

DEVELOPMENT OF AN ELECTRICAL MACHINES ANALYSIS AND OPTIMUM
DESIGN SOFTWARE PACKAGE

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

YILMAZ GÖYNÜK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JUNE 2008

Approval of the thesis:

**DEVELOPMENT OF AN ELECTRICAL MACHINES ANALYSIS AND OPTIMUM DESIGN
SOFTWARE PACKAGE**

submitted by **YILMAZ GÖYNÜK** in partial fulfillment of the requirements for the degree of
**Master of Science in Electrical and Electronics Engineering Department, Middle
East Technical University** by,

Prof. Dr. Canan Özgen

Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. İsmet Erkmen

Head of the Department, **Electrical and Electronics Engineering**

Prof. Dr. H. Bülent Ertan

Supervisor, **Electrical and Electronics Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Muammer Ermiş

Electrical and Electronics Engineering Dept., METU

Prof. Dr. H. Bülent Ertan

Electrical and Electronics Engineering Dept., METU

Prof. Dr. Kemal Leblebicioğlu

Electrical and Electronics Engineering Dept., METU

Assist. Prof. Dr. Cüneyt Bazlamaççı

Electrical and Electronics Engineering Dept., METU

Assist. Prof. Dr. Timur Aydemir

Electrical and Electronics Engineering Dept., Gazi University

Date: 20 June 2008

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Yılmaz GÖYNÜK

Signature :

ABSTRACT

DEVELOPMENT OF AN ELECTRICAL MACHINES ANALYSIS AND OPTIMUM DESIGN SOFTWARE PACKAGE

Göynük, Yılmaz

M.Sc., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. H. Bülent Ertan

June 2008, 174 pages

In this study, three different programs are developed for the analysis of the three-phase induction motor, single-phase capacitor type induction motor and switched reluctance motor. The programs are developed by using Pascal and C++ programming languages. In the performance calculations of motors, analytical methods are used and these methods are tested for accuracy. These programs have also capabilities to design an optimum motor, which meets a set of performance, material and manufacturing constraints while minimizing the weight or any other defined objective function.

In addition, in this study, an optimization tool is used to obtain an appropriate optimization method for the design of different types of motors.

The software is tested over different commercial motors. The results illustrates that the performance calculations and optimization approach of the programs lead to good results.

Keywords: Switched Reluctance Motor, Single-Phase Induction Motor, Three-Phase Induction Motor, Optimum Design, Computer Aided Design.

ÖZ

ELEKTRİK MAKİNALARININ ANALİZ VE OPTİMUM TASARIMI İÇİN YAZILIM PAKETİ GELİŞTİRİLMESİ

Göynük, Yılmaz

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. H. Bülent Ertan

Haziran 2008, 174 sayfa

Bu çalışmada, üç faz ve tek faz endüksiyon motorları ile anahtarlamalı relüktans motorlarının analizi için üç farklı program geliştirilmiştir. Programlar Pascal ve C++ programlama dilleri kullanılarak yazılmıştır. Motorların performans hesaplamalarında analitik metotlar kullanılmış ve bu metotların güvenilirliği test edilmiştir. Ayrıca, bu programların, ağırlık veya herhangi bir tanımlı hedef fonksiyonunu en aza indirgeyerek, bir takım performans, malzeme ve üretim kısıtlamalarını karşılayan optimum motoru tasarlama kabiliyetleri vardır.

Bu çalışmada ayrıca, farklı tip motorların tasarımı için uygun eniyileme metodu elde etmek için, bir eniyileme aracı kullanılmıştır.

Geliştirilen yazılım farklı ticari motorlar üzerinde denenmiştir. Sonuçlar, programların performans hesaplamalarının ve eniyileme algoritmalarının iyi sonuçlar verdiğini göstermektedir.

Keywords: anahtarlamaalı relüktans motoru, tek faz endüksiyon motoru, üç faz endüksiyon motoru, en iyi tasarım, bilgisayar destekli tasarım.

To My Wife

ACKNOWLEDGEMENTS

The author wishes to express his deepest gratitude to his supervisor Prof. Dr. Bülent Ertan for his guidance, advice, criticism, encouragements and insight throughout the research.

The author would also thank his family for their precious support during all stages of his life.

TABLE OF CONTENTS

ABSTRACT.....	iv
ÖZ	vi
ACKNOWLEDGEMENTS	ix
TABLE OF CONTENTS.....	x
LIST OF TABLES	xiv
LIST OF FIGURES	xv
CHAPTER	
1. INTRODUCTION	1
1.1. Introduction	1
2. THREE PHASE INDUCTION MACHINE COMPUTER AIDED DESIGN (TPCAD)	4
2.1. Introduction	4
2.2. Technical Explanation of TPCAD.....	4
2.2.1. <i>Analysis of Performance in TPCAD</i>	6
2.2.1.1. Full Load Current Calculation	7
2.2.1.2. Full Load Torque Calculation	7
2.2.1.3. Full Load Output Power Calculation.....	8
2.2.1.4. Full Load Efficiency Calculation.....	8
2.2.1.5. Full Load Power Factor Calculation.....	9
2.2.1.6. Starting Performance and Torque Speed Variation Calculations	9
2.3. Documentation of TPCAD	11
2.4. GUI of TPCAD	12
2.4.1. <i>Introduction</i>	12
2.4.2. <i>Using the TPCAD Program</i>	12
2.4.2.1. File Menu	13
2.4.2.2. Curve Menu.....	15
2.4.2.2.1. B-H Curve Form	16
2.4.2.3. Analysis Menu	17
2.4.2.3.1. Graphical Analysis Form.....	17
2.4.2.3.2. At a Power Analysis Form.....	19
2.4.2.3.3. At a Speed Analysis Form	20

2.4.2.3.4.	Summary Analysis Form.....	21
2.4.2.4.	Optimization Menu.....	21
2.5.	Results of TPCAD	24
2.5.1.	Analysis Result	24
2.5.2.	Optimization Result.....	26
2.6.	Optimization of Three Phase Induction Machine with modeFRONTIER	28
2.6.1.	Results of Simplex Algorithm	30
2.6.2.	Results of NLPQLP	32
2.6.3.	Conclusion Of Optimization	34
2.7.	Conclusion of TPCAD	35
3.	SINGLE PHASE CAPACITOR TYPE INDUCTION MACHINE COMPUTER AIDED DESIGN (SPCAD).....	36
3.1.	Technical Explanation of SPCAD.....	36
3.2.	Documentation of SPCAD	38
3.3.	GUI of SPCAD.....	38
3.3.1.	Introduction.....	38
3.3.2.	Using the SPCAD Program	39
3.3.2.1.	File Menu	39
3.3.2.2.	Curve Menu.....	41
3.3.2.2.1.	B-H Curve Form	42
3.3.2.3.	Analysis Menu	43
3.3.2.3.1.	Graphical Analysis Form.....	43
3.3.2.3.2.	At a Power Analysis Form.....	45
3.3.2.3.3.	At a Speed Analysis Form.....	46
3.3.2.3.4.	Summary Analysis Form.....	47
3.3.2.4.	Optimization Menu.....	47
3.4.	Results of SPCAD	50
3.4.1.	Analysis Result	50
3.4.2.	Optimization Result.....	53
3.4.2.1.	Results of NLPQLP.....	54
3.4.3.	Conclusion Of Optimization	62
3.5.	Conclusion of SPCAD.....	62
4.	SWITCHED RELUCTANCE MOTOR COMPUTER AIDED DESIGN (SRCAD).....	63
4.1.	Introduction	63
4.2.	Basic Structure of the Switched Reluctance Motor	64
4.3.	Technical Explanation of SRCAD.....	65

4.3.1.	<i>The Analysis Section of SRCAD.....</i>	67
4.4.	<i>Determination of Flux Linkage – Position – Current Curves.....</i>	69
4.4.1.	<i>Teeth Region Flux-Linkage MMF Characteristics</i>	71
4.4.2.	<i>Determination of Back Iron Flux Linkage MMF Characteristic.....</i>	73
4.4.3.	<i>Determination of Pole MMF Drop.....</i>	74
4.4.4.	<i>Determination of Back Core MMF Drop</i>	76
4.4.5.	<i>Determination of the Operating Point.....</i>	78
4.4.6.	<i>End Leakage Correction.....</i>	80
4.5.	<i>Determination of Static Torque – Position – Current Curves.....</i>	89
4.6.	<i>Calculation of Steady – State Performance of an SRM</i>	93
4.6.1.	<i>Calculation of Instantaneous Torque Waveform</i>	101
4.6.2.	<i>Calculation of Steady – State Average Torque and Torque – Ripple</i>	103
4.7.	<i>The Accuracy of the Analysis Routines</i>	104
4.7.1.	<i>The accuracy of Prediction of Flux Linkage Characteristics.....</i>	106
4.7.1.1.	<i>Predicted Flux Linkage Curves Without End Leakage Correction.....</i>	106
4.7.1.2.	<i>Predicted Flux Linkage Curves With End leakage Correction</i>	110
4.7.2.	<i>The accuracy of Torque – Speed Characteristics</i>	114
4.7.2.1.	<i>Torque – Speed Curves Without End Leakage Correction</i>	114
4.7.2.2.	<i>Torque – Speed Curves with End Leakage Correction</i>	117
4.7.3.	<i>The Accuracy of Phase Current Characteristics.....</i>	120
4.7.4.	<i>The Accuracy of Losses for SR2</i>	123
4.8.	<i>Optimization</i>	127
4.8.1.	<i>Introduction.....</i>	127
4.8.2.	<i>Independent Optimization Parameters</i>	128
4.8.3.	<i>Constraints</i>	130
4.8.3.1.	<i>Equality Constraint</i>	130
4.8.3.2.	<i>Inequality Constraints.....</i>	130
4.8.3.2.1.	<i>Inner Diameter of the Rotor.....</i>	130
4.8.3.2.2.	<i>Air Gap Length.....</i>	131
4.8.3.2.3.	<i>Advance Angle.....</i>	131
4.8.3.2.4.	<i>Current Density</i>	131
4.8.3.2.5.	<i>Efficiency.....</i>	132
4.8.4.	<i>The Object Function: Weight of SRM</i>	132
4.8.4.1.	<i>Calculation of Iron Weight.....</i>	132
4.8.4.2.	<i>Calculation of Copper Weight.....</i>	133
4.8.5.	<i>Optimization Project at modeFRONTIER</i>	134
4.8.5.1.	<i>Introduction</i>	134
4.8.5.2.	<i>Design Steps of modeFRONTIER Project</i>	137

4.8.6.	<i>Optimization Result</i>	144
4.8.6.1.	Results of Simplex Algorithm.....	144
4.8.6.2.	Results of MOSA Algorithm	149
4.8.7.	<i>Optimization Conclusion</i>	152
4.9.	GUI of SRCAD	153
4.9.1.	<i>Introduction</i>	153
4.9.2.	<i>Using the SRCAD Program</i>	153
4.9.2.1.	Optimize Button	153
4.9.2.2.	Using modeFRONTIER Design.....	154
4.10.	Conclusion of SRCAD	156
5.	CONCLUSION	157
	REFERENCES	160
	APPENDICES	
A.	ANALYSIS DATA OF TEST MOTORS	163
B.	OPTIMIZATION ALGORITHMS IN modeFRONTIER	169

LIST OF TABLES

Table 2.1 Test Results of TPCAD with Different Motors.....	25
Table 2.2 Error Percentages of the TPCAD	25
Table 2.3 Initial and Final Design of QU100L4A.....	27
Table 2.4 Results of Optimization of QU100L4A.....	27
Table 2.5 Initial and Final Design of QU100L4A by using Simplex	31
Table 2.6 Results of Optimization of QU100L4A.....	32
Table 2.7 Initial and Final Design of QU100L4A by using NLPQLP	33
Table 2.8 Results of Optimization of QU100L4A by NLPQLP	34
Table 3.1 Test Results of SPCAD with Different Motors.....	51
Table 3.2 Error Percentages of the SPCAD	51
Table 3.3 Initial and Final Designs of 187 W,4 pole single phase induction motor by using NLPQLP ..	56
Table 3.4 Results of Optimization of 187 W,4 pole single phase induction motor by NLPQLP	57
Table 4.1 Measured Physical Dimensions of SR2.....	105
Table 4.2 Numerical Values and % Errors Between Measured Curves and Predicted Curves without End Leakage Correction	109
Table 4.3 Numerical Values and % Errors Between Measured Curves and Predicted Curves with End Leakage Correction	113
Table 4.4 Predicted-Measured Torques and Percentage Errors without End-Leakage Correction ...	117
Table 4.5 Predicted-Measured Torques and Percentage Errors with End-Leakage Correction.....	120
Table 4.6 Rise Times and Fall Times of Calculated and Measured Current Waveforms	123
Table 4.7 Comparison of Measured and Predicted Losses of SR2 for Load Torques at 0.5Nm and 1Nm	124
Table 4.8 Comparison of Measured and Predicted Efficiencies of SR2 for Load Torques at 0.5 Nm and 1Nm	125
Table 4.9 Initial and Final Design of SR2 by using Simplex Algorithm	145
Table 4.10 Initial and Final Design of SR2 by using MOSA Algorithm	150

LIST OF FIGURES

Figure 2.1 Visual Model of TPCAD Classes	6
Figure 2.2 The Equivalent Circuit of Three Phase Induction Motor	7
Figure 2.3 Stator slot shapes	10
Figure 2.4 Rotor slot shapes.....	10
Figure 2.5 The Flowchart of Pull-Out Torque Calculation	11
Figure 2.6 Main Form of TPCAD	13
Figure 2.7 Motor Parameters Form	15
Figure 2.8 B-H Curve Form	17
Figure 2.9 Graphical Analysis of the Motor (Torque vs. Speed)	18
Figure 2.10 Analysis of the Motor at a Specific Power.....	19
Figure 2.11 Analysis of the Motor at a Specific Speed	20
Figure 2.12 Summary of the Analysis.....	21
Figure 2.13 Optimization Form.....	22
Figure 2.14 Performance Output of the Optimization Process	23
Figure 2.15 Designed Project at modeFRONTIER for the Three Phase Induction Machine.....	29
Figure 3.1 Visual Model of SPCAD Classes.....	38
Figure 3.2 Main Form of SPCAD.....	40
Figure 3.3 Motor Parameters Form	41
Figure 3.4 B-H Curve Form	43
Figure 3.5 Graphical Analysis of the Motor (Torque vs. Speed)	44
Figure 3.6 Analysis of the Motor at a Specific Power.....	45
Figure 3.7 Analysis of the Motor at a Specific Speed	46
Figure 3.8 Summary of the Analysis.....	47
Figure 3.9 Optimization Form.....	49
Figure 3.10 Performance Output of the Optimization Process	50
Figure 3.11 Error Percentage Graph of M1	52
Figure 3.12 Error Percentage Graph of M2	52
Figure 3.13 Designed Project at modeFRONTIER for the Single Phase Induction Machine	54
Figure 4.1 Basic Structure for Four Phase SRM	64
Figure 4.2 Asymmetric Converter for One Phase of SRM [13]	69
Figure 4.3 Operation of Converter on Phase Current Waveform [13]	69
Figure 4.4 Asymmetrical Structure and Related Symmetrical Structures	72
Figure 4.5 Back Iron Regions of SR motor	73
Figure 4.6 Stator Tooth	74

Figure 4.7 Cross Section of a 8/6 Switched Reluctance Motor	76
Figure 4.8 Rotor	78
Figure 4.9 Operating Point Determination	79
Figure 4.10 Flux Linkage - Position - Current Curves.....	80
Figure 4.11 a)Mirror Image to Obtain an Equivalent Geometry Including End Leakage Flux.....	81
Figure 4.12 Main Flowchart of Determination of Flux Linkage Current Position.....	83
Figure 4.13 BOX 1 of Figure 4.12, Determination of Permeance vs. B Curve for Geometry A	84
Figure 4.14 BOX 2 of Figure 4.12, Calculating Back Iron MMF Drop.....	85
Figure 4.15 BOX 2.1 of Figure 4.14, Finding Stator Tooth MMF Drop vs. Flux Linkage Curve	86
Figure 4.16 BOX 2.2 of Figure 4.14, Finding Stator Backcore MMF Drop vs. Flux Linkage	87
Figure 4.17 BOX 3 of Figure 4.12, Finding the Operating Point Flux Linkage vs. MMF Curve.....	88
Figure 4.18 Torque - Position – Current Curves.....	90
Figure 4.19 Flowchart of Torque vs. Position vs. Current	91
Figure 4.20 Calculation of Force vs. MMF for Geometry A	92
Figure 4.21 Equivalent Circuit of Single Phase of an SRM	93
Figure 4.22 Static Torque Curves.....	96
Figure 4.23 Applying Runge Kutta 4 th Order Method.....	98
Figure 4.24 Flowchart of the Solution Algorithm.....	100
Figure 4.25 Resultant Instantaneous Torque – Position Waveform of an 8-6 SRM Obtained at 2500 rpm with 7.5° Advance Angle	101
Figure 4.26 Determination of Instantaneous Co-Energy, and Hence the Torque [13].....	103
Figure 4.27 Determination of Torque Ripple.....	104
Figure 4.28 Measured Curves and Predicted Curves without End Leakage Correction	108
Figure 4.29 Measured Curves and Predicted Curves with End Leakage Correction	112
Figure 4.30 Torque-Speed Curves without End-Leakage at AA = 0°, 4°, 7.5° and 11° respectively....	116
Figure 4.31 Torque-Speed Curves with End-Leakage at AA = 0°, 4°, 7.5° and 11° respectively.....	119
Figure 4.32 a) Measured Current Waveform at 1000 rpm, AA=0°	121
Figure 4.33 a) Measured Current Waveform at 1600 rpm, AA=0°	121
Figure 4.34 a) Measured Current Waveform at 2000 rpm, AA=4°	122
Figure 4.35 a) Measured Current Waveform at 1000 rpm, AA=11°.....	122
Figure 4.36 Friction Torque vs. rpm Curve	124
Figure 4.37 Measured and Predicted Losses for SR2	125
Figure 4.38 Measured and Predicted Efficiencies for SR2.....	126
Figure 4.39 Independent Physical Parameters	129
Figure 4.40 Winding Arrangement [7]	134
Figure 4.41 Process Integration and Optimization of modeFRONTIER in General.....	135
Figure 4.42 Designed Project at modeFRONTIER	137

Figure 4.43 Input Variables	138
Figure 4.44 Input File Editor	139
Figure 4.45 Application Program	140
Figure 4.46 Output Variables.....	141
Figure 4.47 Constraints	142
Figure 4.48 Optimization Algorithm	143
Figure 4.49 DOE Properties	143
Figure 4.50 Main Form of SRCAD.....	154
Figure 4.51 modeFRONTIER Design Form	155
Figure 4.52 Design Table	155
Figure B.1 Original Simplex (A,B,C) in a 2-dimension.....	170
Figure B.2 Reflected Simplex (A,B,C) with the Reflected Point (D)	171
Figure B.3 Simplex Expansion (A,B,D) with the New Point (E).....	172
Figure B.4 Contracted Simplex (A,B,C) with the Contracted Point (E).	173
Figure B.5 The Tolerance Parameter k Transforms the Constraint Penalization into a Fuzzy Function $g(x)$	174

CHAPTER 1

INTRODUCTION

1.1. Introduction

Before 1960's, the design of motors was done manually without the assistance of any software program. Following that, manual procedures were transferred to software, which could run on early computers. In time, the possibilities the computers offered grew. Early software was developed within important motor producers. Academia was involved in developing sophisticated performance computation methods, which could be implemented using computers. Optimization techniques were also developed, which enabled the designer to seek a better design. Nowadays, due to the advances in computer and software technology a number of software is commercially available. Many others are developed in academia and companies but are not widely available. In this manner, the design office can respond to the needs of the customers more quickly and is able to adopt itself to the rapidly changing production scene in industrial motor design area.

In this context, design software was developed at METU Electrical and Electronics Engineering Department in 1980's. TPCAD (Three-Phase Induction Machine Computer Aided Design) and SPCAD (Single Phase Induction Machine Computer Aided Design) are good examples of software tools developed for the purpose of design and optimization of three-phase and single-phase induction motors respectively [11] [12]. A number of companies in Turkey utilized this software to update their products. Their optimization feature makes them unique in the field of design software. However, both TPCAD and SPCAD run on DOS environment, so they have become obsolete.

In this study, these two software packages are investigated and are rewritten in Microsoft Visual Studio .NET environment using C++ programming language and new Graphical User Interfaces (GUI) are designed to run on a WINDOWS environment. Furthermore, the possibility of improving the optimization features of these software is also investigated. Initial software developed at METU use application specific optimization algorithms. In SPCAD, developed by Aftahi in 1993, "External Penalty Method" is used for optimization of single-phase capacitor type induction motor, but in this study "Augmented Lagrangian Method" is applied in the optimization phase. On the other hand, the optimization of TPCAD uses the "Augmented Lagrangian Method".

Recently general-purpose optimization software called modeFRONTIER became available. This software provides a medium adaptable to any problem. It offers the possibility of using many modern optimization techniques including Genetic Algorithm. modeFRONTIER, which is a multidisciplinary and multi-objective optimization and design environment allows product engineers and designers to integrate their various computer aided engineering (CAE) tools, such as computer aided design (CAD), Finite Element Structural Analysis and Computational Fluid Dynamics (CFD) software.

It is of interest to academia and industry to find out whether any of the optimization methods available in modeFRONTIER package has an advantage in design optimization of electrical motors. In this context, comparison of the performance of various optimization techniques would be of interest to motor designers and would contribute to scientific knowledge. This was one of the tasks undertaken in this thesis. This study is carried out on TPCAD and SPCAD.

A novel design approach was developed at METU for the design of Switched Reluctance motors. However, a user interface was not available for this software. In this thesis, a software based on the design technique developed at METU is developed and tested. This software is called SRCAD; it can design and optimize a switched reluctance motor to design specifications. In SRCAD "Augmented Lagrangian Method" with "Davidon-Fletcher and Powell" minimization technique is employed for the optimization algorithm as this method was already experimented within other thesis work. Here, modeFRONTIER package optimization algorithms are tested to find out their suitability for designing SR motors.

As the result of this study, a software package is developed which can design and optimize single-phase or three-phase induction motors as well as switched reluctance motors. As far as the author knows, the software developed is unique and exclusive in the motor design software tool world.

CHAPTER 2

THREE PHASE INDUCTION MACHINE COMPUTER AIDED DESIGN (TPCAD)

2.1. Introduction

In this study, a software program is developed for the design and optimization of three phase induction motors. For the analysis and performance of the motors the equations in the reference [11] is used. The developed program is tested with three commercial motors and analysis results are compared with the measured performance of the motors. Also, program is used to design a three phase induction machine and optimization results are given in the next sections.

A project is formed by using modeFRONTIER optimization tool and different optimization methods are attempted. And the best optimization method for the three phase induction motors is sought and also compared with the TPCAD`s optimization results.

2.2. Technical Explanation of TPCAD

TPCAD (Three Phase Induction Machine Computer Aided Design) is a personal computer (PC) windows application software product designed for design and optimization of three-phase induction motors. TPCAD is also developed to perform the following items:

- Performance calculation (see Section 2.2.1 as to what is meant by performance) of the user defined parameters

- Analysis of Three Phase Squirrel Cage Induction motors
 - At a Speed
 - At a Power
 - Graphical
- Loss and Temperature calculation at any speed
- Design optimization for cost or weight minimization while satisfying a set of design criteria. Allows the user up to 10 variables to change within the set of design criteria. User can design a specified motor, starting with an initial design.
- Ability to change the core-loss curve and B-H curve.
- Plotting and printing of outputs.

With the above characteristics of TPCAD, effect of materials used in manufacturing the motor on the performance can be investigated. Three slot shapes for stator and five slot shapes for rotor are defined and also by changing the rotor and stator slot shapes motor performance can be observed with different slot shapes.

TPCAD, while satisfying these items, also has a user friendly GUI (Graphical User Interface). To perform this TPCAD is developed in Microsoft Visual Studio .Net Environment using the C++ object oriented language. The Microsoft Visual Studio .Net environment helps the programmer to develop user-friendly interfaces in a fast way by supplying all necessary design tools such as Windows forms, buttons, graphic plotters, list view. C++ programming language is one of the most powerful and easy compiler, with its object oriented programming logic, existing in the market.

To develop the TPCAD the software modulation program UML (Unified Modelling Language) 2.1 is used. The visual model of the software is illustrated in Figure 2.1.

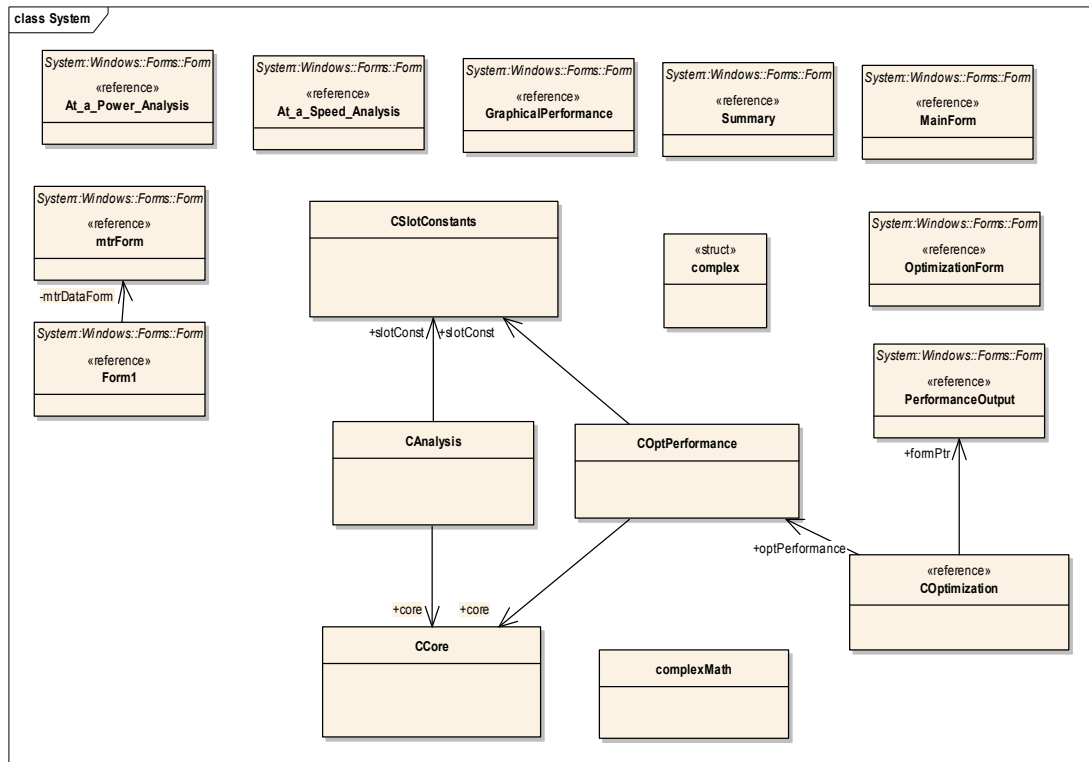


Figure 2.1 Visual Model of TPCAD Classes

2.2.1. Analysis of Performance in TPCAD

Finding out full load current, full load torque, full load output power, full load efficiency, full load power factor, iron losses, friction and windage losses, starting performance and torque-speed variation of the motor gives sufficient information to analyze the performance of a three-phase induction motor. In this study, methods and approaches used are based on the studies of Hamarat [11].

For the performance calculations of the motor, the equivalent circuit model shown in Figure 2.2 is used. The following sections describe the calculation of the performance of the three-phase induction motor generally.

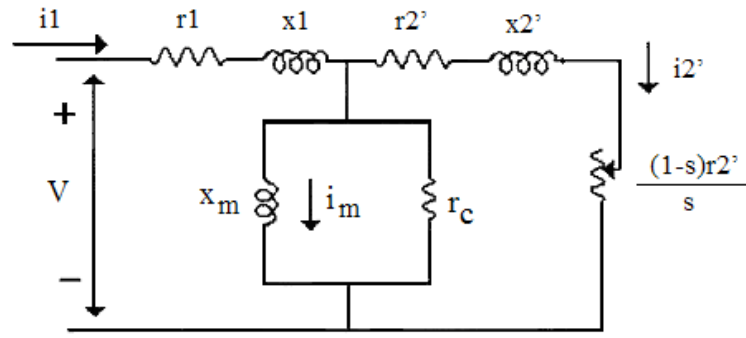


Figure 2.2 The Equivalent Circuit of Three Phase Induction Motor

2.2.1.1. Full Load Current Calculation

By using the equation below, the full load current is calculated.

$$I_1 = \frac{V}{Z_{th}} \quad (2.1)$$

The value of the full load current highly depends on the equivalent impedance of the circuit, Z_{th} . This equivalent impedance is dependent on reactances, which are dependent on the slot shapes of the motor. The calculation of the reactances according to slot shapes are given in Hamarat [11].

2.2.1.2. Full Load Torque Calculation

By using equivalent circuit, full load torque on the shaft of the motor is calculated from the equation below:

$$T = \frac{(9.567) \cdot m \cdot i_2'^2 \cdot r_2'}{n_s \cdot s} \quad (2.2)$$

m: number of phases
 i_2' : referred secondary current
 n_s : synchronous speed
s: slip

2.2.1.3. Full Load Output Power Calculation

The output power is actually the power dissipated on the resistance $r_2' (1-s) / s$ in Figure 2.2. The net output power is calculated by subtracting the friction and windage losses from this dissipated resistance power mentioned above.

$$P_{out} = m \cdot i_2'^2 \cdot \frac{r_2' \cdot (1-s)}{s} - \text{friction losses} - \text{windage losses} \quad (2.3)$$

Friction and windage losses are found from the no load tests.

2.2.1.4. Full Load Efficiency Calculation

The efficiency is calculated by the following equations:

$$\eta = \frac{P_{out}}{P_{in}} \quad (2.4)$$

$$P_{in} = P_{out} + \text{friction} + \text{windage} + P_{\text{Stator Cu Losses}} + P_{\text{Total Fe Losses}} \quad (2.5)$$

$$P_{\text{Stator Cu Losses}} = i_1'^2 \cdot r_1' \cdot m \quad (2.6)$$

Iron losses are divided into two categories; fundamental frequency losses and high frequency losses. Since the fundamental frequency losses are due

to the stator, the high frequency losses originate from the mechanical power developed by the rotor.

Calculation of total iron losses is explained in details in Section 2.2.7 of Hamarat [11].

2.2.1.5. Full Load Power Factor Calculation

The ratio of the equivalent resistance to impedance gives the power factor of the motor:

$$PF = \frac{R_{th}}{Z_{th}} \quad (2.7)$$

2.2.1.6. Starting Performance and Torque Speed Variation Calculations

At the starting, magnetic circuits tends to saturate due to high current, and the infinitely permeability of the magnetic circuit does not hold anymore.

In Figure 2.3 and Figure 2.4, the stator and rotor slot shapes are displayed which is used in TPCAD. For the open slot shapes, the effect of saturation can be ignored. However, for the closed slot shapes, the permeability of the iron cannot be assumed infinite. Hence slot constant become a function of saturation. Therefore, reactances are re-calculated for the starting conditions by considering the saturation effects[11]. The calculation of pull-out torque is as in Figure 2.5. The intervals of slip is taken as 0.01.

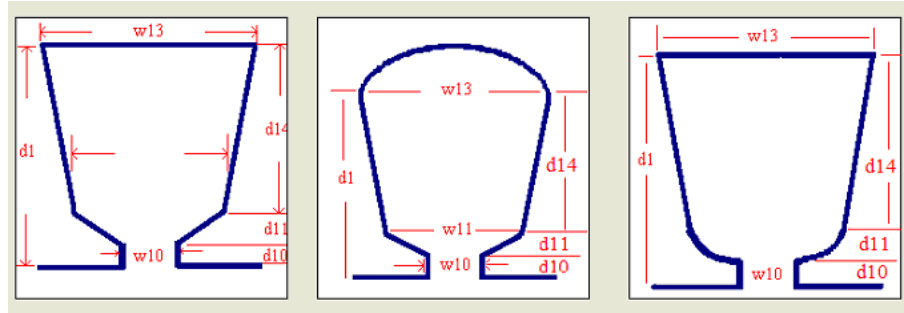


Figure 2.3 Stator slot shapes

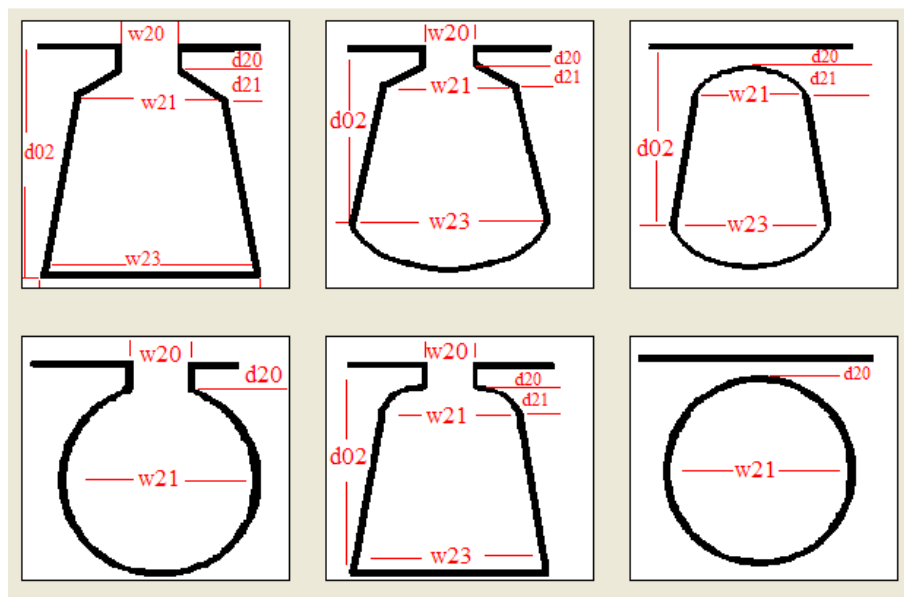


Figure 2.4 Rotor slot shapes

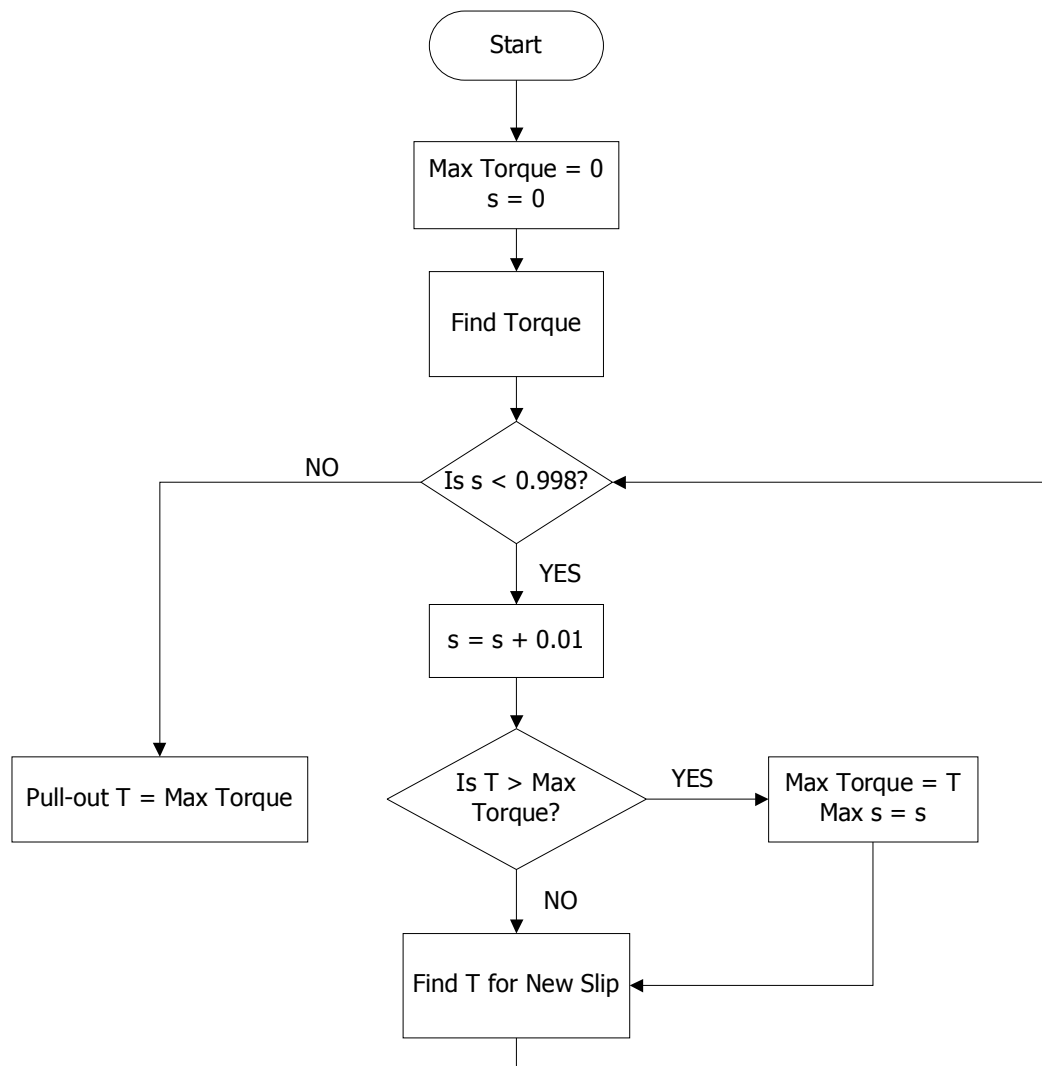


Figure 2.5 The Flowchart of Pull-Out Torque Calculation

2.3. Documentation of TPCAD

In software engineering, to describe the detailed design and architectural design of the software and to allow the new engineers to develop the existing software, documentation is important. For this reason in this study Software Design Description (SDD) is prepared in accordance with Software Design Description Data Item Description (DI-IPSC-81435).

