the objective parameters is considered. The sensitivity analysis enables to discard the *engineering* variables that have least influence on the objectives.

2.3.5 Development of System Model

A model is any incomplete representation of reality, an abstraction but could be close to reality. The purpose in developing a model is to answer a question or a set of questions. If the questions that the model has to answer, about the system under investigation, are specific then it is easier to develop a suitable and useful model. The models that have to answer a wide range of questions or generic questions are most difficult to develop. The most effective process for developing a model is to begin by defining the questions that the model should be able to answer. Broadly models can be classified into following categories [19], Figure 2.5:

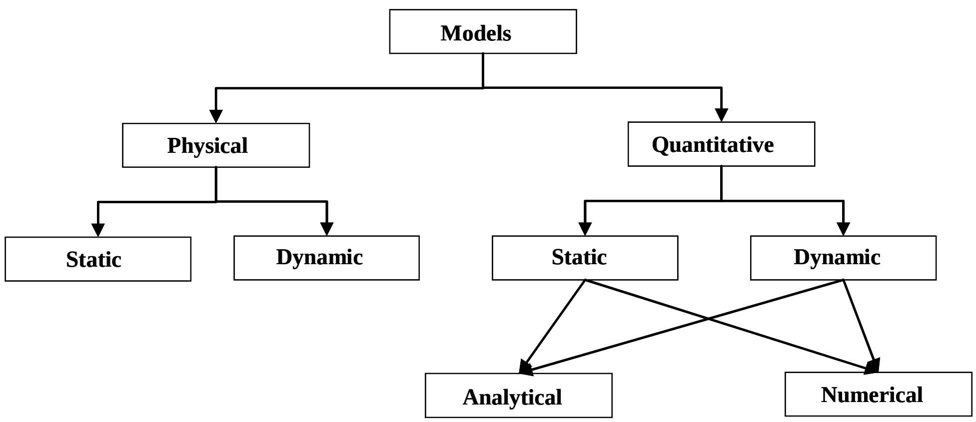


Figure 2.5: Classification of models

1. ***Physical models*****:** These models are full-scale mock-up, sub-scale mock-up or electronic mock up.
2. ***Quantitative models*****:** These models give numerical answers. These models can be either analytical, simulation or judgmental. These models can be dynamic or static. An analytical model is based on system of equations that can be solved to produce a set of closed form solutions. However finding exact solutions of analytical equations is not always feasible. Simulation models are used in situations where analytical models are difficult to develop or are not realistic. The main advantage of analytical models is that they are faster than numerical models and hence are suited for MOOP.

The major aspect of analytical model is that certain approximations are required to develop analytical models. However in certain cases where approximations cannot be made and a very deep insight of the system is required then numerical simulation methods such as Finite element method (FEM), Computational fluid dynamics (CFD), etc. have to be adopted. The main drawback of numerical models is that they are computationally intensive and are not suitable for exhaustive optimisation process.

2.3.6 Deciding on Optimising Strategy

Multi-objective optimisation results in a set of Pareto optimal solutions specifying the design variables and their objective tradeoffs. These solutions can be analysed to determine if there exist some common principles between the design variables and the objectives [20]. If a relation between the design variables and objectives exit they will be of great value to the system designers. This information will provide knowledge of how to design the system for a new application without resorting to solving a completely new optimisation problem again.

The principles of multi-objective optimisation are different from that of a single objective optimisation. When faced with only a single objective an optimal solution is one that minimises the objective subject to the constraints. However, in a multi-objective optimisation problem (MOOP) there are more than one objective functions and each of them may have a different individual optimal solution. Hence, many solutions exist for such problems. The MOOP can be solved in four different ways depending on when the decision maker articulates his preference concerning the different objectives [21]. The classification of the strategies is as follows, Figure 2.6:

1. ***Priori articulation of preference information*****:** In this method the DM gives his preference to the objectives before the actual optimisation is conducted. The objectives are aggregated into one single objective function. Some of the optimisation techniques that fall under this category are ***weighted-sum approach*** [22, 23], ***Non-Linear approaches*** [24], ***Utility theory*** [24, 25].
2. ***Progressive articulation of preference information*****:** In this method the DM indicates the preferences for the objectives as the search moves and the decision-maker learns more about the problem. In these methods the decision maker either

changes the weights in a ***weighted-sum approach*** [26], or by progressively reducing the search space as in the ***STEM method*** of reference [27]. The advantages of this method are that it is a learning process where the decision-maker gets a better understanding of the problem. Since the DM is actively involved in the search it is likely that the DM accepts the final solution. The main disadvantage of this method is that a great degree of effort is required from the DM during the entire search process. Moreover the solution depends on the preference of one DM and if the DM changes his/her preferences or if a new DM comes then the process has to restart.

3. ***Posteriori articulation of preference information*:** In this method the search space is scanned first and Pareto optimal solutions are identified. This set of Pareto optimal solution is finally presented to DM. The main advantage of this method is that the solutions are independent of DM's preferences. The process of optimisation is performed only once and Pareto optimal set does not change as long as the problem description remains unchanged. The disadvantage of this method is that they need large number of computations to be performed and the DM is presented with too many solutions to choose from.

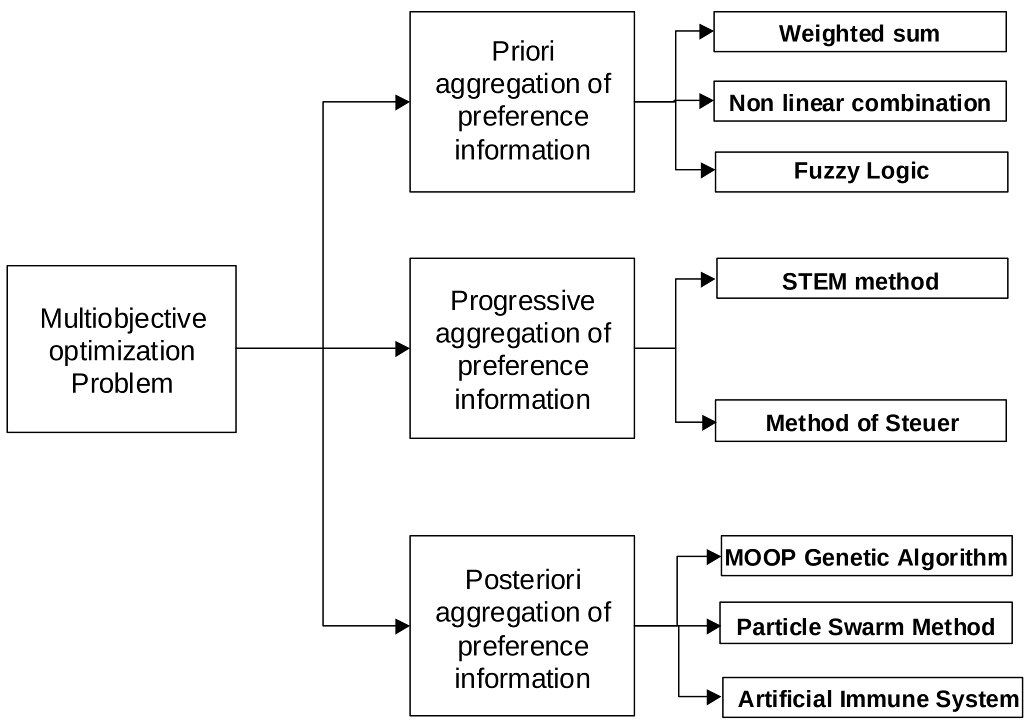


Figure 2.6: Classification of optimisation methods based on aggregation of information

The principle goal of multi-objective optimisation algorithms is to find well spread set of Pareto optimal solutions. Each of the solutions in the Pareto optimal set corresponds to the optimum solution of a composite problem trading-off different objective among the objectives. Hence each solution is important with respect to some trade-off relation between the objectives. However in real situations only one solution is to be implemented. Therefore, the question arises about how to choose among the multiple solutions. The choice may not be difficult to answer in the presence of many trade-off solutions, but is difficult to answer in the absence of any trade-off information. If a designer knows the exact trade-off among objective functions there is no need to find multiple solutions (Pareto optimal solutions) and a priori articulation methods will be well suited. However, a designer is seldom certain about the exact trade-off relation among the objectives. In such circumstance it is better to find a set of Pareto optimal solutions first and then choose one solution from the set by using additional higher level information about the system being designed. With this in view in **PDM** posteriori based optimisation method is used and a novel multiobjective genetic algorithm (**N**on-dominated sorting **B**iologically Motivated **G**enetic **A**lgorithm, NBGA) is developed. The details of the algorithm are given in chapter 4. Choosing a suitable solution from the Pareto optimal set forms the second phase of **PDM** and is described in the next section.

2.4 Intermediate Analysis Phase of PDM

The most important tasks in engineering design, besides modelling and simulation, are to generate various design alternatives and then to make preliminary decision to select a design or a set of designs that fulfils a set of criterion. Many systems and techniques have been developed to address the multicriteria decision making approach in engineering design problems. Some of the notable techniques are quality function deployment (QFD) [28], the analytical hierarchy approach [29], Pug Charts [30]. It is a general assumption that evaluation of a design on the basis of any individual criterion is a simple and straightforward process. However in practice, the determination of the individual criterion may require considerable engineering judgement [31]. In addition to these engineering decision methods, there is an extensive literature on multi- criteria decision making as shown in the survey of Costa and Vinke [32]. In the initial phase of development of an engineering system the details of a design are unknown and design description is still imprecise when the most important decisions are made [33]. In this initial engineering design phase, the final values of the design variables are uncertain [34]. The uncertainties in the design variables are not probabilistic and will be removed by further refinements of the models of the system and specifications later in the design process. Hence, at this stage decision making using fuzzy sets is appropriate [35]. In the initial stage of decision making the designers present their preferences for different values of design variables using fuzzy sets. Each value of design variable is assigned a preference between absolutely unacceptable and absolutely acceptable values. The values of design variables have discrete, continuous or linguistic preference values. Hence the designer's judgement and experience are formally included in the preliminary design problem. The general problem is thus a Multi Criteria Decision-Making problem (MCDM), where the designer has to choose the highest performing design configuration from the available set of design alternatives and each design is judged by several, even competing, performance criteria or variables.

A Multi Criteria Decision-Making problem (MCDM) is expressed as:

$$D = \begin{array}{c c} & \begin{array}{cccc} c_1 & c_2 & \cdots & c_n \end{array} \\ \begin{array}{c} A_1 \\ A_2 \\ \vdots \\ A_m \end{array} & \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} \end{array} \tag{2.1}$$

$$w = \left(w_1, w_2, \ldots w_n \right)$$

where D is the decision matrix, $A_i,\ i = 1, \ldots, m$ are the possible alternatives; $c_j,\ j = 1, \ldots, n$ are the criteria with which alternative performances are measured and x_{ij} is the performance score of the alternative A_i with respect to attribute c_j and w_j are the relative importance of attributes.

The alternative performance rating x_{ij} can be crisp, fuzzy, and/or linguistic. The linguistic approach is an approximation technique in which the performance ratings are represented as linguistic variable [36-38]. The classical MCDM problem consists of two phases:

1. an aggregation phase of the performance values with respect to all the criteria for obtaining a collective performance value for alternatives

2. an exploitation phase of the collective performance value for obtaining a rank ordering, sorting or choice among the alternatives.

The various parts of intermediate analysis phase of PDM are:

1. **Identification of new set of criteria**
2. **Linguistic term set**
3. **Semantic of linguistic term set**
4. **Aggregation operator for linguistic weighted information**

The flow chart of the above steps is shown in Figure 2.7.

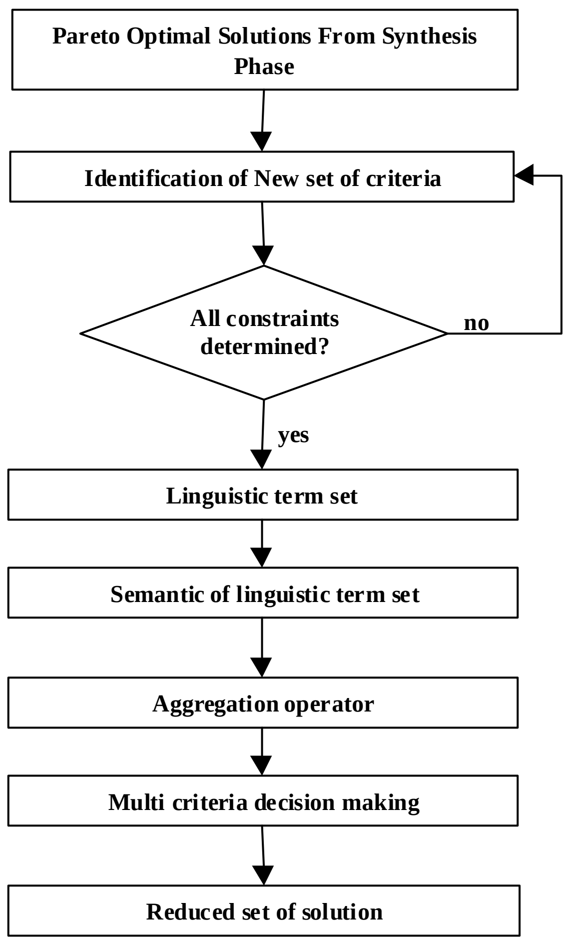


Figure 2.7: Steps in the intermediate analysis phase of PDM

2.4.1 Identification of new set of criteria

In the synthesis stage the constraints imposed on the system are engineering constraints. The *engineering constraints* are specific to the system being designed and can be considered as criteria based on which decision making is done. Besides *engineering constraints* there are other *non-engineering* constraints such as manufacturing limitations. It may be possible that certain Pareto optimal solutions obtained in the ***synthesis*** stage may not be feasible from the manufacturing point of view or may be too expensive to manufacture. Hence, in order to determine these constraints a high level of information is to be collected from various experts.

2.4.2 Linguistic term set

After determining all the constraints, the next step is to determine the *linguistic term set*. This phase consists of establishing the linguistic expression domain used to provide the linguistic performance values for an alternative according to different criteria. The first step in the solution of a MCDM problem is selection of linguistic variable set. The definition of a linguistic variable is as follows [37, 38]:

A linguistic variable is characterised by a quintuple $(L, H(L), U, G, M(x))$ in which L is the name of the variable; $H(L)$ denotes the term set of *L*, i.e. the set of names of linguistic values of *L*, with each value being fuzzy variable denoted generically by *X* and ranging across a universe of discourse *U* which is associated with the base variable *u*; *G* is a syntactic rule for generating the names of values of *L*; and *M* is a semantic rule for associating its meaning with each *L*, *M(X)*, which is a fuzzy subset of *U*.

There are two ways to choose the appropriate linguistic description of term set and their semantic [39]:

1. In the first case by means of a context-free grammar, and the semantic of linguistic terms is represented by fuzzy numbers described by membership functions based on parameters and a semantic rule [40, 41] . A context-free grammar (CFG) is a four-tuple [42] (V_N, V_T, I, P), where V_N is a finite, non-empty set of terminals, the alphabet, V_T is a finite, non-empty set of grammar variables (categories, or non-terminal symbols), $I \in V_N$ is the start symbol, P is a finite set of production rules,

each of the form $A \rightarrow \alpha$, where $A \in V$ and $\alpha \in \left(V_T \cup V_N\right)$. For a rule $A \rightarrow \alpha$, A is the head of the rule and α is its body. The generated language must be easy to understand and does not have to be infinite [43] but large enough to include any possible situation.

2. In the second case the linguistic term set by means of an ordered structure of linguistic terms, and the semantic of linguistic terms is derived from their own ordered structure which may be either symmetrically/asymmetrically distributed on the (0,1) scale. An example of a set of seven terms of ordered structured linguistic terms is as follows:

$$S = \left\{s_0 = none,\ s_1 = very\ low, s_2 = low,\ s_3 = medium,\ s_4 = high,\ s_5 = very\ high,\ s_6 = perfect\right\}$$

The linguistic term set in addition satisfy the following conditions:

Negation operator: $Neg(s_i) = s_j,\ j = T - i$ ($T+1$ is the cardinality)

Maximisation operator: $Max(s_i, s_j) = s_i$, if $s_i \geq s_j$

Minimisation operator: $Min(s_i, s_j) = s_i$, if $s_i \leq s_j$

where *i* and *j* lie within the cardinality of the term set (T+1)

2.4.3 The Semantic of Linguistic Term Set

The semantics of the linguistic term set can be broadly classified into three categories Figure 2.8:

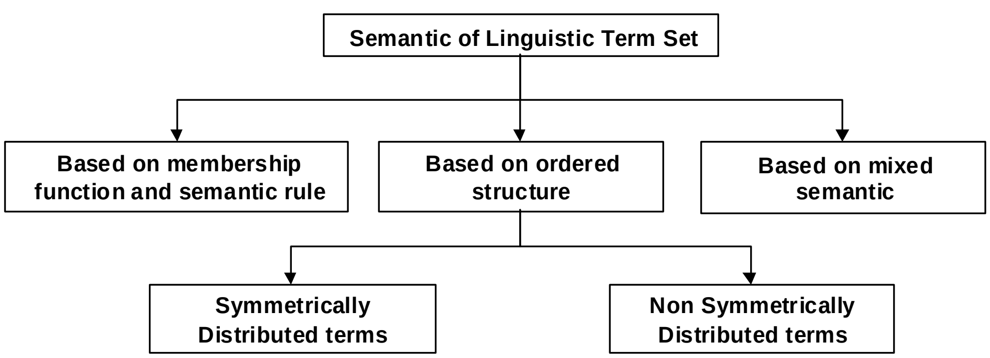


Figure 2.8: Classification of semantic of linguistic term set

1. **Semantic based on membership functions and a semantic rule**: Under this semantic the meaning of each linguistic term is given by a fuzzy subset defined on the interval [0,1], which are usually described by membership functions [44, 45]. Because the linguistic assessments given by the users are approximate, linear trapezoidal membership functions are enough in capturing the vagueness of the assessments [46]. The parametric representation of the membership functions is achieved by a 4-tuple $\left(a_i, b_i, \alpha_i, \beta_i\right)$, where the first two parameters represent the interval in which the membership value is 1; the third and fourth parameters indicate the left and right width [41]. An example of a set of nine terms is as follows [44], Figure 2.9:

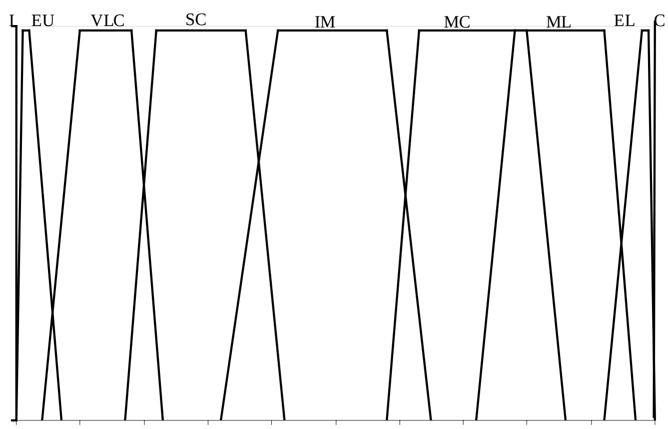


Figure 2.9: A set of Nine Terms with its Semantic

$$
\begin{aligned}
&C = Certain = (1, 1, 0, 0)\\
&EL = Extremely\_Likely = (0.98, 0.99, 0.05, 0.01)\\
&ML = Most\_Likely = (0.78, 0.92, 0.06, 0.05)\\
&MC = Meaningful\_Chance = (0.63, 0.80, 0.05, 0.06)\\
&IM = It\_May = (0.41, 0.58, 0.09, 0.07)\\
&SC = Small\_Chance = (0.22, 0.36, 0.05, 0.06)\\
&VLC = Very\_Low\_Chance = (0.1, 0.18, 0.06, 0.05)\\
&EU = Extremely\_Unlikely = (0.01, 0.02, 0.01, 0.05)\\
&I = Impossible = (0, 0, 0, 0)
\end{aligned}
$$

2. **Semantic based on the ordered structure of the linguistic term set:** In this case the linguistic term set the semantic is defined over the term set. This term set is useful when the user provides assessment by using ordered linguistic term set [45,

47]. Depending upon the distribution of the linguistic term on a scale [0,1] there are two possibilities for defining the semantic of the linguistic term set [39]:

i. **Symmetrically Distributed Terms:** In this case it is assumed that linguistic term sets are distributed on a scale with an odd cardinal and the mid term representing an assessment of “approximately 0.5” and the rest of the terms are placed symmetrically around it, Figure 2.10.

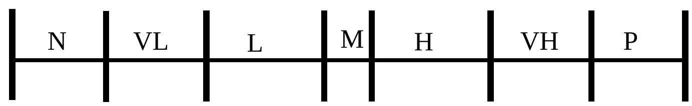


Figure 2.10: A Symmetrically Distributed Ordered Set of Seven Linguistic Terms

where the alphabetical symbols stand for

N = not applicable

VL = very low

L = low

M = medium,

H = high

VH = very high

P = perfect.

ii. **Non-Symmetrically Distributed Terms:** In non-symmetrically distributed terms, a certain sub-domain of the reference domain is more informative than the rest of the domain [47]. In this case the linguistic variable labels are clustered in a particular sub-domain than in the rest of the reference domain. The linguistic term set is non-symmetrically distributed as shown in Figure 2.11.

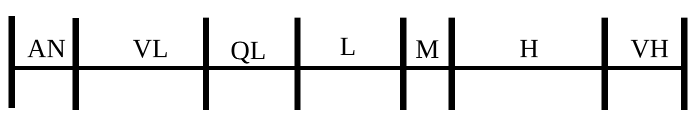


Figure 2.11: A Non-Symmetrically Distributed Ordered Set of Seven Linguistic Terms

where the alphabetical symbols stand for

AN = Absolutely not

VL = very low

QL = quite low

L = low

M = medium

H = high

VH = very high

3. **Mixed semantic:** In mixed semantic approach all the linguistic terms are considered primary and is a mix of ordered structure of the primary linguistic terms and fuzzy sets for the semantic of the linguistic terms. In mixed semantic ordered linguistic term sets are assumed to be distributed on a scale with an odd cardinality and the midterm indicating "approximately 0.5" and rest of the terms are placed symmetrically around it. In this case the semantic of the primary linguistic terms are represented by a trapezoidal or triangular membership functions [39, 48]. An example of the mixed semantic is as follows [39]

$$
\begin{aligned}
&P = Perfect = (1,1,0.16,0)\\
&VH = Very\_High = (0.84,0.84,0.18,0.16)\\
&H = High = (0.66,0.66,016,0.18)\\
&M = Medium = (=.5,0.5,0.16,0.16)\\
&L = Low = (0.34,0.34,0.18,0.16)\\
&VL = Very\_Low = (0.16,0.16,0.16,0.18)\\
&N = None = (0,0,0,0.16)
\end{aligned}
$$

The membership function of the above semantic is shown in Figure 2.12.

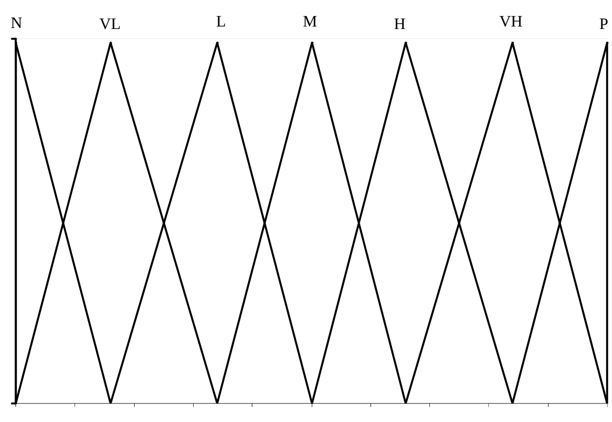


Figure 2.12: Uniformly Distributed Ordered Set of Seven Terms with its Sematics

2.4.4 Aggregation operator for linguistic weighted information

Aggregation of information is an important aspect for all kinds of knowledge based systems, from image processing to decision making. The purpose of aggregation process is to use different pieces of information to arrive at a conclusion or a decision. Conventional aggregation operators such as the weighted average are special cases of more general aggregation operators such as Choquet integrals [49]. The conventional aggregation operators have been articulated with logical connectives arising from many-valued logic and interpreted as fuzzy set unions or intersections [50]. The latter have been generalised in the theory of triangular norms [51]. Other aggregation operators that have been proposed are symmetric sums [52], null-norms [53], uninorm [54], apart from other.

An aggregation operator is a family of functions $\left\{ f^n, n \in N \right\}$ where N is the set of natural numbers, f^n (*f* is the aggregation operator) attaches to each n-tuple $\left(\alpha_1,...,\alpha_n\right)$ of values from L to another value $f^n(\alpha_1,...,\alpha_n)$ in L [55]. Some properties of aggregation operators are as follows:

if $a > b$ then $f(w,a) \geq g(w,b)$, where *f* and *g* are the aggregation operators

$f(0,a) = \mathrm{ID}$

where ID is an identity element, such that if it is added to aggregations it does not change the aggregated value.

The aggregation operators can be grouped into the following broad classes [50]:

1. **Operators generalising the notion of conjunction** are basically the minimum and all those functions f bounded from above by the minimum operators.
2. **Operators generalising the notion of disjunction** are basically the maximum and all those functions f bounded from below by the maximum operations
3. **Averaging operators** are all those functions lying between the maximum and minimum.

For linguistic weighted information the aggregation operators mentioned above have to be modified for linguistic variables and can be placed under two categories [56] Linguistic Weighted Disjunction (LWD) and Linguistic Weighted Conjunction (LWC). In Figure 2.13

the detailed classification of the linguistic aggregation operators is shown. In the following subsections the mathematical formulation of LWD and LWC is given. In order to illustrate each of the above mentioned linguistic aggregation operators the following example is considered [57]:

Example: For each alternative an expert is required to provide his/her opinion in terms of elements from the following scale

$$S = \left\{ OU\left(S_7\right),\ VH(S_6),\ H\left(S_5\right),\ M\left(S_4\right),\ L\left(S_3\right),\ VL\left(S_2\right),\ N\left(S_1\right) \right\}$$

where *OU* stands for Outstanding**,** *VH* for Very High, *H* for High, *M* for Medium, *L* for Low, *VL* for Very Low, *N* for None. The expert provides the opinion on a set of five criteria $\left\{C_1,\ C_2,\ C_3,\ C_4,\ C_5\right\}$. *An example of criteria as for electrical drive can be:*

C_1=Mass of the motor

C_2=Cost of the electrical drive

C_3= Losses in the electrical drive (motor + inverter)

C_4=Electrical time constant

C_5=Moment of inertia of the motor

The problem is to select a drive that has lowest losses, lowest cost, lowest mass, low electrical time constant and low moment of inertia. The motor is to be used in a hand held drill. For this application the mass of the motor and its cost are very important because a lighter motor with a low cost will be most preferred. Hence these two criteria are given Very High (VH) importance. For this application the efficiency of the motor is of moderate importance and is given a Medium (M) importance. The electrical time constant and moment of inertia of the rotor are important from the dynamic behaviour of the motor and are not very important for the application in hand held drill and are given low (L) and Very Low (VL) importance. The relation between the numerical values and the linguistic variables is given in Table 2.1a. The importance to each criterion is shown in Table 2.1b. The performance of an alternative on all the criteria is also shown in Table 2.1b, in brackets the numerical value is given. The performance of each alternative is also defined in terms of the scale $S = \left\{ OU\left(S_7\right),\ VH(S_6),\ H\left(S_5\right),\ M\left(S_4\right),\ L\left(S_3\right),\ VL\left(S_2\right),\ N\left(S_1\right) \right\}$.

Table 2.1a: The relation between numerical values and linguistic variables

	N	VL	L	M	H	VH	OU
C1	100-200	200-300	300-400	400-500	500-600	600-700	700-800
C2	10-20	20-30	30-40	40-50	50-60	60-70	70-80
C3	10-20	20-30	30-40	40-50	50-60	60-70	70-80
C4	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5-0.6	0.6-0.7	0.7-0.8
C5	1-2	2-3	3-4	4-5	5-6	6-7	7-8

Table 2.1b: Importance and score of alternative

Criteria	C1	C2	C3	C4	C5
Importance Weight (w)	VH	VH	M	L	VL
Score of Alternative 1	M (425)	L (34)	OU (77)	VH (0.65)	OU (7.6)
Score of Alternative 2	M (460)	OU (75)	VH (64)	VH (0.67)	H (5.6)
Score of Alternative 3	H(572)	M (47)	VH (64)	H (0.53)	OU (7.8)
Score of Alternative 4	OU (72)	M (45)	H (53)	VH (0.66)	H (5.8)
Score of Alternative 5	H (550)	M (46)	H (55)	OU(0.74)	VH (6.5)

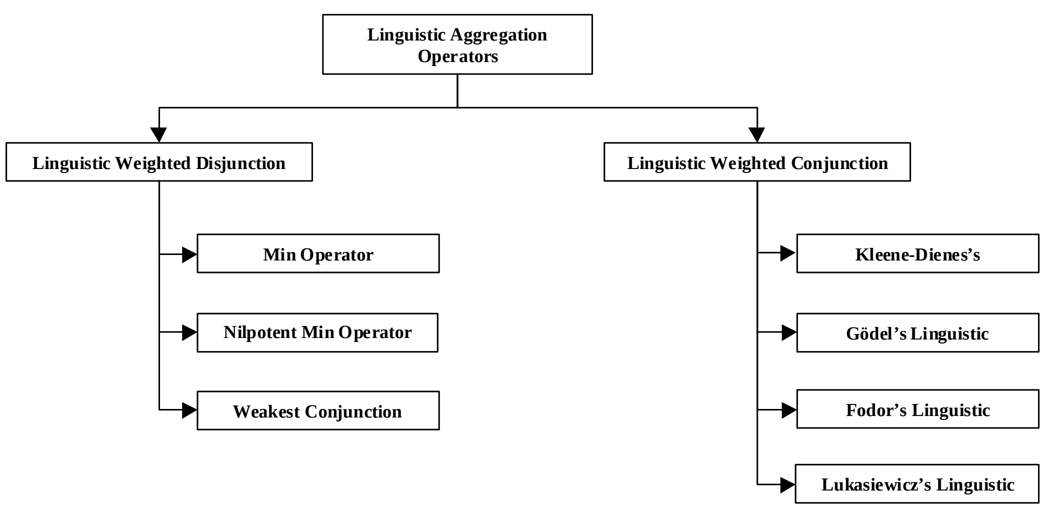


Figure 2.13: Classification of aggregation operator for linguistic variables

2.4.4.1 Linguistic Weighted Disjunction (LWD)

The aggregation of the weighted information using LWD is defined as follows

$$f = LWD\left[\left(w_1, a_1\right), \ldots \left(w_m, a_m\right)\right]$$

where $LWD = MAX_{i=1,\ldots,m} MIN\left(w_i, a_i\right)$. The different MIN operators are [58, 59]:

1. **The MIN Linguistic Disjunction** $LD_1^{\rightarrow}$ **:**

$$LD_1^{\rightarrow}(w,a) = MIN(w,a)$$

Based on the example given in Table 2.1 the net performance of the first alternative based on $LD_1^{\rightarrow}$ is

$$\begin{aligned} f_1 &= MAX\left[MIN(VH,M), MIN(VH,L), MIN(M,OU), MIN(L,VH), MIN(VL,OU)\right] \\ &= MAX\left[M, L, M, L, VL\right] = M \end{aligned}$$

2. **The Nilpotent Linguistic Disjunction** $LD_2^{\rightarrow}$ **:**

$$LD_2^{\rightarrow}(w,a) = \begin{cases} MIN(w,a) & \text{if } w > Neg(a) \\ s_1 & \text{otherwise} \end{cases}$$

Based on the example given in Table 2.1 the net performance of the first alternative based on $LD_2^{\rightarrow}$ is

$$\begin{aligned} f_1 &= MAX\left[LD_2^{\rightarrow}(VH,M), LD_2^{\rightarrow}(VH,L), LD_2^{\rightarrow}(M,OU), LD_2^{\rightarrow}(L,VH), LD_2^{\rightarrow}(VL,OU)\right] \\ &= MAX\left[M, L, M, L, VL\right] = M \end{aligned}$$

3. **The Weakest Linguistic Disjunction** $LD_3^{\rightarrow}$ **:**

$$LD_3^{\rightarrow}(w,a) = \begin{cases} MIN(w,a) & \text{if } MAX(w,a) = s_7 \\ s_1 & \text{otherwise} \end{cases}$$

Based on the example given in Table 2.1 the net performance of the first alternative based on $LD_3^{\rightarrow}$ is

$$f_1 = MAX\left[LD_3^{\rightarrow}(VH,M), LD_3^{\rightarrow}(VH,L), LD_3^{\rightarrow}(M,OU), LD_3^{\rightarrow}(L,VH), LD_3^{\rightarrow}(VL,OU)\right]$$
$$= MAX\left[N,N,M,N,VL\right] = M$$

The calculation of the total score of remaining alternatives of Table 2.1 using all the three LDW aggregation functions is given in Appendix A.

2.4.4.2 Linguistic Weighted Conjunction (LWC):

The aggregation of the weighted information using LWC is defined as follows

$$f = LWC\left[\left(w_1, a_1\right), \ldots \left(w_m, a_m\right)\right]$$

where $LWC = MIN_{i=1,\ldots,m} MAX\left(Neg\left(w_i\right), a_i\right)$ and m is the number of alternatives**.** The different types of MAX operators are [60]

1. **Kleene-Dienes's Linguistic Implication Function** $LI_1^{\rightarrow}$ **:**

$$LI_1^{\rightarrow}(w,a) = Max\left(Neg(w), a\right)$$

Based on the example given in Table 2.1 the net performance of the first alternative based on $LI_1^{\rightarrow}$ is

$$f_1 = MIN\left[LI_1^{\rightarrow}(VH,M), LI_1^{\rightarrow}(VH,L), LI_1^{\rightarrow}(M,OU), LI_1^{\rightarrow}(L,VH), LI_1^{\rightarrow}(VL,OU)\right]$$
$$= MIN\left[M, L, OU, VH, OU\right] = L$$

2. **Gödel's Linguistic Implication Function** $LI_2^{\rightarrow}$ **:**

$$LI_2^{\rightarrow}(w,a) = \begin{cases} s_T & \text{if } w \le a \\ a & \text{otherwise} \end{cases}$$

Based on the example given in Table 2.1 the net performance of the alternative based on $LI_2^{\rightarrow}$ is

$$f_1 = MIN\left[LI_2^{\rightarrow}(VH,M), LI_2^{\rightarrow}(VH,L), LI_2^{\rightarrow}(M,OU), LI_2^{\rightarrow}(L,VH), LI_2^{\rightarrow}(VL,OU)\right]$$
$$= MIN\left[M, L, OU, OU, OU\right] = L$$

3. **Fodor's Linguistic Implication Function** $LI_3^{\rightarrow}$ **:**

$$LI_3^{\rightarrow}(w,a)=\begin{cases} s_T & \text{if } w \le a \\ MAX\left(Neg(w),a\right) & \text{otherwise} \end{cases}$$

Based on the example given in Table 2.1 the net performance of the alternative based on $LI_3^{\rightarrow}$ is

$$\begin{aligned} f_1 &= MIN\left[LI_3^{\rightarrow}(VH,M), LI_3^{\rightarrow}(VH,L), LI_3^{\rightarrow}(M,OU), LI_3^{\rightarrow}(L,VH), LI_3^{\rightarrow}(VL,OU)\right] \\ &= MIN\left[M,L,OU,OU,OU\right] = L \end{aligned}$$

4. **Lukasiewicz's Linguistic Implication Function** $LI_4^{\rightarrow}$ **:**

$$LI_4^{\rightarrow}(w,a)=\begin{cases} s_T & \text{if } w \le a \\ Neg(w-a) & \text{otherwise} \end{cases}$$

Based on the example given in Table 2.1 the net performance of the alternative based on $LI_4^{\rightarrow}$ is

$$\begin{aligned} f_1 &= MIN\left[LI_4^{\rightarrow}(VH,M), LI_4^{\rightarrow}(VH,L), LI_4^{\rightarrow}(M,OU), LI_4^{\rightarrow}(L,VH), LI_4^{\rightarrow}(VL,OU)\right] \\ &= MIN\left[H,M,OU,OU,OU\right] = M \end{aligned}$$

The calculation of the total score of remaining alternatives of Table 2.1 using all the three LWC aggregation functions is given in Appendix A. The results of total score of all the five alternatives based on different aggregation operators is summarised below in Table 2.2.

From the above the following conclusions can be drawn:

1. The choice of linguistic aggregation operator can influence the results of the intermediate analysis process.
2. The linguistic weighted disjunction aggregation operators in general give an optimistic average value to alternatives. The Weakest linguistic disjunction gives the least optimistic value to the alternatives.
3. The linguistic weighted conjunction aggregation operators in general give a pessimistic average value to the alternatives.

4. Out of all the conjunction operators the Lukasiewicz's implication operator gives the least pessimistic final score to all the alternatives.
5. The disjunction aggregation operators are suitable if it is required to select a set of as many alternatives as possible. This situation can arise in the initial design phase when the designer wants to include as many alternatives as possible for further investigation.
6. In the initial design process if the number of alternatives is large and there is limited capability, in terms of manpower and computing power, to investigate each alternative then linguistic weighted conjunction operators are preferred.

Table 2.2: The result of total score of all the alternatives using different aggregation operators

Alternative ->	**1**	**2**	**3**	**4**	**5**
Min $LD_1^{\rightarrow}$	M	VH	H	VH	H
Nilpotent $LD_2^{\rightarrow}$	M	VH	H	VH	H
Weakest $LD_3^{\rightarrow}$	M	VH	VL	VH	L
Kleene-Dienes's $LI_1^{\rightarrow}$	L	M	M	M	M
Gödel's $LI_2^{\rightarrow}$	L	M	M	M	M
Fodor's $LI_3^{\rightarrow}$	L	M	M	M	M
Lukasiewicz's $LI_4^{\rightarrow}$	M	H	M	H	H

2.5 Final Analysis Phase of PDM

In the final analysis detailed simulation model of the target system is developed. After intermediate analysis the set of plausible solutions is greatly reduced and hence a detailed simulation for each solution is feasible. After setting up of the simulation model a new set of *Independent design variables* and *objectives* is identified. The steps involved in this stage are:

1. **Detailed simulation model of the target system is developed.**
2. **Independent design variables and objectives are identified.**
3. **Each solution in the reduced solution set is optimised for the new *objectives* and a set of solutions is obtained**
4. **Final decision is made.**

2.6 Conclusion

In this chapter the progressive design methodology (PDM) was proposed. This methodology is suitable for designing complex systems, such as electrical drive and power electronics (ED&PE), from conceptual stage to final design. The main aspects of PDM discussed are as follows:

1. PDM allows effective and efficient practices and techniques to be used from the start of the project.
2. The computation time required for optimisation is reduced as the bulk of optimisation is done in the synthesis phase and the models of the components of the target system are simple in the synthesis phase.
3. The experience of design engineers and production engineers are included in the intermediate analysis thus ensuring that the target system is feasible to manufacture.

In PDM the decision making factor is critical as proper decisions about dimensions, features, materials, and performance in the conceptual stage will ensure a robust and optimal design of the system. For decision making different types of aggregation operators were presented. In general the linguistic weighted disjunction aggregation operators in general give an optimistic average value to alternatives where as the linguistic weighted conjunction aggregation operators in general give a pessimistic average value to the alternatives.

In Progressive design methodology (PDM) it has been proposed to use linguistic weighted conjunction aggregation operator because the set of feasible alternatives obtained is small. The small number of alternatives is preferred because in the third step of PDM a detailed model of each alternative is to be developed for further investigation and these detailed models are computationally expensive. Hence a smaller set of alternatives will reduce the time to reach the final design.

The most important aspect of **PDM** is multiobjective optimisation algorithm. In the next chapter the general aspects of multiobjective optimisation problem (MOOP) and a survey of different types of multiobjective optimisation algorithms (MOOA) are presented.

2.7 References

[1] R. J. Balling and J. S. Sobieszczanski, "Optimization of coupled systems: A critical overview of approaches," *AIAA*, vol. 34, pp. 6-17, 1996.

[2] K. Lewis and F. Mistree, "Collaboration, Sequential and Isolated Decision Design," *ASME Journal of Mechanical Design*, vol. 120, pp. 643-652, 1998.

[3] J. S. Sobieszczanski and R. T. Haftka, "Multidisciplinary Design Optimization," *Structural Optimization*, vol. 14, pp. 1-23, 1997.

[4] R. Tappeta, S. Nagendra, J. E. Renand, and K. Badhrinath, "Concurrent Sub-Space Optimization (CSSO) Code using iSIGHT," GE 1998.

[5] J. S. Sobieszczanski, T. D. Altus, M. Philips, and R. J. Sandusky, "Bilevel Integrated System Synthesis (BLISS) for Concurrent and Distributed Processing," presented at 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimisation, 2002.

[6] J. S. Sobieszczanski, J. S. Agte, and R. J. Sandusky, "Bilevel Integrated System Synthesis," *AIAA*, vol. 38, pp. 164-172, 2000.

[7] J. S. Sobieszczanski, "Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic System," presented at Second NASA/USAF Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, 1998.

[8] N. M. Alexandrov and R. M. Lewis, "Analytical and Computational Aspects with Multidisciplinary Design," *AIAA*, vol. 40, pp. 301-309, 2002.

[9] J. Marczyk, "Stochastic Multidisciplinary Improvement: Beyond Optimization," presented at Proceedings of 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, USA, 2000.

[10] M. T. Tong, "A Probabilistic Approach to Aeropropulsion System Assessment," presented at Proccedings of ASME TURBOEXPO, Munich, Germany, 2000.

[11] P. N. Koch, B. Wujek, and O. Golovidov, "A Multi-Stage, Parallel Implementation of Probabilistic Design Optimitzation in an MDO Framework," presented at Proceedings of 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, USA, 2000.

[12] N. Egorov, G. V. Kretinin, and I. A. Leshchenko, "Stochastic Optimization of Parameters and Control Laws of Aircraft Gas-Turbine Engines- A Step to a Robust Design," *Inverse Problems in Engineering Mechanics III*, pp. 345-353, 2002.

[13] A.Booker, J. Dennis, P. Frank, D.Serafini, V. Torczon, and M. Trosset, "A Rigorous Framework for Optimization of Expensive Functions by Surrogates," *Structural Optimization*, vol. 17, pp. 1-13, 1999.

[14] C. Audet, J. Dennis, D. W. Moore, A. Booker, and P. D. Frank, "A Surrogate model based method for constraint optimization," presented at Proceedings of the 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, USA, 2000.

[15] C. Audet and J. Dennis, "A Pattern Search Filter Method for Nonlinear Programming without Derivatives," *SIAM Journal of Optimization*, vol. 14, pp. 980-1010, 2004.

[16] S. Dasgupta, "The Structure of Design Process," *Advances in Computers*, vol. 28, pp. 1-67, 1989.

[17] C. Shakeri, "Discovery of Design Methodologies for the Integration of Multi-disciplinary Design," 1998.

[18] E. P. K. Chong and S. H. Zak, *An Introduction to Optimization*, 2nd ed: John Wiley & Sons, 2001.

[19] D. M. Buede, *The Engineering Design of Systems: Models and Methods*: Wiley Interscience, 1999.

[20] K. Deb and A. Srinivasan, "INNOVIZATION: Innovative Design Principles Through Optimization," I. I. T Kanpur, India 2005007, 2005.

[21] S. P. C. Hwang and K. Yoon, "Mathematical Programming With Multiple Objectives: A Tutorial," *Computers & Operations Research*, vol. 7, pp. 5-31, 1980.

[22] R. Steuer, *Multiple Criteria Optimization: Theory, Computation and Application*: John Wiley & Sons, 1986.

[23] I. Das and J. E. Dennis, "A close look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multi-criteria optimization problems," *Structural Optimization*, vol. 14, pp. 63-69, 1997.

[24] P. Krus, J. O. Palmberg, F. Löhr, and G. Backlund, "The Impact of Computational Performance on Optimisation in Aircraft Design," presented at I MECH E, AEROTECH 95, Birmingham, UK, 1995.

[25] D. Thurston, "A formal method for subjective design evaluation with multiple attributes," *Research in Engineering Design*, vol. 3, pp. 105-122, 1991.

[26] R. Steuer and E.-U. Choo, "An Interactive Weighted Tchebycheff Procedure for Multiple Objective Programming," *Mathematical Programming*, vol. 26, pp. 326-344, 1983.

[27] R. Benayon, J. D. Montgolfier, J. Tergny, and O. Laritchev, "Linear Programming with multiple objective functions: Step method (STEM)," *Mathematical Programming*, pp. 366-375, 1971.

[28] J. Hauser and D. Clausing, "The house of Quality," *Harward Business Review*, vol. 66, pp. 63-73, 1988.

[29] T. L. Satty, *The Analytical Heirarchy Process: Planning, Priority Setting, Resource Allocation*: McGraw Hill, 1980.

[30] S. Pug, *Total Design*. New York: Addision-Wesley, 1990.

[31] M. J. Scott and E. K. Antonsson, "Arrow's Theorem and Engineering Design Decision Making," *Research in Engineering Design*, vol. 11, pp. 218-228, 1999.

[32] B. Costa and P. Vincke, "Multiple criteria decision aid: An overview," *Readings in Multiple Criteria Decision Aid*, pp. 3-14, 1990.

[33] D. E. Whitney, "Manufacturing by Design," *Harvard Business Review*, vol. 66, pp. 83-91, 1988.

[34] M. J. Scott and E. K. Antonsson, "Aggregation Functions for Engineering Design Tradeoffs," *Fuzzy Sets and Systems*, vol. 99, pp. 253-264, 1998.

[35] K. L. Wood and E. K. Antonsson, "Computing with Imprecise Parameters in Engineering Design: Backgroung and Theory," *ASME Journal of Mechanisms, Transmissions and Automation in Design*, vol. 111, pp. 616-625, 1989.

[36] L. A. Zadeh, "The concept of a linguistic variable and its applications to approximate reasoning. Part III," *Information Sciences*, vol. 9, pp. 43-80, 1975.

[37] L. A. Zadeh, "The concept of a linguistic variable and its applications to approximate reasoning. Part I," *Information Sciences*, vol. 8, pp. 301-357, 1975.

[38] L. A. Zadeh, "The concept of a linguistic variable and its applications to approximate reasoning. Part II," *Information Sciences*, vol. 8, pp. 357-377, 1975.

[39] F. Herrera and E. Herrera-Viedma, "Linguistic Decision Analysis: Steps for Solving Decision Problems under Linguistic Information," *Fuzzy Sets and Systems*, vol. 115, pp. 67-82, 2000.

[40] G. Bordogna and G. Passi, "A Fuzzy Linguistic Approach Generalizing Boolean Information Retrival: A Model and its Evaluation," *Journal of the American Society for Information Science*, vol. 44, pp. 70-82, 1993.

[41] P. P. Bonissone, "A Fuzzy Sets Based Linguistic Approach," *Approximate Reasoning in Decision Analysis*, pp. 329-339, 1986.

[42] F. Pernkopf, "Natural Language Processing with Application to Machine Translation: A Gentle Introduction," Graz University of Technology.

[43] G. A. Miller, "The Magical Number Seven or Minus Two: Some Limits on our Capacity of Processing Information," *Psychological Review*, vol. 63, pp. 81-97, 1956.

[44] P. P. Bonissone and K. S. Decker, "Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-off Precision and Complexity," *Uncertainty in Artificial Intelligence*, pp. 217-247, 1986.

[45] G. Bordogna, M. Fedrizzi, and G. Passi, "A Linguistic Modelling of Consensus in Group Decision Making Based on OWA Operators," *IEEE Transactions on System, Man, and Cybernatics - Part A: Systems and Humans*, vol. 27, pp. 126-132, 1997.

[46] M. Delago, J. L. Verdegay, and M. A. Vila, "Linguistic Decision Making Models," *International Journal of Intelligent Systems*, vol. 7, pp. 479-492, 1992.

[47] V. Torra, "Negation Functions Based Semantics for Ordered Linguistic Labls," *International Journal of Intelligent Systems*, vol. 11, pp. 975-988, 1996.

[48] F. Herrera and J. L. Verdegay, "A Linguistic Decision Process in Group Decision Making," *Group Decision and Negotiation*, vol. 5, pp. 165-176, 1996.

[49] M. Gabrish, T. Murofushi, and M. Sugeno, *Fuzzy Measures and Integrals*. Heidelberg, Germany: Physica-Verlag, 1999.

[50] D. Dubious and H. Prade, "On the use of aggregation operations in information fusion process," *Fuzzy Sets and Systems*, vol. 142, pp. 143-161, 2004.

[51] E. P. Klement, R. Mesiar, and E. Pap, *Triangular Norms*: Kluwer Academic Publishers, 2000.

[52] W. Sivert, "A Class of Operations Fuzzy Sets," *IEEE Transactions on System, Man, and Cybernatics - Part A: Systems and Humans*, vol. 9, 1979.

[53] T. Calvo, B. D. Baets, and J. Fodor, "The functional equations of Alsina and Frank for uniforms and null-norms," *Fuzzy Sets and Systems*, vol. 120, pp. 15-24, 2001.

[54] R. Yager and J. Fodor, "Structure of Uninorms," *Journal of Uncertainity, Fuzziness and Knowledge based Systems*, vol. 5, pp. 411-427, 1997.

[55] T. Calvo and B. D. Baets, "Aggregation Operators:New Trends and Applications," *Studies in Fuzziness and Soft Computing*, vol. 97, 2002.

[56] F. Herrera and E. Herrera-Viedma, "Aggregation Operators for Linguistic Weighted Information," *IEEE Transactions on Systems, Man and Cybernatics*, vol. 27, pp. 646-656, 1997.

[57] C. Carlsson and R. Fuller, "On fuzzy screening systems," presented at EUFIT'95, Aachen, Germany, 1995.

[58] R. R. Yager, "A Note on Weighted Queries in Information Retrival Systems," *Journal of the American Society for Information Science*, vol. 38, pp. 23-24, 1987.

[59] R. R. Yager, "A New Methodology for Ordinal Multiple Aspect Decision Based on Fuzzy Sets," *Decision Sciences*, vol. 12, pp. 589-600, 1981.

[60] E. Herrera and E. Herrera-Viedma, "Aggregation Operators for Linguistic Weighted Information," *IEEE Transactions on System, Man, and Cybernatics*, vol. 27, pp. 646-656, 1997.

Chapter 3

Multi-objective Optimisation Problems and Algorithms

3.1 Introduction

In chapter 2 the Progressive Design Methodology (PDM) was presented and it was seen that engineering systems design problem falls in the category of multiobjective optimisation problems. In this chapter an overview of the multiobjective optimisation problems and algorithms to solve these problems are presented. In this chapter the multiobjective optimisation problem is dealt with the view of posteriori aggregation of objectives. In a posteriori method the decision maker is presented with a set of Pareto optimal solutions before expressing any preferences. The posteriori method has the advantage that the results are independent of any decision making process and the decision maker expresses his decision after the optimization has been performed. Hence using a posteriori method demands using multiobjective optimization algorithms that result in Pareto optimal solutions. In PDM a posteriori method is used.

In Section 3.2 principles of multiobjective optimisation problems (MOOP) are presented and basic concepts are formally defined. A brief survey of algorithms to solve MOOP is given in section 3.3. The section 3.4 gives an overview of Multiobjective Optimisation Genetic Algorithms (MOOGAs). The issues involving convergence of MOOGAs and diversity of solutions obtained by these algorithms (MOOGAs) are discussed in section 3.5.

3.2 Multiobjective Optimisation Problems: Overview and Definitions

The multi-objective optimisation problems (MOOP) are made up of three basic components: a set of unknowns or variables, a set of objective functions to be minimised or maximised and a set of constraints that specify feasible values of variables. The optimisation problem entails finding values of the variables that optimise (minimise or maximise) the objective functions while satisfying the constraints. Multi-objective optimisation can be defined as the problem of finding [1]:

> *A vector of decision variables which satisfies constraints and optimises a vector function whose elements represent the objective functions. These functions form a mathematical description of performance criteria that are usually in conflict with each other. Hence, the term "optimise" means finding such a solution that would give the values of all the objective functions acceptable to the designer.*

Definition 1 (Multi Objective Optimisation Problem): *Mathematically the MOOP can be defined as optimise*

$$\boldsymbol{f}(\vec{x}) = \left(f_1(\vec{x}), f_2(\vec{x}), ..., f_m(\vec{x}) \right) \tag{3.1}$$

subject to $\boldsymbol{g}(\vec{x}) \leq 0$, $\boldsymbol{h}(\vec{x}) = 0$ *and* $\vec{x}^L \leq \vec{x} \leq \vec{x}^U$ where $\vec{x} = \left(x_1, x_2, ..., x_n \right) \in \boldsymbol{X}$ is the decision variable vector, **X** is denoted as the decision space, $\vec{x}^L = \left(x_1^L, x_2^L ..., x_n^L \right) \in \boldsymbol{X}$ is the vector of lower bound for each variable, $\vec{x}^U = \left(x_1^U, x_2^U ..., x_n^U \right) \in \boldsymbol{X}$ is the vector of upper bound for each variable and *n* is the total number of decision variables. The objective space **f** is a set of **m** objective vectors. The inequality constraint space is represented by $\boldsymbol{g}(\vec{x}) = \left(g_1(\vec{x}), g_2(\vec{x}), ..., g_p(\vec{x}) \right)$, which is a set of **p** functions and $\boldsymbol{h}(\vec{x}) = \left(f_1(\vec{x}), f_2(\vec{x}), ..., f_q(\vec{x}) \right)$ is the equality constraint space, which is a set of **q** functions. The last set of constraints are called variable bounds, restricting each decision variable x_i to take a value within a lower bound x_i^L and an upper bound x_i^U. These bounds constitute a *decision variable space*.

In multi-objective optimisation problems a situation may arise where it is required to minimise all the functions in the objective space, maximise all the functions or minimise some and maximise the others. In order to maintain the uniformity, all the functions in the objective space are converted to either their maximised or minimised form using the following identity:

$$max\, f(\vec{x}) \;=\; -\,min(-f(\vec{x})) \tag{3.2}$$

Hence, without loss of generality, it can be stated that all functions in the objective space in the above definition are to be minimised. The number of equality constraints (***q***) must be less than the number of decision variables (***n***) because if $q \geq n$ the problem is **over-constrained** [2] as there are no degrees of freedom left for optimising. The inequality constraints ($\boldsymbol{g}(\vec{x})$) are represented as 'less-than-equal-to' ($\leq$) type, although the 'greater-than-equal-to' ($\geq$) constraints are also accounted for in the above formulation. In case of '$\geq$' type of constraints, the constraints must be converted into a '$\leq$' type by multiplying the constraints by –1 [3]. A solution vector $\vec{x}$ that does not satisfy all the (p+q) constraints and all of the variable bounds (2n, n lower bounds and n upper bounds) is called an ***infeasible solution*** and the solution vector $\vec{x}$ that satisfy all the constraints and variable bounds is known as ***feasible solution***. In the presence of constraints ($\boldsymbol{g}(\vec{x}), \boldsymbol{h}(\vec{x})$) and variable bounds the entire decision variable space need not necessarily be feasible. The set of all feasible solutions is called the ***feasible region***.

Definition 2 (Feasible Region): The feasible set S is defined as the set of decision vectors $\vec{x} \in \boldsymbol{X}$ that satisfy all the equality constraints, inequality constraints and the variable bounds:

$$S = \left\{ x \in X \ \mid \ g(\vec{x}) \leq 0 \ \wedge \ h(\vec{x}) = 0 \ \wedge \ x_i^L \leq x_i \leq x_i^U \ \ \forall \ i = 1, 2, ...n \right\} \tag{3.3}$$

The principles of multi-objective optimisation are different from that of a single objective optimisation. The goal in single objective optimisation problem is to find the global optimal solution whereas in a multi objective optimisation there are more than one objective functions and each of the objective function has a different optimal solution. If there is difference in the optimal solutions corresponding to different objectives, the objective functions are often known as conflicting [4]. Multi-objective optimisation with such conflicting objective functions give rise to a set of optimal solutions instead of single optimal solution. The reason for the optimality of many solutions is that no one can be considered to be better than any other with respect to all objective functions. These optimal solutions are known as ***Pareto optimal*** solutions as illustrated in Figure 3.1.

In Figure 3.1 the Pareto optimal solutions are shown for minimisation problem with two objectives. It can be seen from Figure 3.1 that solution A has smaller value of function f_1 but a higher value of function f_2 compared to solution D. Hence none of these two solutions can be said to be better than the other with respect to both objectives and are called ***non-dominated*** solutions. In Figure 3.1 solutions B and C are also non-dominated solutions. In order to express this situation mathematically, the relations $=, \geq, \leq$ are extended to objective vectors by analogy to single-objective case [5]:

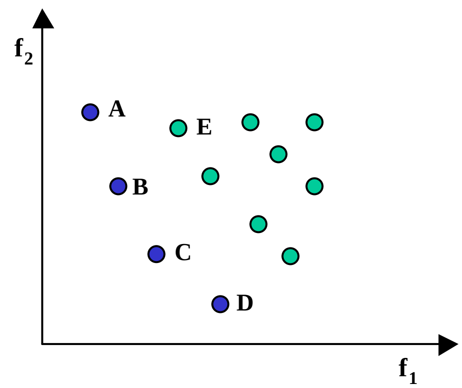


Figure 3.1: Pareto optimal solution for a bi-objective minimisation problem where both f1 and f2 are minimised

Definition 3: For any two objective vectors **u** and **v**

$$\boldsymbol{u} = \boldsymbol{v} \text{ iff } \forall i \in \{1, 2, ..., m\} : u_i = v_i$$
$$\boldsymbol{u} \leq \boldsymbol{v} \text{ iff } \forall i \in \{1, 2, ..., m\} : u_i \geq v_i \qquad (3.4)$$
$$\boldsymbol{u} < \boldsymbol{v} \text{ iff } \forall i \in \{1, 2, ..., m\} : u_i > v_i$$

The relations $\geq$, $>$ are defined similarly.

When solution E is compared with solution C it is observed that solution C is better than solution E because solution C has smaller values for both the objectives (f_1 and f_2), i.e. $C \leq E$. In this case solution C is said to *dominate* solution E or that solution E is *dominated* by solution C. This leads to the definition of Pareto Dominance.

Definition 4 (Pareto Dominance): For two decision vectors $\boldsymbol{a} = (a_1, ..., a_n)$ and $\boldsymbol{b} = (b_1, ..., b_n)$ three conditions arise:

$\boldsymbol{a} \succeq \boldsymbol{b}$ (**a** weakly dominates **b**), if and only if $\boldsymbol{f}(a_i) \leq \boldsymbol{f}(b_i) \wedge \exists\, i \in \{i, ..., n\}$.

$\boldsymbol{a} \succ \boldsymbol{b}$ (**a** dominates **b**), if and only if $\boldsymbol{f}(a_i) < \boldsymbol{f}(b_i) \wedge \exists\, i \in \{i,...,n\}$. (3.5)

$\boldsymbol{a} \sim \boldsymbol{b}$ (**a** is indifferent to **b**), if and only if $\boldsymbol{f}(a_i) \not\leq \boldsymbol{f}(b_i) \wedge \boldsymbol{f}(b_i) \not\leq \boldsymbol{f}(a_i) \wedge \exists\, i \in \{i,...,n\}$.

The definitions for a maximisation problem are analogues.

Based on the concept of Pareto Dominance, the optimality criterion for multi objective optimisation can be introduced. With respect to Figure 3.1, solution A is unique: its corresponding decision vector $\boldsymbol{a}$ is optimal in the sense that it cannot be improved in any objective without causing degradation in at least one other objective. Similarly solutions B, C and D are optimal and improvement in any objective will cause degradation in other objective. Such solutions are known as Pareto Optimal.

Definition 5 (Pareto Optimality): *A solution* $\boldsymbol{a} \in X$ *is said to be Pareto optimal with respect to* X *if and only if there is no* $\boldsymbol{b} \in X$ *for which* $\boldsymbol{v} = \left(f_1(b),..., f_m(b) \right)$ *dominates* $\boldsymbol{u} = \left(f_1(a),..., f_m(a) \right)$.

In Figure 3.1 the points marked in blue represent the Pareto optimal solutions. None of these solutions can be identified as better than the other. These solutions are indifferent to each other i.e., there is no single optimal solution but rather a set of optimal trade-offs. The set of all the Pareto-optimal solutions is known as *Pareto-optimal set*. When the Pareto-optimal solutions are plotted in objective space, the non-dominated vectors are collectively known as the Pareto front (Figure 3.1). To restate, the Pareto optimal set is a subset of all possible solutions in feasible region S .

Definition 6 (Local Pareto Optimal Set): For a given multi objective problem the Pareto optimal set P^* is defined as:

$$P^* := \{ a \in S \mid \neg \exists b \in S \,:\, f(a) \succeq f(b) \} \tag{3.6}$$

where f is the objective function

Definition 7: (Global Pareto Optimal Set): The non-dominated set of the entire feasible search space S is globally Pareto optimal set.

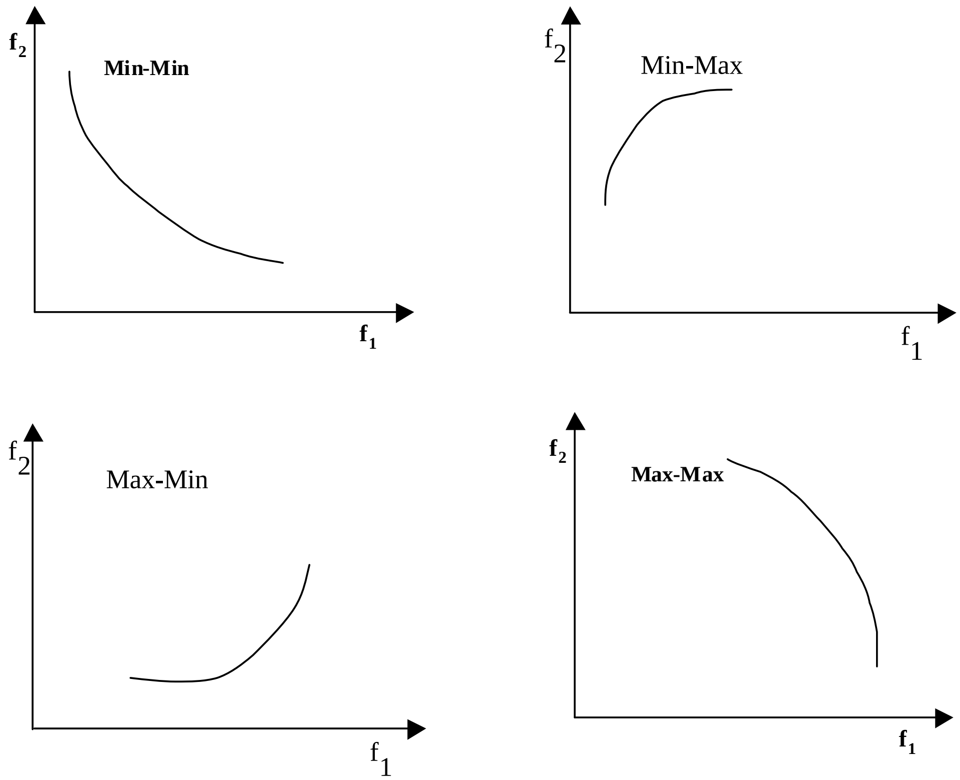


Figure 3.2: Pareto optimal solutions for four different scenarios [3]

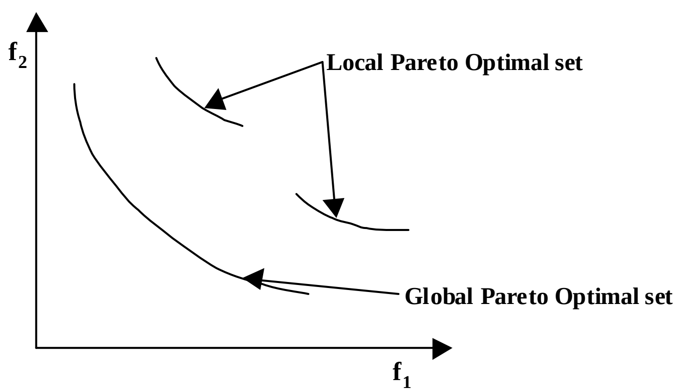


Figure 3.3: Local and global Pareto optimal sets

Definition 8 (Pareto-optimal Front): For a given multi objective optimisation problem and Pareto optimal set P^*, the Pareto front PF^* is defined as:

$$PF^* := \left\{ \boldsymbol{u} = \boldsymbol{F}(x) = \left(f_1(x), ..., f_k(x) \right) | \, x \in P^* \right\} \tag{3.7}$$

In Figure 3.2 Pareto optimal sets for different scenarios with two objectives are shown [4]. Each objective can be either maximised or minimised. In Figure 3.3 two local and a global Pareto optimal sets are shown.

3.3 Solution Methods for Multiobjective Optimisation Problem

In general there are three broad categories of Multiobjective Optimisation Algorithms (MOOA): Scalarisation approaches, population based non-Pareto Approaches and Pareto-based approaches [6]. In Figure 3.4 the classification of the techniques for multiobjective optimisation (MOO) is shown. Reviews of multiobjective optimisation methods can be found in works of Terry [7], Kapur [8], Roy [9], Loucks [10], Cohon and Marks [11], Wierzbicki [12], Hwang and Masud [13], Hwang et.al. [14], Ignizio [15], Osyczka and Koski [16], Stadler [17], Starr and Zeleny [18], Lieberman [19], Evans [20], Fishburn [21] and Colello [22]. Some of the important MOOAs are given in appendix B.

In the scalarisation methods the multiobjective optimization problem (MOOP) is solved by converting the MOOP into single objective problem. The objectives are aggregated to form a single objective function. The formation of the aggregate objective function requires that the preferences or weights between objectives are assigned apriori, i.e. before the results of the optimisation process are known [23]. Some of the disadvantages of scalarisation method are [24]:

1. they require apriori selection of weights or targets for each of the objective functions
2. they provide information for only one design scenario (i.e. a single Pareto solution)
3. they are unable to generate proper Pareto points for non-convex problems

The Pareto methods keep the objectives throughout the optimisation process and uses the concept of dominance to distinguish between Pareto Dominated and Pareto nondominated solutions [23]. Most of the Pareto based methods optimise all the objectives simultaneously.

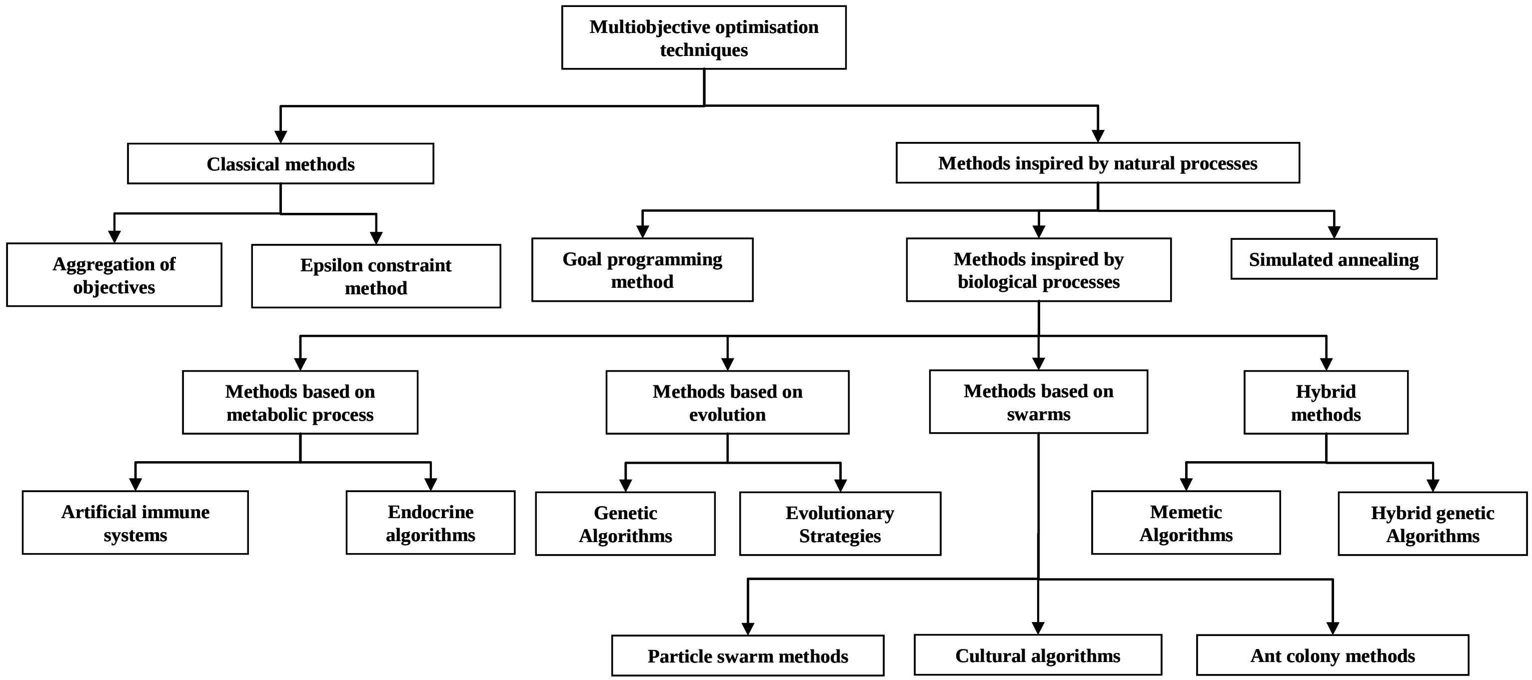


Figure 3.4: Classification of multiobjective optimisation algorithms

Evolutionary Algorithms (EAs) best perform the simultaneous optimisation because they generate multiple solutions in a single run. The EAs are randomised search algorithms inspired by principles of natural evolution. The doctoral study of vector evaluated genetic algorithm (VEGA) by Schaffer [25] and Goldberg's work on non-dominated sorting along with a niching mechanism [26] generated an overwhelming interest on multiobjective evolutionary algorithms (MOEAs). The Genetic Algorithms (GA) and Evolutionary Strategies (ES) are used as baseline algorithms in most of the multiobjective optimisation Problems (MOOP) [27].

Initial MOEAs, such as the Multi Objective Genetic Algorithm (MOGA) [28], the Non-dominated Sorting Genetic Algorithm (NSGA) [29] and the Niched Pareto Genetic Algorithm NPGA [30], were directly based on the suggestions of Goldberg [31] and consisted of two primary steps: (i) the fitness of a solution was determined using its dominance within the population and (ii) the diversity among solutions was preserved using a niching strategy. The above mentioned three genetic algorithms show that these steps can be implemented in different ways resulting in a variety of MOEAs that can be conceived from the suggestions of Goldberg. The elitism operator was absent in these MOEAs which resulted in their poor performance. Hence the focus of later work was mostly concentrated on how elitism could be introduced in a MOEA. As a result of this a number of advanced algorithms emerged such as the Strength Pareto Evolutionary Algorithm SPEA [32], the Pareto Archived Genetic Algorithm PAES [33] and the Non Dominated Sorting Genetic Algorithm II (NSGA-II) [34], among others. In further attempts to improve the quality of the solutions and to obtain well spread solutions of the Pareto Front, algorithms with dynamic population size were developed by Tan et. al. [35]. Adaptive mutation rates were implemented to further accelerate the search for optima and to enhance the ability to locate optima accurately. A detailed explanation and a review of various state-of-the-art evolutionary algorithms for solving multi-objective optimization problems are given by Coello et. al [36].

The Genetic Algorithms (GAs) are computationally simple, yet robust and powerful way to search for potential solutions to multiobjective optimisation problems. The basic feature of GAs is the multiple directional and global searches by maintaining a population of potential solutions from generation to generation. The population-to-population approach is likely to explore Pareto optimal solutions. The GAs do not involve many mathematical requirements about the problems and can handle any kind of objective functions and constraints. The major advantages of GAs for multiobjective optimization are:

1. The GAs are intrinsically parallel. The GAs have multiple offspring and hence they can explore the solution space in multiple directions at once. The parallel nature of the GAs enable them to find several members of the Pareto optimal set in a single run of the algorithm.
2. The parallel nature of GAs are particularly well suited to solve problems where the space of all potential solutions is large.
3. The GAs perform well in situations where the solution space of the problem is complex, i.e. the solution space is discontinuous, noisy or has many local optima.
4. The GAs do not impose any requirements for a problem to be formulated in a particular constraint language and do not need the function to be differentiable, continuous, linear, separable or of any particular data type [37].

These characteristics make them suitable for optimisation problems that have a large and complex solution space. Therefore, in this thesis a Multiobjective Genetic Algorithm is developed and used for design and optimisation of a Permanent Magnet Motor Drive. The details of the developed Genetic Algorithm are given in chapter 4. In the next section a discussion about general principles of genetic algorithms is given.

3.4 Genetic Algorithms

John Holland [38] first developed the concept of genetic algorithm in 1960s. A genetic algorithm (GA) relies on the Darwin’s concept of survival of the fittest with sexual reproduction, where stronger individuals in the population have a higher chance of creating an offspring. The GA begins with random initialisation of the population. The transition of population from one generation to the next takes place via the application of the genetic operators like *selection*, *crossover* and *mutation*. The *selection* operator selects the individuals from the population for reproduction. The *crossover* operator randomly chooses a locus and exchanges the sub-sequences before and after that locus between two chromosomes to create two offspring. For example, the strings 1000100 and 11111111 may be crossed over at fourth locus to yield two offspring 10001111 and 1111100. The *crossover* operator roughly mimics biological recombination between two single chromosomes. The *mutation* operator randomly flips some of the bits in a chromosome. For example the string 11110011 may be mutated in its fifth position to yield 11100011. Mutation can occur at each bit position in a string with some probability. The genetic algorithms (GAs) have the following features:

1. GAs operate on a population of possible solutions instead of single individual. Thus the search is carried out in a parallel form.
2. GAs are able to find optimal or sub-optimal solutions in complex and large search spaces. The GAs can be modified to solve multiobjective optimisation problems.

GAs examine many possible solutions at the same time, hence they have a high probability to converge to a global optimum. The flowchart of a simple genetic algorithm is shown below in Figure 3.5.

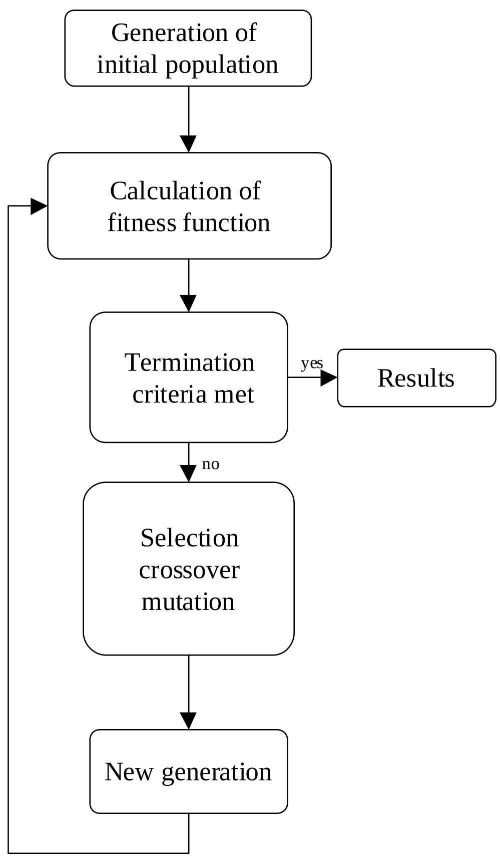


Figure 3.5: Flow chart of a simple Genetic Algorithm

In contrast to single objective optimisation, where objective function and fitness functions are often identical, both fitness assignment and selection must allow for several objectives in multi-objective optimisation problems. To solve this problem the genetic algorithms use the concept of *Pareto dominance* as proposed by Goldberg [26].

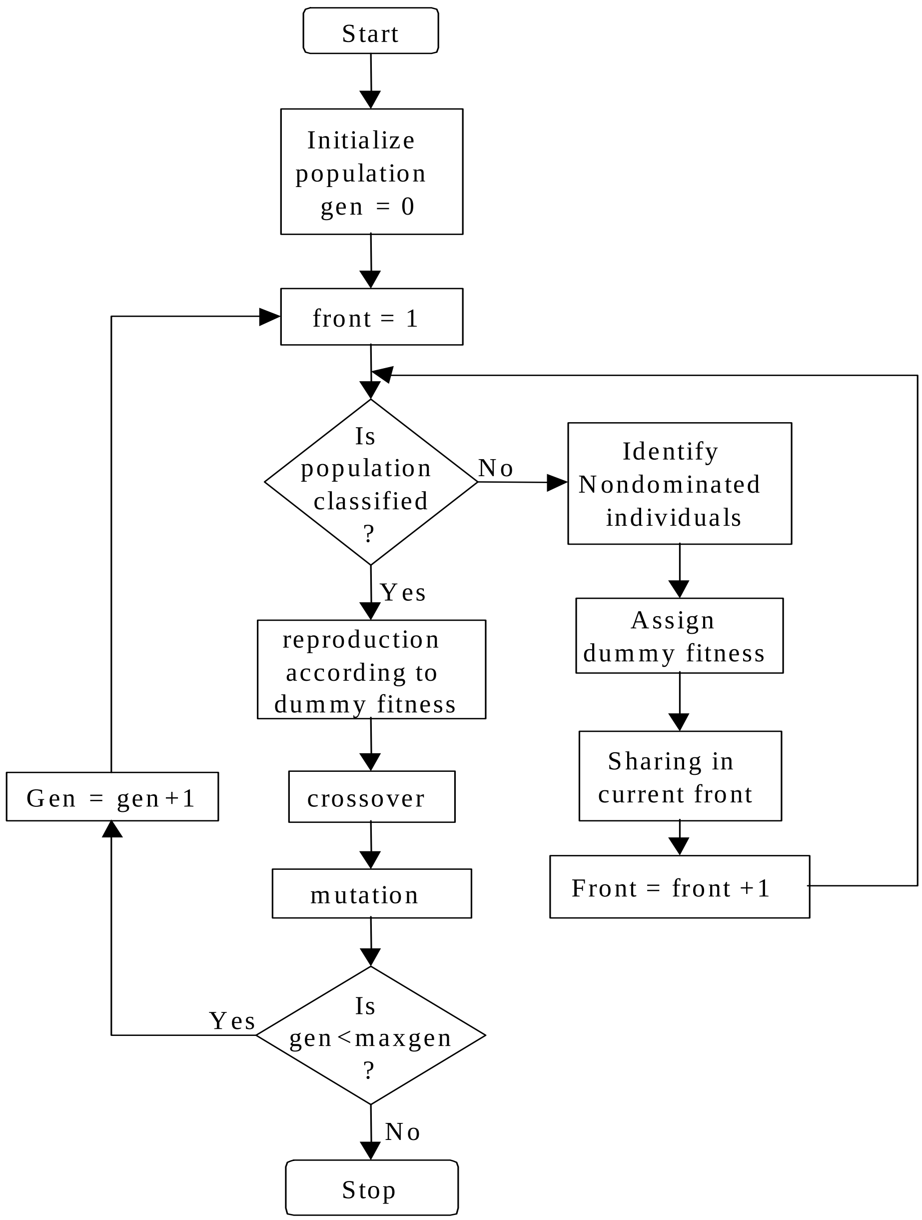


Figure 3.6: Flow chart of NSGA

The idea of non-dominated sorting gave rise to Non-dominated Sorting Genetic Algorithm (NSGA) (Figure 3.6) [3] and Multi-objective Genetic Algorithm (MOGA) [39]. Both NSGA and MOGA also use fitness sharing [37] which is a procedure that reduces the effective fitness of an individual in relation to the number of other individuals that occupy the same niche. In multi-objective optimisation the niche is often defined by the distance of solutions to one another in the objective space. Both NSGA and MOGA use similar methods to convert the shared fitness value to actual reproductive opportunity based on a ranking-based selection [37]. In order to improve the diversity of the solutions the concept of *elitism* was introduced in multi-objective optimisation genetic algorithm (MOGA). Elitism in GA terminology means the retention of good parents in the population from one generation to the next, to allow them to take part in selection and reproduction more than once and across generation [37]. Several early schemes of elitist MOGA are given in [5]. Some of the important MOGA is Strength Pareto Evolutionary Algorithm (SPEA) [40], Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [41] and Pareto Archived Evolutionary Strategy (PAES).

3.5 Conclusions and Discussions

The population based multiobjective optimisation techniques are inspired by the natural phenomenon. The major advantage of these algorithms is in their flexibility and robustness as global search techniques. All these methods do not use gradient information of the objective functions and can deal with highly non-linear and non-differentiable functions. These algorithms are also suited for functions with multiple local optima and are suitable for multi-objective optimisation problems because they are less susceptible to the shape or continuity of Pareto Front.

Despite their robustness and flexibility these algorithms are not free from difficulties. All these algorithms involve control parameters, such as annealing temperature in simulated annealing, *pheromone update* parameters in ant colony, mutation rates in GAs etc. In order to achieve successful optimisation appropriate setting of these parameters is essential. In general some trial and error tuning of the control parameters is necessary for each particular instance of optimisation problem. Additionally, any meta-heuristic approach should not be thought of in isolation: the possibility of utilising hybrid approaches should be considered [42].

The two primary goals in the area of multiobjective optimisation are: (1) to find a set of solutions as close as possible to the Pareto-optimal front and (2) to find a set of solutions as diverse as possible [4]. Both these issues are very important when using multiobjective optimisation techniques for design of engineering systems. In order to address these issues a new Multiobjective Genetic Algorithm has been proposed in this thesis. This algorithm is known as Non-dominated sorting Biologically Motivated Genetic Algorithm (NBGA). In NBGA new types of mutation operators are implemented in order to improve the convergence of solutions towards the Pareto-optimal front and find diverse well-scattered solutions. In the next chapter (Chapter 4) the details of the NBGA are laid out and results of comparison with other state of the art algorithms presented.

3.6 References

[1] A. Osyczka, "Multicriteria optimization for engineering design," in Design Optimization, J. S. Gero, Ed.: Academic Press, 1985.

[2] C. A. C. Coello, "An enperical study of evolutionary techniques for multiobjective optimization in engineering design," in Computer Science: Tulane University, 1996.

[3] K. Deb, Optimization for Engineering Design: Algorithms and Examples. New Delhi: Prentice Hall, 1995.

[4] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms: John Wiley & Sons, 2001.

[5] E. Zitzler, "Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications," in Computer Engineering and Networks Laboratory: Swiss Federal Institute of Technology Zurich, 1999.

[6] C. M. Fonseca and P. J. Fleming, Multiobjective Optimization. Bristol: Institute of Physics Publishing, 2000.

[7] H. Terry, "Comparitive evaluation of performance using multiple criteria," Management Science, vol. 9, pp. 431-442, 1963.

[8] K. C. Kapur, "Mathematical methods of optimization for multiobjective transportation systems," Socio-economic planning science, vol. 4, 1970.

[9] B. Roy, "Problems and methods with multiple objective functions," Mathematical Programming, vol. 1, pp. 239-266, 1971.

[10] D. P. Loucks, "Conflict and choice: Planning for multiple objectives," in Economic wide Models and Development Planning, C. Blitzer, P. Clark, and L. Taylor, Eds. New York: Oxfor University Press, 1975.

[11] J. L. Cohon and D. H. Marks, "A review and evaluation of multiobjective programming techniques," Water Resources Research, vol. 11, pp. 208-220, 1975.

[12] A. P. Wierzbicki, "A methodological guide to multiobjective optimization," IIASA Working Paper, 1979.

[13] C. L. Hwang and A. S. M. Masud, "Multiple objective decision-making methods and applications," Lecture notes in Economics and Mathematical Systems, vol. 164, 1979.

[14] C. L. Hwang, S. R. Paidy, and K. Yoon, "Mathematical programming with multiple objectives: A tutorial," Computing and Operational Research, vol. 7, pp. 5-31, 1980.

[15] J. P. Ignizio, Linear Programming in Single- and Multiple-Objective Systems: Prentice Hall, 1982.

[16] A. Osyczka and J. Koski, "Selected works related to multicriterion optimization methods for engineering design," presented at Euromech, University of Siegen, 1982.

[17] W. Stadler, "A Survey of Multicriteria Optimization or the Vector Maximum Problem, Part I," Journal of Optimization Theory and Applications, vol. 29, pp. 1-52, 1984.

[18] M. K. Starr and M. Zeleny, "MCDM-state and future of the arts," in Multiple Criteria Decision Making, vol. 6, M. K. Starr and M. Zeleny, Eds. Amsterdam: North-Holland publishing Company, 1977, pp. 5-29.

[19] E. R. Lieberman, "Soviet multi-objective mathematical programming methods: An overview," Management Science, vol. 37, pp. 1147-1165, 1991.

[20] G. W. Evans, "An overview of techniques for solving multiobjective mathematical programs," Management Science, vol. 30, pp. 1268-1282, 1984.

[21] P. C.Fishburn, "A survey of multiattribute/multicriterion evaluation theories," in Multiple Criteria Problem Solving, S. Zionts, Ed. Berlin: Springer Verlag, 1978, pp. 181-224.

[22] C. A. C. Coello, "An Updated Survey of GA-Based Multiobjective Optimization Techniques," ACM Computing Surveys, vol. 32, 2000.

[23] O. L. d. Weck, "Multiobjective Optimization: History and Promise," presented at The Third China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical Systems, Kanazawa, Japan, 2004.

[24] R. V. Tappeta and J. E. Renaud, "Interactive Multiobjective Optimization Design Strategy for Decision Based Design," ASME Journal of Mechanical Design, vol. 123, pp. 205-215, 2001.

[25] J. D. Schaffer, "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms," presented at Iternational Coference on Genetic Algorithms and Their Applications, 1985.

[26] D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Reading, Maasachusetts: Addison-Wesley Publishing Company, 1989.

[27] M. M. Raghuwanshi and O. G. Kakde, "Survey on Multiobjective Evolutionary and Real Coded Genetic Algorithms," Complexity International, vol. 11, pp. 150-161, 2005.

[28] C.M.Fonseca and P.J.Fleming, "Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization," Proceedings of the Fifth International Conference on Genetic Algorithms, 1993.

[29] N.Srinivas and K.Deb, "Multi-objective function optimization using non-dominated sorting genetic algorithms," Evolutionary Computation Journal, vol. 3, pp. 221-248, 1994.

[30] J.Horn, N.Nafploitis, and D.E.Goldberg, "A niched Paretogenetic algorithm for multi-objective optimization," Proceedings of the First IEEE Conference on Evolutionary Computation, 1994.

[31] D. E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning," Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.

[32] E.Zitzler and L.Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," IEEE Transactions on Evolutionary Computation, vol. 3, pp. 257-271, 1999.

[33] J. D.Knowles and D. W.Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," Evolutionary Computation, vol. 8, pp. 149-172, 2000.

[34] K.Deb, S.Agrawal, A.Pratap, and T.Meyarivan, "A fast elitist nondominated sorting genetic algorithm for multi-objective optimization: NSGA-II," Proceedings of the Parallel Problem Solving from Nature VI(PPSN-VI), pp. 849-858, 2000.

[35] K. C. Tan, T. H. Lee, and E. F. Khor, "Evolutionary Algorithm With Dynamic Population Size and Local Exploration for Multiobjective Optimization," IEEE transactions on Evolutionary Computation, vol. 5, pp. 565-586, 2001.

[36] C. A. C. Coello, V. Veldhuizen, and et.al., "Evolutionary Algorithms for Solving Multi-Objective Problems." KluwerAcademic Publishers, New York, 2002.

[37] J. Knowles and D. Corne, "Memetic Algorithms for Multiobjective Optimization: Issues, Methods and Prospects," in Recent Advances in Memetic Algorithms, N. Krasnogor, J. E. Smith, and W. E. Hart, Eds.: Springer Verlag, 2004, pp. 313-352.

[38] J. H. Holland, Adaptation in Natural and Artificial Systems: MIT Press, 1975.

[39] C. M. Fonseca and P. J. Fleming, "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization," presented at Fifth International Conference on Genetic Algorithms, San Mateo, California, 1993.

[40] E. Zitzler and L. Thiele, "An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach," Computer Engineering and

Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH) 1998.

[41] K. Deb, S. Agarwal, A. Pratap, and T. Meyarivan, "Fast elitist non-dominated sorting genetic algorithm for multio-objective optimization: NSGA-II," presented at 6th International Conference on Parallel Problem Solving from Nature, 2000.

[42] K. O. Jones, "Comparison of Genetic Algorithm and Particle Swarm Optimization," presented at International Conference on Computer systems and Technologies (CompSysTech'2005), 2005.

Chapter 4

Non-dominated Sorting Biologically Motivated Genetic Algorithms (NBGA)

4.1 Introduction

The process of mutation has been studied extensively in the field of biology and it has been shown there that it is one of the major factors that aid the process of evolution. The impact of the mutation operator has been investigated by Aguirre et al. [1]. In this work they investigated the impact of selection, drift and mutation operators on the performance of evolutionary algorithms. The certain mutation operators: viz. duplication, segregation and transposition, were studied by Goldberg [2]. Similarly, Brizuela et.al. [3] performed experimental analysis of the genetic operators for a multi-objective Genetic Algorithm (GA) applied to the Flow shop problem. Furthermore, Chan et.al [4] used the concept of jumping genes and applied it to multi-objective resource management in wideband Code Division Multiple Access (CDMA) systems. In the jumping gene GA the transposition mutation operator was used to improve the performance of the GA.

Most genetic algorithms still use an elementary form of point mutation. Research in evolutionary biology has shown that mutation is one of the primary sources of diversity in nature. In this chapter a novel genetic algorithm using different types of mutations is presented to increase diversity in the solutions and to improve the convergence of the GAs. The issue of diversity of solutions and convergence of GAs become very important when applied to real life problems such as design and optimisation of engineering problems. If an optimisation algorithm fails to deliver Pareto optimal solutions that are not well spread then the designer has less choice of feasible designs. On the other hand if the optimisation algorithm does not converge then true Pareto optimal solutions will not be obtained and hence the system designed will not be optimal. In order to improve the convergence and diversity of the solutions one of the important prerequisite is to define proper rate of mutation. If the rate of mutation is too high or too low the convergence of the algorithm and diversity of the solutions obtained is adversely affected.

In the Non-dominated Sorting Biologically Motivated Genetic Algorithm (NBGA) a novel method for representation of variables has been proposed in order to facilitate the assignment of the rate of mutation for each type of mutation. In NBGA nine different types of mutations are implemented and defining the rate of mutation for each of them can be difficult for a novice. Hence to facilitate the use of algorithm a new method for representation of variables has been proposed. With this type of representation the user needs not to define the rate of mutation for each type of mutation but can define a global rate of mutation. The rational behind introducing such a scheme of representation of variables was to make the algorithm more user friendly and accessible to professionals who might not be familiar with the nuances of GA.

In Section 4.2 a brief discussion is given on the evolution from a biological perspective. The different types of mutation that are implemented in our algorithms are discussed in Section 4.3. The structure of NBGA, proposed by the authors, is given in Section 4.4. Section 4.5 gives an overview of the performance parameters that are used to evaluate the NBGA. The test functions on which the NBGA is evaluated and the results of the performance parameters for these test functions, along with the comparison with the other GAs, are given in Section 4.6. The performance of NBGA for multivariable test functions is discussed in section 4.7. The importance of the proposed mutation types is stressed in Section 4.8. Finally, the conclusions are drawn in Section 4.9.

4.2 Biological Perspective of Evolution

In general evolution is defined as any process of change occurring with time. In terms of life sciences evolution is a change in gene frequency in a population. The genes are the fundamental physical and functional units of heredity. Genes are made up of DNA, many genes constitute a chromosome and each organism in turn has many chromosomes. For example, humans have 46 chromosomes.

The mechanisms of evolution are: natural selection, mutation, recombination, genetic drift and gene flow, as depicted in Figure 4.1.

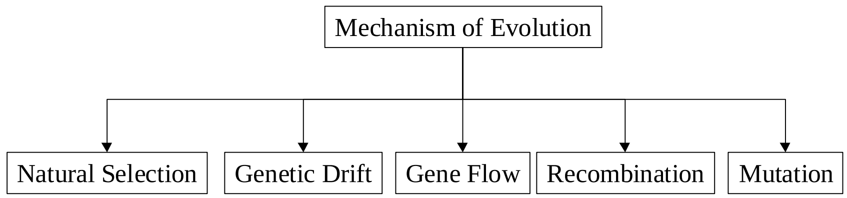


Figure 4.1: Mechanisms of evolution

Natural selection is the principle mechanism that causes evolution. Natural selection was expressed as a general law by Darwin [5], as quoted below:

1. IF there are organisms that reproduce and,
2. IF offspring inherit traits from their progenitor, and,
3. IF there is variability of traits and,
4. IF the environment cannot support all members of a growing population,
5. THEN those members of the population with inferior traits will die out, and,
6. THEN those members with better traits will thrive."

Natural selection can be subdivided into two types:

1. Ecological selection,
2. Sexual selection.

Ecological selection takes place in situations where inheritance of specific traits is solely determined by ecology. Sexual selection is the theory that states that competition for mates, between individuals of the same sex, drives the evolution of certain traits.

Natural selection occurs only when the individuals of a population have diverse characteristics. Natural selection ceases to operate when the population does no longer has any genetic variation. For evolution to continue, mechanisms that increase the genetic diversity are necessary. Mutation, recombination and gene flow are the mechanisms that increase the diversity in the population so that evolution can proceed onwards.

Genetic drift is the mechanism that acts in conjunction with natural selection and changes the characteristics of the species over a period of time. This is a stochastic process and is caused by random sampling in the reproduction of offspring. Like natural selection, genetic drift changes the frequencies of alleles but decreases the genetic variations.

Gene flow is the transfer of genes from one population to another. Migration into or out-of a population may be responsible for a significant change in the gene pool frequency. Addition of new genetic material is facilitated by immigration whereas emigration results in the removal of genetic material.

Recombination is the process by which the combination of genes in an organism's offspring differs from that of its parents. Recombination results in a shuffling of the genes. Recombination is a mechanism of evolution because it adds new alleles to the gene pool.

Mutations are permanent changes to the genetic material of a cell. The process of mutation introduces new genetic variations and this facilitates the process of evolution. Most biologists believe that adaptation occurs through the accumulation of many mutations that in themselves have only small effects. Neutral mutations do not have an impact on the organism's chance of survival but they accumulate over time and might result in, what is known as, punctuated equilibrium.

In essence, Genetic algorithms have all the features of these evolution mechanisms. If the greater details of all the mechanisms of biological evolution are understood and implemented in genetic algorithms then the efficiency of GAs will increase many fold. In the genetic algorithm presented here the concept of mutation is extended, based on the mutation idea as perceived in the biological field (explained elaborately in Section 4.3). Besides that, the expression of variables as binary strings is also modified to resemble the genetic makeup of a living being. The implementation of some of these evolutionary biology and genetic concepts in the algorithm developed in this chapter, has shown improvement in terms of convergence and quality of the solutions.

4.3 Types of Mutation

Mutations are permanent changes to the genetic material of an organism, which are transferred from one generation to the next. The importance of mutations in the evolution process was investigated by Nei [6] and Li [7]. Molecular studies have shown that mutations include not only nucleotide substitutions but also important processes as gene duplication and recombination. Mutations are considered the driving force of evolution, where less favourable ones are removed by the process of selection and the favourable ones tend to propagate from generation to generation, thereby improving the fitness of individuals in the population. The various types of mutation can be broadly put into three categories namely:

1. Point mutations
2. Large mutations
3. Chromosomal Mutations

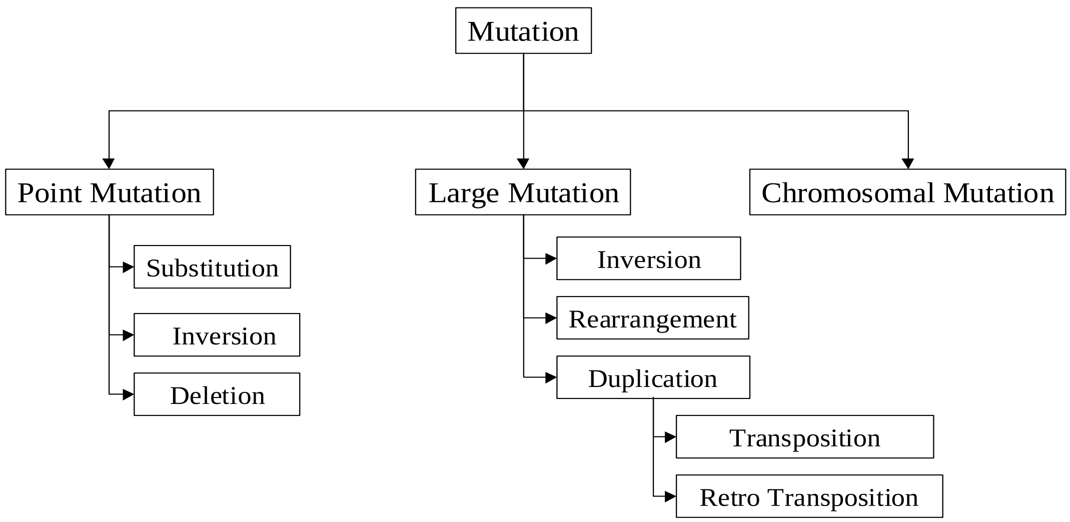


Figure 4.2: Classification of different types of mutation

Each of the above mutations can be further subdivided into various classes. Figure 4.2 gives an over view of the possible mutations. These mutations have been implemented in the NBGA and a brief description of each of these mutations is given below. The way in which the variables are encoded in NBGA is investigated before discussing the implementation of the mutations in detail. For the sake of simplicity a function with only two variables, $f(x, y)$ is considered. The variables x and y are real valued and are bounded between upper and lower limits. For ease of implementation these variables are represented as binary strings. The binary string for each variable is called a ***chromosome*** and each chromosome in turn consists of subsequent strings known as ***genes***. The chromosomes of both the variables x and y are known as the ***chromosomal genome*** (Figure 4.3).

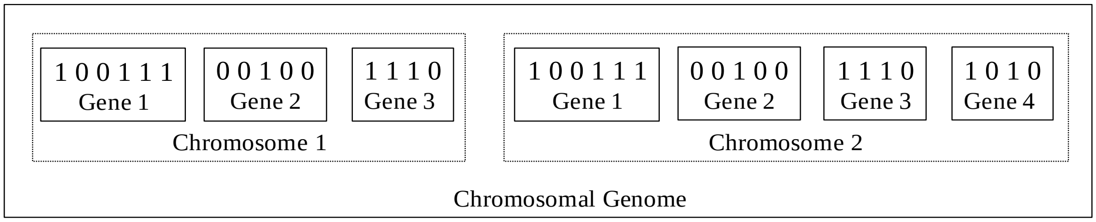


Figure 4.3: Representation of variables in NBGA. The chromosome 1 represents variable x and chromosome 2 represents variable y.

As can be seen from Figure 4.3 the variable x consists of three genes: gene1 is a binary string of length 6, gene2 is a binary string of length 5 and gene 3 is a binary string of length 4. Together these three genes constitute chromosome 1. Similarly, the variable y consists of four genes: gene 1 is of length 6, gene 2 of length 5, gene 3 of length 4 and gene 4 of length 4, and together they constitute chromosome 2 for variable y. The combination of these chromosomes constitutes the chromosomal genome. Having outlined the structure of how the variables are encoded in the NBGA, the explanation of the various mutations and their equivalent in the NBGA are discussed below. The chromosomal genome in Figure 4.3 is taken as the reference for the discussion of mutations below.

4.3.1 Point Mutation

These are changes in the single DNA nucleotide. A point mutation may consist of the *deletion* of a nucleotide, the *insertion* of additional nucleotide or the *substitution* of one nucleotide for another. The deletion type point mutation is shown in Figure 4.4a. In this case a bit from gene1 of chromosome 1 has been deleted. This type of mutation is very common. Figure4.4b shows the case for insertion mutation. In this case a bit has been added to the binary string (marked in bold) of gene 2 from chromosome 2. In Figure4.4 c the substitution mutation is shown. In substitution mutation a bit in gene3 from chromosome 2 is flipped (marked bold). Traditionally this type of mutation has been implemented in most of the genetic algorithms.

4.3.2 Large Mutation

These mutations involve a whole gene at a time. Various types of large mutation that are implemented in the NBGA are: *deletion*, *inversion*, *insertion* and *gene duplication*. Gene duplication can be categorised into *transposition* and *retro transposition*

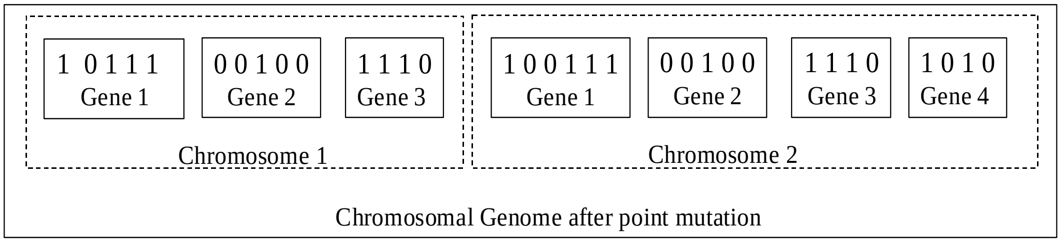


Figure 4.4a: Deletion type point mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

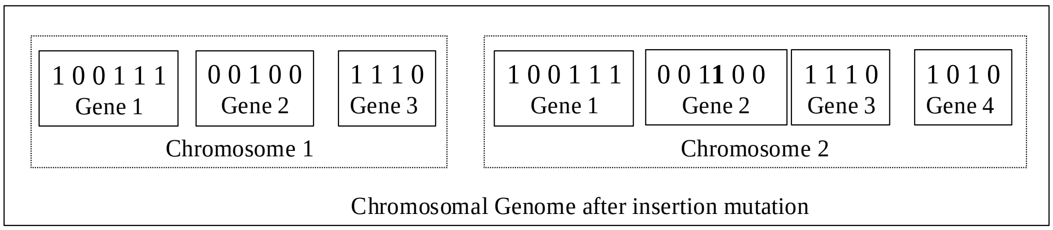


Figure 4.4b: Insertion type point mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

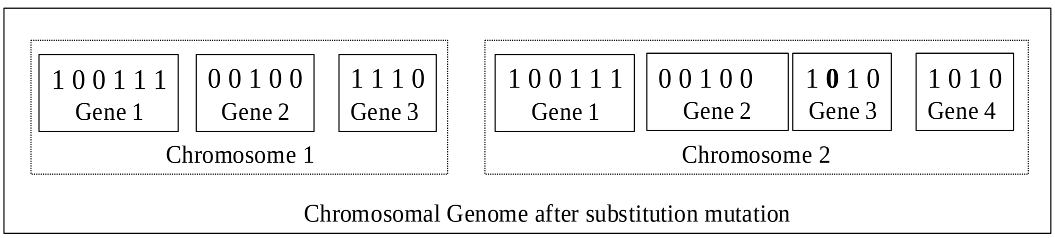


Figure 4.4c: Substitution type point mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

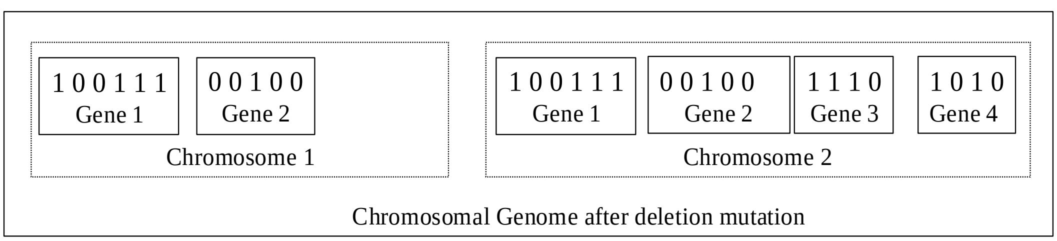


Figure 4.5 a: Deletion type of mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

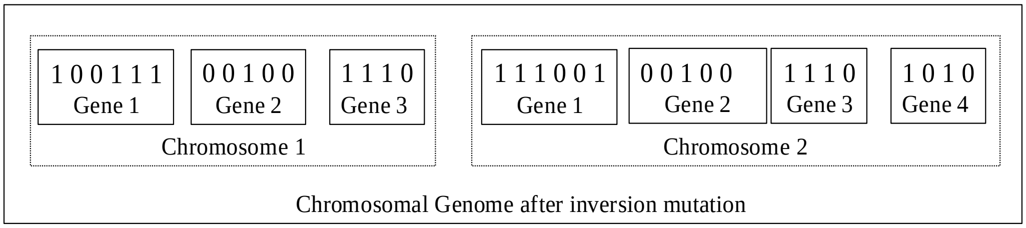


Figure 4.5 b: Inversion type of mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

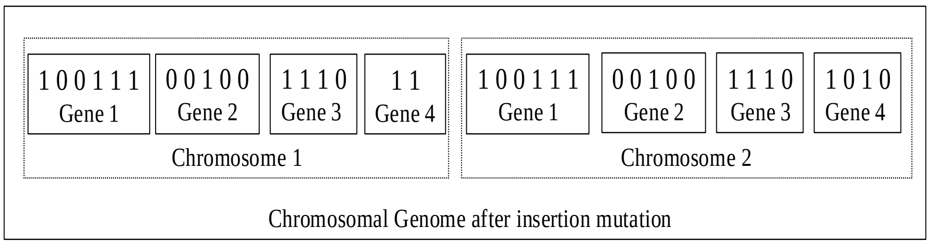


Figure 4.5 c: Insertion type of mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

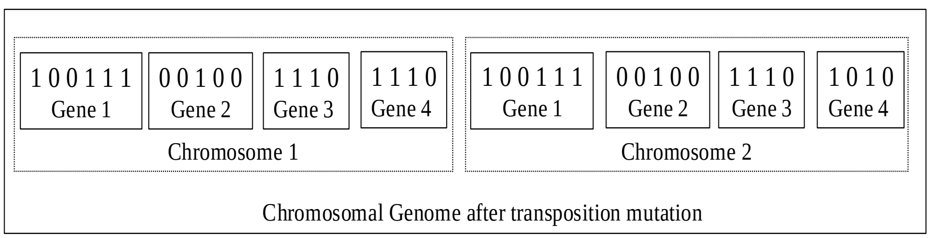


Figure 4.5 d: Transposition type of mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

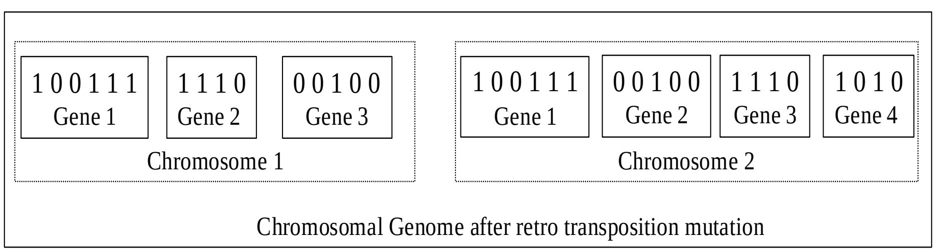


Figure 4.5 e: Retro transposition type of mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

Figure 4.5a shows the deletion type of large mutation. In this case gene 3 from chromosome 1 is deleted (chromosomal genome in Figure 4.3 is used as reference). The inversion mutation is illustrated in Figure 4.5b. The string in gene 1 of chromosome 2 is inverted backwards. In Figure 4.5c a gene is inserted into chromosome 1, indicating that insertion mutation has occurred. Sometimes a whole gene is duplicated and then inserted at random in the chromosomal genome. Such a mutation is known as transposition duplication. Figure 4.5d shows the transposition type of mutation. Gene 3 from chromosome 1 is copied and positions itself next to gene 3. The transposition mutation has been implemented in the jumping gene GA [4]. However the difference between the two approaches is that in the NBGA an attempt is made to mimic the biological process of transposition mutation by representing the members of the population as a set of two chromosomes with different number of genes rather than a continuous binary string. Retro transposition is similar to transposition except that a gene is copied and repositioned in a new position and deleted from its original location. This situation is shown in Figure 4.5e. In this case gene 2 from chromosome 1 is removed from its original position and replaced at the end.

4.3.3 Chromosomal Mutation

These are very large scale mutations and involve whole chromosomes or a piece of them and can alter many genes at a time in that chromosome. They are an important source of new genetic material. Figure 4.6 shows a chromosomal mutation where the gene sequence in chromosome 2 has been inverted.

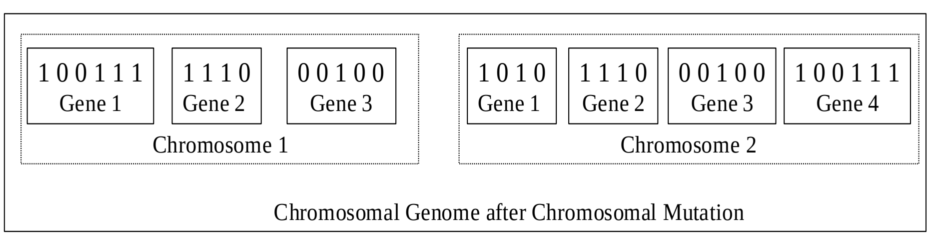


Figure 4.6: Chromosomal mutation. The chromosome 1 represents variable x and chromosome 2 represents variable y.

These are the mutation operators that are used in our genetic algorithm: NBGA. The next section discusses the structure of the NBGA in detail.

4.4 Non-dominated Sorting Biologically Motivated Genetic Algorithm (NBGA)

The non-dominated sorting biologically motivated genetic algorithm (NBGA) implements Pareto ranking. For diversity preservation the crowding distance, as proposed by Deb et. al [8], is used besides the mutation operators described in the previous section. The algorithm of the NBGA is as follows:

1. Randomly initialise the population
2. For i=1 to Members in population
3. Initialise the population (binary string)
4. Initialise the rate of point mutation (binary string)
5. Initialise the rate of large mutation (binary string)
6. Initialise the rate of chromosome mutation (binary string)
7. Decode the population
8. Evaluate the objective functions
9. Classify the population into Pareto Fronts [8]
10. Assign the dummy fitness values [8]
11. Select the parents using tournament selection
12. Perform multipoint crossover
13. Perform the crossover of the chromosomes pertaining to the variables
14. Cross of the binary string representing rate of mutations
15. Perform point mutation
16. Perform large mutation

17. Perform Chromosomal Mutation
18. Combine the offspring and parent population.
19. If termination criteria satisfied then stop else go to step 3

The NBGA differs from Non-dominated Sorting Genetic Algorithm-II (NSGA-II) on the following accounts:

1. The method in which the variables are represented as binary. This method of representation of variables helps in the implementation of the proposed mutation operators.
2. The mutation operators implemented in the NBGA are different from the point mutation used in NSGA-II.

Based on the above mentioned features this algorithm has been named: Non-dominated Sorting Biologically Motivated Genetic algorithm (NBGA). This algorithm uses the principle of non-dominated sorting as used by NSGA-II and the representation of the variables is similar to that of biological systems. In the next section the performance parameters for the NBGA are discussed.

4.5 Performance Parameters for NBGA

An important issue in multi-objective optimisation is the quantitative performance comparison of different algorithms. The most popular comparison methods are based on unary quality measures, i.e. the measure assigns a number to each approximation set that reflects a certain quality aspect. Usually a combination of them is used [9]. Other methods are based on binary quality measures, which assign numbers to pairs of approximation sets [10]. A comprehensive survey of different performance measurement indices is given in [11].

Four performance parameters are considered in order to analyse the performance of the NBGA for the test functions. These performance parameters are discussed in the following subsections.

4.5.1 Generational Distance (GD):

The concept of generational distance (GD) was proposed by Van Valdhuizen and Lamont [9]. The purpose of this parameter is to estimate how far the Pareto front, obtained by a genetic algorithm, is from the actual Pareto front. Mathematically this parameter is defined as

$$GD=\frac{\sqrt{\sum_{i=1}^{p} d_i^2}}{p} \tag{4.1}$$

Where p is the number of points in the Pareto front obtained by the algorithm and d_i is the Euclidean distance between each solution point in the obtained Pareto front and the actual Pareto front for the problem under consideration. If the value of GD is zero then this indicates that all the points obtained by the algorithm lie on the true Pareto front. Any other value of GD will indicate how far the obtained Pareto front is from the actual one.

4.5.2 Spacing (S):

This factor indicates the spread of the solutions obtained. This metric was proposed by Schott [12] and is defined as

$$S = \sqrt{\frac{1}{p-1}\sum_{i=1}^{p}(\bar{d}-d_i)^2} \tag{4.2}$$

Where

$d_i = \min_j (| f_1^i - f_1^j | + | f_2^i - f_2^j |)$ and i, j=1,2,..p, $\bar{d}$ is the mean of all d_i ,and p is the number of non-dominated vectors found so far. If this value is zero then all the points in the obtained Pareto front are equidistantly placed.

4.5.3 Error Ratio (e):

This factor was proposed by Van Valdhuizen and Lamont [9] to determine (in percentage) the number of solutions in the obtained Pareto front that are not members of the true Pareto front. The mathematical definition of this factor is

$$e = \frac{\sum_{i=1}^{p} e_i}{p} \tag{4.3}$$

Where p is the number of points in the obtained Pareto front and $e_i = 0$ if the point i is a member of the true Pareto front, else $e_i = 1$. If $e = 0$ then all the points of the obtained Pareto front lie on the true Pareto front.

4.5.4 The Two Set Coverage (SC)

This measurement index was proposed by Zitzler and Thiele [10] to determine the relative coverage comparison of two sets. For two sets X' and X'', SC is defined as the mapping of the order pair (X' , X'') to the interval [0,1] as

$$SC(X', X'') \triangleq \left|\{a'' \in X''; \exists a' \in X' : a' \preceq a''\}\right| \Big/ \left|X''\right| \tag{4.4}$$

Two set coverage is a binary performance measure index.

Using the above performance parameters the NBGA was evaluated on some of the standard test functions for the sake of comparison. The NBGA was run 30 times on each of the test functions and the average values of the three performance (GD, S, e) parameters are reported here. The parameters of the NBGA used for the analysis in this chapter are as follows:

Number of generations =50

Number of individuals =100

Crossover probability = 80%

Single point crossover was used.

The mutation rate was fixed between 0 and 10%. Since each type of mutation was represented as a binary string and subjected to cross-over the mutation rate for each individual changed during each generation.

These mutation rates were chosen after experimentation with different mutation rates.

4.6 Comparison of NBGA with Other Genetic Algorithms

In this section the NBGA proposed by the authors is compared with other well known genetic algorithms, namely: Non-dominated Sorting Genetic Algorithm II [8], Micro-genetic Algorithm for Multi-objective Optimisation [13] and Pareto Archived Evolution Strategy [14] which is an algorithm based on evolutionary strategy. The performance parameters considered for comparison are generational distance (GD), Spacing (SP), Error ratio (ER) and two set coverage (SC). These parameters are already discussed in Section 4.5. The setting of NBGA is the same as discussed in Section 4.5. The NBGA was run 30 times for each test function and the average value of the performance parameters is reported.

4.6.1 Test Function 1:

This test function was proposed by Kursawe [15]. Mathematically this function is defined as

$$\begin{aligned} \min f_1(\vec{x}) &= \sum_{i=1}^{n-1}\left(-10\exp(-0.2\sqrt{x_i^2 + x_{i+1}^2})\right) \\ \min f_2(\vec{x}) &= \sum_{i=1}^{n}\left(\left|x_i\right|^{0.8} + 5\sin(x_i)^3\right) \end{aligned} \tag{4.5}$$

where n=3, $-5 \le x_1, x_2, x_3 \le 5$ and f_1 and f_2 are functions of variables x_1, x_2 and x_3

The result of this test function is shown in Figure 4.7 and the values of the performance parameters are shown in Table 4.1. The values of the performance parameters for Non-dominated Sorting Genetic Algorithm II, Pareto Archived Evolution Strategy and Micro-genetic Algorithm for Multi-objective Optimization are taken from Coello et. al. [16]. As in Coello et. al. the fitness function was evaluated 12000 times for the sake of comparison.

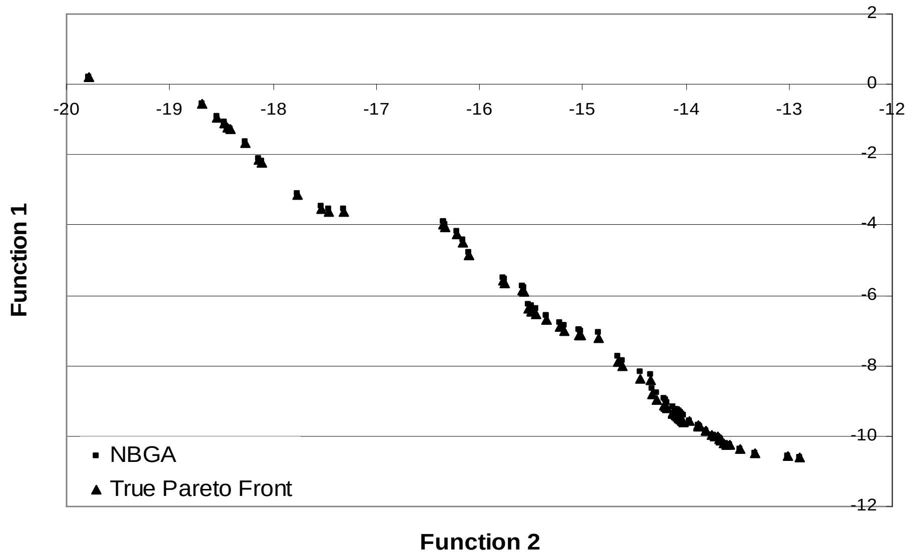


Figure 4.7: Pareto Front produced by the NBGA and actual front for test function 1

Table 4.1a: Result of Error Ratio for test function 1

Error Ratio	NBGA	NSGA II	micro GA	PAES
Best	0.10	0.06	0.18	0.1
Worst	0.33	1.01	0.36	0.68
Average	0.17	0.56	0.27	0.27
Median	0.163	0.495	0.245	0.245
Std. Deviation	0.057	0.384	0.053	0.104

Table 4.1b: Result of Generational Distance for test function 1

Spacing	NBGA	NSGA II	micro GA	PAES
Best	0.03676	0.01842	0.07169	0.06411
Worst	0.10974	0.06571	0.20313	0.34096
Average	0.08622	0.03614	0.12890	0.19753
Median	0.08950	0.03609	0.12666	0.18663
Std. Deviation	0.021	0.010	0.029	0.064

Table 4.1d: Result of Two Set Coverage Measure for test function 1

Generational Distance	NBGA	NSGA II	micro GA	PAES
Best	0.00694	0.006905	0.006803	0.0147
Worst	0.10448	0.103095	0.010344	0.1572
Average	0.05	0.029255	0.008456	0.5491
Median	0.048	0.017357	0.008489	0.0494
Std. Deviation	0.029	0.027	0.00099	0.030

SC	NBGA	Micro GA	NSGA-II	PAES
NBGA	0.00	0.98	0.42	0.75
Micro GA	0.02	0.00	0.42	0.72
NSGA-II	0.04	0.04	0.00	0.68
PAES	0.03	0.03	0.17	0.00

Table 4.1c: Result of Spacing for test function 1

The NBGA is able to cover the entire Pareto Front. From Table 4.1 it can be seen that the NBGA performs better than the other three algorithms in terms of error ratio. For the generational distance performance parameter only micro GA performs better than the NBGA. The NSGA II performs better than the NBGA for spacing performance metric. The better performance of the NSGA-II compared to the NBGA for spacing performance metric and generational distance does not necessarily indicate that the NBGA performs worse compared to the NSGA-II. The spacing metric gauges how evenly the points in the Pareto Set, obtained by a GA, are distributed in the objective space and it is quite possible that the True Pareto front has non-uniform distribution of points, i.e. the True Pareto Front might have higher concentration of solutions at some sections and lower concentration in other. Similarly, the generational distance metric has certain disadvantages. According to this metric it is better to find one solution close to the Pareto front than to find a set of solutions in which many solutions are on the true Pareto front and one solution is a small distance away from the true Pareto Front. Thus evaluation of the performance of any GA, based on spacing metric and generational distance parameter, may lead to erroneous conclusions.

Hence, to compare two GAs the two set convergence metric is more suited. From Table 4.1d it can be seen that $SC(NBGA, MicroGA) = 0.98$ and $SC(MicroGA, NBGA) = 0.02$, since $SC(NBGA, MicroGA) > SC(MicroGA, NBGA)$, hence, the NBGA is relatively better than the MicroGA. Similarly, $SC(NBGA, NSGA-II) = 0.98$ and $SC(NSGA-II, NBGA) = 0.04$,i.e. $SC(NBGA, NSGA-II) > SC(NSGA-II, NBGA)$ hence, it can be concluded that the NBGA again performs relatively better than the NSGA-II. The analysis of the two set coverage measurement between the NBGA and the PAES gives $SC(NBGA, PAES) = 0.75$ and $SC(PAES, NBGA) = 0.03$, since $SC(NBGA, PAES) > SC(PAES, NBGA)$, so the NBGA is *relatively* better than the PAES. From this analysis of the results of the two set coverage measurement it can be concluded that the NBGA performs better for this test function, as compared to the other comparison algorithms.

4.6.2 Test Function 2

This test function was proposed by Kita et.al [17]. Mathematical definition of this problem is as follows

$$\max\ F = \left(f_1(x, y),\ f_2(x, y) \right) \tag{4.6}$$

where

$$f_1(x, y) = -x^2 + y$$
$$f_2(x, y) = \frac{1}{2}x + y + 1$$

subject to

$$0 \geq \frac{1}{6}x + y - \frac{13}{2}$$
$$0 \geq \frac{1}{2}x + y - \frac{15}{2}$$
$$0 \geq 5x + y - 30$$

and $x,\ y \geq 0$. The range used in this case is $0 \leq x,\ y \leq 7$ [17]. The results are shown in Figure 4.8 and Table 4.2. The values of the performance parameters for the NBGA II, the PAES and the MicroGA are taken from Coello et. al. [16]. As in Coello et. al. the fitness function was evaluated 5000 times for the sake of comparison.