This SDD specifies the design of the TPCAD and describes the design decisions, architectural design, and the detailed design needed to implement the program. It provides information needed for software support. In addition, the purpose of SDD document is to provide a comprehensive explanation of TPCAD software modules, architectural and interface design, TPCAD components, and the detailed explanations of designing TPCAD components providing users minimum and sufficient information to enable an experienced programmer understand the application program and the procedures of the software system.

2.4. GUI of TPCAD

2.4.1. Introduction

TPCAD is developed in Microsoft Visual Studio .Net environment using the C++ since the Microsoft Visual Studio .Net environment helps the programmer to develop user-friendly interfaces in a fast way by supplying all necessary design tools such as forms, buttons, graphic plotters and list view. Therefore, TPCAD has a user friendly GUI that anyone can use it without knowing much about the software and hardware of the computer systems.

2.4.2. Using the TPCAD Program

The main form of the TPCAD provides communication between the user and the program and can be seen in Figure 2.6 below. By using this form, user can reach all the facilities of the program. The main menu has the "File", "Curve", "Analysis" and "Optimization" items. Moreover, all items have sub-menus in each.

2.4.2.1. File Menu

In the "File" menu, the user is asked to enter the file name to load. This file stores the data related to winding details, lamination dimensions, rotor and stator parameters and other related data of the motor. By selecting a file from this menu, the motor data, which will be analyzed, is loaded and a new form will be opened in the main form like seen in Figure 2.7.

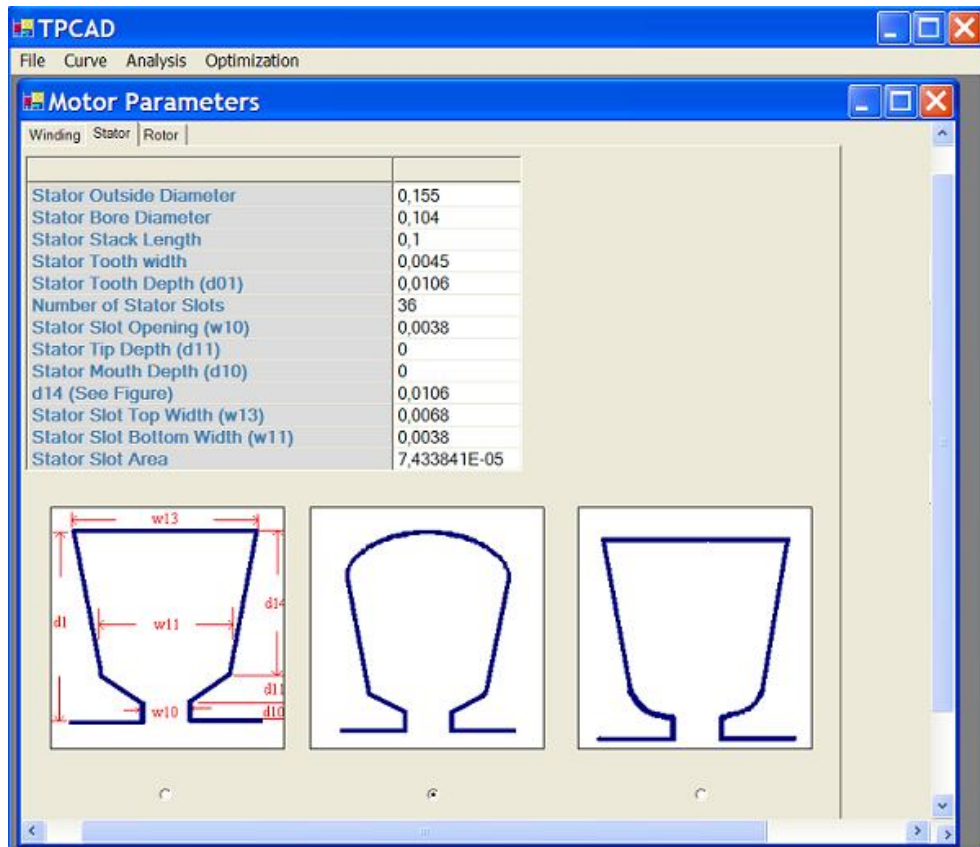


Figure 2.6 Main Form of TPCAD

All the material properties and physical dimensions of the motor to be analyzed are loaded into Motor Parameters form. Dimensions are in metric size.

There are 3 tabs on the form that contains:

a) Winding information such as

- Number of conductors
- Number of poles
- Stacking Factor
- Per unit pitch
- Reactance voltage drop factor
- Total flux factor
- Cross-sectional area of Conductor
- Resistivity of copper, etc...

b) Stator dimensions such as

- Stator outside diameter
- Stator bore diameter
- Stator stack length
- Stator tooth width
- Stator tooth depth
- Number of stator slots
- Stator slot opening, etc...

c) Rotor dimensions

- Rotor outside diameter
- Rotor shaft diameter
- Rotor stack length
- Rotor tooth width
- Rotor tooth depth
- Number of rotor slots
- Rotor slot opening, etc...

User can edit or change the data of the motor by clicking each of the parameters on the list view, and user can change slot shapes by clicking the radio button below the shapes. Three slot shapes for stator and five slot shapes for rotor are defined. Then user can see the effects of the slot shapes on the analysis and performance of the motor.

After editing or changing the data of the investigated data, user can save the new file that contains motor data by using "Save" and "Save As" Buttons.

Stator Outside Diameter	0.155
Stator Bore Diameter	0.104
Stator Stack Length	0.1
Stator Tooth width	0.0045
Stator Tooth Depth (d01)	0.0106
Number of Stator Slots	36
Stator Slot Opening (w10)	0.0038
Stator Tip Depth (d11)	0
Stator Mouth Depth (d10)	0
d14 (See Figure)	0.0106
Stator Slot Top Width (w13)	0.0068
Stator Slot Bottom Width (w11)	0.0038
Stator Slot Area	7.433841E-05

Figure 2.7 Motor Parameters Form

2.4.2.2. Curve Menu

In the "Curve" menu, the user is asked to enter the B-H curve and Core-Loss curve. By choosing "B-H curve" or "Core Loss curve" options, a new form is

opened. If the user chooses "B-H curve" option, the form B-H curve is opened like in Figure 2.8.

2.4.2.2.1. B-H Curve Form

To calculate the performance of the motor, giving the motor data to program is not sufficient. The program needs to material data used for manufacturing the motor. By clicking the "Open" Button, user can enter the material data file. After loading material data file, the B and H values are seen on the list view and B - H curve is plotted on the graph Figure 2.8. User can change or edit material data by clicking the list view. After changing the new curve graph is plotted again, and effects of the change is observed on the graph. In addition, these changes affect the motor analysis and optimization process. B-H Curve can be saved as well.

B – H Curve is not linear and it is not possible to represent it by a simple nonlinear function. Therefore, to represent the material characteristics linear interpolation is used. Values between points that user entered are calculated by linear interpolation of the two end points.

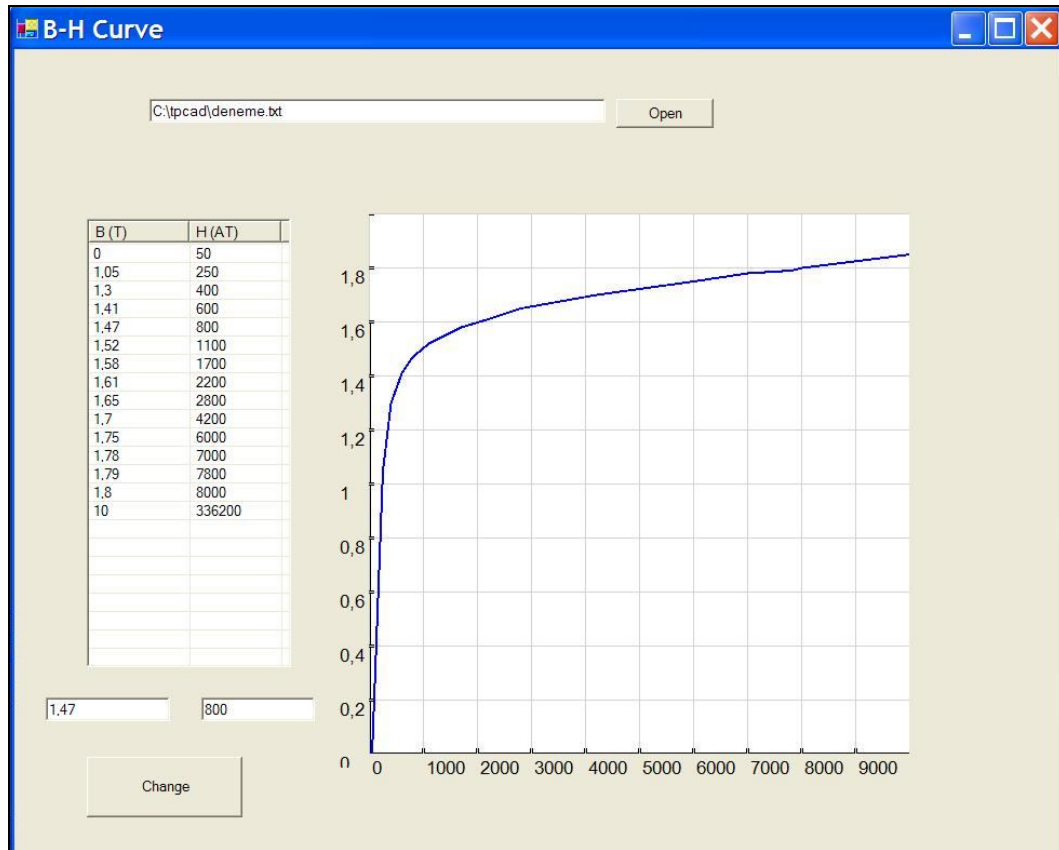


Figure 2.8 B-H Curve Form

2.4.2.3. Analysis Menu

In the Analysis menu, the performance of the motor will be calculated. User can choose one of the four different formats to view the performance of the motor from the sub menus of Analysis menu. These formats are as follows.

2.4.2.3.1. Graphical Analysis Form

When this option is chosen, the performance of the motor is calculated and saved in a temporary file. After then recorded data are read and are given as

graphic charts. The first graphic chart is Torque versus Speed Analysis of motor like in Figure 2.9.

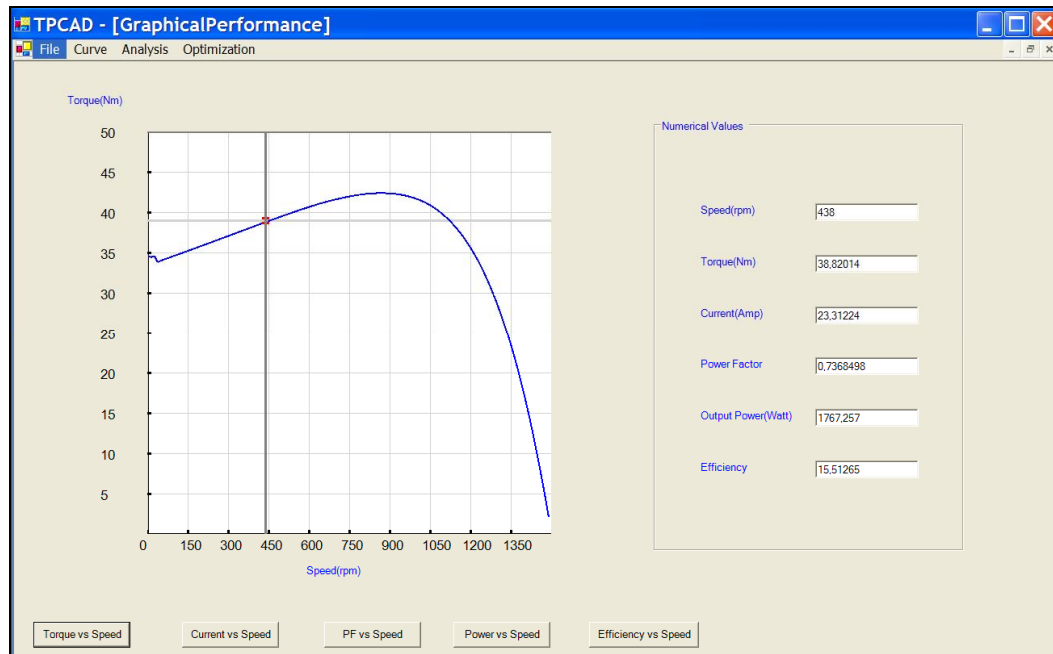


Figure 2.9 Graphical Analysis of the Motor (Torque vs. Speed)

User can choose different graphic charts by pressing buttons under the graphic. There are 5 different graphic charts options. These are:

- 1) Torque versus Speed
- 2) Line current versus Speed
- 3) Power Factor versus Speed
- 4) Output Power versus Speed
- 5) Efficiency versus Speed

By using the cursor on the graphic chart, user can take the numerical values from the graph. These numerical values are seen on the right side of the form.

2.4.2.3.2. At a Power Analysis Form

This form gives the performance of the motor at a specific output power. When this option is chosen, the performance of the motor is calculated and saved in a temporary file. Then a new form is opened like in Figure 2.10. Then user can enter the value of the investigated output power in watts into the “Enter Power” box. The entered power cannot be greater than the maximum output power of the motor. After entering the appropriate power, user presses the “Calculate” button. Then the performance of the motor at the entered power is calculated and displayed on the form.

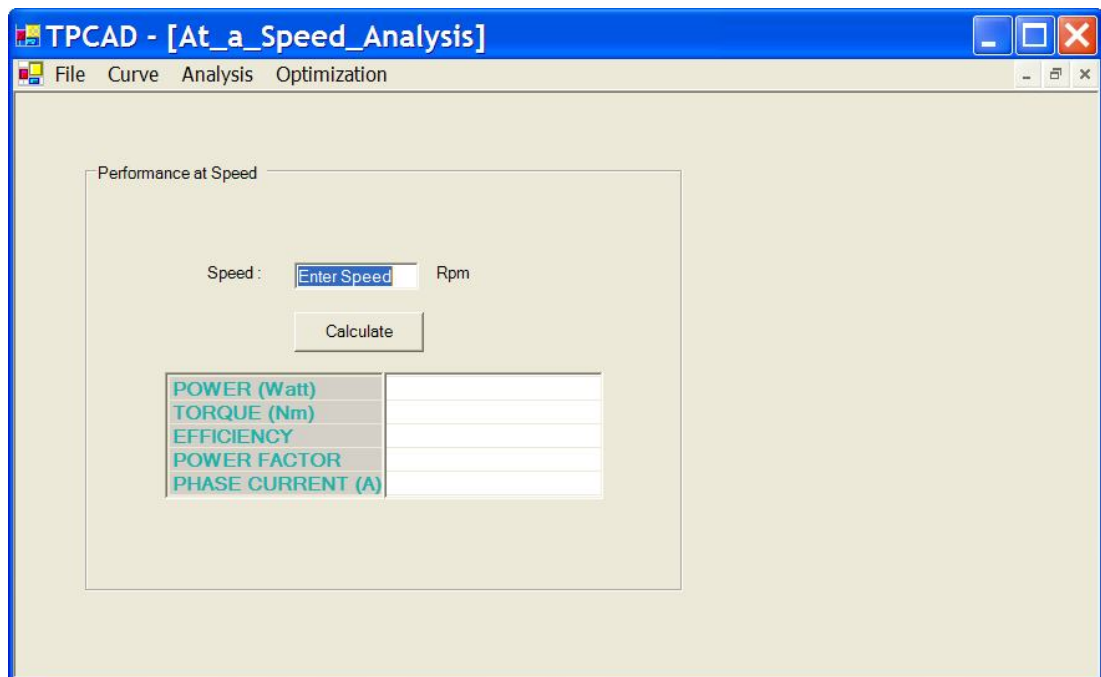
The screenshot shows a software window titled "TPCAD - [At a Power Analysis]". The window has a menu bar with "File", "Curve", "Analysis", and "Optimization". The main area contains a form titled "Performance at Power". Inside this form, there is a label "Power :" followed by a text input field containing "Enter Power" and the unit "Watt". Below this is a "Calculate" button. At the bottom of the form, there is a table with five rows and two columns. The first column lists motor parameters: "SPEED (rpm)", "TORQUE (Nm)", "EFFICIENCY", "POWER FACTOR", and "PHASE CURRENT(A)". The second column contains five empty text input fields for the calculated values.

Performance at Power	
Power :	<input type="text" value="Enter Power"/> Watt
<input type="button" value="Calculate"/>	
SPEED (rpm)	<input type="text"/>
TORQUE (Nm)	<input type="text"/>
EFFICIENCY	<input type="text"/>
POWER FACTOR	<input type="text"/>
PHASE CURRENT(A)	<input type="text"/>

Figure 2.10 Analysis of the Motor at a Specific Power

2.4.2.3.3. At a Speed Analysis Form

This form gives the performance of the motor at a specific speed. When this option is chosen, the performance of the motor is calculated and saved in a temporary file. Then a new form is opened like in Figure 2.11. Then user can enter the value of the investigated speed in rpm into the "Enter Speed" box. The entered speed cannot be greater than the maximum speed of the motor. After entering the appropriate speed, user presses the "Calculate" button. Then the performance of the motor at the entered speed is calculated and displayed on the form.



The screenshot shows a software window titled "TPCAD - [At_a_Speed_Analysis]" with a menu bar containing "File", "Curve", "Analysis", and "Optimization". The main area is titled "Performance at Speed" and contains the following elements:

- A label "Speed :" followed by a text input field containing "Enter Speed" and the unit "Rpm".
- A "Calculate" button below the input field.
- A table with five rows of motor performance metrics, each with a corresponding empty input field to its right:

POWER (Watt)	
TORQUE (Nm)	
EFFICIENCY	
POWER FACTOR	
PHASE CURRENT (A)	

Figure 2.11 Analysis of the Motor at a Specific Speed

2.4.2.3.4. Summary Analysis Form

This form gives the summary of the performance motor. Full load and starting performance, rated voltage, speed and frequency of the and pull out torque produced by the motor is displayed in this form like seen in Figure 2.12 below.

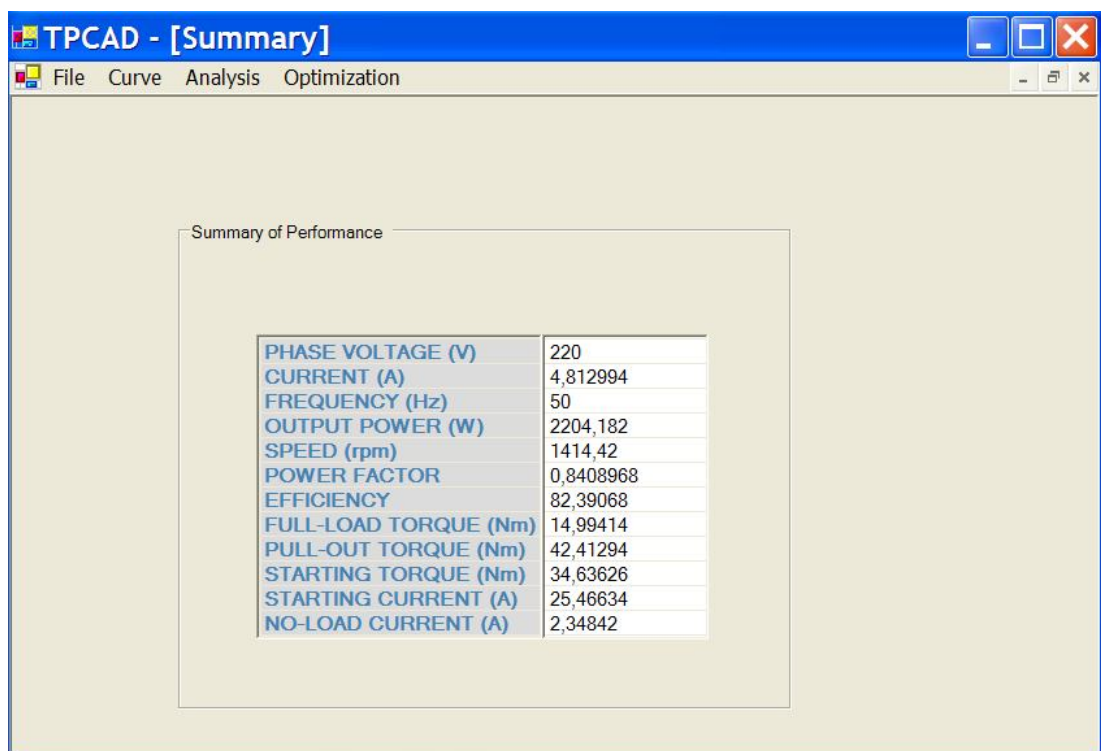


Figure 2.12 Summary of the Analysis

2.4.2.4. Optimization Menu

In the Optimization Menu, user is asked to enter 3 different file. The first file is **Constant Data** file which has the extension .con. The data is kept constant during the optimization process.

The second file holds the Initial Design of the motor. It has the extension .xve.

The third file is the Constraints file. In this file the performance and geometric specifications of the motor is defined. There are 28 inequality and 1 equality constraints in this file.

After loading these files, a new form is opened like in Figure 2.13. All the parameters of the motor to be optimized are loaded into Optimization form. There are 3 tabs on the form that contains Constant Data parameters, Initial Design and Constraints data. User can edit or change the data of the motor by clicking each of the parameters on the list view.

After editing or changing the data of the investigated data, user can save the new file that contains motor data by using "Save" and "Save As" Buttons.

The screenshot shows a software window titled "TPCAD - [OptimizationForm]" with a menu bar (File, Curve, Analysis, Optimization) and three tabs: "Constant Data", "Initial Design", and "Constraints". The "Constant Data" tab is active, displaying a list of 29 parameters in a table. To the right of the table are two buttons: "Save" and "Save As".

Pull-out Torque	42
Starting Torque	33.5
Starting Current	26.5
Output Power	2200
Power Factor	0.83
Efficiency	0.83
Full-load Current	5
Full-load Torque	14.5
Stator Tooth Flux Density	1.8
Rotor Tooth Flux Density	1.8
Stator Core Flux Density	1.8
Rotor Core Flux Density	1
Rotor Current Density	5
Air Gap Flux Density	1
Temperature Rise	80
Cross-sectional Area of Single Bar	1E-05
Cross-sectional Area of End Ring	9E-05
Mean Diameter of End Ring	0.05
Stator Slot Area	1E-05
d14	0.001
Stator Slot Top Width	0.001
Stator Slot Bottom Width	0.001
Rotor Slot Area	1E-05
Rotor Outside Diameter	0.05
Rotor Stack Length	0.05
Rotor Slot Top Width	0.001
Rotor Slot Bottom width	0.001
Yoke Depth	0.01
Stator Back-core Mmf Drop Constant	0.001

Figure 2.13 Optimization Form

After all the data is loaded then user can start the Optimization by choosing **Start Optimization** sub-item of the Optimization menu. Then a new form is opened which is called Performance Output. In this form, the optimization process is seen in every iteration. Design vector and the output of the motor are seen also on this form like in Figure 2.14. The percentage of the completed iterations is seen on the progress bar.

The optimization algorithm employed in the software uses "Augmented Lagrangian" method with "Davidon-Fletcher and Powell" minimization technique[11].

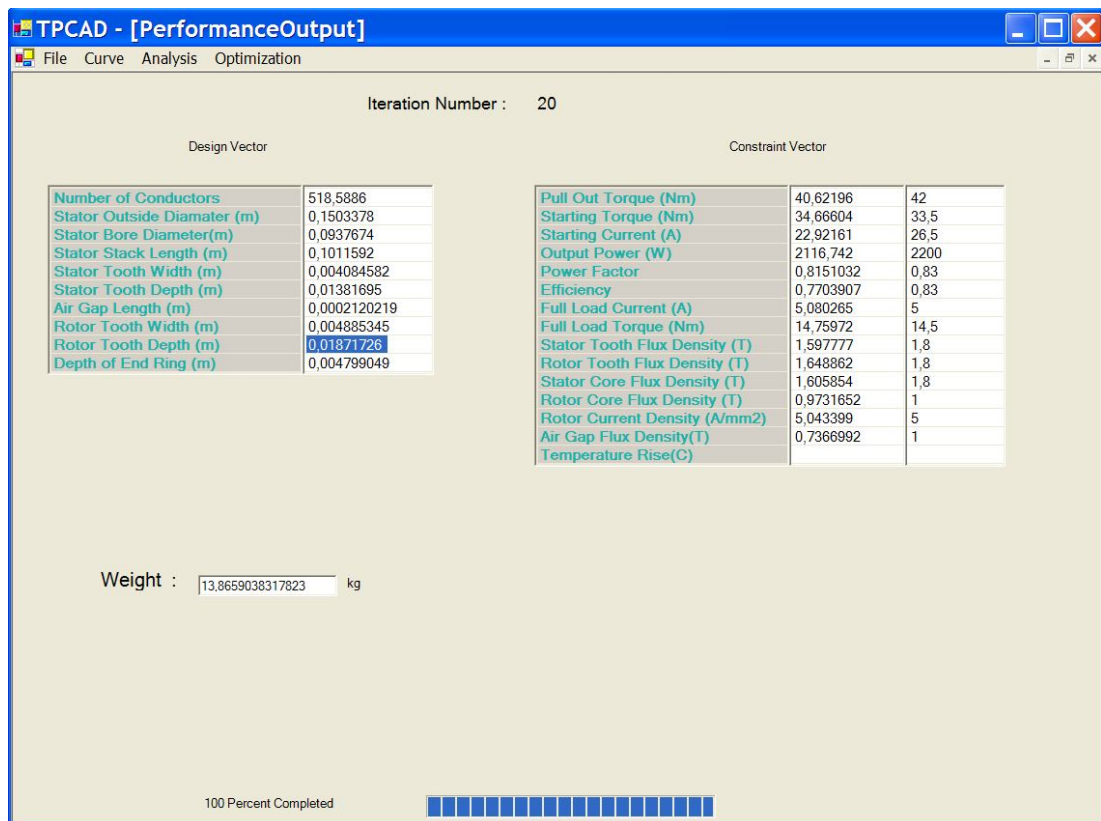


Figure 2.14 Performance Output of the Optimization Process

2.5. Results of TPCAD

In this section, the results of the performance calculations of the TPCAD software is compared with the existing commercial motors. Also, to test the design ability of the optimization part of the software, a new motor is designed with the TPCAD.

2.5.1. Analysis Result

The analysis part of the TPCAD is tested for some different commercial motors, results are accurate, and errors between the actual performance of the motor is negligible. Three different motors are used for testing the TPCAD. These motors are 90L6B, 100L4A, and QU112M4B, manufactured in TEE A.Ş. The all the parameters of the data are in Appendix A and these parameters are taken from [11]. Table 2.1 shows the calculated and measured values of the motors.

After calculating the analysis of the 3 different motors, the percentage errors of the motors are found and these values are shown in Table 2.2.

Table 2.1 Test Results of TPCAD with Different Motors

	M1		M2		M3	
Motor Performance	Test Results	Calculated Results	Test Results	Calculated Results	Test Results	Calculated Results
Output Power(W)	1130	1105	2200	2204	4020	4021
Pull-out Torque(Nm)	31.09	35.99	39.34	42.41	82.80	85.87
Starting Torque(Nm)	29.50	32.20	33.25	34.64	65.50	68.51
Starting Current(Amp)	16.19	14.64	27.51	25.47	30.02	30.24
Power factor	0.78	0.76	0.80	0.84	0.83	0.85
Efficiency (1=100%)	0.66	0.68	0.83	0.82	0.86	0.84
Full-load Current(Amp)	3.34	3.18	5.09	4.81	4.88	4.90
Full-load Torque(Nm)	11.28	11.27	14.51	14.99	26.75	27.02

Table 2.2 Error Percentages of the TPCAD

	M1	M2	M3
Motor Performance	Error Percentages	Error Percentages	Error Percentages
	%	%	%
Output Power	2.20	0.18	0.02
Pull-out Torque	15.76	7.08	3.80
Starting Torque	9.15	4.18	4.60
Starting Current	9.50	7.41	0.73
Power factor	2.56	5.00	2.40
Efficiency	3.00	1.20	2.32
Full-Load Current	4.79	5.50	0.40
Full-load Torque	0.09	3.30	1.00

As a result, when we look at the error percentages of the motors listed in Table 2.2, the error of the pull out torque starting torque and starting current is high. However, if we look at the full-load conditions, the error percentage is below the 5 %. In addition, in some cases it is nearly zero. By looking these results, we can conclude that TPCAD can predict the motor performance accurately for full-load conditions. However, for the starting condition it is difficult for the TPCAD to calculate the exact solution. The higher percentage error is due to the higher starting current, which causes saturation at tooth tips. TPCAD employs a simple two-dimensional resistive equivalent of the magnetic circuit. As a consequence, the prediction accuracy is around 10 %.

2.5.2. Optimization Result

TPCAD is tested on the commercially available motor QU100L4A to confirm the ability of reaching optimum designs. Several initial designs are given as the starting point to the program. These initial designs are listed in Table 2.3. The results are shown in Table 2.4.

By looking the results, it can be said that program is successful for optimizing the motor weight while satisfying the constraints.

Table 2.3 Initial and Final Design of QU100L4A

Design Vector	I1	F1	I2	F2	I3	F3
Number Of Conductors	516	539	556	543	516	538
Stator Outside Diameter (mm)	155	159	162	162	148	160
Stator Bore Diameter(mm)	104	102	104	98.2	100	105
Stack Length(mm)	100	98	115	98.8	88	97.6
Stator Tooth Width(mm)	4.5	3.9	4.8	3.98	4.5	3.79
Stator Tooth Depth(mm)	10.6	16.0	10.6	16.6	10.6	16.8
Air Gap Length(mm)	0.25	0.244	0.25	0.242	0.25	0.27
Rotor Tooth Width(mm)	5.72	4.95	5.72	5.04	5.2	4.95
Rotor Tooth Depth(mm)	17.37	17.95	17.37	18.4	17.37	17.95
Depth of End Ring(mm)	7.49	5.05	7.49	4.89	7.49	5.00
Weight(kg)	14.61	15.22	18.08	15.31	12.48	15.64

Table 2.4 Results of Optimization of QU100L4A

Performance	I1	F1	I2	F2	I3	F3
Output Power (W)	2390	2195	1988	2188	2234	2190
Pull-Out Torque (Nm)	38.76	41.75	30.85	42.15	36.76	41.95
Starting Torque (Nm)	31.2	34.34	23.63	34.81	30.53	34.26
Starting Current (Amp)	23.30	22.93	18.44	22.999	24.55	22.91
Power Factor	0.87	0.826	0.886	0.826	0.68	0.813
Efficiency	0.769	0.802	0.78	0.802	0.66	0.811
Full-Load Current (Amp)	5.387	4.999	4.33	4.979	7.43	5.014
Full-Load Torque (Nm)	16.65	15.30	13.88	15.25	15.58	15.27
Stator Tooth Flux Density (T)	1.47	1.6	1.12	1.6	1.68	1.5997
Stator Core Flux Density (T)	1.57	1.56	0.97	1.52	2.0	1.52
Rotor Tooth Flux Density (T)	1.43	1.54	1.16	1.559	1.79	1.514
Rotor Core Flux Density (T)	0.73	0.801	0.59	0.837	0.906	0.792
Rotor Current Density (A/mm ²)	5.56	4.984	4.996	4.989	5.272	4.996
Air Gap Flux Density (T)	0.68	0.669	0.545	0.687	0.798	0.659
Weight (kg)	14.88	15.22	18.081	15.31	12.48	15.64

After examining the final results in Table 2.4 it can be observed that the optimization constraints are satisfied.

For Initial Design 1:

- Pull-Out Torque: 41.75 not satisfied by the percentage error 0.5 % so it is negligible.
- Starting Torque: 34.34, satisfied
- Starting Current: 22.93 Amp, satisfied
- Power Factor: 0.826, Not satisfied by the percentage error 0.4 % so it is negligible
- Efficiency: 0.802, not satisfied by the percentage error 3 %, but it is acceptable since the error smaller than 5 %.
- Full-Load Current: 4.999 Amp, satisfied.
- Full-Load Torque: 15.30 Nm, satisfied.
- Stator Tooth Flux Density: 1.6 T, satisfied.
- Stator Core Flux Density: 1.56 T, satisfied.
- Rotor Tooth Flux Density: 1.54 T, satisfied.
- Rotor Core Flux Density: 0.801 T, satisfied.
- Rotor Current Density: 4.984 A/mm², satisfied.
- Air Gap Flux Density: 0.669 T, satisfied.
- Weight: 15.22 kg satisfied.
- Time Elapsed: 350 seconds

2.6. Optimization of Three Phase Induction Machine with modeFRONTIER

One of the aims of this thesis is to use an optimization tool, which is named as modeFRONTIER to optimize a three-phase induction machine. For this purpose, a project is designed in modeFRONTIER like seen in Figure 2.15.

Design steps of project at modeFRONTIER and knowledge about the modeFRONTIER is explained in Section 4.8.5 of this study.