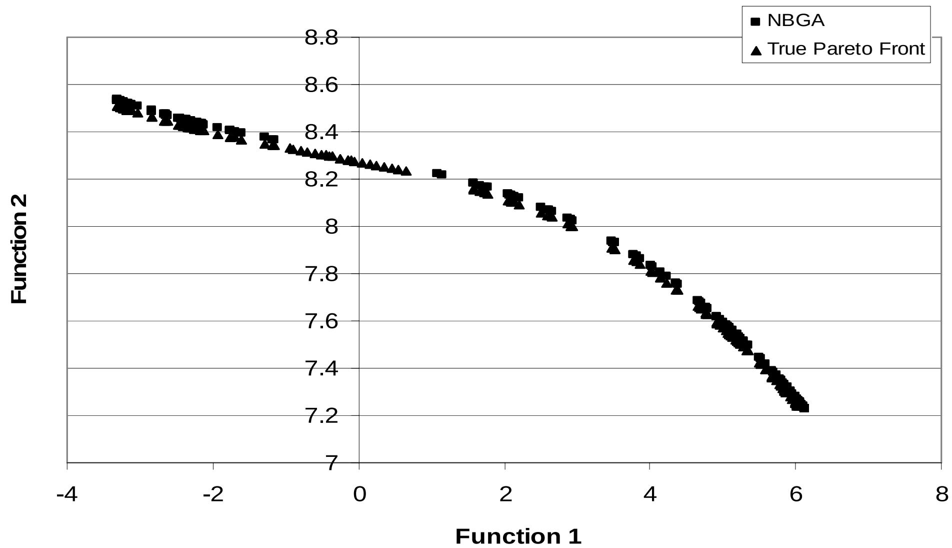


Figure 4.8: Pareto Front produced by the NBGA and actual front for test function 2

Table 4.2a: Result of Error Ratio for test function 2

Error Ratio	NBGA	NSGA II	micro GA	PAES
Best	0.05405	0.75	0.734694	0.93
Worst	0.18	0.99	1.01639	1.01
Average	0.05	0.8965	0.927706	0.993
Median	0.042	0.92	0.936365	1.01
Std. Deviation	0.04282	0.06714	0.06874	0.02536

Table 4.2b: Result of Generational Distance for test function 2

Generational Distance	NBGA	NSGA II	micro GA	PAES
Best	0.04808	0.003885	0.00513	0.0113
Worst	0.11628	0.678449	0.912065	0.9192
Average	0.07	0.084239	0.150763	0.1932
Median	0.066	0.011187	0.089753	0.0333
Std. Deviation	0.02125	0.16524	0.21656	0.24965

Table 4.2c: Result of Spacing for test function 2

Spacing	NBGA	NSGA II	micro GA	PAES
Best	0.00319	0.00103	0.06561	0.00667
Worst	0.01714	1.48868	1.64386	0.43287
Average	0.00992	0.09849	0.31502	0.11010
Median	0.01045	0.02717	0.12977	0.08200
Std. Deviation	0.00416	0.32738	0.42174	0.09960

Table 4.2d: Result of Two Set Coverage Measure for test function 2

SC	NBGA	Micro GA	NSGA-II	PAES
NBGA	0.00	0.73	0.63	0.64
Micro GA	0.12	0.00	0.26	0.37
NSGA-II	0.21	0.08	0.00	1.00
PAES	0.21	0.21	0.00	0.00

From Table 4.2d it can be seen that $SC(NBGA, MicroGA) = 0.73$ and $SC(MicroGA, NBGA) = 0.12$,since $SC(NBGA, MicroGA) > SC(MicroGA, NBGA)$, hence, the NBGA is relatively better than the MicroGA. Similarly, $SC(NBGA, NSGA-II) = 0.63$ and $SC(NSGA-II, NBGA) = 0.21$, i.e. $SC(NBGA, NSGA-II) > SC(NSGA-II, NBGA)$ hence, it can be concluded that the NBGA again performs relatively better than the NSGA-II. The analysis of the two set coverage measurement between the NBGA and the PAES gives $SC(NBGA, PAES) = 0.64$ and $SC(PAES, NBGA) = 0.21$, since $SC(NBGA, PAES) > SC(PAES, NBGA)$, so the NBGA is *relatively* better than the PAES. From this analysis of the results of the two set coverage measurement it can be concluded that the NBGA performs better for this test function as compared to the other comparison algorithms. The performance of the NBGA, based on error ratio, generational distance and spacing metric, is better than the MicroGA, the NSGA-II and the PAES (Table 4.2a-4.2c).

4.6.3 Test Function 3

This test function was proposed by Deb [18]. Mathematically this function is represented as

$$\begin{aligned} &\min\ f_1(x_1,x_2)=x_1 \\ &\min\ f_2(x_1,x_2)=g(x_1,x_2)\cdot h(x_1,x_2) \end{aligned} \tag{4.7}$$

where $g(x_1,x_2)=11+x_2^2-10\cos(2\pi x_2)$

$$h(x_1,x_2)=\begin{cases} 1-\sqrt{\dfrac{f_1(x_1,x_2)}{g(x_1,x_2)}}, & \text{if } f_1(x_1,x_2)\le g(x_1,x_2) \\ 0, & \text{otherwise} \end{cases}$$

and $0\le x_1\le 1,\ -30\le x_2\le 30$.

The result of this test function is shown in Figure 4.9. The values of the performance parameters are given in Table 4.3. The values of the performance parameters for the NBGA II, the PAES and the MicroGA are taken from Coello et. al [16] . As in Coello et. al. the fitness function was evaluated 5000 times for the sake of comparison.

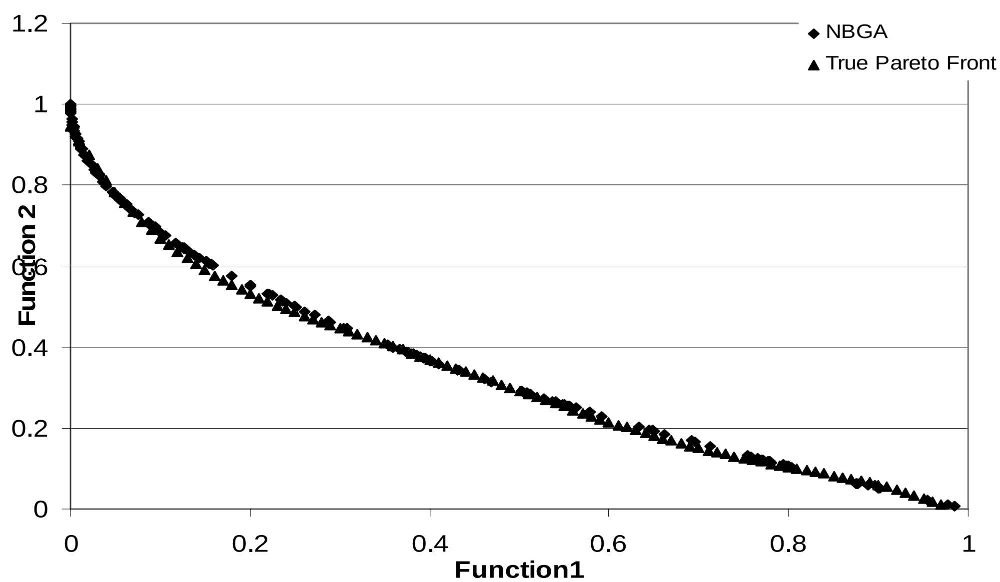


Figure 4.9: Pareto Front produced by the NBGA and actual front for test function 3

Table 4.3a: Result of Error Ratio for test function 3

Error Ratio	NBGA	NSGA II	micro GA	PAES
Best	0.0070067	0	0.02	0.06
Worst	0.0186447	1.01	1.04545	1.01
Average	0.0113304	0.35	0.2568	0.4485
Median	0.0112476	0.2	0.19	0.24
Std. Deviation	0.0035859	0.39615	0.25646	0.38199

Table 4.3b: Result of Generational Distance for test function 3

Gen. Dist.	NBGA	NSGA II	micro GA	PAES
Best	0.008621	0.000133	8.74×10^{-5}	0.000114
Worst	0.073171	0.163146	0.811403	1.99851
Average	0.03	0.023046	0.047049	0.163484
Median	0.028	0.000418	0.000236	0.058896
Std. Deviation	0.02072	0.04543	0.18116	0.44130

Table 4.3c: Result of Spacing for test function 3

Spacing	NBGA	NSGA II	micro GA	PAES
Best	0.00035	0.00021	0.00760	0.00916
Worst	0.00170	0.01023	5.56270	19.88640
Average	0.00087	0.00369	0.34166	1.11462
Median	0.00075	0.00209	0.29950	0.01876
Std. Deviation	0.00037	0.00337	1.24756	4.43459

Table 4.3d: Result of Two Set Coverage Measure for test function 3

SC	NBGA	Micro GA	NSGA-II	PAES
NBGA	0.00	1.00	0.49	1
Micro GA	0.00	0.00	0.00	0.36
NSGA-II	0.50	1.00	0.00	0.29
PAES	0	0.00	0.00	0.00

From the above figure it can be seen that the NBGA was able to cover the entire Pareto front. From Table 4.3a it can be seen that in terms of error ratio the NBGA performs better than other comparison algorithms. In terms of generational distance the NSGA-II performs better than the other algorithms (Table 4.3b). Based on spacing performance index the NBGA performs better than other algorithms (Table 4.3c). Hence based on the unary performance metric, the NBGA performs better than other algorithms in terms of error ratio and spacing. The NSGA-II has better performance in terms of generational distance.

From Table 4.3d it can be seen that $SC(NBGA, MicroGA) = 1$ and $SC(MicroGA, NBGA) = 0$, since $SC(NBGA, MicroGA) > SC(MicroGA, NBGA)$, hence, the NBGA is relatively better than the MicroGA. Similarly, $SC(NBGA, NSGA-II) = 0.49$ and $SC(NSGA-II, NBGA) = 0.18$, i.e. $SC(NBGA, NSGA-II) > SC(NSGA-II, NBGA)$, hence, it can be concluded that the NBGA performs relatively better than the NSGA-II. The analysis of the two set coverage measurement between the NBGA and the PAES gives $SC(NBGA, PAES) = 1$ and $SC(PAES, NBGA) = 0$, since $SC(NBGA, PAES) > SC(PAES, NBGA)$, so the NBGA is *relatively* better than the PAES. From this analysis of the results of two set coverage measure it can be concluded that the NBGA performs better for this test function as compared to the other comparison algorithms.

4.6.4 Test Function 4

This test function was proposed by Deb [18]. The mathematical form of this test function is

$$\begin{aligned} &\min\ f_1(x_1, x_2) = x_1 \\ &\min\ f_2(x_1, x_2) = \frac{g(x_2)}{x_1} \end{aligned} \tag{4.8}$$

where

$$g(x_2) = 2 - \exp\left\{-\left(\frac{x_2 - 0.2}{0.004}\right)^2\right\} - 0.8\exp\left\{-\left(\frac{x_2 - 0.6}{0.4}\right)^2\right\}$$

and $0.1 \le x_1 \le 1.0,\ 0.1 \le x_2 \le 0.1$.

The result of this test function is shown in Figure 4.10 and the values of the performance metric are given in Table 4.4. The values of the performance parameters for the NBGA II, the PAES and the MicroGA are taken from Coello et. al [16]. As in Coello et. al. the fitness function was evaluated 10000 times for the sake of comparison.

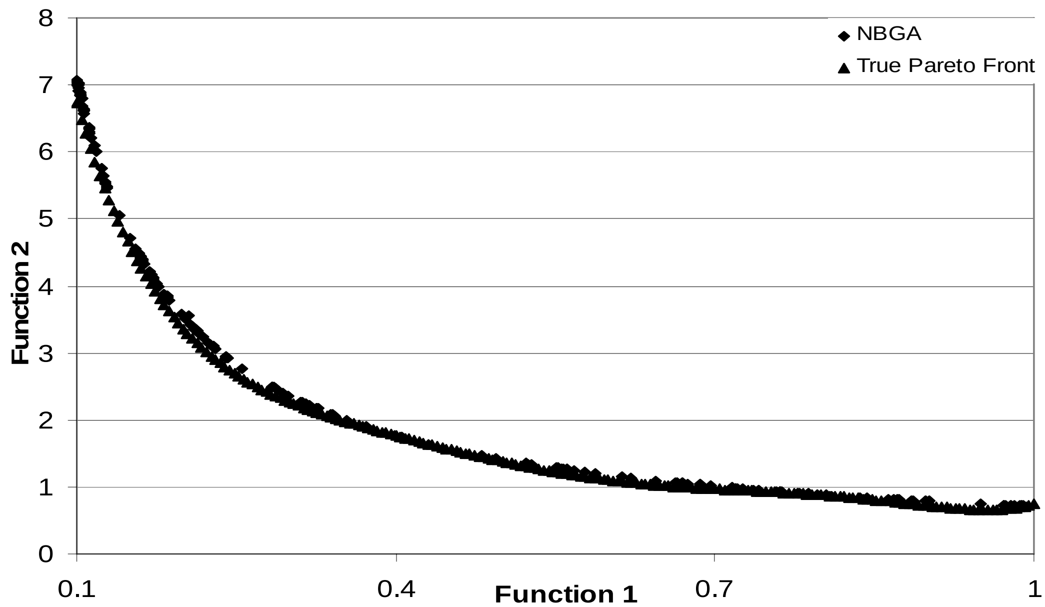


Figure 4.10: Pareto Front produced by the NBGA and actual front for test function 4

Table 4.4a: Result of Error Ratio for test function 4

Error Ratio	NBGA	NSGA II	micro GA	PAES
Best	0.036251	0.02	0.08	0.02
Worst	0.14	1.01	1.01	1.01
Average	0.06	0.4145	0.252	0.489
Median	0.060	0.115	0.16	0.28
Std. Deviation	0.025	0.459	0.231	0.438

Table 4.4b: Result of Generational Distance for test function 4

Generation Distance	NBGA	NSGA II	micro GA	PAES
Best	0.00806	0.0007	0.00047	0.00045
Worst	0.02718	0.20847	0.1835	0.22167
Average	0.01274	0.04424	0.04347	0.19476
Median	0.016	0.00086	0.05004	0.07036
Std. Deviation	0.007	0.073	0.048	0.204

Table 4.4c: Result of Spacing for test function 4

Spacing	NBGA	NSGA II	micro GA	PAES
Best	0.00031	0.02609	0.03027	0.04784
Worst	0.08113	0.06142	0.81764	0.66468
Average	0.08185	0.03745	0.21358	0.19477
Median	0.05368	0.03553	0.06301	0.07037
Std. Deviation	0.084	0.009	0.250	0.204

Table 4.4d: Result of Two Set Coverage Measure for test function 4

SC	NBGA	Micro GA	NSGA-II	PAES
NBGA	0.00	1.00	0.75	0.62
Micro GA	0.00	0.00	0.04	0.46
NSGA-II	0.20	0.06	0.00	0.43
PAES	0.15	0.00	0.01	0.00

Figure 4.10 and Table 4.4a - Table 4.4c show that the NBGA was able to cover the entire Pareto front and its performance was better than the other three algorithms in terms of all the performance parameters. From Table 4.4d it can be seen that $SC(NBGA, MicroGA)=1$ and $SC(MicroGA, NBGA)=0$, since $SC(NBGA, MicroGA) > SC(MicroGA, NBGA)$, hence, the NBGA is relatively better than the MicroGA. Similarly, $SC(NBGA, NSGA-II)=0.75$ and

$SC(NSGA-II, NBGA) = 0.2$,i.e. $SC(NBGA, NSGA-II) > SC(NSGA-II, NBGA)$, hence, it can be concluded that the NBGA again performs relatively better than the NSGA-II. The analysis of two set coverage measure between the NBGA and the PAES gives $SC(NBGA, PAES) = 0.62$ and $SC(PAES, NBGA) = 0.15$, since $SC(NBGA, PAES) > SC(PAES, NBGA)$, so the NBGA is *relatively* better than the PAES. From this analysis of the results of the two set coverage measurement it can be concluded that the NBGA performs better for this test function as compared to the other comparison algorithms.

4.7 Performance of NBGA on Multivariable Test Functions

Deb [18] has identified several features that may cause difficulties for multi-objective GAs, e.g.: 1) converging to the Pareto optimal front and 2) maintaining diversity within the population. Concerning the first issue, multimodality, deception, and isolated optima are well known problem areas in single-objective evolutionary optimization. The second issue is important in order to achieve a well distributed non-dominated front. However, certain characteristics of the Pareto optimal front may prevent a GA from finding diverse Pareto optimal solutions like convexity or non-convexity, discreteness, and non-uniformity. For each of the six problem features (multimodality, deception, isolated optima non-convexity, discreteness, and non-uniformity), a corresponding test function is constructed, following the guidelines in Deb [18].

In the following analysis the population size was 200 and 100 iterations were performed; other parameters remain the same as discussed in Section 4.5.

4.7.1 Test Function 5

The test function is described mathematically as [18]:

$$\begin{aligned} &\min\ f_1(x_1, x_2) = x_1 \\ &\min\ f_2(x_1, x_2) = g(\vec{x})h(f_1, g) \end{aligned} \tag{4.9}$$

where

$$g(x) = 1 + 9 \sum_{i=2}^{m} x_i / (m-1)$$
$$h(f_1, g) = 1 - \sqrt{f_1 / g}$$
and

$m = 30$ and $x_i \in [0,1]$

The result of this test function is shown in Figure 4.11 and the values of the performance metric are given in Table 4.5.

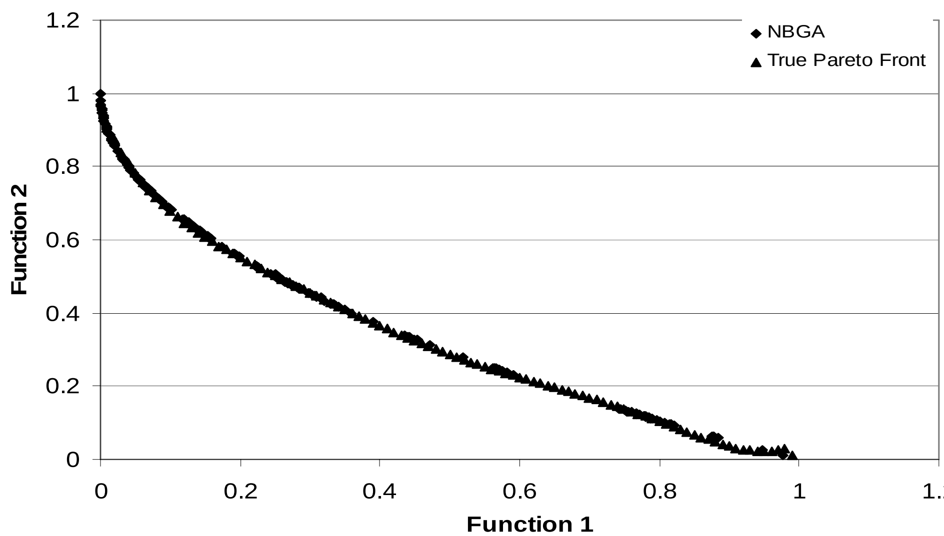


Figure 4.11: Pareto Front produced by the NBGA and actual front for multivariable test function 5

Table 4.5a: Result of Error Ratio for test function 5

Error Ratio	NBGA	NSGA II	micro GA	PAES
Best	0.0047	0.0093	0.0102	0.0111
Worst	0.0367	0.0732	0.0804	0.0877
Average	0.0147	0.0293	0.0322	0.0351
Median	0.0121	0.0241	0.0265	0.0289
Std. Deviation	0.010	0.020	0.022	0.024

Table 4.5b: Result of Generational Distance for test function 5

Generational Distance	NBGA	NSGA II	micro GA	PAES
Best	0.0051	0.0101	0.0101	0.0102
Worst	0.0758	0.1513	0.1506	0.1521
Average	0.0367	0.0734	0.0730	0.0738
Median	0.0334	0.0667	0.0664	0.0670
Std. Deviation	0.020	0.040	0.039	0.040

Table 4.5c: Result of Spacing for test function 5

Spacing	NBGA	NSGA II	micro GA	PAES
Best	0.00064	0.00128	0.00141	0.00154
Worst	0.00126	0.00251	0.00276	0.00301
Average	0.00083	0.00165	0.00182	0.00198
Median	0.00079	0.00157	0.00173	0.00188
Std. Deviation	0.0001	0.0003	0.0003	0.0004

Table 4.5d: Result of Two Set Coverage Measure for test function 5

SC	NBGA	Micro GA	NSGA-II	PAES
NBGA	0.00	0.44	0.99	1.00
Micro GA	0.06	0.00	0.98	1.00
NSGA-II	0.02	0.02	0.00	0.39
PAES	0.00	0.05	0.63	0.00

In terms of error ratio (Table 4.5a), Generation distance (Table 4.5b) and spacing (Table 4.5c), the NBGA performs better than the other algorithms. From Table 4.5d it can be seen that $SC(NBGA, MicroGA) = 0.44$ and $SC(MicroGA, NBGA) = 0.06$, since $SC(NBGA, MicroGA) > SC(MicroGA, NBGA)$, hence the NBGA is relatively better than the MicroGA. Similarly, $SC(NBGA, NSGA-II) = 0.99$ and $SC(NSGA-II, NBGA) = 0.02$,i.e.

$SC(NBGA, NSGA-II) > SC(NSGA-II, NBGA)$ hence, it can be concluded that the NBGA again performs relatively better than the NSGA-II. The analysis of the two set coverage measurement between the NBGA and the PAES gives $SC(NBGA, PAES) = 1$ and $SC(PAES, NBGA) = 0$, since $SC(NBGA, PAES) > SC(PAES, NBGA)$, so the NBGA is *relatively* better than the PAES. From this analysis of the results of the two set coverage measurement it can be concluded that the NBGA performs better for this test function as compared to the other comparison algorithms.

4.7.2 Test Function 6

The test function is described mathematically as [19]:

$$\begin{aligned} &\min\ f_1(x_1, x_2) = x_1 \\ &\min\ f_2(x_1, x_2) = g(\vec{x})h(f_1, g) \end{aligned} \tag{4.10}$$

where
$$g(x) = 1 + 9 \sum_{i=2}^{m} x_i / (m-1)$$
$$h(f_1, g) = 1 - \left(f_1 / g\right)^2$$

and $m = 30$ and $x_i \in [0,1]$

The result of this test function is shown in Figure 4.12 and the values of the performance metric are given in Table 4.6.

Table 4.6a: Result of Error Ratio for test function 6

Error Ratio	NBGA	NSGA II	micro GA	PAES
Best	0.00178	0.00196	0.00196	0.00194
Worst	0.02913	0.03204	0.03201	0.03178
Average	0.01182	0.01300	0.01298	0.01289
Median	0.00943	0.01037	0.01036	0.01029
Std. Deviation	0.008	0.008	0.008	0.008

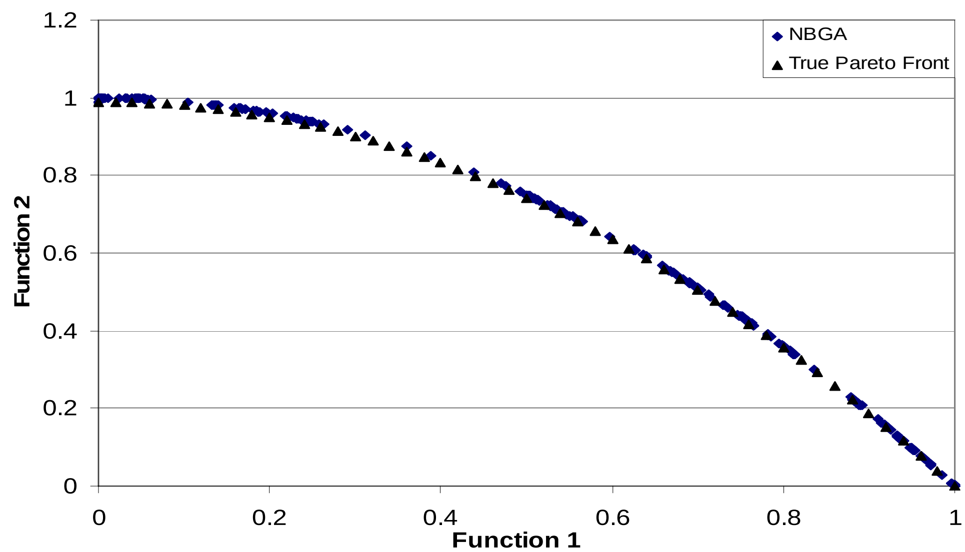


Figure 4.12: Pareto Front produced by the NBGA and actual front for multivariable test function 6

Table 4.6b: Result of Generational Distance for test function 6

Generational Distance	NBGA	NSGA II	micro GA	PAES
Best	0.00503	0.30130	0.30063	0.29722
Worst	0.30435	0.30130	0.30063	0.29722
Average	0.03003	0.02973	0.02966	0.02933
Median	0.00559	0.00553	0.00552	0.00546
Std. Deviation	0.075	0.075	0.075	0.074

Table 4.6c: Result of Spacing for test function 6

Spacing	NBGA	NSGA II	micro GA	PAES
Best	0.00011	0.00012	0.00014	0.00026
Worst	0.00533	0.00586	0.00644	0.01228
Average	0.00100	0.00110	0.00121	0.00230
Median	0.00056	0.00062	0.00068	0.00130
Std. Deviation	0.001	0.0015	0.0017	0.003

Table 4.6d: Result of Two Set Coverage Measure for test function 6

SC	NBGA	Micro GA	NSGA-II	PAES
NBGA	0.0	0.68	0.73	0.91
Micro GA	0.5	0.0	0.5	0.5
NSGA-II	0.42	0.41	0.0	0.5
PAES	0.037	0.32	0.42	0.0

In terms of error ratio (Table 4.6a) the NBGA performs better than the other algorithms. The MicroGA performs best in terms of generation distance (Table 4.6b) and in terms of spacing (Table 4.6c), the NBGA performs better than the other algorithms. From Table 4.6d it can be seen that $SC(NBGA, MicroGA) = 0.68$ and $SC(MicroGA, NBGA) = 0.5$,since $SC(NBGA, MicroGA) > SC(MicroGA, NBGA)$, hence the NBGA is relatively better than the MicroGA. Similarly, $SC(NBGA, NSGA-II) = 0.73$ and $SC(NSGA-II, NBGA) = 0.42$,i.e. $SC(NBGA, NSGA-II) > SC(NSGA-II, NBGA)$ hence, it can be concluded that the NBGA again performs relatively better than the NSGA-II. The analysis of the two set coverage measurement between the NBGA and the PAES gives $SC(NBGA, PAES) = 0.91$ and $SC(PAES, NBGA) = 0.037$, since $SC(NBGA, PAES) > SC(PAES, NBGA)$, so the NBGA is *relatively* better than the PAES. From this analysis of the results of the two set coverage measurement it can be concluded that the NBGA performs better for this test function as compared to the other comparison algorithms.

4.7.3 Test Function 7

The test function is described as [19]:

$$\begin{aligned} &\min\ f_1(x_1, x_2) = x_1 \\ &\min\ f_2(x_1, x_2) = g(\vec{x})h(f_1, g) \end{aligned} \tag{4.11}$$

where $g(x) = 1 + 9\sum_{i=2}^{m} x_i / (m-1)$ and $h(f_1, g) = 1 - \sqrt{f_1 / g} - (f_1 / g)\sin(10\pi f_1)$

and $m = 30$ and $x_i \in [0,1]$

The result of this test function is shown in Figure 4.13 and the values of the performance metric are given in Table 4.7.

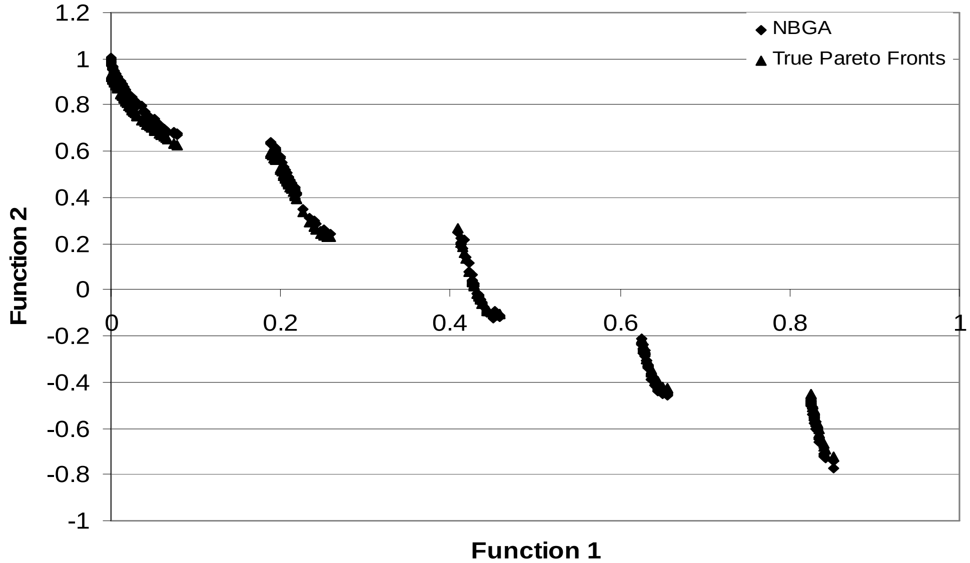


Figure 4.13: Pareto Front produced by the NBGA and actual front for multivariable test function 7

Table 4.7a: Result of Error Ratio for test function 7

Error Ratio	NBGA	NSGA II	micro GA	PAES
Best	0.00178	0.00196	0.00195	0.00196
Worst	0.02777	0.03052	0.03039	0.03048
Average	0.01263	0.01388	0.01382	0.01386
Median	0.00994	0.01092	0.01088	0.01091
Std. Deviation	0.006	0.007	0.007	0.007

Table 4.7b: Generational Distance of Error Ratio for test function 7

Generational Distance	NBGA	NSGA II	micro GA	PAES
Best	0.00503	0.00552	0.00603	0.00660
Worst	0.30435	0.33403	0.36498	0.40003
Average	0.05602	0.06149	0.06718	0.07363
Median	0.02764	0.03034	0.03315	0.03633
Std. Deviation	0.083	0.091	0.099	0.109

Table 4.7c: Spacing Distance of Error Ratio for test function 7

Spacing	NBGA	NSGA II	micro GA	PAES
Best	0.00012	0.00014	0.00015	0.00016
Worst	0.05134	0.05630	0.06128	0.06695
Average	0.01521	0.01668	0.01816	0.01984
Median	0.01524	0.01672	0.01820	0.01988
Std. Deviation	0.01478	0.01620	0.01764	0.01927

Table 4.7d: Result of Two Set Coverage Measure for test function 7

SC	NBGA	MicroGA	NSGA-II	PAES
NBGA	0	0.68	0.66	0.68
Micro GA	0.28	0	0.65	0.66
NSGA-II	0.29	0.28	0	0.65
PAES	0.26	0.289	0.28	0

In terms of error ratio (Table 4.7a), generation distance (Table 4.7b) and spacing (Table 4.7c), the NBGA performs better than the other algorithms. From Table 4.7d it can be seen that $SC(NBGA, MicroGA) = 0.68$ and $SC(MicroGA, NBGA) = 0.28$, since $SC(NBGA, MicroGA) > SC(MicroGA, NBGA)$, hence the NBGA is relatively better than the Micro-GA. Similarly, $SC(NBGA, NSGA-II) = 0.66$ and $SC(NSGA-II, NBGA) = 0.29$,i.e. $SC(NBGA, NSGA-II) > SC(NSGA-II, NBGA)$ hence, it can be concluded that the NBGA again performs relatively better than the NSGA-II. The analysis of the two set coverage measurement between the NBGA and the PAES gives $SC(NBGA, PAES) = 0.68$ and $SC(PAES, NBGA) = 0.26$, since $SC(NBGA, PAES) > SC(PAES, NBGA)$, so the NBGA is *relatively* better than the PAES. From this analysis of the results of two set coverage measure it can be concluded that the NBGA performs better for this test function as compared to the other comparison algorithms.

4.8 Experimental Evidence of Importance of Mutations in Performance of NBGA

These experiments were designed to compare the performance of the NBGA, proposed in this paper, with and without different types of mutations (point mutation, large mutation and chromosome mutation). In this work the first four test functions out of 7 functions, discussed in the previous sections (4.6 and 4.7), have been used together with the three performance parameters: error ratio, generational distance and spacing. The following eight computational experiments were performed:

1. ***With Mutation***: In this case the NBGA was run with all the types of mutations and the performance parameters for the test functions, defined in Section 4.6, were considered.
2. ***Without Mutation***: The NBGA was run without Point mutation, Large mutation and Chromosome mutation. The performance parameters for the test functions defined in Section 4.6 were considered.
3. ***Without Point Mutation:*** The NBGA was run without Point mutation and other mutations (Large and Chromosome mutation) were engaged. The performance parameters for the test functions defined in Section 4.6 were considered.
4. ***Without Large Mutation:*** The NBGA was run without Large mutation and other mutations (Point and Chromosome mutation) were engaged. The performance parameters for the test functions defined in Section 4.6 were considered.
5. ***Without Chromosome Mutation:*** The NBGA was run without Chromosome mutation and other mutations (Point and Large mutation) were engaged. The performance parameters for the test functions defined in Section 4.6 were considered.
6. ***Without Point and Large Mutation:*** The NBGA was run without Point and Large mutation and Chromosome mutation was engaged. The performance parameters for the test functions defined in Section 4.6 were considered.
7. ***Without Point and Chromosome Mutation:*** The NBGA was run without Point and Chromosome mutation and Large mutation was engaged. The performance parameters for the test functions defined in Section 4.6 were considered.

8. ***Without Chromosome and Large Mutation:*** The NBGA was run without Chromosome and Large mutation and point mutation was engaged. The performance parameters for the test functions defined in Section 4.6 were considered.

These experiments were designed to establish whether the mutation operators implemented in the NBGA played a significant role, or not. The results of the experiments are summarised in Appendix C.

From the results summarized in Appendix C it is evident that when the NBGA is run with all the mutation operators best results are obtained. From the results of this section it is evident that the performance of the NBGA is best, for all the test functions, when all the mutation operators are used.

4.9 Conclusions

In this chapter the concept of mutation was introduced in genetic algorithms. These mutations are well studied in the field of evolutionary biology. Also in evolutionary biology, it is gradually being established that mutations are one of the prime sources of diversity in nature. A simplified implementation of these mutations is done in the NBGA proposed here. The performance of the NBGA on various test functions was better than the other genetic algorithms. Furthermore, the influence of these mutation operators was validated by a series of experiments. These experiments prove that mutations improve the performance of the NBGA. The future direction of work will be to investigate the impact of the rate of mutation and the rate of reproduction on the performance of the NBGA. Furthermore, the study regarding impact of mass extinction on the performance of the NBGA will also be of interest.

After having presented the NBGA and its aspects in detail, in the next chapter various issues modelling of the system to be designed are discussed. Since modelling is an important feature in PDM and the success of the design of the system depends on the proper modelling, a proper modelling approach is very important. In the next chapter the classification of the models and their suitability in different phases of PDM is discussed.

4.10 References

[1] H. Aguirre and K. Tanaka, "Selection, Drift, Recombination, and Mutation in Multiobjective Evolutionary Algorithms on Scalable MNK-Landscapes," presented at Third International Conference on Evolutionary Multi-Criteria Optimization, EMO 2005, Guanajuato, Mexico, 2005.

[2] D. E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning," Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.

[3] C. A. Brizuela and R. Aceves, "Experimental Genetic Operators Analysis for the Multi-objective Permutation Flowshop," presented at Second International Conference on Evolutionary Multi-Criteria Optimization, EMO 2003, Faro, Portugal, 2003.

[4] T. M. Chan, K. F. Man, K. S. Tang, and S. Kwong, "A jumping gene algorithm for multiobjective resource management in wideband CDMA systems," Computer Journal, vol. 48, pp. 749-768, 2005.

[5] C. Darwin and J. Huxely, "Origin of Species," Signet Classics.

[6] M. Nei, "Molecular Evolutionary Genetics," Columbia University Press, New York, 1986.

[7] W. H. Li, "Molecular Evolution," Sinauer, Sunderland, MA, 1993.

[8] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist nondominated sorting genetic algorithm for multi-objective optimization: NSGA-II," Proceedings of the Parallel Problem Solving from Nature VI(PPSN-VI), pp. 849-858, 2000.

[9] D. A. V. Veldhuizen and G. B. Lamont, "On measuring multiobjective evolutionary algorithm performance," Congress on Evolutionary Computation (CEC 2000), vol. 1, pp. 204-211, 2000.

[10] E. Zitzler and L. Thiele, "Multiobjective Optimization Using Evolutionary Algorithms-A Comparitive Study," Parallel Problem Solving from Nature, vol. 5, pp. 292-301, 1998.

[11] J. D. Bagley, "The Behavior of Adaptive Systems Which Employ Genetic and Correlation Algorithms,," PhD thesis, University of Michigan, 1967.

[12] J. R. Schott, "Fault tolerant design using single and multicriteria genetic algorithm optimization," M.S. thesis, Dept. Aeronautics and Astronautics, MIT, Cambridge, 1995.

[13] C. A. C. Coello, "A micro-genetic algorithm for multiobjective optimization," Lecture Notes in Computer Science, 1993.

[14] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," Evolutionary Computation, vol. 8, pp. 149-172, 2000.

[15] F. Kursawe, "A variant of evolution strategies for vector optimization," in Lecture Notes in Computer Science, vol. 496, H. P. S. a. R. Männer, Ed.: Springer Verlag, 1991, pp. 193-197.

[16] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling Multiple Objectives with Particle Swarm Optimization," IEEE Transactions on Evolutionary Computation, vol. 8, pp. 256-279, 2004.

[17] H. Kita, Y. Yabumoto, N. Mori, and Y. Nishikawa, "Multi-objective optimization by means of the thermodynamical genetic algorithm," Parallel Problem Solving From Nature- PPSN IV, H.-M Voigt, W. Ebeling, I. Rechenberg, and H.-P Schwefel, Eds. Berlin Germany: Springer Verlag, pp. 504-512, 1996.

[18] K. Deb, "Multiobjective genetic algorithms: problem difficulties and construction of test problems," Evol. Comput, vol. 7, pp. 205-230.

[19] E. Zitzler, K.Deb, and L. Thiele, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results," Evol. Comput, vol. 8, pp. 173-195, 2000.

Chapter 5

Engineering System Models Suitable for PDM

5.1 Introduction

Modelling and simulation enables designers to test whether design specifications are met by virtual rather than physical experiments. The use of virtual prototypes significantly shortens the design cycle and reduces the cost of design. It further provides the designer with immediate feedback on design decisions which, in turn, promises a more comprehensive exploration of design alternatives and a better final design. Virtual prototypes need to model the behaviour of the equivalent physical prototype adequately and accurately, otherwise the predicted behaviour does not match the actual behaviour resulting in poor design decisions. A model is a representation of an actual system. The model should be complex enough to answer the questions raised during the design process, but should not too complex.

The issue of modelling becomes very important when an engineering system is to be designed and optimised. The success of design and optimization will depend on the models of the system. The progressive Design Methodology (PDM) involves models of the system to be designed. The success of PDM also depends on the models of the system that is to be designed. With this in view different aspects of modelling are discussed in this chapter.

5.2 Analysis versus Design Models

Analysis of a system means investigation, under specified condition, of the performance of a system whose mathematical model is known. Design of a system refers to the process of finding a system that accomplishes a given task.

Engineering analysis involves the application of engineering formulae and calculation methods to predict the behaviour of a given system. The calculation methods might be applied in hand calculations, spreadsheets or modelling software [1]. Such systems are satisfactory for evaluating the performance of an existing system or design. However applying engineering analysis to a new design is difficult because analysis cannot yield a design of the system.

Models for design have the same elements as in general system models: design variables, parameters and constants. To determine how these quantities relate to each other for proper performance of function of the design analysis must first be conducted. In order to predict the overall performance of the design a model has to be constructed that incorporates the results of the analyses. In modelling distinction must be made between analysis models and design models. Analysis models are developed based on the principles of science whereas design models are constructed from the analysis models for specific prediction tasks and are problem dependent.

As an example, a simple situation of the design of Permanent Magnet motor is considered. The general sizing equation of a radial flux Permanent Magnet motor is [2]

$$P_R = \frac{1}{1+K_\phi} K_e K_i K_p \eta B_g A \frac{f}{p} \lambda_0^2 D_0^2 L_e \tag{5.1}$$

where P_R is the output power of the motor

K_ϕ is the ratio of electrical loading on the stator and rotor

K_e is the induced voltage factor

K_i is the current waveform factor

K_p is the electrical power waveform factor

η is the efficiency of the motor

B_g is the flux density in the air gap

A is the total electric loading

f is the frequency of the power supply

p is the number of poles of the motor

λ_0 is the diameter ratio of the motor

D_0 is the outer diameter of the motor

L_e is the length of the motor.

Equation 5.1 is a very simple model of a permanent magnet motor. For a given motor all the parameters in the equation are known and hence the equation can be solved for the output power P_R. Thus equation 5.1 serves as an analysis model. However a design engineer may view this as a design problem and try to find the outer diameter (D_0) and length (L_e) of the motor for a required output power (P_R), a fixed volume and other parameters as specified by the requirement for which the motor is designed. Hence the equation 5.1 becomes a design model where D_0 and L_e are the design variables.

5.3 Modelling and Simulation

A model is a simplified representation of a system at some particular point in time or space [3]. According to Overstreet [4]: a model is an abstraction of a system intended to replicate some properties of that system. The collection of properties the model is intended to replicate must include the modelling objectives. A proper formulation of the objectives is essential for successful optimization of the system being designed. A model is similar to but simpler than the system it represents. The purpose of a model is to enable the analyst to predict the effect of the changes to the system. On one hand a model should be a close approximation of the real system but on the other hand it should not be so complex that it is impossible to understand and experiment with it. A good model is a judicious trade-off between reality and simplicity.

A simulation is the manipulation of a model in such a way that it operates on time or space to compress it, thus enabling the perception of iterations that would not otherwise be apparent [3]. A simulation is the operation of a model of the system. The operation of the model can be studied and hence properties concerning the behaviour of the actual system or its subsystem can be inferred. According to Shannon [5] a simulation may be regarded as the use of a mathematical model as an experimental vehicle to answer questions about the system under consideration. A simulation forms the foundation for making some decisions and these decisions are based on the answers provided by the simulation.

In the ***synthesis phase*** of **Progressive Design Methodology** (PDM) an engineering system is optimised based on certain objectives and the optimisation algorithm performs trade-offs among the objectives based on their values. Hence, in the case of optimization the answers provided by the simulation are in fact the values of the objective functions. The decisions that the optimisation algorithm makes are the trade-offs among objectives to reach an optimal solution (in case of single objective optimisation) or a set of optimal solutions (in case of multiobjective optimisation).

5.4 Classification of models

A model can be ***formal*** (mathematical expression) or ***judgmental***. Some models are ***causal***, i.e. they reflect cause-effect relationships while some models are ***correlational***. The ***deterministic*** models are that for which the value of their variables is known with certainty, i.e. a deterministic model generates the response to a given input by a fixed set of law. On the other hand the models that have values that are not known with certainty are said to be ***stochastic*** or ***probabilistic*** models. In Figure 5.1 different types of mathematical models are shown.

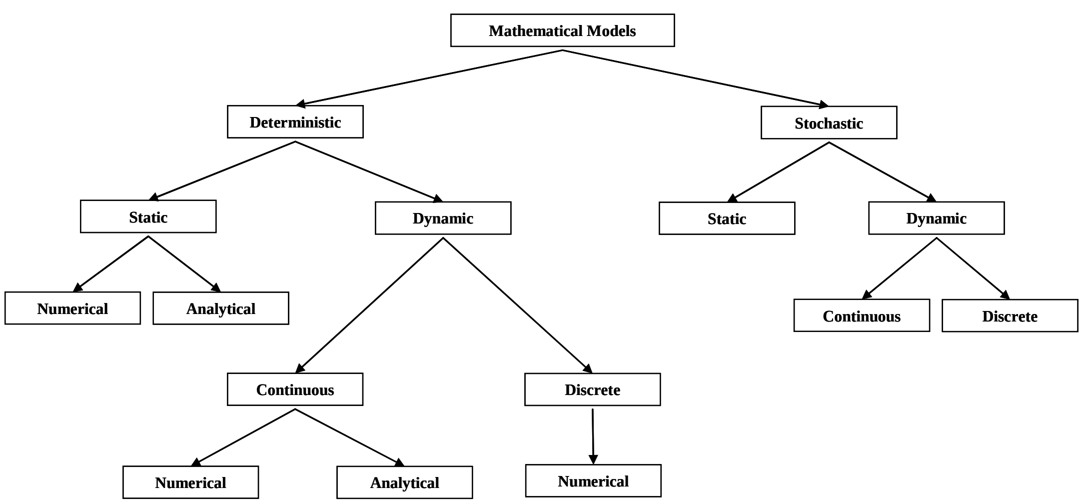


Figure 5.1: Classification of models

5.5 Elements of Models

The model of a system has following elements:

1. ***System Variables*: These are the quantities that specify different states of a system by assuming different values (usually within defined acceptable limits). In the example of the design of a Permanent Magnet Motor design the variables can be number of slots, number of poles, length of the motor, number of turns in the coil etc.
2. ***System Parameters*:** The quantities that are given specific values in any particular model statement. They are fixed by the application of the model rather than by the underlying phenomenon. In the case of a Permanent Magnet Motor design the system parameters are the input voltage to the motor and operational speed.
3. ***System Constants*:** These are the quantities fixed by the underlying phenomenon rather than by the particular model statement. Typically, they are natural constants. In the case of a Permanent Magnet Motor the system constants are resistivity of copper, density of permanent magnet, density of lamination material etc.
4. ***Mathematical Relations*:** These are equalities and inequalities that relate the system variables, parameters and constants. The relations include some type of functional representation such as

$$y = f(x) \tag{5.2}$$

Developing these relations is the most difficult part of modelling and often such a relation is referred as the model of the system. These relations attempt to describe the function of the system within the conditions imposed by its environment. The mathematical relation given in equation 5.2 may be a system of equations, algebraic or differential or a computer based subroutine.

The clear distinction between variables and parameters is very important at the modelling stage. The choice which quantities will be classified as variables or parameters is a subjective decision dictated by choices in the hierarchical level, boundary isolation and intended use of the model of the system [3].

5.6 Modelling and Simulation Hierarchy

A model or simulation that has high fidelity represents a system with great details (a model that is at the bottom of hierarchy). A model or simulation of a system that is highly aggregated does not have much detail and the system is represented in an abstract manner. Such models are known as low fidelity models. At the top of the hierarchy are the abstract models and simulation (low fidelity models) and at the bottom of the hierarchy are the high fidelity models.

Every system is analysed at a particular level of complexity that corresponds to the interests of the individual who studies the system. Thus a hierarchical level can be identified in the system definition. Each system is broken down into subsystems that can be further divided, with the various subsystems or components that are interconnected. A boundary around any system will determine the links with its environment and determine the input/output characterisation. These observations are very important for an appropriate identification of the system that will form the basis for constructing a mathematical model.

It is possible to represent a system as a single unit at one level or as a collection of subsystems that must be co-ordinated at an overall system level. This is an important modelling decision when the size of the system becomes large. The design process is iterative and hierarchical in nature. To solve complex design problems the team of designers consider the problem at different levels of abstraction, ranging from very high-level system decompositions to very low-level detailed specification of components [6, 7]. During this process the design team adds information and hence transforms the design representation. For example, a "needs assessment" is transformed into design specifications and engineering requirements; engineering requirements in turn are converted into a family of solutions that are evaluated and compared to iterate on the description of the artefact in terms of form, function and behaviour [8]. As a result all representations evolve simultaneously from the initial high-level decompositions to increasingly detailed descriptions of the design artefact.

However, not always are the most detailed and accurate simulation models also the most appropriate. Sometimes it is more important to evaluate many different alternatives quickly with only coarse, high-level models. For example, in the initial phase of the design process detailed models are often unnecessary as many of the design details still have to be decided and accurate parameter values are still unknown.

5.7 Suitability of Models for PDM

The engineering systems can be modelled at many levels of approximation. The right model will depend, in general, on the problem to be solved. The type of model needed to synthesise a new design may be different from the type of models required to accurately predict the performance of a single proposed design or to diagnose problems with an existing design. When an engineering system is to be designed and optimised the choice of proper models will have a profound influence on the results. One of the problems with modelling is that, according to the definition, models simplify the reality. This means that some information will be lost somewhere along the line that can cause problems. Hence, it is important to know how the model relates to real system.

The Progressive Design Methodology (PDM) involves three essential features namely ***Design***, ***Selection***, and ***Tuning***. The main goal of design is to create a set of feasible new artefact based on requirements. This process is carried out in the ***Synthesis Phase*** of PDM. In this phase the ***design models*** of the system under consideration, as discussed in section 5.2, are used. Using the design models together with multiobjective optimization algorithms an initial set of feasible solution is generated. Since the multiobjective optimization is employed and the detailed knowledge of the system is not available, it is prudent to use simple ***low fidelity*** models of the system. The advantage of low fidelity models is that they are computationally less intensive and hence are suitable for multiobjective optimization. The suitable low fidelity models are the ***analytical models***. For many situations it is possible to develop an analytical model of the system by making suitable assumptions. However if analytical models are not possible then simple numerical models of the system should be used in the synthesis phase of PDM.

In the ***Intermediate Analysis*** phase of PDM the ***selection*** is performed. The central challenge of this phase is to select from the set of solutions, obtained in the ***Synthesis Phase***, a subset of suitable solutions. The ***selection*** process involves evaluating the alternatives available. In PDM the alternatives are evaluated based on criteria that cannot be expressed mathematically such as manufacturability of the system. In order to achieve this the ***judgmental models*** are used. The ***judgmental models*** are formed by the deductions and assessments contained in the mind of an expert. In ***Intermediate Analysis*** the expert evaluates each alternative based on judgmental models and assigns preference based on linguistic variables and the entire multi attribute decision making is carried out (chapter 2). After the ***selection*** process a small set of suitable solutions is generated.

The ***Final Analysis*** phase of PDM involves the ***tuning*** process. In the ***tuning*** process the system performance criteria are improved by varying system parameters. In order to achieve this, ***high fidelity model*** of the system that is to be designed is developed. Each alternative obtained after ***Intermediate Analysis*** phase is evaluated using the ***high fidelity model*** and ***tuning*** of the system is performed. The ***high fidelity*** models can be developed using finite element methods (FEM), computational fluid dynamic (CFD), etc. These models are computationally intensive but are closer to the actual system and are suitable for ***Final Analysis*** phase of PDM.

5.8 Conclusions

In this chapter a general description of the models and the suitability of models for PDM are given. Each phase of PDM has special requirements of the models. In the ***Synthesis phase*** of PDM ***low fidelity*** models are used because they are computationally less intensive and hence are suitable for multiobjective optimisation. The ***Intermediate Analysis phase*** involves decision making and in order to facilitate the decision making ***judgmental models*** of the system are used. In the ***Final Analysis*** phase of PDM, ***high fidelity*** models are used to evaluate a small set of alternatives and to perform fine-tuning.

Up to this chapter various aspects and issues of PDM have been discussed. From the next chapter onwards the application of PDM using the example of design of a BLDC motor drive is demonstrated. In the next two chapters (chapter 6 and chapter 7) the models of the BLDC motor drive, based on the discussions of this chapter, are developed. In chapter 8 a case study is presented that demonstrates the impact of the choice of system boundaries on the design of the BLDC motor drive. After having demonstrated the importance of system boundaries, in chapter 9 the entire PDM process is applied to the design of a BLDC motor drive and through experimental results it is shown that satisfactory final design can be obtained using PDM.

5.9 References

[1] J. Hodgson and T. Walters, "Optimizing Pumping Systems to Minimize First or Life-Cycle Cost," presented at 19th International Pump Users Symposium, Houston, 2002.

[2] S. Huang, J. Luo, F. Leonardi, and T. A. Lipo, "A Comparison of Power Density for Axial Flux Machines Based on the General Purpose Sizing Equation," 1999.

[3] G. Bellinger, "Modelling & Simulation: An Introduction," 2004.

[4] C. M. Overstreet, "Model Specification and Analysis for Discrete Event Simulation," in Department of Computer Science. Blacksburg: Virginia Tech, 1982.

[5] R. E. Shannon, Systems Simulation: The Art and Science: Prentice-Hall, 1975.

[6] T. J. A. d. Vries and A. P. J. Breuness, "Structuring Product Models to Facilitate Design Manipulations," presented at International Conference on Engineering Design, Prague, Czech Republic, 1995.

[7] S. B. Shooter, W. Keirouz, S. Szykman, and S. J. Fenves, "A Model for the Flow of Design Information," presented at ASME DETC 2000, 12th International Conference on Design Theory and Methodology, Baltimore, MD, 2000.

[8] G. Phal and W. Beitz, Engineering Design: A Systematic Approach. London: Springer-Verlag, 1996.

Chapter 6

Magnetic Model of Permanent Magnet Brushless Direct Current (BLDC) Motor

6.1 Introduction

In this chapter the magnetic model of the motor is developed. In the synthesis phase of PDM the multi objective optimisation (MOO) is used to determine the Pareto optimal solution. Based on the discussion of chapter 5 a formal (mathematical) deterministic and a low fidelity model of the BLDC motor is required. For this purpose a simple magnetic model of the BLDC motor as given in [1] is sufficient. The details of this model are given in Appendix D. In the Final Analysis phase of PDM the fine tuning of design has to be performed on a small set of competent in order to select a final solution. In this stage it is important to obtain accurate value of induced voltage, cogging torque, torque profile etc. Hence in this case formal (mathematical) deterministic and a high fidelity model of the BLDC motor is required.

In the subsequent sections of this chapter a ***high fidelity formal deterministic*** magnetic model of BLDC motor based on ***the analytical methods*** is presented. The reason behind using analytical model is that they are fast compared to finite element or boundary element models.

Eid and Mouilett [2] proposed an analytical model for predicting the two dimensional airgap field distribution of the internal rotor motors having a cylindrical permanent magnet with uniform diametric magnetisation. In addition Gu and Gao [3] used the method of separation of variables to analyse the air gap field of a multi pole permanent magnet motor. Their analysis was based on rectangular co-ordinates. Boules [4] also presented a model of multipole permanent magnet motor in rectangular co-ordinates but used an equivalent magnet pole arc to take into account that in a cylindrical machine the circumferential width of the magnet generally varies with the radius. A further refined model was proposed by Boules [5]. In his work he formulated a model in polar co-ordinates which utilises the concept of equivalent current carrying coils to determine the flux density at the stator and rotor surfaces of a permanent magnet motor.

Zhu et.al. [6] developed an improved analytical method for determination of field distribution. The methodology proposed by Zhu. et. al. was based on two dimensional polar co-ordinates and addressed both internal and external rotor motor topologies. This technique involved solution of governing Laplacian / quasi-Poissonian field equations in the airgap/magnet regions without any assumption regarding the relative recoil permeability of the magnets. However the model of Zhu et.al. [6] was applicable only to machines having radially magnetised magnets. Rasmussan [7] extended the model to include the parallel magnetisation. Recently, Zhu et.al. [8] extended the model further to taker into account both radial and parallel magnetisation.

In the above works of different authors it was assumed that the permeability of iron was infinite and the thickness of stator iron is also infinite. In this chapter an analytical model for the instantaneous air gap field density with the assumption that the iron (both stator and rotor yoke) has finite permeability and the thickness of the stator yoke is also finite is presented. Apart from that the proposed model is valid for radial magnetisation, parallel magnetisation, radial sinusoidal amplitude magnetisation and sinusoidal angle magnetisation. The magnetic field distribution obtained from this model can be further utilised to determine the cogging torque, induced back emf and iron losses. In the next section (6.2) different types of magnetisation of the permanent magnets are presented. The analytical model of the field distribution for a slotless stator and comparison with the FEM is presented in section 6.3. In section 6.4 the model of the slots in BLDC motor is given. In section 6.5 the analytical model of field produced in a slotted motor by magnets is presented. The calculation of back emf is given in section 6.6 and the model of cogging torque is presented in section 6.7. A simple model for calculation of inductance is given in section 6.8. The comparison between analytical model and FEM are given in section 6.9. and the comparison between the results obtained by the analytical model and experimental results are presented in section 6.10. Finally conclusions are drawn in section 6.11.

6.2 Magnetisation of Permanent Magnets

The general configuration of a permanent magnet brushless DC (BLDC) motor considered is shown in Figure 6.1.

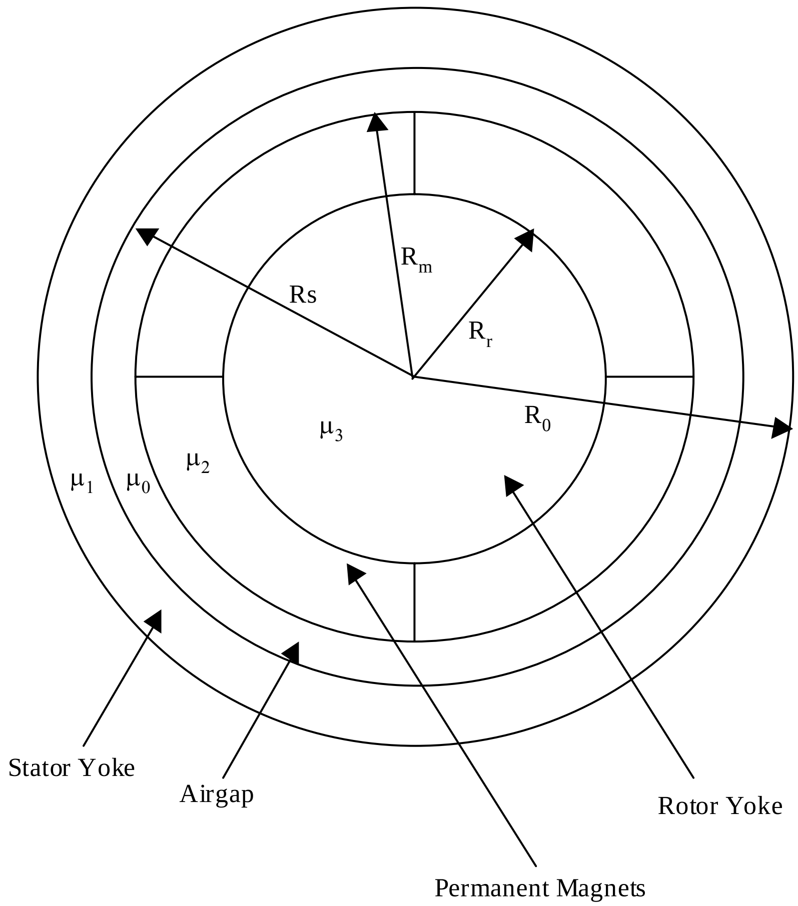


Figure 6.1: The general schematic diagram of a BLDC motor

In the above figure μ_1, μ_2 and μ_3 represent the relative permeability of stator iron, permanent magnets and rotor iron respectively. The parameter μ_0 represents the permeability of free space. The radii R_o, R_s, R_m and R_r represent outer radius of the motor, inner radius of the stator, radius of the magnets and radius of the rotor respectively. In the present analysis it is assumed that the region exterior to the motor is air.

For the motor shown in Figure 6.1 the magnetic field vector $\boldsymbol{H}$ and magnetic field density vector $\boldsymbol{B}$ are, in different regions of the motor, coupled by the following set of equations:

$$\boldsymbol{B_o} = \mu_0 \boldsymbol{H_o} \text{ in the exterior region} \tag{6.1a}$$

$$\boldsymbol{B_s} = \mu_0 \mu_1 \boldsymbol{H_s} \text{ in the stator region} \tag{6.1b}$$

$$\boldsymbol{B_A} = \mu_0 \boldsymbol{H_A} \text{ in the airgap region} \tag{6.1c}$$

$$\boldsymbol{B_M} = \mu_0 \mu_2 \boldsymbol{H_M} + \mu_0 \boldsymbol{M} \text{ in the magnet region} \tag{6.1d}$$

$$\boldsymbol{B_R} = \mu_0 \mu_2 \boldsymbol{H_R} \text{ in the rotor region} \tag{6.1e}$$

where $\boldsymbol{M}$ is the magnetisation vector of the permanent magnets. The amplitude of the magnetisation vector $\boldsymbol{M}$, for a multipole motor with permanent magnets having a linear second quadrant demagnetisation characteristics, is given by

$$M = \frac{B_r}{\mu_0} \tag{6.2}$$

The direction of $\boldsymbol{M}$ depends on the orientation and magnetisation of the permanent magnets. In polar co-ordinates the magnetisation vector $\boldsymbol{M}$ is expressed as

$$\boldsymbol{M} = M_r \boldsymbol{r} + M_\theta \boldsymbol{\theta} \tag{6.3}$$

In the above equation M_r and M_θ are the magnitudes of magnetisation in radial and parallel directions respectively.

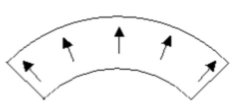


Figure 6.2a: Radial Magnetisation

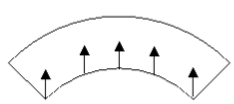


Figure 6.2b: Parallel Magnetisation

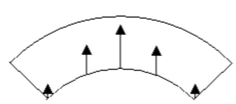


Figure 6.2c: Sinusoidal Amplitude Magnetisation

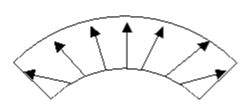


Figure 6.2c: Sinusoidal Angle Magnetisation

The different types of magnetisation viz. radial magnetisation, parallel magnetisation, sinusoidal amplitude magnetisation and sinusoidal angle are shown in Figure 6.2. The components of radial magnetisation over one pole pair is given by:

$$\left.\begin{aligned} M_r &= 0 \\ M_\theta &= 0 \end{aligned}\right\} \; -\frac{\pi}{2N_p} \le \theta \le -\alpha_m \frac{\pi}{2N_p} \tag{6.4a}$$

$$\left.\begin{aligned} M_r &= \frac{B_r}{\mu_0} \\ M_\theta &= 0 \end{aligned}\right\} \; -\alpha_m \frac{\pi}{2N_p} \le \theta \le \alpha_m \frac{\pi}{2N_p} \tag{6.4b}$$

$$\left.\begin{aligned} M_r &= 0 \\ M_\theta &= 0 \end{aligned}\right\} \; \alpha_m \frac{\pi}{2N_p} \le \theta \le (2-\alpha_m)\frac{\pi}{2N_p} \tag{6.4c}$$

$$\left.\begin{aligned} M_r &= -\frac{B_r}{\mu_0} \\ M_\theta &= 0 \end{aligned}\right\} \; \alpha_m \frac{\pi}{2N_p} \le \theta \le (2-\alpha_m)\frac{\pi}{2N_p} \tag{6.4d}$$

$$\left.\begin{aligned} M_r &= 0 \\ M_\theta &= 0 \end{aligned}\right\} \; (2-\alpha_m)\frac{\pi}{2N_p} \le \theta \le \frac{3\pi}{2N_p} \tag{6.4e}$$

In case of parallel magnetisation, the components of magnetisation over one pole pair is given by:

$$\left.\begin{aligned} M_r &= 0 \\ M_\theta &= 0 \end{aligned}\right\} \; -\frac{\pi}{2N_p} \le \theta \le -\alpha_m \frac{\pi}{2N_p} \tag{6.5a}$$

$$\left.\begin{aligned} M_r &= \frac{B_r}{\mu_0}\cos(\theta) \\ M_\theta &= -\frac{B_r}{\mu_0}\sin(\theta) \end{aligned}\right\} \; -\alpha_m \frac{\pi}{2N_p} \le \theta \le \alpha_m \frac{\pi}{2N_p} \tag{6.5b}$$

$$\left.\begin{array}{l} M_r = 0 \\ M_\theta = 0 \end{array}\right\} \alpha_m \frac{\pi}{2N_p} \le \theta \le (2-\alpha_m)\frac{\pi}{2N_p} \tag{6.5c}$$

$$\left.\begin{array}{l} M_r = -\frac{B_r}{\mu_0}\cos(\theta - \frac{\pi}{N_p}) \\ M_\theta = \frac{B_r}{\mu_0}\sin(\theta - \frac{\pi}{N_p}) \end{array}\right\} \alpha_m \frac{\pi}{2N_p} \le \theta \le (2-\alpha_m)\frac{\pi}{2N_p} \tag{6.5d}$$

$$\left.\begin{array}{l} M_r = 0 \\ M_\theta = 0 \end{array}\right\} (2-\alpha_m)\frac{\pi}{2N_p} \le \theta \le \frac{3\pi}{2N_p} \tag{6.5e}$$

For sinusoidal angle magnetisation the components of magnetisation over one pole pair is given by:

$$\left.\begin{array}{l} M_r = \frac{B_r}{\mu_0}\cos(\theta) \\ M_\theta = -\frac{B_r}{\mu_0}\sin(\theta) \end{array}\right\} -\frac{\pi}{2N_p} \le \theta \le \frac{3\pi}{2N_p} \tag{6.6}$$

The components of sinusoidal amplitude magnetisation are given by:

$$\left.\begin{array}{l} M_r = \frac{B_r}{\mu_0}\cos(\theta) \\ M_\theta = 0 \end{array}\right\} -\frac{\pi}{2N_p} \le \theta \le \frac{3\pi}{2N_p} \tag{6.7}$$

where N_p is the number of pole pairs, B_r is the reminance of permanent magnets, α_m is the magnet pole arc to pole pitch ratio and θ is the angular position with reference to the centre of a magnet. It is assumed that the magnetisation is uniform throughout the cross section of the magnets. In case the ratio α_m is less than one, i.e. the magnet pole arc is less than pole pitch, the space between the adjacent magnets is assumed to filled with an unmagnetised material having the same relative permeability (μ_2) as that of the permanent magnets. The waveforms of M_r and M_θ are shown in Figure 6.3. The components M_r and M_θ can be expressed in terms of Fourier series as follows:

$$M_r = \sum_{n=1,3,5...}^{\infty} M_{rn} \cos(nN_p\theta) \tag{6.8a}$$

$$M_\theta = \sum_{n=1,3,5...}^{\infty} M_{\theta n} \sin(nN_p\theta) \tag{6.8b}$$

For radial magnetisation

$$M_{rn} = 2\frac{B_r}{\mu_0} sinc\left\{\frac{n\alpha_m\pi}{2}\right\} \tag{6.9a}$$

$$M_{\theta n} = 0 \tag{6.9b}$$

where

$$sinc\{\beta\} = \frac{\sin(\beta)}{\beta} \tag{6.9c}$$

For parallel magnetisation the components M_{rn} and $M_{\theta n}$ are given by:

$$M_{rn} = \frac{B_r}{\mu_0}\alpha_m(K_1 + K_2) \tag{6.9d}$$

$$M_{\theta n} = \frac{B_r}{\mu_0}\alpha_m(K_1 - K_2) \tag{6.9e}$$

where

$$K_1 = sinc\left[\left(nN_p + 1\right)\alpha_m \frac{\pi}{2N_p}\right] \tag{6.9f}$$

$$K_2 = 1 \text{ when } nN_p = 1 \tag{6.9g}$$

$$\text{Otherwise } K_2 = sinc\left[\left(nN_p - 1\right)\alpha_m \frac{\pi}{2N_p}\right] \tag{6.9h}$$

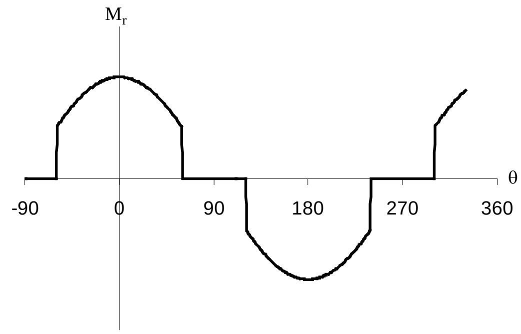


Figure 6.3a: Waveforms of magnetisation components M_r for Parallel Magnetisation

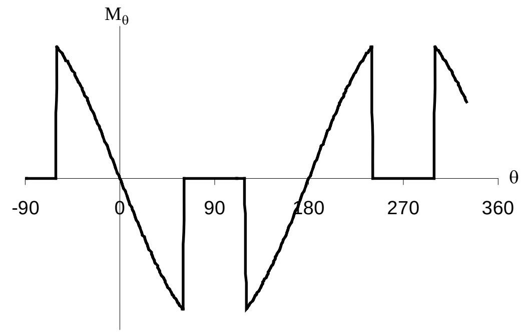


Figure 6.3b: Waveforms of magnetisation components M_θ for Parallel Magnetisation

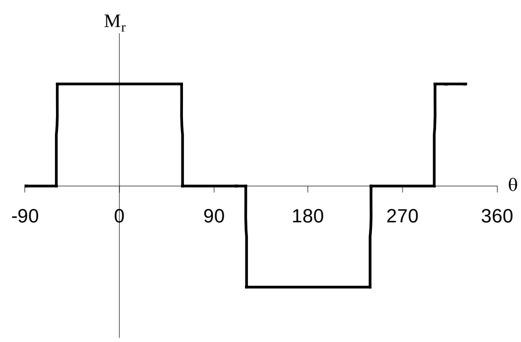


Figure 6.3c: Waveforms of magnetisation components M_r for Radial Magnetisation

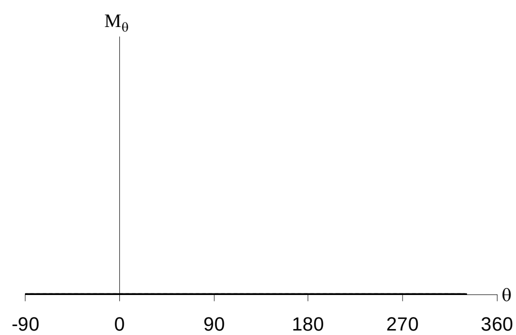


Figure 6.3d: Waveforms of magnetisation components M_θ for Radial Magnetisation

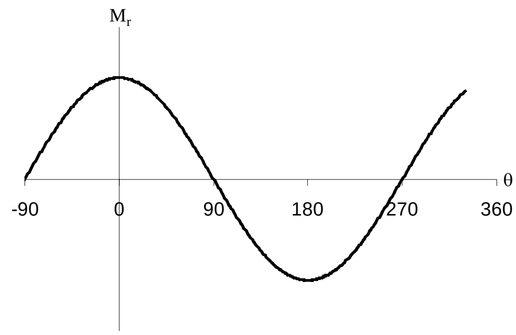


Figure 6.3e: Waveforms of magnetisation components M_r for Sinusoidal Amplitude Magnetisation

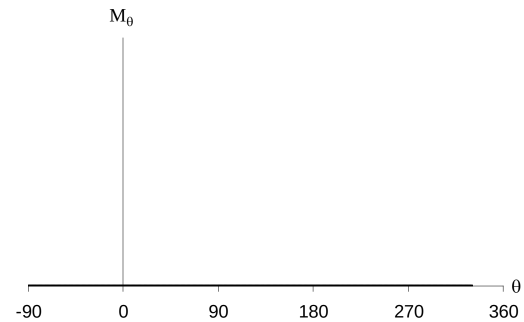


Figure 6.3f: Waveforms of magnetisation components M_θ for Sinusoidal Amplitude Magnetisation

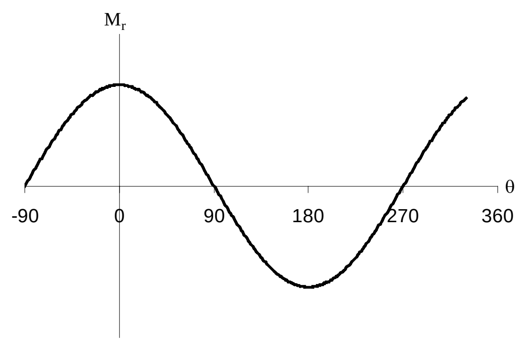


Figure 6.3g: Waveforms of magnetisation components M_r for Sinusoidal Angle Magnetisation

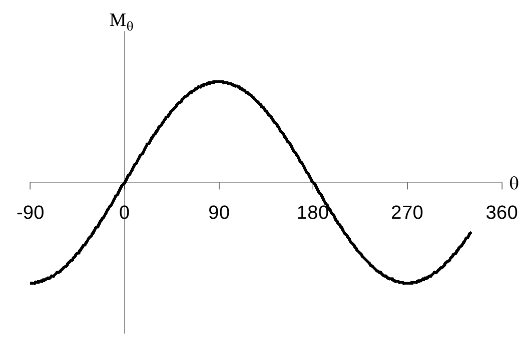


Figure 6.3h: Waveforms of magnetisation components M_θ for Sinusoidal Angle Magnetisation

In case of sinusoidal amplitude magnetisation, the components M_{rn} and $M_{\theta n}$ are given by:

$$\left.\begin{aligned} M_{rn} &= \frac{B_r}{\mu_0} \text{ for } n=1 \\ M_{rn} &= 0 \text{ otherwise} \end{aligned}\right\} \tag{6.9i}$$

$$M_{\theta n} = 0 \tag{6.9j}$$

For sinusoidal angle magnetisation, the components M_{rn} and $M_{\theta n}$ are given by:

$$\left.\begin{aligned} M_{rn} &= \frac{B_r}{\mu_0} \text{ for } n=1 \\ M_{rn} &= 0 \text{ otherwise} \end{aligned}\right\} \tag{6.9k}$$

$$\left.\begin{aligned} M_{\theta n} &= \frac{B_r}{\mu_0} \text{ for } n=1 \\ M_{\theta n} &= 0 \text{ otherwise} \end{aligned}\right\} \tag{6.9l}$$

6.3 Field Produced in a Slotless Motor by Magnets

To facilitate the analysis and to obtain a closed form solution for the airgap field distribution produced by magnets mounted on the rotor surface, the following assumptions are made:

1. The permanent magnets have a linear demagnetisation characteristic.
2. End effects are neglected.
3. The stator and rotor back iron have constant permeability and the saturation is neglected.

The radial and tangential components of the magnetic field can be written in terms of scalar potential (φ) as follows

$$H_r = -\frac{\partial \varphi}{\partial r} \text{ and } H_\theta = -\frac{1}{r}\frac{\partial \varphi}{\partial \theta} \tag{6.10}$$

The scalar magnetic potential distribution in the air gap, stator iron, rotor iron and the exterior region is governed by the Laplace equation. In the magnet region the scalar magnetic potential distribution is governed by quasi-Poissonian equation.

For the air gap region the Laplace equation is:

$$-\frac{\partial^2\varphi_A}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_A}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_A}{\partial\theta^2}=0 \qquad (6.11a)$$

In the magnet region the quasi-Poissonian equation is

$$-\frac{\partial^2\varphi_M}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_M}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_M}{\partial\theta^2}=\frac{1}{\mu_2}\nabla.\boldsymbol{M} \qquad (6.11b)$$

In the stator iron region the Laplace equation is

$$-\frac{\partial^2\varphi_S}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_S}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_S}{\partial\theta^2}=0 \qquad (6.11c)$$

For the rotor iron region the Laplace equation is

$$-\frac{\partial^2\varphi_R}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_R}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_R}{\partial\theta^2}=0 \qquad (6.11d)$$

Finally for the exterior region the Laplace equation is

$$-\frac{\partial^2\varphi_O}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_O}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_O}{\partial\theta^2}=0 \qquad (6.11e)$$

where $\varphi_A, \varphi_M, \varphi_s, \varphi_R$ and φ_O represent the magnetic scalar potential in the air gap, magnet, stator, rotor and the exterior (outer) region respectively.

From equation 6.8 we get

$$\nabla.\boldsymbol{M}=\frac{M_r}{r}+\frac{\partial M_r}{\partial r}+\frac{1}{r}\frac{\partial M_\theta}{\partial\theta} = \sum_{n=1,3,5,\ldots}^{\infty}\frac{1}{r}M_n\cos(nN_p\theta) \qquad (6.12)$$

where

$$M_n=M_{rn}+nN_pM_{\theta n} \qquad (6.13)$$

The boundary conditions for the motor showing in Figure 6.1 are as follows:

(i) at the interface between the stator and the exterior region

$$H_{\theta O}(r,\theta)\Big|_{r=R_o} = H_{\theta S}(r,\theta)\Big|_{r=R_o} \tag{6.14a}$$

$$B_{rO}(r,\theta)\Big|_{r=R_o} = B_{rS}(r,\theta)\Big|_{r=R_o} \tag{6.14b}$$

where $H_{\theta O}$ and $H_{\theta S}$ are the tangential component of the magnetic field vector in the exterior region and the stator iron respectively whereas B_{rO} and B_{rS} are the radial component of the magnetic field density vector in the exterior region and the stator iron respectively.

(ii) at the interface between the stator and the air gap

$$H_{\theta S}(r,\theta)\Big|_{r=R_S} = H_{\theta A}(r,\theta)\Big|_{r=R_S} \tag{6.14c}$$

$$B_{rS}(r,\theta)\Big|_{r=R_S} = B_{rA}(r,\theta)\Big|_{r=R_S} \tag{6.14d}$$

where $H_{\theta A}$ is the tangential component of the magnetic field vector in the air gap region whereas B_{rA} is the radial component of the magnetic field density vector in the air gap region.

(iii) at the interface between the air and the permanent magnet

$$H_{\theta A}(r,\theta)\Big|_{r=R_M} = H_{\theta M}(r,\theta)\Big|_{r=R_M} \tag{6.14e}$$

$$B_{rA}(r,\theta)\Big|_{r=R_M} = B_{rM}(r,\theta)\Big|_{r=R_M} \tag{6.14f}$$

where $H_{\theta M}$ is the tangential component of the magnetic field vector in the magnet region whereas B_{rM} is the radial component of the magnetic field density vector in the magnet region.

(iv) at the interface between the magnet and the rotor iron

$$H_{\theta M}(r,\theta)\Big|_{r=R_r} = H_{\theta R}(r,\theta)\Big|_{r=R_r} \tag{6.14g}$$

$$B_{rM}(r,\theta)\Big|_{r=R_r} = B_{rR}(r,\theta)\Big|_{r=R_r} \tag{6.14h}$$

where $H_{\theta R}$ is the tangential component of the magnetic field vector in the rotor iron region whereas B_{rR} is the radial component of the magnetic field density vector in the rotor iron region.

The dimensions R_r, R_S, R_M and R_O are depicted in Figure 6.1.

The general solutions of the system of equations equation 6.11 and equation 6.12 are:

$$\varphi_O(r,\theta) = B_{nO} r^{-nN_p} \cos(nN_p\theta) \tag{6.15}$$

$$\varphi_S(r,\theta) = (A_{nS} r^{nN_p} + B_{nS} r^{-nN_p}) \cos(nN_p\theta) \tag{6.16}$$

$$\varphi_A(r,\theta) = (A_{nA} r^{nN_p} + B_{nA} r^{-nN_p}) \cos(nN_p\theta) \tag{6.17}$$

$$\varphi_M(r,\theta) = (A_{nM} r^{nN_p} + B_{nM} r^{-nN_p}) \cos(nN_p\theta) + \frac{M_n r \cos(nN_p\theta)}{\mu_2(1-n^2N_p^2)} \tag{6.18}$$

$$\varphi_R(r,\theta)=A_{nR}r^{nN_p}\cos(nN_p\theta) \tag{6.19}$$

where $B_{nO}, A_{nS}, B_{nS}, A_{nA}, B_{nA}, A_{nM}, B_{nM}$ and A_{nR} are constants to be determined. These constants are determined by solving the boundary conditions given in equation 6.14 and using equation 6.10 and equation 6.1. Upon substituting back the constants A_{nA} and B_{nA} into equation 6.17 and using equation 6.10 and equation 6.1 the air radial and tangential components of the air gap field distribution are obtained as follows:

$$B_r(r,\theta)=\sum_{n=1,3,5,\ldots}^{\infty}(nN_p)\mu_0 M_n\frac{B_1B_2}{B_3}\cos(nN_p\theta) \tag{6.20}$$

$$B_\theta(r,\theta)=\sum_{n=1,3,5,\ldots}^{\infty}(nN_p)\mu_0 M_n\frac{B_1B_4}{B_3}\sin(nN_p\theta) \tag{6.21}$$

where

$$B_1=R_mR_r^{(nN_p+1)}\left[2\left(\frac{R_m}{R_r}\right)^{nN_p-1}-\left(nN_p+1\right)+\left(\frac{R_m}{R_r}\right)^{2nN_p}\left(nN_p-1\right)\right]\mu_3$$
$$-R_mR_r^{(nN_p+1)}\left[nN_p-\left(\frac{R_r}{R_m}\right)^{nN_p-1}\left(nN_p+1\right)-\left(\frac{R_m}{R_r}\right)^{nN_p+1}\left(nN_p-1\right)\right]\mu_1$$

$$B_2=\left(\frac{r}{R_s}\right)^{nN_p}\left(\frac{R_o}{R_s}\right)^{nN_p}\left(\mu_2^2-1\right)+\left[\left(1-\mu_2^2\right)\left(\frac{r}{R_o}\right)^{nN_p}+\left(1+\mu_2\right)^2\left(\frac{R_o}{r}\right)^{nN_p}\right]-\left(\frac{R_s}{r}\right)^{nN_p}\left(\frac{R_s}{R_o}\right)^{nN_p}\left(1-\mu_2\right)^2$$

$$B_3=r\left[\left(K_1+K_2\right)K_3+\left(K_4+K_5\right)K_6\right]\left(1-n^2N_p^2\right)$$

$$K_1=\left(1+\mu_1\right)\left(\frac{R_o}{R_s}\right)^{nN_p}\left[\left(1+\mu_2\right)^2-\left(1-\mu_2\right)^2\left(\frac{R_s}{R_o}\right)^{2nN_p}\right]$$

$$K_2 = \left(\mu_1 - 1\right)\left(\mu_2^2 - 1\right)\left(\frac{R_m}{R_r}\right)^{nN_p}\left(\frac{R_m}{R_s}\right)^{nN_p}\left[\left(\frac{R_o}{R_S}\right)^{nN_p} - \left(\frac{R_s}{R_o}\right)^{nN_p}\right]$$

$$K_3 = \mu_3 R_r R_m^{nN_p} + \mu_1 R_m R_r^{nN_p}$$

$$K_4 = \left(1+\mu_1\right)\left(\frac{R_r}{R_m}\right)^{nN_p}\left(\frac{R_o}{R_m}\right)^{nN_p}\left[\left(1+\mu_2\right)^2 - \left(1-\mu_2\right)^2\left(\frac{R_s}{R_o}\right)^{2nN_p}\right]$$

$$K_5 = \left(\mu_1 + 1\right)\left(\mu_2^2 - 1\right)\left(\frac{R_r}{R_s}\right)^{nN_p}\left[\left(\frac{R_o}{R_S}\right)^{nN_p} - \left(\frac{R_s}{R_o}\right)^{nN_p}\right]$$

$$K_6 = \mu_1 R_m R_r^{nN_p} - \mu_3 R_r R_m^{nN_p}$$

$$B_4 = \left(\frac{r}{R_s}\right)^{nN_p}\left(\frac{R_o}{R_s}\right)^{nN_p}\left(\mu_2^2 - 1\right) + \left[\left(1\text{-}\mu_2^2\right)\left(\frac{r}{R_o}\right)^{nN_p} - \left(1+\mu_2\right)^2\left(\frac{R_o}{r}\right)^{nN_p}\right] + \left(\frac{R_s}{r}\right)^{nN_p}\left(\frac{R_s}{R_o}\right)^{nN_p}\left(1-\mu_2\right)^2$$

when $nN_p = 1$, the field distribution in the airgap is given by

$$B_r(r,\theta) = \frac{1}{2}\mu_o M_n \left(\frac{R_r}{r}\right)^2 \frac{B_1 B_2}{B_3}\cos(\theta) \tag{6.22}$$

$$B_\theta(r,\theta) = \frac{1}{2}\mu_o M_n \left(\frac{R_r}{r}\right)^2 \frac{B_1 B_4}{B_3}\cos(\theta) \tag{6.23}$$

where the values of constant B1, B2, B3 and B4 are:

$$B_1 = 2\left(\mu_3 - \mu_1\right)\ln\left(\frac{R_m}{R_r}\right) + \left(\frac{R_m^2}{R_r^2} - 1\right)\left(\mu_1 + \mu_3\right)$$

$$B_2=\left(\mu_2-1\right)^2+\left(\frac{r^2}{R_s^2}-\frac{R_o^2}{R_s^2}-\frac{r^2}{R_s^2}\frac{R_o^2}{R_s^2}\right)\mu_2^2-2\left(\frac{R_o}{R_s}\right)^2\mu_2-\frac{r^2}{R_s^2}-\frac{R_o^2}{R_s^2}+\frac{r^2}{R_s^2}\frac{R_o^2}{R_s^2}$$

$$B_3=\left(\mu_2-1\right)^2\left(\mu_1^2\left(\frac{R_m^2}{R_r^2}-1\right)+\left(\mu_1+\mu_3\right)\mu_1\left(\frac{R_m^2}{R_r^2}+1\right)+\left(\frac{R_m^2}{R_r^2}-1\right)\mu_3\right)$$

$$B_4=\left(\mu_2-1\right)^2+\left(-\frac{r^2}{R_s^2}-\frac{R_o^2}{R_s^2}-\frac{r^2}{R_s^2}\frac{R_o^2}{R_s^2}\right)\mu_2^2-2\left(\frac{R_o}{R_s}\right)^2\mu_2+\frac{r^2}{R_s^2}-\frac{R_o^2}{R_s^2}-\frac{r^2}{R_s^2}\frac{R_o^2}{R_s^2}$$

This model has been applied to three phase slotless BLDC motor with radial, parallel, sinusoidal magnitude and sinusoidal angle magnetisation. The parameters of the motor taken are

$$N_p=6,\ R_s=x,\ R_r=y,\ R_m=z,\ B_r=1.2\text{T},\ \mu_1=1000, \mu_2=1.05,\ \mu_3=1000,\ \alpha_m=0.8$$

The results obtained by our model and the FEM are shown in Figure 6.4 for radial magnetisation, Figure 6.5 for parallel magnetisation, Figure 6.6 for sinusoidal angle magnetisation and Figure 6.7 for sinusoidal amplitude magnetisation.

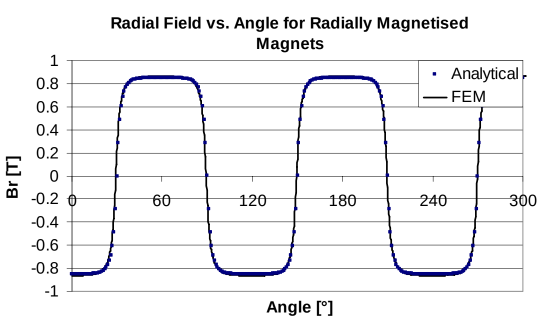


Figure 6.4a: Distribution of Radial component of airgap field in a slotless motor with radially magnetised magnets in airgap (r=16.75mm)

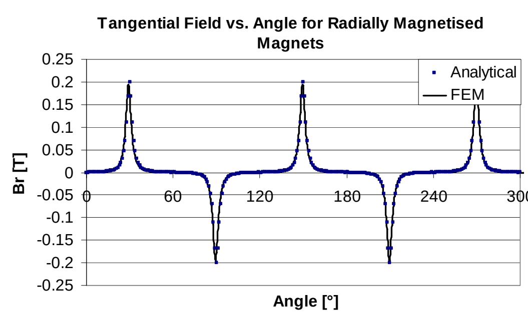
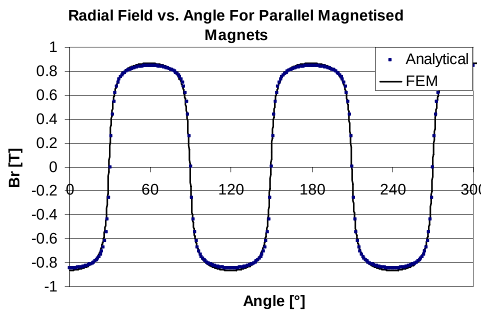


Figure 6.4b: Distribution of Tangential component of airgap field in a slotless motor with radially magnetised magnets in airgap (r=16.75mm)



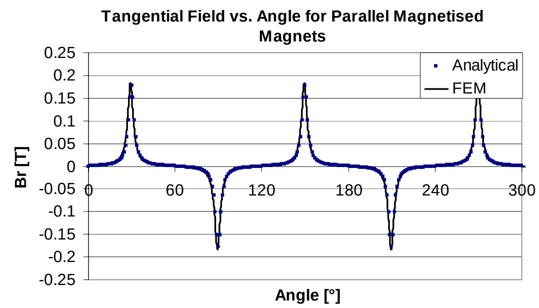
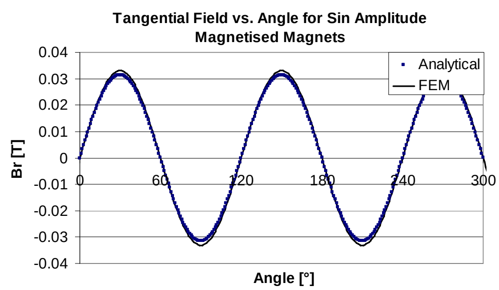


Figure 6.5a: Distribution of Radial component of airgap field in a slotless motor with parallel magnetised magnet(r=16.75mm)

Figure 6.5b: Distribution of Tangential component of airgap field in a slotless motor with parallel magnetised magnet(r=16.75mm)



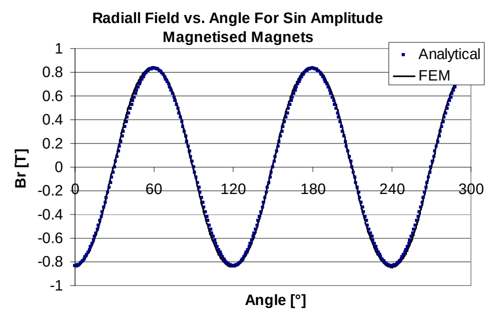
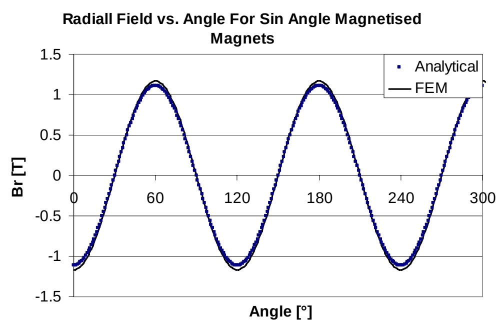


Figure 6.6a: Distribution of Radial component of airgap field in slotless motor with sin amplitude magnetised magnets (r=16.75mm)

Figure 6.6b: Distribution of Tangential component of airgap field in slotless motor with sin amplitude magnetised magnets (r=16.75mm)



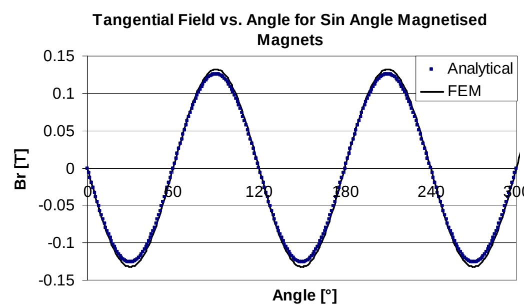


Figure 6.7a: Distribution of Radial component of airgap field in a slotless motor with parallel magnetised magnets in airgap (r=16.75mm)

Figure 6.7b: Distribution of Tangential component of airgap field in a slotless motor with parallel magnetised magnets in airgap (r=16.75mm)

6.4 Model of Slots in BLDC Motor

In this section development of analytical model of the slots is presented. The analysis is based on the lines of Zhu, et.al. [9]. In the case of a slotted stator (Figure 6.8) the magnetic field is changed throughout the airgap and magnet region due to presence of slots. The change in the magnetic field due to slotting is a function of distance from the slots. The influence of slots is minimum at the magnet and rotor iron interface, whereas the greatest influence of slots is experienced at the stator surface. Besides this the slotting is a function of saturation of the ferromagnetic material used in the rotor and stator. Since saturation effect is very difficult to describe analytically it has been ignored in the present analysis.

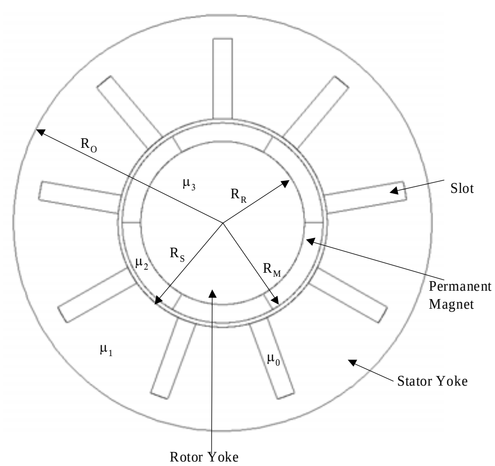


Figure 6.8: BLDC motor with slots

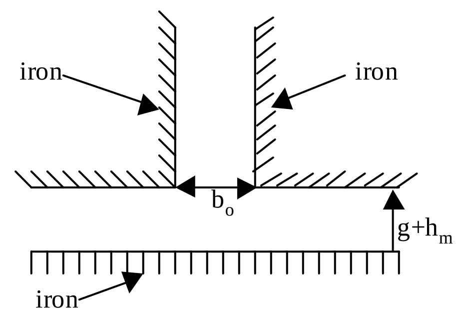


Figure 6.9: The general schematic diagram of a BLDC motor with slots

The permeance of the slotted airgap/magnet region can be determined by conformal transformation method. To simplify the calculation the slots are considered to be rectangular and infinitely deep however in reality the slots are not infinitely deep. In Figure 6.9 the slot geometry for permeance calculation is shown. The permeance function ($\lambda(\theta, r)$) is given by

$$\lambda(\theta, r) = \begin{cases} \Lambda_o \left[1 - \beta(r) - \beta(r) \cos\left(\dfrac{\pi}{0.8\alpha_o} \right) \right] & \text{for } 0 \le \theta \le 0.8\alpha_0 \\ \Lambda_0 & \text{for } 0.8\alpha_0 \le \theta \le \alpha_t / 2 \end{cases} \tag{6.24}$$

where $\Lambda_0 = \mu_0 / (g + h_m)$, g is the air gap length, h_m is the height of the magnet, α_o is slot opening expressed in radians, α_t is the tooth pitch in radians, and the function $\beta(r)$ at a radius *r* is given by

$$\beta(r) = \frac{1}{2} \left[1 - \frac{1}{\sqrt{1 + \left(\dfrac{b_o}{2g'} \right)^2 (1 + v^2)}} \right] \tag{6.25}$$

where g' is

$$g' = g + h_m$$

and *v* is given by the equation below

$$y \frac{\pi}{b_o} = \frac{1}{2} \ln \left[\frac{\sqrt{a^2 + v^2} + v}{\sqrt{a^2 + v^2} - v} \right] + \frac{2g'}{b_o} \arctan \frac{2g'}{b_o} \frac{v}{\sqrt{a^2 + v^2}}$$

and *a* is given by

$$a^2 = 1 + \left(\frac{2g'}{b_o} \right)^2$$

where b_o is the slot opening.

For internal rotor motor *y* is given by

$$y = r - (R_s - g - h_m)$$

where R_s is the inner radius of the stator.

The relative permeance function ($\lambda'(\theta,r)$) is given by

$$\lambda'(\theta,r)=\frac{\lambda(\theta,r)}{\Lambda_o}=\frac{\lambda(\theta,r)}{\mu_o/g'} \tag{6.26}$$

The above expression can be expressed as a Fourier series:

$$\lambda'(\theta,r)=\sum_{m=0}^{\infty}\Lambda'_m(r)\cos(mN_s\theta) \tag{6.27}$$

where N_s is the number of slots in the stator.

The Fourier Coefficient Λ'_m is given by

$$\Lambda'_0=\frac{1}{K_c}\left(1-1.6\beta\frac{b_o}{\tau_t}\right) \tag{6.28}$$

and

$$\Lambda'_m(r)=-\beta(r)\frac{4}{n\pi}\left[0.5+\frac{\left(m\frac{b_o}{\tau_t}\right)^2}{0.78125-2\left(m\frac{b_o}{\tau_t}\right)^2}\right]\sin\left(1.6m\pi\frac{b_o}{\tau_t}\right) \tag{6.29}$$

where τ_t is the tooth pitch.

From the above set of equations the relative permeance function can be calculated.

Once the relative permeance function is determined the radial and the tangential components of the airgap field for a slotted stator are determined as discussed in the next section.

6.5 Field Produced in a Slotted Motor by Magnets

The analysis starts with the calculation of the radial flux density along the outer surface of a smooth stator as discussed in the section 6.3. The actual radial flux density at the stator surface is calculated by multiplying the relative permeance function with the radial flux density of a slotless motor ($B_r(r,\theta)$) in the previous section). This actual radial flux density is considered as a new boundary condition and hence a new boundary value problem is set up.

$$B_{rI}(r,\theta)=B_r(r,\theta)\lambda_m(r,\theta) \tag{6.30}$$

where $B_r(r,\theta)$ is the air gap field distribution (equation 6.20, section 6.3) and $\lambda_m(r,\theta)$ is the relative permeance function as defined in the previous section. At the stator surface

$$B_{rI}(r,\theta)\Big|_{r=R_s}=\left\{\sum_{n=1,3,5,\ldots}^{\infty} B_n\cos\left(nN_p\theta\right)\right\}\left\{\sum_{m=0,1,2,\ldots}^{\infty}\lambda_m\cos\left(mN_s\theta\right)\right\} \tag{6.31}$$

where N_P is the number of poles of the motor.

The above equation can also be written as

$$\begin{aligned}B_{rI}(r,\theta)\Big|_{r=R_s} &= \sum_{n=1,3,5,\ldots}^{\infty} B_n\lambda_o\cos\left(nN_p\theta\right)\\ &+\frac{1}{2}\sum_{n=1,3,5,\ldots}^{\infty}\sum_{m=1,2,3,\ldots}^{\infty} B_n\lambda_m\{\cos[(nN_p+mN_s)\theta]+\cos[(nN_p-mN_s)\theta]\}\end{aligned} \tag{6.32}$$

where B_n is the magnitude flux density at the stator and can be obtained by substituting $r=R_s$ in equation 6.20. The scalar magnetic potential distribution in the air gap, stator iron, rotor iron and the exterior region is governed by the Laplace equation. In the magnet region the scalar magnetic potential distribution is governed by quasi-Poissonian equation

For the airgap region the Laplace equation is:

$$-\frac{\partial^2\varphi_A}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_A}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_A}{\partial\theta^2}=0 \tag{6.33a}$$

In the magnet region the quasi-Poissonian equation is

$$-\frac{\partial^2\varphi_M}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_M}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_M}{\partial\theta^2}=\frac{1}{\mu_2}\nabla.\boldsymbol{M} \tag{6.33b}$$

In the stator iron region the Laplace equation is

$$-\frac{\partial^2\varphi_S}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_S}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_S}{\partial\theta^2}=0 \tag{6.33c}$$

For the rotor iron region the Laplace equation is

$$-\frac{\partial^2\varphi_R}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_R}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_R}{\partial\theta^2}=0 \tag{6.33d}$$

Finally for the exterior region the Laplace equation is

$$-\frac{\partial^2\varphi_O}{\partial r^2}+\frac{1}{r}\frac{\partial\varphi_O}{\partial r}+\frac{1}{r^2}\frac{\partial^2\varphi_O}{\partial\theta^2}=0 \tag{6.33e}$$

The general solution of the above equations is similar to that given in the section 6.3 except that certain modifications are made to accommodate the slotting effect. The general solution of the scalar potential in the exterior region of the motor is given as

$$\varphi_O(r,\theta)=\varphi_{O1}(r,\theta)+\varphi_{O2}(r,\theta)+\varphi_{O3}(r,\theta) \tag{6.34}$$

where the terms $\varphi_{O1}(r,\theta)$, $\varphi_{O2}(r,\theta)$, and $\varphi_{O3}(r,\theta)$ are given by

$$\varphi_{O1}(r,\theta)=B_{nO1}r^{-nN_p}\cos(nN_p\theta) \tag{6.35}$$

$$\varphi_{O2}(r,\theta)=B_{nO2}r^{-nN_p-mN_s}\cos(nN_p\theta+mN_s\theta) \tag{6.36}$$

$$\varphi_{O3}(r,\theta)=B_{nO3}r^{-nN_p+mN_s}\cos(nN_p\theta-mN_s\theta) \tag{6.37}$$

In the stator region of the scalar potential has a general solution of

$$\varphi_S(r,\theta)=\varphi_{S1}(r,\theta)+\varphi_{S2}(r,\theta)+\varphi_{S3}(r,\theta) \tag{6.38}$$

where the terms in equation 6.38 are given by

$$\varphi_{S1}(r,\theta)=(A_{nS1}r^{nN_p}+B_{nS1}r^{-nN_p})\cos(nN_p\theta) \tag{6.39}$$

$$\varphi_{S2}(r,\theta)=(A_{nS2}r^{nN_p+mN_S}+B_{nS2}r^{-nN_p-mN_S})\cos(nN_p\theta+nN_S\theta) \tag{6.40}$$

$$\varphi_{S3}(r,\theta)=(A_{nS3}r^{nN_p-mN_S}+B_{nS3}r^{-nN_p+mN_S})\cos(nN_p\theta-nN_S\theta) \tag{6.41}$$

For the air-gap region the scalar potential has a solution of the form

$$\varphi_A(r,\theta)=\varphi_{A1}(r,\theta)+\varphi_{A2}(r,\theta)+\varphi_{A3}(r,\theta) \tag{6.42}$$

where the terms $\varphi_{A1}(r,\theta)$, $\varphi_{A2}(r,\theta)$, $\varphi_{A3}(r,\theta)$ in equation 6.42 are given as by:

$$\varphi_{A1}(r,\theta)=(A_{nA1}r^{nN_p}+B_{nA1}r^{-nN_p})\cos(nN_p\theta) \tag{6.43}$$

$$\varphi_{A2}(r,\theta)=(A_{nA2}r^{nN_p+mN_s}+B_{nA2}r^{-nN_p-mN_s})\cos(nN_p\theta+nN_s\theta) \tag{6.44}$$

$$\varphi_{A3}(r,\theta)=(A_{nA3}r^{nN_p-mN_s}+B_{nA3}r^{-nN_p+mN_s})\cos(nN_p\theta-nN_s\theta) \tag{6.45}$$

In the magnet region the scalar potential can be expressed as

$$\varphi_M(r,\theta)=\varphi_{M1}(r,\theta)+\varphi_{M2}(r,\theta)+\varphi_{M3}(r,\theta) \tag{6.46}$$

where the terms in equation 6.46 are given by

$$\varphi_{M1}(r,\theta)=(A_{nM1}r^{nN_p}+B_{nM1}r^{-nN_p})\cos(nN_p\theta)+\frac{M_n r\cos(nN_p\theta)}{\mu_2(1-n^2N_p^2)} \tag{6.47}$$

$$\varphi_{M2}(r,\theta)=(A_{nM2}r^{nN_p+mN_s}+B_{nM2}r^{-nN_p-mN_s})\cos(nN_p\theta+mN_s\theta) \tag{6.48}$$

$$\varphi_{M3}(r,\theta)=(A_{nM3}r^{nN_p-mN_s}+B_{nM3}r^{-nN_p+mN_s})\cos(nN_p\theta-mN_s\theta) \tag{6.49}$$

Finally in the rotor iron region the general solution of the scalar potential is as follows

$$\varphi_R(r,\theta)=\varphi_{R1}(r,\theta)+\varphi_{R2}(r,\theta)+\varphi_{R3}(r,\theta) \tag{6.50}$$

where the factors in the equation 6.50 are given by

$$\varphi_{R1}(r,\theta)=A_{nR1}r^{nN_p}\cos(nN_p\theta) \tag{6.51}$$

$$\varphi_{R2}(r,\theta)=A_{nR2}r^{nN_p+mN_s}\cos(nN_p\theta+mN_s\theta) \tag{6.52}$$

$$\varphi_R(r,\theta)=A_{nR2}r^{nN_p-mN_s}\cos(nN_p\theta-mN_s\theta) \tag{6.53}$$

The constants B_{nO1}, B_{nO2}, B_{nO3}, A_{nS1}, B_{nS1}, A_{nS2}, B_{ns2}, A_{nS3}, B_{nS3}, A_{nA1}, B_{nA1}, A_{nA2}, B_{nA2}, A_{nA3}, B_{nA3}, A_{nM1}, B_{nM1}, A_{nM2}, B_{nM2}, A_{nM3}, B_{nM3}, A_{nR1}, A_{nR2} and A_{nR3} are to be determined. In order to determine these constants the set of equation, equation 6.34, equation 6.38, equation 6.42, equation 6.46 and equation 6.50 are solved. The necessary boundary conditions required to solve this set of equations are given below.

(i) At the interface between the stator and the exterior region the tangential component of the magnetic field is continuous, i.e.,

$$-\frac{1}{r}\frac{\partial\varphi_{o1}}{\partial\theta}\bigg|_{r=R_o}=-\frac{1}{r}\frac{\partial\varphi_{S1}}{\partial\theta}\bigg|_{r=R_o} \tag{6.54a}$$

$$-\frac{1}{r}\frac{\partial\varphi_{o2}}{\partial\theta}\bigg|_{r=R_o}=-\frac{1}{r}\frac{\partial\varphi_{S2}}{\partial\theta}\bigg|_{r=R_o} \tag{6.54b}$$

$$-\frac{1}{r}\frac{\partial\varphi_{o3}}{\partial\theta}\bigg|_{r=R_o} = -\frac{1}{r}\frac{\partial\varphi_{S3}}{\partial\theta}\bigg|_{r=R_o} \tag{6.54c}$$

The radial component of the magnetic field is constant, hence the mathematical expression of the boundary condition is

$$-\mu_o\frac{\partial\varphi_{o1}}{\partial r}\bigg|_{r=R_o} = -\mu_o\mu_1\frac{\partial\varphi_{S1}}{\partial r}\bigg|_{r=R_o} \tag{6.54d}$$

$$-\mu_o\frac{\partial\varphi_{o2}}{\partial r}\bigg|_{r=R_o} = -\mu_o\mu_1\frac{\partial\varphi_{S2}}{\partial r}\bigg|_{r=R_o} \tag{6.54e}$$

$$-\mu_o\frac{\partial\varphi_{o3}}{\partial r}\bigg|_{r=R_o} = -\mu_o\mu_1\frac{\partial\varphi_{S3}}{\partial r}\bigg|_{r=R_o} \tag{6.54f}$$

(ii) At the interface between the stator and the air gap, the tangential component of the field is continuous, i.e.,

$$-\frac{1}{r}\frac{\partial\varphi_{S1}}{\partial\theta}\bigg|_{r=R_o} = -\frac{1}{r}\frac{\partial\varphi_{A1}}{\partial\theta}\bigg|_{r=R_o} \tag{6.55a}$$

$$-\frac{1}{r}\frac{\partial\varphi_{S2}}{\partial\theta}\bigg|_{r=R_o} = -\frac{1}{r}\frac{\partial\varphi_{A2}}{\partial\theta}\bigg|_{r=R_o} \tag{6.55b}$$

$$-\frac{1}{r}\frac{\partial\varphi_{S3}}{\partial\theta}\bigg|_{r=R_o} = -\frac{1}{r}\frac{\partial\varphi_{A3}}{\partial\theta}\bigg|_{r=R_o} \tag{6.55c}$$

The radial component of the magnetic field is constant, hence the mathematical expression of the boundary condition is

$$-\mu_o\mu_1\frac{\partial\varphi_{S1}}{\partial r}\bigg|_{r=R_o} = \sum_{n=1,3,5,\ldots}^{\infty} B_n\lambda_o\cos(nN_p\theta) \tag{6.55d}$$

$$-\mu_o\mu_1\left.\frac{\partial\varphi_{S2}}{\partial r}\right|_{r=R_o}=\frac{1}{2}\sum_{n=1,3,5,\ldots}^{\infty}\sum_{m=1,2,3,\ldots}^{\infty}B_n\lambda_m\cos(nN_p\theta+nN_s\theta) \quad (6.55e)$$

$$-\mu_o\mu_1\left.\frac{\partial\varphi_{S3}}{\partial r}\right|_{r=R_o}=\frac{1}{2}\sum_{n=1,3,5,\ldots}^{\infty}\sum_{m=1,2,3,\ldots}^{\infty}B_n\lambda_m\cos(nN_p\theta-nN_s\theta) \quad (6.55f)$$

(iii) At the interface between the air and the permanent magnet the tangential component of the field is given by

$$-\frac{1}{r}\left.\frac{\partial\varphi_{A1}}{\partial\theta}\right|_{r=R_o}=-\frac{1}{r}\left.\frac{\partial\varphi_{M1}}{\partial\theta}\right|_{r=R_o} \quad (6.56a)$$

$$-\frac{1}{r}\left.\frac{\partial\varphi_{A2}}{\partial\theta}\right|_{r=R_o}=-\frac{1}{r}\left.\frac{\partial\varphi_{M2}}{\partial\theta}\right|_{r=R_o} \quad (6.56b)$$

$$-\frac{1}{r}\left.\frac{\partial\varphi_{A3}}{\partial\theta}\right|_{r=R_o}=-\frac{1}{r}\left.\frac{\partial\varphi_{M3}}{\partial\theta}\right|_{r=R_o} \quad (6.56c)$$

The radial component of the magnetic field is constant and hence the boundary condition is

$$-\mu_o\left.\frac{\partial\varphi_{A1}}{\partial r}\right|_{r=R_o}=-\mu_o\mu_2\left.\frac{\partial\varphi_{M1}}{\partial r}\right|_{r=R_o} \quad (6.56d)$$

$$-\mu_o\left.\frac{\partial\varphi_{A2}}{\partial r}\right|_{r=R_o}=-\mu_o\mu_2\left.\frac{\partial\varphi_{M2}}{\partial r}\right|_{r=R_o} \quad (6.56e)$$

$$-\mu_o\left.\frac{\partial\varphi_{A3}}{\partial r}\right|_{r=R_o}=-\mu_o\mu_2\left.\frac{\partial\varphi_{M3}}{\partial r}\right|_{r=R_o} \quad (6.56f)$$

(iv) At the interface between the magnet and the rotor iron the tangential component of the field is given by

$$-\frac{1}{r}\frac{\partial\varphi_{M1}}{\partial\theta}\bigg|_{r=R_o}=-\frac{1}{r}\frac{\partial\varphi_{R1}}{\partial\theta}\bigg|_{r=R_o} \tag{6.57a}$$

$$-\frac{1}{r}\frac{\partial\varphi_{M2}}{\partial\theta}\bigg|_{r=R_o}=-\frac{1}{r}\frac{\partial\varphi_{R2}}{\partial\theta}\bigg|_{r=R_o} \tag{6.57b}$$

$$-\frac{1}{r}\frac{\partial\varphi_{M3}}{\partial\theta}\bigg|_{r=R_o}=-\frac{1}{r}\frac{\partial\varphi_{R3}}{\partial\theta}\bigg|_{r=R_o} \tag{6.57c}$$

The radial component of the magnetic field is constant and hence the boundary condition is

$$-\mu_o\mu_2\frac{\partial\varphi_{M1}}{\partial r}\bigg|_{r=R_o}=-\mu_o\mu_3\frac{\partial\varphi_{R1}}{\partial r}\bigg|_{r=R_o} \tag{6.57d}$$

$$-\mu_o\mu_2\frac{\partial\varphi_{M2}}{\partial r}\bigg|_{r=R_o}=-\mu_o\mu_3\frac{\partial\varphi_{R2}}{\partial r}\bigg|_{r=R_o} \tag{6.57e}$$

$$-\mu_o\mu_2\frac{\partial\varphi_{M3}}{\partial r}\bigg|_{r=R_o}=-\mu_o\mu_3\frac{\partial\varphi_{R3}}{\partial r}\bigg|_{r=R_o} \tag{6.57f}$$

The constants mentioned above are determined by solving the boundary conditions given in equation 6.54 to equation 6.57. Upon substituting back the constants $A_{nS1}, B_{nS1}, A_{nS2}, B_{ns2}, A_{nS3}$ and B_{nS3} into equation 6.34, equation 6.38, equation 6.42, equation 6.46 and equation 6.50 the tangential components of the air gap field distribution for the slotted case at the stator surface is obtained:

$$\begin{aligned} B_\theta(r,\theta) &= \sum_{n=1,3,5,\ldots}^{\infty} B_I \sin(nN_p\theta) \\ &+ \sum_{n=1,3,5,\ldots}^{\infty}\sum_{m=1,2,3,\ldots}^{\infty} B_{II}\{\sin(nN_p\theta+mN_s\theta)+\sin(nN_p\theta-mN_s\theta)\} \end{aligned} \tag{6.58}$$

The constant B_I is as follows

$$B_I = (nN_p)\mu_0 M_n \frac{B_1 B_2}{B_3} \tag{6.59}$$

The constants B_1, B_2 and B_3, in equation 6.59, are as follows

$$\begin{aligned} B_1 &= R_m R_r^{(nN_p+1)} \left[2\left(\frac{R_m}{R_r}\right)^{nN_p-1} - \left(nN_p+1\right) + \left(\frac{R_m}{R_r}\right)^{2nN_p} \left(nN_p-1\right) \right] \mu_3 \\ &- R_m R_r^{(nN_p+1)} \left[nN_p - \left(\frac{R_r}{R_m}\right)^{nN_p-1} \left(nN_p+1\right) - \left(\frac{R_m}{R_r}\right)^{nN_p+1} \left(nN_p-1\right) \right] \mu_1 \end{aligned} \tag{6.60}$$

$$B_2 = \left(\frac{R_o}{R_s}\right)^{nN_p} \left(\mu_2^2 - 1\right) + \left[\left(1 - \mu_2^2\right)\left(\frac{R_s}{R_o}\right)^{nN_p} - \left(1+\mu_2\right)^2 \left(\frac{R_o}{R_s}\right)^{nN_p} \right] + \left(\frac{R_s}{R_o}\right)^{nN_p} \left(1-\mu_2\right)^2 \tag{6.61}$$

$$B_3 = r\left[\left(K_1 + K_2\right)K_3 + \left(K_4 + K_5\right)K_6\right]\left(1 - n^2 N_p^2\right) \tag{6.62}$$

The constants K_1, K_2, K_3, K_4, K_5 and K_6 are given below:

$$K_1 = \left(1+\mu_1\right)\left(\frac{R_o}{R_s}\right)^{nN_p} \left[\left(1+\mu_2\right)^2 - \left(1-\mu_2\right)^2 \left(\frac{R_s}{R_o}\right)^{2nN_p} \right] \tag{6.63}$$

$$K_2 = \left(\mu_1 - 1\right)\left(\mu_2^2 - 1\right)\left(\frac{R_m}{R_r}\right)^{nN_p} \left(\frac{R_m}{R_s}\right)^{nN_p} \left[\left(\frac{R_o}{R_S}\right)^{nN_p} - \left(\frac{R_s}{R_o}\right)^{nN_p} \right] \tag{6.64}$$

$$K_3 = \mu_3 R_r R_m^{nN_p} + \mu_1 R_m R_r^{nN_p} \tag{6.65}$$

$$K_4 = \left(1+\mu_1\right)\left(\frac{R_r}{R_m}\right)^{nN_p} \left(\frac{R_o}{R_m}\right)^{nN_p} \left[\left(1+\mu_2\right)^2 - \left(1-\mu_2\right)^2 \left(\frac{R_s}{R_o}\right)^{2nN_p} \right] \tag{6.66}$$

$$K_5 = (\mu_1 + 1)(\mu_2^2 - 1)\left(\frac{R_r}{R_s}\right)^{nN_p}\left[\left(\frac{R_o}{R_S}\right)^{nN_p} - \left(\frac{R_s}{R_o}\right)^{nN_p}\right] \tag{6.67}$$

$$K_6 = \mu_1 R_m R_r^{nN_p} - \mu_3 R_r R_m^{nN_p} \tag{6.68}$$

The constant B_{II} in equation 6.58 is given as follows:

$$B_{II} = B_4 / B_5$$

$$\begin{aligned} B_4 = &-\left(\frac{R_R}{R_M}\right)^{nN_p + mN_s}\left(\frac{R_S}{R_M}\right)^{nN_p + mN_s}(\mu_2 - 1)(\mu_2 - \mu_3) \\ &+ (\mu_2 + 1)\left((\mu_2 - \mu_3)\left(\frac{R_R}{R_S}\right)^{nN_p + mN_s} + (\mu_2 + \mu_3)\left(\frac{R_S}{R_R}\right)^{nN_p + mN_s}\right) \\ &-\left(\frac{R_M}{R_S}\right)^{nN_p + mN_s}\left(\frac{R_M}{R_R}\right)^{nN_p + mN_s}(\mu_2 - 1)(\mu_2 + \mu_3) \end{aligned} \tag{6.69}$$

$$\begin{aligned} B_5 = &(\mu_2 - 1)(\mu_3 - \mu_2)\left(\frac{R_s}{R_M}\right)^{nN_p + mN_s}\left(\frac{R_R}{R_M}\right)^{nN_p + mN_s} \\ &+ (\mu_2 - 1)(\mu_3 + \mu_2)\left(\frac{R_M}{R_S}\right)^{nN_p + mN_s}\left(\frac{R_M}{R_R}\right)^{nN_p + mN_s} \\ &- (\mu_2 + 1)(\mu_3 + \mu_2)\left(\frac{R_s}{R_R}\right)^{nN_p + mN_s} - (\mu_2 + 1)(\mu_3 - \mu_2)\left(\frac{R_R}{R_S}\right)^{nN_p + mN_s} \end{aligned} \tag{6.70}$$

6.6 Back EMF Calculation

The voltage induced in the stator windings by varying magnetic field in the air gap is known as back-emf. Since all coils in the stator can be described in terms of a sequence of equivalent single tooth coils [1], the flux linked by each coil is the sum of fluxes linked by each individual tooth coils. Figure 6.10 shows coils and its single tooth equivalent. The flux linked by the coil in Figure 6.10 is given by

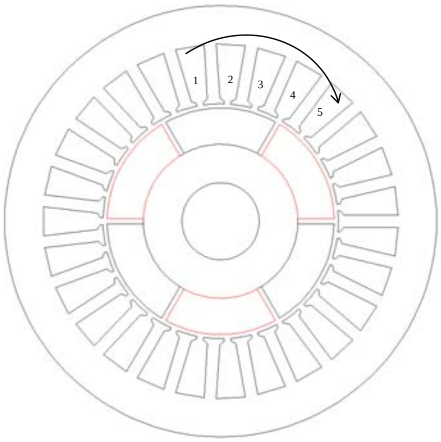


Figure 6.10a: A coil with a span of 4 slot pitches

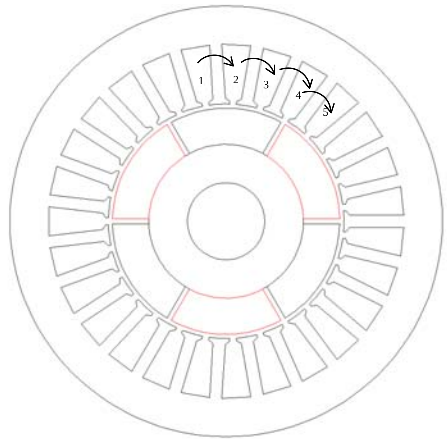


Figure 6.10b: Single slot equivalent of the coil

$$\phi_c(\theta)=\phi_1(\theta)+\phi_1(\theta-\theta_s)+\phi_1(\theta-2\theta_s) \tag{6.71}$$

where

θ_s is the angular tooth offset

ϕ_1 is the flux linked by first tooth

The flux linked by a tooth is given by

$$\phi=\int \vec{B}.d\vec{A} \tag{6.72}$$

the above equation, by substituting $\vec{B}$ the radial component of the airgap field at the stator inner surface becomes

$$\phi=\int_{-L/2}^{L/2}\int_{-\theta_s/2}^{\theta_s/2} B_{rI}(r,\theta)R_s\,d\theta dz \tag{6.73}$$

where $B_{rI}(r,\theta)$ is the airgap field at the stator surface and R_s is the inner radius of the stator and L is the length of the motor. In the above equation the integrand is independent of the axial direction. Hence the above equation can be simplified as

$$\phi = \frac{2LR_s}{N_p} \int_{-\theta_s/2}^{\theta_s/2} B_{rI}(r,\theta) d\theta \tag{6.74}$$

The back emf of a single tooth equivalent coil is given by

$$e = N_{turns} N_p / 2\omega \frac{d\phi}{d\theta} \tag{6.75}$$

where N_{turns} is the number of turns in the coil.

The back emf of a general coil is the sum of back EMFs of its single tooth equivalent coils. For the coil shown in Figure 6.10 the back EMF is given by

$$e_c(\theta) = e_1(\theta) + e_1(\theta - \theta_s) + e_1(\theta - 2\theta_s) \tag{6.76}$$

Having developed the necessary set of equations, this model is tested and results for the air gap field is compared with FEM in the next section. In section 6.10 the back emf obtained by the analytical method is compared with experimental results. The airgap field density is not compared with the experimental results due to difficulty in measurement of the airgap field. However since the back EMF is evaluated from the air gap field distribution, comparison of experimental values of back emf and that obtained by the analytical methods implicitly establish the validity of the analytical methods.

6.7 Cogging Torque Calculation Using Maxwell Stress Tensor

The Maxwell's stress gives the force per unit area produced by the magnetic field on a surface. In differential form it is described as

$$df = \frac{1}{2}\left(\vec{H}\left(\vec{B} \bullet \hat{n}\right) + \vec{B}\left(\vec{H} \bullet \hat{n}\right) - \left(\vec{H} \bullet \vec{B}\right)\hat{n}\right) \tag{6.77}$$

where $\hat{n}$ denotes the direction normal to the surface at the point of interest. The net force on an object is obtained by creating a surface totally enclosing the object of interest and integrating the magnetic stress over that surface. In rotating motors the tangential component of the force contributes to the torque. The tangential component of the force and torque acting on the surface that encloses the rotating part of the motor is given by:

$$f = \frac{1}{\mu_0} \int_S B_r B_\theta dS \tag{6.78}$$

and

$$T = rf \tag{6.79}$$

Using equation 6.79 the cogging torque in the motor can be determined. The comparisons between cogging torque results obtained analytically and by FEM are given in section 6.9.

6.8 Inductance Calculation

The total phase induction composed of air gap inductance L_g, slot leakage inductance L_s and end turn inductance L_e is expressed as follows [1]

$$L_{ph} = \frac{N_s}{3}\left(L_s + L_e + L_g\right) \tag{6.80}$$

where

$$L_g = \frac{N_{turns}^2 \mu_R \mu_0 \tau_c k_d}{4\left(h_m + \mu_R k_c g\right)} \tag{6.81}$$

$$L_s = N_{turns}^2 \left[\frac{\mu_0 d_3^2 L}{3A_s} + \frac{\mu_0 d_2 L}{\left(b_o + w_{si}\right)/2} + \frac{\mu_0 d_1 L}{w_s}\right] \tag{6.82}$$

$$L_e = \frac{N_{turns}^2 \mu_0 \tau_c}{8} \ln\left[\frac{\tau_c^2 \pi}{4A_s}\right] \tag{6.83}$$

In equation 6.81-6.83 A_s is the slot area for conductor, μ_0 is the permeability of free space, τ_c is the coil pitch, N_s is the number of slots, N_{turns} is the number of turns per coil. The air gap inductance is relatively small because of the low recoil permeability and large thickness of the magnet with respect to the air gap [1]. The other geometric dimensions are shown in Figure 6.11.

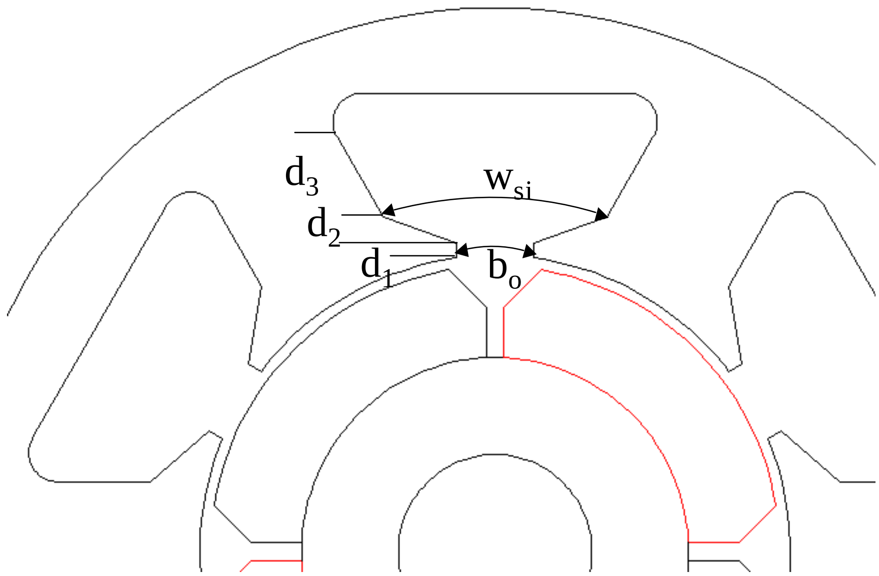


Figure 6.11: Dimensions of the stator for inductance calculation

6.9 Comparison of the Analytical Model with Finite Element Method.

The model developed in the previous section was applied to three phase slotless BLDC motor with radial, parallel, sinusoidal magnitude and sinusoidal angle magnetisation. The parameters of the motor taken are

$N_p = 6$, $N_s = 9$, $b_o = 1.5mm$, $R_s = 16.95mm$, $R_r = 13.25mm$, $R_m = 16.25mm$, $B_r = 1.2\text{T}$, $\mu_1 = 1000$, $\mu_2 = 1.05$, $\mu_3 = 1000$, $\alpha_m = 0.8$

The results obtained by our model and the FEM are shown in Figure 6.12 for radial magnetisation, Figure 6.13 for parallel magnetisation, Figure 6.14 for sinusoidal angle magnetisation and Figure 6.15 for sinusoidal amplitude magnetisation.