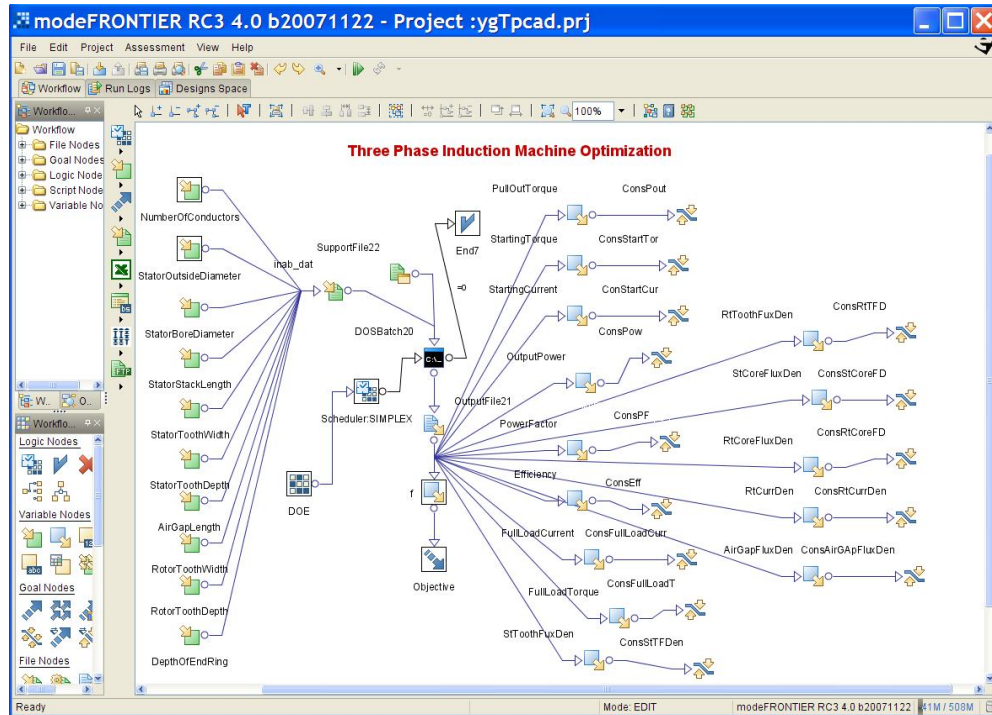


Figure 2.15 Designed Project at modeFRONTIER for the Three Phase Induction Machine

In this section, different optimization methods are applied to problem to obtain an optimal design of Three-Phase Induction Machine. But, only two methods give the feasible results. These are Simplex Method and NLP. Also the same motor, QU100L4A, which is used to test the TPCAD's optimization part, is used for test of the modeFRONTIER project.

2.6.1. Results of Simplex Algorithm

For the optimization of Three-Phase Induction Machine, firstly Simplex Algorithm is used for different initial designs. The results are listed in Table 2.6.

Simplex is a simple optimization algorithm seeking the vector of parameters corresponding to the global extreme (maximum or minimum) of any n -dimensional function $F(x_1, x_2, \dots, x_n)$, searching through the parameter space ("search area").

From a geometrical point of view, a simplex is a polyhedron containing $N + 1$ points in a N dimensional space, thus in two dimensions is a triangle, in three dimension is a tetrahedron, and so forth. When the points are equidistant, the simplex is said to be regular. The Simplex method compares the values of the objective function at the $N + 1$ vertices and then moves this polyhedron gradually towards the optimum point during the iterative process. For each iteration of the algorithm, it attempts to replace simplex vertices that yield high function values with vertices whose new function values are lower. The aim of this algorithm is to move the simplex, by replacing vertices, into the neighborhood of a minimizer. For the further information about Simplex method, see the Appendix B of this study.

Parameters of Simplex Algorithm:

Maximum Number of Iterations: 1000

Final Termination Accuracy: 1.0E-5

Constraint Penalty Policy: Automatic

Table 2.5 Initial and Final Design of QU100L4A by using Simplex

Design Vector	I1	F1	I2	F2	I3	F3
Number Of Conductors	516	491	556	443	516	508
Stator Outside Diameter (mm)	155	170	162	188	148	149
Stator Bore Diameter (mm)	104	121	104	143	100	114
Stack Length (mm)	100	126	115	162	88	109
Stator Tooth Width (mm)	4.5	3.1	4.8	3.6	4.5	4.0
Stator Tooth Depth (mm)	10.6	10.3	10.6	4.0	10.6	11.4
Air Gap Length (mm)	0.25	0.32	0.25	0.37	0.25	0.33
Rotor Tooth Width (mm)	5.72	8.13	5.72	8.11	5.20	6.74
Rotor Tooth Depth (mm)	17.37	21.25	17.37	32.15	17.37	19.71
Depth of End Ring (mm)	7.49	8.36	7.49	13.10	7.49	6.95

By looking the results, it can be said that program is not successful for optimizing the motor weight while satisfying the constraints.

Table 2.6 Results of Optimization of QU100L4A

Performance	I1	F1	I2	F2	I3	F3
Output Power (W)	2390	2210	1988	2921	2234	2455
Pull-Out Torque (Nm)	38.76	42.00	30.85	43.61	36.76	43.08
Starting Torque (Nm)	31.2	34.09	23.63	33.59	30.53	34.48
Starting Current (Amp)	23.30	23.05	18.44	26.33	24.55	24.89
Power Factor	0.87	0.80	0.886	0.92	0.68	0.80
Efficiency	0.769	0.82	0.78	0.80	0.66	0.80
Full-Load Current (Amp)	5.387	5.090	4.33	5.999	7.43	5.790
Full-Load Torque (Nm)	16.65	15.40	13.88	20.32	15.58	17.10
Stator Tooth Flux Density (T)	1.47	1.799	1.12	1.331	1.68	1.565
Stator Core Flux Density (T)	1.57	1.589	0.97	0.948	2.0	1.732
Rotor Tooth Flux Density (T)	1.43	0.842	1.16	0.727	1.79	1.134
Rotor Core Flux Density (T)	0.73	0.518	0.59	0.450	0.906	0.617
Rotor Current Density (A/mm ²)	5.56	4.998	4.996	2.900	5.272	4.990
Air Gap Flux Density (T)	0.68	0.485	0.545	0.355	0.798	0.573
Weight (kg)	14.88	20.795	18.081	31.739	12.48	17.390

2.6.2. Results of NLPQLP

For the optimization of Three-Phase Induction Machine, NLPQLP (Nonlinear Programming for Parallel Computing) Algorithm is used. But for the initial design I1, I2 and I3 are given at the same time as a table. The results are listed in Table 2.8.

NLPQLP is developed by K. Schittkowski and represents a very robust implementation of a sequential quadratic programming algorithm. It requires only very few user-provided parameters [20][21].

Sequential quadratic programming or SQP methods are the standard general-purpose tool for solving smooth nonlinear optimization problems under the following assumptions:

- The problem is not too large.
- Functions and gradients can be evaluated with sufficiently high precision.
- The problem is smooth and well scaled.
- There is no further model structure that can be exploited.

SQP methods allow the solution of a wide range of nonlinear programming problems in an efficient and reliable way. Either implicitly or proceeding from simple modifications of the underlying optimization problem, a much larger class of different nonlinear programming problems can be solved by NLPQLP, for example least squares or min-max optimization [20].

Table 2.7 Initial and Final Design of QU100L4A by using NLPQLP

Design Vector	I1	I2	I3	F
Number Of Conductors	516	556	516	556
Stator Outside Diameter (mm)	155	162	148	162
Stator Bore Diameter (mm)	104	104	100	100
Stack Length (mm)	100	115	88	93
Stator Tooth Width (mm)	4.5	4.8	4.5	4.1
Stator Tooth Depth (mm)	10.6	10.6	10.6	15.9
Air Gap Length (mm)	0.25	0.25	0.25	0.28
Rotor Tooth Width (mm)	5.72	5.72	5.20	5.05
Rotor Tooth Depth (mm)	17.37	17.37	17.37	18.14
Depth of End Ring (mm)	7.49	7.49	7.49	4.69

By looking the results, it can be said that NLPQLP algorithm is successful for optimizing the motor weight while satisfying the constraints. Also when it is compared with the Augmented Lagrangian Method which is applied in the TPCAD program, it is seen that NLPQLP Algorithm is more efficient.

Table 2.8 Results of Optimization of QU100L4A by NLPQLP

Performance	F
Output Power (W)	2200
Pull-Out Torque (Nm)	42.00
Starting Torque (Nm)	34.90
Starting Current (Amp)	23.38
Power Factor	0.80
Efficiency	0.80
Full-Load Current (Amp)	5.18
Full-Load Torque (Nm)	15.34
Stator Tooth Flux Density (T)	1.60
Stator Core Flux Density (T)	1.59
Rotor Tooth Flux Density (T)	1.62
Rotor Core Flux Density (T)	0.83
Rotor Current Density (A/mm²)	4.998
Air Gap Flux Density (T)	0.703
Weight (kg)	14.671
Time Elapsed (seconds)	1300

2.6.3. Conclusion Of Optimization

It is observed that Simplex Algorithm and the NLPQLP algorithm for the optimization of the three-phase induction machine are the most suitable algorithms. Although Simplex algorithm is faster than the NLPQLP algorithm, it finds the local minimums. NLPQLP algorithm can find global minimum but user should give more initial designs.

2.7. Conclusion of TPCAD

A software tool TPCAD that can be used for three-phase induction machine design was developed. This tool also is tested with using 3 different commercial motors. After the test results it is seen that the performance prediction for the full load conditions lie in range in 5 % range with the actual motors. The error in starting performance calculation is higher, however, still around 10 % and is acceptable. The prediction accuracy of these predictions can be improved. However, computational burden of the calculation would increase considerably.

The construction of optimum motor geometry, which satisfies required performance specifications and constraints, is a very important subject. In this study, the optimization of three-phase induction motor is formulated as a weight minimization problem. For the solution of the optimization problem, Augmented Lagrangian method developed earlier[11] is used. Three initial design vectors are defined, one smaller than the available commercial motor, one on the same size, finally a design which is larger than the commercial motor. It is found that this method works well and converges to similar local minima. Next, modeFRONTIER optimization algorithms Simplex and NLPQLP are applied to the problem. Simplex algorithm failed to converge to minimum point, on the other hand, NLPQLP lead to excellent results.

At last, using TPCAD and modeFRONTIER makes faster the design of three-phase induction motor, and performance of the existing motor can be calculated easily.

CHAPTER 3

SINGLE PHASE CAPACITOR TYPE INDUCTION MACHINE COMPUTER AIDED DESIGN (SPCAD)

3.1. Technical Explanation of SPCAD

SPCAD (Single-Phase Induction Machine Computer Aided Design) is a personal computer (PC) windows application software product designed for design and optimization of single-phase induction motors. SPCAD is also developed to perform the following items:

- Performance calculation of the user defined parameters
- Analysis of Single Phase Squirrel Cage Induction motors
 - At a Speed
 - At a Power
 - Graphical
- Loss and Temperature calculation at any speed
- Design optimization for cost or weight minimization while satisfying a set of design criteria. Allows the user up to 10 variables to change within the set of design criteria. User can design a specified motor, starting with an initial design.
- Ability to change the core-loss curve and B-H curve.
- Plotting and printing of outputs.

With the above characteristics of SPCAD, effect of materials used in manufacturing the motor on the performance can be investigated. Three slot shapes for stator and five slot shapes for rotor are defined and also by

changing the rotor and stator slot shapes motor performance can be observed with different slot shapes.

SPCAD, while satisfying these items, also has a user friendly GUI (Graphical User Interface). To perform this SPCAD is developed in Microsoft Visual Studio .Net Environment using the C++ object oriented language. The Microsoft Visual Studio .Net environment helps the programmer to develop user-friendly interfaces in a fast way by supplying all necessary design tools such as Windows forms, buttons, graphic plotters, list view. C++ programming language is one of the most powerful and easy compiler, with its object oriented programming logic, existing in the market.

To develop the SPCAD the software modulation program UML 2.1 is used. Visual model of SPCAD classes is illustrated in Figure 3.1.

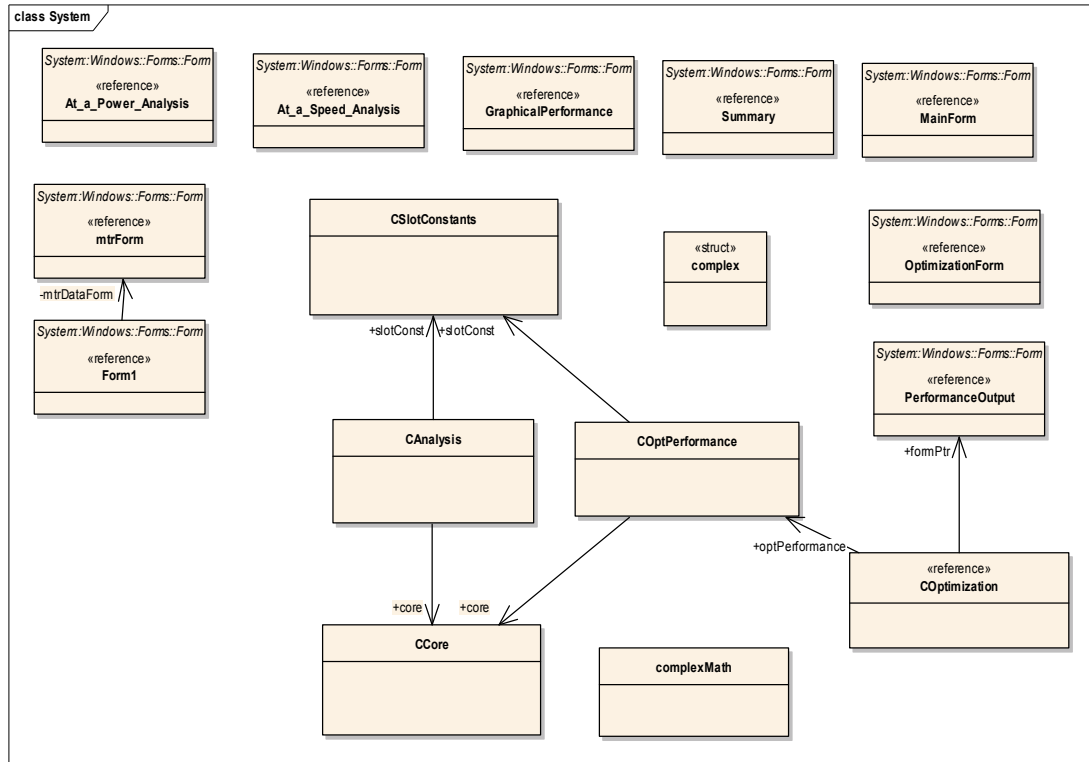


Figure 3.1 Visual Model of SPCAD Classes

3.2. Documentation of SPCAD

In this study for the calculations of performance of the single-phase capacitor type motor, the equations and algorithms in the reference [12] are used.

3.3. GUI of SPCAD

3.3.1. Introduction

SPCAD is developed in Microsoft Visual Studio .Net environment using the C++ since the Microsoft Visual Studio .Net environment helps the programmer to develop user-friendly interfaces in a fast way by supplying all necessary design tools such as forms, buttons, graphic plotters and list view.

Therefore, SPCAD has a user friendly GUI that anyone can use it without knowing much about the software and hardware of the computer systems.

3.3.2. Using the SPCAD Program

The main form of the SPCAD provides communication between the user and the program and can be seen in Figure 3.2 below. By using this form, user can reach all the facilities of the program. The main menu has the "File", "Curve", "Analysis" and "Optimization" items. In addition, all items have sub-menus in each.

3.3.2.1. File Menu

In the "File" menu, the user is asked to enter the file name to load. This file stores the data related to winding details, lamination dimensions, rotor and stator parameters and other related data of the motor. By selecting a file from this menu, the motor data, which will be analyzed, is loaded and a new form will be opened in the main form like seen in Figure 3.3.

Parameter	Value
Title	70 FRAME 4P 1/4HP
Date	7/11/1991
Impressed Voltage	220
Frequency	50
Number of Poles	4
Number of Conductor	1568
Stacking Factor	0,96
Reactance Voltage Drop Factor	0,94
Total Flux Factor	0,94
Primary Flux Factor	0,96
Number of Parallel Paths in Stator Winding	1
Cross-sectional Area of Conductor	2,4E-07
Resistivity of Copper	1,72E-08
Resistivity of Aluminium	2,78E-08
Copper Resistivity Measuring Temperature	25
Aluminium Resistivity Measuring Temperature	20
Cross-Sectional Area of End Ring	0,00012

Figure 3.2 Main Form of SPCAD

All the parameters of the motor to be analyzed are loaded into Motor Parameters form. There are 3 tabs on the form that contains Winding parameters, stator and rotor dimensions. User can edit or change the data of the motor by clicking each of the parameters on the list view, and user can change slot shapes by clicking the radio button below the shapes. Three slot shapes for stator and five slot shapes for rotor are defined. Then user can see the effects of the slot shapes on the analysis and performance of the motor.

After editing or changing the data of the investigated data, user can save the new file that contains motor data by using "Save" and "Save As" Buttons.

Parameter	Value
Stator Outside Diameter	0,1457
Stator Bore Diameter	0,0889
Stator Stack Length	0,08
Stator Tooth width	0,00381
Stator Tooth Depth (d01)	0,0142
Number of Stator Slots	36
Stator Slot Opening (w10)	0,00254
Stator Tip Depth (d11)	0,0009
Stator Mouth Depth (d10)	0,00055
d14 (See Figure)	0,0127
Stator Slot Top Width (w13)	0,0064
Stator Slot Bottom Width (w11)	0,0042
Stator Slot Area	0,0001
Shaft Length	0,25

Figure 3.3 Motor Parameters Form

3.3.2.2. Curve Menu

In the "Curve" menu, the user is asked to enter the B-H curve and Core-Loss curve. By choosing "B-H curve" or "Core Loss curve" options, a new form is opened. If the user chooses "B-H curve" option, the form B-H curve is opened like in Figure 3.4. However, if the user chooses "Core – Loss Curve", then the form Core – Loss curve is opened in Figure 3.5.

3.3.2.2.1. B-H Curve Form

To calculate the performance of the motor, giving the motor data to program is not sufficient. The program needs to material data used for manufacturing the motor. By clicking the "Open" Button, user can enter the material data file. After loading material data file, the B and H values are seen on the list view and B – H curve is plotted on the graph (Figure 3.4). User can change or edit material data by clicking the list view. After changing the new curve graph is plotted again, and effects of the change is observed on the graph. Moreover, these changes affect the motor analysis and optimization process.

B – H Curve is not linear and it is not possible to represent it by a simple nonlinear function. Therefore, to represent the material characteristics linear interpolation is used. Values between points that user entered are calculated by linear interpolation of the two end points.

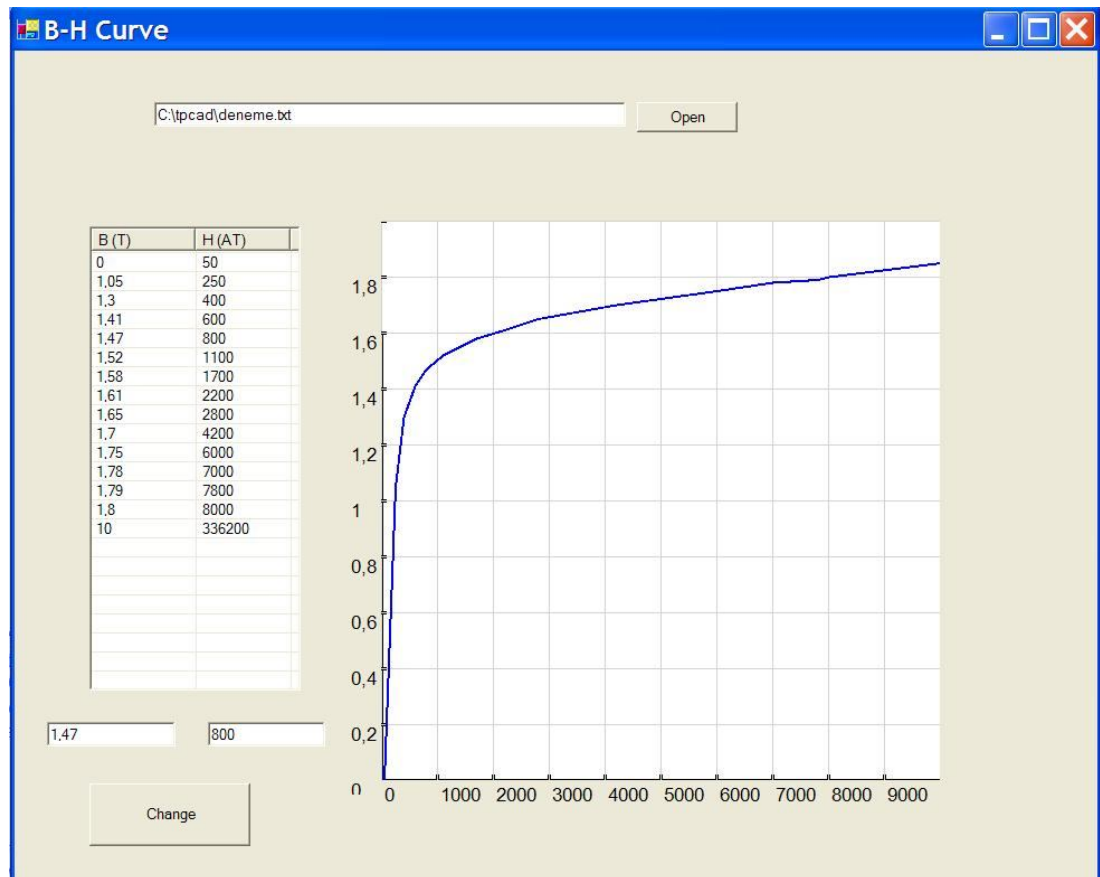


Figure 3.4 B-H Curve Form

3.3.2.3. Analysis Menu

In the Analysis menu, the performance of the motor will be calculated. User can choose four different formats to view the performance of the motor from the sub menus of Analysis menu. These formats are as follows:

3.3.2.3.1. Graphical Analysis Form

When this option is chosen, the performance of the motor is calculated and saved in a temporary file. After then recorded data are read and are given as graphic charts. The first graphic chart is Torque versus Speed Analysis of motor like in Figure 3.5.

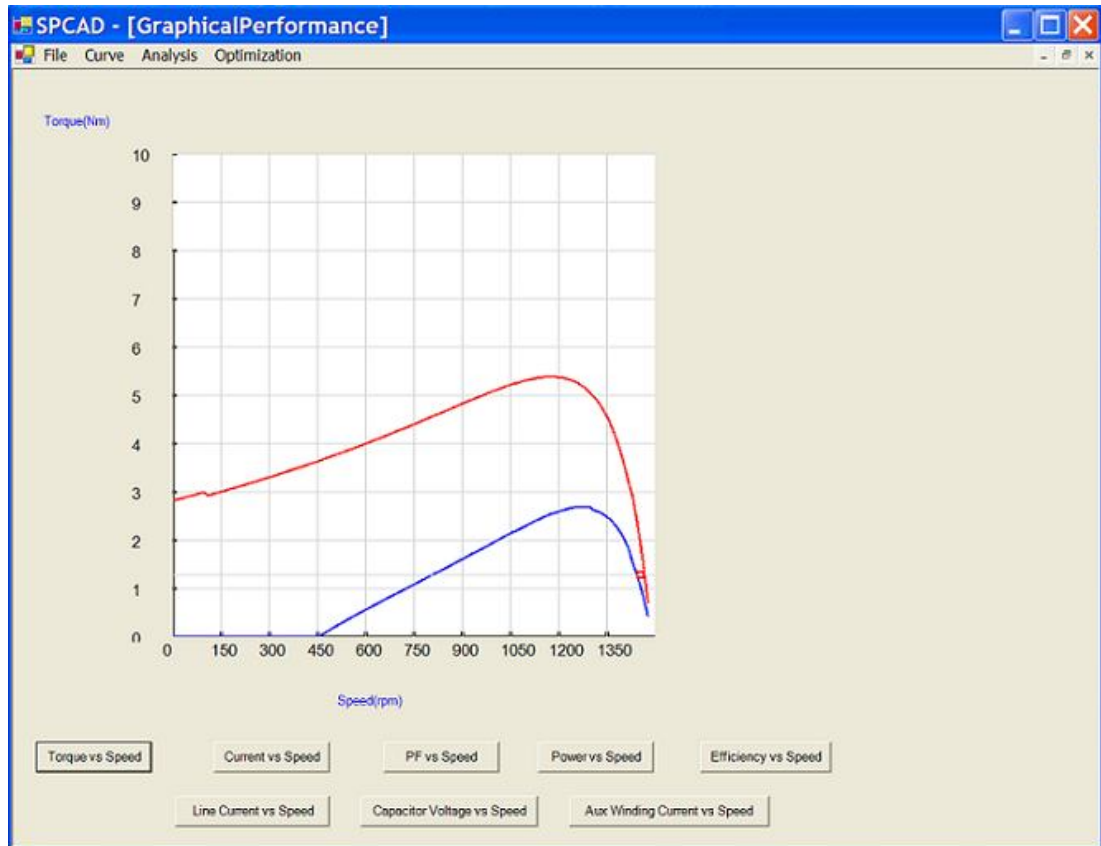


Figure 3.5 Graphical Analysis of the Motor (Torque vs. Speed)

User can choose different graphic charts by pressing buttons under the graphic. There are 8 different graphic charts options. These are:

- 1) Torque versus Speed
- 2) Main Winding Current versus Speed
- 3) Power Factor versus Speed
- 4) Output Power versus Speed
- 5) Efficiency versus Speed
- 6) Line Current versus Speed
- 7) Capacitor Voltage versus Speed
- 8) Auxiliary Winding Current vs Speed

By using the cursor on the graphic chart, user can take the numerical values from the graph. These numerical values are seen on the right side of the form.

3.3.2.3.2. At a Power Analysis Form

This form gives the performance of the motor at a specific output power. When this option is chosen, the performance of the motor is calculated and saved in a temporary file. Then a new form is opened like in Figure 3.6. Then user can enter the value of the investigated output power in watts into the "Enter Power" box. The entered power cannot be greater than the maximum output power of the motor. After entering the appropriate power, user presses the "Calculate" button. Then the performance of the motor at the entered power is calculated and displayed on the form.

The screenshot shows a software window titled "SPCAD - [At_a_Power_Analysis]". The window has a menu bar with "File", "Curve", "Analysis", and "Optimization". The "Analysis" menu is open, showing options: "Graphical", "At a Power" (which is highlighted), "At a Speed", and "Summary".

Inside the window, there is a form with the following elements:

- An "Enter Power" text input box followed by the unit "Watt".
- A "Calculate" button.
- A table with 7 rows and 2 columns. The first column contains labels for motor parameters, and the second column contains empty input boxes for the results.

SPEED (rpm)	
TORQUE (Nm)	
EFFICIENCY	
POWER FACTOR	
MAIN WINDING CURRENT(A)	
LINE CURRENT(A)	
CAPACITOR VOLTAGE(V)	
AUXILIARY WINDING CURRENT(A)	

Figure 3.6 Analysis of the Motor at a Specific Power

3.3.2.3.3. At a Speed Analysis Form

This form gives the performance of the motor at a specific speed. When this option is chosen, the performance of the motor is calculated and saved in a temporary file. Then a new form is opened like in Figure 3.7. Then user can enter the value of the investigated speed in rpm into the "Enter Speed" box. The entered speed cannot be greater than the maximum speed of the motor. After entering the appropriate speed, user presses the "Calculate" button. Then the performance of the motor at the entered speed is calculated and displayed on the form.

The screenshot shows a software window titled "SPCAD - [At_a_Speed_Analysis]". The window has a menu bar with "File", "Curve", "Analysis", and "Optimization". The "Analysis" menu is open, showing options: "Graphical", "At a Power", "At a Speed" (which is highlighted), and "Summary".

Inside the window, there is a form with the following elements:

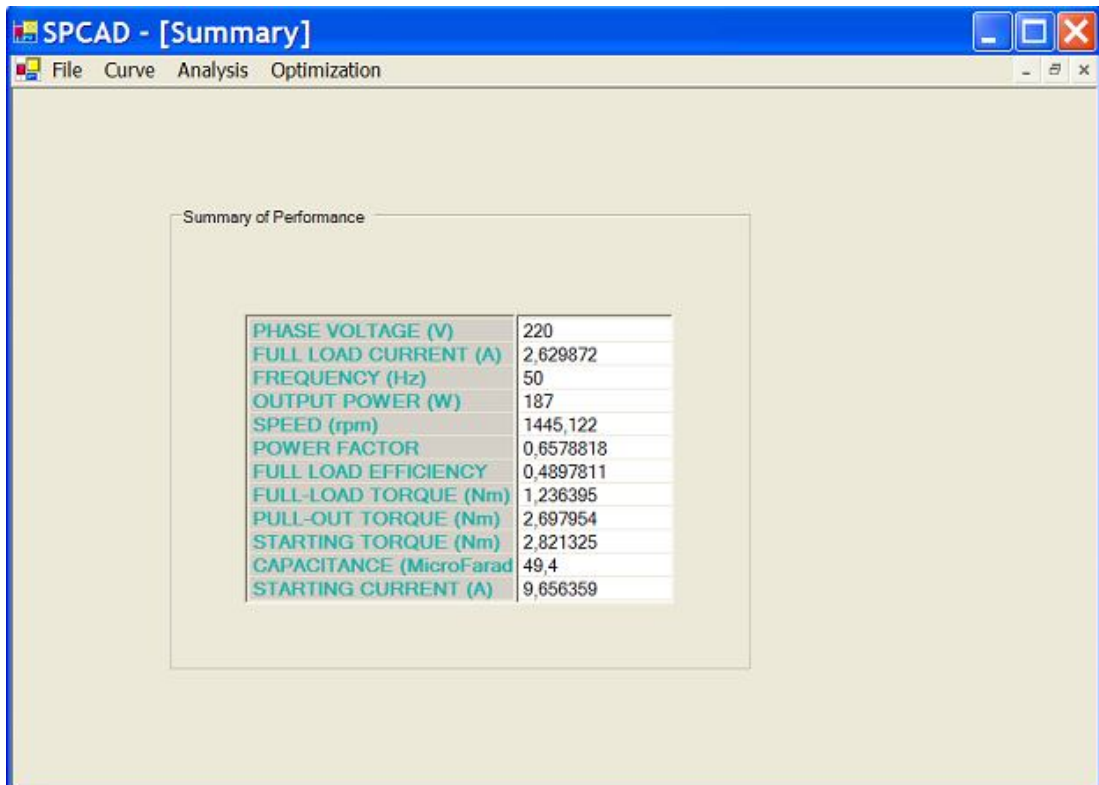
- A label "Enter Speed" followed by a text input field and the unit "rpm".
- A "Calculate" button below the input field.
- A table with two columns. The first column lists motor parameters, and the second column is for the calculated values.

POWER (Watt)	
TORQUE (Nm)	
EFFICIENCY	
POWER FACTOR	
MAIN WINDING CURRENT(A)	
LINE CURRENT(A)	
CAPACITOR VOLTAGE(V)	
AUXILIARY WINDING CURRENT(A)	

Figure 3.7 Analysis of the Motor at a Specific Speed

3.3.2.3.4. Summary Analysis Form

This form gives the summary of the performance of the motor. Full load and starting performance, rated power and voltage is displayed in this form like seen in Figure 3.8.



The screenshot shows a software window titled "SPCAD - [Summary]" with a menu bar containing "File", "Curve", "Analysis", and "Optimization". The main area displays a "Summary of Performance" table with the following data:

Parameter	Value
PHASE VOLTAGE (V)	220
FULL LOAD CURRENT (A)	2,629872
FREQUENCY (Hz)	50
OUTPUT POWER (W)	187
SPEED (rpm)	1445,122
POWER FACTOR	0,6578818
FULL LOAD EFFICIENCY	0,4897811
FULL-LOAD TORQUE (Nm)	1,236395
PULL-OUT TORQUE (Nm)	2,697954
STARTING TORQUE (Nm)	2,821325
CAPACITANCE (MicroFarad)	49,4
STARTING CURRENT (A)	9,656359

Figure 3.8 Summary of the Analysis

3.3.2.4. Optimization Menu

In the Optimization Menu, user is asked to enter 3 different file. The first file is **Constant Data** file which has the extension .con. The data is kept constant during the optimization process.

The second file holds the Initial Design of the motor. It has the extension .xve.

The third file is the Constraints file. In this file the performance and geometric specifications of the motor is defined. There are 28 inequality and 1 equality constraints in this file.

After loading these files, a new form is opened like in Figure 3.9. All the parameters of the motor to be optimized are loaded into Optimization form. There are 3 tabs on the form that contains Constant Data parameters, Initial Design and Constraints data. User can edit or change the data of the motor by clicking each of the parameters on the list view.

After editing or changing the data of the investigated data, user can save the new file that contains motor data by using "Save" and "Save As" Buttons.

The image shows a software window titled "SPCAD - [OptimizationForm]". It has a menu bar with "File", "Curve", "Analysis", and "Optimization". Below the menu bar are three tabs: "Constant Data", "Initial Design", and "Constraints". The "Constant Data" tab is active, displaying a table of motor parameters and their values.

Pull-out Torque	4
Starting Torque	6
Starting Current	16
Output Power	250
Power Factor	0,73
Efficiency	0,66
Full-load Current	4
Full-load Torque	2,6
Stator Tooth Flux Density	1,8
Rotor Tooth Flux Density	1,8
Stator Core Flux Density	1,8
Rotor Core Flux Density	1
Rotor Current Density	6
Air Gap Flux Density	1
Temperature Rise	60
Cross-sectional Area of Single Bar	210
Cross-sectional Area of End Ring	1E-05
Mean Diameter of End Ring	9E-05
Stator Slot Area	0,05
d14	1E-05
Stator Slot Top Width	0,001
Stator Slot Bottom Width	0,001
Rotor Slot Area	0,001
Rotor Outside Diameter	1E-05
Rotor Stack Length	0,05
Rotor Slot Top Width	0,05
Rotor Slot Bottom width	0,001
Yoke Depth	0,001
Stator Back-core Mmf Drop Constant	0,01

Figure 3.9 Optimization Form

After all the data is loaded then user can start the Optimization by choosing **Start Optimization** sub-item of the Optimization menu. Then a new form is opened which is called Performance Output. In this form, the optimization process is seen in every iteration. Design vector and the output of the motor are seen also on this form like in Figure 3.10. The percentages of the completed iterations are seen on the progress bar.

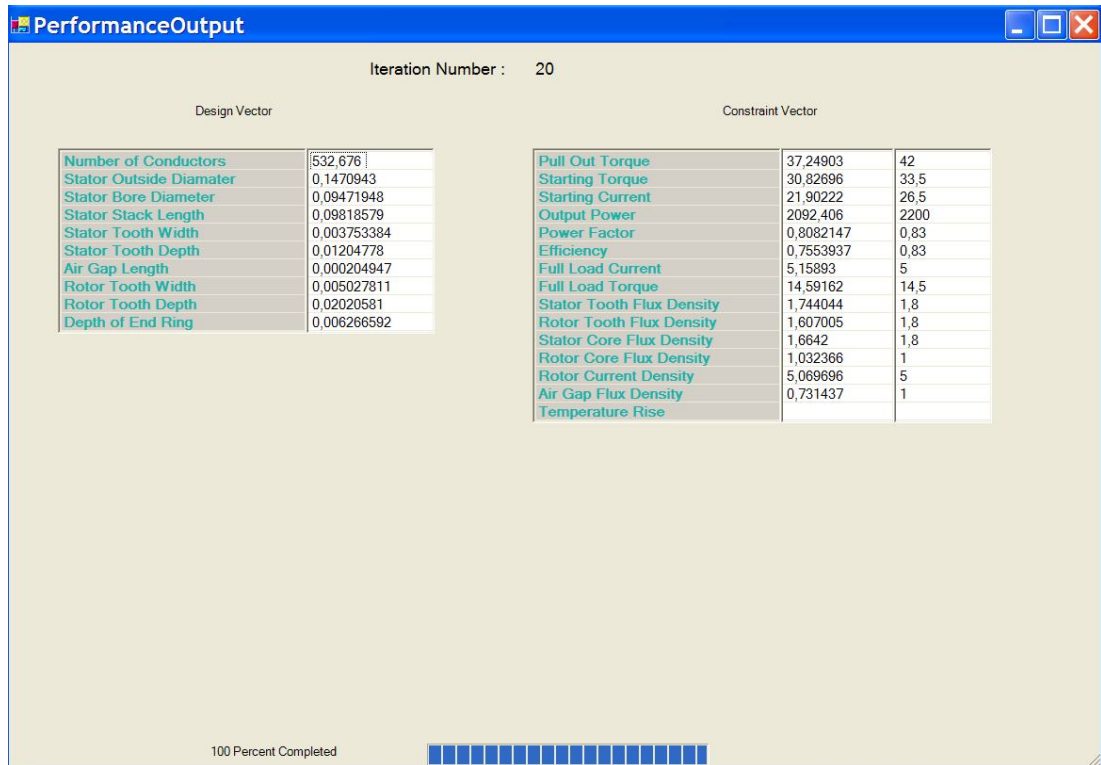


Figure 3.10 Performance Output of the Optimization Process

3.4. Results of SPCAD

3.4.1. Analysis Result

The analysis part of the SPCAD is tested for two different commercial motors. Table 3.1 shows the calculated and measured values of the motors.

Table 3.1 Test Results of SPCAD with Different Motors

	M1		M2	
Motor Performance	Test Results	Calculated Results	Test Results	Calculated Results
Full-load Output Power (Watt)	383.70	370	188.5	187
Pull-out Torque(N.m)	5.29	5.74	2.63	2.70
Starting Torque(N.m)	5.75	6.0	2.97	2.82
Starting Current(Amp)	17.2	17.67	9.32	9.65
Power factor	0.66	0.66	0.63	0.66
Efficiency	0.60	0.59	0.54	0.49
Full-load Current(Amp)	4.35	4.3	2.5	2.63
Full-load Torque(N.m)	2.54	2.45	1.24	1.24
Full-load Speed(rpm)	1436	1443	1444	1445

After calculating the analysis of the 2 different motors, the percentage errors of the motors are found and these values are shown in Table 3.2.

Table 3.2 Error Percentages of the SPCAD

	M1	M2
Motor Performance	Error %	Error %
Full-load Output Power	-3,5705	-0,79576
Pull-out Torque	8,506616	2,661597
Starting Torque	4,347826	-5,05051
Starting Current	2,732558	3,540773
Power Factor	0	4,761905
Efficiency	-1,66667	-5,76923
Full-load Current	-1,14943	5,2
Full-load Torque	-3,54331	0

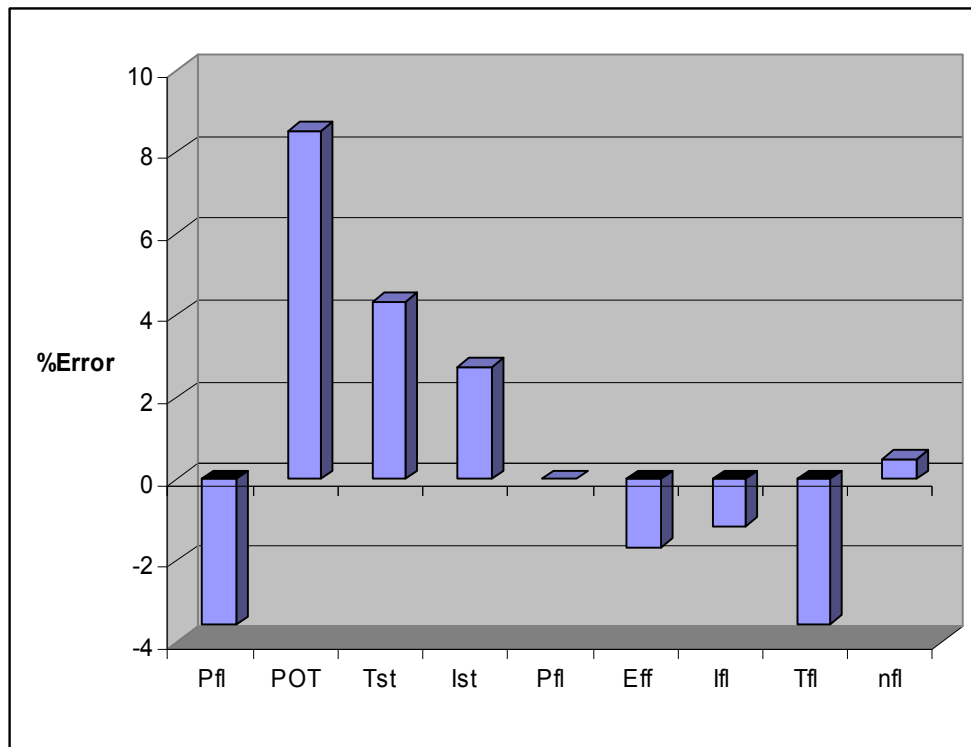


Figure 3.11 Error Percentage Graph of M1

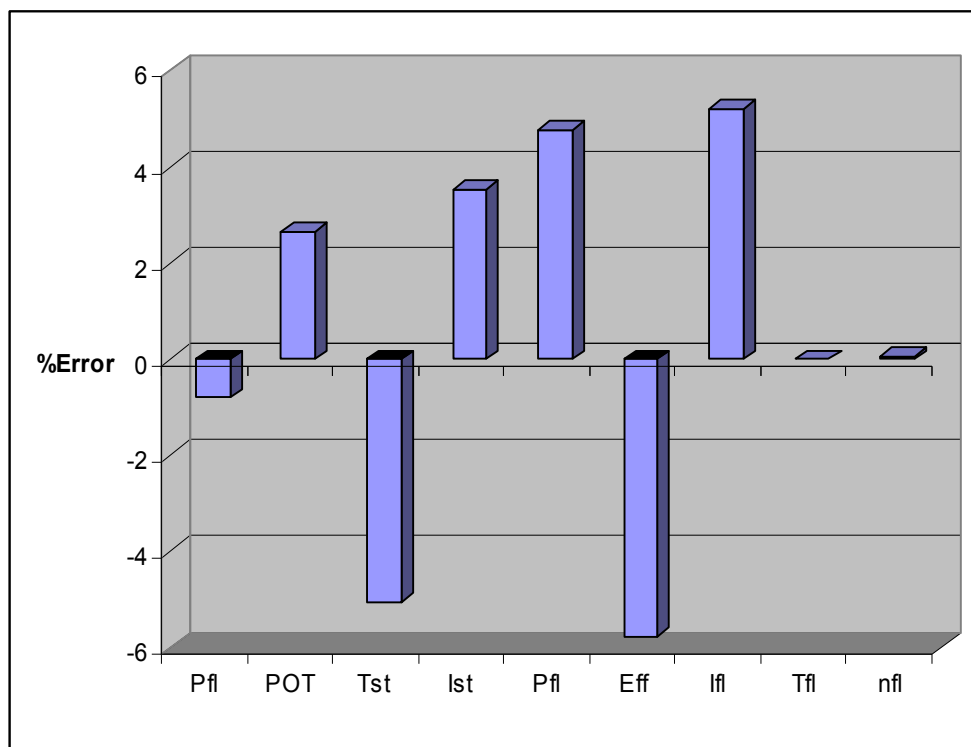


Figure 3.12 Error Percentage Graph of M2

As a result, when we look at the error percentages of the motors listed in Table 3.2, and the figures above the error of the pull out torque is high. However, if we look at the full-load conditions, the error percentage is below the 5 %. Moreover, in some cases it is nearly zero. By looking these results, we can conclude that SPCAD can predict the motor performance accurately for full-load conditions. However, the error of the pull out torque is because of the temperature rise of the rotor during the tests. The temperature rise of the rotor increases the rotor resistance. High rotor resistance results with the low torques at the tests. As a consequence, the prediction accuracy is around 10 % and the results are acceptable.

3.4.2. Optimization Result

One of the aims of this thesis is to use an optimization tool, which is named as modeFRONTIER to optimize a single-phase induction machine. For this purpose, a project is designed in modeFRONTIER like seen in Figure 3.13. Design steps of project at modeFRONTIER and knowledge about the modeFRONTIER is explained in Section 4.8.5 of this study.

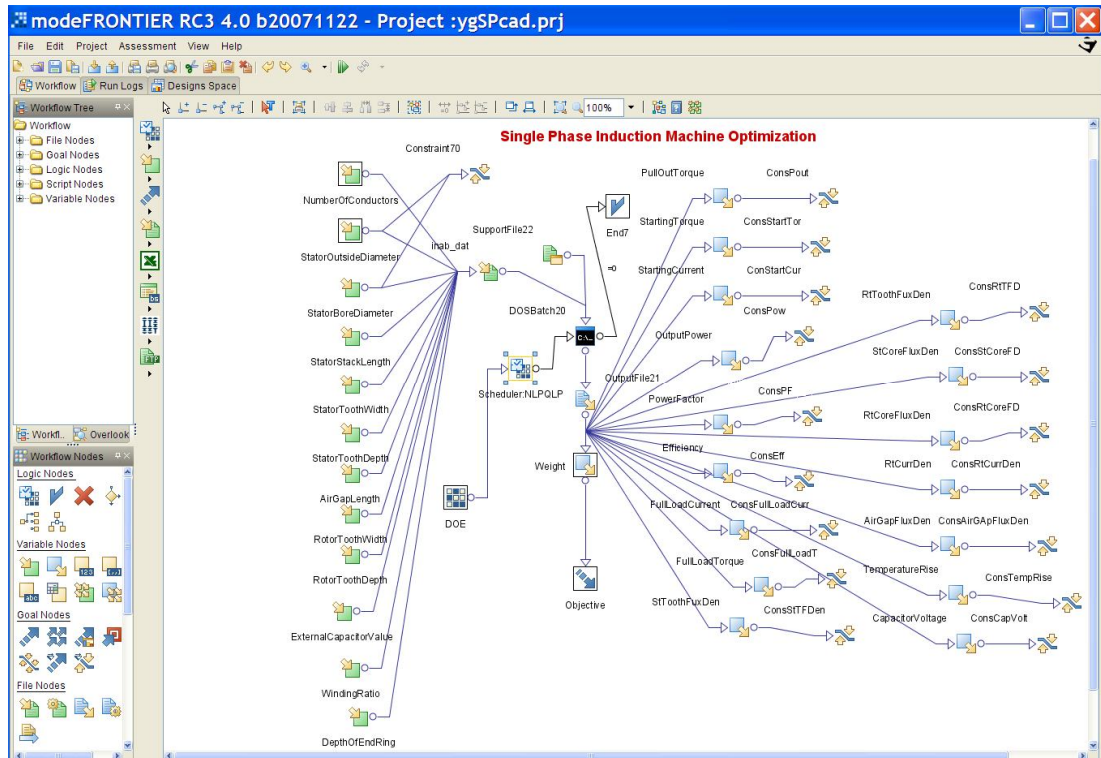


Figure 3.13 Designed Project at modeFRONTIER for the Single Phase Induction Machine

In this section, different optimization methods such as SIMPLEX, B-BGFS, Levenberg-Marquardt and MOSA are applied to problem to obtain an optimal design of Single-Phase Induction Machine. But, only one method gave the feasible results. This method is the NLPQLP method. Commercial available motor, 77C14, is used for test of the modeFRONTIER project.

3.4.2.1. Results of NLPQLP

For the optimization of Single-Phase Induction Machine, NLPQLP Algorithm is used.

The details about the NLPQLP algorithm is given in Section 2.6.2 where Three-Phase Induction machine optimization is discussed.

To start the optimization with NLPQLP algorithm 3 different initial conditions are specified. I1 is the same size as the reference motor. I2 is 37% larger in weight than the reference motor. I3 is 20% lighter than the reference motor. Table 3.3 summarises the initial conditions and the final design vectors obtained. Table 3.4 presents the performance of both the initial designs and final designs. The optimization conditions are specified below.

Parameters of NLPQLP Algorithm:

Maximum Number of Iterations	: 30
Approximate Derivatives with	: Central Differences
Final Termination Accuracy	: 1.0E-5
Finite Difference Relative Perturbation	: 1.0E-5
Finite Difference Minimum Perturbation Policy	: Percentage of Range
Range Percentage Minimum Perturbation	: 0.001

Table 3.3 Initial and Final Designs of 187 W,4 pole single phase induction motor by using NLPQLP

Design Vector	I1	F1	I2	F2	I3	F3
Number Of Conductors	1568	1568	1368	1368	1450	1450
Stator Outside Diameter (mm)	124.58	114.4	134.58	108.05	112	111.95
Stator Bore Diameter(mm)	73	71.1	83	67.25	66	70.3
Stack Length (mm)	48	61	58	76.8	48	68.88
Stator Tooth Width (mm)	3.33	2.81	5.6	2.59	3.5	2.74
Stator Tooth Depth (mm)	13.3	11.08	13.3	10.56	12	10.48
Air Gap Length (mm)	0.3	0.23	0.3	0.227	0.3	0.229
Rotor Tooth Width (mm)	2.25	1.87	2.25	1.76	2.25	1.89
Rotor Tooth Depth (mm)	9.85	16.88	9.85	15.38	8.85	16.11
Depth of End Ring (mm)	12.18	13.19	12.18	13.53	12.18	10.95
Capacitor Value (microF)	49.4	49.47	49.4	49.41	49.4	49.44
Winding Ratio	1.25	0.78	1.25	0.717	1.25	0.73
Weight(kg)	4.547	4.655	6.231	5.051	3.69	4.909
Number of Iterations	---	736	---	468	---	650

By looking the results, it can be said that NLPQLP algorithm is successful for optimizing the motor weight while satisfying the constraints.

Table 3.4 Results of Optimization of 187 W,4 pole single phase induction motor by NLPQLP

Performance	I1	F1	I2	F2	I3	F3	Constraints
Output Power (W)	164*	200	220	200	91.5*	200	>200
Pull-Out Torque (Nm)	3.1	3.31	2.89*	3.51	2.41*	3.42	>3
Starting Torque (Nm)	3.71	3.0	2.46*	3.0	2.92*	3.0	>3
Starting Current (Amp)	10.24*	10	9.86	10	9.69	9.95	<10
Power Factor	0.59*	0.73	0.73	0.70	0.51*	0.72	>0.64
Efficiency	50.1*	60	55.7*	60	22.4*	60.4	>60
Full-Load Current (Amp)	2.51	2.07	2.44	2.16	3.64	2.08	<4
Full-Load Torque (Nm)	1.09*	1.33	1.46	1.33	0.61*	1.33	>1.25
Stator Tooth Flux Density (T)	1.68	1.56	0.95	1.54	1.73	1.54	<1.8
Stator Core Flux Density (T)	1.58	1.53	1.41	1.50	1.90*	1.50	<1.8
Rotor Tooth Flux Density (T)	1.69	1.59	1.60	1.54	1.83*	1.52	<1.8
Rotor Core Flux Density (T)	0.67	0.83	0.52	0.78	0.82	0.78	<1.0
Rotor Current Density(A/mm ²)	4.44	2.97	3.69	3.04	4.96	2.95	<6.0
Air Gap Flux Density (T)	0.80	0.64	0.66	0.62	0.95	0.62	<1.0
Capacitor Voltage (V)	176	204	155	213	176	210	<300
Weight (kg)	4.547	4.655	6.231	5.05	3.69	4.909	Minimize
Number of Iterations	---	736	---	468	---	650	---

In Table 3.4, it can be observed that for the initial design I1, output power, efficiency, power factor and full-load torque is lower than the desired values. After 736 steps of iteration of the optimization process, final optimum value is reached. It is seen that for the F1, all the constraints are satisfied and also weight converges to a minimum. The final design is about 2.3 % heavier than the reference motor (I1).

For Initial Design 1:

- Output Power : 164 W, Not satisfied, percentage error is 18 %, this error is high, it is not acceptable.
- Pull-Out Torque : 3.1 Nm, satisfied.
- Starting Torque : 3.71 Nm, satisfied.
- Starting Current : 10.24, Not satisfied, percentage error is 2.4 %, this error is in acceptable range.
- Power Factor : 0.59, Not satisfied, percentage error is 7.8 %, this error is high, it is not acceptable.
- Efficiency : 50.1, Not satisfied, percentage error is 16.5 %, this error is high, it is not acceptable.
- Full-Load Current : 2.51 A, satisfied.
- Full-Load Torque : 1.09 Nm, Not satisfied, percentage error is 12.8 %, this error is high, it is not acceptable.
- Stator Tooth Flux Density : 1.68 T, satisfied.
- Stator Core Flux Density : 1.58 T, satisfied.
- Rotor Tooth Flux Density : 1.69 T, satisfied.
- Rotor Core Flux Density : 0.67 T, satisfied.
- Rotor Current Density : 4.44 A/mm², satisfied.
- Air Gap Flux Density : 0.80 T, satisfied.
- Capacitor Voltage : 176 V, satisfied.
- Weight : 4.547 kg, motor is the reference motor.

For Final Design 1:

- All the constraints are satisfied.
- Weight : 4.655 kg, larger than the reference motor with a percentage of 2.3%
- Time Elapsed : 736 seconds

For the initial design I2, pull-out torque, efficiency and starting torque is lower than the desired values. At the end of the 468 iterations, NLPQLP method converges and also satisfies the constraints although the initial point of the weight is larger than the reference motor. This final design is about 11 % larger than the reference motor weight.

For Initial Design 2:

- Output Power : 220 W, satisfied.
- Pull-Out Torque : 2.89 Nm, Not satisfied, percentage error is 3.67 %, this error is in the acceptable range.
- Starting Torque : 2.46 Nm, Not satisfied, percentage error is 18 %, this error is high, it is not acceptable.
- Starting Current : 9.86 A, satisfied.
- Power Factor : 0.73, satisfied.
- Efficiency : 55.7, Not satisfied, percentage error is 18 %, this error is high, it is not acceptable.
- Full-Load Current : 2.44 A, satisfied.
- Full-Load Torque : 1.46 Nm, satisfied.
- Stator Tooth Flux Density : 0.95 T, satisfied.
- Stator Core Flux Density : 1.41 T, satisfied.
- Rotor Tooth Flux Density : 1.60 T, satisfied.
- Rotor Core Flux Density : 0.52 T, satisfied.

- Rotor Current Density : 3.69 A/mm², satisfied.
- Air Gap Flux Density : 0.66 T, satisfied.
- Capacitor Voltage : 155 V, satisfied.
- Weight : 6.231 kg, motor is more heavier than the reference motor with a percentage of 37 %.

For Final Design 2:

- All the constraints are satisfied.
- Weight : 5.05 kg, larger than the reference motor with a percentage of 11%
- Time Elapsed : 468 seconds

For the initial design I3, in this case most of the constraints are not satisfied. The initial weight is about 80 % of the reference motor weight. At the end of the 650 iterations, it also converges to a minimum value by satisfying the constraints. This time the final design is about 8 % larger than the reference motor in weight.

For Initial Design 3:

- Output Power : 91.5 W, Not satisfied, percentage error is 54 %, this error is high, it is not acceptable.
- Pull-Out Torque : 2.41 Nm, Not satisfied, percentage error is 20 %, this error is high, it is not acceptable.
- Starting Torque : 2.92 Nm, Not satisfied, percentage error is 2.7 %, this error is in acceptable range.
- Starting Current : 9.69 A, satisfied.
- Power Factor : 0.51, Not satisfied, percentage error is 20 %, this error is high, it is not acceptable.

- Efficiency : 22.4 %, Not satisfied, percentage error is 63 %, this error is high, it is not acceptable.
- Full-Load Current : 3.64 A, satisfied.
- Full-Load Torque : 0.61 Nm, Not satisfied, percentage error is 51 %, this error is high, it is not acceptable.
- Stator Tooth Flux Density : 1.73 T, satisfied.
- Stator Core Flux Density : 1.90 T, Not satisfied, percentage error is 5.5 %, this error is in the acceptable range.
- Rotor Tooth Flux Density : 1.83 T, Not satisfied, percentage error is 1.67 %, this error is in the acceptable range.
- Rotor Core Flux Density : 0.82 T, satisfied.
- Rotor Current Density : 4.96 A/mm², satisfied.
- Air Gap Flux Density : 0.95 T, satisfied.
- Capacitor Voltage : 176 V, satisfied.
- Weight : 3.69 kg, motor is lighter than the reference motor with a percentage of 19 %. But since the other constraints are not satisfied, this motor is not feasible.

For Final Design 2:

- All the constraints are satisfied.
- Weight : 4.909 kg, larger than the reference motor with a percentage of 8 %.
- Time Elapsed : 650 seconds

By looking the results, it can be said that program is successful for optimizing the motor weight while satisfying the constraints. If we compare the results according to weight, it can be observed that although the initial designs do not satisfy the constraints, they all converge to a minima. It appears that this algorithm gives successful results for the SR motor optimization problem.

3.4.3. Conclusion Of Optimization

It is observed that NLPQLP algorithm for the optimization of the single-phase induction machine is the most suitable algorithm. For design optimization of single phase induction motor, NLPQLP algorithm can find minimum weight, but as in the case in the optimization of three-phase induction machine, the initial designs of the algorithm should be properly defined. Various initial designs should be experimented and the results should be assessed using experience to reach a successful design.

It can be observed that the design optimization takes in the order of 10 minutes on a dual core processor computer. This performance is very suitable for a design office.

3.5. Conclusion of SPCAD

A software tool SPCAD that can be used for single-phase induction machine design is developed. This tool also is tested with using 2 different commercial motors. Comparison of simulations with the test results illustrates that the performance prediction for the full load conditions for the test motor lie in range in 5 % range with the actual motors.

For finding an optimum design, modeFRONTIER optimization algorithm NLPQLP is applied to problem. NLPQLP algorithm is found to be the most suitable optimization algorithm for the single-phase induction machine design as in the case three-phase induction machine design.

Finally, it is illustrated here that SPCAD is an acceptably accurate single phase induction machine design and optimization tool, and also it leads to quick results.

CHAPTER 4

SWITCHED RELUCTANCE MOTOR COMPUTER AIDED DESIGN (SRCAD)

4.1. Introduction

In this study, a software program is developed for the analysis of a given SR (Switched Reluctance) motor and then for the design and optimization of switched reluctance motors.

For the analysis and prediction performance of SR motors, the approach presented in the reference [13] is used. The software developed is used to predict the performance of an 8/6, stepping 4-phase motor. The simulation results are compared with measurements on the test motor and the accuracy of the analysis software is illustrated.

The next stage is the design optimization. This involves seeking a design vector (a set of motor dimensions) which satisfy a set of constraints and satisfy design criteria such as achieving minimum volume, cost or any other desired objective function. For mathematical design optimization, various optimization techniques exist. The optimum design part of the software is developed using modeFRONTIER package, which is briefly described in Section 4.8.5. modeFRONTIER calls the performance calculation modules described in the analysis section to calculate the performance of a given design vector. Using this software, different optimization methods are attempted. The results are presented in the following sections.

As discussed in Section 4.4, analysis of an SR motor is based on flux linkage –current-position curves of a given motor. Therefore, an important aspect of the software developed is prediction of the flux linkage curves of a motor described by a design vector, which is a set of dimensions. Clearly, the user should also define the material properties of the motor to get an accurate result.

4.2. Basic Structure of the Switched Reluctance Motor

First, a brief introduction to SRM and its basic structure and principle of operation is explained. Although the concept variable reluctance motor is very old, this motor reappeared in 1969, as an alternative for variable speed applications. This motor has some important features. Basically, this motor is a type of synchronous machine. It has coils for stator windings but has no coils or permanent magnets on its rotor, also, both the rotor and stator have salient poles. The basic magnetic geometry with the stator and rotor pole numbers of 8-6 are shown in Figure 4.1. The number of stator poles is not equal to the number of rotor poles in order to continuous rotation of rotor.

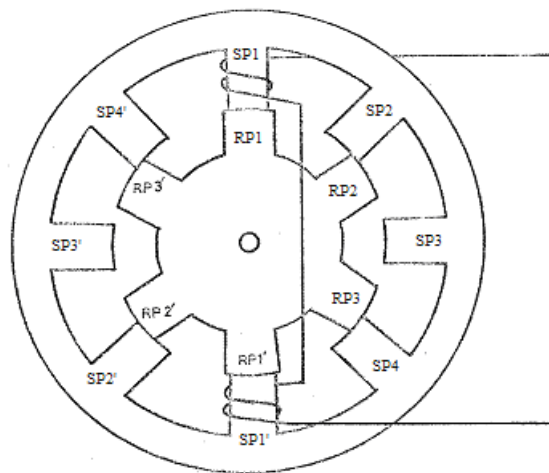


Figure 4.1 Basic Structure for Four Phase SRM

In Figure 4.1, the permeance of the magnetic circuit for pole pair SP1- SP1' is maximum, but for SP3-SP3', the permeance is minimum for the position of the rotor. If the SP3-SP3' pole is excited then rotor comes to aligned position for SP3-SP3' pole, and permeance become maximum for SP3-SP3' and minimum for SP1-SP1'. Likewise, by sequentially switching the currents into the stator windings, the rotor is rotated. The movement of the rotor, hence the production of torque and power, involves switching of currents into stator windings when there is a variation of reluctance; thus, this variable speed motor drive is referred to as a switched reluctance motor drive [4].

4.3. Technical Explanation of SRCAD

SRCAD (Switched Reluctance Motor Computer Aided Design) is a personal computer (PC) windows application software product designed for design and optimization of switched reluctance motors. SRCAD is also developed to perform the following tasks:

- Predicting the flux-linkage curves of SR motor of given dimensions and specified materials.
- Analysis of an SRM using measured flux linkage-current-position curves to obtain steady – state performance of the SR motor. The drive strategy is taken into account by allowing user to define turn on instant and conduction angle. The outputs of SRCAD are:
 - Average torque
 - Phase currents and current density
 - Output power, input power, and efficiency
 - Torque ripple
 - Temperature rise
- Design optimization for cost or weight minimization while satisfying a set of design criteria. This feature allows the user to seek up

to 10 independent motor dimensions, within the set of design criteria. User can design a specified motor, starting with an initial design.

- Ability to define the core-loss curve and B-H curve of the core material.
- Plotting and printing of outputs.

With the above characteristics of SRCAD, analysis of an existing motor under various operating conditions can be made and a new motor can be designed. In addition, the effect of the materials used in manufacturing the motor on the performance can be investigated.

In this thesis, since the analysis part of the SRCAD is developed earlier [11], the emphasis is placed on developing the optimization part of the SRCAD. In addition, documentation of the analysis algorithms is prepared.

For the calculation of SR motor performance Flux-Linkage-Current Curves are essential. So any design process needs to calculate these curves as a function of independent variables (motor dimensions). This work also presents an approach for the calculations of the said curves.

The optimization process involves repeated calculation of the motor performance for a given design vector. Therefore, the first issue is the calculation of the Flux-Linkage-Current Curves for a given design vector. This approach must be fast enough, so that the computation time is short enough to use within the optimization routine. The approach used in this work has been explained in the next section. The next issue is the computation of the motor performance. In the optimization process, this is done using the methods developed for the analysis part of SRCAD. As discussed earlier the analysis routine computes the Average Torque, Phase Currents and Current Density,

Output Power, Input Power, Efficiency, Torque Ripple and Weight of the SRM according to input dimensions and operating conditions.

For optimization, a software package called "modeFRONTIER" is used, and a new GUI is designed for SRM design optimization. The interface calls the modeFRONTIER project that we have developed for the optimization of the SRM. After the modeFRONTIER project is opened, project calls the program that we have developed for analysis part and uses it to calculate SR motor performance for a given design vector. modeFRONTIER has different optimization routines embedded in it. This gives the user the opportunity to solve the problem with different algorithms. An investigation of the suitability of various algorithms is also presented in this section.

SRCAD also has a user-friendly GUI (Graphical User Interface). The GUI of SRCAD is developed in CodeGear RAD Studio Environment using the Pascal object oriented language. CodeGear RAD Studio Environment helps the programmer to develop user-friendly interfaces in a fast way by supplying all necessary design tools such as Windows forms, buttons, graphic plotters, list view.

In the next section, the analysis routines used in the optimization process will be briefly described.

4.3.1. The Analysis Section of SRCAD

The SRM has a highly non-linear behavior because of the non-linear characteristics of the magnetic materials used in SRM structure and dependency of flux-linkage to both current and rotor position. For this reason, the simulation of SRM is complex.

For the calculation of the steady state performance (average torque, etc), flux linkage characteristics should be determined as a function of current level and rotor position should be determined.

After the determination of the flux linkage characteristics, steady-state performance of the SRM should be calculated.

After obtaining the steady state performance, the core and copper losses should be obtained. These calculations are essential for the estimation of the efficiency of the motor.

SR motor performance depends on the drive topology and control strategy because of the dependency of the current to flux. Therefore, in this work SR motor is assumed to be driven by a chopper driver like given in Figure 4.2 for one of the phases.

There are four states of the chopper driver [13]:

Conduction period, during this period T1 is ON, T2 is ON, $V = V_{in}$

Free – wheeling period, during this period T1 is ON, T2 is OFF, $V = 0$

Fast – decay period, during this period T1 is OFF, T2 is OFF, $V = -V_{in}$

Off state

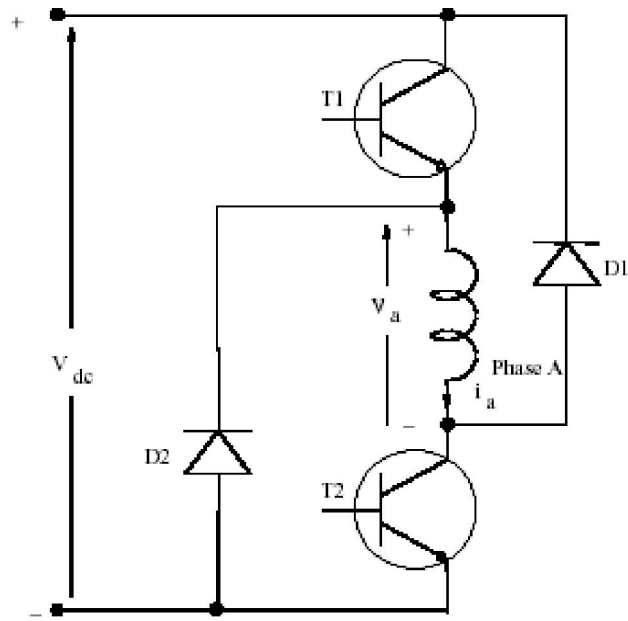


Figure 4.2 Asymmetric Converter for One Phase of SRM [13]

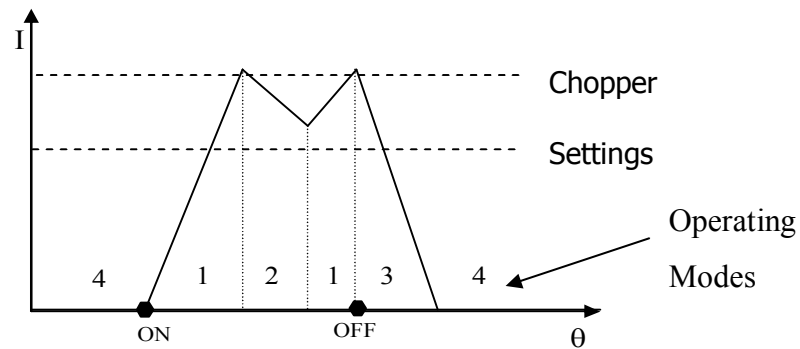


Figure 4.3 Operation of Converter on Phase Current Waveform [13]

4.4. Determination of Flux Linkage – Position – Current Curves

Relationships between the flux linkages vs. rotor position, as a function of the machine phase currents is the requirement for finding the performance characteristics of the SRM. The flux linkage characteristics and performance of the SRM can be computed by finite element analysis techniques, but this is not a viable option for an optimization process. In this kind of approach the relationship of motor output variables to motor dimensions, number of

poles, number of turns per phase, excitation current and current conduction angle is not explicit. Therefore, a change in one or many of the motor and control variables requires renewed finite element analysis computations, which is either in two or three dimensions. Each set of finite element computations takes a considerable amount of time. [4]. So finite element analysis computation is not suitable for design optimization, since the calculation of performance is required to be done many times during this process. Analytical methods are more suitable for optimization of SRM than finite methods. There are lots of analytical methods that are developed by Corda and Stephenson [6], Krishnan [4], Mukherji and Neville [5], Jones [9], Chai [10], Rajagopal [18] and Ertan [1], [2], [3]. In this study, the analytical solution, which is proposed by Ertan [2], will be used. Ertan computed and tabulated normalized permeance and force per unit length for a doubly salient geometry with equal and identical rotor and stator teeth as being functions of the following variables.

- λ/g = Rotor tooth pitch / air gap, (values 40, 70, 100, 150, 200, 250),
- t/λ = tooth width / tooth pitch, (values 0.3, 0.4, 0.5)
- X_n = Normalized positions of rotor, where $X = 0$ is IN position, and $X_n = 1$ is OUT position, (values 0, 0.2, 0.4, 0.6, 0.8, 1)

Data are computed for $\lambda = 0.0172$ m, and core length $L = 1$ m, and slot depth = 40.airgap. [7]. A typical B-H characteristics is used in the computations; so that the saturation of the material is taken into account.

This approach is based on assuming that the force (torque) contributions of individual pairs of teeth can be added to compute the torque produced by the motor.

The magnetic structure is viewed as if it is divided into two sections, the teeth region and the rest of the magnetic structure (Figure 4.5). In the teeth region, the flux distribution is non-uniform and the teeth may be heavily saturated. The rest of the core is assumed to have a uniform flux distribution.

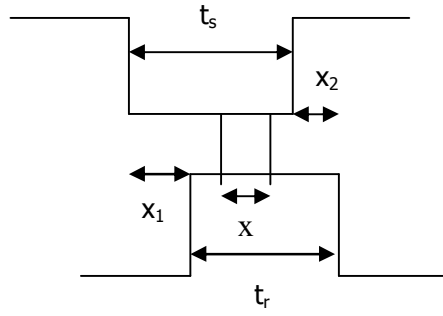
For a given set of dimensions, at a given position, teeth region flux linkage vs. current characteristics can be obtained using the data provided in [2]. For the rest of the core (back iron) flux linkage – current characteristics can be obtained using the B-H curve of the material used.

One can plot the two characteristics back to back as shown in Figure 4.9. For a given excitation level, F_e , an operating point can be found ie, teeth region MMF and flux linkage for the particular operating point. Once the flux linkage characteristics is known it is a simple matter to extract from the data the corresponding force produced by the pair.

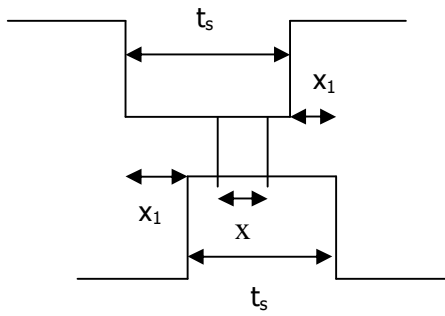
In the next section, the calculation procedure is explained. In the description, stator and rotor teeth are assumed asymmetrical to consider the most general case. The flow chart in Figure 4.10 describes the general procedure for this calculation.