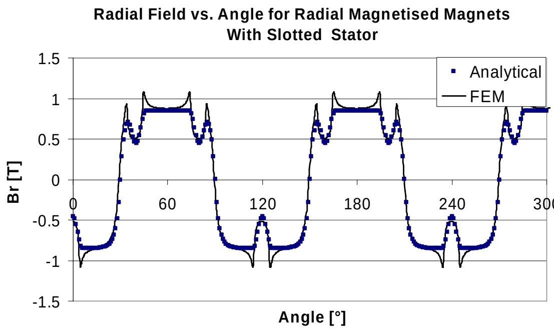


Figure 6.12a: Radial component of airgap field in a slotted motor with radially magnetised magnets (r=16.75)

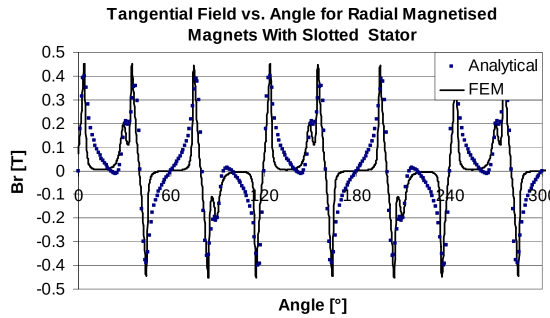


Figure 6.12b: Tangential component of airgap field in a slotted motor with radially magnetised magnets (r=16.75)

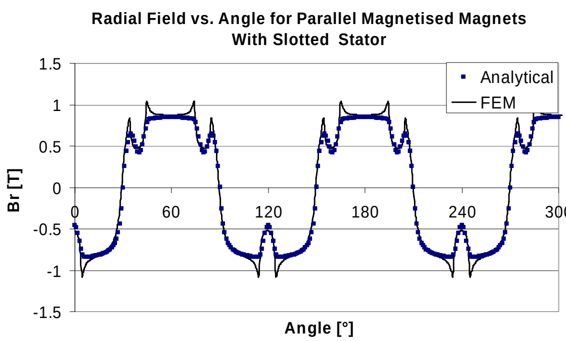


Figure 6.13a: Radial component of airgap field in a slotted motor with parallel magnetised magnets (r=16.75)

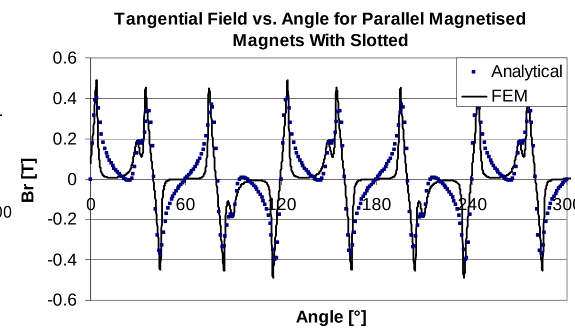


Figure 6.13b: Tangential component of airgap field in a slotted motor with parallel magnetised magnets (r=16.75)

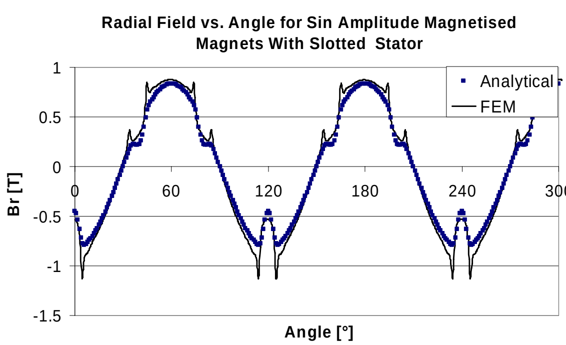


Figure 6.14a: Radial component of airgap field in a slotted motor with Sinusoidal amplitude magnetised magnets (r=16.75)

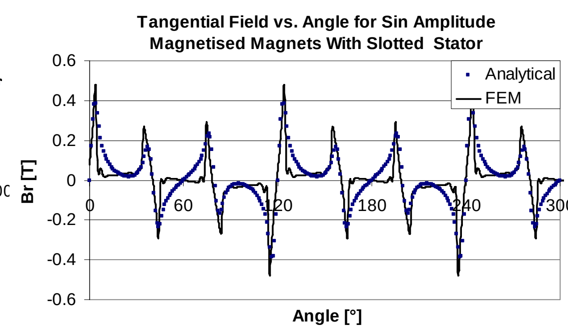


Figure 6.14b: Tangential component of airgap field in a slotted motor with Sinusoidal amplitude magnetised magnets (r=16.75)

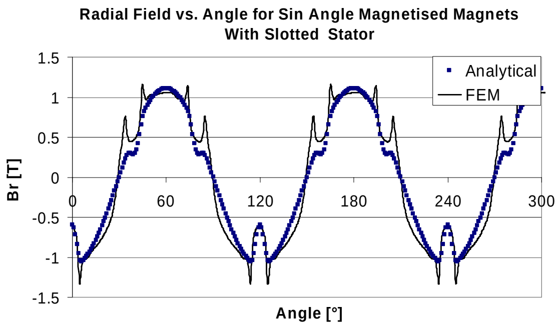


Figure 6.15a: Radial component of airgap field in a slotted motor with Sinusoidal angle magnetised magnets (r=16.75)

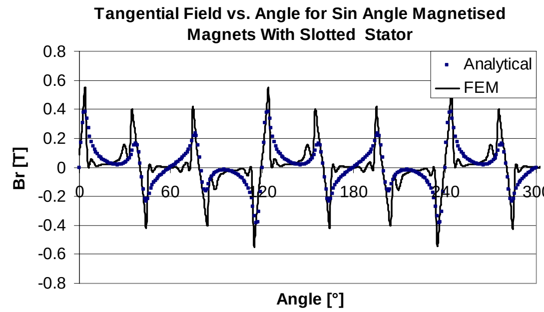


Figure 6.15b: Tangential component of airgap field in a slotted motor with Sinusoidal angle magnetised magnets (r=16.75)

From the above figures it can be seen that the results of analytical model are close to the results of FEM. The radial component of the field obtained by the analytical method follows the FEM results in amplitude and waveform shape. However at the edges of the slots the field due to FEM is higher than that obtained by analytical method. This difference can be attributed to the fact that at tooth edges flux concentration occurs and the relative permeance function is unable to take into account this flux concentration effect. The tangential component of the air gap field shows discrepancy with the FEM results.

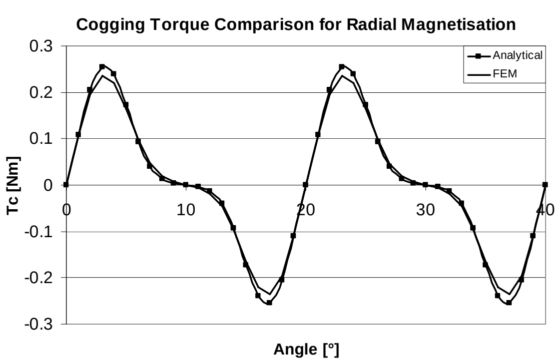


Figure 6.16a: Comparison of cogging torque obtained by analytical model and FEM for Radially Magnetised magnets

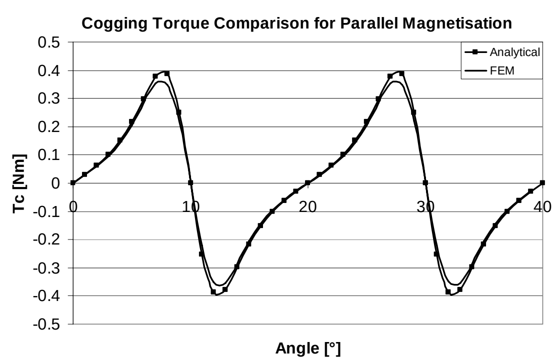


Figure 6.16b: Comparison of cogging torque obtained by analytical model and FEM for Parallel Magnetised magnets

The comparison of the cogging torque obtained by the analytical model and the FEM are shown in Figure 6.16. The parameters of the motor, for radial and parallel magnetised magnets, for which the cogging torque was calculated, are:

$N_p = 6,\ \mathrm{N_s} = 9, b_o = 1.5mm, R_s = 16.95mm,\ R_r = 13.25mm,\ R_m = 16.25mm,\ B_r = 1.2\mathrm{T},\ \mu_1 = 1000,$
$\mu_2 = 1.05,\ \mu_3 = 1000,\ \alpha_m = 0.8$

From Figure 6.16a it can be seen that the result of the cogging torque obtained by the analytical model is close to the FEM result. The shape of cogging torque from FEM is similar to that of the analytical values. For parallel magnetised magnets the results of cogging torque obtained by analytical method and FEM are shown in Figure 6.16b. In this case also the shape and the values of the cogging torque from analytical model and FEM is similar. The maximum value of cogging torque calculated by analytical model, for both radial and parallel magnetised magnets, is about 7% higher than the values obtained by FEM.

6.10 Comparison of the Analytical Model with Experimental Results.

In this section the results obtained by analytical method for back EMF (back EMF is expressed as voltage constant k_e) are compared with the experimental values of the back EMF. The parameters of the motor are as follows:

$N_p = 8,\ \mathrm{N_s} = 12, R_s = 11mm,\ R_r = 9.5mm,\ \mathrm{R_o} = 19.5, R_m = 10.8,\ B_r = 0.573\mathrm{T},\ \mu_1 = 550, \mu_2 = 1.05,$
$\mu_3 = 550, \alpha_m = 1, N_{turns} = 10$

The four coils of a phase are connected in parallel and the phases are connected in series. The rotor and stator yoke is made of M-250-35A steel. The B-H curve of this steel is shown in Figure 6.17.

From the Figure 6.17 it can be seen that the material has linear characteristics up to 1.2T, i.e. the relative permeability of the material is constant and its value is $\mu_1 = 550$. This value of relative permeability is used in the analytical model. The back EMF obtained by the analytical method for the above motor and the experimental values are shown in is shown in

Figure 6.18. From the above comparison it can be seen that the results obtained by analytical method for k_e closely resemble with the experimental results.

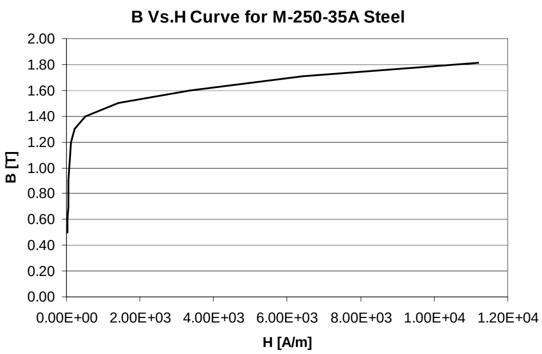


Figure 6.17: B-H curve of M-250-35A steel

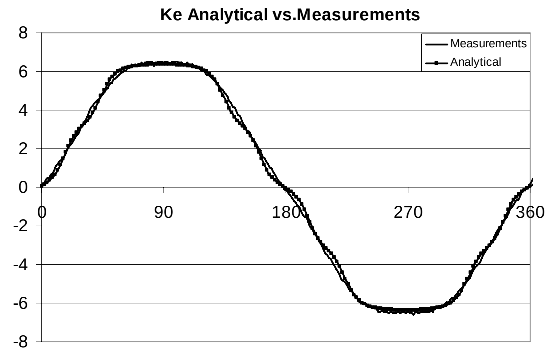


Figure 6.18: Comparison between analytical result and experimental values for back emf

6.11 Conclusion

An analytical technique, to predict the airgap field distribution due to permanent magnets mounted on the rotor of a BLDC motor with slotless and slotted stator, has been presented in this chapter. This model takes into account that the stator yoke has finite permeability and finite thickness. Different types of magnetisation of the permanent magnets viz., radial, parallel, sinusoidal magnitude and sinusoidal angle magnetisation have been considered. The model developed in this work has been validated by finite element analysis. There is discrepancy between the radial component of the airgap field obtained by analytical method and FEM. This difference is due to flux concentration effect at the tooth edges and the relative permeance function is unable to take into account this effect. Furthermore, the cogging torque results obtained by analytical model and FEM were compared and were similar to each other except that the maximum value of cogging torque by analytical method was about 7% higher than the values of FEM results. The analytical model was also compared with experimental results based on back EMF values and both the results were in good agreement. Hence, the analytical model gives results that are very close to FEM. The other major advantage of the analytical method is that it is very fast as compared to the FEM and can be used in the initial multiobjective optimisation of the BLDC motor. After having developed the magnetic model of the motor, in the next chapter the steady state model of the motor is presented. The results obtained from the magnetic models viz***., shape and magnitude of the induced voltage, inductance and resistance***, are used as the input parameters for the steady state model.

6.12 References

[1] D. Hanselman, 2004: The Writers' Collective, Brushless permanent magnet motor design.

[2] G. Eid and A. Mouillet, "Transistorized DC brushless micromotor with rare-earth permanent magnets," presented at ICEM 84, Luassane (Switzerland), 1984.

[3] Q. Gu and H. Gao, "Effect of slotting in PM electrical machines," Electrical Machines and Power Systems, vol. 10, pp. 273-284, 1985.

[4] N. Boules, "Two dimentional field analysis of cylindrical machines with permanent magnet excitation," IEEE Transaction on Industrial Application, vol. 20, pp. 1267-1277, 1984.

[5] N. Boules, "Prediction of no-load flux density distribution in PM machines," IEEE Transaction on Industrial Application, vol. 21, pp. 633-643, 1985.

[6] Z. Q. Zhu, D. Howe, E. Bolte, and B. Ackermann, "Instantaneous magnetic field distribution in brushless permanent magnet dc motors Part I: open-circuit field," IEEE Transactions on Magnetics, vol. 29, pp. 124-135, 1993.

[7] K. F. Rasmussen, "Analytical prediction of magnetic field from surface mounted permanent magnet motors," presented at International Conference on Electrical Machines and Drives, Seattle WA, 1999.

[8] Q. Zhu, D. Howe, and C. C. Chan, "Improved analytical model of predicting the magnetic field distribution in brushless permanent magnet motor," IEEE Transactions on Magnetics, vol. 38, pp. 229-238, 2002.

[9] Z. Q. Zhu, D. Howe, E. Bolte, and A. Ackermann, "Instantaneous magnetic field distribution in brushless permanent meagnet dc motors, Part III: Effect of stator slotting," IEEE Transactions on Magnetics, vol. 29, 1993.

Chapter 7

Steady State Performance model of Permanent Magnet Brushless DC Motor Drive

7.1 Introduction

In this chapter the steady state performance model of the BLDC motor drive is presented. The model presented in this chapter is formal (mathematical) and deterministic and is suitable for the Synthesis phase as well as the Final Analysis phase of PDM.

The steady state model to determine the performance of an electronically commutated permanent magnet motor was presented by Nucera et.al. [1]. However, they considered back emf to be sinusoidal in shape. The analysis of a BLDC motor with trapezoidal back emf was presented by Huth [2]. Zhu et. al [3] presented a model for steady state analysis of the BLDC motor for any shape of back emf. In this model in order to take into account the shape of the back emf a look up table is used and numerical methods are used to determine the solution of the resulting integrals.

In this chapter an analytical method is developed for determination of steady state performance of permanent magnet brushless DC motor with any shape of the back emf. In order to facilitate the solution the back emf is expressed as Fourier series. The main contributions of the presented work are:

1. Valid for any shape of back emf, without the necessity of look up tables
2. Takes into account normal, advanced and delayed commutation
3. Closed form solution of the steady state current
4. Takes into account both 120° and 180° conduction modes for the voltage source inverter

Since this model is analytical the initial conditions that yield steady state solutions are calculated directly by exploiting the periodicity of the currents in the stator. The main advantages of the presented analytical method are:

1. a fast means for determining the operating characteristics such as torque vs. speed, current vs. speed, etc.
2. a fast means for determining torque ripples and influence of firing delay on the torque ripples.
3. a fast means for determining the impact of back emf shape on torque ripples.

In section 7.2 the model of a permanent magnet brushless DC (BLDC) motor together with the converter is presented. The steady state solution for 120° conduction mode is given in section 7.3. Section 7.4 deals with steady state solution for 180° conduction mode. The comparison between analytical model and experimental verification is given in section 7.5. Some additional results obtained by analytical method are given in section 7.6. In section 7.7 a simple model of switching losses in the MOSFET switches is given and finally the conclusions are drawn in section 7.8.

7.2 Converter Motor Model

In this section a model of BLDC motor used in computation of steady state performance is presented. In Figure 7.1 electric equivalent circuit of a typical BLDC motor is shown.

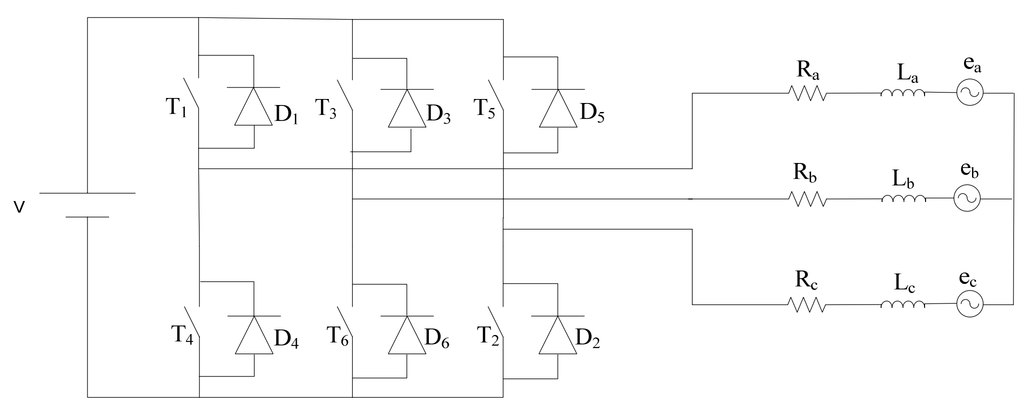


Figure 7.1: The general schematic diagram of the BLDC motor drive

In the present analysis it is assumed that the phases of the motor are wye connected. The following assumptions are:

1. The air gap of the motor is free of saliency. This is due to the fact that magnet material such as samarium cobalt, neobdium boron or ferrite magnets have permeability equal to that of air.
2. The back EMF is independent of armature reaction for normal operating range.

The coupled circuit equations of the stator windings in terms of the motor electrical constants are

$$[V]=[R][i]+[L]\frac{d[i]}{dt}+[e] \tag{7.1}$$

where $[V]=\left[V_a,V_b,V_c\right]'$ (7.2)

$$[R]=\begin{bmatrix} R_{ph} & 0 & 0 \\ 0 & R_{ph} & 0 \\ 0 & 0 & R_{ph} \end{bmatrix} \tag{7.3}$$

$$[i]=\left[i_a,i_b,i_c\right]' \tag{7.4}$$

$$[L]=\begin{bmatrix} L_{ph} & 0 & 0 \\ 0 & L_{ph} & 0 \\ 0 & 0 & L_{ph} \end{bmatrix} \tag{7.5}$$

$$[e]=\left[e_a,e_b,e_c\right]' \tag{7.6}$$

The parameters e_a,e_b,e_c, R_{ph} and L_{ph} are determined from the magnetic model discussed in chapter 6. The back emf e_a, e_b, e_c is periodic and can be of any shape. To simplify the solution the back emf is represented as Fourier series:

$$e_a(t) = \sum_{n=1}^{\infty} a_n \cos(n\omega_r t + \theta_n) \tag{7.7}$$

$$e_b(t) = \sum_{n=1}^{\infty} a_n \cos\left(n(\omega_r t - \frac{2\pi}{3}) + \theta_n \right) \tag{7.8}$$

$$e_c(t) = \sum_{n=1}^{\infty} a_n \cos\left(n(\omega_r t + \frac{2\pi}{3}) + \theta_n \right) \tag{7.9}$$

The magnitude (a_n) and argument (θ_n) of the harmonics of the back emf waveforms can be determined using Fast Fourier Transform and ω_r is the rotational speed of the rotor. The inverter is triggered symmetrically. Hence, the applied stator voltage waveform has the following relationship [1]:

$$V_a\left(\omega_r t + \frac{\pi}{3} \right) = -V_b(\omega_r t) \tag{7.10}$$

$$V_b\left(\omega_r t + \frac{\pi}{3} \right) = -V_c(\omega_r t) \tag{7.11}$$

$$V_c\left(\omega_r t + \frac{\pi}{3} \right) = -V_a(\omega_r t) \tag{7.12}$$

Since the differential equations representing the system (equation 7.1) are time invariant, the stator currents i_a, i_b and i_c also show the same symmetry relation as the voltage. The symmetry relations of the current are as follows:

$$i_a\left(\omega_r t + \frac{\pi}{3} \right) = -i_b(\omega_r t) \tag{7.13}$$

$$i_b\left(\omega_r t + \frac{\pi}{3} \right) = -i_c(\omega_r t) \tag{7.14}$$

$$i_c\left(\omega_r t + \frac{\pi}{3} \right) = -i_a(\omega_r t) \tag{7.15}$$

The differential equations of the system and the above symmetry conditions are used to determine the steady state performance of BLDC motor.

7.3 Solution for 120° conduction mode

In this section the steady state solution for 120° conduction mode is derived. In order to facilitate the solution it is assumed that the motor has reached steady state operation, i.e. the rotor has constant speed. As a result of the symmetry conditions discussed in the previous section, it is sufficient to obtain the solution for one switching interval. The solution for one switching interval can then be used to generate the solution for the remaining cycle using symmetry relations, equation 7.13-7.15.

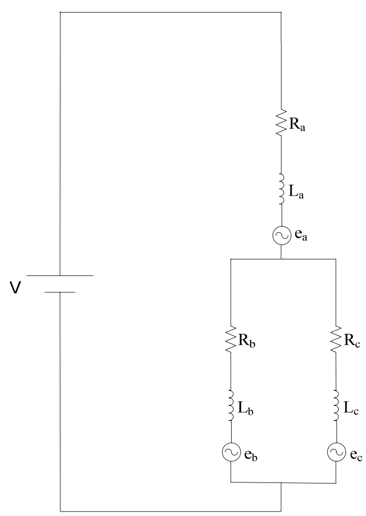


Figure 7.2: Commutation period $I_o > 0$

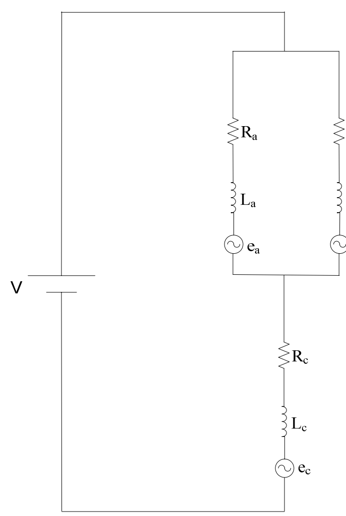


Figure 7.3: Commutation period $I_o < 0$

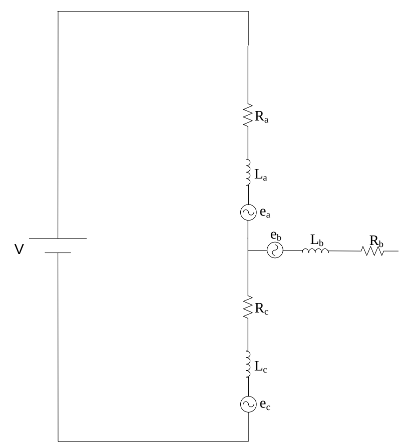


Figure 7.4: Conduction period

In 120° conduction mode each transistor conducts for 60° as shown in Figure 7.2. The 60° duration for analysis starts when gate T6 is turned off and ends when the gate T3 is turned on. When the transistor T6 is turned off, due to inductance of the winding, the current (i_b) in phase B continues to flow. The phase B current i_b flows through either diode D3 or D6 depending upon the direction of i_b at the end of the previous switching interval. The Figure 7.3 shows the circuit condition when the current in phase B continues to flow. Once the current in phase B dies out, then only phase A and phase C conduct, Figure 7.4. Hence the switching process consists of two parts, first when all the three phases conduct current, known as commutation period and in the second part only two phases conduct current, known as conduction period. In the present analysis at the start of the commutation period the rotor angle (θ_r) is defined as,

$$\theta_r = -\phi + \frac{\pi}{6} \tag{7.16}$$

where ϕ is the advanced firing angle.

To develop the steady state model of the motor the commutation angle (θ_c) is determined first. The solution for steady state is obtained in two steps. In the first step the differential equations for the commutation and the conduction modes are set up. In the second part the commutation angle (θ_c) and the initial values of current are calculated.

7.3.1 System Equations for Commutation Mode

During the commutation period all the three phases conduct and the circuit conditions are shown in Figure 7.3. The general network equations for commutation mode are given by

$$\begin{aligned}
\frac{di_a(t)}{dt} &= -\frac{R_{ph}}{L_{ph}} i_a - \frac{1}{L_{ph}}\left(\frac{2e_a(t) - e_b(t) - e_c(t)}{3}\right) + \frac{V_a}{L_{ph}} \\
\frac{di_b(t)}{dt} &= -\frac{R_{ph}}{L_{ph}} i_b - \frac{1}{L_{ph}}\left(\frac{2e_b(t) - e_c(t) - e_c(t)}{3}\right) + \frac{V_b}{L_{ph}} \\
\frac{di_c(t)}{dt} &= -\frac{R_{ph}}{L_{ph}} i_c - \frac{1}{L}\left(\frac{2e_c(t) - e_a(t) - e_b(t)}{3}\right) + \frac{V_c}{L_{ph}}
\end{aligned} \tag{7.17}$$

Since the three phase voltages add to zero it can be shown that for $i_b < 0$

$$\begin{pmatrix} V_a \\ V_b \\ V_c \end{pmatrix} = \begin{pmatrix} \frac{1}{3}V_{dc} \\ \frac{1}{3}V_{dc} \\ -\frac{2}{3}V_{dc} \end{pmatrix} \tag{7.18}$$

and for $i_b > 0$

$$\begin{pmatrix} V_a \\ V_b \\ V_c \end{pmatrix} = \begin{pmatrix} \frac{2}{3}V_{dc} \\ -\frac{1}{3}V_{dc} \\ -\frac{1}{3}V_{dc} \end{pmatrix} \tag{7.19}$$

To simplify the solutions of system of equations (7.17) the rotor angle $\omega_r t$ is replaced by $\omega_r t - \phi + \frac{\pi}{6}$ in equation 7.7 to equation 7.8.

The differential equation of the system equation 7.1 as can be written as

$$\frac{d}{dt}[i] = \left[\frac{R}{L}\right][i] + \left[\frac{1}{L}\right][V - e] \tag{7.20}$$

where $$\left[\frac{R}{L}\right] = \begin{bmatrix} -\frac{R_{ph}}{L_{ph}} & 0 & 0 \\ 0 & -\frac{R_{ph}}{L_{ph}} & 0 \\ 0 & 0 & -\frac{R_{ph}}{L_{ph}} \end{bmatrix} \tag{7.21}$$

$$\left[\frac{1}{L}\right] = \begin{bmatrix} \frac{1}{L_{ph}} & 0 & 0 \\ 0 & \frac{1}{L_{ph}} & 0 \\ 0 & 0 & \frac{1}{L_{ph}} \end{bmatrix} \tag{7.22}$$

and

$$[V-e]=\begin{bmatrix} V_a-\dfrac{2e_a(t)-e_b(t)-e_c(t)}{3} \\ V_b-\dfrac{2e_b(t)-e_c(t)-e_a(t)}{3} \\ V_c-\dfrac{2e_c(t)-e_a(t)-e_b(t)}{3} \end{bmatrix} \tag{7.23}$$

The solution of the time invariant first order differential equation is of the form

$$[i(t)]=e^{[A]t}\left[i_o\right]+\int_0^t e^{[A]\tau}[B][u(\tau)]d\tau \tag{7.24}$$

where

$$[A]=\left[\frac{R}{L}\right],\ [B]=\left[\frac{1}{L}\right],\ [u(\tau)]=[V-e]$$

For the commutation period, the initial condition for the current is as follows

$$\left[i_o\right]=\begin{bmatrix} i_a(0) \\ i_b(0) \\ -\left(i_a(0)+i_b(0)\right) \end{bmatrix} \tag{7.25}$$

The complete solution of equation 7.24 is of the form

$$i_a(t)=i_a(0)e^{-\left(\frac{R_{ph}t}{L_{ph}}\right)}-\frac{V_a}{R_{ph}}\left(e^{\left(-\frac{R_{ph}t}{L_{ph}}\right)}-1\right)+K_1\left[R_{ph}\sin\left(\theta_1\right)\cos(\frac{n\pi}{3})-n\omega L_{ph}\sin\left(\theta_2\right)\sin(\frac{n\pi}{3})\right]$$
$$+K_1\left[-R_{ph}\sin\left(\theta_3\right)\cos(\frac{n\pi}{3})+n\omega L_{ph}\sin\left(\theta_4\right)\sin(\frac{n\pi}{3})\right]e^{\left(-\frac{R_{ph}t}{L_{ph}}\right)} \tag{7.26}$$

where

$$\theta_1 = n\omega t - n\phi - \frac{n\pi}{6} + \xi,\ \theta_2 = n\omega t - n\phi + \frac{n\pi}{6} + \xi,$$
$$\theta_3 = -n\phi + \frac{n\pi}{6} + \xi,\ \theta_4 = n\phi - \frac{n\pi}{6} - \xi,$$
$$K_1 = \frac{2a_n}{3}\frac{R_{ph} + n\omega L_{ph}}{R_{ph}^2 + n^2\omega^2 L_{ph}^2}\sin(\frac{n\pi}{3})$$

and

$$i_b = i_b(0)e^{-\left(\frac{R_{ph}t}{L_{ph}}\right)} - \frac{V_b}{R_{ph}}\left(e^{\left(-\frac{R_{ph}t}{L_{ph}}\right)} - 1\right) - K_2\left[\sin\left(\theta_5\right)\sin\left(\frac{n\pi}{3}\right) + \sin\left(\theta_6\right)\sin\left(\frac{2n\pi}{3}\right)\right]$$
$$+ K_2\left[\sin\left(\theta_7\right)\sin\left(\frac{2n\pi}{3}\right) + \sin\left(\theta_8\right)\sin\left(\frac{n\pi}{3}\right)\right]e^{\left(-\frac{R_{ph}t}{L_{ph}}\right)} \tag{7.27}$$

where

$$\theta_5 = n\omega t - n\phi - \frac{n\pi}{6} + \xi,\ \theta_6 = n\omega t - n\phi + \frac{n\pi}{6} + \xi,$$
$$\theta_7 = -n\phi + \frac{n\pi}{6} + \xi,\ \theta_8 = -n\phi - \frac{n\pi}{6} + \xi,$$
$$K_2 = \frac{2a_n}{3}\frac{R_{ph} + n\omega L_{ph}}{R_{ph}^2 + n^2\omega^2 L_{ph}^2}$$

The current is phase C is given by

$$i_c(t) = -(i_a(t) + i_b(t)) \tag{7.28}$$

7.3.2 System Equations for Conduction Mode

The conduction period starts when the diode D3 of D6 stops conduction $t = t_c$, i.e. the phase B current is zero, Figure 7.4. The network equations for conduction mode is given by

$$\frac{di_a(t)}{dt} = -\frac{R_{ph}}{L_{ph}} i_a - \frac{1}{L_{ph}} \left(\frac{e_a(t) - e_c(t)}{2} \right) + \frac{1}{2} \frac{V_a}{L_{ph}}$$
$$\frac{di_c(t)}{dt} = -\frac{R_{ph}}{L_{ph}} i_c - \frac{1}{L_{ph}} \left(\frac{e_c(t) - e_a(t)}{2} \right) + \frac{1}{2} \frac{V_c}{L_{ph}} \tag{7.29}$$

where

$$\begin{pmatrix} V_a \\ V_c \end{pmatrix} = \begin{pmatrix} \frac{1}{2} V_{dc} \\ -\frac{1}{2} V_{dc} \end{pmatrix}$$

For the conduction period, the initial condition for the current is as follows

$$\left[i_o \right] = \begin{bmatrix} i_a(t_c) \\ 0 \\ -i_a(t_c) \end{bmatrix} \tag{7.30}$$

The solution of the time invariant first order differential equation is of the form

$$\left[i(t) \right] = e^{[A](t - t_c)} \left[i_o \right] + \int_{t_c}^{t} e^{[A][t-\tau]} [B][u(\tau)] d\tau \tag{7.31}$$

The current in phase A, obtained by solving equation 7.31, is given by

$$i_a(t) = i_a(t_c) e^{-\left(\frac{R_{ph}(t - t_c)}{L_{ph}} \right)} + \frac{V_{dc}}{2R_{ph}} \left(1 - e^{-\left(\frac{R_{ph}\left(t - t_c\right)}{L_{ph}} \right)} \right)$$

$$-2a_n R_{ph} \sin\left(\frac{n\pi}{3}\right)\left(R_{ph} \sin\left(n\omega t - n\phi + \frac{n\pi}{2} + \xi\right) - n\omega L_{ph} \cos\left(n\omega t - n\phi + \frac{n\pi}{2} + \xi\right)\right) e^{\left(\frac{R_{ph}\left(t - t_c\right)}{L_{ph}}\right)}$$
$$+2a_n R_{ph} \sin\left(\frac{n\pi}{3}\right)\left(R_{ph} \sin\left(n\omega t_c - n\phi + \frac{n\pi}{2} + \xi\right) - n\omega L_{ph} \cos\left(n\omega t - n\phi + \frac{n\pi}{2} + \xi\right)\right) \tag{7.32}$$

The current is phase C is same as current in phase A and current in phase B is zero, i.e.,

$$i_c(t) = -i_b(t) \tag{7.33}$$

$$i_b(t) = 0 \tag{7.34}$$

Above set of equations are valid for the time duration $t \in \left[t_c, \dfrac{\pi}{3\omega_r} \right]$.

7.3.3 Calculation of Commutation Time

The initial values of the current $i_a(0)$, $i_b(0)$ and t_c in equation 7.26, equation 7.27, equation 7.32 respectively are unknown. To obtain the initial values the symmetry of the solution is exploited. When the gate G6 is turned off, in case of 120° conduction mode of the inverter, the current in phase C falls to zero $i_c(0) = 0$ and $i_a(0) = -i_b(0) = I_0$. Thus the initial condition for the current is as follows

$$\begin{bmatrix} i_a(0) \\ i_b(0) \\ i_c(0) \end{bmatrix} = \begin{bmatrix} I_0 \\ -I_0 \\ 0 \end{bmatrix} \tag{7.35}$$

When the conduction period ends, i.e. the gate G3 is turned off, then equation 7.13-equation 7.14 imply

$$\begin{bmatrix} i_a\left(\dfrac{\pi}{3\omega_r}\right) \\ i_b\left(\dfrac{\pi}{3\omega_r}\right) \\ i_c\left(\dfrac{\pi}{3\omega_r}\right) \end{bmatrix} = \begin{bmatrix} I_0 \\ 0 \\ -I_0 \end{bmatrix} \tag{7.36}$$

The above equations (equation 7.35, equation 7.36) provide the necessary and sufficient condition for determining the initial values of the current $i_a(0)$, $i_b(0)$ and the conduction duration t_c. The commutation period ends when the current in phase B falls to zero, i.e. $i_b(t_c)=0$. Thus by substituting $t=t_c$ and $i_a(0)=-i_b(0)=I_0$ in equation 7.27 gives,

$$
\begin{aligned}
0=&-I_0 e^{-\left(\frac{R_{ph}t_c}{L_{ph}}\right)}-\frac{V_b}{R_{ph}}\left(e^{\left(-\frac{R_{ph}t_c}{L_{ph}}\right)}-1\right)\\
&-K_2\left[\sin\left(n\omega t_c - n\phi-\frac{n\pi}{6}+\xi\right)\sin\left(\frac{n\pi}{3}\right)+\sin\left(n\omega t_c - n\phi+\frac{n\pi}{6}+\xi\right)\sin\left(\frac{2n\pi}{3}\right)\right]\\
&+K_2\left[\sin\left(-n\phi+\frac{n\pi}{6}+\xi\right)\sin\left(\frac{2n\pi}{3}\right)+\sin\left(-n\phi-\frac{n\pi}{6}+\xi\right)\sin\left(\frac{n\pi}{3}\right)\right]e^{\left(-t_c\right)}
\end{aligned}
\tag{7.37}
$$

where

$$K_2=\frac{2a_n}{3}\frac{R_{ph}+n\omega L_{ph}}{R_{ph}{}^2+n^2\omega^2L_{ph}{}^2}$$

Again by substituting $t=\frac{\pi}{3\omega}$ and $i_a\left(\frac{\pi}{3\omega}\right)=I_0$ in equation 7.32 gives,

$$
\begin{aligned}
I_0=&\,i_a(t_c)e^{-\left(\frac{R_{ph}}{L_{ph}}\left(\frac{\pi}{3\omega}-t_c\right)\right)}+\frac{V_{dc}}{2R_{ph}}\left(1-e^{-\left(\frac{R_{ph}}{L_{ph}}\left(\frac{\pi}{3\omega}-t_c\right)\right)}\right)\\
&-2a_nR_{ph}\sin\left(\frac{n\pi}{3}\right)\left(R_{ph}\sin\left(n\frac{\pi}{3}-n\phi+\frac{n\pi}{2}+\xi\right)-n\omega L_{ph}\cos\left(n\frac{\pi}{3}-n\phi+\frac{n\pi}{2}+\xi\right)\right)e^{\left(\frac{R_{ph}}{L_{ph}}\left(\frac{\pi}{3\omega}-t_c\right)\right)}\\
&+2a_nR_{ph}\sin\left(\frac{n\pi}{3}\right)\left(R_{ph}\sin\left(n\omega t_c-n\phi+\frac{n\pi}{2}+\xi\right)-n\omega L_{ph}\cos\left(n\omega t_c-n\phi+\frac{n\pi}{2}+\xi\right)\right)
\end{aligned}
\tag{7.38}
$$

In the above expression equation 7.38 the value of $i_a(t_c)$ is obtained by substituting $t=t_c$ and $i_a(0)=I_0$ in equation 7.26 as follows,

$$i_a(t_c)=I_0 e^{-\left(\frac{R_{ph}t_c}{L_{ph}}\right)}-\frac{V_a}{R_{ph}}\left(e^{\left(-\frac{R_{ph}t}{L_{ph}}\right)}-1\right)$$
$$+K_1\left[R_{ph}\sin\left(n\omega t_c\text{-}n\phi-\frac{n\pi}{6}+\xi\right)\cos(\frac{n\pi}{3})-n\omega L_{ph}\sin\left(n\omega t_c\text{-}n\phi+\frac{n\pi}{6}+\xi\right)\sin(\frac{n\pi}{3})\right]$$
$$+K_1\left[-R_{ph}\sin\left(\text{-}n\phi+\frac{n\pi}{6}+\xi\right)\cos(\frac{n\pi}{3})+n\omega L_{ph}\sin\left(n\phi-\frac{n\pi}{6}-\xi\right)\sin(\frac{n\pi}{3})\right]e^{\left(-\frac{R_{ph}t_c}{L_{ph}}\right)} \tag{7.39}$$

where

$$K_1=\frac{2a_n}{3}\frac{R_{ph}+n\omega L_{ph}}{R_{ph}{}^2+n^2\omega^2L_{ph}{}^2}\sin(\frac{n\pi}{3})$$

Substituting equation 7.38 in equation 7.37 for I_0 results in an expression that has only one unknown variable t_c. This expression is a transcendental function of t_c and can be solved using iterative methods only. Once the value of t_c is determined, the initial value of the current I_0 can be obtained from equation 7.38. Hence the solution for the interval $\left[0,\frac{\pi}{3\omega}\right]$ is given by equation 7.26-equation 7.28 and equation 7.32 – equation 7.34. The solution for the remaining period is obtained by exploiting the symmetry condition given in equation 7.13-equation 7.15.

7.4 Solution for 180° conduction mode

When the inverter operates in 180° conduction mode, three transistors conduct at any given instant of time and hence current flows in all the three phases of the motor. As in the previous case of 120° mode conduction the solution is obtained in two steps.

7.4.1 System Equations for Commutation Mode

The equations 27-29 obtained in previous section describe the system dynamics over the complete interval $0 \le t \le \pi/3\omega$.

7.4.2 Initial Conditions

Similar to 120° mode conduction proper initial conditions must be determined for 180° mode conduction to define complete solution. In case of 180° mode of operation all the three phases of the stator conduct at any given instant of time, hence the initial condition can be expressed as follows:

$$i_a\left(\frac{\pi}{3\omega_r}\right) = -i_b(0) \tag{7.40}$$

$$i_b\left(\frac{\pi}{3\omega_r}\right) = i_{a(0)} + i_b(0) \tag{7.41}$$

The above relations provide enough and sufficient condition to determine the initial conditions. To determine the initial values of the current the first step is to set $t = \pi/3\omega$ in equation 7.26 and equation 7.27. This results in a system of two equations with two unknowns $i_a(0)$ and $i_b(0)$. Substituting $t = \pi/3\omega$ in equation 7.26 gives the following expression:

$$\begin{aligned} -i_b(0) = {} & i_a(0)e^{-\left(\frac{R_{ph}\pi}{3\omega_r L_{ph}}\right)} - \frac{V_a}{r}\left(e^{\left(-\frac{R_{ph}\pi}{3\omega_r L_{ph}}\right)} - 1\right) \\ & + K_1\left[R_{ph}\sin\left(\theta_1\right)\cos\left(\frac{n\pi}{3}\right) - n\omega L_{ph}\sin\left(\theta_2\right)\sin\left(\frac{n\pi}{3}\right)\right] \\ & + K_1\left[-R_{ph}\sin\left(\theta_3\right)\cos\left(\frac{n\pi}{3}\right) + n\omega L_{ph}\sin\left(\theta_4\right)\sin\left(\frac{n\pi}{3}\right)\right]e^{\left(-\frac{R_{ph}\pi}{3\omega_r L_{ph}}\right)} \end{aligned} \tag{7.42}$$

where

$$\theta_1 = \frac{n\pi}{6} - n\phi + \xi,\ \theta_2 = \frac{n\pi}{2} - n\phi + \xi,$$
$$\theta_3 = -n\phi + \frac{n\pi}{6} + \xi,\ \theta_4 = n\phi - \frac{n\pi}{6} - \xi,$$
$$K_1 = \frac{2a_n}{3} \frac{R_{ph} + n\omega L_{ph}}{R_{ph}^2 + n^2\omega^2 L_{ph}^2} \sin(\frac{n\pi}{3})$$

Similarly substituting $t = \pi/3\omega$ in equation 7.27 yields gives the following expression:

$$i_a(0) + i_b(0) = i_b(0) e^{-\left(\frac{R_{ph}\pi}{3\omega L_{ph}}\right)} - \frac{V_b}{R_{ph}}\left(e^{\left(-\frac{R_{ph}\pi}{3\omega_r L_{ph}}\right)} - 1 \right)$$
$$-K_2 \left[\sin\left(\theta_5\right)\sin\left(\frac{n\pi}{3}\right) + \sin\left(\theta_6\right)\sin\left(\frac{2n\pi}{3}\right)\right] \tag{7.43}$$
$$+K_2 \left[\sin\left(\theta_7\right)\sin\left(\frac{2n\pi}{3}\right) + \sin\left(\theta_8\right)\sin\left(\frac{n\pi}{3}\right)\right] e^{\left(-\frac{R_{ph}\pi}{3\omega L_{ph}}\right)}$$

where

$$\theta_5 = \frac{n\pi}{6} - n\phi + \xi,\ \theta_6 = \frac{2n\pi}{3} - n\phi + \xi,\ \theta_7 = -n\phi + \frac{n\pi}{6} + \xi,\ \theta_8 = n\phi - \frac{n\pi}{6} - \xi$$

The equation 7.42 and equation 7.43 have two unknown variables $i_a(0)$ and $i_b(0)$, hence solving these equations gives the values of the initial conditions. Once the initial conditions are known the currents for the complete period are determined using the symmetry relation given in equation 7.13 to equation 7.15.

In the next section the comparison between the results obtained by analytical model and measurement are given.

7.5 Comparison between results of Measurements and Analytical Model

This section deals with the comparison between the analytical model results and the measurement results. The phase resistance and phase inductance of the motor, on which measurement are performed, are $R_{ph} = 0.75\Omega$ and $L_{ph} = 0.71mH$ respectively. The number of poles of the motor are $N_p = 8$ and the magnitude of the back emf (K_e) is $K_e = 61 mVs/rad$. The mechanical speed of the motor is 3000 revolutions per minute and the input voltage to the inverter is $V_{dc} = 12V$. The shape of K_e as measured is shown in Figure 7.5 and the comparison of the currents obtained by analytical model and the measurement is shown in Figure 7.6.

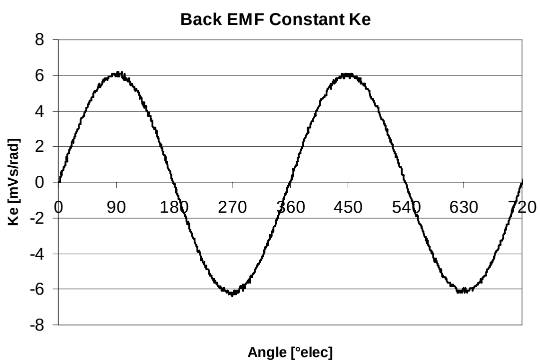


Figure 7.5: Measured values of back emf constant

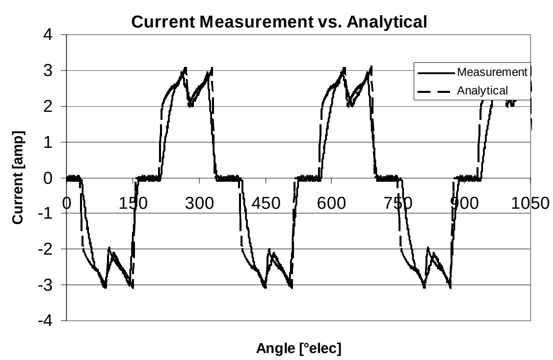


Figure 7.6: Comparison between current obtained by Measurement and analytical model

From the above comparison it can be seen that the analytical model gave results similar to the measurement. The switching periods given by the analytical model are the same as obtained by measurements.

In the next section the results obtained by analytical model for different back emf shapes and firing delays are given.

7.6 Results of analytical Model

The model developed above was implemented to determine the steady state current performance of a permanent magnet motor. The phase resistance of the motor is $r_{ph} = 30m\Omega$ and the phase inductance is $L_{ph} = 0.19mH$. The number of poles in the motor are $N_p = 6$, the magnitude of the back emf (K_e) is $K_e = 0.05077 Vs/rad$ and the mechanical speed of the motor is 1000 revolutions per minute. The battery voltage is $V_{dc} = 12V$. The shape of the back emf for phase A is shown in Figure 7.7. The steady state solution of the current for 120° conduction mode is shown in Figure 7.8 and the torque ripples are shown in Figure 7.9.

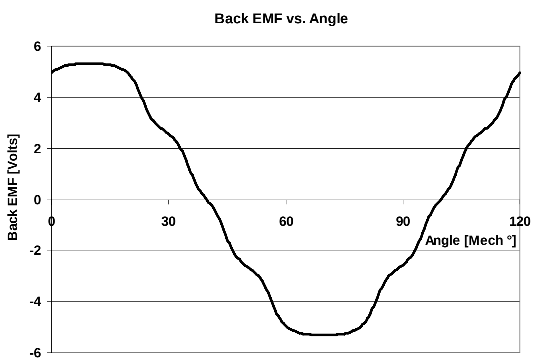
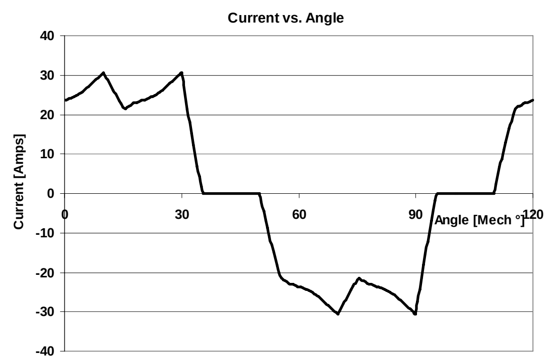


Figure 7.7: Back emf for phase A



Figur 7.8: Current for Phase A

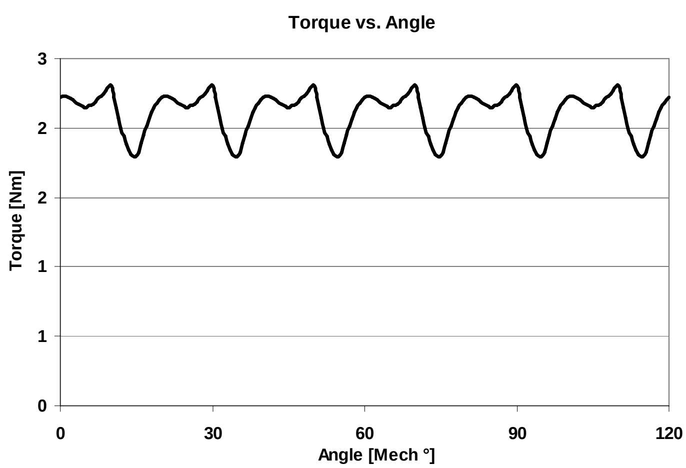


Figure 7.9: Steady State Torque produced by the motor

In the second example the phase resistance of the motor is $R_{ph} = 59.5 m\Omega$ and the phase inductance is $L_{ph} = 0.43 mH$. The number of poles in the motor are $N_p = 6$, the magnitude of the back emf (K_e) is $K_e = 0.0303 Vs/rad$ and the mechanical speed of the motor is 1000 revolutions per minute. The battery voltage is $V_{dc} = 10V$. The shape of the back emf for phase A is shown in Figure 7.10. The steady state solution of the current for 120° conduction mode is shown in Figure 7.11 and the torque ripples are shown in Figure 7.12.

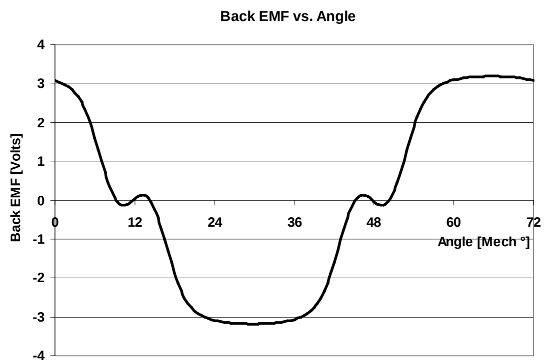


Figure 7.10: Back emf for phase A for the second motor for firing angle $\phi = 30°$

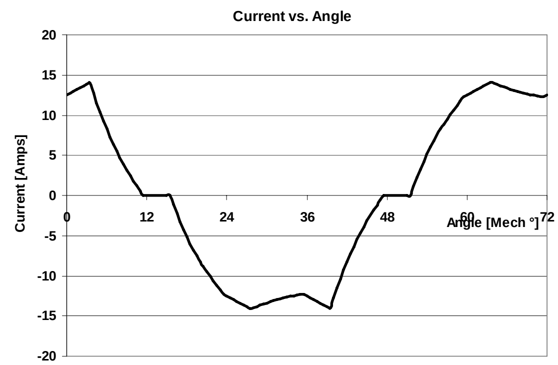


Figure 7.11: Current for Phase A for the second for firing angle $\phi = 30°$

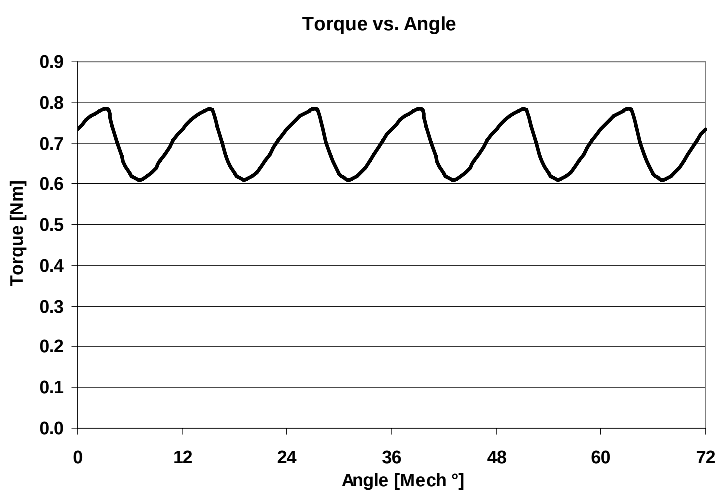
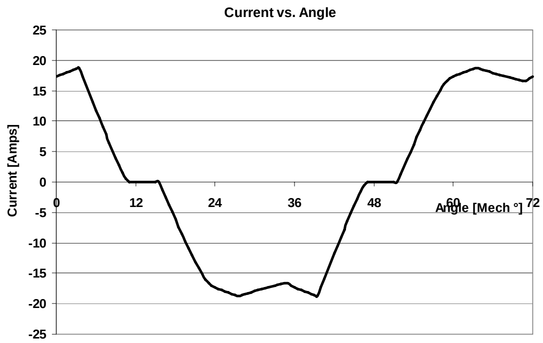


Figure 7.12: Steady State Torque produced by the second motor for firing angle $\phi = 30°$



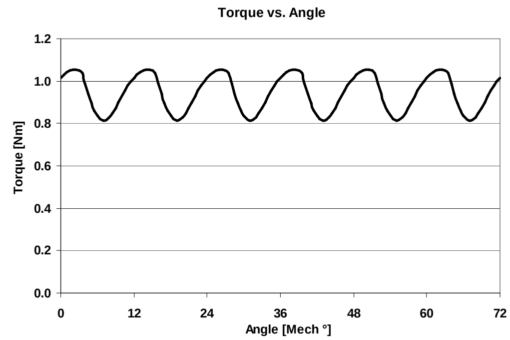
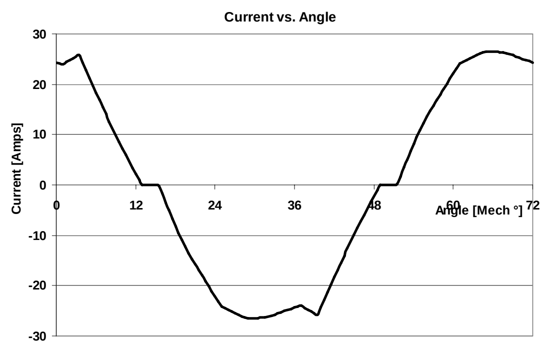


Figure 7.13: Current for Phase A for the second motor for firing angle ϕ =60°

Figure 7.14: Torque Ripple for the second motor for firing angle ϕ =60°



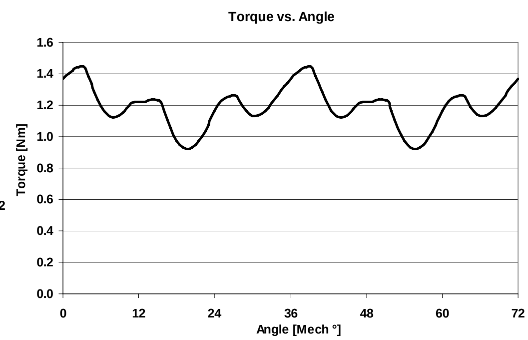
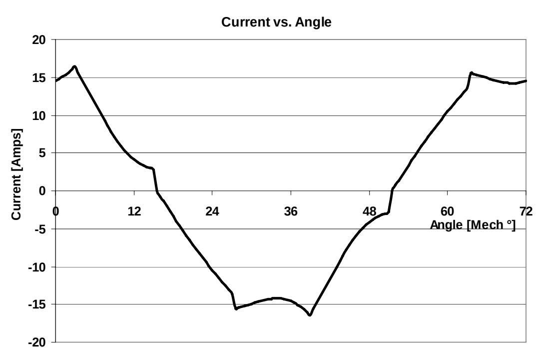


Figure 7.15: Current for Phase A for the second motor firing angle ϕ =90°

Figure 7.16: Torque Ripple for the second motor for firing angle ϕ =90°



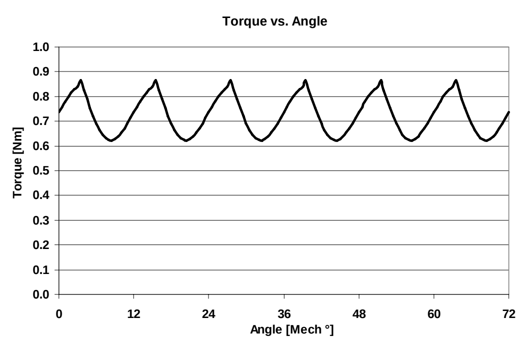


Figure 7.17: Current for Phase A for the second motor for firing angle ϕ =0°

Figure 7.18: Torque Ripple for second motor for firing angle ϕ =0°

In case of 120° mode of inverter operation the firing angle (ϕ) has influence on the motor operation. The influence of the firing angle (ϕ=60°) on the current and the torque ripple, for the motor data given above, are shown in Figure 7.13 and Figure 7.14 respectively.

Similarly the current and torque ripple for firing angle (ϕ=90°) are shown in Figure 7.15 and Figure 7.16 respectively and for ϕ=0° are in Figure 7.17 and Figure 7.18. As can be seen from the figures, the firing angle has influence on the current shape and magnitude and hence the torque ripples. The impact of firing angle on torque ripple is summarised in Table 7.1.

Table 7.1: Torque Ripple for different firing angles

Firing Angle [ϕ]	Max. Torque [Nm]	Min. Torque [Nm]	Torque Ripple [%]
0	0.864	0.623	27.893
30	0.783	0.611	22.015
60	1.053	0.814	22.690
90	1.445	0.922	36.232

7.7 Switching Losses in the MOSFET Switches

So far an analytical model of the VSI was considered without taking into consideration the switching frequency. In reality PWM techniques are employed when controlling a permanent magnet motor. While designing and optimising a permanent magnet motor it is important to consider the switching frequency of the as well. The switching causes losses in the MOSFET and in the current work are taken as an objective function. In this section a model of the switching losses of the MOSFET is discussed, however in the optimization process the conduction losses of the MOSFET are also considered. A crude estimation of the MOSFET switching losses can be calculated using simplified linear approximations of the gate drive current, drain current and drain voltage waveforms during periods 2 and 3 (Figure 7.19) of the switching transition [4]. First the gate drive currents must be determined for the second and third time intervals respectively:

$$I_{G2} = \frac{V_{drv} - 0.5\left(V_{GS,Miller} + V_{TH}\right)}{R_{HI} + R_{Gate} + R_{GI}} \tag{7.44}$$

$$I_{G3} = \frac{V_{drv} - V_{GS,Miller}}{R_{HI} + R_{Gate} + R_{GI}} \tag{7.45}$$

Assuming that I_{G2} charges the input capacitor of the device from V_{th} to $V_{GS,Miller}$ and I_{G3} is the discharge current of the C_{Rss} capacitor while the drain voltage changes from $V_{DS(off)}$ to 0V, the approximate switching times are given as:

$$t_2 = C_{ISS} \frac{V_{GS,Miller} - V_{TH}}{I_{G2}} \tag{7.46}$$

$$t_2 = C_{RSS} \frac{V_{DS,off}}{I_{G3}} \tag{7.47}$$

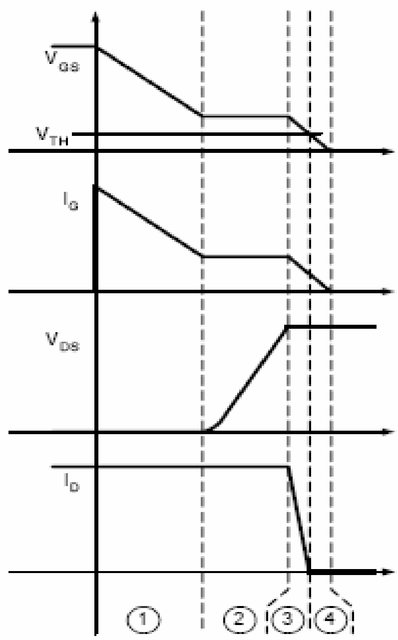


Figure.7.19: Typical switching time intervals of a MOSFET [4]

During t_2 the drain voltage is $V_{DS(off)}$ and the current is ramping from 0 A to load current, I_L while in t_3 time interval the drain voltage is falling from $V_{DS(off)}$ to near =V. Again using linear approximations the waveforms, the power loss components for the respective time interval can be estimated:

$$P_2 = \frac{t_2}{T} V_{DS,off} \frac{I_L}{2} \tag{7.48}$$

$$P_3 = \frac{t_3}{T} V_{DS,off} \frac{I_L}{2} \tag{7.49}$$

Where T is the switching period. The total switching loss is the sum of the two loss components, which yields the following simplified expression:

$$P_{SW} = \frac{V_{DS,off} I_L}{2} \frac{t_2 + t_3}{T} \tag{7.50}$$

7.8 Conclusion

In this work an analytical mathematical model for determining the steady-state performance of a BLDC motor is presented. The presented model is valid for both 120° and 180° conduction modes of voltage source inverters and for any shape of back emf. The results obtained by the analytical model are compared with the measurement results and the current obtained by the analytical model is in good agreement with the measurement results. The advantage of analytical model is that it provides a fast means for evaluating the impact of the system parameters and firing angle on the performance of the BLDC motors.

After having the developed the models necessary models of the BLDC motor drive, a case study (using the example of BLDC motor drive) demonstrating the influence of the system boundaries on the results of optimisation is presented in next chapter.

7.9 References

[1] R. R. Nucera, "Computation of Steady-State Performance of an Electronically Commutated Motor," IEEE Transactions on Industrial Application, vol. 25, pp. 1110-1117, 1989.

[2] G. Huth, "Grenzkennlinien von Drehstrom-servoantrieben in Blockstromtechnik," ETZ Archive, vol. 11, pp. 401-408, 1989.

[3] Z. O. Zhu, "Anayltical Prediction of Dynamic Performance Characteristics of Brushless DC Drives," Electric Machines and Power Systems, pp. 661-678, 1992.

[4] L. Balogh, "Design and Applition Guide for High Speed MOSFET Gate Drive Circuits," Texas Instruments.