4.4.1. Teeth Region Flux-Linkage MMF Characteristics

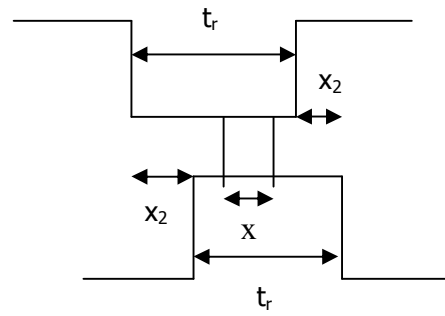
The permeance data is calculated for symmetrical tooth pairs. However, the permeance of an asymmetrical tooth pairs can be calculated using the method developed by Ertan. In this approach, the asymmetrical structure is divided into two symmetrical structures like Geometry A and Geometry B as in Figure 4.4. This allows us to predict the Permeance vs. MMF for Geometry A and B. Figure 4.13 shows the determination algorithm of Permeance vs. MMF for Geometry A. The algorithm for Geometry B is the same as the algorithm for Geometry A.



a) Asymmetrical geometry



b) Geometry A



c) Geometry B

Figure 4.4 Asymmetrical Structure and Related Symmetrical Structures

Suppose the saliency considered is asymmetrically slotted as shown in Figure 4.4. Two symmetrically slotted geometries, A and B, are obtained from the asymmetrically toothed geometry, one with tooth width t_s and other with tooth width t_r . The tooth pitch of geometries A and B are:

$$\lambda_A = \max(\lambda_r, t_r + X_2 + 25g)$$

$$\lambda_B = \max(\lambda_r, t_s + X_1 + 25g)$$

(4.1)

The permeance of both geometries can be found from existing permeance data, and the permeance of the asymmetrically toothed geometry is:

$$P_n = \frac{2P_{na}P_{nb}}{P_{na} + P_{nb}} \quad (4.2)$$

Where, P_{na} and P_{nb} are the normalized permeance values of obtained symmetrical geometries A and B respectively.

4.4.2. Determination of Back Iron Flux Linkage MMF Characteristic

After finding the flux linkage characteristics for the tooth region, back iron flux linkage characteristic is calculated by assuming a uniform flux distribution in this section of the core. The procedure developed is shown in the flowchart in Figure 4.14. The back iron region MMF for a given flux linkage is calculated, by separately calculating the MMF drops on;

- Stator back tooth
- Rotor back tooth
- Stator back core
- Rotor back core which are shown in Figure 4.5.

The flowchart in Figure 4.14 gives the procedure used for the calculation. The procedure starts with an assumed value of flux linkage per pole.

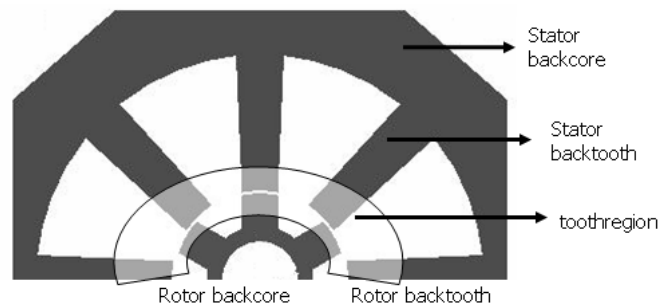


Figure 4.5 Back Iron Regions of SR motor

4.4.3. Determination of Pole MMF Drop

For the sake of generality, a tapered stator or rotor pole is considered. The algorithm that shows the calculation of Flux vs. MMF Drop at the stator back tooth is in Figure 4.15. Here the procedure is described for a stator pole but same procedure can be used for the rotor pole.

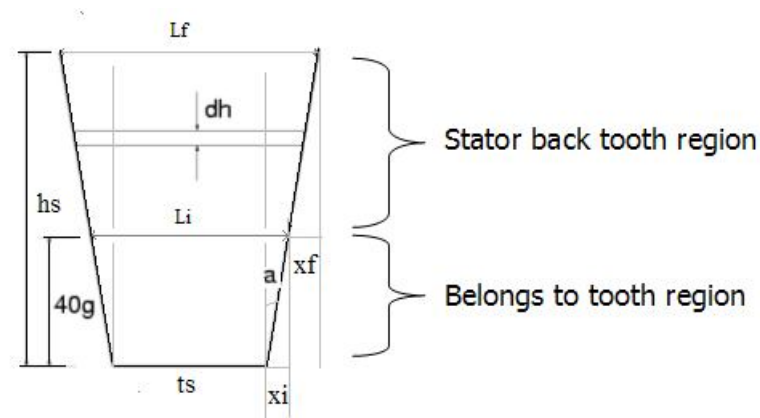


Figure 4.6 Stator Tooth

To calculate the MMF Drop at the stator pole back core region (Figure 4.6), tapering angle defined by the user should be taken into account. For this purpose, tooth region is divided into n pieces as shown in Figure 4.6. In this study, n is chosen as 30. Therefore, the length of the pieces is:

$$\Delta h = \frac{h_s - 40g}{30} \quad (4.3)$$

and the width of the m^{th} piece is:

$$t_i = t_s + 2m\Delta h \tan(a) \quad (4.4)$$

The total MMF drop at the stator back tooth is the sum of the MMF drops of all the pieces. Therefore, to find the MMF drop at the pieces, firstly, flux density is found by using equation below;

$$B_i = \frac{\Phi}{t_i \cdot L_c} \quad (4.5)$$

where;

L_c : core length

t_i : the width of the m^{th} piece

Then H_i value corresponding to B_i is found by using the BH curve. Then total MMF drop is equal to:

$$F_{sbt} = \sum_{i=1}^n H_i \Delta h \quad (4.6)$$

For the rotor back tooth region, it is assumed that there is no tapering angle. So MMF drop in rotor back tooth region is found by:

$$B_{rbt} = \frac{\phi}{t_r L_c} \quad (4.7)$$

Then, find H_{rbt} from the B-H curve.

$$F_{rbt} = \int H_{rbt} dl = H_{rbt} \cdot (h_r - 40g) \quad (4.8)$$

4.4.4. Determination of Back Core MMF Drop

Flux vs. MMF Drop calculation at the stator back core is explained in Figure 4.16. The same algorithm is applied for the rotor back core.

In the stator and rotor backcore, flux is divided into two equal, path because uniform flux density is assumed as seen in Figure 4.7. Flux density for stator back core is found as using equation below.

$$B_{sbc} = \frac{\phi/2}{y_b L_c} \quad (4.9)$$

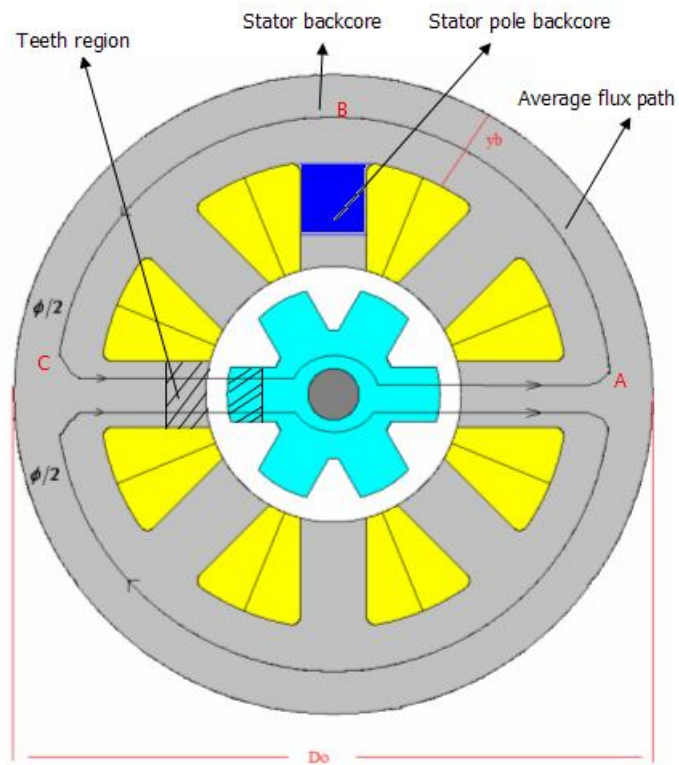


Figure 4.7 Cross Section of a 8/6 Switched Reluctance Motor

After finding the flux density, using the B-H curve of the material used in manufacturing the core, the value of H is found. The MMF drop in the particular region is $H_i \cdot l_i$, where l_i is the length of the flux path in the particular region.

In the stator back core, the length of average flux path which is denoted by ABC in the Figure 4.7 is given by

$$l_i = \pi \cdot (D_0 - y_b) / 2 \quad (4.10)$$

However, to take the flux path at the cross section of the tooth and backcore into account, the diameter is chosen as D_0 in this study.

Then MMF Drop at the stator back core is:

$$F_{sbc} = \int H_{sbc} dl = H_{sbc} \cdot (\pi \cdot D_0 / 2) \quad (4.11)$$

Also for the rotor back core the same procedures are applied as the stator case;

$$B_{rbc} = \frac{\phi / 2}{((d_u - d_a) / 2) \cdot Lc} \quad (4.12)$$

where;

d_u : rotor inside diameter

d_a : shaft diameter

Lc : Core length

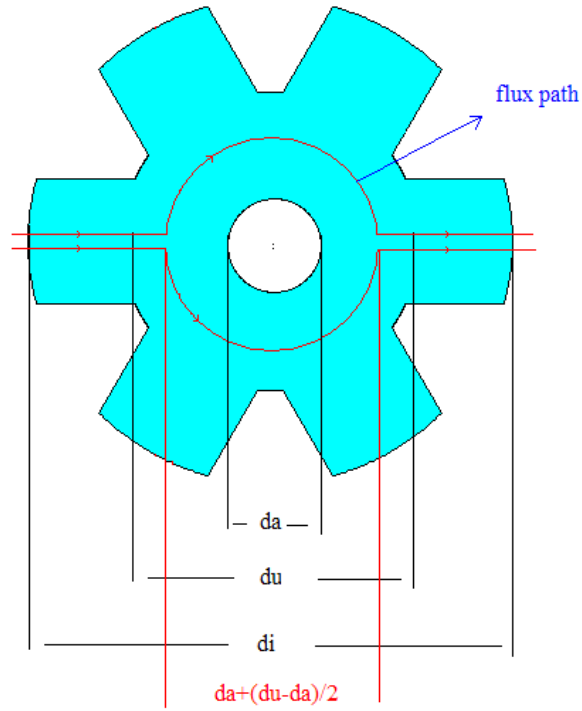


Figure 4.8 Rotor

The total MMF drop for the back iron is found by summing up the MMF drops at the stator pole and rotor pole back core and rest of the stator and rotor back core that have been calculated above. So total Back iron MMF drop is:

$$F_{Bi} = 2(F_{rbl} + F_{sbl}) + F_{rbc} + F_{sbc} \quad (4.13)$$

4.4.5. Determination of the Operating Point

A particular position of rotor teeth is chosen. Tooth region flux linkage curve is constructed using the data available as described in Section 4.4. As shown in Figure 4.9, this is plotted back to back with the back iron MMF characteristics obtained as in Section 4.4.4.

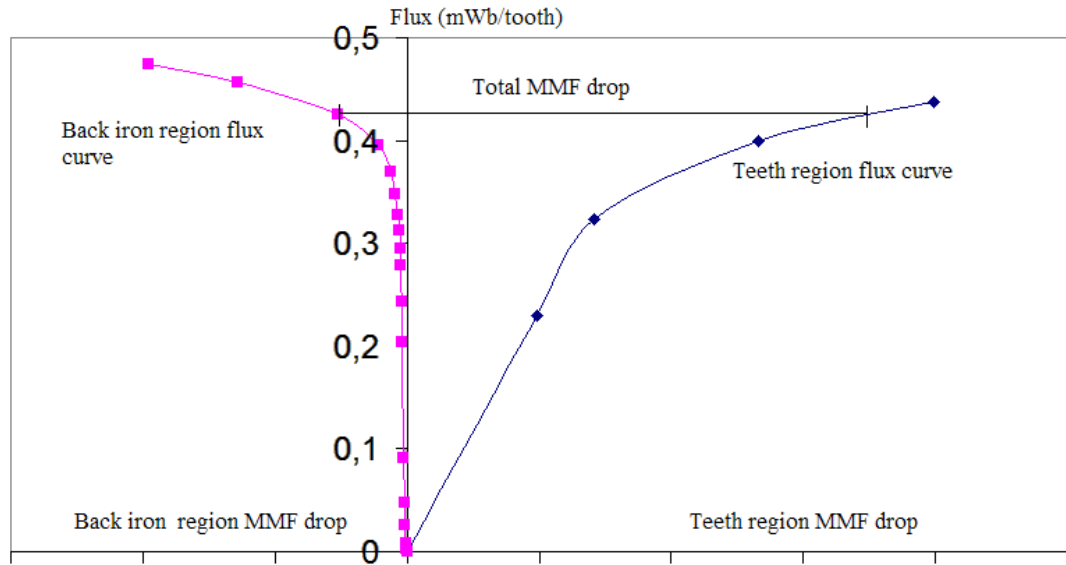


Figure 4.9 Operating Point Determination

In order to find the flux passing through the poles, operating point calculation between the tooth region and back iron characteristics is found by applying the algorithm shown in Figure 4.17. In this algorithm, bisection method is applied. The essence of the approach is assuming a pole MMF value and determining the corresponding flux linkage for the phase winding. This is repeated for a number of pole MMF values to determine the flux linkage vs. MMF curve of the motor for a given position. The procedure is repeated for different positions ($X_n=0, 0.2, 0.4, 0.6, 0.8, 1.0$) to determine a family of curves as shown in Figure 4.10.

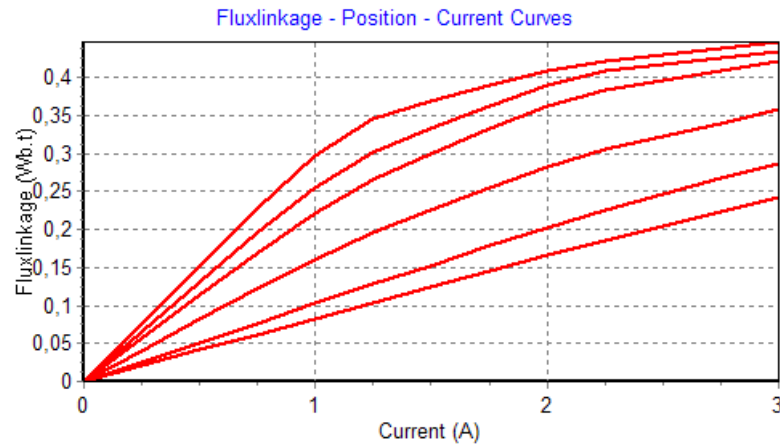


Figure 4.10 Flux Linkage - Position - Current Curves

After then, end-leakage correction is applied to generate the Flux-Linkage vs. Current vs. Position curve of the specified motor.

4.4.6. End Leakage Correction

In the prediction procedure described above, only a two dimensional field solution is taken into account. However, at the two ends of the core, flux leakage occurs, which are at significant amounts especially in small overlap positions. If this is not accounted for in predicting the flux linkage curves, significant errors occur in calculating the performance analysis of motor. To include this effect into flux-linkage curves prediction method, an analytical method, which is based on the approach [13], is used. This approach is based on using Carter's coefficient, σ in order to obtain an effective core length, L_{cf} .

In this approach, side view of the magnetic circuit is considered. To account for the end leakage field, a series of infinitely permeable magnetic bodies of core length L_c are placed with a gap of $2n$. In this approach, n is chosen as 40g in order not to cause a modification of individual leakage field of cores.

Figure 4.11 b shows the equivalent magnetic circuit with core length L_{cf} , which accounts for the end leakage.

$$L_{cf} = L_c + 2n(1 - \sigma) \quad (4.14)$$

$$\sigma = \frac{2}{\pi} \left\{ \arctan\left(\frac{2n}{g_F}\right) - \frac{g_F}{4n} \ln \left[1 + \left(\frac{2n}{g_F} \right)^2 \right] \right\} \quad (4.15)$$

where, σ is the Carter's coefficient, g_F is the effective air gap length, and n is the length where end leakage fluxes in both sides where their magnitude drops to 1% of peak value.

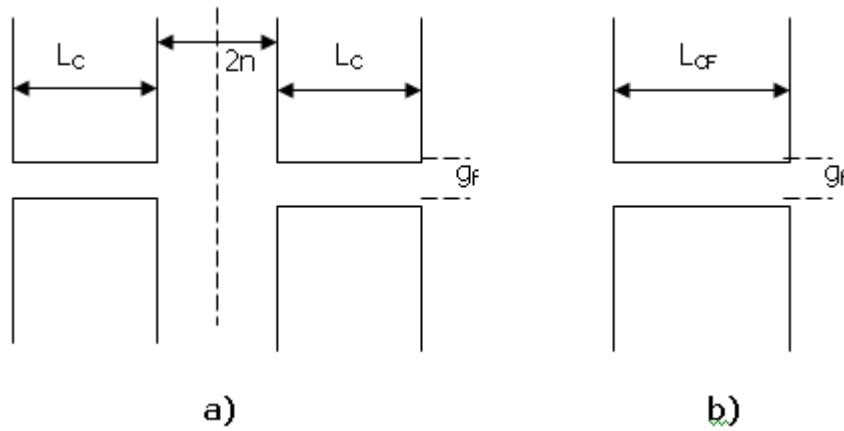


Figure 4.11 a) Mirror Image to Obtain an Equivalent Geometry Including End Leakage Flux
b) Equivalent Geometry

The above approach leads to good results when teeth are aligned. The situation becomes more complicated when the teeth are displaced from each other. The best approach for handling such cases is to obtain an equivalent magnetic circuit where the saliencies are aligned.

Suppose the unaligned structure has tooth pitch λ , tooth width t , airgap g , average flux density B_t and normalized permeance P_1 . To find the aligned structure with the same tooth pitch λ , tooth width t , permeance P_1 and same B_t value, the normalized permeance data is interpolated and new λ/g value is found. Then g_f value can be found by;

$$g_f = \frac{\lambda}{\left(\frac{\lambda}{g}\right)_{NEW}} \quad (4.16)$$

After finding the aligned structure, the effective core length L_{cf} is determined by using the Equation 4.14. Then the flux linkage including the end effect will be;

$$3D \text{ flux linkage} = 2D \text{ flux linkage} \times (L_{cf}/L_c)$$

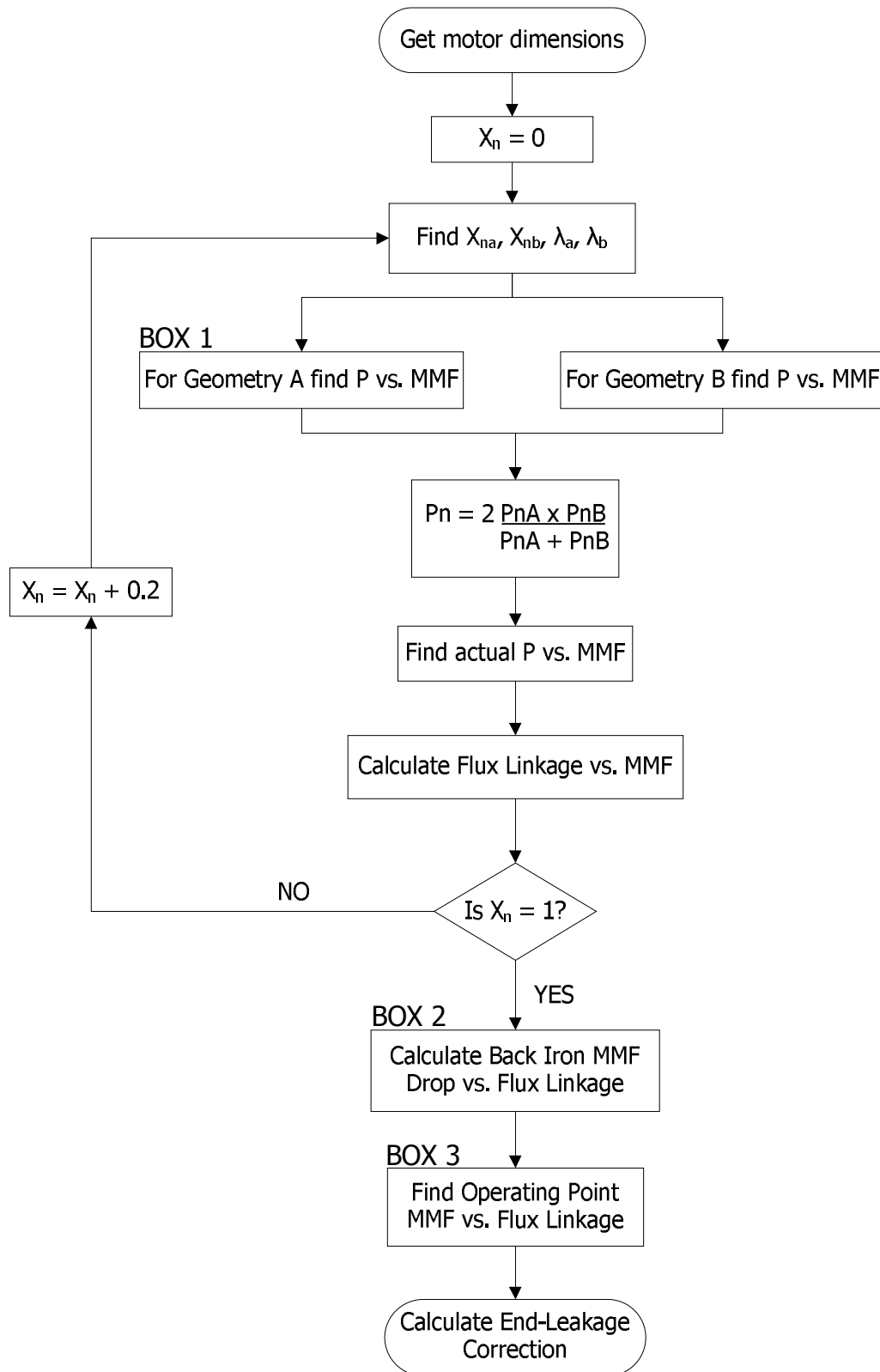


Figure 4.12 Main Flowchart of Determination of Flux Linkage Current Position

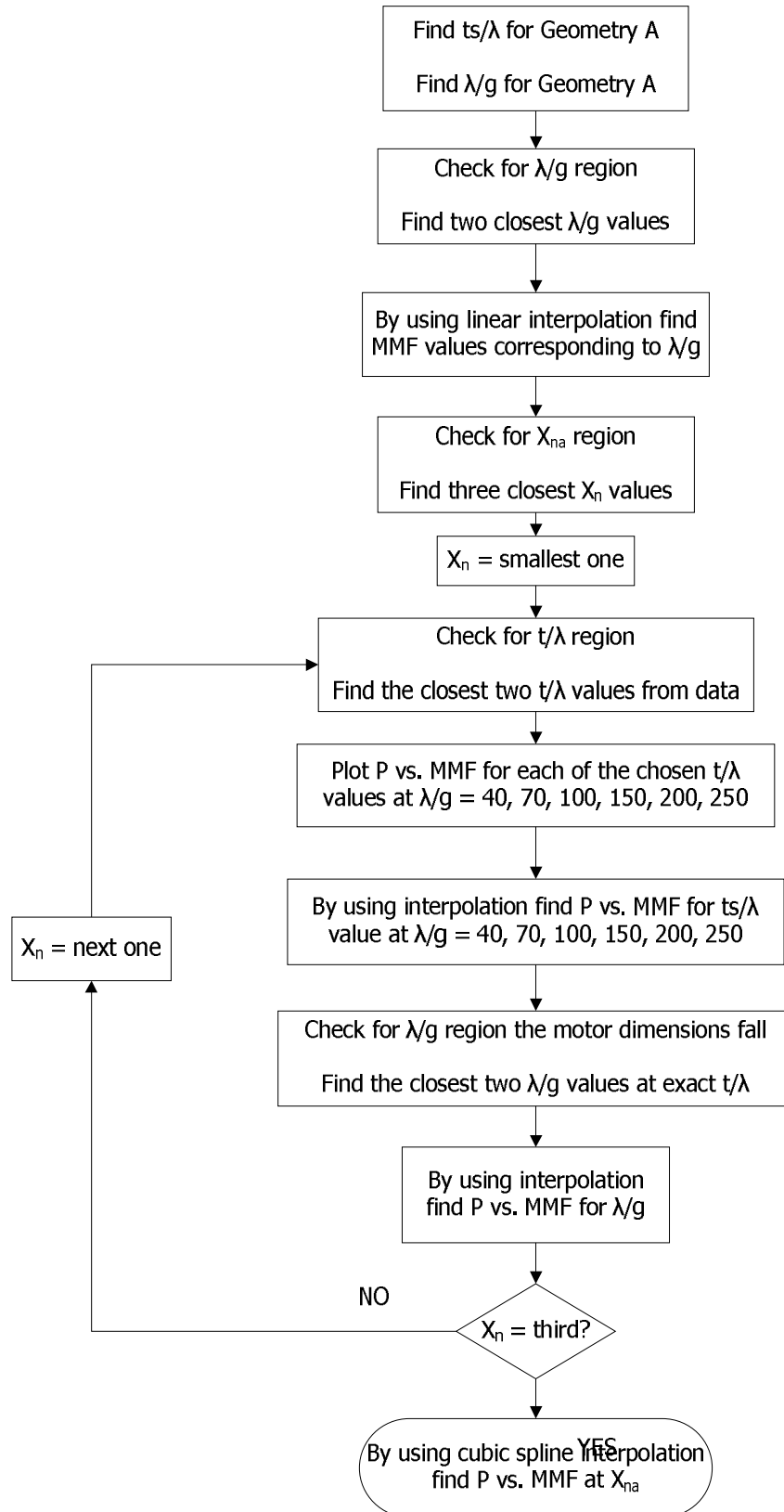


Figure 4.13 BOX 1 of Figure 4.12, Determination of Permeance vs. B Curve for Geometry A

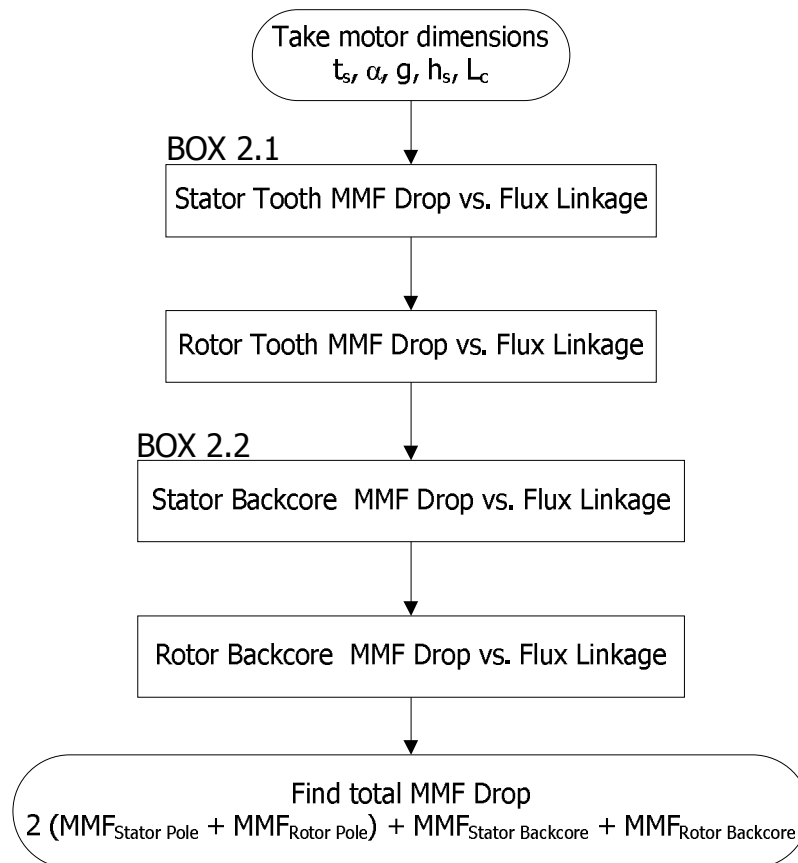


Figure 4.14 BOX 2 of Figure 4.12, Calculating Back Iron MMF Drop

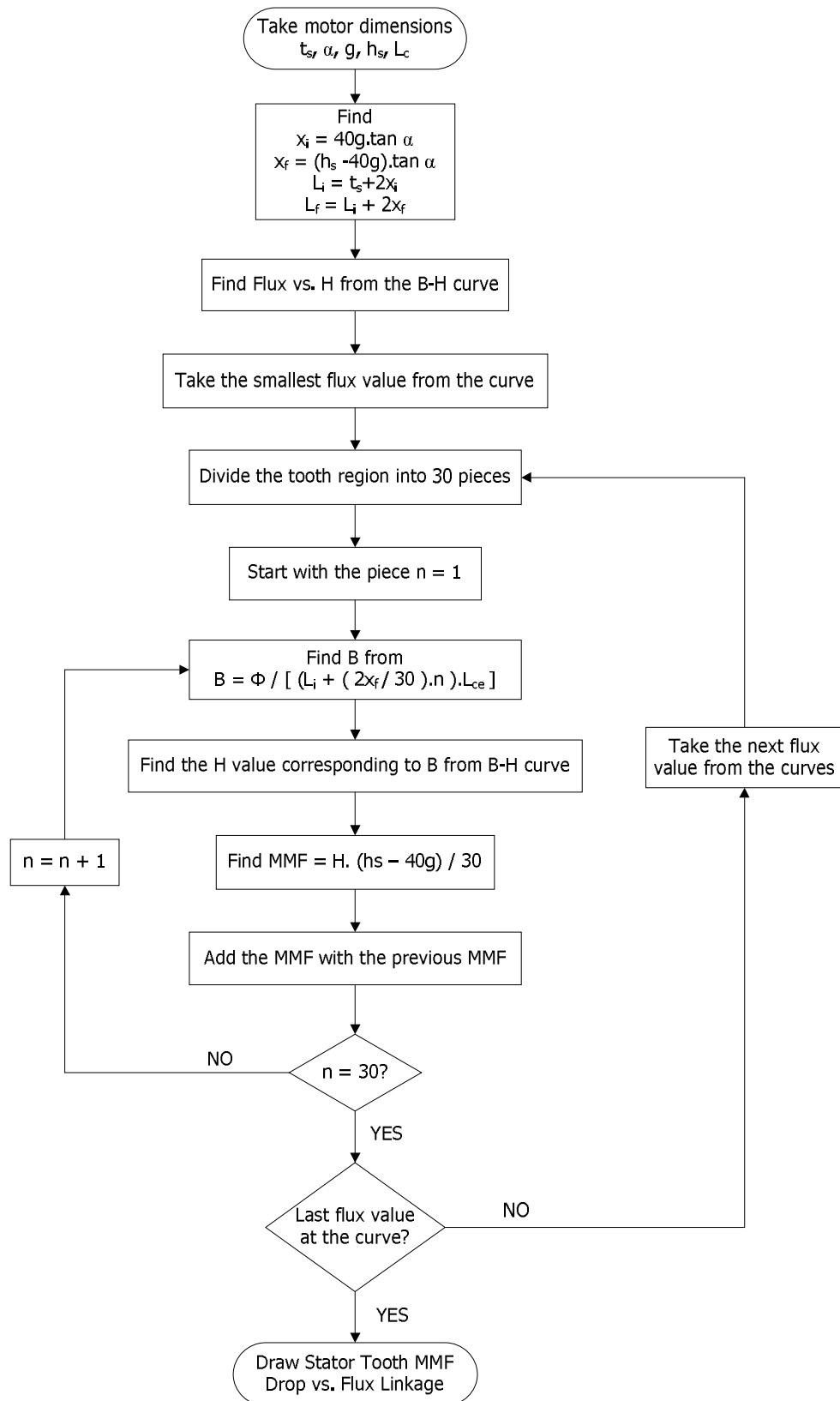


Figure 4.15 BOX 2.1 of Figure 4.14, Finding Stator Tooth MMF Drop vs. Flux Linkage Curve

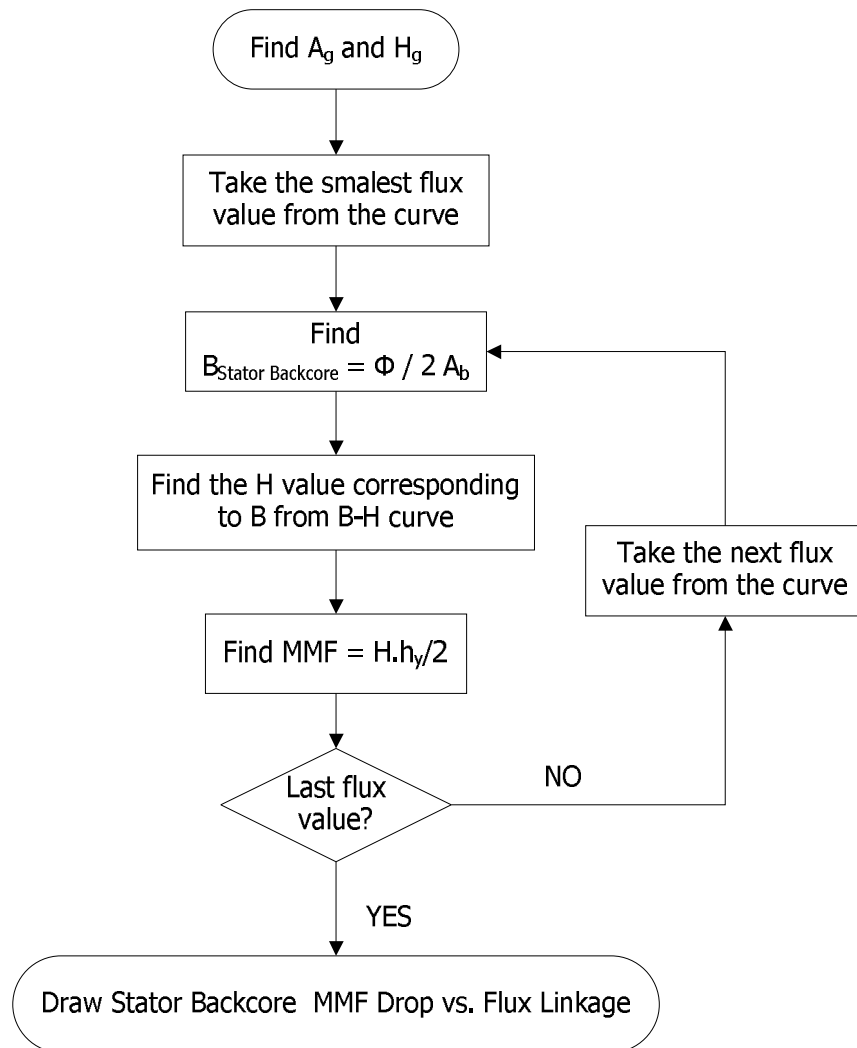


Figure 4.16 BOX 2.2 of Figure 4.14, Finding Stator Backcore MMF Drop vs. Flux Linkage

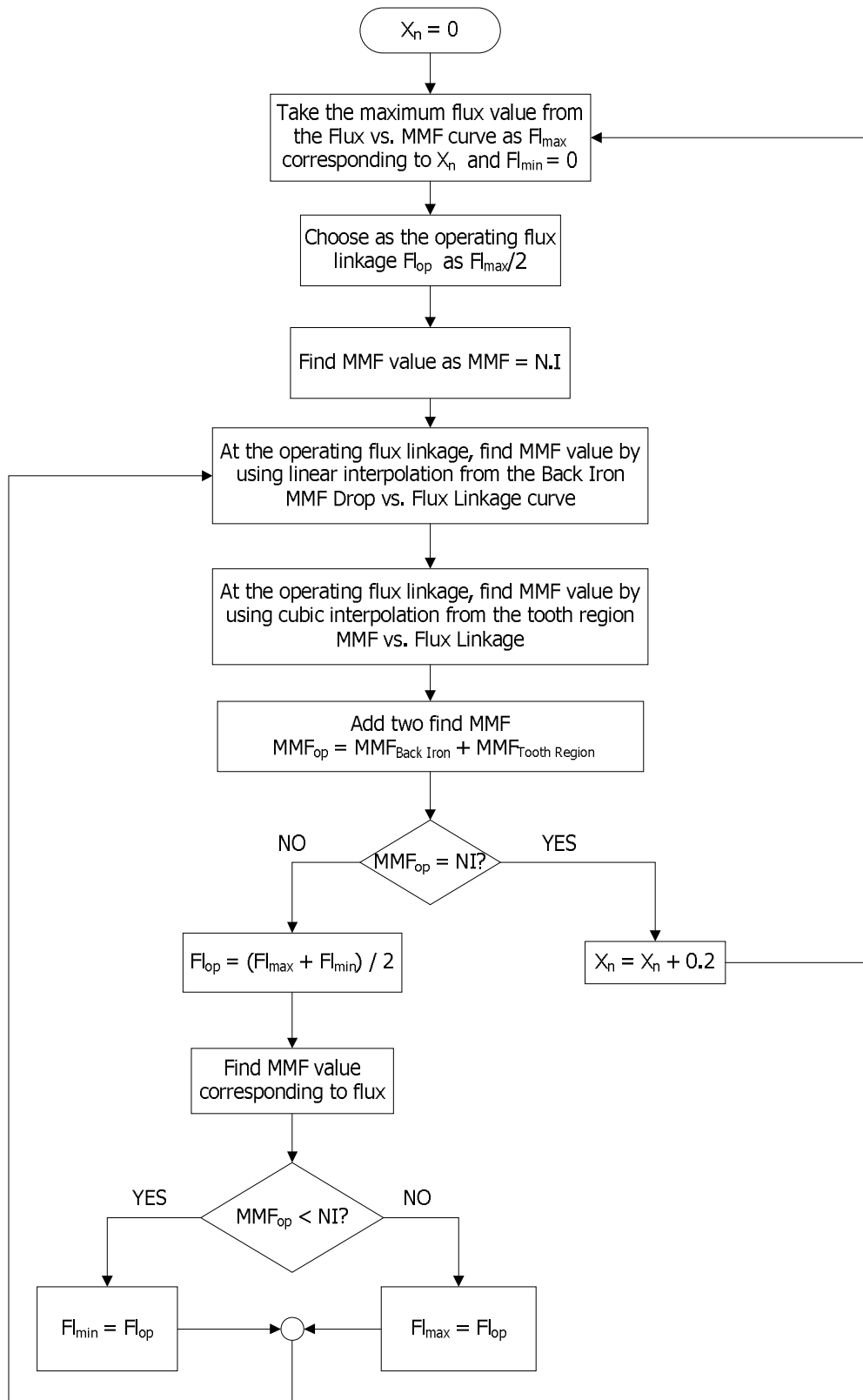


Figure 4.17 BOX 3 of Figure 4.12, Finding the Operating Point Flux Linkage vs. MMF Curve

4.5. Determination of Static Torque – Position – Current Curves

In this section, prediction of Static Torque – Position – Current Curves is introduced by the flowcharts in Figure 4.19 and Figure 4.20. For the prediction, the Normalized Force data, which is computed by Ertan is used.

As in the prediction of Flux Linkage-Position-Current curves, stator and rotor teeth are assumed asymmetrical. The force of an asymmetrical tooth pairs can be calculated using the method developed by Ertan. This approach is introduced in Section 4.4.1. As stated in Section 4.4.1 asymmetrical structures is divided into two symmetrical structures like Geometry A and Geometry B as in Figure 4.4. The determination algorithm of Force vs. MMF for Geometry A is shown in Figure 4.20. The algorithm for Geometry B is same as the algorithm for Geometry A.

After finding the Force vs. MMF curves both Geometry A and Geometry B, then by using the Equation 4.17, total Force vs. MMF curves are found.

$$Force = \frac{ForceA + ForceB}{2} \quad (4.17)$$

These steps are applied for positions $X_n=0, 0.2, 0.4, 0.6, 0.8$ and 1.0 . After applying these steps to positions, Torque vs. MMF vs. Position curve is obtained for the teeth region. However, to obtain the actual Torque vs. MMF vs. Position curve, back iron mmf drop should be considered.

To predict the actual static torque characteristics, firstly current is assumed as zero. Then total MMF is found by multiplying number of turns with current. Back iron MMF drop is found, determination of Back Iron MMF Drop is introduced in Section 4.4.2. The found Back Iron MMF drop is subtracted

from total MMF drop and actual MMF value is found. Since we have obtained the teeth Torque vs. MMF vs. Position curve, by using the cubic spline interpolation, the Torque value, which corresponds to actual MMF value, is found. At this point the end-leakage correction is applied. Then the current is increased by 1 Amper. Then the same steps are applied. In this study the maximum current is taken as 3 Amper. The same procedure is applied until 3 Amper is achieved. At the end of the procedure, the Torque vs. Current vs. position curves is obtained as in Figure 4.18.

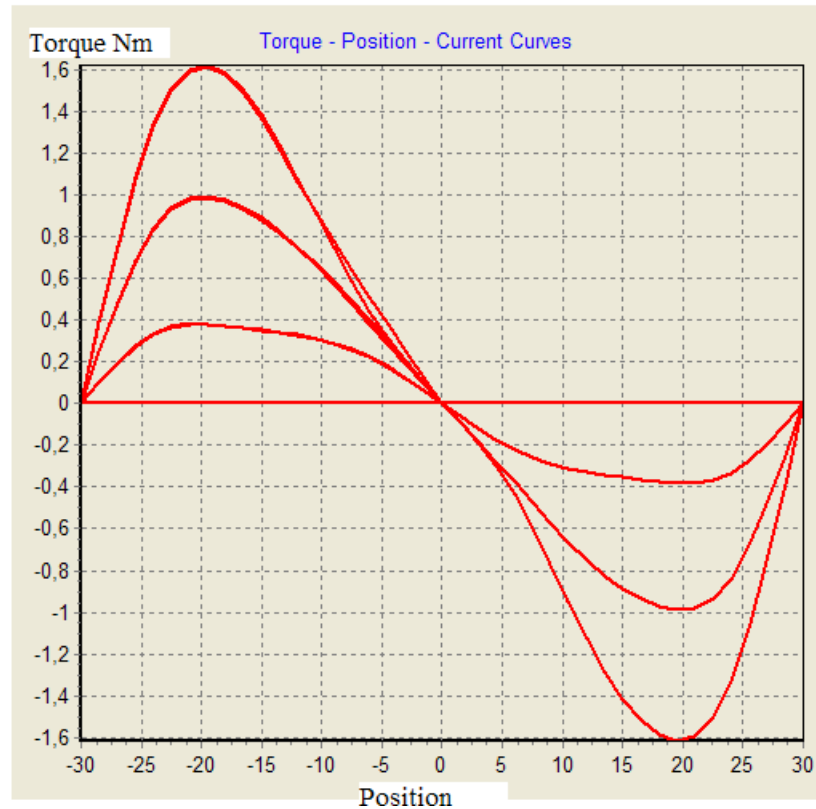


Figure 4.18 Torque - Position – Current Curves

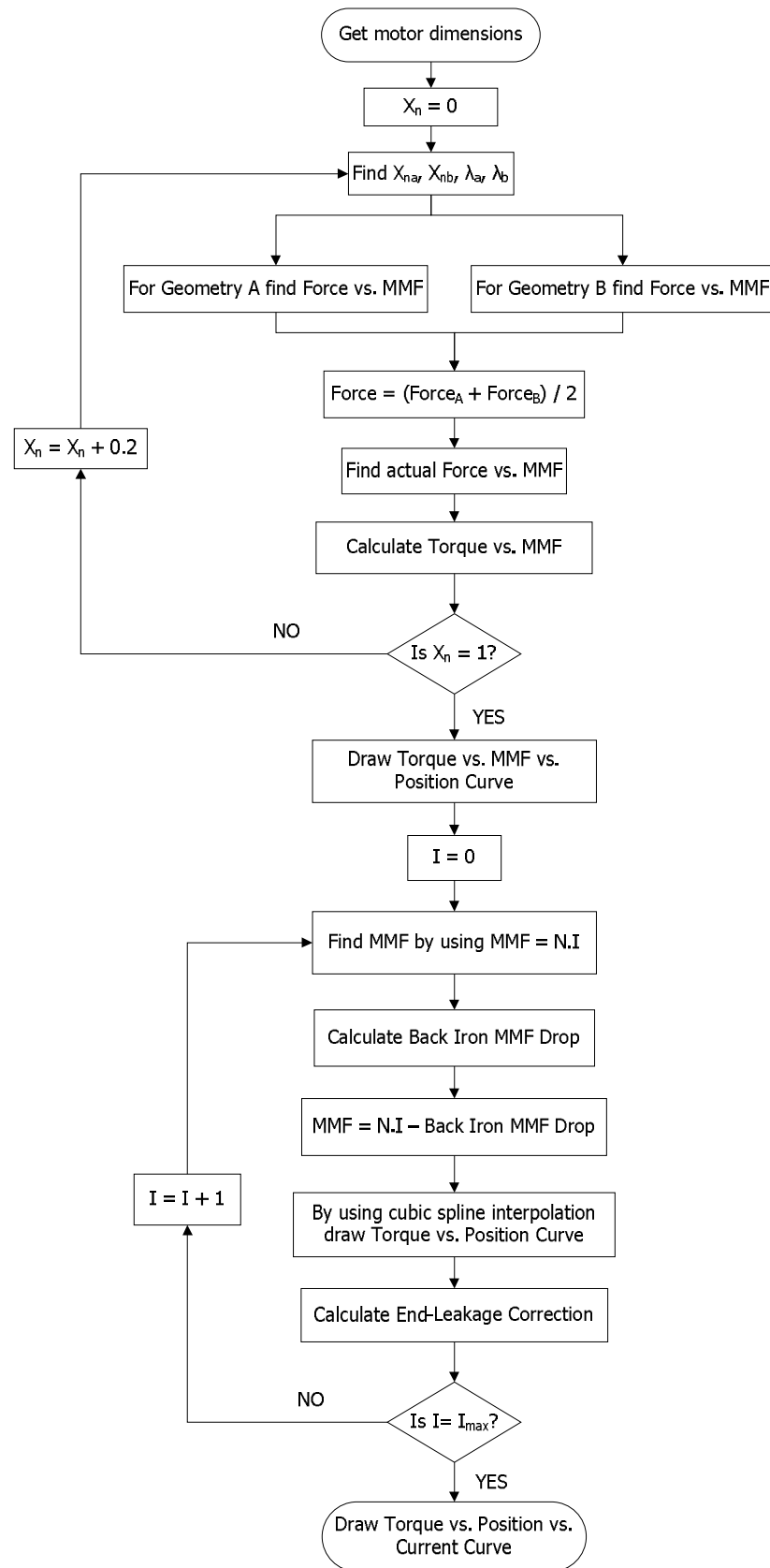


Figure 4.19 Flowchart of Torque vs. Position vs. Current

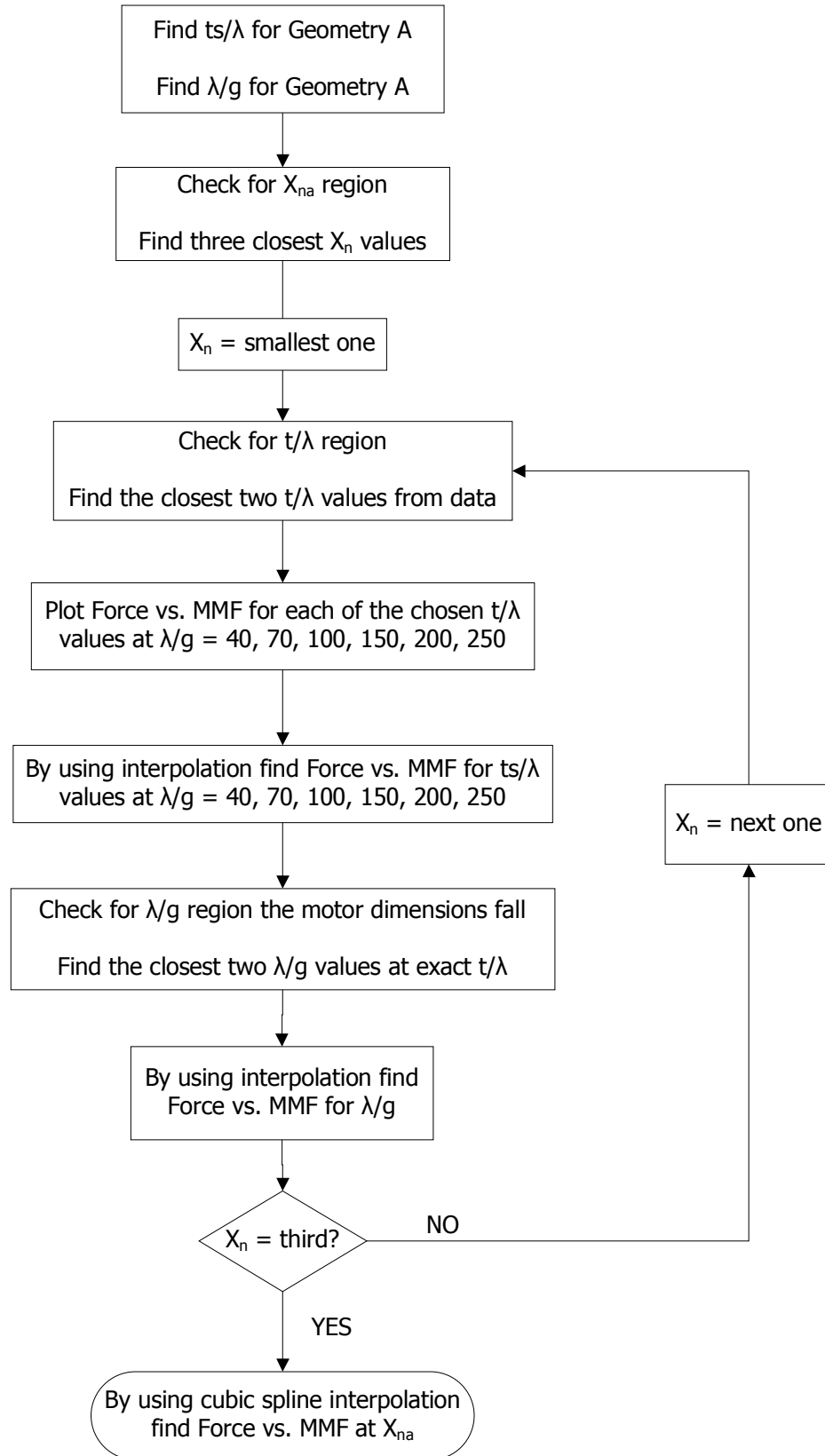


Figure 4.20 Calculation of Force vs. MMF for Geometry A

4.6. Calculation of Steady – State Performance of an SRM

To calculate the performance of the SRM, the equivalent circuit model shown in Figure 4.21 is used.

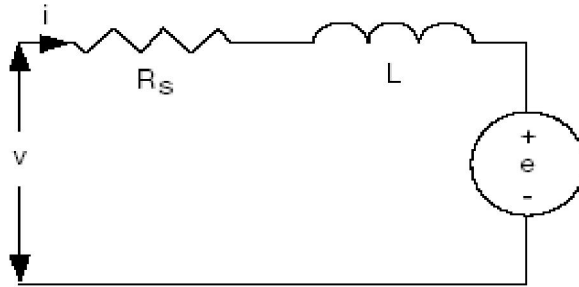


Figure 4.21 Equivalent Circuit of Single Phase of an SRM

The equation of the Voltage according to Equivalent circuit model is:

$$V = R \cdot i + \frac{d \lambda(\theta, i)}{dt} \quad (4.18)$$

To solve this equation the method which is developed by Stephenson and Corda is used [6].

With constant speed assumption, the Equation 4.18 becomes:

$$V = R \cdot i(\lambda, \theta) + \omega \cdot \frac{d\lambda}{d\theta} \quad (4.19)$$

Taking the $d\lambda / d\theta$ term in the left hand side:

$$\frac{d\lambda(\theta, i)}{d\theta} = \frac{1}{\omega} \cdot [V - R \cdot i(\lambda, \theta)] \quad (4.20)$$

To solve the Equation 4.20, the numerical differential solution method Runge Kutta 4th order is chosen to obtain Flux linkage – Current locus of the active phase.

Runge-Kutta 4th order method is a numerical technique to solve ordinary differential equation of the form

$$\frac{dy}{dx} = f(x, y) \quad y(0) = y_0 \quad (4.21)$$

Runge-Kutta 4th order method for this problem is given by the following equations

$$\begin{aligned} y_{n+1} &= y_n + (k_1 + 2k_2 + 2k_3 + k_4) \cdot h / 6 \\ x_{n+1} &= x_n + h \end{aligned} \quad (4.22)$$

where y_{n+1} is the RK4 approximation of $y(x_{n+1})$, and

$$\begin{aligned} k_1 &= f(x_n, y_n) \\ k_2 &= f(x_n + h/2, y_n + k_1 \cdot h/2) \\ k_3 &= f(x_n + h/2, y_n + k_2 \cdot h/2) \\ k_4 &= f(x_n + h, y_n + k_3 \cdot h) \end{aligned} \quad (4.23)$$

Since we try to solve the 4.20, so RK4 method becomes;

$$\begin{aligned} \lambda_{n+1} &= \lambda_n + (h/6)(k_1 + 2k_2 + 2k_3 + k_4) \\ \theta_{k+1} &= \theta_{k+h} \end{aligned} \quad (4.24)$$

where;

$$\begin{aligned}k_1 &= f(\theta_k, \lambda_k) \\k_2 &= f(\theta_k + h/2, \lambda_k + hk_1/2) \\k_3 &= f(\theta_k + h/2, \lambda_k + hk_2/2) \\k_4 &= f(\theta_k + h, \lambda_k + hk_3)\end{aligned}\tag{4.25}$$

where f is $d\lambda/d\theta$ which is calculated by substituting the Equation 4.20.

But also to solve the Equation 4.20, RK4 method needs initial condition.

Initial conditions for the solution are:

- V : Input voltage ,chosen by the user
- R : Phase resistance,chosen by the user
- $I = 0$; it is assumed that at the steady state the current of the excitation phase decreases to zero until the next excitation. However, at the high speed, current may not be decreases to zero, so to handle this case a new study should be made .
- $\lambda = 0$; flux linkage is also chosen as zero since at the steady state the current of the excitation phase decreases zero. But also, it is assumed that there is no flux linkage between phases.
- θ : Rotor Position, determined by the Advance Angle which is chosen by the user, like in Figure 4.22.

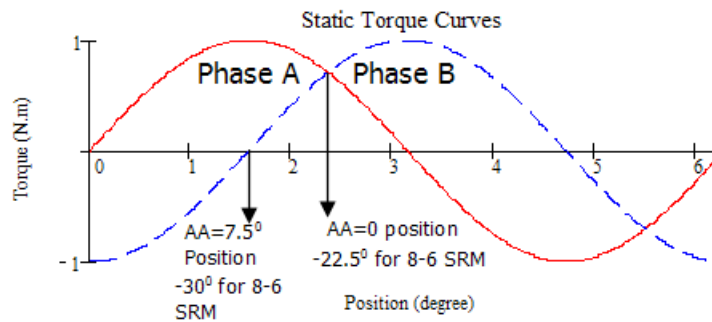


Figure 4.22 Static Torque Curves

- $h = 1/4000$; step size should be as small as possible for the accurate solution.

We can introduce the RK4 method and solution algorithm step by step by using Figure 4.23. Then;

- Initial values are λ_0 and θ_0 .
- k_1 is found by using the Equation 4.25.
- To find k_1 , in the equation λ_0 and θ_0 are used. But if we look at the equation it is seen that i_0 is dependent to λ_0 and θ_0 . To find i_0 , we can use the flux linkage-current-position curve. As seen in the Figure 4.23, i_0 is found from the curve.
- k_2 is found by using the Equation 4.25
- To find k_2 ,
 - position is taken as $\theta_0 + h/2$
 - flux linkage is taken as $\lambda_0 + k_1 \cdot h/2$
 - to find $i(\lambda, \theta)$, the new value of position should be taken into account. But the position may be not the exact value on the curve.

- To find the exact flux linkage-current curve, we find the three nearest position value to $\theta_0 + h/2$, after then by applying the cubic spline interpolation, the flux linkage-current curve at the position $\theta_0 + h/2$, is obtained.
- At the obtained curve, the current value which corresponds to $\lambda_0 + k_1 \cdot h/2$ is found.
- By applying the Equation 4.20, k_2 is found.
- To find k_3 ,
 - position is taken as $\theta_0 + h/2$, same as the k_2
 - flux linkage is taken as $\lambda_0 + k_2 \cdot h/2$
 - to find $i(\lambda, \theta)$, the new value of position should be taken into account. But the position may be not the exact value on the curve.
 - To find the exact flux linkage-current curve, we find the three nearest position value to $\theta_0 + h/2$, after then by applying the cubic spline interpolation, the flux linkage-current curve at the position $\theta_0 + h/2$, is obtained.
 - At the obtained curve, the current value which corresponds to $\lambda_0 + k_2 \cdot h/2$ is found.
 - By applying the Equation 4.20, k_3 is found.
- To find k_4 ,
 - position is taken as $\theta_0 + h$,
 - flux linkage is taken as $\lambda_0 + k_3 \cdot h$
 - to find $i(\lambda, \theta)$, the new value of position should be taken into account. But the position may be not the exact value on the curve.
 - To find the exact flux linkage-current curve, we find the three nearest position value to $\theta_0 + h$, after then by applying the cubic spline interpolation, the flux linkage-current curve at the position $\theta_0 + h$, is obtained.

- At the obtained curve, the current value which corresponds to $\lambda_0 + k_3 \cdot h$ is found.
- By applying the Equation 4.20, k_4 is found.
- Since the k_1, k_2, k_3 and k_4 values are found, λ_1 can be found easily by using the Equation 4.24
- Position value, θ_1 , is found by adding the step size.

$$\theta_1 = \theta_0 + 1/4000$$
- Since the θ_1 and λ_1 is found, the value of i_1 can be found by using the flux-linkage vs. current curve at the position θ_1 . Again, if the curve has not the exact position value, then by using the cubic spline interpolation, the flux-linkage vs. current curve is obtained.

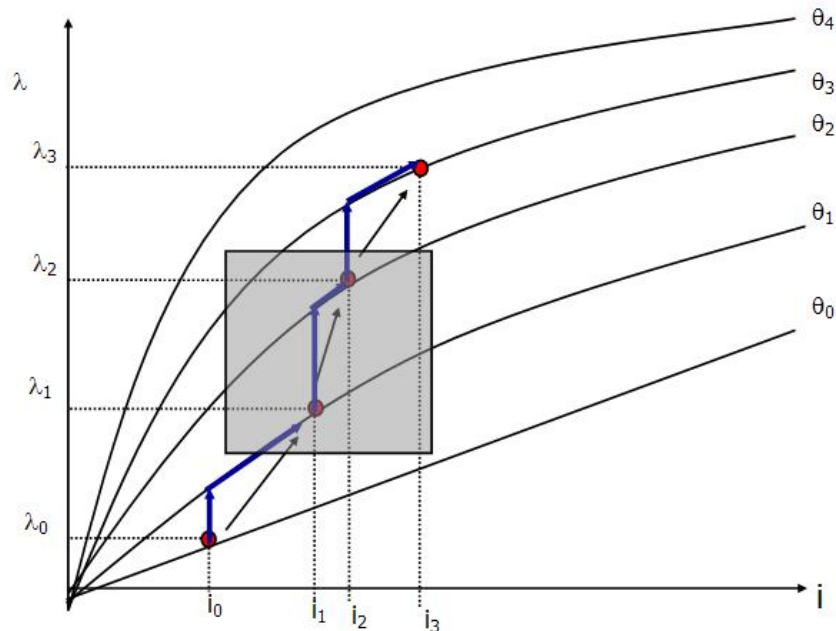


Figure 4.23 Applying Runge Kutta 4th Order Method

Flowchart of the steady – state solution algorithm is given in Figure 4.24. At the end of the process, we obtain:

- Current – Position ($I - \theta$) waveform,
- Flux linkage – Position ($\lambda - \theta$) waveform,
- Instantaneous Torque – Position ($T - \theta$) waveform,
- Flux linkage – Current ($\lambda - I$) locus.

To obtain these forms, in the software 4 data arrays are defined. These arrays are;

- Iplot : The values of current is recorded
- FipLOT: The values of flux linkage is recorded
- Xplot: The values of position is recorded
- Tplot: The values of instantaneous torque is recorded, at every step.

Initially, current and flux linkage is set to zero and position is set to on position. Then; Runge Kutta algorithm is applied for one step. At the end of the step, firstly flux linkage is found. Then new position value is found by incrementing the position with the step size like in Figure 4.23. After getting the flux linkage and position values, determining the new current value is a simple step. By using cubic spline interpolation , current value is found. Since the position and current values are found, torque value at that step is found by using interpolation at the static torque curves. Although torque can be found by coenergy method, in this study it is found by using static torque curves since we have determined the static torque curve. At the end of the Runge Kutta step, Flux linkage, position and current values are found and these values are recorded to their data arrays. As seen in the flowchart of the solution algorithm, Figure 4.24, these steps are applied until the OFF state. Then, voltage becomes negative and again Runge Kutta algorithm is

applied until the current becomes zero. At the end of the solution algorithm, all the data arrays are obtained.

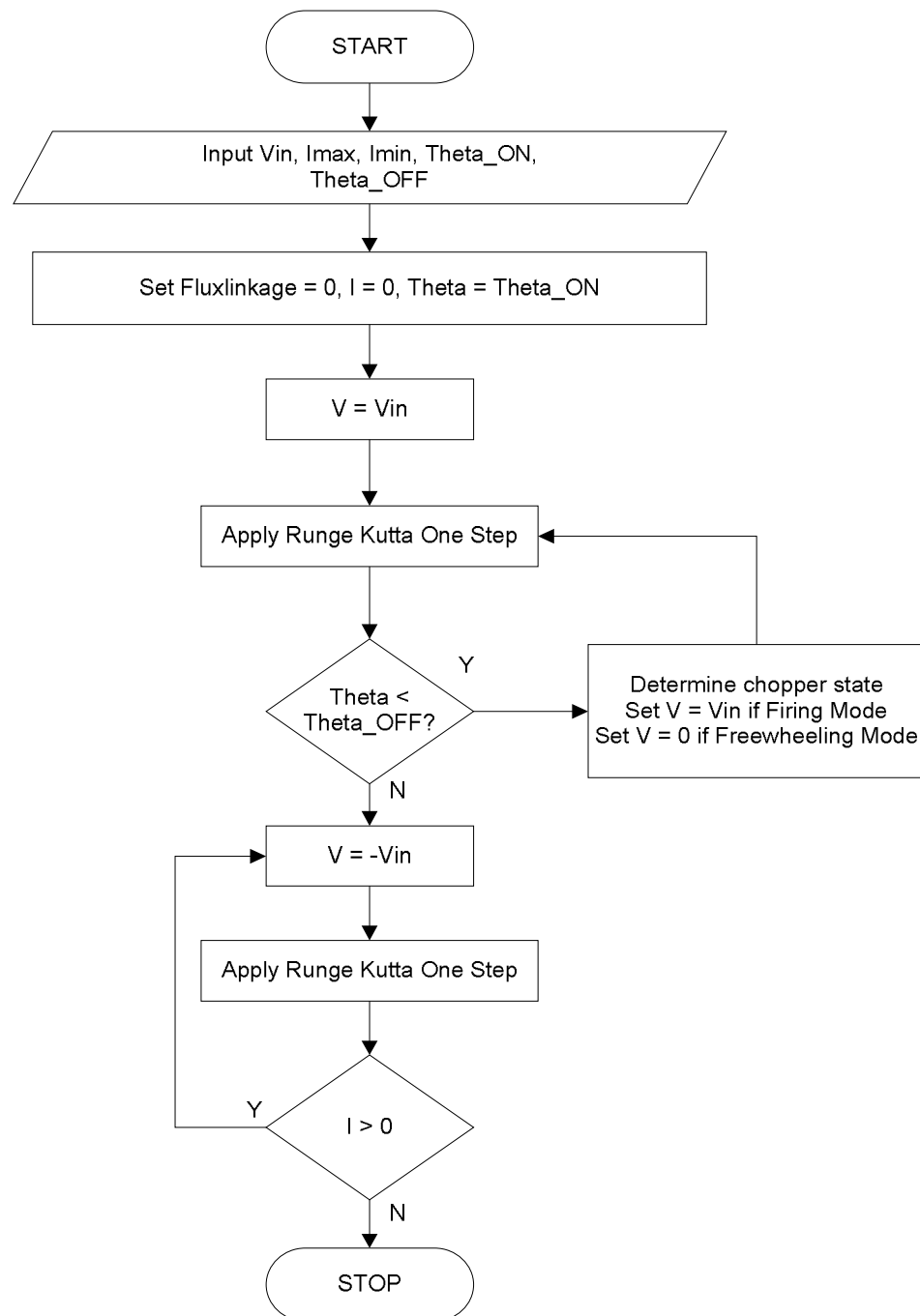


Figure 4.24 Flowchart of the Solution Algorithm

4.6.1. Calculation of Instantaneous Torque Waveform

To calculate the steady-state average torque and torque ripple, Instantaneous torque waveform is needed. This waveform can be found by using Static Torque curves. In this study, we have found this waveform by using the Static Torque curves. As discussed above, solution algorithm (Figure 4.24) is applied. By using Runge Kutta algorithm, current and position values are found. Then, by applying the interpolation method, torque value is found.

At the end of the solution algorithm, instantaneous torque array is found. But this array is for one phase. Since the obtained waveforms are periodic with a shift of a phase angle for successive phases, to find the resultant waveform, the waveforms for all phases are added and the sum of waveforms of all phases is seen in Figure 4.25.

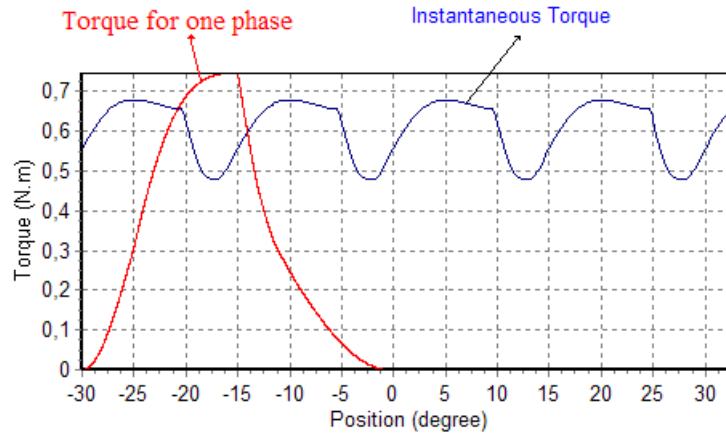


Figure 4.25 Resultant Instantaneous Torque – Position Waveform of an 8-6 SRM Obtained at 2500 rpm with 7.5° Advance Angle

Another approach to find the Instantaneous torque, is finding the co-energy change from one position to other position.

$$T(\theta) = \frac{\Delta W_c}{\Delta \theta} \quad (4.26)$$

where,

$$\Delta \theta = \theta_{k+1} - \theta_k \quad (4.27)$$

and,

$$\Delta W_c = (\lambda_{k+1} - \lambda_k) \cdot (i_{k+1} - i_k) \quad (4.28)$$

ΔW_c is the change in co-energy while rotor moves from θ_k to θ_{k+1} .

The change in co-energy is found by using the Flux linkage-Current-Position curve like in figures below.

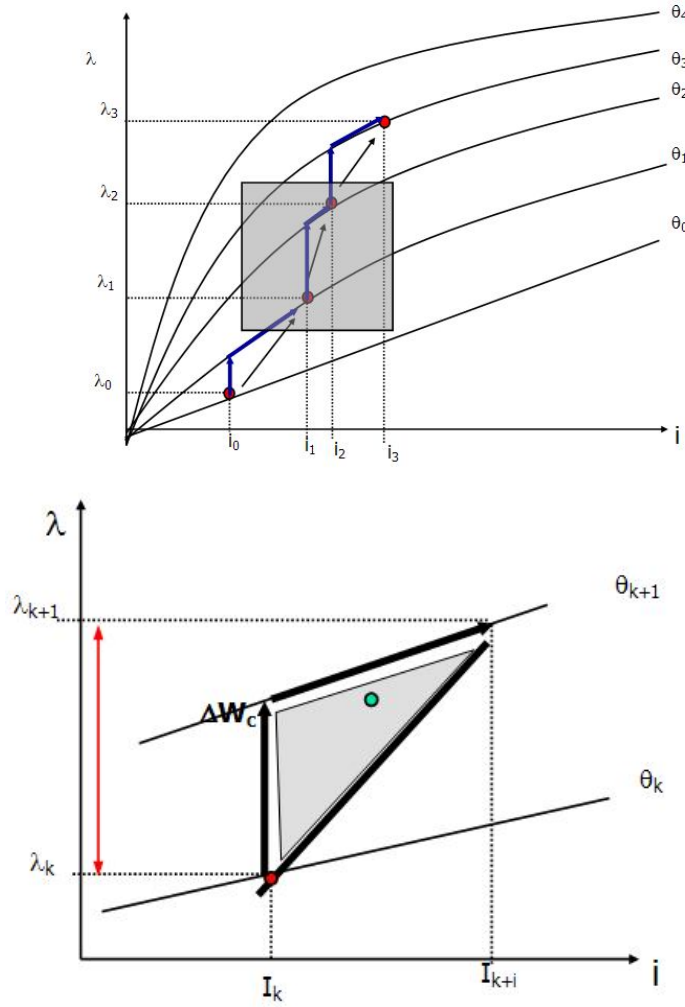


Figure 4.26 Determination of Instantaneous Co-Energy, and Hence the Torque [13]

4.6.2. Calculation of Steady – State Average Torque and Torque – Ripple

For a given geometry, Instantaneous Torque – Position ($T - \theta$) waveform seen in Figure 4.27 is calculated and is stored as a data array in the software as discussed above. Calculation of the “Average Torque” is done by taking the average of this data array over rotor tooth pitch.

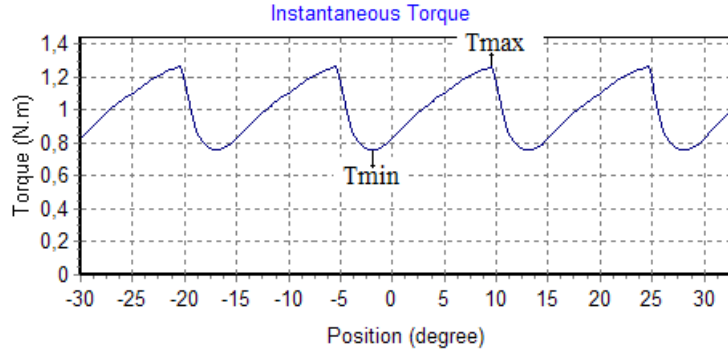


Figure 4.27 Determination of Torque Ripple

To calculate the torque ripple the Equation 4.29 and Figure 4.27 is used:

$$T_{ripple} = \frac{T_{max} - T_{min}}{T_{max}} \times 100 \quad (4.29)$$

To find the T_{max} , firstly it is set to -100000, then the instantaneous torque array is controlled from start to end. If the value of the Instantaneous torque is bigger than T_{max} , then new value of T_{max} becomes the Instantaneous torque value at that point.