Chapter 8

Importance of System Boundaries in Results of Optimisation

8.1 Introduction

In this chapter a case study is presented to demonstrate the importance of proper system boundaries before proceeding to optimise the system. The definition of the clear system boundaries helps in the process of approximating the real system. Since an engineering system consists of many subsystems it may be necessary to expand the system boundaries to include those subsystems that have a strong influence on the operation of the system that is to be designed. To demonstrate this fact the optimisation of a Permanent Magnet Brushless Direct Current (BLDC) Motor is considered. In this chapter the PDM is not applied to design to BLDC motor drive as the main purpose of the chapter is to demonstrate the effect of system boundary definition on the results of optimisation. The formal (mathematical) deterministic and a low fidelity magnetic model of the BLDC motor given in Appendix D is used.

Many researchers have made efforts to improve motor performance in terms of efficiency, maximum torque, back EMF, power/ weight ratio, and minimum losses in iron, coils, friction, and windage. A scheme for optimisation of a three phase electric motor based on genetic algorithms (GA) was presented by Bianchi et. al. [1]. As a demonstration of this technique the authors took a surface mounted permanent magnet motor as an example and applied genetic algorithm to minimise the permanent magnet weight. Similarly an optimal design of Interior Permanent Magnet Synchronous Motor using genetic algorithms was performed by Sim et. al [2]. In this case the efficiency of the motor was taken as the objective function. Usually there are many conflicting design objectives in the optimal design of electrical machines. So multiobjective optimisation (MOOP) technique is required to meet design purposes. The presence of several conflicting objectives is typical for engineering design problems. In many cases where optimisation techniques are utilised, the multiple objectives are aggregated into one single objective function. Optimisation is then conducted with one optimal design. Another approach to handle multiobjective design problems is to employ the concept of Pareto optimality. Pareto optimality was introduced in the late eighteen hundreds by the economist Vilfredo Pareto, and is defined as follows: A solution is said to be Pareto optimal if there no other solution exists that is better in all attributes. This implies that in order to achieve a better value in one objective at least one of the other objectives is going to deteriorate if the solution is Pareto optimal. Thus, the outcome of a Pareto optimisation is not one optimal point, but a set of Pareto optimal solutions that visualises the trade-off between the objectives. In recent years research has been pursued in the area of multiobjective optimisation of permanent magnet (PM) motors. Multiobjective optimisation of PM motor using genetic algorithms was performed by Yamada et. al [3]. A surface mounted PM synchronous motor was taken for optimisation and -constraint method was used to obtain the solution. The objective functions that were considered for optimisation were motor weight and material cost. The authors used a two step method for optimisation. First a preliminary design was carried out in which the design is formulated as a constraint non-linear programming problem by using space harmonic analysis. Then the motor configuration was optimised using a procedure that combined the finite element method (FEM) with the optimisation algorithm. Sim et. al [4] implemented multiobjective optimisation for a permanent magnet motor design using a modified genetic algorithm. The genetic algorithm used in this case was adjusted to the vector optimisation problem. Multiobjective optimisation of an interior permanent magnet synchronous motor was carried out again by Sim et.al [5]. In both cases the authors chose weight of the motor and the loss as objective functions. In the present work the MOOP of PM motors is taken a step further. The optimisation of the motor so far laid focus mainly on the magnetic circuit of the motor. Here the power supply, namely a H- bridge voltage source inverter, along with the magnetic model of the motor is included. The advantage of this procedure is that it always ensures that the optimised motor will deliver the required torque under steady state operation. The results obtained are discussed in section 8.2. Finally conclusions are drawn in section 8.3.

8.2 Results of Optimisation

Many unknown parameters are involved in the design of a BLDC. Therefore it is necessary to fix some of the parameters and then determine the others by optimisation. Table 8.1 below describes the parameters involved in the design process.

Table 8.1: List of variables used in the present work

Symbol	**Description**
T_{motor}	Power or rated torque
N_r	Rated Speed
N_{ph}	Number of phases
g	Air gap
R_{ro}	Outer radius of rotor
K_{cu}	Copper fill factor
ρ	Conductor resistivity
B_r	Reminance field of permanent magnets
μ_r	Relative permeability of permanent magnet
$B_{\max}$	Max. steel flux density
C_h	Hysteresis loss coefficient
C_e	Eddy current loss factor
N_s	Number of slots
N_m	Number of magnets
L_{motor}	Length of the motor
N_{turns}	Number of turns
I_{ph}	Current per phase
α_{dido}	Ratio of inner diameter to outer diameter
α_{mp}	Ratio of pole pitch

For a typical situation, it is usually required to design a motor subject to certain boundary conditions and certain parameters of the motor are to be optimised. In this work a scenario is considered where the mass and the loss of the BLDC motor are to be minimised. Besides that the other restrictions are:

1. the motor produces a torque of 1Nm.
2. the motor is connected to a 3 phase H-bridge voltage source inverter (VSI) with 120° block commutation, i.e. only two switches are conducting at any instant of time (chapter 7).
3. the VSI is connected to a battery of 24 volts.
4. There is no current control.

It is further required that the motor should fit in a certain volume. Hence, it becomes very important in this case to perform optimisation of machine taking into account the power supply. If optimisation of the machine is done purely on the basis of magnetic circuit of the motor then we will reach erroneous result as will be evident from the results of the different cases discussed below.

Case I:

In this case the multi-objective optimisation of the BLDC motor is done without taking into account the power supply. The model of the motor used is given in Appendix D. Here the phase current of the motor (I_{ph}) is a variable that is changed during the optimisation process. The parameters of the motor that are held constant and the parameters that are varied are listed in Table 8.2 and Table 8.3 respectively.

The objectives that are to be optimised in this case are the losses (iron and copper) and the mass of the motor. Mathematically the present optimisation problem can be stated as follows:

$$\text{minimise}\begin{cases} f_1(\vec{x}) = P_{cu} + P_{hys} + P_{eddy} \\ f_2(\vec{x}) = M_{iron} + M_{magnet} + M_{copper} \end{cases} \tag{8.1}$$

where P_{cu}, P_{hys} and P_{eddy} are the copper loss, hysteresis loss in the stator yoke and the eddy current loss in the stator yoke respectively and M_{iron} , M_{magnet} and M_{copper} are the mass of yoke (stator and rotor), mass of permanent magnets and mass of copper respectively respectively.

Table 8.2: Constant Parameters for case 1

Parameter	**Symbol**	**Value**	**Units**
Rated Speed	N_r	1000	Rpm
Reminance field of magnets	B_r	1.2	T
Density of iron	ρ_{Fe}	7700	Kg/m^3
Copper fill factor	K_{cu}	0.5	
Density of magnets	ρ_m	5000	Kg/m^3
Outer radius of the stator	R_o	20	Mm
Resistivity of copper	ρ	$1.68 \cdot 10^{-8}$	Ohm m
Air gap length	g	0.5	Mm

$$\text{subject to } \boldsymbol{h}(\vec{x}) = T_{motor} \geq 1 \text{ Nm} \tag{8.2}$$

where,

$$\vec{x} = (L_{motor},\ \alpha_{mp},\ \alpha_{dido},\ N_m,\ N_s,\ N_{turns}, I_{ph}) \tag{8.3}$$

and

$$\begin{aligned} &1 \leq L_{motor} \leq 100,\ 0.1 \leq \alpha_{mp} \leq 1,\ 0.1 \leq \alpha_{dido} \leq 0.7 \\ &2 \leq N_m \leq 20,\ 3 \leq N_s \leq 30, 1 \leq N_{turns} \leq 100,\ 5 \leq I_{ph} \leq 30 \end{aligned} \tag{8.4}$$

The results obtained are shown in Figure. 8.1.

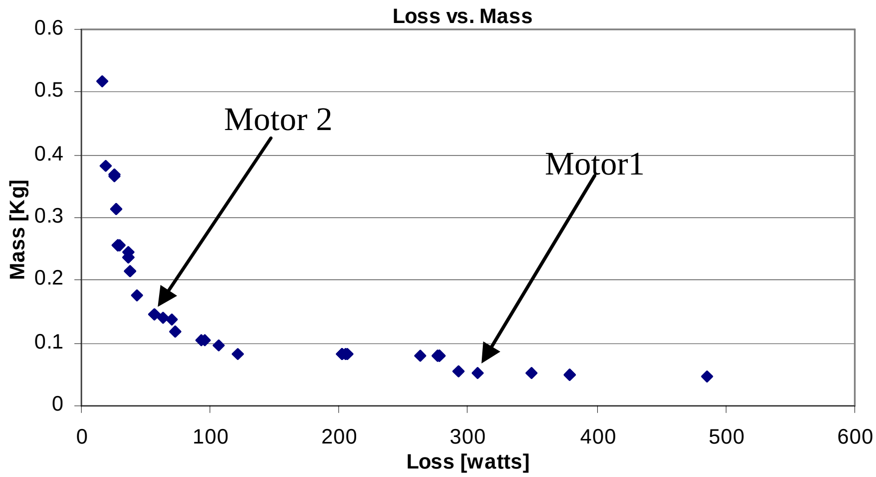


Figure 8.1: Pareto optimal solutions for loss vs. mass for case 1

Table 8.3: Parameters that are varied in Case 1

Parameter	Symbol	Units
Length of the motor	L_{motor}	Mm
Ratio of magnet angle to pole pitch	α_{mp}	
Ratio of inner motor diameter to outer diameter	α_{dido}	
Number of magnets	N_m	
Number of slots	N_s	
Number of turns per coil	N_{turns}	
Phase current	I_{ph}	Amps

In Table 8.4 the parameters of two sample motors from the above Pareto Front are shown. The solutions shown in the above Pareto Front are mathematical optimal solutions and it may be possible that some of the solutions may not be feasible practically. In such case only that section of Pareto Front should be taken into account which is feasible. From the set of Pareto Optimal solutions, shown in Figure 8.1, two s olutions (marked Motor1 and Motor2) are taken and their parameters are given in Table 8.4.

Table 8.4: Values of the motor parameters for the analysis for case 1

Variables	**Motor 1**	**Motor 2**	**Variable**	**Motor 1**	**Motor 2**
N_s	21	30	Mass [Kg]	0.05	0.1407
N_m	14	20	η	0.38	0.75
N_{turns}	22	8	L_{ph} [H]	5E-04	3E-04
I_{ph} [Amps]	20.66	10.83	R_{ph} [Ohm]	0.72	0.53
L_{motor} [mm]	7.62	25.77	T_{avg} [Nm]	1.02	1.16
α_{dido}	0.58	0.66	$V_{\max}$ [Volts]	5.43	11.28
α_{mp}	0.93	0.99	P_{losses} [Watts]	306	63.1

When the motors with parameters listed in Table 4 are fed with a H-bridge voltage source inverter the motors do not produce the required torque. The first motor requires a rms current of 20.66 amps to produce an average torque of 1.02 Nm. The actual current and torque produced by this motor are shown in Figure 8.2a and Figure 8.2b respectively. In these figures the results are from the simulation when the motor and VSI were both taken into account. The rms value of the phase current is 25.5 amps and the average torque produced by the motor is 0.88Nm.

The second motor requires a rms current of 10.83 amps to produce a torque of 1.7 Nm. The actual current and torque produced by this motor is shown in Figure 8.3a and Figure 8.3b respectively. In the figure the results are from the simulation when the motor and VSI were both taken into account. The rms value of the phase current is 17.2 amps and the average torque produced by the motor is 1.7Nm.

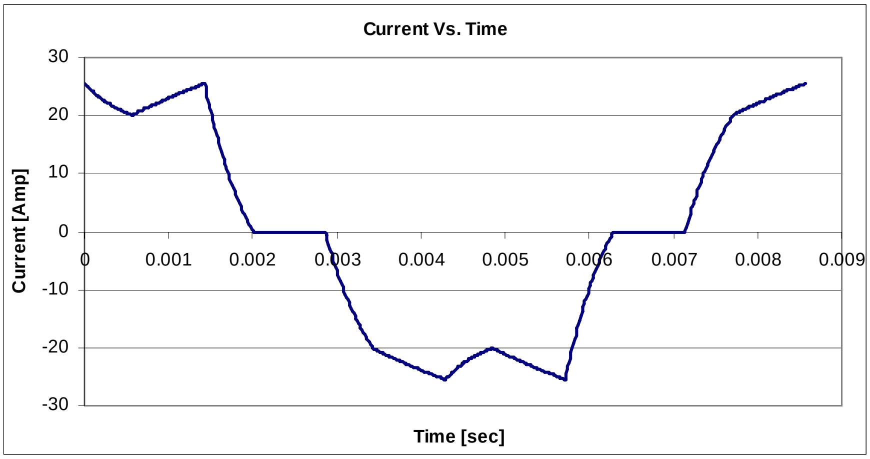


Figure 8.2a: Simulation results of current for phase A for motor 1 in Table 8.4

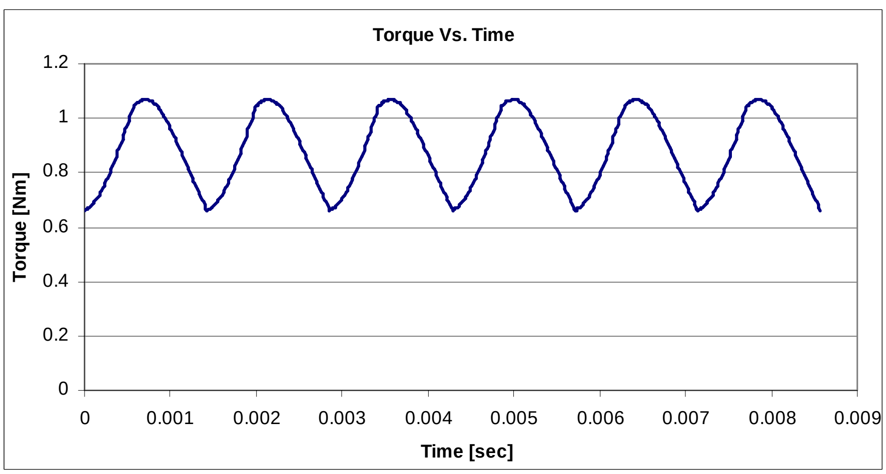


Figure 8.2b: Simulation results of torque for motor 1 in Table 8.4

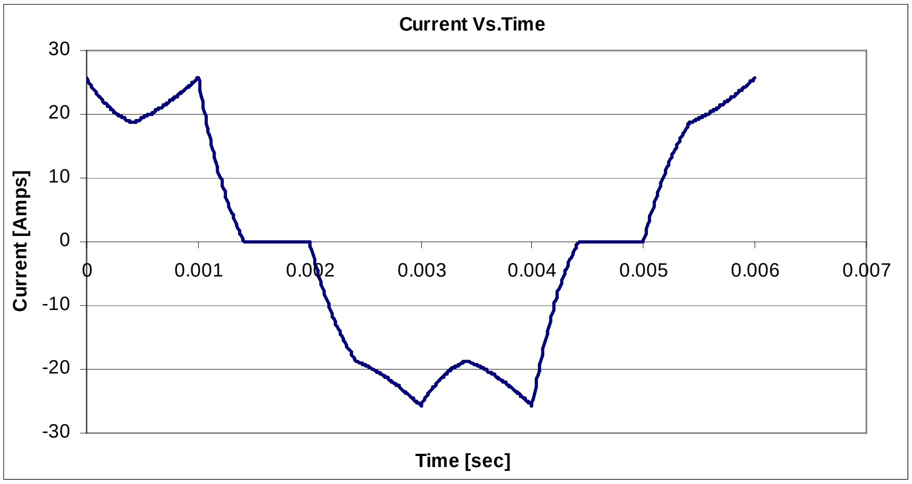


Figure 8.3a: Simulation results of current for phase A for motor 2 in Table 8.4

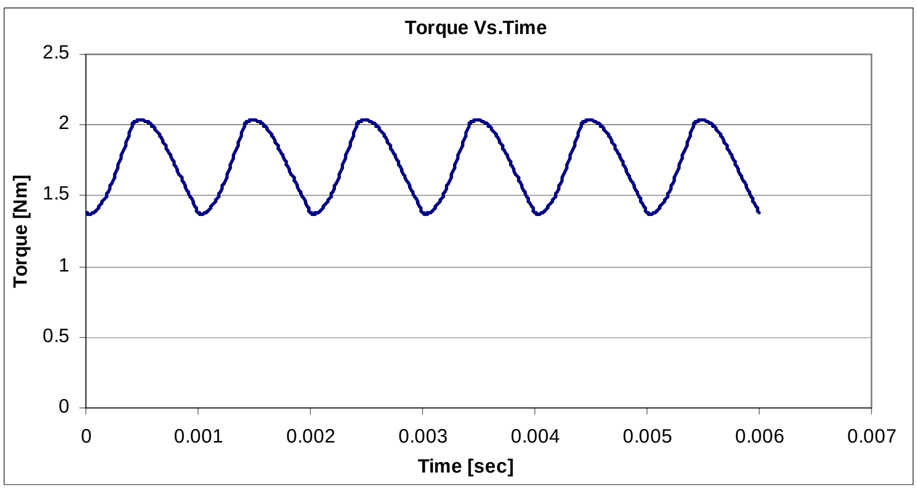


Figure 8.3b: Simulation results of torque for motor 2 in Table 8.4

From the above results it can be seen that the first motor did not produce the required torque and the second motor produced the required torque. Moreover if we look at the dimensions of the motor and the rms values of phase currents then it can be seen that the motor size is small for such high values of current. The motor dimensions are not appropriate in practice for an average torque of 1Nm torque. From these results it can be seen that the motors optimised without taking into account the VSI may result in misleading results. In the next case study, the motor will be design taking into account VSI. In this case the current value will not be a parameter that is varied during the optimisation. The proper values will be determined by the model of the VSI as described in chapter 7.

Case 2

In this case 6 parameters were taken as variables. These parameters are listed in Table 8.5 and the constant parameters are listed in Table 8.2.

Table 8.5: Parameters that are varied for Case 2

Parameter	Symbol	Units
Length of the motor	L_{motor}	mm
Ratio of magnet angle to pole pitch	α_{mp}	
Ratio of inner motor diameter to outer diameter	α_{dido}	
Number of magnets	N_m	
Number of slots	N_s	
Number of turns per coil	N_{turns}	

Here the current is not a variable as in case 1. The current values are determined by the voltage source inverter (VSI) model developed in chapter 7. From the motor model the size of the motor, losses, mass, back emf, inductance and resistances are obtained. The back emf, resistance and inductance values obtained from the motor model are used in the model of VSI to determine the phase current magnitude and waveform. Having obtained the back emf shape and magnitude and current values the torque produced by the motor in the steady state is calculated. Hence, in this case an extended model of the BLDC drive that includes the motor model and the VSI model is used. The objectives that are to be optimised in this case are losses (iron and copper) and the mass of the motor and during the optimisation process the VSI is also taken into account. Mathematically, the present optimisation problem can be written as follows:

$$\text{minimise}\begin{cases} f_1(\vec{x}) = P_{cu} + P_{hys} + P_{eddy} \\ f_2(\vec{x}) = M_{iron} + M_{magnet} + M_{copper} \end{cases} \tag{8.5}$$

$$\text{subject to } \boldsymbol{h}(\vec{x}) = T_{motor} \geq 1\text{ Nm} \tag{8.6}$$

$$\text{where, } \vec{x} = (L_{motor},\ \alpha_{mp},\ \alpha_{dido},\ N_m,\ N_s,\ N_{turns}) \tag{8.7}$$

and

$$1 \leq L_{motor} \leq 100,\ 0.1 \leq \alpha_m \leq 1,\ 0.1 \leq \alpha_d \leq 0.7,\ 2 \leq N_m \leq 20,\ 3 \leq N_s \leq 30,\ 1 \leq N_{turns} \leq 100 \tag{8.8}$$

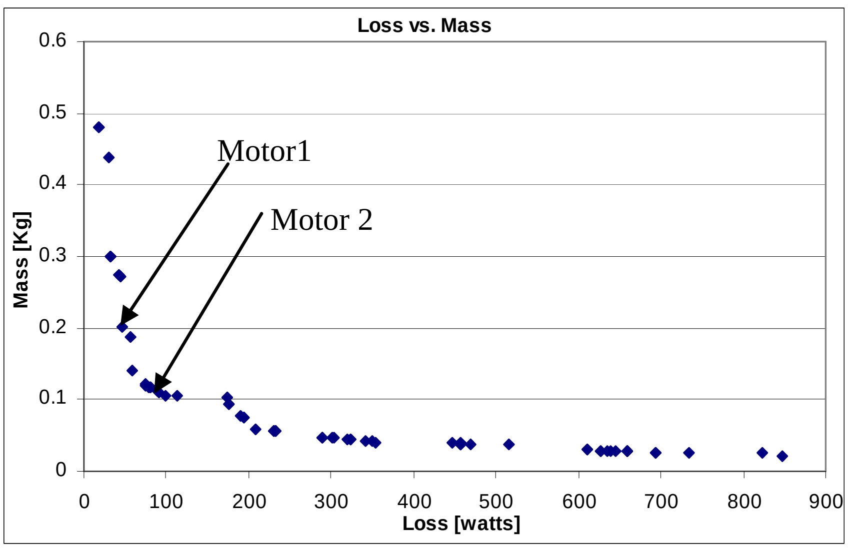


Figure 8.4: Pareto optimal solutions for loss vs. mass for case 2

The results obtained are shown in Figure.8.4. In Table 8.6 the parameters of two sample motors from the above Pareto Front are shown. The solutions shown in the above Pareto fronts are mathematically optimum solutions and it is possible that only a section of Pareto

Front is practically feasible. Only those solutions must be taken into account that are feasible from engineering point of view. The current and torque profiles for the first motor for case 2 are shown in Figure 8.5a and Figure 8.5b respectively. Similarly the current and torque profile for the second motor are shown in Figure 8.6a and Figure 8.6b respectively.

Table 8.6: Values of the motors for analysis for case 2

Variables	Motor 1	Motor 2	Variable	Motor 1	Motor 2
N_s	15	27	Mass [Kg]	0.206	0.105
N_m	10	18	η	0.72	0.59
N_{turns}	12	11	L_{ph} [H]	5.9E-04	3.74E-04
L_{motor} [mm]	47.72	21.1	R_{ph} [Ohm]	1.02	0.98
α_{dido}	0.69	0.7	T_{avg} [Nm]	1	1.09
α_{mp}	1	0.99	$V_{\max}$ [Volts]	16.8	12.3
P_{losses} [Watts]	68	98			

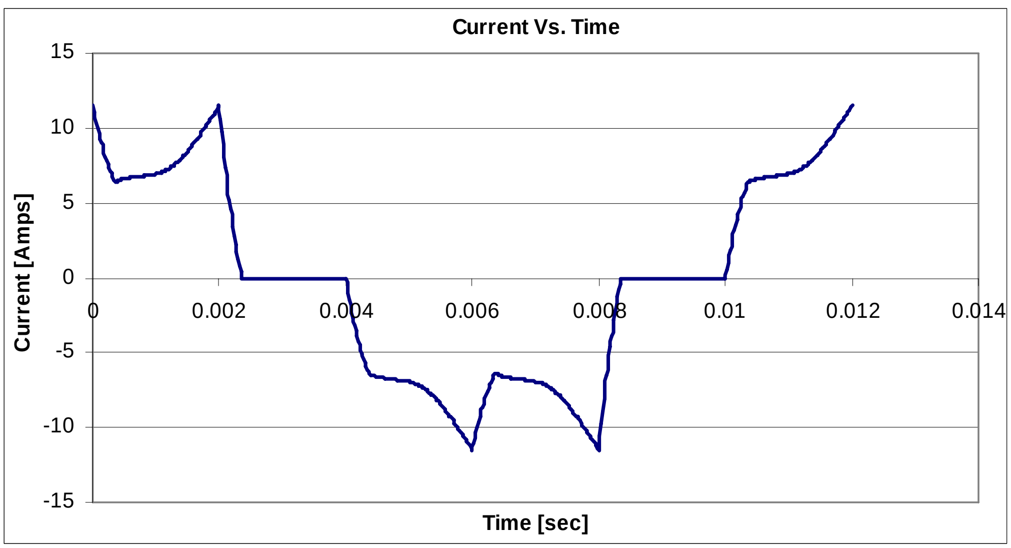


Figure 8.5a: Simulation results of current for phase A for motor 1 in Table 8.6

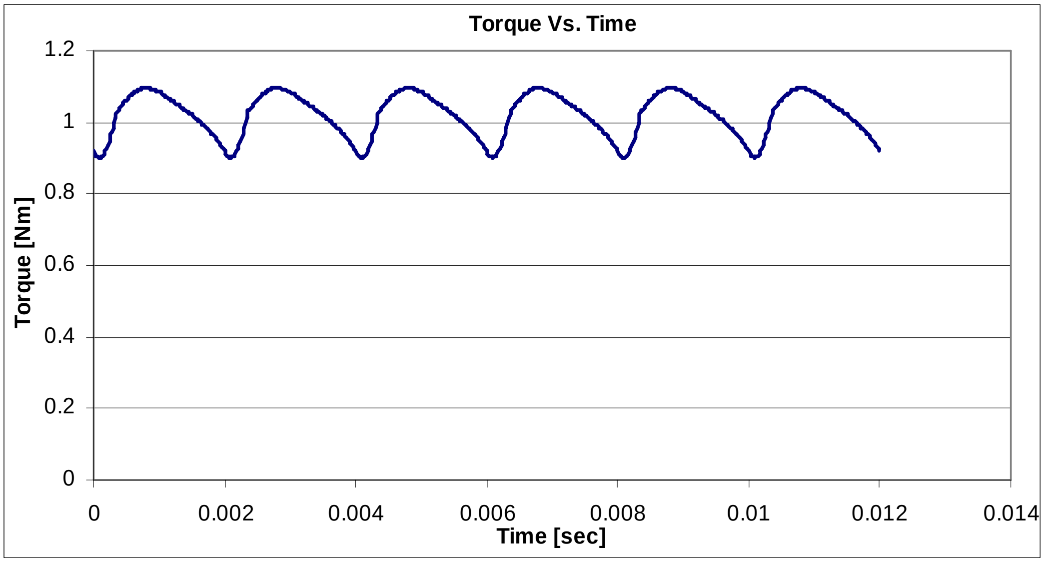


Figure 8.5b: Simulation results of torque for motor 1 in Table 8.6

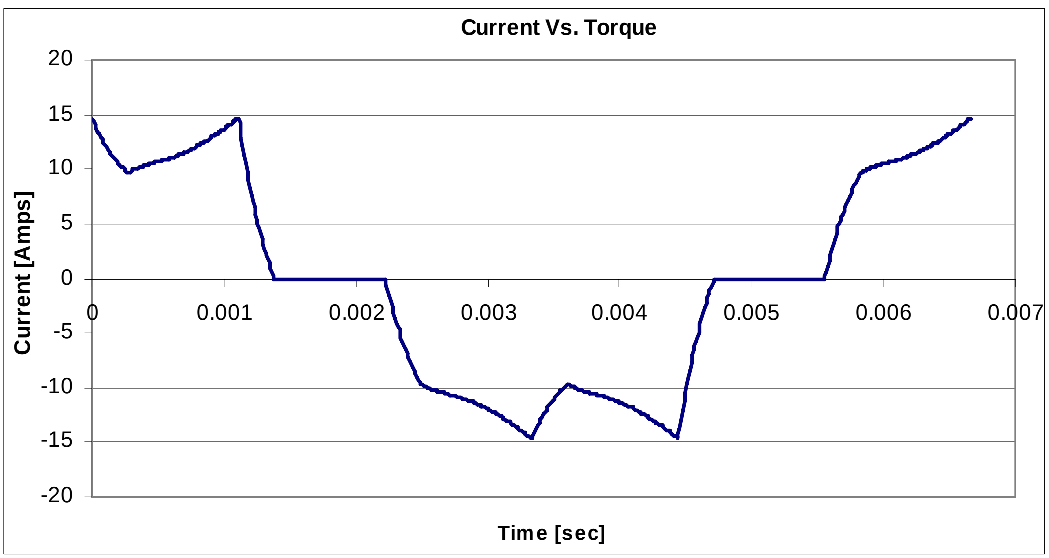


Figure 8.6a: Simulation results of current for phase A for motor 2 in Table 8.6

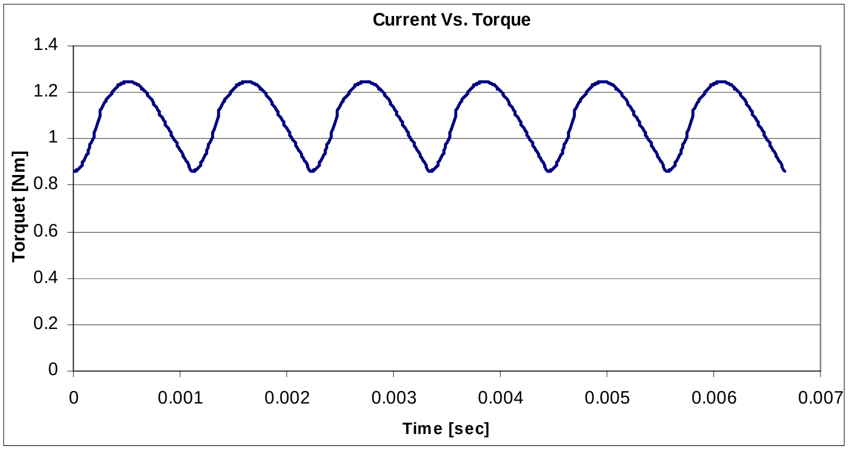


Figure 8.6b: Simulation results of torque for motor 2 in Table 8.6

From the above figures it is seen that both the motors meet the torque constraints. Thus, when the optimisation of the motor is done together with the VSI we reach more realistic solutions. Moreover, the dimensions of the motor obtained in this case are appropriate for the required torque. Hence, in order to achieve proper dimensions of the motor it is important to consider the VSI during the optimisation of the motor. This will result in motor configurations that are realistic and the motors will neither be under dimensioned nor over dimensioned. If the motor is designed without taking into account the VSI, the motor may be either under dimensioned or over dimensioned as seen from case study 1.

8.3 Conclusions

In this chapter the utility of genetic algorithms for multi-objective optimisation is shown. Besides, a very important issue on the modelling itself has been discussed: i.e. to achieve proper optimisation the system boundaries should be identified and clearly defined. The definition of the clear system boundaries helps in the process of approximating the real system. Since an engineering system consists of many subsystems it may be necessary to expand the system boundaries to include those subsystems that have a strong influence on the operation of the system that is to be optimised. To demonstrate this, an example of optimisation of BLDC motor drive using simple analytical models was considered. It was shown that if a BLDC motor is optimised without taking into account the voltage source inverter (VSI) the results might be wrong. Hence, it is necessary to expand the boundary of the system to include the VSI while optimising the BLDC motor. An extended model of the BLDC drive, including the magnetic model of the motor and the VSI model, is closer to reality. The optimisation of the motor based on this extended model resulted in motor designs that were more realistic for the amount of average torque delivered. Hence, in case of design of BLDC motor a suitable model should include the VSI as well. Using this conclusion in the next chapter the complete PDM is applied to the design of a BLDC motor drive.

8.4 References

[1] N. Bianchi and S. Bolognani, "Design optimisation of electric motors by genetic algorithms," IEEE Proceedings on Power Applications, vol. 145, pp. 475-483, 1998.

[2] D. J. Sim, D. H. Cho, J. S. Chun, H. K. Jung, and T. K. Chung, "Efficiency Optimisation of Interior Permanent Magnet Synchronous Motor Using Genetic Algorithms," IEEE Transactions on Magnetics, vol. 33, pp. 1880-1883, 1997.

[3] E. Yamada, T. Higuchi, J. Oyama, E. Chiricozzi, F. Parasiliti, and M. Villani, "Optimisation Procedure of Surface PM Synchronous Motors," IEEE Transactions on Magnetics, vol. 33, pp. 1943-1946, 1997.

[4] D. J. Sim, H. K. Jung, S. Y. Hahn, and J. S. Won, "Application of Vector Optimisation Employing Modified Genetic Algorithm to Permanent Magnet Motor Design," IEEE Transactions on Magnetics, vol. 33, pp. 1888-1891, 1997.

[5] D. J. Sim, D. H. Cho, and H. K. Jung, "Multiobjective Design of Interior Permanent Magnet Synchronous Motors Considering Improved Core Loss Formula," IEEE Transactions on Energy Conversion, vol. 14, pp. 1347-1352, 1999.

Chapter 9

Application of PDM to Design a BLDC Motor

9.1 Introduction

In this chapter the PDM is applied for the design of a BLDC motor for a specific application. All the steps of PDM are applied and the motor is designed that is optimal with respect to the system in which it has to work. In section 9.2 the application of Synthesis phase of PDM to design a BLDC motor is presented. Section 9.3 deals with the application of intermediate analysis phase of PDM to the design of BLDC motors. The application of final analysis phase of PDM to the design of BLDC motor is given in section 9.4. Finally the conclusions are drawn in section 9.5.

9.2 Synthesis Phase of Progressive Design Methodology for the Design of a BLDC motor

In the following subsections the steps of the Synthesis phase of the PDM are applied to design of a BLDC motor.

9.2.1 System Requirement Analysis

The specified parameters of the motor are:

Rated speed	800 rpm (mechanical)
Torque at speed	0.2 Nm
Number of phases	3

The aim of the problem is to design a motor with a cogging torque of less than 20 milliNm, maximum efficiency, minimum mass and trapezoidal back emf.

Inverter Full bridge	Voltage source inverter
Motor topology	Inner rotor with surface mount magnets
Phase connection	The phases are connected in star

The additional constraints of the motor are:

Outer stator diameter	40 mm
Max. Length	50 mm
Air gap length	0.2 mm
Maximum input voltage	50 Volts

9.2.2 Definition of system boundaries

The BLDC motor to be designed is driven by a voltage source inverter (VSI). The VSI topology used here is a full bridge inverter and MOSFETs are used as switches. Hence, while designing the motor it is important to include the VSI in the system boundaries as the choice of MOSFETs and the motor parameters are not mutually exclusive. This will ensure that the designed motor will produce the required torque when it is integrated with the VSI and will also ensure that proper MOSFETs are selected. The model of the system that includes the BLDC motor and the VSI is more complicated but will give a well designed motor. Hence, the system boundary under consideration in the *synthesis* phase consists of:

1. The BLDC motor (Primary system)
2. Three phase VSI including the MOSFET and switching frequency.

9.2.3 Determining of Performance Criteria

From the requirement analysis the primary objectives that have to be satisfied are:

1. Minimum Cogging Torque
2. Maximum Efficiency
3. Minimum Mass
4. Trapezoidal shape of back EMF

In the *synthesis* phase of PDM only simple model of the BLDC drive is developed. However, determining parameters like cogging torque and shape of the back emf require detailed analytical or FEM models. The mass and efficiency of the motor can be calculated with relative ease compared to the cogging torque and back emf shape. Hence, in the *synthesis* phase the objectives that will be considered are

1. Minimise the mass
2. Maximise the efficiency

A generic topology of BLDC motor with surface mount magnets, as shown in Figure 9.1, is considered. This topology is optimised for minimum mass and maximum efficiency. In the final design the parameters of this optimised generic topology are fine-tuned to reduce the cogging torque and obtain sinusoidal back emf shape.

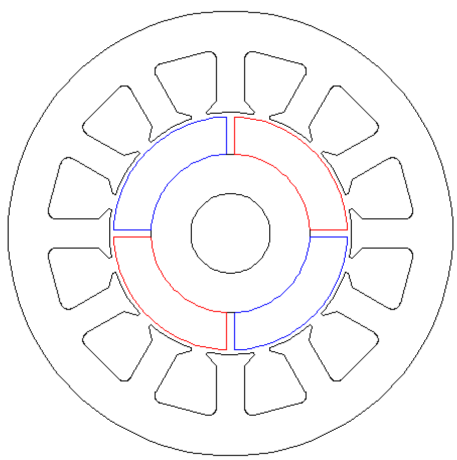


Figure 9.1: Typical lamination of a BLDC motor

9.2.4 Selection of Variables and Sensitivity Analysis

The *independent design variables* that are used in this case are shown in Table 9.1.

The selection of the system variables and sensitivity analysis is done based on the ***formal*** (mathematical) ***deterministic*** and a ***low fidelity*** magnetic model of the BLDC motor as given in Appendix D.

Table 9.1: List of *independent variables* used in the synthesis phase

Variable Name	Symbol	Units
Number of poles	N_p	
Number of slots	N_s	
Length of the motor	L_{motor}	mm
Ratio of inner diameter of motor to outer diameter	α_{dido}	
Ratio of magnet angle to pole pitch	α_{mp}	
Height of the magnet	h_m	mm
Reminance field of the permanent magnets	B_r	T
Max. steel flux density	$B_{\max}$	T
Number of turns in the coils of the motor	N_{turns}	
Switching frequency	F_{sw}	Hz
Input Voltage	V_{dc}	V
Type of MOSFETs	Mos_{typ}	

Sensitivity analysis is performed to determine the influence of the *engineering variables* on the *objectives* viz. mass and efficiency. In Figure 9.2 to 9.5 the sensitivity curves of losses and mass w.r.t. each design variable are shown. From Figure 9.2 it can be seen that as the number of turns in the coil increases the losses in the motor decreases. This is due to the fact that with higher number of turns the induced back emf increases, as a result of this the difference between the input voltage (24 Volts in the present case) and the back emf reduces. This in turn, reduces the magnitude of the phase current and hence the Ohmic losses, proportional to square of the current, reduce. The losses of the motor are also sensitive to length of the motor (length of the motor is same as length of the magnet in the present analysis) and reach a minimum value as the length increases Figure 9.3. The ratio of inner diameter to outer diameter of the stator has an influence on the losses in the motor, Figure 9.4. As the ratio increases the losses reduce because the stator yoke is thicker and as a result of this the field density in the yoke is less resulting in reduction of eddy current and hysteresis losses. As the ratio of magnet angle to pole pitch increases the losses reduce, Figure 9.5. A smaller ratio of magnet angle to pole pitch results in smaller magnet and hence less field density in the iron part,thereby reducing the eddy current and hysteresis losses in the iron parts of the motor. The reminance field of the permanent magnet and maximum allowable field density in iron for linear characteristics have influence on the losses, Figure 9.6 and 9.7 respectively. The height of the magent also influences the losses in the motor, Figure9.8.

The mass of the motor more or less remains constant with the increase in the number of turns of the coil, Figure 9.9. This is due to the fact that the slot fill ratio is kept constant in the present analysis. Hence, the increase in the number of turns does not have an influence on the total mass of copper. The mass is directly proportional to motor length, Figure 9.10. As the ratio of stator inner and outer diameter increases the mass of the motor reduces because the amount of iron in the stator reduces, Figure 9.11. The mass reaches a maximum value as the ratio of the magnet angle to the pole pitch increases, Figure 9.12. The influence of reminance field density of the permanent magnet on the mass of the motor is shown in Figure 9.13. Similarly the influence of maximum allowable field density in iron for linear characteristics on motor mass is shown in Figure 9.14. The height of the magnet has influence on the mass, as shown in Figure 9.15.

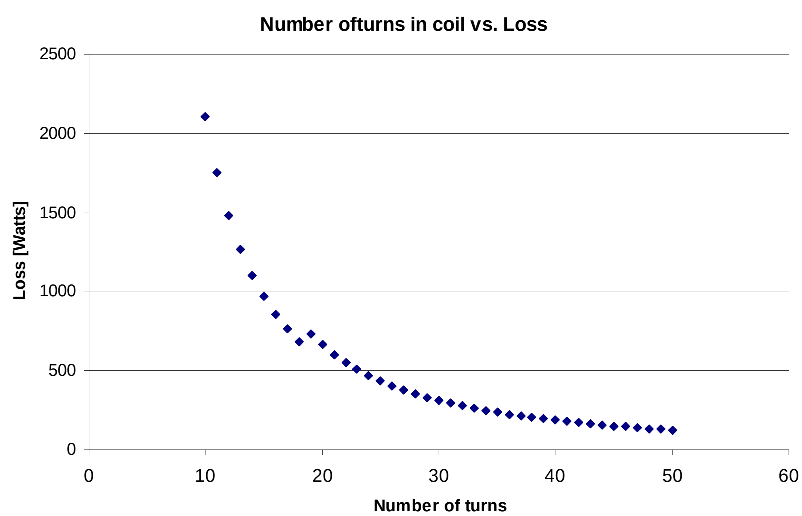


Figure 9.2: Sensitivity of loss w.r.t. number of turns in the coil

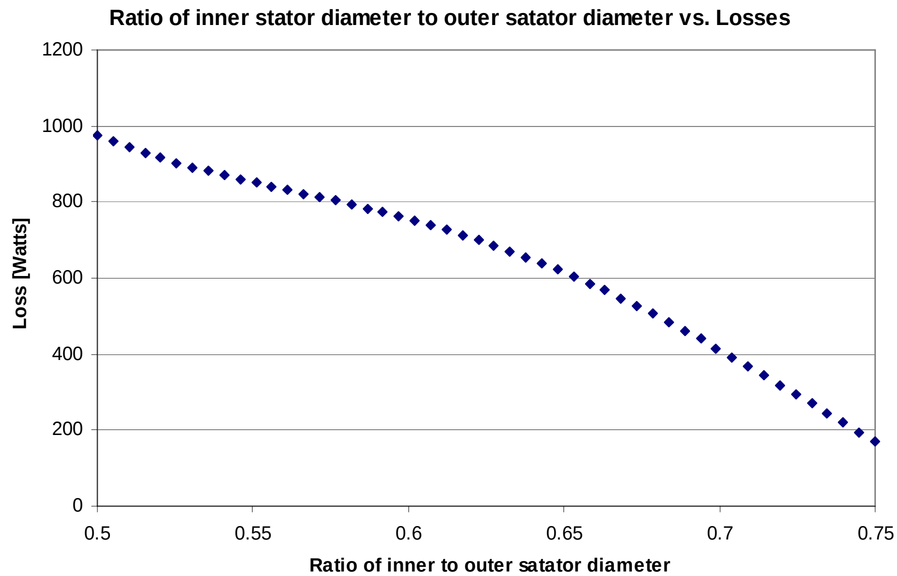


Figure 9.3: Sensitivity of loss w.r.t. length of themotor

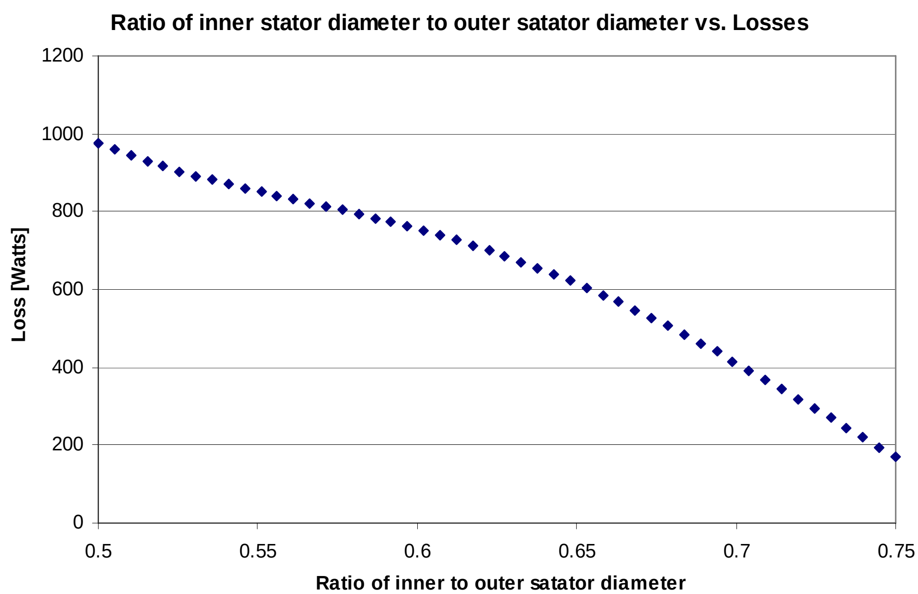


Figure 9.4: Sensitivity of loss w.r.t. ratio of inner to outer stator diameters

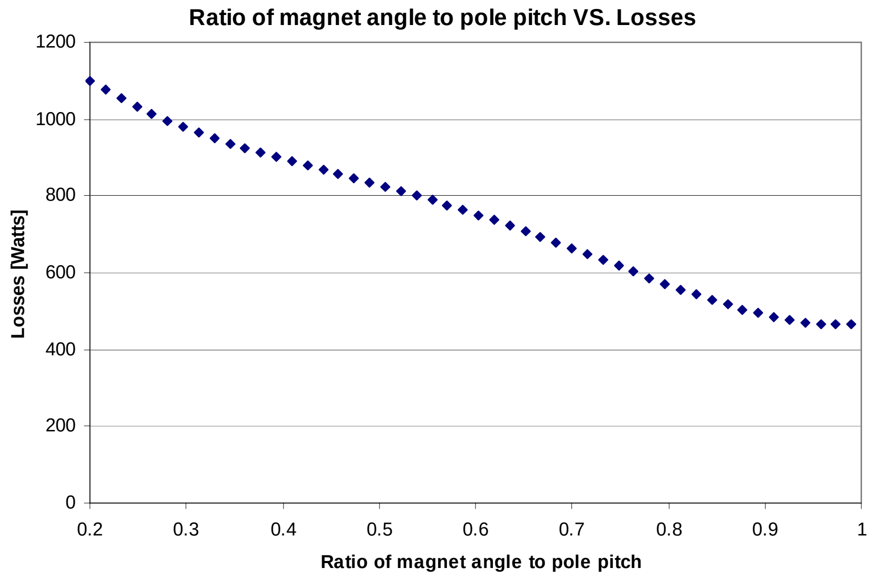


Figure 9.5: Sensitivity of loss w.r.t. ratio of magnet angle to pole pitch

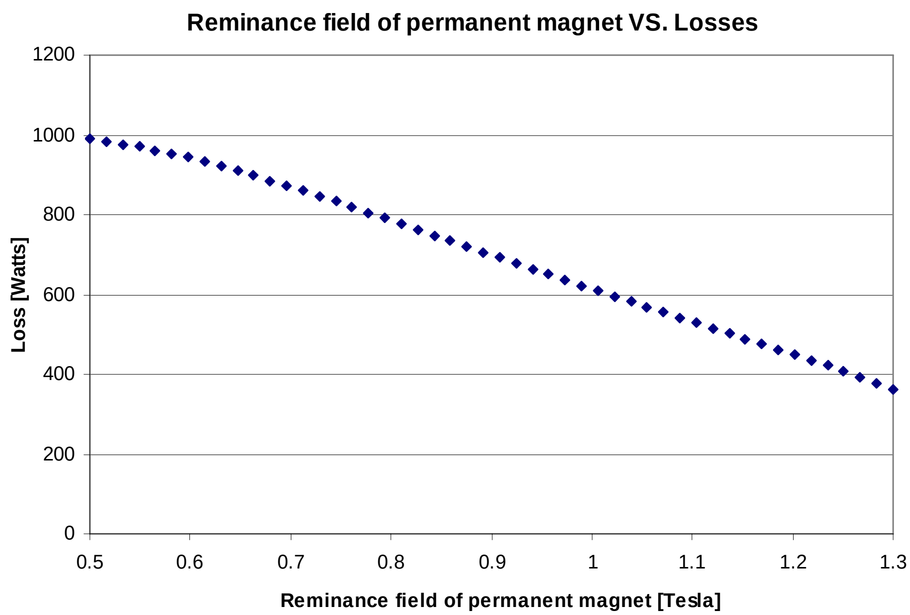


Figure 9.6: Sensitivity of loss w.r.t. reminance field density of permanent magnet

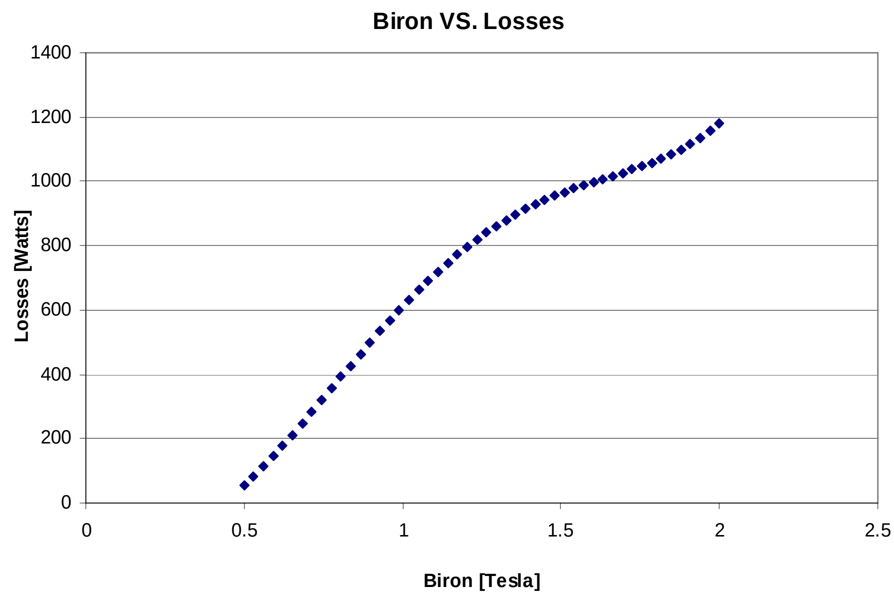


Figure 9.7: Sensitivity of loss w.r.t. maximum allowable field density in iron for linear characteristics

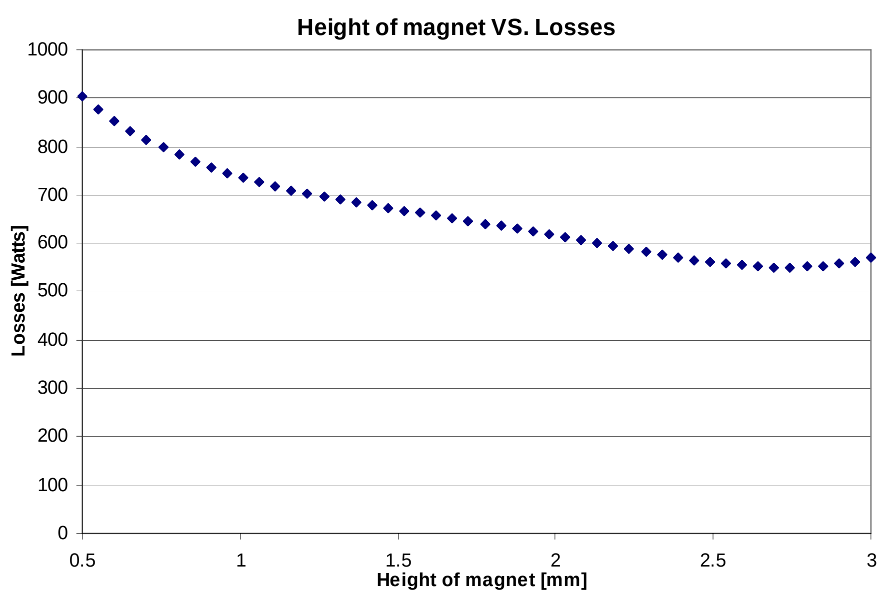


Figure 9.8: Sensitivity of loss w.r.t. height of magnet

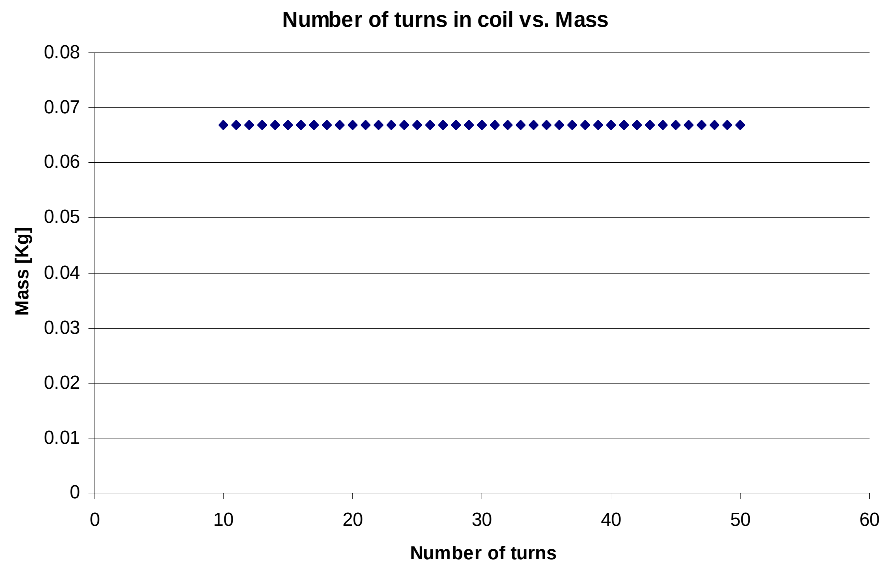


Figure 9.9: Sensitivity of mass w.r.t. number of turns

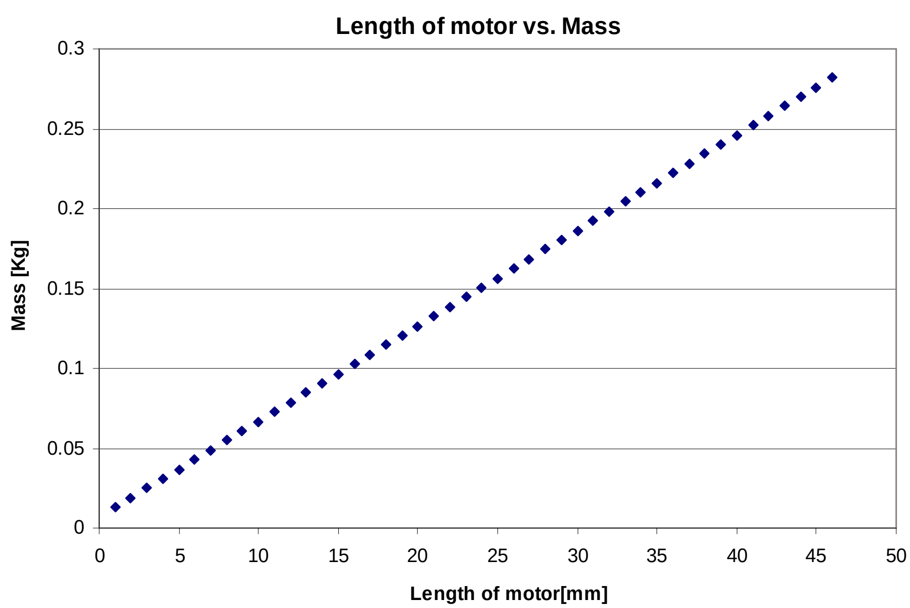


Figure 9.10: Sensitivity of mass w.r.t. motor length

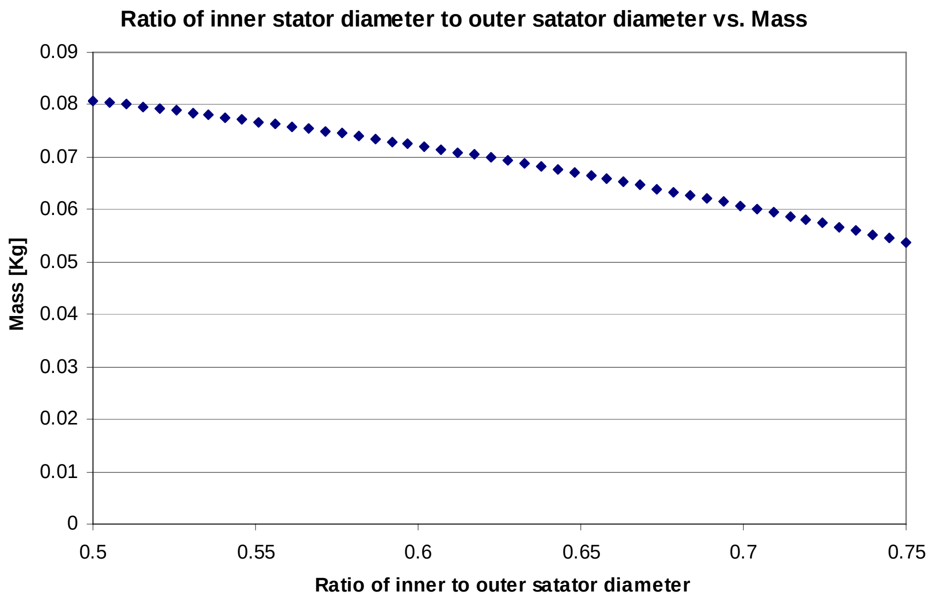


Figure 9.11: Sensitivity of mass w.r.t. ratio of stator inner and outer diameter

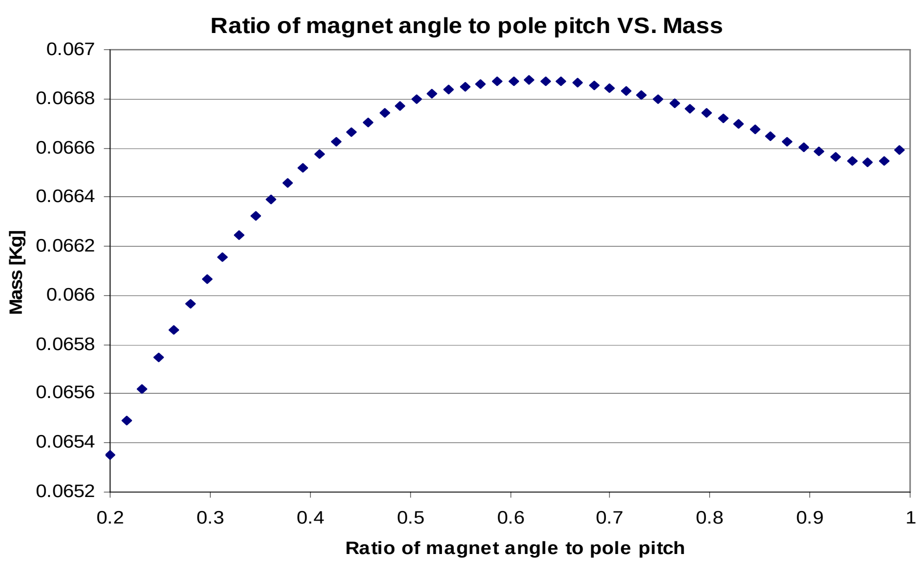


Figure 9.12: Sensitivity of mass w.r.t. ratio of maget to polepitch

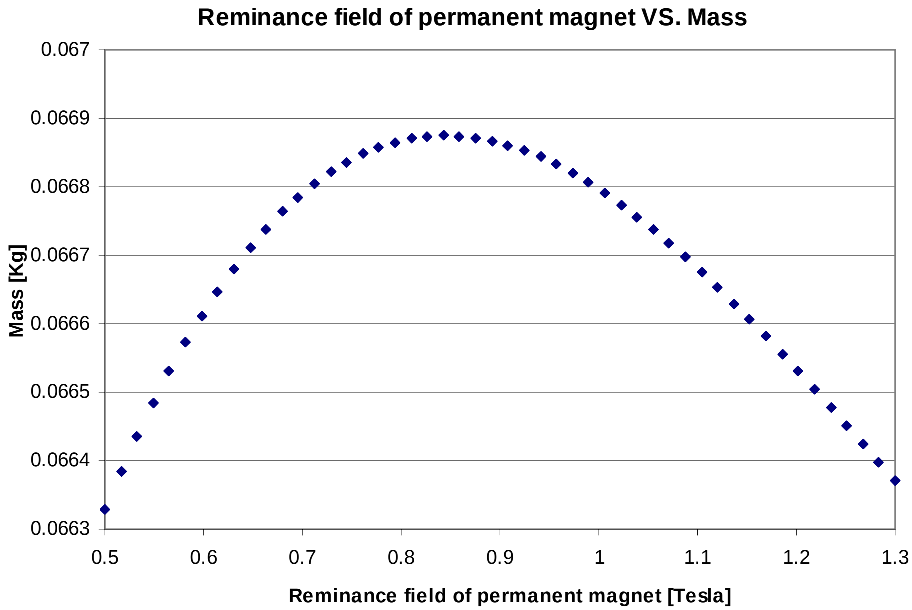


Figure 9.13: Sensitivity of mass w.r.t. reminance field density of permanent magnet

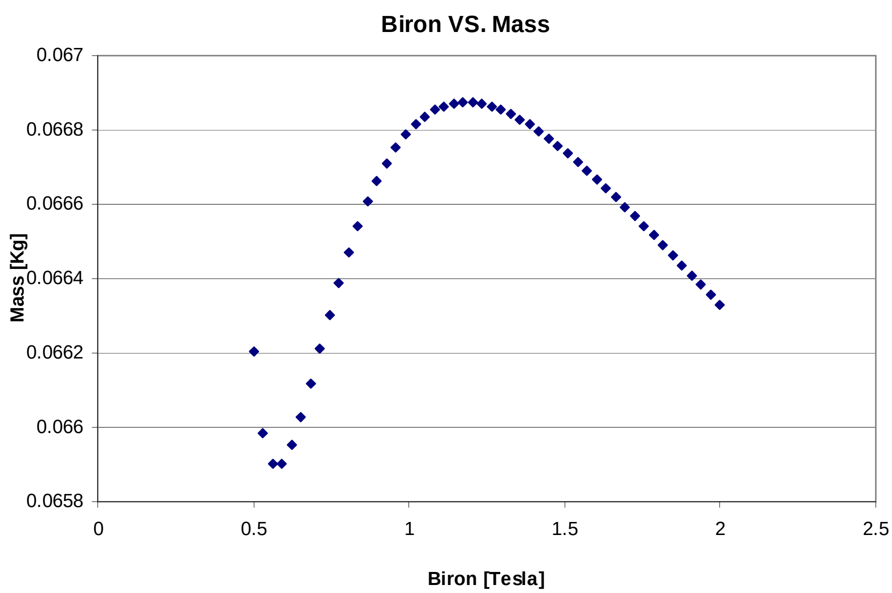


Figure 9.14: Sensitivity of mass w.r.t. maximum allowable field density in iron

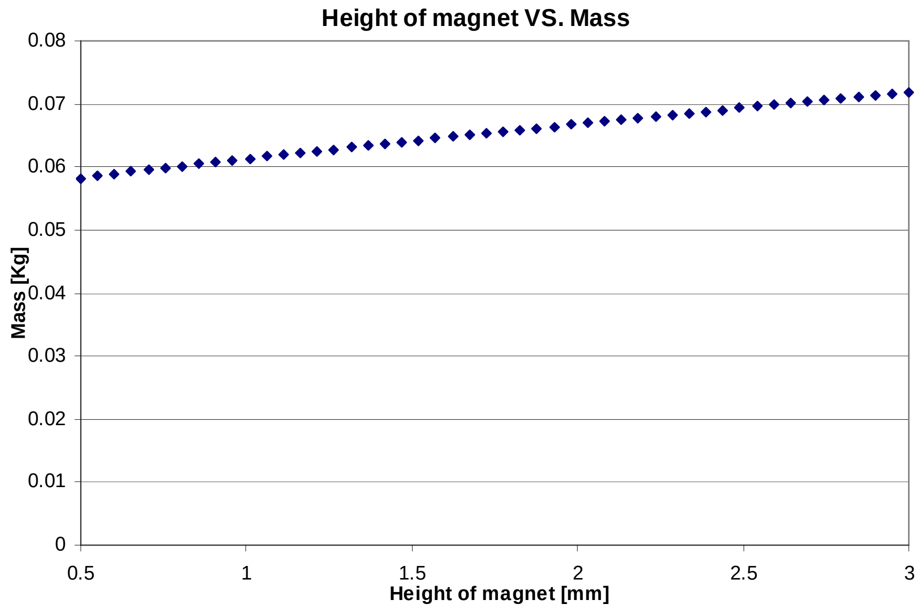


Figure 9.15: Sensitivity of mass w.r.t. height of the magnet

The results of the sensitivity analysis show that the selected *engineering* variables have an influence on the *objectives* (mass and losses) selected for the synthesis analysis. After having performed the sensitivity analysis the next step is to develop the system models. These steps are described in the next subsection.

9.2.5 Development of System Model

The most important aspect of the PDM is to develop appropriate model of the system to be designed. In the following subsections the models of various components of the PM motor drive viz., the motor, the voltage source inverter (VSI) and the MOSFET are given.

9.2.5.1 Motor Model

The motor model is a ***formal*** (mathematical) ***deterministic*** and a ***low fidelity*** model of the BLDC. To develop this model certain assumptions have been made. The assumptions made are:

1. No saturation in iron parts

2. Magnets are symmetrically placed
3. Slots are symmetrically placed
4. Back emf is trapezoidal in shape.
5. Motor has balanced windings
6. Permeability of iron is infinite

The general configuration of the motor is shown in Figure 9.1. The motor design equations are developed for determining the following parameters:

1. Electrical Design
2. General Sizing
3. Inductance and Resistance Calculation

The details of the motor model are given in Appendix D.

9.2.5.2 Dynamic Performance of BLDC Motor:

The derivation of this model is based on the assumption that the induced currents in the rotor due to the stator harmonic fields are neglected. The coupled circuit equations, as discussed in chapter 7, are reproduced here:

$$[V]=[R][i]+[L]\frac{d[i]}{dt}+[e] \tag{9.1a}$$

where $$[V]=\left[V_a,V_b,V_c\right]' \tag{9.1b}$$

$$[R]=\begin{bmatrix} R_{ph} & 0 & 0 \\ 0 & R_{ph} & 0 \\ 0 & 0 & R_{ph} \end{bmatrix} \tag{9.1c}$$

$$[i]=\left[i_a,i_b,i_c\right]' \tag{9.1d}$$

$$[L]=\begin{bmatrix} L_{ph} & 0 & 0 \\ 0 & L_{ph} & 0 \\ 0 & 0 & L_{ph} \end{bmatrix} \tag{9.1e}$$

$$[e]=\left[e_a, e_b, e_c\right]' \tag{9.1f}$$

where R_{ph} and L_{ph} are the phase resistance and phase inductance values respectively defined earlier and V_a, V_b, and V_c are the input voltages to each phase *a, b* and *c* respectively. The induced emf e_a, e_b, e_c are sinusoidal in shape. The electromagnetic torque (T_e) is given by

$$T_e = [e_a i_a + e_b i_b + e_c i_c]\frac{1}{\omega_m} \tag{9.2}$$

where ω_m is the mechanical speed of the motor.

The analytical solution of the eq. (9.1a) is given in chapter 7 of this thesis. The model developed in chapter 7 is used to determine the actual current in motor.

9.2.5.3 Model of losses in MOSFETs:

A simple estimation of the MOSFET switching losses can be calculated using simplified linear approximations of the gate drive current, drain current and drain voltage waveforms. The model for determining the switching losses in the MOSFET is given in chapter 7.

9.2.6 Optimisation Strategy

In the present case study optimisation strategy based on ***Posteriori articulation of preference information*** is used. To achieve the multiobjective optimisation the Nondominated sorting Biologically Motivated Genetic Algorithm (NBGA) [1] is used. The NBGA as been discussed in chapter 4 of this thesis. The parameters of NBGA are as follows

Number of generations = 50

Number of individuals = 100

Crossover probability = 80%

Single point crossover was used.

The mutation rate was fixed between 0 and 10%

Hence, the multiobjective optimisation problem to be solved is expressed mathematically as

$$\text{minimise}\begin{cases} f_1(\vec{x}) = P_{cu} + P_{hys} + P_{eddy} + P_{MOSFET} \\ f_2(\vec{x}) = M_{iron} + M_{magnet} \end{cases}$$

where $P_{cu}, P_{hys}, P_{eddy}$ and P_{MOSFET} are the copper loss, hysteresis loss in the stator yoke ,the eddy current loss in the stator yoke and switching and conduction losses in MOSFETs respectively and M_{iron} and M_{magnet} are the mass of yoke (stator and rotor) and mass of permanent magnets respectively.

Subject to

$$\boldsymbol{h}(\vec{x}) = T_{motor} \geq 0.2 \text{ Nm}$$

Where,

$\vec{x} = (B_r, B_{\max}, h_m, L_{motor}, \alpha_{mp}, \alpha_{dido}, N_m, N_s, N_{turns}, MOS_{typ})$ are *the independent design variables*

and

$$\begin{aligned} &0.5 \leq B_r \leq 1.2,\ 0.5 \leq B_{Fe} \leq 2,\ 1 \leq h_m \leq 3,\ 1 \leq L_{motor} \leq 100 \\ &0.1 \leq \alpha_m \leq 1,\ 0.1 \leq \alpha_{dido} \leq 0.7,\ 2 \leq N_m \leq 10,\ 3 \leq N_s \leq 15 \\ &1 \leq N_{turns} \leq 100, 1 \leq MOS_{typ} \leq 248 \end{aligned}$$

The results of the optimisation are given in the next subsection.

9.2.7 Results of Multiobjective Optimisation

The results of optimisation are given in Figure 9.16 to 9.21. From the results it can be seen that for each pole slot combination a number of Pareto optimal solutions are present and as the mass of the motor increases the losses decrease.

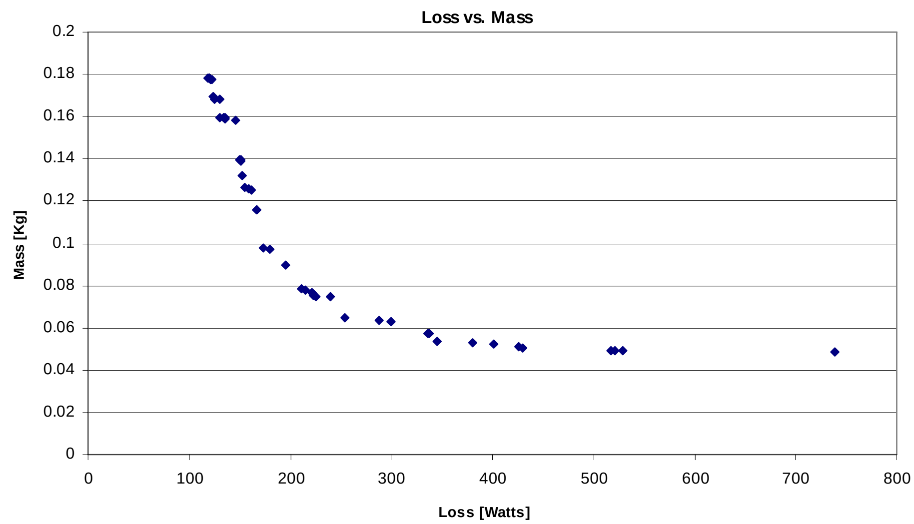


Figure 9.16: Pareto optimal solutions for a BLDC motor with Ns=6 and Np=4

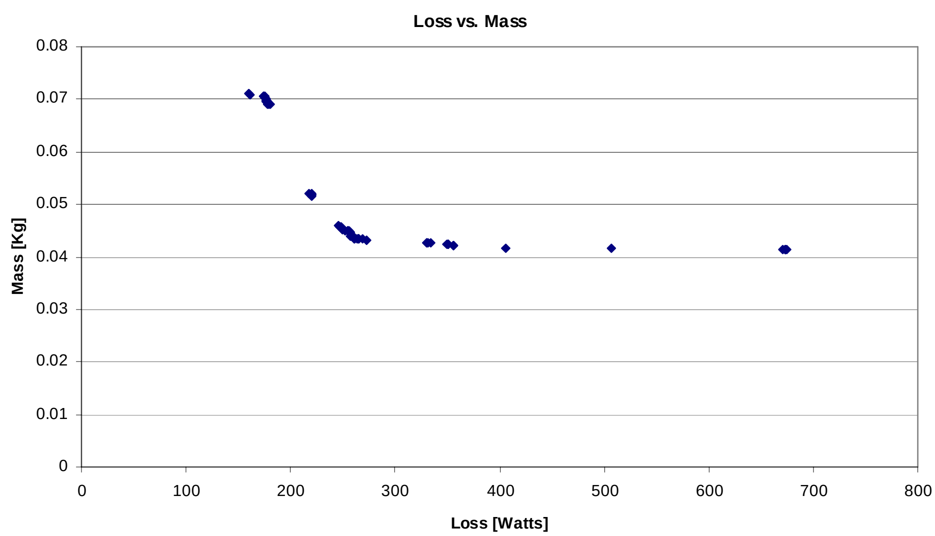


Figure 9.17: Pareto optimal solutions for a BLDC with Ns=9 and Np=6

Since the number of feasible solutions is large the results have to be screened so that a reduced set is obtained. Detailed analysis can then be performed on the reduced set. In the

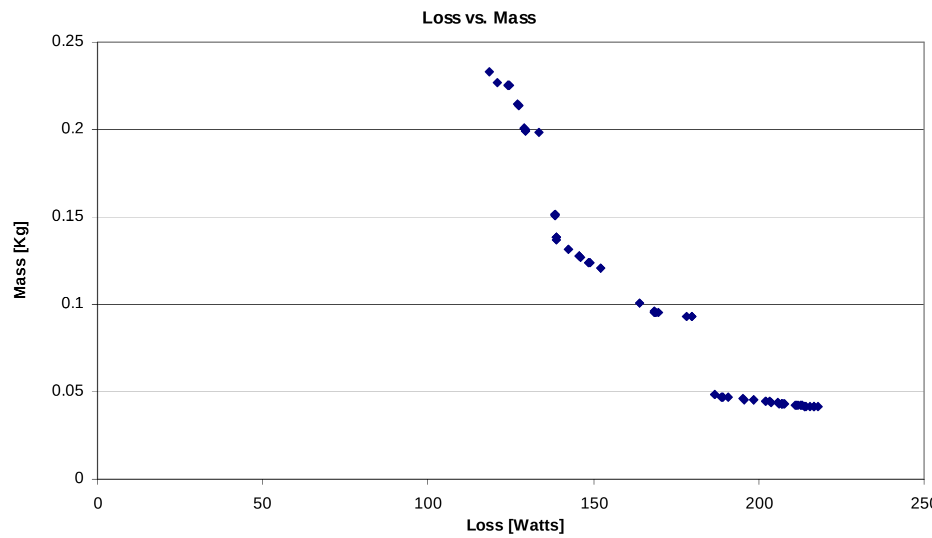


Figure 9.18: Pareto optimal solutions for a BLDC motor with Ns=12 and Np=8

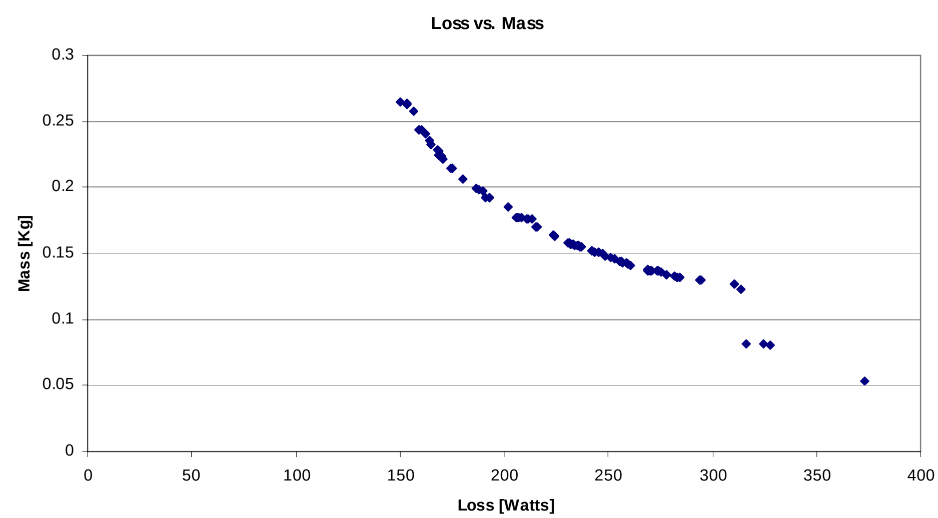


Figure 9.19: Pareto optimal solutions for a BLDC motor with Ns=6 and Np=8

next section the screening process is performed.

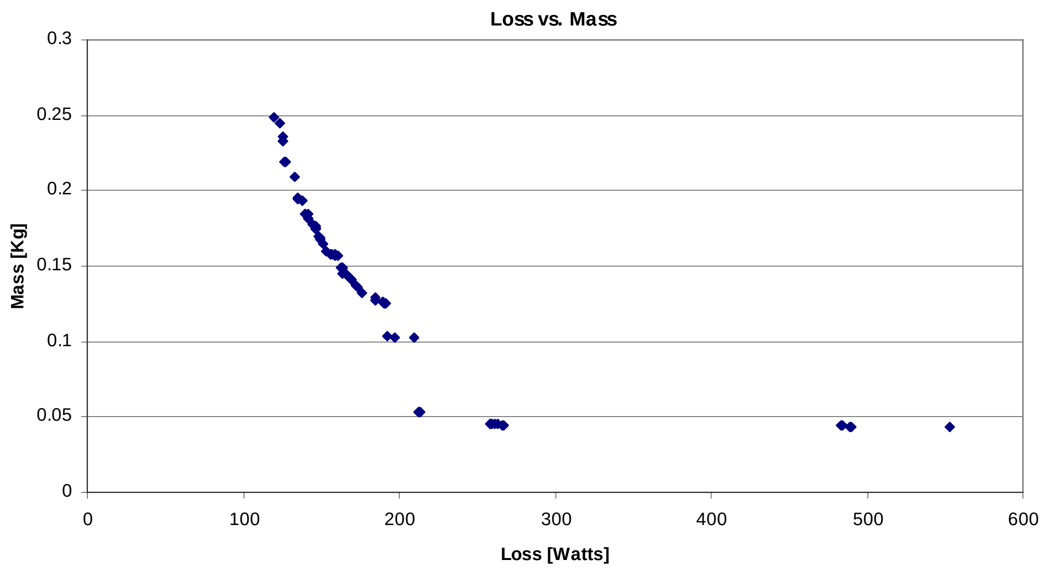


Figure 9.20: Pareto optimal solutions for a BLDC motor with Ns=9 and Np=8

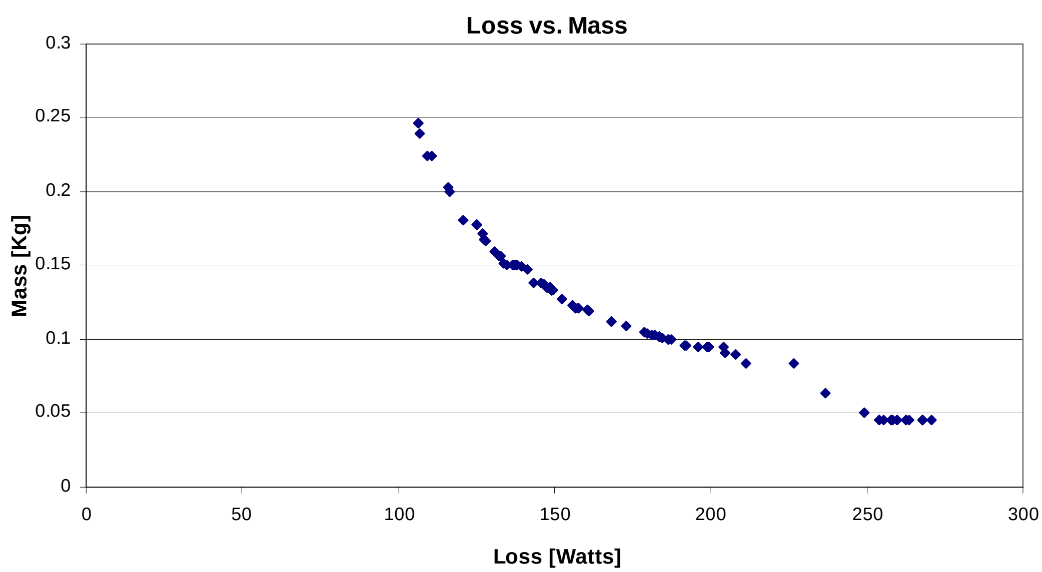


Figure 9.21: Pareto optimal solutions for a BLDC motor with Ns=9 and Np=10

9.3 Intermediate Analysis Phase of Progressive Design Methodology for the Design of a BLDC motor

In this section the results from the multi-criteria multiobjective optimisation obtained in the previous section are screened to reduce the number of feasible solution set. The application of various steps of *intermediate analysis* is explained in the following subsection.

9.3.1 Identification of new set of objectives:

For decision making the following parameters of the motor are taken into consideration

1. Stack length
2. Losses
3. Mass
4. Electrical time constant
5. Inertia of the rotor
6. Ratio of inner diameter of stator to outer diameter
7. Number of turns
8. Switching frequency
9. Width of the tooth
10. Thickness of the stator yoke
11. Input Voltage
12. Area of slots

The losses and mass of the motor are the primary parameters. A motor with smallest losses and smallest mass is preferable. However, as can be seen from the results of the previous section as the mass increases the losses decrease. Hence, in the *intermediate analysis* both are considered for the screening purpose. Electrical time constant of the motor has a direct influence on the dynamic performance of the motor. A motor with lower time constant has a better dynamic response compared to the motor with higher electrical time constant. Similarly, the inertia of the rotor is an important parameter because it influences the dynamic performance of the motor. A motor with high inertia will accelerate slowly compared to the motor with lower inertia. The ratio of inner diameter to outer diameter of stator is considered because it has an influence on the end turns of the winding.

Switching frequency has an impact on the performance of the motor. Higher switching frequency results in lower torque ripple but higher switching losses and a lower switching frequency results in higher torque ripple but lower switching frequency.

The magnetic loading and the mechanical aspects determine the width of the tooth. If the tooth is too thin then it may not be able to withstand the mechanical forces acting on it. Hence, in this analysis tooth with higher thickness is preferred. The thickness required for the stator yoke depends on the magnetic loading of the machine as well as on the mechanical properties. If the number of the pole pairs is small, often the allowable magnetic loading and the mechanical loading determines the thickness of the stator yoke. However, if the number of pole pairs is high enough the stator yoke may be thin if it is sized according to the allowed magnetic loading. The mechanical constraints may thus determine the minimum thickness of the stator yoke. In the decision making process smaller thickness of stator yoke is better. A smaller yoke thickness is preferred because it reduces the mass of the steel lamination required. The area of the slot is considered as an objective because it influences the winding. A slot with smaller area is difficult to wind. Hence, in this analysis a larger slot area is preferred.

9.3.2 Linguistic Term Set:

For the screening purpose the Linguistic term set based on the ordered structure is used. A set of seven terms of ordered structured linguistic terms is used here:

$$S = \left\{ S_0 = none,\ s_1 = very\ low, s_2 = low,\ s_3 = medium,\ s_4 = high,\ s_5 = very\ high,\ s_6 = perfect \right\}$$

where $s_a < s_b\ iff\ a < b$ (where, ab=0..6). The linguistic terms set in addition satisfy the following conditions:

1. Negation operator: $Neg(s_i) = s_j,\ j = T - i\ (T+1 \text{ is the cardinality})$
2. Maximisation operator: $Max(s_i, s_j) = s_i,\ \text{if } s_i \geq s_j$
3. Minimisation operator: $Min(s_i, s_j) = s_i,\ \text{if } s_i \leq s_j$

9.3.3 The Semantic of Linguistic Term Set:

In this case the Semantic Based on the Ordered Structure is used. The terms are symmetrically distributed, i.e. it is assumed that linguistic term sets are distributed on a scale with an odd cardinal and the mid term representing an assessment of “approximately 0.5” and the rest of the terms are placed symmetrically around it.

9.3.4 Aggregation Operator for Linguistic Weighted Information:

In this case the Linguistic Weighted conjunction aggregation operator is used.

9.3.5 The Screening Process:

The importance of different parameters discussed in the previous section is shown in Table 9.2 below.

Table 9.2: Importance of different parameters used in screening process

Parameter	Importance	Direction
Length of the stack	M	L
Losses	H	L
Mass	VH	L
Electrical time constant	H	L
Inertia of the rotor	L	L
Ratio of inner to outer stator diameters	H	H
Number of turns	M	L
Reminance field of permanent magnet	N	L
Switching frequency	M	L
Max. field density in stator lamination material	N	L
Width of the tooth	VL	L
Width of the yoke	L	L
Input Voltage	H	L
Area of slot	H	H

The length of the motor stack is given medium importance and the smaller the length of the motor the better it is, i.e. a smaller stack length is preferred over the larger length. For the losses a high importance is given and lower the losses the better it is. Similarly, for the mass a very high importance is given and smaller the mass the more preferred is the motor.

Table 9.3: Parameters of the set of solutions for Ns=6 and Np=4

Nturn	Lstack	DiDo	Bmag	Biron	Vdc	Freq	Losses	Mass	Jmot	Wt	wy	Time constant	Area of slot	Value
60	25.92	0.60	1.19	1.61	18.01	5.70	118.37	0.18	3.47	6.89	5.16	0.000274	3.22	VL
59	25.92	0.60	1.19	1.61	18.00	498.33	119.97	0.18	3.46	6.90	5.18	0.000270	3.21	VL
59	25.95	0.60	1.19	1.61	36.18	206.03	121.41	0.18	3.37	6.79	5.09	0.000281	3.26	VL
59	25.93	0.60	1.19	1.61	31.49	74.61	122.16	0.18	3.35	6.76	5.07	0.000291	3.29	VL
60	24.59	0.60	1.16	1.59	17.03	205.95	123.74	0.17	3.28	6.80	5.10	0.000293	3.30	L
60	24.59	0.60	1.16	1.59	17.18	200.10	124.07	0.17	3.27	6.80	5.10	0.000293	3.30	VL
60	24.56	0.60	1.16	1.60	34.61	200.10	129.79	0.17	3.20	6.62	4.97	0.000321	3.43	VL
59	23.26	0.60	1.15	1.61	10.34	206.05	130.26	0.16	3.12	6.69	5.02	0.000295	3.35	L
58	23.24	0.60	1.15	1.61	10.40	5.70	134.84	0.16	3.12	6.69	5.02	0.000295	3.35	L
52	23.08	0.60	1.12	1.53	10.34	498.78	146.08	0.16	2.99	6.62	4.96	0.000314	3.44	VL
59	20.23	0.60	1.19	1.69	18.12	40.70	150.23	0.14	2.63	6.45	4.84	0.000340	3.68	VL
59	20.19	0.60	1.19	1.69	21.43	29.01	151.07	0.14	2.61	6.43	4.82	0.000346	3.72	VL
59	19.13	0.60	1.19	1.69	17.96	40.78	152.10	0.13	2.49	6.46	4.84	0.000330	3.67	VL
59	18.21	0.60	1.19	1.69	17.92	6.84	154.58	0.13	2.40	6.54	4.91	0.000318	3.60	VL
59	18.13	0.60	1.16	1.69	48.00	5.05	158.43	0.13	2.43	6.41	4.81	0.000335	3.76	VL
59	18.13	0.60	1.15	1.69	39.10	5.18	161.31	0.13	2.44	6.35	4.76	0.000348	3.86	VL
59	16.50	0.60	1.17	1.70	17.12	41.37	166.13	0.12	2.25	6.33	4.75	0.000340	3.91	VL
59	13.77	0.60	1.17	1.78	17.99	5.17	173.64	0.10	1.95	6.19	4.64	0.000339	4.19	VL
59	13.76	0.60	1.19	1.78	47.97	5.70	180.04	0.10	1.78	6.09	4.57	0.000362	4.42	VL
59	12.25	0.60	1.15	1.85	42.35	16.39	194.80	0.09	1.99	6.02	4.52	0.000364	4.66	VL
59	10.82	0.60	1.15	1.85	17.68	5.05	211.54	0.08	1.47	5.83	4.37	0.000371	5.15	L
59	10.76	0.60	1.15	1.85	48.00	5.05	214.53	0.08	1.42	5.79	4.34	0.000378	5.27	L
59	**10.35**	**0.60**	**1.19**	**1.88**	**36.24**	**206.17**	**221.74**	**0.08**	**1.38**	**5.80**	**4.35**	**0.000381**	**5.35**	**M**
59	10.35	0.60	1.19	1.89	17.06	5.40	222.36	0.08	1.32	5.71	4.28	0.000388	5.55	VL
59	10.22	0.60	1.19	1.91	31.41	175.20	223.41	0.08	1.38	5.73	4.30	0.000379	5.46	VL
59	10.22	0.60	1.19	1.91	31.41	5.62	225.90	0.07	1.32	5.69	4.27	0.000390	5.61	VL
55	10.22	0.60	1.17	1.85	18.00	356.54	239.86	0.07	1.35	5.84	4.38	0.000356	5.12	VL
59	8.63	0.60	1.19	2.00	32.96	126.13	253.74	0.07	1.17	5.57	4.18	0.000374	6.03	VL
55	8.29	0.60	1.19	2.00	32.96	98.84	299.16	0.06	1.12	5.57	4.18	0.000364	6.04	VL
60	6.72	0.56	1.19	1.89	10.31	15.63	336.09	0.06	0.75	5.40	4.05	0.000453	7.92	VL
59	6.14	0.57	1.19	2.00	32.86	175.87	344.63	0.05	0.71	5.25	3.94	0.000441	8.53	VL
55	6.16	0.58	1.15	2.00	31.44	5.58	380.19	0.05	0.74	5.10	3.83	0.000423	8.69	VL
60	5.70	0.56	1.17	1.96	38.75	496.16	401.07	0.05	0.62	5.17	3.88	0.000458	9.28	VL
52	5.71	0.56	1.15	1.85	10.34	40.01	425.52	0.05	0.68	5.50	4.12	0.000398	7.60	VL
52	5.69	0.56	1.19	1.89	17.97	40.00	430.19	0.05	0.62	5.40	4.05	0.000401	7.90	VL
39	6.23	0.60	1.17	1.91	10.42	205.68	517.45	0.05	0.83	5.70	4.28	0.000285	5.67	VL
39	6.15	0.60	1.17	1.91	10.42	207.61	528.45	0.05	0.80	5.59	4.19	0.000291	5.96	VL
33	6.10	0.60	1.17	1.91	10.34	205.82	738.59	0.05	0.79	5.59	4.19	0.000289	5.96	VL

The electrical time constant of the motor is given a high importance and a lower value is better. Ratio of inner to outer stator diameters is given a higher importance and higher the value better it is. A medium importance is given to number of turns and lower number of turns is preferred. The reminance field of permanent magnet and maximum allowable field density of stator lamination is given no importance. The width of the tooth and width of the yoke are given very low and low importance respectively and lower the values of both the parameters the better it is. The area of the slot is given a high importance and the higher value of the slot area is preferred. The results of the multicriteria decision for motors with 6 slots and 4 poles is given in Table 9.3. and the best solution is marked in bold. The screening process has eliminated 37 solutions and only one competent solution was selected. This screening process was carried out for other pole slot combinations. The best solutions from all the pole slot combinations are given in Table 9.4.

Table 9.4: Parameters of the set of solutions after final screening

N_{turns}	L_{motor}	α_{dido}	α_{mp}	B_r	$B_{\max}$	h_m	V_{dc}	Mos_{typ}	F_{sw}	w_t	w_y	N_s	N_s
59	10.35	0.60	0.94	1.19	1.88	1.61	36.24	165	206.17	5.80	4.35	6	4
60	10.65	0.60	0.94	1.19	1.55	1.59	21.10	168	207.34	3.39	2.54	9	6
60	10.36	0.60	0.98	1.20	1.85	1.57	25.05	145	149.98	3.92	2.94	12	8
60	18.21	0.50	0.88	0.82	1.99	1.53	22.51	139	192.57	2.90	1.09	6	8
60	19.93	0.60	0.78	0.82	1.94	1.51	19.56	190	115.40	2.14	1.20	9	8
60	19.49	0.54	0.96	1.01	1.97	1.55	25.36	143	120.08	2.61	1.18	9	10

From the above Table 9.4 solutions were obtained after the screening process. Hence, these 6 solutions will be considered for detailed analysis. For detailed analysis FEM models of the motor are developed using FEMAG and smartFEM. In the next section the results of the final analysis are given.

9.4 Final Analysis Phase of Progressive Design Methodology for the Design of a BLDC motor

In this section detailed analysis of the motors obtained in the previous section is done. For the detailed analysis the ***formal deterministic*** and ***high fidelity*** model developed in chapter 6 is used. The results of cogging torque and the peak value of cogging torque for all the 6 alternatives in Table 9.4 are shown in Figure 9.22 and Figure 9.23 respectively.

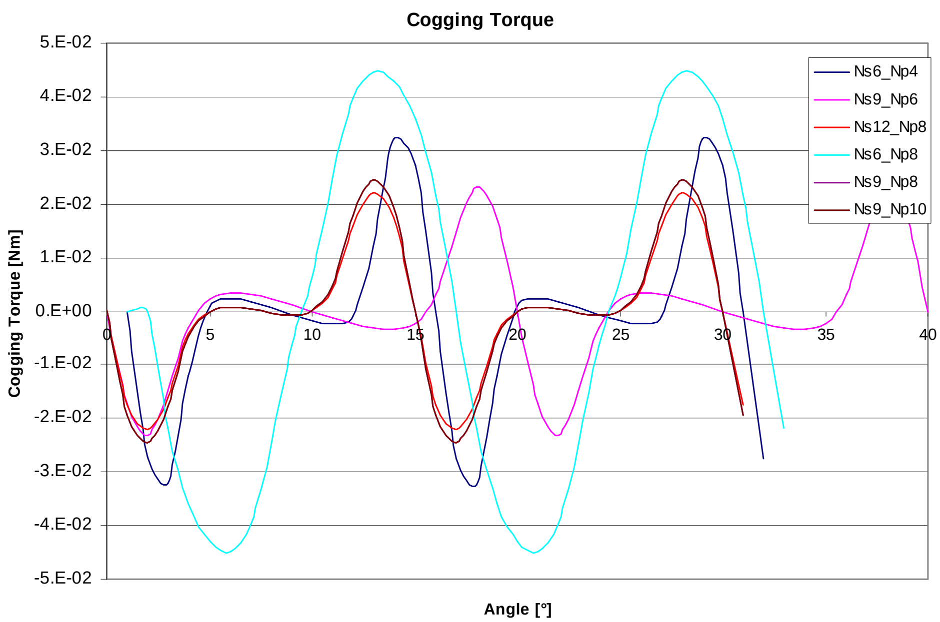


Figure 9.22: Cogging Torque for all the 6 motors

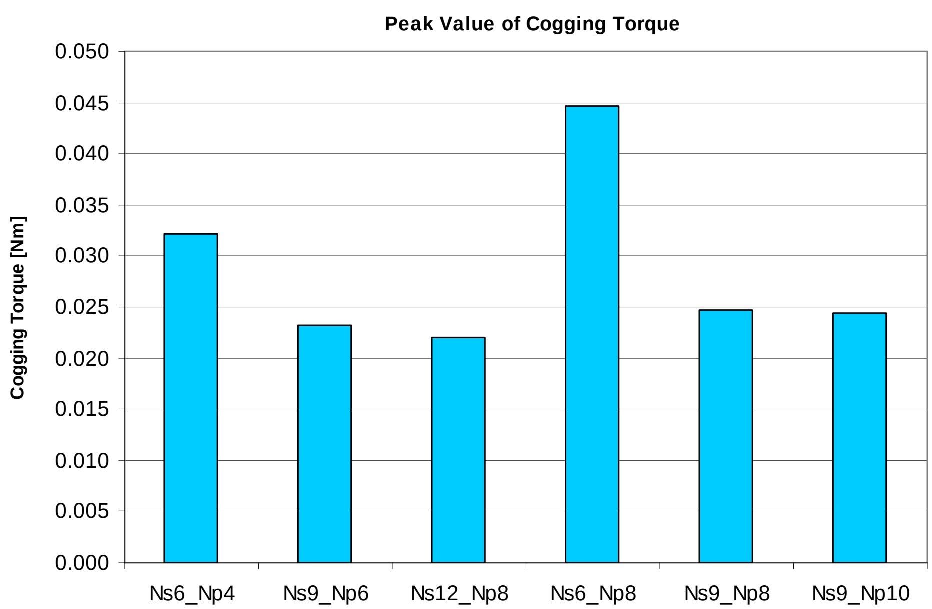


Figure 9.23: Peak values of the cogging torque for all the 6 motors

it is seen that motor with 12 slots and 8 poles has the minimum cogging torque, hence this motor was considered for detailed analysis and its parameters were determined so as to meet all the required criteria. The geometric parameters of the motor were fine-tuned so as to obtain cogging torque less than 0.02Nm and a trapezoidal back emf.

The final configuration of the motor is given in table below. Finally a prototype based on configuration given in Table 9.5 was made.

Table 9.5: Parameters of the motor after fine tuning

N_{turns}	L_{motor}	α_{diode}	α_{mp}	B_r	B_{max}	h_m	V_{dc}	Mos_{typ}	F_{sw}	w_t	w_y	N_s	N_s
60	10	0.60	1	0.65	1.57	1.505	24	165	206.17	2.015	1.511	12	8

The characteristics curves of the prototype are given in Figure 9.24 to Figure 9.26.

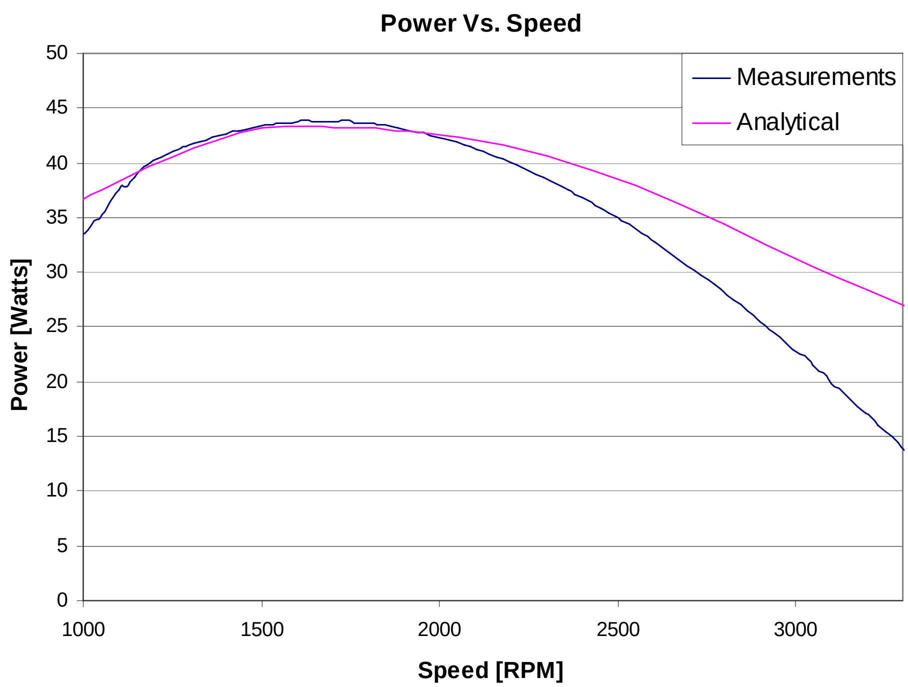


Figure 9.24: Power vs. Speed characteristics comparison between simulation and experiments

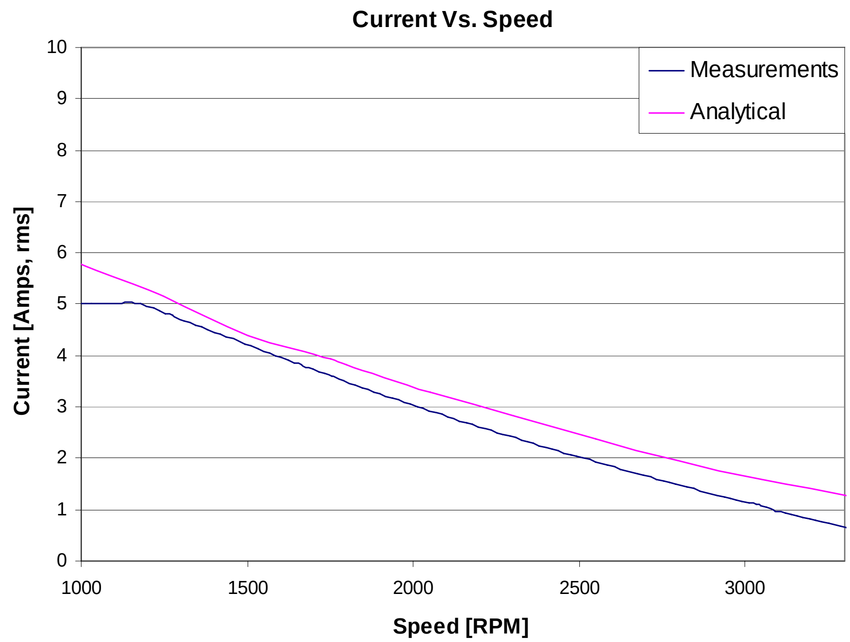


Figure 9.25: Current vs. Speed Characteristic comparison between simulations and experiment values

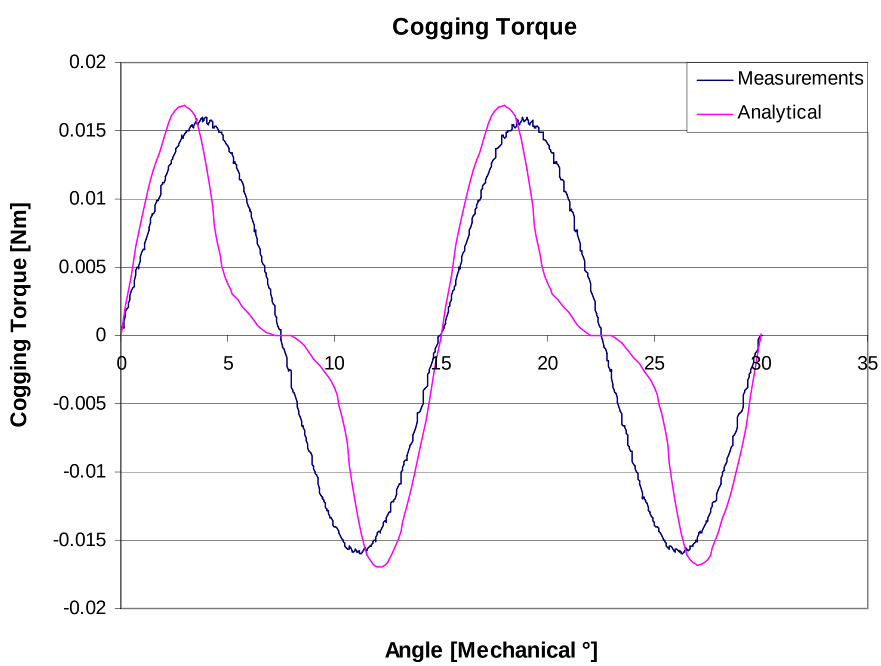


Figure 9.26: Cogging torque comaprison between simulations and experimental values

From the above figures it can be seen that the performance of the motor is close to the simulated values.

9.5 Conclusions

In this chapter the PDM (as proposed in chapter 2) was applied to the design of a BLDC motor. This methodology is suitable for designing a system from the conceptual stage to the final design. The goal of PDM is to enable design of optimal systems and reduce the cost of computing. In PDM the decision making factor is critical as proper decisions about dimensions, features, materials, and performance in the conceptual stage will ensure a robust and optimal design of the system. The different stages of PDM are explained using the example of the design of a BLDC motor.

9.6 References

[1] P. Kumar, D. Gospodaric, and P.Bauer, "Improved Genetic Algorithm Inspired by Biological Evolution," Soft Computing- A Fusion of Foundations, Methodologies and Applications, vol. 11, pp. 923-941, 2006.

Chapter 10

Discussion and Conclusions

10.1 Conclusions

In this thesis a methodology has been presented that enables the use of modelling, multiobjective optimisation and multicriteria decision making techniques to design engineering systems. The design methodology was explained and validated using a case study for the design of a BLDC motor drive. The focus of this thesis has been twofold. The first issue was to develop a framework where modelling and optimisation is used to accelerate and improve the design of the complex engineering systems. The second issue was to develop a reliable multiobjective optimisation algorithm.

In the first issue, the main goal is to develop a design process. In view of this, Progressive Design Methodology (PDM) has been presented in this thesis. An ideal situation in the design of a system will be when all the objective and constraints can be expressed by a simple model. However, in practical design problems this is seldom the case due to the complexity of the system. Hence, a multi-step PDM was proposed. The proposed PDM has three steps. In the first step (***Synthesis Phase***) a simple model of the components of a system is developed and the design problem is reformulated as a multiobjective optimisation problem (MOOP). In the second step (***Intermediate Analysis Phase***) the results obtained in the MOOP process are analysed and a small set of feasible solutions is selected. In the final step (***Final Analysis Phase***) a detailed model of the variants of the system, as selected from the previous set, are developed and the design variables of the system are fine tuned. Finally the final design of the system is selected.

The second issue has been to develop a reliable multiobjective optimisation algorithm. In this thesis it has been concluded that non-gradient based optimisation methods are suited for engineering optimisation problems since obtaining the derivatives of the objective functions may not be straightforward in most of the cases. Moreover, the non-gradient based methods are more robust in finding the global optimum and can be applied to a wide range of problems without any modification. The other advantage of these methods is that they can handle the mix of continuous and discrete variables. Hence, non-gradient based optimisation methods are applicable to a variety of problems without the need to tailor the method according to the problem. However, the non-gradient methods suffer from the disadvantage of high computational burden because they require many function evaluations. Here the advantages are considered to outweigh the disadvantages. In view of this the Nondominated Sorting Biologically Motivated Genetic Algorithms (NBGA) was proposed (Chapter 4).

The salient features of NBGA are use of point, large and chromosomal mutation to increase the diversity of solutions and improve the convergence of the Genetic Algorithms (GAs). Moreover, to facilitate the use of the algorithm, a new way of representing the variables was proposed in this thesis. The advantage of this new way is that the user need not define the rate of mutation for each type of mutation but can define a global rate of mutation. The reason for introducing this scheme of representation of variables was to make the algorithm more user friendly and accessible to professionals who may not be familiar with the details of GAs. The NBGA was compared with other state of the art GAs viz., NSGA-II, micro GA and PAES. For comparison, six benchmark test functions were used and the comparison was based on four performance parameters, Generational Distance (GD), Spacing (SP), Error Ratio (ER) and two set convergence (SC). The comparison showed that NBGA performed better than all the other GAs for all the six test functions. Due to the good diversity of solutions obtained and the better convergence, the NBGA has been used in PDM for multiobjective optimisation. The multiobjective optimisation forms a major part of ***Synthesis Phase*** of PDM.

When multiobjective optimisation is performed, a set of Pareto optimal solutions is obtained. However, in real situations the goal is to obtain a single solution. In order to achieve this, the designer has to eliminate the rest solutions based on certain criteria. The most important tasks in engineering design, besides modelling and simulation, are to generate various design alternatives and then to make preliminary decision to select a design or a set of designs that fulfils a set of criteria. Hence, the engineering design decision problem is a multi criteria decision-making problem. Thus, a multicriteria decision making (MCDM) process was presented in this thesis.

For MCDM different types of aggregation operators were presented. It was shown that in general, the linguistic weighted disjunction aggregation operators give an optimistic value to alternatives whereas the conjunction operators give a pessimistic value to the alternatives. Thus, a decision-making based on disjunction operator will result in a larger set of feasible alternatives as compared to conjunction operators. In PDM it was demonstrated that conjunction operators are suited for MCDM because it results in a smaller set of feasible alternatives. The small number of alternatives is preferred because in the ***Final Phase*** of PDM, detailed model of the system to be designed is developed for further investigation. These detail models are computationally expensive to evaluate. Hence, a smaller set of alternatives will reduce the time needed to reach the final design.

In PDM each phase has special requirements of the models. The ***Synthesis Phase*** involves multiobjective optimisation. Hence, this phase needs computationally inexpensive models. In view of this, it was concluded that ***low fidelity*** models were best suited for this phase. The characteristics of low fidelity models are that they are not computationally expensive and are able to capture the major performance parameters of the system (Chapter 5). The ***Intermediate Analysis phase*** involves decision making and in order to facilitate the decision making, ***judgmental models*** of the system are used. In the ***Final Analysis*** phase ***high fidelity*** models are used to evaluate a small set of alternatives and to perform fine-tuning.

In order to validate the PDM, it is applied to the design of a BLDC motor drive. In order to design the drive, simple and detailed analytical models of the motor and the voltage source inverter (VSI) are developed. The detailed analytical model of the motor was presented in chapter 6. It was shown that the model is suitable for the instantaneous air gap field density calculation and can be used to determine the cogging torque, induced back emf and iron losses. The salient feature of the model is that the iron (both stator and rotor yoke) has finite permeability and the thickness of the stator yoke is also finite. Besides this, the it was also shown that the model is flexible to take into account different types of magnetisation viz., radial magnetisation, parallel magnetisation, radial sinusoidal amplitude magnetisation and sinusoidal angle magnetisation. The comparison between the results obtained by the analytical model and Finite Element Method (FEM) showed very good agreement. The accuracy of the results and the speed makes this model suitable for Multiobjective Optimisation. In PDM this model was used in the final analysis phase of PDM.

The simple models are used in the synthesis phase, as they are highly suitable for multi objective optimisation (MOO) due to less computational time. The detailed analytical model is used in the final analysis phase to fine-tune the selected few variants obtained after the intermediate analysis phase. The final decision on the choice of variants is based on the results of the detailed models. In this thesis analytical methods have been used for detailed model of the motor. However, in this stage Finite element Models or Boundary element models can also be used.

To conclude, this thesis presents a design methodology that addresses many important aspects of the design of an engineering system. These aspects are presented together in a framework which demonstrates how modelling of the system, multiobjective optimisation and multicriteria decision making could be introduced in the design process in a way that results in a good final design of the system. Thus the thesis constitutes a step towards a framework for the design of engineering systems.

10.2 Future Direction

Paleontological findings have revealed that mass extinction has been a common phenomenon in evolution process [1]. The mass extinction has been suggested to be an important mechanism of evolution in the biological world [2]. This is because extinction allows the repopulating of niches and this in turn allows adaptation. In the field of evolutionary algorithms this idea has been introduced recently [3, 4]. To improve the NBGA further, one direction will be to implement the concept of mass extinction in NBGA. To accommodate the methodology for electrical engineering systems where different types of models are automatically connected.

Further work has to be done in the area of implementing methods that will make PDM more interactive and self-learning. When the PDM is used in a specific field for a long time then it should be able to learn with time. The advantage of this will be that when a new system is to be designed then PDM will be able to propose solutions based on its previous experiences. In such a case it will not be necessary to begin the design of the engineering system from scratch. Moreover, a self-learning PDM will be more interactive and can also be used to train new engineers in an organisation and also students in a classroom.

10.3 References

[1] C. Newton and L. Laporte, Ancient Environments. New Jersey: Prentice Hall, 1989.

[2] P. Bak, How Nature Works. New York: Springer Verlag, 1986.

[3] G. B. Fogel, "Evolutionary Computation with Extinction: Experiments and Analysis," presented at International Conference on Evolutionary Computation, 2000.

[4] T. Krink and R. Thomsen, "Self-organized Criticality and Mass Extinction in Evolutionary Algorithms," presented at International Conference on Evolutionary Computation, 2001.

Latin Letters

A_S	Slot area
$\boldsymbol{B}$	Magnetic field density
$B_{\max}$	Maximum flux density in steel
b_o	Slot opening
B_r	Reminance of permanent magnets
$\boldsymbol{B_o}$	Magnetic field density in the region exteriror of the motor
$\boldsymbol{B_s}$	Magnetic field density in the stator
$\boldsymbol{B_A}$	Magnetic field density in the air gap
$\boldsymbol{B_M}$	Magnetic field density in the magnet
$\boldsymbol{B_R}$	Magnetic field density in the rotor
C_e	Eddy current loss coefficient
C_h	Hysteresis loss coefficient
e	Error ratio metric
e_a	Back electromotive force of phase A
e_b	Back electromotive force of phase B
e_c	Back electromotive force of phase C
$\boldsymbol{f}(\vec{x})$	Set of objective functions
F_{sw}	Switching frequency
$\boldsymbol{g}(\vec{x})$	Set of inequality constraints
g	Air gap length

GD	Generational distance metric
$\boldsymbol{H}$	Magnetic field vector
h_m	Height of the magnet
$\boldsymbol{h}(\vec{x})$	Set of equality constraints
i_a	Current in phase A
i_b	Current in phase B
i_c	Current in phase C
I_0	Initial value of current
I_{ph}	Phase current
K_{cu}	Copper fill factor
K_e	Back electromotive force constant
LD	Linguistic disjunction
L_e	End turn inductance
L_g	Air gap inductance
LI	Linguistic implication
L_{motor}	Length of the motor
L_{ph}	Phase inductance
L_s	Slot inductance
$\boldsymbol{M}$	Magnetisation vector
M_{copper}	Mass of copper
M_{iron}	Mass of iron
M_{magnet}	Mass of magnet
N_m	Number of magnets in the motor
N_p	Number of poles in the motor

N_{ph}	Number of phases in the motor
N_r	Rated speed of motor
N_s	Number of slots
N_{turns}	Number of turns per coil
P^*	Pareto optimal set
P_{cu}	Copper loss
P_{eddy}	Eddy current loss
PF^*	Pareto front
P_{hys}	Hysteresis loss
P_{losses}	Losses in the motor
P_{MOSFET}	Switching losses in MOSFET
R_m	Radius of magnet
R_o	Outer radius of motor
R_{ro}	Outer radius of rotor
R_{ph}	Phase resistance
R_r	Radius of rotor
R_s	Inner radius of the stator
S	Spacing metric
T_{avg}	Average torque produced by the motor
T_{mot}	Instantaneous torque produced by the motor
t_c	Conduction time
V_a	Input voltage to phase A
V_b	Input voltage to phase B
V_c	Input voltage to phase C

V_{dc}	Input DC voltage
$V_{\max}$	Maximum value of induced voltage
w_t	Width of the stator tooth
w_y	Width of the stator yoke
$\vec{x}^L$	Vector of lower bound of variables
$\vec{x}^U$	Vector of upper bound of variables

Greek Letters

α_{dido}	Ratio of inner diameter of motor to outer diameter
α_m	Magnet angle
α_{mp}	Ratio of magnet angle to pole pitch
ω_m	Mechanical speed of the motor
ω_r	Electrical rotational speed of the motor
ϕ	Advanced Firing Angle
θ_r	Rotor Angle
θ_c	Commutation Angle
μ_0	Permeability of Free Space
μ_1	Relative Permeability of Stator Iron
μ_2	Relative Permeability of Permanent Magnets
μ_3	Relative Permeability of Rotor Iron
φ_A	Magnetic Scalar Potential in the Air Gap
φ_M	Magnetic Scalar Potential in the Magnet
φ_S	Magnetic Scalar Potential in the Stator
φ_R	Magnetic Scalar Potential in the Rotor

φ_O	Magnetic Scalar Potential in the exterior (outer) region
ρ	Resistivity of copper
ρ_{Fe}	Density of iron
ρ_m	Density of magnet

Acronyms

BLDC	Brushless Direct Current
CFD	Computational Fluid Dynamics
DM	Decision Maker
DNA	Deoxyribonucleic Acid
EMF	Electromotive Force
FEM	Finite element Method
GAs	Genetic Algorithms
GD	Generational Distance
IDV	Independent Design Variable
LDW	Linguistic Weighted Disjunction
LWC	Linguistic Weighted Conjunction
MCDM	Multicriteria Decision Making
MOOA	Multiobjective Optimisation Algorithms
MOOGAs	Multiobjective Optimisation Genetic Algorithms
MOOP	Multiobjective Optimisation Problem
NBGA	Non-dominated sorting Biologically Motivated Genetic Algorithm
NSGA	Non-dominated Sorting Genetic Algorithm
NSGA-II	Non Dominated Sorting Genetic Algorithm II
PAES	Pareto Archived Genetic Algorithm
PDM	Progressive Design Methodology
PM	Permanent Magnet
SBO	Surrogate Based Optimisation
SC	Set convergence
SPEA	Strength Pareto Evolutionary Algorithm

VSI Voltage Source Inverter

Mathematical Symbols

$\wedge$ And operator

$\forall$ For all, for each (for e.g., $\forall\ x\ P(x)$ means $P(x)$ is true for all x)

$\exists$ There exists (for e.g., $\exists\, x\ P(x)$ means there is at least one x such that $P(x)$ is true)

$\in$ Is an element of (for e.g., $a \in S$ means a is an element of the set S)

$\neg$ Logical negation

$\{:\}, \{|\}$ The set of…such that (for e.g., $\{x : P(x)\}$ means the set of all x for which $P(x)$ is true)

$\approx$ Approximately equal

□ Equal by definition

Appendix A

Results of Linguistic Aggregation Functions

The calculation of the total score of the alternatives of table 2.1 using different aggregation functions are given here.

A.1 Linguistic Weighted Disjunction (LWD)

The total score of different alternatives using LWD are given below:

1. The MIN Linguistic Disjunction $LD_1^{\rightarrow}$:

$$LD_1^{\rightarrow}(w,a) = MIN(w,a)$$

Based on the example given in table 2.1 the net performance of the first alternative based on $LC_1^{\rightarrow}$ is

$$f_1 = MAX\left[MIN(VH,M), MIN(VH,L), MIN(M,OU), MIN(L,VH), MIN(VL,OU)\right]$$
$$= MAX[M,L,M,L,VL] = M$$

The total score of the second alternative is

$$f_2 = MAX\left[MIN(VH,M), MIN(VH,OU), MIN(M,VH), MIN(L,VH), MIN(VL,H)\right]$$
$$= MAX[M,VH,M,L,VL] = VH$$

Total score of the third alternative is

$$f_3 = MAX\left[MIN(VH,H), MIN(VH,M), MIN(M,VH), MIN(L,H), MIN(VL,OU)\right]$$
$$= MAX[H,M,M,L,VL] = H$$

The total score of the fourth alternative is

$$f_4 = MAX\left[MIN\left(VH,OU\right), MIN\left(VH,M\right), MIN\left(M,H\right), MIN\left(L,VH\right), MIN\left(VL,H\right)\right]$$
$$= MAX\left[VH,M,M,L,VL\right] = VH$$

Finally the total score of the fifth alternative is

$$f_5 = MAX\left[MIN\left(VH,H\right), MIN\left(VH,M\right), MIN\left(M,H\right), MIN\left(L,OU\right), MIN\left(VL,VH\right)\right]$$
$$= MAX\left[H,M,M,L,VL\right] = H$$

Hence on the basis of $LD_1^{\rightarrow}$ the net performance of the various alternatives is $\left[M,VH,H,VH,H\right]$. Based on

$LD_1^{\rightarrow}$ the third and the fifth alternatives are selected because both have highest (H) final score.

The Nilpotent Linguistic Disjunction $LD_2^{\rightarrow}$:

$$LD_2^{\rightarrow}\left(w,a\right) = \begin{cases} MIN\left(w,a\right) & \text{if } w > Neg(a) \\ s_1 & \text{otherwise} \end{cases}$$

Based on the example given in table 2.1 the net performance of the first alternative based on $LD_2^{\rightarrow}$ is

$$f_1 = MAX\left[LD_2^{\rightarrow}\left(VH,M\right), LD_2^{\rightarrow}\left(VH,L\right), LD_2^{\rightarrow}\left(M,OU\right), LD_2^{\rightarrow}\left(L,VH\right), LD_2^{\rightarrow}\left(VL,OU\right)\right]$$
$$= MAX\left[M,L,M,L,VL\right] = M$$

The total score of the second alternative is

$$f_2 = MAX\left[LD_2^{\rightarrow}\left(VH,M\right), LD_2^{\rightarrow}\left(VH,OU\right), LD_2^{\rightarrow}\left(M,VH\right), LD_2^{\rightarrow}\left(L,VH\right), LD_2^{\rightarrow}\left(VL,H\right)\right]$$
$$= MAX\left[M,VH,M,L,N\right] = VH$$

The total score of the third alternative is

$$f_3 = MAX\left[LD_2^{\rightarrow}(VH,H), LD_2^{\rightarrow}(VH,M), LD_2^{\rightarrow}(M,VH), LD_2^{\rightarrow}(L,H), LD_2^{\rightarrow}(VL,OU)\right]$$
$$= MAX\,[H,M,M,N,VL] \; = H$$

The total score of the fourth alternative is

$$f_4 = MAX\left[LD_2^{\rightarrow}(VH,OU), LD_2^{\rightarrow}(VH,M), LD_2^{\rightarrow}(M,H), LD_2^{\rightarrow}(L,VH), LD_2^{\rightarrow}(VL,H)\right]$$
$$= MAX\,[VH,M,M,L,N] \; = VH$$

The total score of the fifth alternative is

$$f_5 = MAX\left[LD_2^{\rightarrow}(VH,H), LD_2^{\rightarrow}(VH,M), LD_2^{\rightarrow}(M,H), LD_2^{\rightarrow}(L,OU), LD_2^{\rightarrow}(VL,VH)\right]$$
$$= MAX\,[H,M,M,L,N] \; = H$$

Hence on the basis of $LD_2^{\rightarrow}$ the final score of all the five alternatives is $[M,VH,H,VH,H]$. Based on $LD_2^{\rightarrow}$ the third and the fifth alternatives are selected because they have the highest (H) final values.

2. The Weakest Linguistic Disjunction $LD_3^{\rightarrow}$:

$$LD_3^{\rightarrow}(w,a) = \begin{cases} MIN(w,a) & \text{if } MAX(w,a) = s_7 \\ s_1 & \text{otherwise} \end{cases}$$

Based on the example given in table 2.1 the net performance of the first alternative based on $LD_3^{\rightarrow}$ is

$$f_1 = MAX\left[LD_3^{\rightarrow}(VH,M), LD_3^{\rightarrow}(VH,L), LD_3^{\rightarrow}(M,OU), LD_3^{\rightarrow}(L,VH), LD_3^{\rightarrow}(VL,OU)\right]$$
$$= MAX\,[N,N,M,N,VL] \; = M$$

The total score of the second alternative is

$$f_2 = MAX\left[LD_3^{\rightarrow}(VH, M), LD_3^{\rightarrow}(VH, OU), LD_3^{\rightarrow}(M, VH), LD_3^{\rightarrow}(L, VH), LD_3^{\rightarrow}(VL, H)\right]$$
$$= MAX\,[N, VH, N, N, N] \;= VH$$

The total score of the third alternative is

$$f_3 = MAX\left[LD_3^{\rightarrow}(VH, H), LD_3^{\rightarrow}(VH, M), LD_3^{\rightarrow}(M, VH), LD_3^{\rightarrow}(L, H), LD_3^{\rightarrow}(VL, OU)\right]$$
$$= MAX\,[N, N, N, N, VL] \;= VL$$

The total score of the fourth alternative is

$$f_4 = MAX\left[LD_3^{\rightarrow}(VH, OU), LD_3^{\rightarrow}(VH, M), LD_3^{\rightarrow}(M, H), LD_3^{\rightarrow}(L, VH), LD_3^{\rightarrow}(VL, H)\right]$$
$$= MAX\,[VH, N, N, N, N] \;= VH$$

The total score of the fifth alternative is

$$f_5 = MAX\left[LD_3^{\rightarrow}(VH, H), LD_3^{\rightarrow}(VH, M), LD_3^{\rightarrow}(M, H), LD_3^{\rightarrow}(L, OU), LD_3^{\rightarrow}(VL, VH)\right]$$
$$= MAX\,[N, N, N, L, N] \;= L$$

Hence on the basis of $LD_3^{\rightarrow}$ the final score of all the five alternatives is $[M, VH, VL, VH, L]$. Based on $LD_3^{\rightarrow}$ the first alternative is selected because it has the highest score (M) among all the alternatives.