To find the T_{min} , firstly it is set to 100000, then the instantaneous torque array is controlled from start to end. If the value of the Instantaneous torque is smaller than T_{min} , then new value of T_{min} becomes the Instantaneous torque value at that point.

4.7. The Accuracy of the Analysis Routines

In this study one of SRM motors of METU Electrical and Electronics Department, called SR2 is chosen as the test motor. This motor is designed for a washing machine application and is required to produce 2.5 Nm at 520 rpm

and 0.25Nm at 12500 rpm. Note that the washing machine has a 1/10 reduction ratio belt. The specifications of SR2 are listed below:

Rated power	: 350 Watt	136 Watt
Rated speed	: 12500 rpm	520 rpm
Torque	: 0.25 Nm	2.5 Nm
Operating Voltage	: 300 Volt DC	300 Volt DC
Rated Current	: 3 Ampere/pole	3 Ampere/pole
Number of phases	: 4	4
Number of rotor teeth	: 6	6

Table 4.1 Measured Physical Dimensions of SR2

d_i	Rotor outer diameter (mm)	38.6
D_o	Stator outer length (mm)	110.4
D_i	Stator core inner diameter (mm)	99.99
t_s	Stator pole tip width (mm)	8.35
α	Tapering angle (degree)	2.2
t_r	Rotor tooth width (mm)	8.4
g	Airgap length (mm)	0.325
h_r	Rotor tooth depth (mm)	7.2
h_s	Stator pole depth (mm)	30.37
d_a	Shaft diameter (mm)	16.5
L_c	Core length (mm)	41

The measured dimensions of the test motor is given in Table 4.1 and to analyze this motor firstly, flux-linkage characteristics are predicted according to given dimensions. The predictions are based on normalized performance – position-current curves and is calculated using the approach described in Section 4.4. In the next sections, the predicted and measured flux-linkage

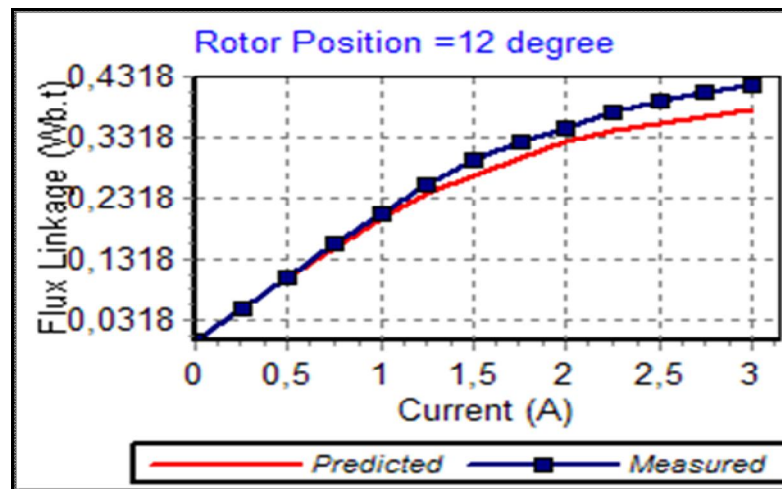
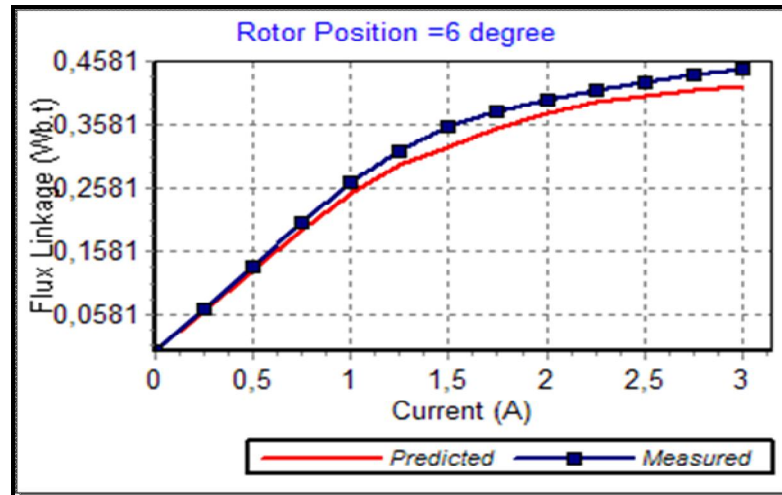
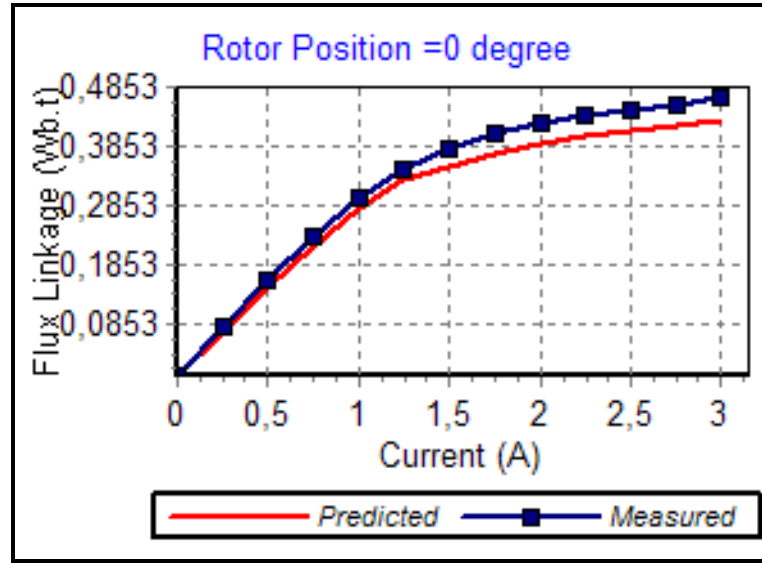
characteristics are tabulated and compared; the accuracy of the prediction method is investigated.

4.7.1. The accuracy of Prediction of Flux Linkage Characteristics

The “Flux linkage-Position-Current” data are obtained by M.Özgür KIZILKAYA and E.Bizkevelci [8]. In this study, by using the analysis part of SRCAD the flux linkage characteristics of SR2 is predicted with and without End-Leakage Flux correction. In the predictions the normalized permeance data produced by Ertan [1] is used. At first no correction is made for end leakage effect. The measured and predicted data are compared in Figure 4.28 and tabulated at Table 4.2, below.

4.7.1.1. Predicted Flux Linkage Curves Without End Leakage Correction

In this section, Figure 4.28 presents the predicted and measured Flux-linkage - current - position curves for 0° to 30° in 6° steps. Note that these positions in normalized form correspond to 0, 0.2, 0.4, 0.6, 0.8 and 1.0. The predicted curves are also shown in the same figures.



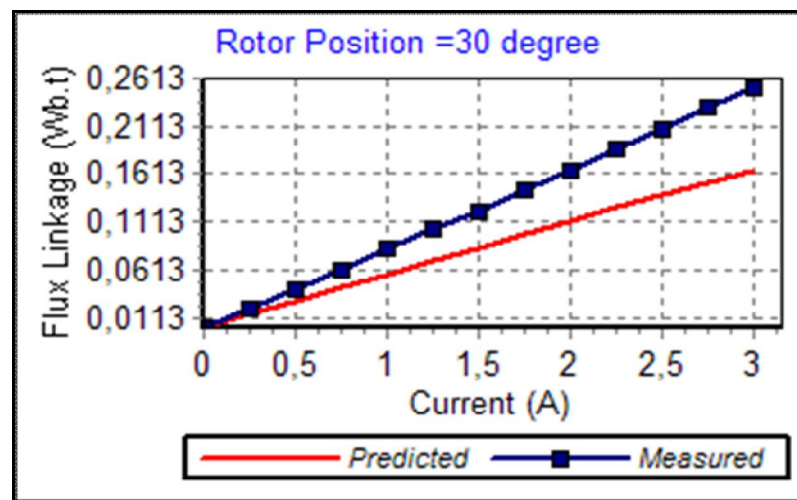
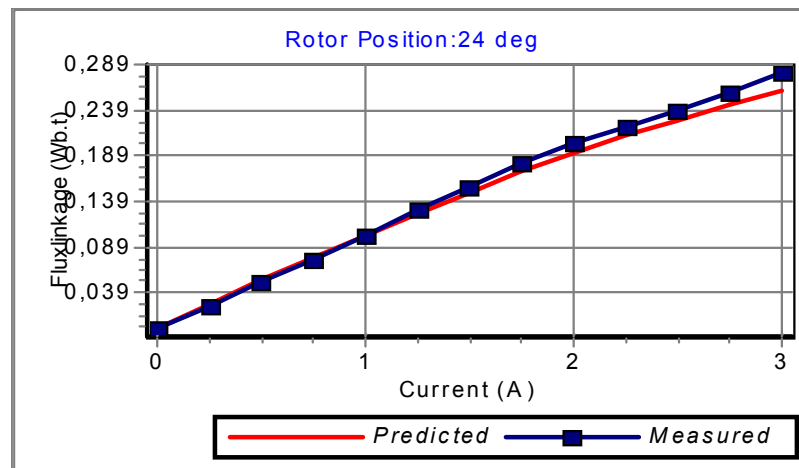
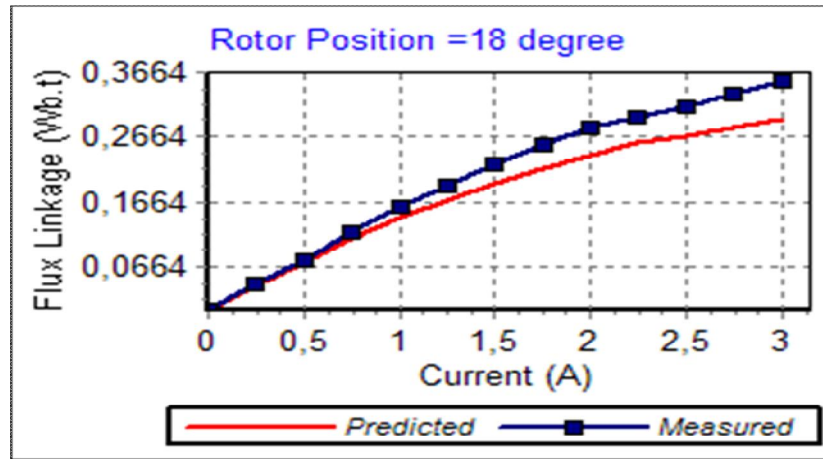


Figure 4.28 Measured Curves and Predicted Curves without End Leakage Correction

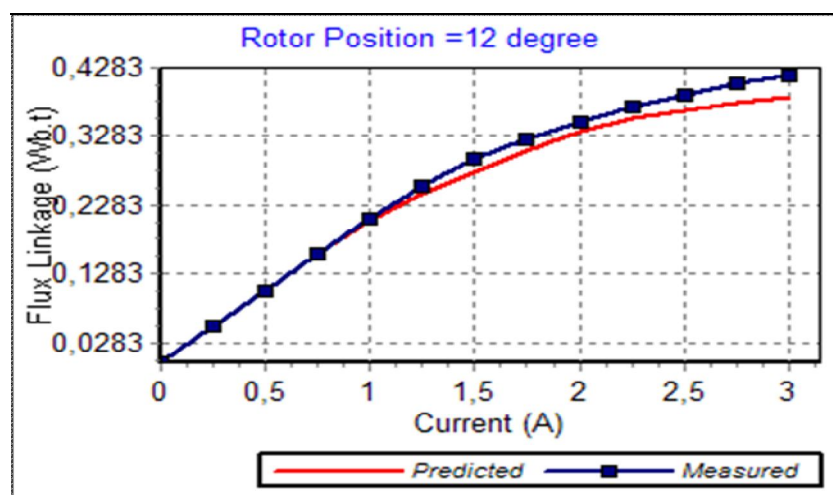
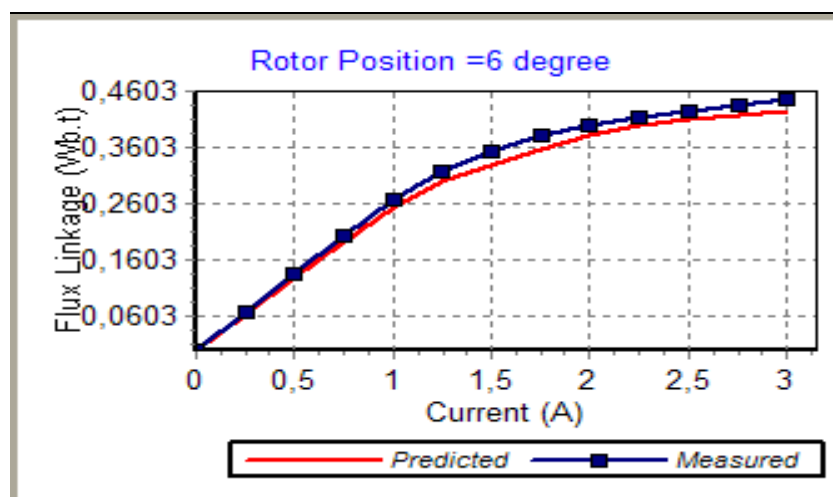
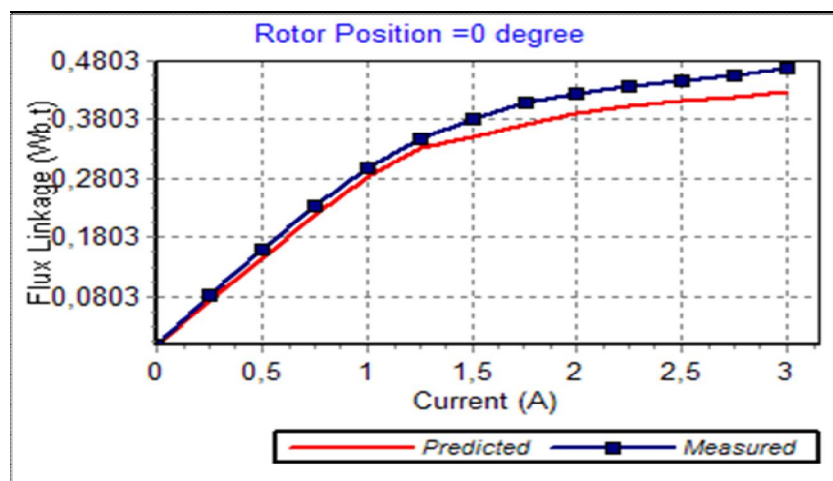
Table 4.2 Numerical Values and % Errors Between Measured Curves and Predicted Curves without End Leakage Correction

Flux linkage (Wb.t)									
	0°			6°			12°		
<i>I</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>
0	0	0	0	0	0	0	0	0	0
0.25	0.083	0.0730	12.03	0.067	0.0638	4.71	0.053	0.0510	3.71
0.5	0.160	0.1460	8.73	0.134	0.1277	4.71	0.105	0.1021	2.80
0.75	0.234	0.2185	6.64	0.203	0.1904	6.19	0.157	0.1522	3.07
1	0.299	0.2838	5.09	0.267	0.2479	7.16	0.208	0.1988	4.41
1.25	0.347	0.3313	4.54	0.316	0.2936	7.09	0.255	0.2383	6.56
1.5	0.383	0.3519	8.13	0.354	0.3229	8.79	0.296	0.2681	9.42
1.75	0.408	0.3725	8.70	0.380	0.3522	7.31	0.325	0.2980	8.32
2	0.425	0.3907	8.08	0.398	0.3776	5.14	0.348	0.3242	6.85
2.25	0.437	0.4042	7.51	0.413	0.3953	4.29	0.371	0.3433	7.46
2.5	0.446	0.4119	7.64	0.425	0.4036	5.04	0.390	0.3538	9.28
2.75	0.457	0.4196	8.18	0.437	0.4119	5.75	0.406	0.3642	10.29
3	0.468	0.4273	8.69	0.446	0.4201	5.80	0.417	0.3747	10.15
	18°			24°			30°		
<i>I</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>
0	0	0	0	0	0	0	0	0	0
0.25	0.038	0.0362	4.63	0.025	0.0225	10.06	0.020	0.0141	29.49
0.5	0.077	0.0725	5.87	0.051	0.0450	11.83	0.040	0.0282	29.49
0.75	0.118	0.1077	8.70	0.076	0.0670	11.88	0.061	0.0421	30.93
1	0.157	0.1405	10.49	0.102	0.0882	13.55	0.083	0.0559	32.59
1.25	0.192	0.1692	11.86	0.129	0.1085	15.92	0.103	0.0699	32.16
1.5	0.224	0.1933	13.72	0.155	0.1282	17.32	0.122	0.0843	30.91
1.75	0.253	0.2173	14.10	0.180	0.1478	17.86	0.144	0.0987	31.46
2	0.279	0.2391	14.28	0.202	0.1665	17.59	0.165	0.1128	31.62
2.25	0.298	0.2567	13.87	0.221	0.1830	17.18	0.187	0.1264	32.40
2.5	0.315	0.2689	14.65	0.239	0.1970	17.56	0.209	0.1393	33.33
2.75	0.33357	0.2810	15.75	0.25897	0.2110	18.51	0.23001	0.1522	33.81
3	0.3532	0.2932	16.98	0.28018	0.2250	19.68	0.25118	0.1652	34.25

In Figure 4.28 it is seen that, prediction of flux linkage curves are close to measured curves near IN position of the rotor; however, the percentage error increases while the rotor position is near OUT position. This is because the end leakage contribution is neglected.

4.7.1.2. Predicted Flux Linkage Curves With End leakage Correction

In this section, the flux linkage current position curves are corrected by adding the end leakage effect. In accounting for the end leakage, the approach presented in [13] is used. This approach is briefly described in Section 4.4.6.



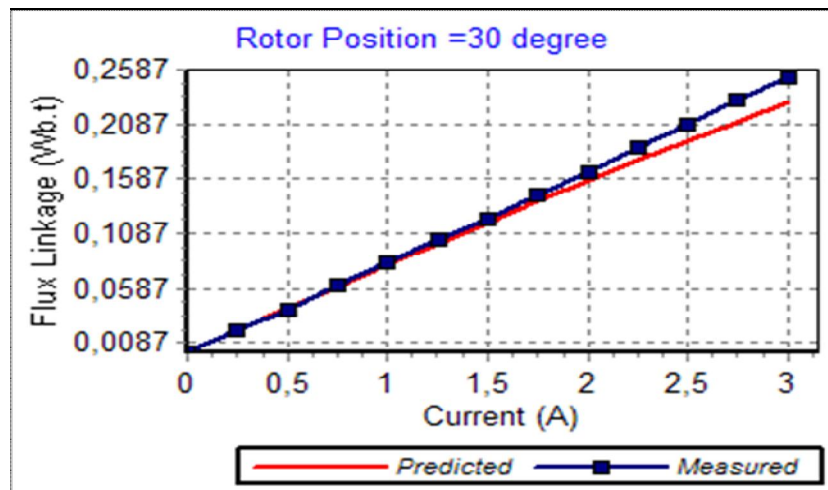
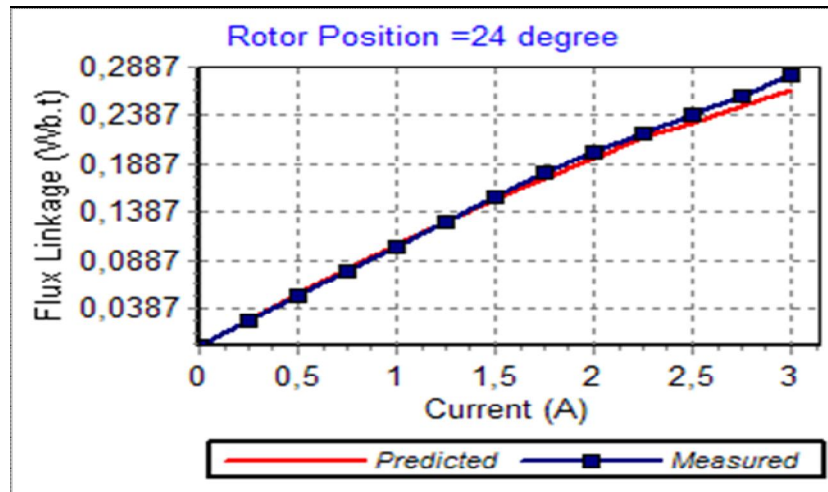
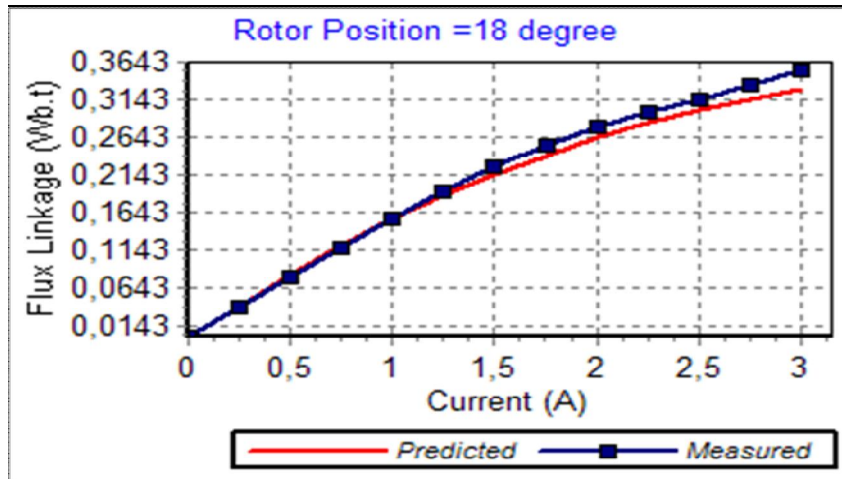


Figure 4.29 Measured Curves and Predicted Curves with End Leakage Correction

Table 4.3 Numerical Values and % Errors Between Measured Curves and Predicted Curves with End Leakage Correction

Flux linkage (Wb.t)									
	0°			6°			12°		
<i>I</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>
0	0	0	0	0	0	0	0	0	0
0.25	0.083	0.0731	11.91	0.067	0.0649	3.12	0.053	0.0528	0.45
0.5	0.160	0.1462	8.60	0.134	0.1297	3.18	0.105	0.1054	0.39
0.75	0.234	0.2187	6.53	0.203	0.1933	4.74	0.157	0.1570	0.01
1	0.299	0.2841	4.98	0.267	0.2516	5.77	0.208	0.2049	1.47
1.25	0.347	0.3316	4.43	0.316	0.2980	5.69	0.255	0.2456	3.68
1.5	0.383	0.3523	8.02	0.354	0.3281	7.33	0.296	0.2767	6.53
1.75	0.408	0.3729	8.60	0.380	0.3576	5.90	0.325	0.3076	5.34
2	0.425	0.3912	7.97	0.398	0.3828	3.82	0.348	0.3349	3.77
2.25	0.437	0.4046	7.41	0.413	0.4005	3.03	0.371	0.3546	4.43
2.5	0.446	0.4123	7.55	0.425	0.4089	3.79	0.390	0.3656	6.25
2.75	0.457	0.4200	8.08	0.437	0.4172	4.53	0.406	0.3765	7.26
3	0.468	0.4278	8.60	0.446	0.4254	4.62	0.417	0.3873	7.12
	18°			24°			30°		
<i>I</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>	<i>Meas.</i>	<i>Pred.</i>	<i>Err%</i>
0	0	0	0	0	0	0	0	0	0
0.25	0.038	0.0401	5.49	0.025	0.0267	6.79	0.020	0.0203	1.37
0.5	0.077	0.0800	3.86	0.051	0.0533	4.44	0.040	0.0405	1.18
0.75	0.118	0.1186	0.50	0.076	0.0791	4.13	0.061	0.0603	1.18
1	0.157	0.1545	1.57	0.102	0.1040	2.00	0.083	0.0798	3.90
1.25	0.192	0.1862	3.01	0.129	0.1278	0.90	0.103	0.0992	3.73
1.5	0.224	0.2132	4.84	0.155	0.1509	2.66	0.122	0.1188	2.60
1.75	0.253	0.2399	5.16	0.180	0.1738	3.46	0.144	0.1383	3.96
2	0.279	0.2644	5.24	0.202	0.1955	3.24	0.165	0.1573	4.66
2.25	0.298	0.2845	4.53	0.221	0.2150	2.70	0.187	0.1756	6.07
2.5	0.315	0.2994	4.97	0.239	0.2320	2.92	0.209	0.1932	7.57
2.75	0.33357	0.3141	5.83	0.25897	0.2489	3.89	0.23001	0.2106	8.45
3	0.3532	0.3288	6.91	0.28018	0.2657	5.18	0.25118	0.2278	9.30

From the results presented in Figure 4.29 and Table 4.3, it is clear that the end leakage correction greatly increases the accuracy of the flux linkage prediction of SR motors. According to Table 4.3, when the rotor is at 30 degree and the current is 3A, the percentage of flux linkage error is 23.170 % without end leakage correction, but if the end-leakage correction is taken into account then this error percentage reduces to 1.244 %.

4.7.2. The accuracy of Torque – Speed Characteristics

The “Torque – Speed” data of the test motor are measured by M.Özgür KIZILKAYA [8]. In this study by using the analysis part of SRCAD the Torque – Speed characteristics of SR2 for different advance angles are calculated by using the predicted curves with and without End – Leakage Flux correction. Thus, the measured and the calculated data are compared and tabulated below. For the test motor, the operating conditions at the measurement are as below, also for the simulation same conditions are used.

$$V_{in} = 300V$$

$$I_{max} = 3.15A$$

$$I_{min} = 2.85A$$

$$\text{Excitation Period} = 0.5 \text{ pu}$$

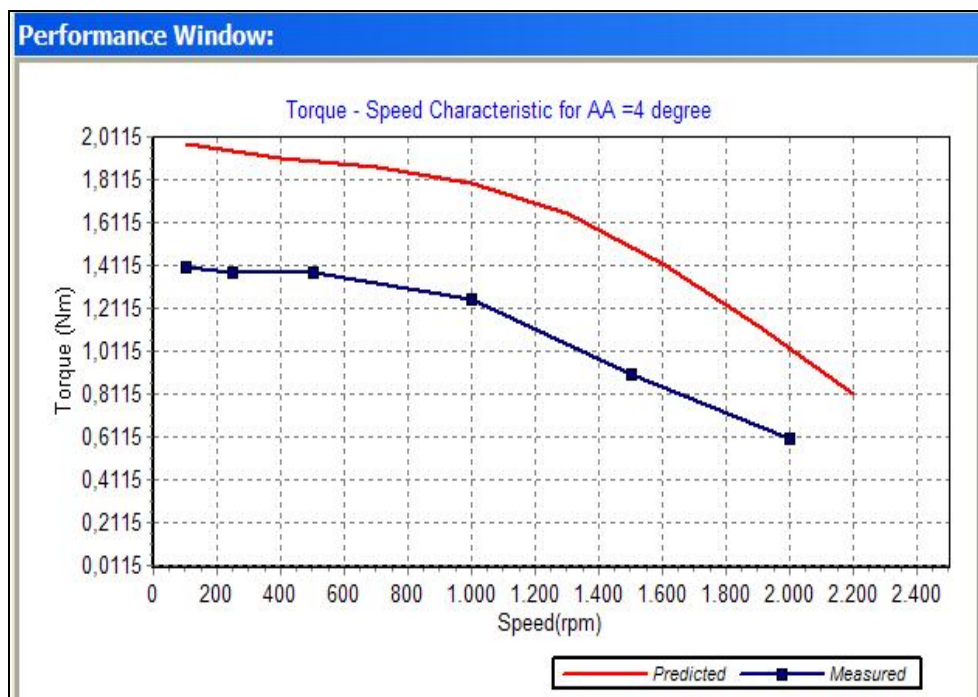
$$\text{Phase Resistance} = 5.2 \text{ Ohm}$$

4.7.2.1. Torque – Speed Curves Without End Leakage Correction

The figures below show the results of SRCAD performance analysis in which Torque Speed characteristics are calculated without end-leakage correction.



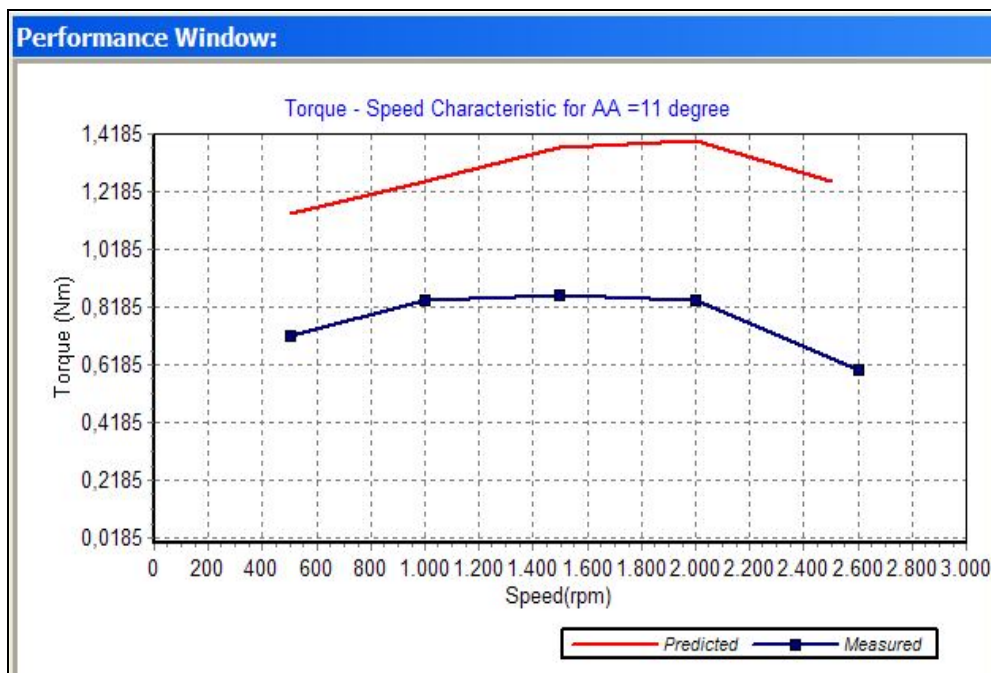
a) AA=0.0°



b) AA=4.0°



c) AA=7.5°



d) AA=11°

Figure 4.30 Torque-Speed Curves without End-Leakage at AA = 0°, 4°, 7.5° and 11° respectively

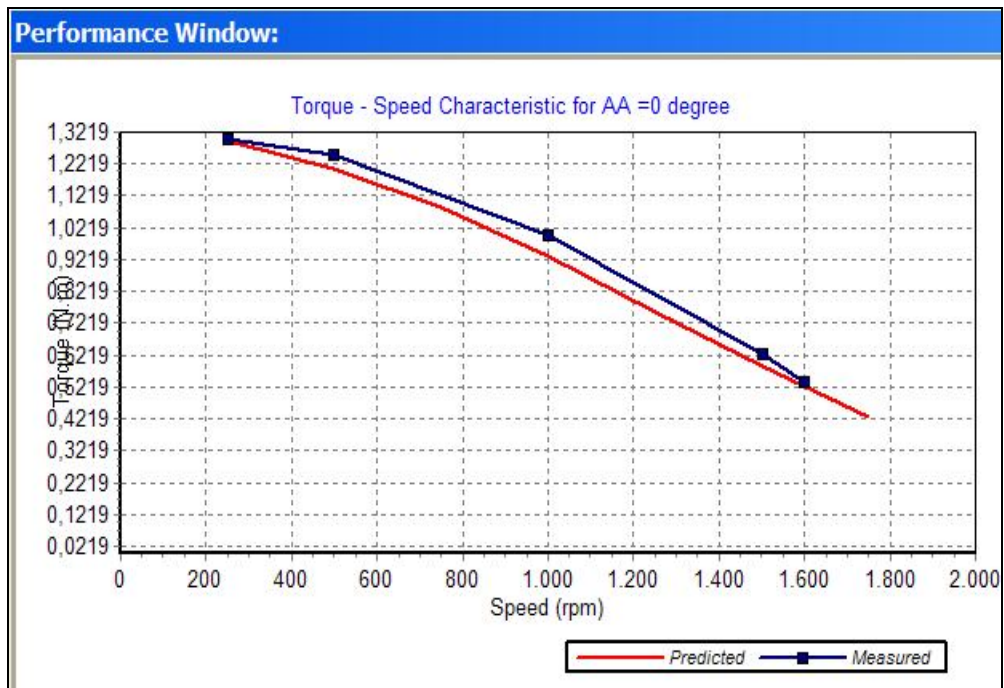
Table 4.4 Predicted-Measured Torques and Percentage Errors without End-Leakage Correction

AA = 0°				AA = 4°			
rpm	Measured	Predicted	Err %	rpm	Measured	Predicted	Err %
250	1.300	1.765	-35.74	250	1.380	1.935	-40.25
500	1.250	1.655	-32.43	500	1.380	1.913	-38.63
1000	1.000	1.353	-35.25	1000	1.250	1.797	-43.74
1500	0.625	0.874	-39.76	1500	0.900	1.508	-67.51
1600	0.540	0.786	-45.47	2000	0.600	0.990	-65.07
AA = 7.5°				AA = 11°			
rpm	Measured	Predicted	Err %	rpm	Measured	Predicted	Err %
500	1.250	1.738	-39.04	500	0.720	1.139	-58.21
1000	1.255	1.766	-40.72	1000	0.840	1.253	-49.15
1500	1.170	1.707	-45.90	1500	0.860	1.371	-59.42
2000	0.800	1.480	-85.02	2000	0.840	1.395	-66.06
2400	0.600	1.060	-76.60	2600	0.600	1.157	-92.78

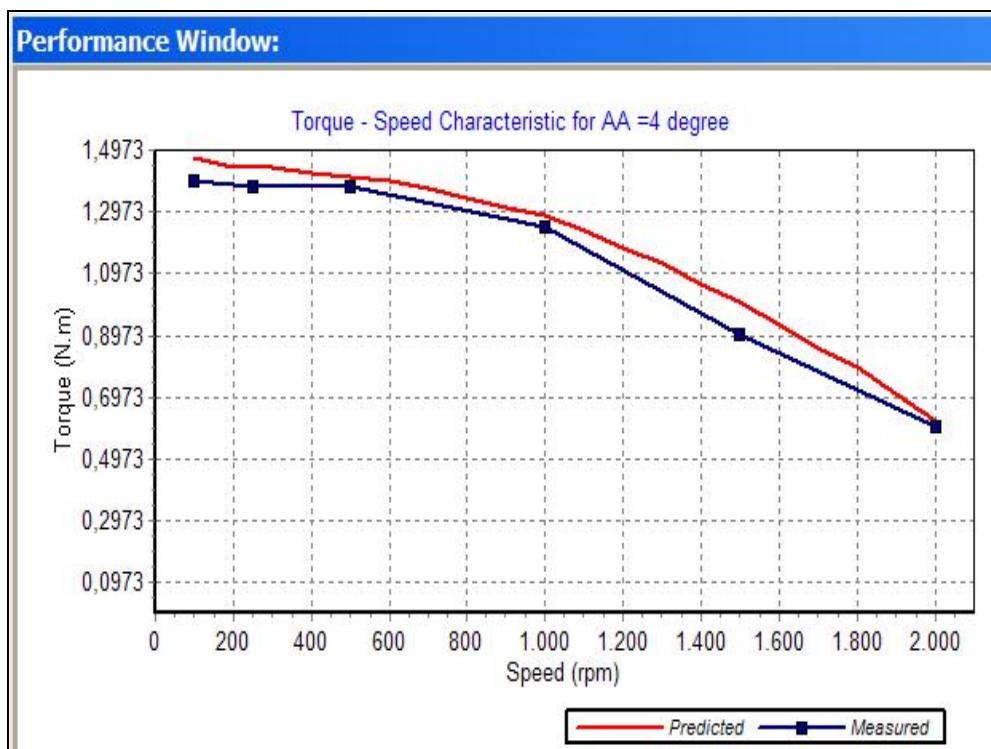
According to Table 4.4 and the figures above, the percentage error between the calculated and measured values of Torque reaches to 92.78 %. This high value adversely affects the performance calculations. In the next section, accuracy of the Torque-Speed curves, which includes the end leakage effect, are presented.