A.2 Linguistic Weighted Conjunction (LWC):

The total score of different alternatives using LWC are given below:

1. Kleene-Dienes's Linguistic Implication Function $LI_1^{\rightarrow}$:

   $$LI_1^{\rightarrow}(w, a) = Max\left(Neg(w), a\right)$$

   Based on the example given in table 2.1 the net performance of the first alternative based on $LI_1^{\rightarrow}$ is

$$f_1 = MIN\left[LI_1^{\rightarrow}(VH,M), LI_1^{\rightarrow}(VH,L), LI_1^{\rightarrow}(M,OU), LI_1^{\rightarrow}(L,VH), LI_1^{\rightarrow}(VL,OU)\right]$$
$$= MIN[M,L,OU,VH,OU] = L$$

The final score of the second alternative is

$$f_2 = MIN\left[LI_1^{\rightarrow}(VH,M), LI_1^{\rightarrow}(VH,OU), LI_1^{\rightarrow}(M,VH), LI_1^{\rightarrow}(L,VH), LI_1^{\rightarrow}(VL,H)\right]$$
$$= MIN[M,OU,VH,VH,VH] = M$$

The final score of the third alternative is

$$f_3 = MIN\left[LI_1^{\rightarrow}(VH,H), LI_1^{\rightarrow}(VH,M), LI_1^{\rightarrow}(M,VH), LI_1^{\rightarrow}(L,H), LI_1^{\rightarrow}(VL,OU)\right]$$
$$= MIN[H,M,VH,H,OU] = M$$

The final score of the fourth alternative is

$$f_4 = MIN\left[LI_1^{\rightarrow}(VH,OU), LI_1^{\rightarrow}(VH,M), LI_1^{\rightarrow}(M,H), LI_1^{\rightarrow}(L,VH), LI_1^{\rightarrow}(VL,H)\right]$$
$$= MIN[OU,M,H,VH,VH] = M$$

The final score of the fifth alternative is

$$f_5 = MIN\left[LI_1^{\rightarrow}(VH,M), LI_1^{\rightarrow}(VH,M), LI_1^{\rightarrow}(M,H), LI_1^{\rightarrow}(L,OU), LI_1^{\rightarrow}(VL,H)\right]$$
$$= MIN[M,M,H,OU,VH] = M$$

Hence on the basis of $LI_1^{\rightarrow}$ the final score of all the alternatives is $[L,M,M,M,M]$.

2. Gödel's Linguistic Implication Function $LI_2^{\rightarrow}$:

$$LI_2^{\rightarrow}(w,a) = \begin{cases} s_T & \text{if } w \le a \\ a & \text{otherwise} \end{cases}$$

Based on the example given in table 2.1 the net performance of the alternative based on $LI_2^{\rightarrow}$ is

$$f_1 = MIN\left[LI_2^{\rightarrow}(VH,M), LI_2^{\rightarrow}(VH,L), LI_2^{\rightarrow}(M,OU), LI_2^{\rightarrow}(L,VH), LI_2^{\rightarrow}(VL,OU)\right]$$
$$= MIN[M,L,OU,OU,OU] = L$$

The overall performance of second alternative is

$$f_2 = MIN\left[LI_2^{\rightarrow}(VH,M), LI_2^{\rightarrow}(VH,OU), LI_2^{\rightarrow}(M,VH), LI_2^{\rightarrow}(L,VH), LI_2^{\rightarrow}(VL,H)\right]$$
$$= MIN[M,OU,OU,OU,OU] = M$$

The overall performance of third alternative is

$$f_3 = MIN\left[LI_2^{\rightarrow}(VH,H), LI_2^{\rightarrow}(VH,M), LI_2^{\rightarrow}(M,VH), LI_2^{\rightarrow}(L,H), LI_2^{\rightarrow}(VL,OU)\right]$$
$$= MIN[H,M,OU,OU,OU] = M$$

The overall performance of fourth alternative is

$$f_4 = MIN\left[LI_2^{\rightarrow}(VH,OU), LI_2^{\rightarrow}(VH,M), LI_2^{\rightarrow}(M,H), LI_2^{\rightarrow}(L,VH), LI_2^{\rightarrow}(VL,H)\right]$$
$$= MIN[OU,M,OU,OU,OU] = M$$

The overall performance of the fifth alternative is

$$f_5 = MIN\left[LI_2^{\rightarrow}(VH,H), LI_2^{\rightarrow}(VH,M), LI_2^{\rightarrow}(M,H), LI_2^{\rightarrow}(L,OU), LI_2^{\rightarrow}(VL,VH)\right]$$
$$= MIN[H,M,OU,OU,OU] = M$$

Hence on the basis of $LI_2^{\rightarrow}$ the net performance of all the alternatives is $[L,M,M,M,M]$.

3. Fodor’s Linguistic Implication Function $LI_3^{\rightarrow}$:

$$LI_3^{\rightarrow}(w,a) = \begin{cases} s_T & \text{if } w \le a \\ MAX\left(Neg(w),a\right) & \text{otherwise} \end{cases}$$

Based on the example given in table 2.1 the net performance of the alternative based on $LI_3^{\rightarrow}$ is

$$f_1 = MIN\left[LI_3^{\rightarrow}(VH,M), LI_3^{\rightarrow}(VH,L), LI_3^{\rightarrow}(M,OU), LI_3^{\rightarrow}(L,VH), LI_3^{\rightarrow}(VL,OU)\right]$$
$$= MIN[M,L,OU,OU,OU] = L$$

The total performance of the second alternative is

$$f_2 = MIN\left[LI_3^{\rightarrow}(VH,M), LI_3^{\rightarrow}(VH,OU), LI_3^{\rightarrow}(M,VH), LI_3^{\rightarrow}(L,VH), LI_3^{\rightarrow}(VL,H)\right]$$
$$= MIN[M,OU,OU,OU,OU] = M$$

The total performance of the third alternative is

$$f_3 = MIN\left[LI_3^{\rightarrow}(VH,H), LI_3^{\rightarrow}(VH,M), LI_3^{\rightarrow}(M,VH), LI_3^{\rightarrow}(L,H), LI_3^{\rightarrow}(VL,OU)\right]$$
$$= MIN[H,M,OU,OU,OU] = M$$

The net performance of the fourth alternative is

$$f_4 = MIN\left[LI_3^{\rightarrow}(VH,OU), LI_3^{\rightarrow}(VH,M), LI_3^{\rightarrow}(M,H), LI_3^{\rightarrow}(L,VH), LI_3^{\rightarrow}(VL,H)\right]$$
$$= MIN[OU,M,OU,OU,OU] = M$$

The net performance of the fifth alternative is

$$f_5 = MIN\left[LI_3^{\rightarrow}(VH,H), LI_3^{\rightarrow}(VH,M), LI_3^{\rightarrow}(M,H), LI_3^{\rightarrow}(L,OU), LI_3^{\rightarrow}(VL,VH)\right]$$
$$= MIN[H,M,OU,OU,OU] = M$$

Hence on the basis of $LI_3^{\rightarrow}$ the performance of the five alternatives is $[L,M,M,M,M]$.

4. Lukasiewicz's Linguistic Implication Function $LI_4^{\rightarrow}$:

$$LI_4^{\rightarrow}(w,a) = \begin{cases} s_T & \text{if } w \le a \\ Neg(w-a) & \text{otherwise} \end{cases}$$

Based on the example given in table 2.1 the net performance of the alternative based on $LI_4^{\rightarrow}$ is

$$f_1 = MIN\left[LI_4^{\rightarrow}(VH,M), LI_4^{\rightarrow}(VH,L), LI_4^{\rightarrow}(M,OU), LI_4^{\rightarrow}(L,VH), LI_4^{\rightarrow}(VL,OU)\right]$$
$$= MIN[H,M,OU,OU,OU] = M$$

The total performance of the second alternative is

$$f_2 = MIN\left[LI_4^{\rightarrow}(VH,M), LI_4^{\rightarrow}(VH,OU), LI_4^{\rightarrow}(M,VH), LI_4^{\rightarrow}(L,VH), LI_4^{\rightarrow}(VL,H)\right]$$
$$= MIN[H,OU,OU,OU,OU] = H$$

The total performance of the third alternative is

$$f_3 = MIN\left[LI_4^{\rightarrow}(VH,H), LI_4^{\rightarrow}(VH,M), LI_4^{\rightarrow}(M,VH), LI_4^{\rightarrow}(L,H), LI_4^{\rightarrow}(VL,OU)\right]$$
$$= MIN[H,M,OU,OU,OU] = M$$

The total performance of the fourth alternative is

$$\begin{aligned} f_4 &= MIN\left[LI_4^{\rightarrow}(VH,OU), LI_4^{\rightarrow}(VH,M), LI_4^{\rightarrow}(M,H), LI_4^{\rightarrow}(L,VH), LI_4^{\rightarrow}(VL,H)\right] \\ &= MIN[OU,H,OU,OU,OU] = H \end{aligned}$$

The total performance of the fifth alternative is

$$\begin{aligned} f_5 &= MIN\left[LI_4^{\rightarrow}(VH,H), LI_4^{\rightarrow}(VH,M), LI_4^{\rightarrow}(M,H), LI_4^{\rightarrow}(L,OU), LI_4^{\rightarrow}(VL,VH)\right] \\ &= MIN[VH,H,OU,OU,OU] = H \end{aligned}$$

Hence on the basis of $LI_4^{\rightarrow}$ the total performance of all the five alternatives is $[M,H,M,H,H]$.

Appendix B

State of the Art Multiobjective Optimisation Algorithms

In this appendix some of the important Multiobjective Optimisation Algorithms (MOOA) other than GAs are given. The algorithms presented in the subsequent sections are inspired from various processes in the nature (except aggregation of objective based methods).

B.1 Aggregation of Objectives Based Methods

One of the earliest techniques for MOO are the *aggregation* methods. In *aggregation* methods the objective functions are replaced by some parametrised single function such as a weighted sum of the objectives [1]:

$$\text{minimise } \sum_{\mathrm{i=1}}^{\mathrm{k}} w_i f_i(x) \tag{B.1}$$

where $\sum_k w_i = 1$ and $w_i \geq 0$, for $i \in 1,...,k$.

The other *aggregation* methods are possible e.g. Tchebycheff problem [1]:

$$\text{minimise } \max_{i\in 1,...,k} \left[w_i \,|\, f_i(x) - z_i^* \right] \tag{B.2}$$

where z^* is a reference point beyond the ideal point, i.e. each of its component is less than the minimum value possible on the corresponding objective. With such a reference point correctly specified, every Pareto optimal solution minimises the function for some particular value of the weights. In order to achieve well spread Pareto front care must be exercised with respect to how the weights are adjusted [1].

Various other *aggregation* algorithms are epsilon-constraint methods [2], weighted metric methods [3] , Weighted goal programming method [3], etc. Detailed survey of these methods can be found in references [3-5].

The major disadvantage of these methods is that only one Pareto optimal solution can be obtained in each run of the algorithms [3]. The other difficulty with *aggregation* methods is determining the appropriate weights when enough information about the objective functions is not available [5]. In this case any optimal solution obtained will be a function of the weights used to combine the objectives. In many cases a simple linear combination of the objectives and than trade-off solutions are obtained by varying the weights. This approach is simple to implement but has the disadvantage of missing concave portion of the trade-off curve [6].

To overcome these problems algorithms based on stochastic methods are better suited for MOO problems. In the subsequent sub-sections a survey of different stochastic methods is given.

B.2 Simulated Annealing Method

Simulated annealing (SA) method was introduced by Kirkpatrick et.al. [7] while working for the research division at IBM. The strength of SA lies in the good selection schemes and annealing techniques. Generally SA used two kinds of selection techniques, Metropolis algorithm and logistic selection algorithm [8]. Originally any kind of selection that satisfies the detailed balance equation can be used as a selection scheme because the detailed balance equation guarantees the convergence of SA [9] .

The simulated annealing method is based on the analogy of thermodynamics, specifically with the way liquids freeze and crystallise or metals cool and anneal so that it adopts a low-energy, crystalline state. At high temperature the molecules of a liquid move freely with respect to one another. If the liquid is cooled the thermal mobility is lost and is confined due to high-energy cost of movement. The atoms are often able to arrange themselves and form a crystal. Based on this analogy the function to be minimised is called *energy*, $E(x)$, of the state x and a parameter T (computational temperature) is introduced that is lowered throughout the simulation according to an annealing schedule. At each T the simulated annealing (SA) algorithm aims to draw samples from the equilibrium distribution

$$\pi_T(x) \propto \exp\left\{ {-E(x)}\big/{T} \right\} \qquad \text{(B.3)}$$

As $T \rightarrow 0$ more and more of the probability mass of π_T, is concentrated in the region of the global minimum of E, so eventually, assuming a sufficiently slow annealing schedule is used, any sample from $\pi_T(x)$ will lie at the minimum of *E* [10].

Sampling from the equilibrium distribution $\pi_T(x)$ at any particular temperature is usually achieved by Metropolis-Hastings sampling [8], which involves making proposals x' that are accepted with probability

$$A = \min\left(1, \exp\left\{ -\delta E(x', x) \big/ T \right\}\right) \tag{B.4}$$

where $\delta E(x', x) \equiv E(x') - E(x)$ (B.5)

Intuitively when *T* is high, perturbations from x to x' which increase the energy are likely to be accepted and the samples can explore the state space. Subsequently, as *T* is reduced, only perturbations leading to small increase in E are accepted, so that only limited exploration is possible as the system settles on the global minimum. The algorithm of SA is given in Algorithm A.1. In Algorithm A.1 in each of the K epochs (where epoch is the iterations, the computational temperature is fixed at T_K and L_K samples are drawn from π_K before the temperature is lowered in the next epoch. Each sample is a perturbation (mutation) of the current state from a proposal density; the perturbation state x' is accepted with probability given in equation A.4.

The extension of simple SA to multiobjective problem involves using the Pareto dominance concept together with the annealing scheme. The main obstacle for SA in multiobjective optimisation is its inability to find multiple solutions. To overcome this problem Nam et.al. [11] proposed a multi-objective SA as shown in Algorithm A.2 where s represents the current search space position and T is the temperature parameter that is gradually reduced as time progresses. A new search position s' is generated by the $N(s)$ function, its cost is evaluated and compared with the previous cost and if this cost is found to be better (non-dominated) then the new state is accepted. In case the new position (s') is dominated by the current state (s), it is accepted with some acceptance probability. If there is no superiority between the current state and the next state then the new state is accepted instead of the current state.

Algorithm B.1: Pseudo code of simulated annealing

Inputs:

$\{L_k\}_{k=1}^{K}$ Sequence of epoch duration

$\{T_k\}_{k=1}^{K}$ Sequence temperatures, $T_{k+1} < T_k$

x Initial feasible solution

for k=1 to K

    for I=1 to L_k

        $x' = perturb(x)$

        $\delta E = E(x') - E(x)$

        $u = rand(0,1)$

        if $u < \min(1, \exp(\delta E / T_k))$

            $x = x'$

        end

    end

end

Algorithm B.2: Pseudo code of multiobjective optimisation SA

$s = s_0$

$T = T_0$

Repeat

    Generate a neighbour $s' = N(s)$

    If $C(s')$ dominated $C(s)$

        Moves to s'

    Else if $C(s)$ dominated $C(s')$

        Move to s' with transitional probability

    Else if $C(s)$ and $C(s')$ do not dominate each other

        move to s'

    end if

    $T = annealing(T)$

repeat (until the termination criteria are satisfied)

Shu et.al. proposed [12] a multi-objective optimisation Simulated Annealing (MOOSA) using a generalised Pareto based fitness function. Serafini [13] proposed MOOSA based on target-vector approach to solve bi-objective optimisation problem and Czyzak [14] developed a MOOSA based on population based approach using weighted sum.

B.3 Ant Colony Optimisation

Social insects such as termites, ants, bees and some species of wasps are capable of complex and intelligent group behaviour [15]. Inspired from the group behaviour of ants Dorigo et.al. [16] introduced Ant Colony Optimisation (ACO) algorithm. The foraging group behaviour of the ants enables them to find the shortest paths between food sources and their nests [17]. As soon as an ant finds a source of food it evaluates the source quantity and quality and carries some food sample to the nest. While returning back, the ant deposits a chemical substance called *pheromone* on the ground. This pheromone serves to attract other ants to follow the same path. The ants taking the shorter path will return to the nest sooner than the ants taking the longer path thereby increasing the concentration of the pheromone on the shorter path. The higher the pheromone intensity, the higher will be the probability that the following ants take the respective (shorter) path. The primitive behaviour of leaving a pheromone trail results in collective intelligence behaviour [18]. This collective intelligent behaviour is the inspiration for artificial ant colonies that are developed to solve optimisation problems. Artificial ants are simple agents that use numerical information (artificial pheromone information) to communicate their experience while solving a particular problem to other ants [16, 19, 20].

The ACO algorithm incorporates artificial ants that follow the artificial pheromone trails represented by a parameterised probabilistic model termed as the pheromone model. The pheromone model consists of a set of model parameters whose values are called the pheromone values. The unique element of the ACO algorithm is the probabilistic construction of solutions using the pheromone values. The key issues of this solution construction technique are:

1. To generate a solution using the pheromone model from a large solution space that contains solutions of different quality.

2. To narrow the search towards the high quality solutions in the solution space by updating the pheromone values with the solutions that were constructed in earlier iterations.

The pseudo-code of ACO [19] is shown in Algorithm A.3. Initially parameters and pheromone information are initialised. A main loop is carried until a termination condition is met. The termination criterion may be a given number of solution constructions or a limit on the available computation time. In the main loop ants construct feasible solutions by adding solutions components. These solutions may be additionally improved using a local search. Next, the new best solution found (best-so-far solution) is determined. Subsequently, a number of solutions that may include the best-so-far solution are selected to update the pheromone information. Similar to the biological metaphor, *pheromone update* consists of two elements:

1. *Pheromone evaporation*: In this phase all pheromone values are uniformly decreased. This process implements a useful form of *forgetting*. Pheromone evaporation is important in diversifying the search process in the solution space.

2. *Pheromone deposit*: One or more solutions from the current and/or from earlier iterations are used to increase the values of pheromone trail parameters on solution components that are part of these solutions.

Algorithm B.3: Pseudo code of Ant Colony System

While termination condition not met do
    Solution construction with ants
    Phermones update
    Daemon Actions
    $s_{best} \leftarrow$ best solution in the population of solutions
end while
output: s_{best}

Several types of pheromone update procedures are available that aim in intensification and diversification of the search process and they differ in the way they update the pheromone values. An explanation of each pheromone update method is beyond the scope of this thesis. However, it is worthwhile to mention some of the ACO variants: Ant Colony System (ACS) [21], MAX-MIN Ant System (MMAS) [22], Elitist Ant Systems [21, 23].

Relatively few approaches of ACO for multiobjective optimisation problems have been proposed so far [21]. Many of the proposed algorithms have been applied to problems where the objectives can be ordered according to their importance [24-26]. Gambardella et. al. [25] studied bi-objective vehicle routing problems with time window constraints. The first objective is number of vehicles and the second objective is total travel time and Gambardella et.al. [25] gave more importance to number of vehicles compared to total travel time.

An attempt to create an ACO algorithm to solve continuous optimisation problems was made by Bilchev et. al. [27]. This algorithm was a combination of ACO with Genetic Algorithm. Later it was extended to Continuous Ant Colony Optimisation (CACO) algorithm [28]. Recently Christodoulou [29] [29] proposed an algorithm based on ant colony to optimise truss design.

B.3.1 Particle Swarm Optimisers

Particle Swarm Optimiser (PSO) was first presented by Kennedy and Eberhart [30, 31]. In recent years, PSO has been applied to a variety of problems such as multiobjective optimisation problem [32, 33], minimax problems [34, 35], integer programming problems [36], noisy and continuously changing environments [37-39] and numerous engineering applications. The PSO is a stochastic optimisation technique inspired by the behaviour of a flock of birds or sociological behaviour of a group of people. The simplest version of PSO lets every individual move from a given point to a new one which is a weighted combination of the individual's best position ever found ("nostalgia") and of the group's best position ("publicised knowledge") [40].

A simple pseudo code of PSO is given in Algorithm A.4 [41]. A population of particles is initialised with random positions $\vec{x}_i$ and velocities $\vec{v}_i$ and a function is evaluated using the particle's positional co-ordinate as input values. Position and velocities are adjusted and the function evaluated with new co-ordinates at each time step. When a particle finds a pattern that is better than anyone it has found previously, it stores the co-ordinates in a vector $\vec{p}_i$. The difference between $\vec{p}_i$ and the individual's current position is stochastically added to the current velocity causing the trajectory to oscillate around that point. Additionally every particle is defined within the context of a topological neighbourhood comprising itself and some other particles in the population. The stochastically weighted difference between the neighbourhood's best position $\vec{p}_g$ and the individual's current position is also added to its velocity, adjusting it for the next time step. These adjustments of the particle's movement through the space cause it to search around two best positions. The variables φ_1 and φ_2 are random positive numbers drawn from a uniform distribution and defined by an upper limit $\varphi_{\max}$, which is a parameter of the system. The velocity of the particle at dth dimension v_{id} is limited to the range $\pm v_{\max}$. The values of the elements in $\vec{p}_g$ are determined by comparing the best performances of all the members of *i*'s topological neighbourhood ($\vec{p}_{neighbors}$), defined by indices of some other population members and assigning the best performer's index to the variable *g*. Hence, $\vec{p}_g$ represents the best position found by any member of the neighbourhood.

Algorithm B.4: Pseudo code of simple Particle Swarm Optimisation algorithm

Do
For i=1 to population size
    If $f(\vec{x}_i) < f(\vec{p}_i)$ then $\vec{p}_i = \vec{x}_i$
    $\vec{p}_g = \min(\vec{p}_{neighbors})$
        for d=1 to dimensions
            $v_{id} = v_{id} + \varphi_1 (p_{id} - x_{id}) + \varphi_2 (p_{gd} - x_{id})$
            $v_{id} = sign(v_{id}).\min(abs(v_{id}), v_{\max})$
        next d
    next I

The simple PSO has been modified to be made applicable to Multi-objective Optimisation Problems. The algorithm proposed by Moore and Chapman [42] was based on Pareto dominance. The authors emphasised on the importance of performing both an individual and a group search but did not adopt any scheme to maintain diversity. An algorithm combining the concepts of evolutionary techniques with particle swarm was proposed by Ray et.al. [43]. This algorithm uses crowding to maintain diversity and a multilevel sieve to handle constraints. The algorithm proposed by Parsopoulos and Vrahatis [44] adopts an aggregation function to combine the multiple objectives into a single objective. The authors proposed three types of approaches for aggregating the objective functions: a conventional linear aggregation function, a dynamic aggregation function and the bang-bang weighted aggregation approach [45] in which weights are varied in such a way that concave regions of the Pareto front can be generated. A novel Multiobjective optimisation PSO (MOPSO) was proposed by Coello et.al [46]. In this algorithm (MOPSO) the authors incorporated Pareto dominance in the PSO to handle problems with several objective functions. The MOPSO also uses a secondary repository of particles that is later used by other particles to guide their own movement in the search space. A special mutation operator to enrich the exploratory capabilities was also implemented in MOPSO.

B.4 Artificial Immune System

The biological immune system is a complex adaptive system that has evolved in vertebrates to protect them from invading pathogens. It is capable of recognising most antigens' (antigens are the foreign molecules belonging to pathogens that invade the body) attacks by immune cells known as B-cells. The B-cells circulate in the blood and lymphatic network and in case of invasion by antigens the B-cells attack and destroy them. Each antigen has a specific shape that is recognised by the receptors present on the B-cells surface, that is the B-cells synthesise and carry on their surface molecules called antibodies that act like detectors to identify antigens. If a B-cell is useful to recognise the antigen, it may be stimulated to clone itself and hence increases its population. Clonal selection ensures that only useful B-cells (higher affinity with antigens) can be cloned to represent the next generation [47-49]. The clones with lower affinity to antigen do not divide and will be discarded. This process ensures sufficient number of antigen-specific B-cells to build up an effective immune response.

Based on the principle of working of immune system, artificial immune system (AIS) algorithms are developed. The use of the immune system capabilities in artificial systems depends on the nature of the problem. The different areas of application of AIS are network security [48], parallel processing, image processing [49], robotics [50], Travelling

Salesman Problems (TSP) [51-53] and other areas [54, 55]. The pseudo code of AIS is given in Algorithm A.5.

Algorithm B.5: Pseudo code of Artificial Immune System for Multiobjective optimisation

```
Population Initialise
Classify population into antibody (AB) and Antigen (A)
Do (while termination criterion satisfied)
    Select an antigen A
    For each antibody belonging to AB
        Compute its affinity with the antigen A
        Mutate antibodies according to affinity
    New population formed by union of original AB and clones
    Population returned to original value allowing non-
dominated solutions to
    Survive
Loop
```

Artificial Immune System Optimisation methods have been applied to various engineering design problems. Chun et. al. [56] used an AIS based algorithm for optimisation of shapes of electromagnetic devices. An immune system based genetic algorithm with improved and faster global convergence was proposed by Tazawa et.al. [57]. Mori et.al. [58] proposed an immune algorithm for a multi modal function optimisation using ideas from immune diversity, clonal selection and genetic algorithms. Chun et.al. [59] applied a slightly modified immune algorithm developed by Mori et.al. to several function optimisation problems and compared its performance with that of evolutionary strategy and genetic algorithms. In addition the authors also applied this algorithm to determine the optimal design of a surface permanent magnet synchronous motor and a pole shape of an electromagnet [60]. An immune algorithm based upon the somatic theory and network hypothesis of immune system was proposed by Fukuda et.al. [61] to solve multi-modal function optimisation problems partly using genetic algorithms.

B.5 Endocrine Multi-objective Optimisation Algorithms (EMOOA)

The multiobjective optimisation algorithm based on the endocrine systems was proposed by Rotar [62]. The endocrine system comprises of different glands and each gland produces chemical messengers called hormones. These circulate in the blood until reaching the target organs upon which they are design to act, bind with the cells and begin to influence the working of those cells. The receptors of the cells recognise and tie with one type of hormone. The hormones induce changes in the target cells. The concentration of specific hormones is controlled through a feedback mechanism. This feedback is controlled by hormone called tropes. This hormone controls the releasing or inhibition of specific hormones. This principle is used in Endocrine Multi-objective Optimisation Algorithms (EMOOA).

The principle of EMOOA as proposed by Rotar [62] relies on maintaining two populations: an active population of hormones, H_t and a passive population of non-dominated solutions A_t. The members of the passive population behave as a population of elite members and also have a supplementary function to lead the hormones toward the Pareto Front, keeping them as much as possible well distributed among the search space. The two populations correspond to two classes of hormones in endocrine paradigm:

1. Specific hormones, which are released by different glands of the body – the active population H_t.
2. The hormones of control (tropes) that are produced by control level of the endocrine systems in order to supervise the density of each type of hormone-the passive population A_t.

The population of control, A_t, is modified at each generation. The new population A_{t+1} gathers all non-dominated solutions from the population U_t, which has resulted from merging current population of hormones H_t and the previous population A_t. This manner of changing the population of controllers assures us that the non-dominated solutions from the previous population A_t cannot be lost if they still remain non-dominated after the active population H_t has changed.

The passive population (A_t) set does not undergo modifications at the individual's level. This set behaves as an elite population of non-dominated solutions from the current generation, which is only actualised at each generation. Finally, the population A_t contains a predetermined number of non-dominated vectors and provides a good approximation of the Pareto Front. At each generation t, the members of H_t are classified into S_a classes. A corresponding controller from A_t supervises a particular class. The idea is that each hormone h from H_t is supervised by the nearest controller a_t from A_t:

$$C(a_i) = \left\{ h \in H_t,\ dist(h, a_i) = \min\left(dist(h, a_j) \right),\ j = 1, \ldots, S_a \right\} \tag{B.6}$$

Each member a from the set A_t has a similar control function as a trop from endocrine paradigm. Among the members of the current population H_t a special type of sharing is performed. Due to the fact that each individual a_i from the population A_t controls a class of hormones $C(a_i)$, this individual of control, a_i, imprints to each hormone a probability of selecting it as first parent. The selection of the first parent is made proportionally to the value of its class, which is calculated as follows:

$$val_i = 1 / sizeof\,(C(a_i))$$

The first parent is selected from the population H_t proportional to the value of its class. By this method the less crowded hormones are preferred resulting in wide spread solution. The second parent h is selected based on its performance value given by:

$$performance(h) = \text{nr\_dominated} / sh$$

where *nr_dominated* is the number of solutions from H_t that are dominated by the hormone h and s_h is the initial size of the population. The pseudo-code of EMOOA is given in Algorithm A.6.

Algorithm B.6: Pseudo code of endocrine algorithm for multi-objective Optimisation problem

Initialise the population

T=0 (number of current generation)

S_h=s_a (initial size of the population *A* is equal to the size of the population *H*)

Randomly generate the populations: H_t and A_t where:

$S_h=size(H_t)$ and $s_a=size(A_t)$

Repeat

Merge the two populations H_t and A_t resulting in U_t

The population U_t contains all individuals from H_t and A_t.

($sha=size(U_t)=size(H_t)+size(A_t)$)

Generate new A_{t+1}. Population A_{t+1} embodies all non-dominated solution from U_t:

$A_{t+1} = \cup\left\{u \in U_t,\ u \text{ is non-dominated}\right\}$, $s_a=size(A_{t+1})$

Classify the hormones from Ht according to A_{t+1} and set the crowding degrees of hormones. Each individual from A_{t+1} controls the hormones from its neighbourhood. Further, a particular hormone can recombine only with hormones from its class.

Evaluate H_t. The performance of each hormone h is proportional with the number of other individuals of H_t, which are dominated by *h*.

Generate H_{t+1}

$H_{t+1} = \varphi$

For each *h* from H_t, which is selected accordingly to its crowding degree, do:

Select a mate h' from the class of *h* hormone

Recombine *h* and h', in order to produce the descendant *d*.

Include the descendant into the next generation:

$H_{t+1} = H_{t+1} \cup \{d\}$

t=t+1

Until s_a=s_h

B.6 References

[1] J. Knowles and D. Corne, "Memetic Algorithms for Multiobjective Optimization: Issues, Methods and Prospects," in *Recent Advances in Memetic Algorithms*, N. Krasnogor, J. E. Smith, and W. E. Hart, Eds.: Springer Verlag, 2004, pp. 313-352.

[2] K. Miettinen, "Some methods for nonlinear multi-objective optimization," presented at First International Conference on Evolutionary Multi-Criterion Optimization (EMO 2001), Berlin, 2001.

[3] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*: John Wiley & Sons, 2001.

[4] C. A. C. Coello, "An enperical study of evolutionary techniques for multiobjective optimization in engineering design," in *Computer Science*: Tulane University, 1996.

[5] C. A. C. Coello, "An Updated Survey of GA-Based Multiobjective Optimization Techniques," *ACM Computing Surveys*, vol. 32, 2000.

[6] B. J. Ritzel, J. W. Eheart, and S. Ranjithan, "Using genetic algorithms to solve a multiple objective groundwater pollution containment problems," *Water Resources Research*, vol. 30, pp. 1589-1603, 1994.

[7] S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671-680, 1983.

[8] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbloth, A. H. Teller, and E. Teller, "Equation of State Calculation by Fast Computing Machines," *Journal of Chemical Physics*, vol. 21, pp. 1087-1092, 1953.

[9] D. Mitra, F. Romeo, and A. Sangiovanni-Vincentelli, "Convergence and Finite-time Behaviour of Simulated Annealing," *Advanced Applied Probability*, vol. 18, pp. 747-771, 1986.

[10] K. I. Smith, R. M. Everson, and J. E. Fieldsend, "Dominance Measures for Multiobjective Simulated Annealing," presented at IEEE Congress on Evolutionary Computation (CEC'04), Oregon, 2004.

[11] D. K. Nam and C. H. Park, "Multiobjective simulated annealing: a comparative study of evolutionary algorithms," *International Journal of fuzzy Systems*, vol. 2, pp. 87-97, 2000.

[12] D. K. Nam and C. H. Park, "Pareto-based cost simulated annealing for multiobjective optimization," presented at International Conference on Neural Information Processing, Singapore, 2002.

[13] P. Serafini, "Simulated annealing for multiobjective optimization problems," *Multiple criteria decision making. Expand and enrich the domains of thinking and application*, pp. 283-292, 1994.

[14] P. Czyzak and A. Jaszkiewicz, "Pareto simulated annealing- a metaheuristic technique for multiple-objective combinatorial optimization," *Journal of Multi-Criteria Decision Analysis*, vol. 7, pp. 34-47, 1998.

[15] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. New York: Oxford University Press, 1999.

[16] M. Dorigo and G. D. Caro, "The Ant Colony Optimization meta-heuristic," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds. London: McGraw-Hill, 1999.

[17] J. L. Deneubourg, S. Aron, S. Goss, and J. M. Pasteels, "The self-organizing exploratory pattern of the argentine ant," *Journal of Insect Behaviour*, vol. 3, pp. 159-168, 1990.

[18] M. Dorigo, E. Bonabeau, and G. Theraulaz, "Ant algorithms and stigmergy," *Future Generation Computer Systems*, vol. 16, pp. 851-871, 2000.

[19] M. Dorigo and T. Stützle, *Ant Colony Optimization*. Cambridge, MA: MIT Press, 2004.

[20] M. Dorigo, G. D. Caro, and L. M. Gambardella, "Ant algorithms for distributed discrete optimization," *Artificial Life*, vol. 5, pp. 137-172, 1999.

[21] M. Dorigo and L. M. Gambardella, "Ant Colony Systems: A cooperative learning approach to the travelling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, pp. 53-66, 1997.

[22] T. Stützle and H. Hoos, "MAx-MIN Ant System," *Future Generation Computer Systems*, vol. 16, pp. 889-914, 2000.

[23] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant System: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man and Cybernatics - Part B*, vol. 26, pp. 29-41, 1996.

[24] C. E. Mariano and E. Morales, "MOAQ an ant-Q algorithm for multiple objective optimization problems," presented at Genetic and Evolutionary Computation Conference, Orlando, Florida, 1999.

[25] L. M. Gambardella, E. D. Taillard, and G. Agazzi, "MACS-VRPTW: A multiple ant colony system for vehicle routing problems with time window," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds. London: McGraw-Hill, 1999, pp. 63-76.

[26] M. Gravel, W. L. Price, and C. Gagne, "Scheduling continuous casting of aluminium using a multiple objective ant colony optimization metaheuristics," *European Journal of Operational Research*, vol. 143, pp. 218-229, 2002.

[27] G. Bilchev and I. C. Parmee, "The ant colony metaphor for searching continuous design spaces," in *Lecture Notes in Computer Science*, T. Fogarty, Ed.: Springer Verlag, 1995, pp. 25-39.

[28] M. Wodrich and G. Bilchev, "Cooperative distributed search: the ant's way," *Control and Cybernetics*, vol. 26, pp. 413-445, 1997.

[29] S. Christodoulou, "Optimal Truss Design Using Ant Colony Optimization," presented at 5th GRACM International Congress on Computational Mechanics, Limassol, 2005.

[30] R. C. Eberhart, P. Simpson, and R. Dobbins, in *Computational Intelligence PC Tools*: Academic Press, 1996, pp. 212-226.

[31] R. C. Eberhart and J.Kennedy, "A new optimizer using particle swarm theory," presented at 6th International Symposium on Micro Machines and Human Science, Nagoya, Japan, 1995.

[32] X. Li, "A nondominated sorting particle swarm optimizer for multiobjective optimization," in *Lecture Notes in Computer Science (LNCS)*, vol. 2723, E. C. Paz, J. A. Foster, K. Deb, D. Davis, R. Roy, U.-M. O'Reilly, H. G. Beyer, R. Standish, G. Kendall, S. Wilson, M. Harman, J. Wegener, D. Dasgupta, M. A. Potter, A. C. Schultz, K. Dowsland, N. Jonoska, and J. Miller, Eds. New York: Springer Verlag, 2003, pp. 37-48.

[33] C. A. C. Coello, "MOPSO: A proposal for multiple objective particle swarm optimization," presented at IEEE Congress on Evolutionary Computation, Honolulu, HI, 2002.

[34] Y. Shi and R. A. Krohling, "Co-evolutionary particle swarm optimization to solve min-max problems," presented at IEEE 2002 Conference on Evolutionary Computation, Honolulu, HI, 2002.

[35] E. C. Laskari, K. E. Parsopoulos, and M. N. Vrahatis, "Particle swarm optimization for minmax problems," presented at IEEE 2002 Congress on Evolutionary Computation, Honolulu, HI, 2002.

[36] E. C. Laskari, K. E. Parsopoulos, and M. N. Vrahatis, "Particle swarm optimization for integer programming," presented at IEEE 2002 Congress on Evolutionary Computation, Honolulu, HI, 2002.

[37] K. E. Parsopoulos, "Particle swarm optimizer in noisy and continuously changing environments," in *Artificial Intelligence and Soft Computing*, M. Hamza, Ed. Cancun, Mexico: IASTED/ACTA Press, 2001, pp. 289-294.

[38] A. Carlisle, "Applying the particle swarm optimizer to non-stationary environments." Auburn: Auburn University, 2002.

[39] R. C. Eberhart and Y. Shi, "Tracking and optimizing dynamic systems with particle swarms," presented at IEEE 2001 Congress on Evolutionary Computation, Seoul, South Korea, 2001.

[40] G. Ciuprina, D. Ioan, and I. Munteanu, "Use of Intelligent-Particle Swarm Optimization in Electromagnetics," *IEEE Transactions on Evolutionary Computation*, vol. 38, pp. 1037-1040, 2002.

[41] M. Clerc and J. Kennedy, "The Particle Swarm-Explosion, Stability and Convergence in a Multidimentional Complex System," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 58-73, 2002.

[42] J. Moore and R. Chapman, "Application of Particle Swarm to Multiobjective Optimization," Auburn University, Auburn 1999.

[43] T. Ray and K. M. Liew, "A swarm metaphore for multiobjective design optimization," *Engineering Optimization*, vol. 34, pp. 141-153, 2002.

[44] K. E. Parsopoulos and M. N. Vrahatis, "Particle swarm optimization method in multiobjective problems," presented at ACM Symposium on Applied Computing (SAC'2002), Madrid, Spain, 2002.

[45] Y. Jin, T.Okabe, and B. Sendhoff, "Dynamic weighted aggregation for evolutionary multi-objective optimization: why does it work and how?," presented at Genetic and Evolutionary Computation Conference (GECC0'2001), San Francisco, CA, 2001.

[46] C. A. C. Coello, "Handling Multiple Objectives With Particle Swarm Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, pp. 256-279, 2004.

[47] S. A. Hofmeyr, "An Interpretative Intorduction to the Immune System.," in *In Design Principles for the Immune System and other Distributed Autonomous Systems*, I. Cohen and L. Segel, Eds.: Oxford University Press, 2000.

[48] S. A. Hofmeyr, "An Immunological Model of Distributed Detection and its Application to Computer Security," University of New Mexico, 1999.

[49] V. Rodin, A. Benzinou, A. Guillaud, P. Ballet, F. Harrouet, J. Tisseau, and J. L. Bihan, "An Immune Oriented Multi-agent System for Biological Image Processing," *Pattern Recognition*, vol. 37, pp. 631-645, 2004.

[50] Y. Watanabe, A. Ishiguro, and Y. Uchkawa, "Decentralized Behaviour Arbitration Mechanism for Autonomous Mobile Robot using Immune System," in *Artificial Immune Systems and Their Applications*: Springer Verlag, 1999.

[51] S. Endo, N. Toma, and K. Yamada, "Immune Algorithm for n-TSP. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics," pp. 3844-3849, 1998.

[52] N. Toma, S. Endo, and K. Yamada, "An Immune Co-evolutionary Algorithm for N-th Agent's Travelling Salesman Problems.," presented at IEEE International Symposium on Computational Inteligence in Robotics and Automation, Kobe, Japan, 2003.

[53] Y. Hui, L. Kang, Z. Yan, and X.Zou, "Guiding Genetic Operators with Immunology Principle: A Case Study in TSP," presented at Congress on Evolutionary Computation (CEC 2003), 2003.

[54] D. Dasgupta, "Artificial Immune Systems and their Applications," in *Artificial Immune Systems and Their Applications*: Springer Verlag, 1999.

[55] M. Bakhouya, J.Gaber, and A. Gondron, "An Optimization Algorithm Inspired by the Clonal and the Positive/Negative Selection Mechanism of the Natural Immune System," presented at Metaheuristics (META'06), Hammamet, Tunisia, 2006.

[56] J. S. Chun, J. P. Lim, H. K. Jung, and J. S. Yoon, "Optimal design of synchronous motors with parameter correction using immune algorithm," *IEEE Transactions on Energy Conversion*, vol. 14, pp. 610-615, 1999.

[57] I. Tazawa, S. Koakutsu, and H. Hirata, "An Immunity Based Genetic Algorithm and its Application to the VLSI Floorplan Design Problems," presented at IEEE International Conference on Evolutionary Computation, 1996.

[58] M. Mori, M. Tsukiyama, and T. Fukuda, "Immune Algorithm with Searching Diversity and its Application to Allocation Problems," *Transactions of the Institute of Electrical Engineering of Japan*, vol. 113-C, pp. 872-878, 1993.

[59] J. S. Chun, H. K. Jung, and S. Y. Hahn, "A Study of Optimization Performances Between Immune Algorithm and Other Heuristic Algorithms," *IEEE Transactions on Magnetics*, vol. 34, pp. 2972-2975, 1998.

[60] J. S. Chun, M. K. Kim, H. K. Jung, and S. K. Hong, "Shape Optimization of Electromagnetic Device using Immune Algorithm," *IEEE Transactions on Magnetics*, vol. 33, pp. 1876-1879, 1997.

[61] T. Fukuda, K. Mori, and M. Tsukiyama, "Parallel Search for Multi-Model Function Optimization with Deversity and Learning of Immune Algorithm," in *Artificial Immune Systems and Their Applications*, D. Dasgupta, Ed.: Springer Verlag, 1998, pp. 211-219.

[62] R. Corina, "An Evolutionary Technique for Multicriterial Optimization Based on Endocrine Paradigm," presented at Genetic and Evolutionary Computation Conference (GECCO'04), Seattle, Washington, 2004.I

Appendix C

Results of Experiments Performed on NBGA

The results of the experiments are summarised below.

C.1.1 Test Function 1

The results for the first test function defined in Section 4.5 are summarised in Table C.1

Table C.1a: Experimental values of performance parameters for test function 1 with all mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.0061284	0.0110497	0.01786
Worst	0.0250286	0.1348315	0.04249
Average	0.0152484	0.06	0.03495
Median	0.0143543	0.0569337	0.03637
Std. Deviation	0.0061901	0.0338507	0.00707

Table C.1b: Experimental values of performance parameters for test function 1 without chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.076973	0.018541	0.086829
Worst	0.314360	0.226247	0.206617
Average	0.191520	0.098847	0.169921
Median	0.180290	0.095535	0.176864
Std. Deviation	0.077748	0.056801	0.034391

Table C.1c: Experimental values of performance parameters for test function 1 without Large mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.081447	0.009259	0.060763
Worst	0.388474	0.088710	0.157557
Average	0.197284	0.044053	0.113201
Median	0.178941	0.040239	0.113094
Std. Deviation	0.100128	0.025787	0.032612

Table C.1d: Experimental values of performance parameters for test function 1 without Small mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.123981	0.021429	0.123981
Worst	0.238703	0.147059	0.238703
Average	0.172170	0.059345	0.172170
Median	0.163728	0.044263	0.163728
Std. Deviation	0.040283	0.044126	0.040283

Table C.1e: Experimental values of performance parameters for test function 1 without Large and Chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.107453	0.024000	0.087042
Worst	0.505248	0.131148	0.409273
Average	0.223468	0.085236	0.181019
Median	0.188319	0.093639	0.152547
Std. Deviation	0.115602	0.038825	0.093643

Table C.1f: Experimental values of performance parameters for test function 1 without Small and Chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.105169	0.014358	0.10517
Worst	0.310676	0.093023	0.31068
Average	0.196150	0.038681	0.19615
Median	0.190054	0.033333	0.19005
Std. Deviation	0.062143	0.031107	0.06214

Table C.1g: Experimental values of performance parameters for test function 1 without Small and Large mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.056609	0.018634	0.056609
Worst	0.467372	0.142857	0.467372
Average	0.176223	0.066027	0.176223
Median	0.148774	0.056141	0.148774
Std. Deviation	0.125256	0.041151	0.125256

Table C.1h: Experimental values of performance parameters for test function 1 without any mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.162712	0.000000	0.100152
Worst	0.384013	0.107692	0.236367
Average	0.220684	0.058129	0.135835
Median	0.197426	0.046750	0.121519
Std. Deviation	0.065886	0.039763	0.040554

From the above table it is clear that the best performance of the NBGA is obtained when all the mutation types are activated.

C.1.2 Test Function 2

The results for the second test function are summarised in Table C.2. From the table below it is evident that the performance of the NBGA is best when all mutation operators are engaged.

Table C.2a: Experimental values of performance parameters for test function 2 with all mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.001042	0.020348	0.000963
Worst	0.058807	0.163800	0.054376
Average	0.012415	0.101232	0.011479
Median	0.006169	0.110466	0.005704
Std. Deviation	0.017342	0.040208	0.016036

Table C.2b: Experimental values of performance parameters for test function 2 without chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.087042	0.043478	0.007137
Worst	0.402788	0.350000	0.402788
Average	0.181019	0.216307	0.085032
Median	0.152547	0.236037	0.042253
Std. Deviation	0.118782	0.085914	0.118782

Table C.2c: Experimental values of performance parameters for test function 2 without Large mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.064145	0.073171	0.011158
Worst	0.554539	0.354430	0.128159
Average	0.315458	0.275865	0.048853
Median	0.359667	0.318223	0.041975
Std. Deviation	0.177693	0.089896	0.032078

Table C.2d: Experimental values of performance parameters for test function 2 without Small mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.087042	0.225806	0.021045
Worst	1.069168	0.636364	1.069168
Average	0.181019	0.454252	0.237581
Median	0.152547	0.386364	0.061069
Std. Deviation	0.377243	0.168480	0.377243

Table C.2e: Experimental values of performance parameters for test function 2 without Large and Chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.114225	0.017271	0.012622
Worst	1.297382	0.771350	0.335019
Average	0.360635	0.446554	0.104474
Median	0.221646	0.509777	0.084343
Std. Deviation	0.356276	0.219659	0.094380

Table C.2f: Experimental values of performance parameters for test function 2 without Small and Chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.012430	0.066667	0.012430
Worst	1.335841	0.400000	1.335841
Average	0.192754	0.213556	0.192754
Median	0.029238	0.213821	0.029238
Std. Deviation	0.412924	0.103887	0.412924

Table C.2g: Experimental values of performance parameters for test function 2 without Small and Large mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.010803	0.093220	0.008672
Worst	0.229761	0.325581	0.184448
Average	0.059341	0.202101	0.047638
Median	0.038890	0.226061	0.031220
Std. Deviation	0.065954	0.097568	0.052947

Table C.2h: Experimental values of performance parameters for test function 2 without any mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.056318	0.057471	0.005632
Worst	0.338023	0.277228	0.338023
Average	0.129815	0.123162	0.069713
Median	0.088712	0.122200	0.041063
Std. Deviation	0.092002	0.062070	0.099446

C.1.3 Test Function 3

The results for the third test function are summarised in Table C.3.

Table C.3a: Experimental values of performance parameters for test function 3 with all mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.005542	0.010155	0.011519
Worst	0.028164	0.046329	0.018156
Average	0.010171	0.028615	0.013965
Median	0.006356	0.030161	0.013573
Std. Deviation	0.007376	0.011354	0.002165

Table C.3b: Experimental values of performance parameters for test function 3 without chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.044169	0.034014	0.027184
Worst	0.224457	0.155172	0.042846
Average	0.081062	0.095841	0.032954
Median	0.050652	0.101019	0.032030
Std. Deviation	0.058788	0.038029	0.005109

Table C.3c: Experimental values of performance parameters for test function 3 without Large mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.044691	0.015504	0.035335
Worst	0.093148	0.258065	0.241645
Average	0.061268	0.135715	0.061325
Median	0.062464	0.142857	0.039453
Std. Deviation	0.014012	0.073680	0.063617

Table C.3d: Experimental values of performance parameters for test function 3 without Small mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.011229	0.015038	0.000828
Worst	0.299241	0.080000	0.715964
Average	0.067761	0.040471	0.151429
Median	0.017965	0.037282	0.006117
Std. Deviation	0.096177	0.019762	0.298000

Table C.3e: Experimental values of performance parameters for test function 3 without Large and Chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.008474	0.008000	0.000730
Worst	0.145713	0.148515	0.001378
Average	0.041155	0.045651	0.001106
Median	0.018487	0.035344	0.001107
Std. Deviation	0.055232	0.042118	0.000246

Table C.3f: Experimental values of performance parameters for test function 3 without Small and Chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.007849	0.028986	0.000711
Worst	0.187073	0.241379	0.075598
Average	0.030406	0.076117	0.012564
Median	0.011690	0.067886	0.001123
Std. Deviation	0.055333	0.061065	0.023720

Table C.3g: Experimental values of performance parameters for test function 3 Small and Large mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.009351	0.018018	0.000943
Worst	0.273262	0.225806	0.153115
Average	0.066809	0.051788	0.023786
Median	0.030422	0.035057	0.001238
Std. Deviation	0.08847	0.061907	0.050960

Table C.3h: Experimental values of performance parameters for test function 3 without any mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.008934	0.010101	0.001147
Worst	0.811530	0.368421	0.725834
Average	0.120845	0.098136	0.085323
Median	0.048608	0.057144	0.002334
Std. Deviation	0.245662	0.115493	0.227382

From the above table it can be seen that the NBGA performs best in comparison when all the types of mutation operators are included.

C.1.4 Test Function 4

The results for his test function are summarised in Table C.4.

Table C.4a: Experimental values of performance parameters for test function 4 with all mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.002816	0.011472	0.002360
Worst	0.224425	0.231899	0.198270
Average	0.055564	0.058757	0.044591
Median	0.018163	0.032950	0.007018
Std. Deviation	0.089515	0.068470	0.069140

Table C.4b: Experimental values of performance parameters for test function 4 without chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.010183	0.026549	0.001259
Worst	0.811530	0.536679	0.105783
Average	0.200921	0.135981	0.020499
Median	0.065679	0.076255	0.003657
Std. Deviation	0.323691	0.158458	0.037089

Table C.4c: Experimental values of performance parameters for test function 4 without Large mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.048314	0.0265487	0.00126
Worst	0.3171449	0.5366789	0.10578
Average	0.1324197	0.1359808	0.02050
Median	0.0728534	0.0762553	0.00366
Std. Deviation	0.10806	0.15846	0.03709

Table C.4d: Experimental values of performance parameters for test function 4 without Small mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.048028	0.075188	0.001871
Worst	0.350790	0.183007	0.203832
Average	0.101910	0.141991	0.101278
Median	0.070038	0.150971	0.120175
Std. Deviation	0.091231	0.036893	0.076806

Table C.4e: Experimental values of performance parameters for test function 4 without Large and Chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.038025	0.025641	0.002566
Worst	0.80086	0.165680	0.206318
Average	0.134931	0.124859	0.081911
Median	0.053114	0.152178	0.068889
Std. Deviation	0.23569	0.053692	0.084912

Table C.4f: Experimental values of performance parameters for test function 4 without Small and Chromosome mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.045403	0.027972	0.003666
Worst	0.421778	0.164706	0.158907
Average	0.122842	0.131472	0.101494
Median	0.088802	0.155154	0.128731
Std. Deviation	0.112174	0.054880	0.059716

Table C.4g: Experimental values of performance parameters for test function 4 without Small and Large mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.038670	0.034884	0.001479
Worst	0.287102	0.192308	0.177334
Average	0.143837	0.123554	0.050430
Median	0.076175	0.154696	0.003440
Std. Deviation	0.108723	0.061616	0.077186

Table C.4h: Experimental values of performance parameters for test function 4 without any mutation

	Error Ratio	Gen. Dist.	Spacing
Best	0.041263	0.036364	0.004254
Worst	0.297187	0.194444	0.194298
Average	0.156210	0.157713	0.073749
Median	0.144896	0.164394	0.028736
Std. Deviation	0.099514	0.047936	0.079097

Appendix D

Simple Model of the Permanent Magnet Brushless Direct Current (BLDC) Motor

D.1 Motor Model

In this appendix the model of a permanent magnet brushless DC motor is discussed. The design of BLDC motors is not a simple task. It requires knowledge of magnetics, mechanics, electronics and material science. In this section a simple design methodology for the surface mounted BLDC motor is given [1]. To develop this model certain assumptions have been made.

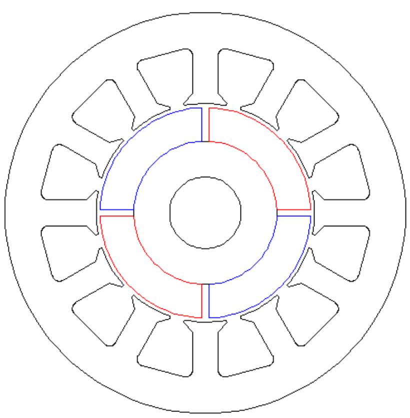


Figure D.1: Typical geometry of surface mount BLDC motor

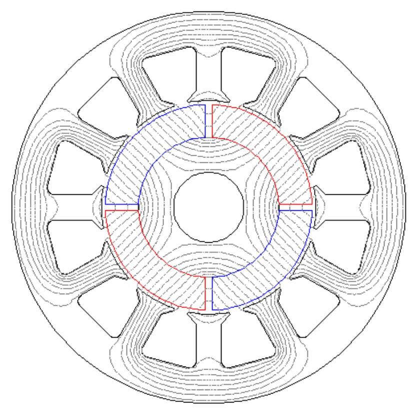


Figure D.2: Flux distribution in a typical surface mount motor

The assumptions made are:

1. No saturation in iron parts

2. Magnets are symmetrically placed
3. Slots are symmetrically placed
4. Back emf is trapezoidal in shape.
5. Motor has balanced windings
6. Permeability of iron is infinite

The general configuration of the motor is shown in Figure D.1. The motor design equations are developed in the following sequence:

1. Electrical Design
2. General Sizing

Inductance and Resistance Calculation

D.2 Electrical Design (Back emf and Torque):

The back emf voltage induced in a stator coil due to magnet flux crossing the air gap is given by

$$e_{ph} = \frac{d\lambda}{dt} = \frac{d\theta}{dt}\frac{d\lambda}{d\theta} = \omega_m \frac{d\lambda}{d\theta} \tag{D.1}$$

where ω_m is the mechanical speed of the rotor (radians/sec) and λ is the flux linked by the coil.

The magnitude of back emf is given by

$$\left|e_{ph}\right| = 2B_g N_{turns} L_{stack} R_{ro} \omega_m \tag{D.2}$$

where N_{turns} is the number of turns in a coil, L_{stack} is the length rotor of the stack, R_{ro} is the outer radius of the B_g is the air gap field density given by

$$B_g = B_r \frac{h_m}{h_m + g} \tag{D.3}$$

where h_m is the height of the magnet, g is the airgap and the outer radius of the rotor R_{ro} is given by

$$R_{ro} = \alpha_d R_o \tag{D.4}$$

where the R_o is outer radius of the stator and α_d. is the ratio of outer diameter of the rotor to the outer diameter of the stator.

For fractional pitched magnets the coil back emf and coil torque is given by

$$e_{ph} = 2\alpha_m B_g N_{turns} L_{stack} R_{ro} \omega_m \tag{D.5}$$

$$T_{coil} = 2\alpha_m N_{turns} B_g L_{stack} R_{ro} i \tag{D.6}$$

where α_m is the ratio of the magnet angle to pole pitch and i is the phase current.

D.3 General Sizing:

If the alternating direction of flux flow over alternating magnet faces is ignored, the total flux crossing the air gap is given by

$$\phi_{total} = B_g A_g = 2\pi B_g R_{ro} L_{stack} \tag{D.7}$$

where B_g is the amplitude of the air gap flux density and is given by equation D.3 and A_g is area of the airgap. This flux is divided among the teeth on the stator and the direction of the flux depends on the polarity of the magnet under each tooth. As a result, the magnitude of the flux flowing in each tooth is given by

$$\phi_t = \frac{\phi_{total}}{N_S} = \frac{2\pi B_g R_{ro} L_{stack}}{N_S} \tag{D.8}$$

where N_S is the number of slots on the stator.

This flux travels through the body of the tooth resulting in a flux density B_t whose magnitude is given by

$$B_t = \frac{\phi_t}{w_{tb} L_{stack}} \tag{D.9}$$

where w_{tb} is the width of the tooth.

The value of B_t is generally known, as it is the maximum allowable flux density in iron. Hence once the value of B_t is determined the width of the tooth is given from equation D.8 and equation D.9

$$w_{tb} = \frac{2\pi R_{ro} B_g}{N_s B_t} \tag{D.10}$$

From the above expression it can be seen that the tooth width is directly dependent on the rotor outer radius and inversely proportional to number of slots. As the number of slot increase, the width of the tooth decreases. The width of the tooth is independent of the number of poles (magnets) because the total flux crossing the air gap is not a function of the number of poles. Hence w_{tb} does not vary with the change in the number of poles.

The flux from each magnet splits into two halves, with each half forming a flux loop, Figure D.2. The stator flux density is B_{sy} given by

$$B_{sy} = \frac{\phi_{total} / 2}{w_{sy} L_{stack}} \tag{D.11}$$

where w_{sy} is the width of the yoke and is given by

$$w_{sy} = \frac{\pi R_{ro} B_g}{N_m B_{sy}} \tag{D.12}$$

B_{sy} is again the maximum permissible flux density in iron. The above expression shows that width of the yoke is directly proportional to the outer radius of the rotor R_{ro} and inversely proportional to the number of magnets. The stator yoke width is independent of number of slots. The general shape of the slot considered here is shown in Figure D.3.

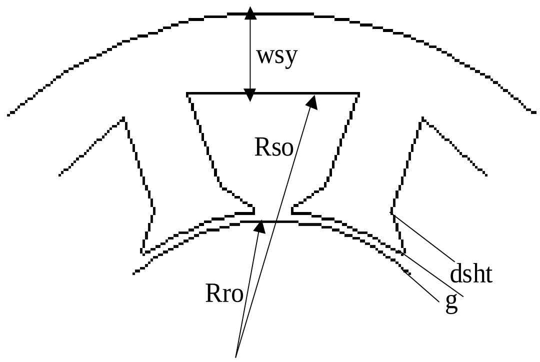


Figure D.3: The general shape of the slots and main dimensions

The area of the above slot is given by

$$A_s = \frac{\pi}{N_s}\left[(R_{so} - w_{sy})^2 - (R_{ro} + g + d_{sht})^2\right] - w_{tb}(R_{so} - w_{sy} - R_{ro} - g - d_{sht}) \quad (D.13)$$

The area of the copper in the slot is given by

$$A_{cu} = K_{cu} A_s \quad (D.14)$$

where k_{cu} is the copper fill factor.

With this the calculation of the main dimensions of the motor is done.

D.4 Inductance and Resistance Calculation:

The total phase inductance L_{ph} composed of air gap inductance L_g, slot leakage inductance L_s, and end turn inductance L_e is given [1]

$$L_{ph} = N_s / 3(L_s + L_e + L_g) \quad (D.15)$$

where

$$L_g = \frac{N_{turns}^2 \mu_r \mu_o L_{stack} \tau_c k_d}{4(L_{stack} + \mu_r k_c g)} \quad \text{(D.16)}$$

$$L_s = N_{turns}^2 \left[\frac{\mu_0 d_3^2 L_{stack}}{3A_s} + \frac{\mu_0 d_2 L_{stack}}{(w_s + w_{si})/2} + \frac{\mu_0 d_1 L_{stack}}{w_s} \right] \quad \text{(D.17)}$$

$$L_e = \frac{\mu_0 N_{turn}^2 \tau_c}{8} \ln \left[\frac{\tau_c^2 \pi}{4A_s} \right] \quad \text{(D.18)}$$

where μ_0 is the permeability of free space.

The parameters of the motor are shown in Fig.B.3. Since each slot has two coil sides and each coil has N_{turns} turns, the resistance R_{slot} per slot is given by

$$R_{slot} = \frac{\rho L_{stack} 4N_{turns}^2}{K_{cu} A_{slot}} \quad \text{(D.19)}$$

A three phase star connected motor is considered. Hence the phase resistance of the motor is given by

$$R_{ph} = \frac{N_s}{3} R_{slot} \quad \text{(D.20)}$$

D.5 Loss Calculation:

The Ohmic P_r and the core loss P_{Fe} can be determined from the following relation:

$$P_r = 3I_{ph}^2 R_{ph} \quad \text{(D.21)}$$

$$P_{Fe} = \rho_{bi} V_{st} \Gamma(f_e, B_{Fe}) \quad \text{(D.22)}$$

where I_{ph} is the RMS value of the driving current for each phase, R_{ph} is the resistance value for each phase, ρ_{bi} is the mass density of the back iron, V_{st} is the stator volume, and Γ is the core loss density of the stator material at the flux density B_{Fe} and frequency f_e.

D.6 References

[1] D. C. Hanselman, *Brushless Permanent Magnet Motor Design*, 2nd ed: The Writers' Collective, 2005.

Propositions

1. A good approach to design a complex engineering system is to start with the simplified model of the concerned system. Development of a simplified model of a complex system is more of an art than exact science.
2. As a result of evolution there are various species in nature and each of them is well adapted to their specific niches. Hence, evolutionary algorithms are best suitable for multiobjective optimisation problems because they mimic the evolutionary process as it occurs in nature.
3. When an engineering artefact is to be designed, it is important to look at it from the perspective of the system of which it will be a part. An artefact designed in isolation will seldom work efficiently when integrated into the system.
4. Designs of engineering systems based on the biological systems will be more efficient and in harmony with the environment because nature has already found optimal solution to most of the engineering problems that we face.
5. No engineering design is completely objective. It is possible that two engineers with equivalent skill might come up with different but equally valid designs, depending on the degree of design freedom.
6. Multiobjective optimisation problems are in principle problems that involve making compromises. A good solution, under a given situation, is one that makes a balanced compromise between conflicting requirements.
7. String theory is often referred as the first candidate of Theory of Everything. This claim appears to be bold because we do not know everything about the universe.
8. The claim that climate change is human made and is disastrous for human species is disputable. The basic flaw in this claim is that we consider us to be separate from the nature. In reality we are very much a part of the nature and any change in the environment due to our activity is a natural phenomenon.
9. Throughout the history of mankind the conventional notion of god and religion has been the root cause, in one way or the other, of atrocities committed by one human against the other.
10. The laws of nature fit the conventional definition of god perfectly because just as god they are neither benign nor harmful, they are omnipotent, omnipresent and are elusive to normal human mind. The best religion is continuous pursuit to understand these laws of nature and to put them to good use.