4.7.2.2. Torque – Speed Curves with End Leakage Correction

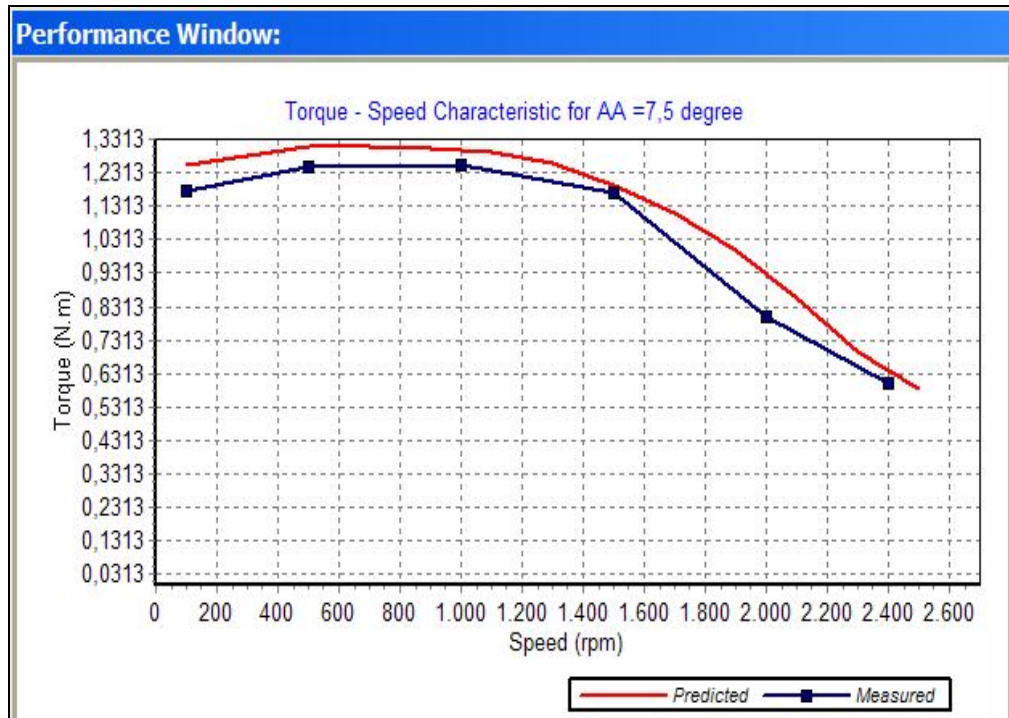
The figures below show the results of SRCAD performance analysis in which Torque Speed characteristics are calculated with end-leakage correction [13].



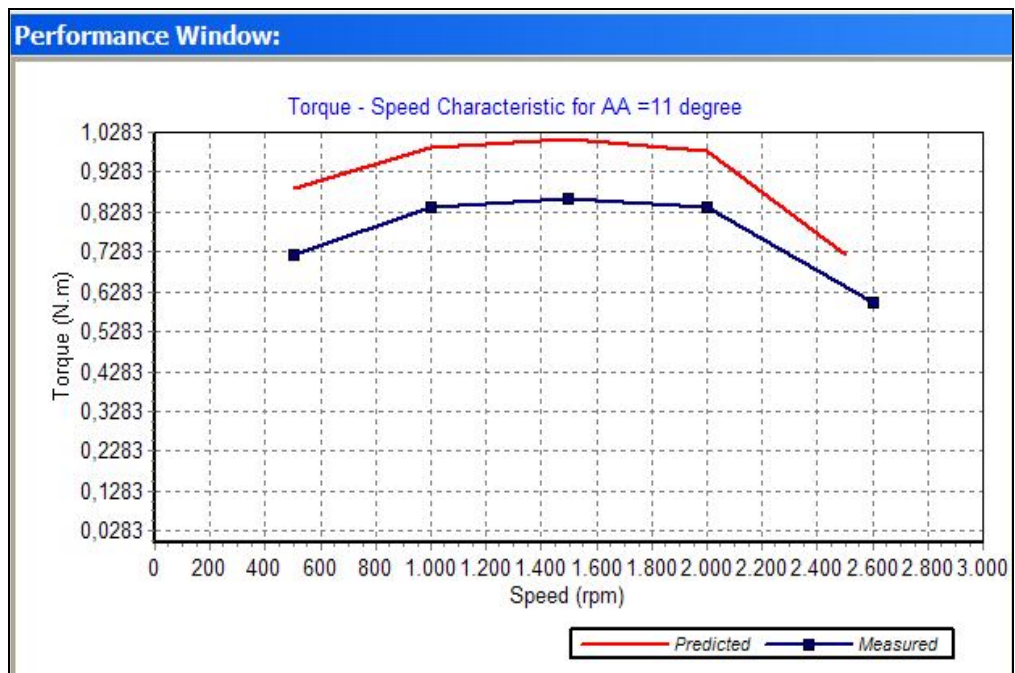
a) AA=0.0°



b) AA=4.0°



c) AA=7.5°



d) AA=11°

Figure 4.31 Torque-Speed Curves with End-Leakage at AA = 0°, 4°, 7.5° and 11° respectively

Table 4.5 Predicted-Measured Torques and Percentage Errors with End-Leakage Correction

AA = 0°				AA = 4°			
Rpm	Measured	Predicted	Err %	rpm	Measured	Predicted	Err %
250	1.300	1.296	0.29	250	1.380	1.443	-4.58
500	1.250	1.206	3.51	500	1.380	1.411	-2.22
1000	1.000	0.935	6.54	1000	1.250	1.287	-2.93
1500	0.625	0.586	6.32	1500	0.900	1.005	-11.69
1600	0.540	0.516	4.36	2000	0.600	0.624	-3.92
AA = 7.5°				AA = 11°			
Rpm	Measured	Predicted	Err %	rpm	Measured	Predicted	Err %
500	1.250	1.306	-4.52	500	0.720	0.887	-23.23
1000	1.255	1.296	-3.24	1000	0.840	0.989	-17.80
1500	1.170	1.195	-2.11	1500	0.860	1.011	-17.58
2000	0.800	0.956	-19.47	2000	0.840	0.980	-16.68
2400	0.600	0.639	-6.43	2600	0.600	0.662	-10.31

According to Table 4.5 and the figures above, the percentage error between the calculated and measured values of Torque is not more than 23 %. This value is in the tolerable limits. In addition, in some cases the percentage error is nearly 0 %. Therefore, while calculating the Torque - Speed curves, the end-leakage correction method that is proposed by L. Burak Yalçiner [13] should be used. So in the optimization algorithm this method is used.

4.7.3. The Accuracy of Phase Current Characteristics

The "Phase Current – Time" data and waveforms are measured by L. Burak Yalçiner [13]. In this study by using the analysis part of SRCAD the Phase Current –Time characteristics of SR2 for advance angles 0°, 4° and 11° at speeds 1000, 1600 and 2000 rpm are calculated, compared with the measured data, and tabulated at Table 4.6.

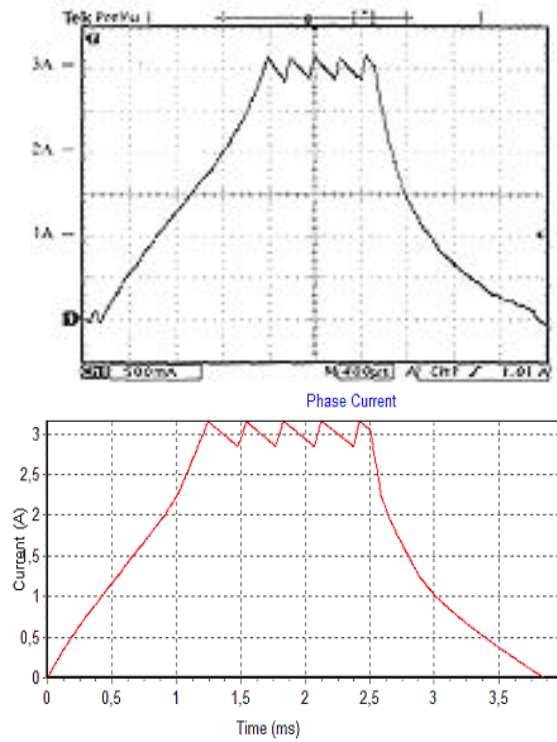


Figure 4.32 a) Measured Current Waveform at 1000 rpm, AA=0°
b) Calculated Current Waveform at 1000 rpm, AA=0°

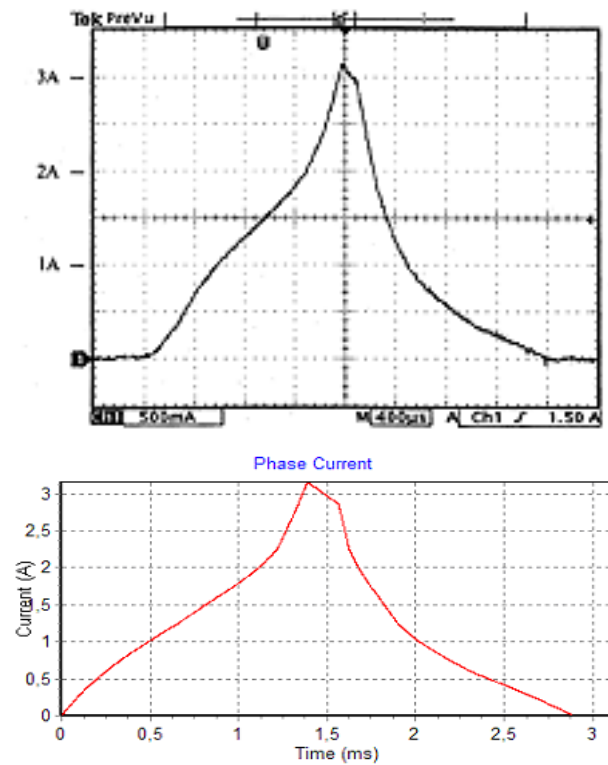


Figure 4.33 a) Measured Current Waveform at 1600 rpm, AA=0°
b) Calculated Current Waveform at 1600 rpm, AA=0°

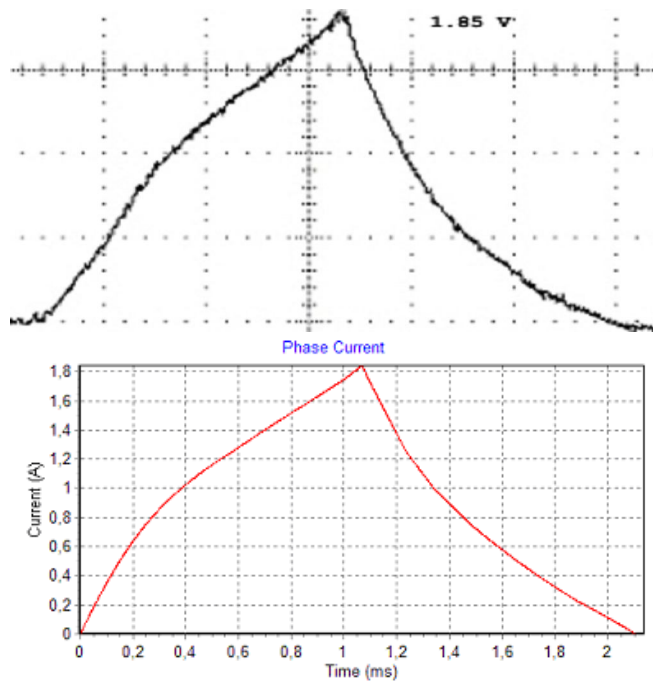


Figure 4.34 a) Measured Current Waveform at 2000 rpm, AA=4°
b) Calculated Current Waveform at 2000 rpm, AA=4°

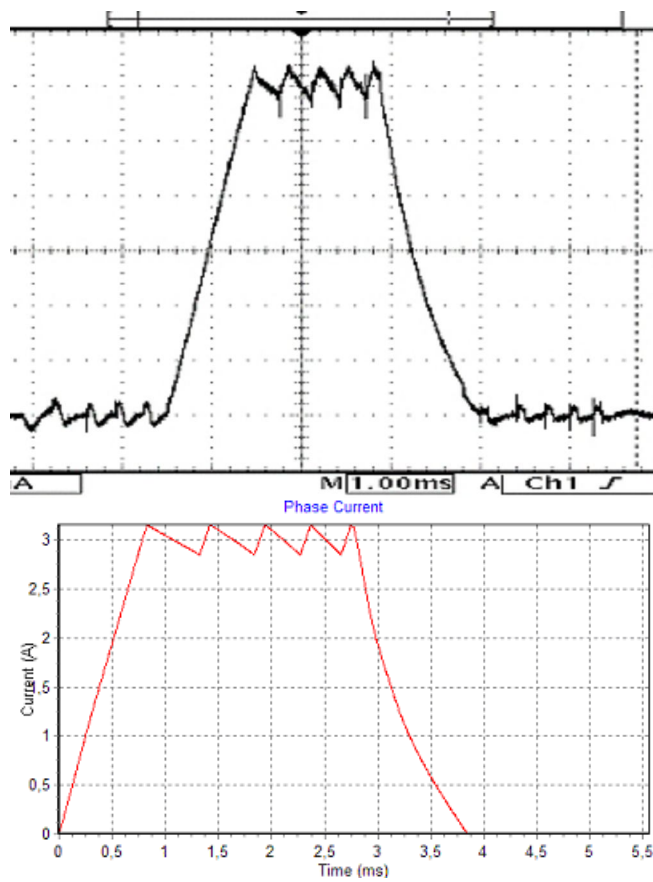


Figure 4.35 a) Measured Current Waveform at 1000 rpm, AA=11°
b) Calculated Current Waveform at 1000 rpm, AA=11°

Table 4.6 Rise Times and Fall Times of Calculated and Measured Current Waveforms

Speed (rpm)	Advance Angle	Measured and Predicted Rise Times (ms)	Measured and Predicted Fall Times (ms)	Peak Val- ues (A)	Number of Chops
1000	0	1.36 – 1.30	1.36 – 1.35	3.15 – 3.15	5 – 5
1600	0	1.47 – 1.40	1.44 – 1.35	3.15 – 3.15	1 – 1
2000	4	1.16 – 1.08	1.09 – 1.15	1.85 – 1.87	0 – 0
1000	11	0.9 – 0.83	1.1 – 1.1	3.15 – 3.15	5 – 5

According to Table 4.6 and above figures, calculated phase current waveforms is very similar to measured waveforms of SR2. Therefore, it is concluded that analysis part of the software program is running accurately. This analysis part can be used in the optimization program.

4.7.4. The Accuracy of Losses for SR2

In this study, Friction torque vs. rpm curve is measured as in Figure 4.36. After finding the electrical Torque that produced SR2, to find the mechanical torque, friction torque is found corresponding to speed and friction torque is decreased from the electrical torque.

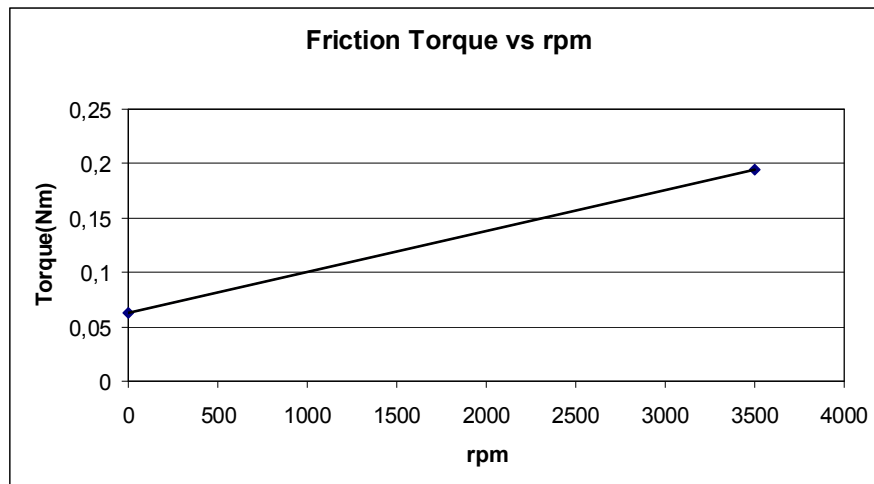


Figure 4.36 Friction Torque vs. rpm Curve

In this study, also, loss vs. speed and efficiency vs. speed curves are obtained. These curves are also compared with the measured curves of SR2.

Table 4.7 Comparison of Measured and Predicted Losses of SR2 for Load Torques at 0.5Nm and 1Nm

$T_L = 0.5 \text{ Nm}$				$T_L = 1.0 \text{ Nm}$			
Rpm	Measured Loss(W)	Predicted Loss(W)	Err %	rpm	Measured Loss(W)	Predicted Loss(W)	Err %
1200	64	75.943	-18.66	1200	104	96.21	7.49
1600	67	84.338	-25.88	1600	118	117.88	0.10
2000	73	99.91	-36.86	2000	133	135.49	-1.87
2500	95	122.81	-29.27	2500	175	172.25	1.57

If the Load torque 0.5 Nm is considered, the maximum percentage error is 36.86% at 2000 rpm. But if the load torque is increased to 1.0 Nm, the percentage error between the predicted and measured losses decreases. For example at 2000 rpm, the percentage error becomes 1.87%.

Table 4.8 Comparison of Measured and Predicted Efficiencies of SR2 for Load Torques at 0.5 Nm and 1Nm

$T_L = 0.5 \text{ Nm}$				$T_L = 1.0 \text{ Nm}$			
Rpm	Measured Efficiency	Predicted Efficiency	Err %	rpm	Measured Efficiency	Predicted Efficiency	Err %
1200	50	45.25	9.50	1200	55	56.64	-2.98
1600	56	49.88	10.93	1600	59	58.81	0.32
2000	58	51.18	11.76	2000	60	60.76	-1.26
2500	58	51.55	11.12	2500	60	60.33	-0.55

By looking at the Table 4.8, it is seen that the percentage error between the predicted and measured efficiency is not more than 12% for the load torque 0.5 Nm and also not more than 3% for the load torque 1.0 Nm.

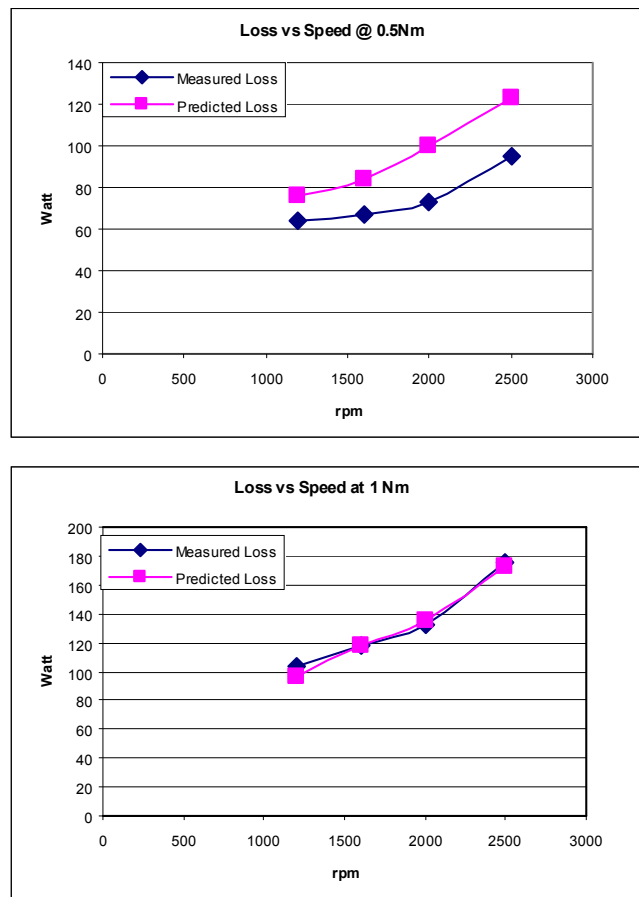


Figure 4.37 Measured and Predicted Losses for SR2

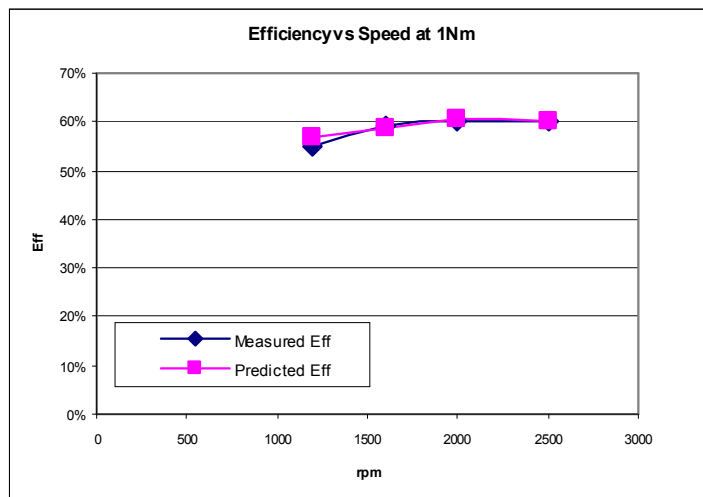
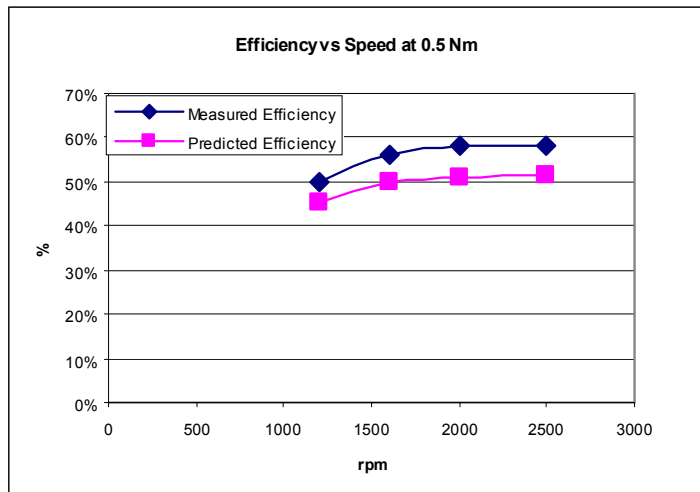


Figure 4.38 Measured and Predicted Efficiencies for SR2

By looking the Figure 4.37, it is seen that the shape of the predicted loss curves are similar to measured loss curves. The same manner is also true for the Figure 4.38, the predicted efficiency curves and measured efficiency curves are similar to each other.

4.8. Optimization

4.8.1. Introduction

The purpose of this part of the thesis is to manage the optimization of SRM by minimizing the weight and cost of the motor while satisfying the performance requirements. For this purpose, a project is designed in modeFRONTIER and an analysis program is written.

The basic mathematical optimization problem is to minimize a function of n system parameters x_1, x_2, \dots, x_n subject to certain constraints.

A typical constraint minimization problem can be formulated as follows.

Minimize $f(\bar{x})$, so subject to

$$\begin{aligned} p_i(\bar{x}) &= a_i & \text{for } i = 1, \dots, m_1 \\ q_i(\bar{x}) &\leq b_i & \text{for } i = 1, \dots, m_2 \\ \bar{x} &\in X \subset R \\ c_k &\leq x_k \leq d_k & \text{for } k = 1, \dots, n \end{aligned} \tag{4.30}$$

where:

- $f(\bar{x})$ is the function to be minimized, the objective function,
- $p_i(\bar{x})$ are the equality constraint functions,
- $q_i(\bar{x})$ are the inequality constraint functions,
- \bar{x} is the search vector, vector of independent optimization parameters in terms of the cost function and constraints can be computed [13].

In the optimization process, objective function, equality constraint functions and the inequality constraint functions are required. The designer should determine these functions.

Since the purpose of this part of the thesis is to manage the optimization of SRM by minimizing the weight and cost of the motor while satisfying the performance requirements objective function is chosen as the weight of the SRM. The chosen constraint functions and the independent optimization parameters are briefly explained in the next sections.

For the optimization, there are many methods in literature. However, in this study, an optimization tool, which is named as modeFRONTIER, is used. Therefore, different optimization methods involved in modeFRONTIER is attempted for the optimal design of SRM and their performance is evaluated.

In the optimization process, system starts with an initial design and goes on with iterative steps. At each step, objective functions' and constraint functions' values should be computed by using independent optimization parameters. These values are computed by the Analysis part of the SRCAD, which is newly designed for the optimization of SRMs.

4.8.2. Independent Optimization Parameters

In this section, independent optimization parameters for the optimized design method will be introduced.

In this study, 9 independent parameters are selected. These are dimensional parameters of SRM [13]:

d_u : inner diameter of rotor
 d_i : outer diameter of rotor
 D_o : outer diameter of stator
 L_c : core length
 y_b : back core width
 t_s : stator pole width
 t_r : rotor tooth width
 g : airgap length
 AA : Advance Angle

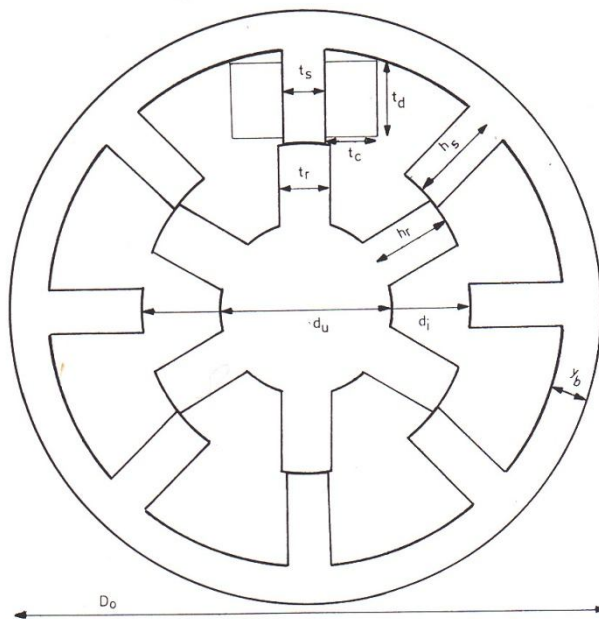


Figure 4.39 Independent Physical Parameters

The set of independent optimization parameters is not unique. It is possible to use different combinations of independent parameters. In addition, the number of phases is chosen by the user as necessary.

4.8.3. Constraints

Limitations imposed by properties of materials and manufacturing difficulties put some constraints on the final design of the motors. Therefore, as well as the performance functions these constraints should be considered in the optimization process.

The constraints are divided into two groups:

- Equality Constraints
- Inequality Constraints

4.8.3.1. Equality Constraint

The steady state average torque produced by SRM at given speed is set as the only equality constraint. This constraint is necessary to specify the rating of the motor.

For SR2 [13] the equality constraint is:

$$T_{avg} = 1.7 Nm \quad (at \ 520 \ rpm) \quad (4.31)$$

4.8.3.2. Inequality Constraints

4.8.3.2.1. Inner Diameter of the Rotor

Since the rotor lamination should support itself and shaft diameter should be capable of withstanding maximum overloads; a lower limit for inner diameter of rotor should be imposed [7]. In the software minimum motor shaft diameter is taken as 10mm.

For SR2 [13] the constraint of inner diameter of rotor is:

$$d_u \geq 10mm \quad (4.32)$$

4.8.3.2.2. Air Gap Length

Due to the limitations introduced by the manufacturing process, a minimum airgap distance of 0.25 mm is assumed.

For SR2, the constraint of air gap length is:

$$0.25mm \leq g \leq 0.5mm \quad (4.33)$$

4.8.3.2.3. Advance Angle

For SR2, the constraint of advance angle is:

$$7.5^{\circ} \leq AA \leq 22.5^{\circ} \quad (4.34)$$

4.8.3.2.4. Current Density

An upper limit for rms value of current density, J_c is set in order to compare the resultant motor with similar rating induction motors.

For SR2, the constraint of current density is:

$$J_c \leq 6.5 \text{ A/mm}^2 \quad (4.35)$$

4.8.3.2.5. Efficiency

The steady state running efficiency of the motor at 520 rpm is desired to be larger than 40 %.

For SR2, the constraint of efficiency is:

$$\eta \geq 40\% \quad (4.36)$$

4.8.4. The Object Function: Weight of SRM

In this study, the object function to be minimized is chosen as the total weight of SRM including iron and copper weight. The formulation of the object function is used as given in [7].

$$W_T = W_{Fe} + W_{Cu} \quad (4.37)$$

4.8.4.1. Calculation of Iron Weight

In terms of the motor dimensions, total iron weight of a given SRM can be calculated as follows:

$$W_{Fe} = \sigma_{Fe} (\pi (D_o - y_b) y_b L_c + N_s t_s h_s L_c + \frac{\pi}{4} d_u^2 L_c + N_r t_r h_r L_c) \quad (4.38)$$

where σ_{Fe} is density of iron, 7.55 gr/cm³.

4.8.4.2. Calculation of Copper Weight

Total copper weight may be expressed in terms of the motor dimensions as given in Equation 4.33:

$$W_{Cu} = \sigma_{Cu} N_s k_w t_c t_d l_{av} \quad (4.39)$$

σ_{Cu} : density of copper, 8.88 gr/cm³

k_w : is winding factor

t_c : coil width

t_d : coil depth

l_{av} : is the average coil length and

$$l_{av} = 2 (t_s + t_c) + 2 (L_c + t_c) \quad (4.40)$$

The values, t_c and t_d which are width and height of coil are can be calculated from independent parameters [7], using equations Equation 4.35 – Equation 4.40 and Figure 4.40.

From Figure 4.40;

$$c_1 = \frac{d_i}{2} + g + a \quad (4.41)$$

$$c_2 = c_1 \cdot \tan\left(\frac{\pi}{N_s}\right) \quad (4.42)$$

$$c_3 = \frac{D_o}{2} - y_b \quad (4.43)$$

$$c_4 = c_3 \cdot \cos\left(\sin^{-1} \frac{c_2}{c_3}\right) \quad (4.44)$$

$$t_c = c_2 - \frac{t_s}{2} \quad (4.45)$$

$$t_d = c_4 - c_1$$

(4.46)

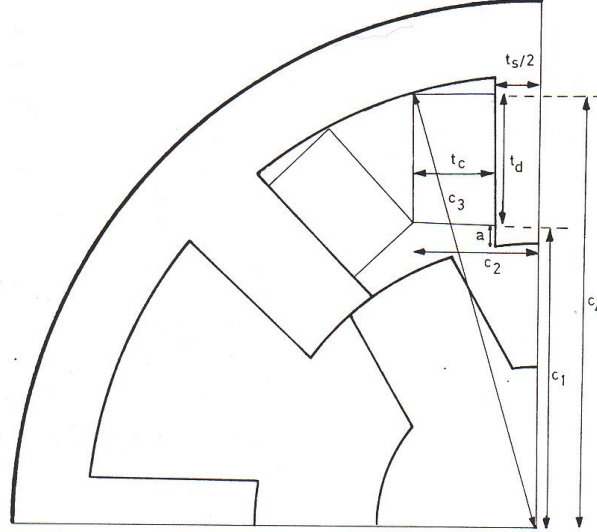


Figure 4.40 Winding Arrangement [7]

4.8.5. Optimization Project at modeFRONTIER

4.8.5.1. Introduction

One of the aims of this thesis is to design and optimize of switch reluctance motors. For this purpose, a project is designed in modeFRONTIER.

modeFRONTIER is a multidisciplinary design optimization and multi-objective optimization and design environment, written to allow easy coupling to almost any computer aided engineering (CAE) tool whether commercial or in-house. Created by Esteco, modeFRONTIER provides an environment, which allows product engineers and designers to integrate their various CAE tools,

such as CAD, Finite Element Structural Analysis and Computational Fluid Dynamics (CFD) software.

modeFRONTIER is a GUI driven wrapper around the CAE tool, performing the optimization by modifying the value assigned to the input variables, and analyzing the outputs as they can be defined as objectives and/or constraints of the design problem. [19]. The user manual of modeFRONTIER describes how a given problem can be handled [20].

Running an analysis tool within the modeFRONTIER framework is extremely straightforward. There are also direct interfaces for Excel, Matlab and Simulink, CATIA, Pro/ENGINEER, the ANSYS Workbench and Amesim.

To understand the modeFRONTIER generally, Figure 4.41 may be inspected , which shows a simple example of both process integration and optimization.

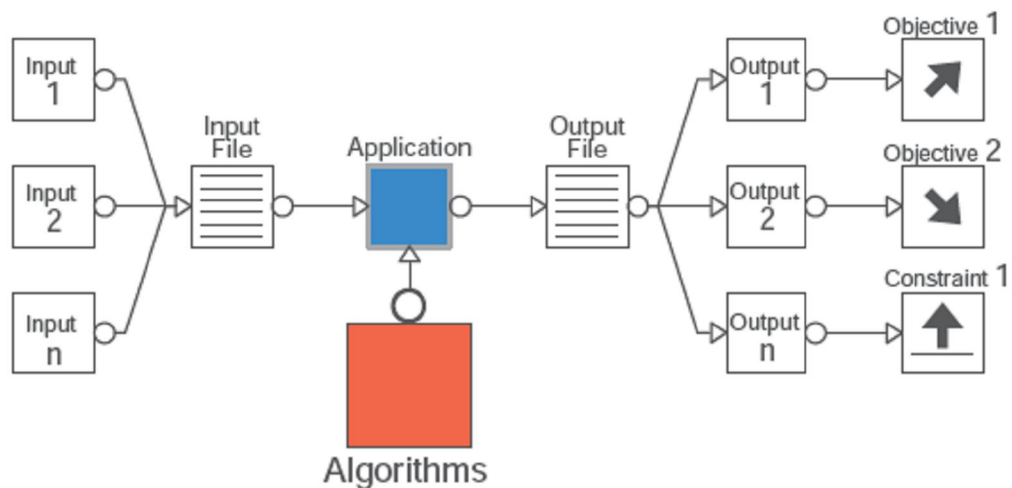


Figure 4.41 Process Integration and Optimization of modeFRONTIER in General

With modeFRONTIER only few steps are required for achieving our goals:

- Describe the problem (parameterize)
- Set goals (objectives)
- Choose optimization strategy

Firstly, the input parameters are defined to describe the problem. These parameters are linked to an input file that can be chosen as a notepad, excel or word file. This input file is for the analysis program, which finds the performance of the system according to input parameters.

The application program finds the performance of the system, then writes these parameters to an output file that can be chosen as a notepad, excel or word file. The output file is the file generated by the application part of the design for storing the output data. The output variables can be the objects or constraints of the problem.

After the application part computes the output variables, modeFRONTIER recomputes the input variables for the problem to reach the objectives and constraints put on the output variables according to the optimization algorithm.

Using a wide set of DOE (Design of Experiment) and Optimization Algorithms, modeFRONTIER efficiently searches the design space for the optimum solution.

The figure below is the project that we have designed for the SRM. In this design, input output files are named as SRM_input.dat and SRM_output.dat respectively. In addition, application program is named as calculateSRM.exe.

In this section, the steps how to design a SRM project is explained.

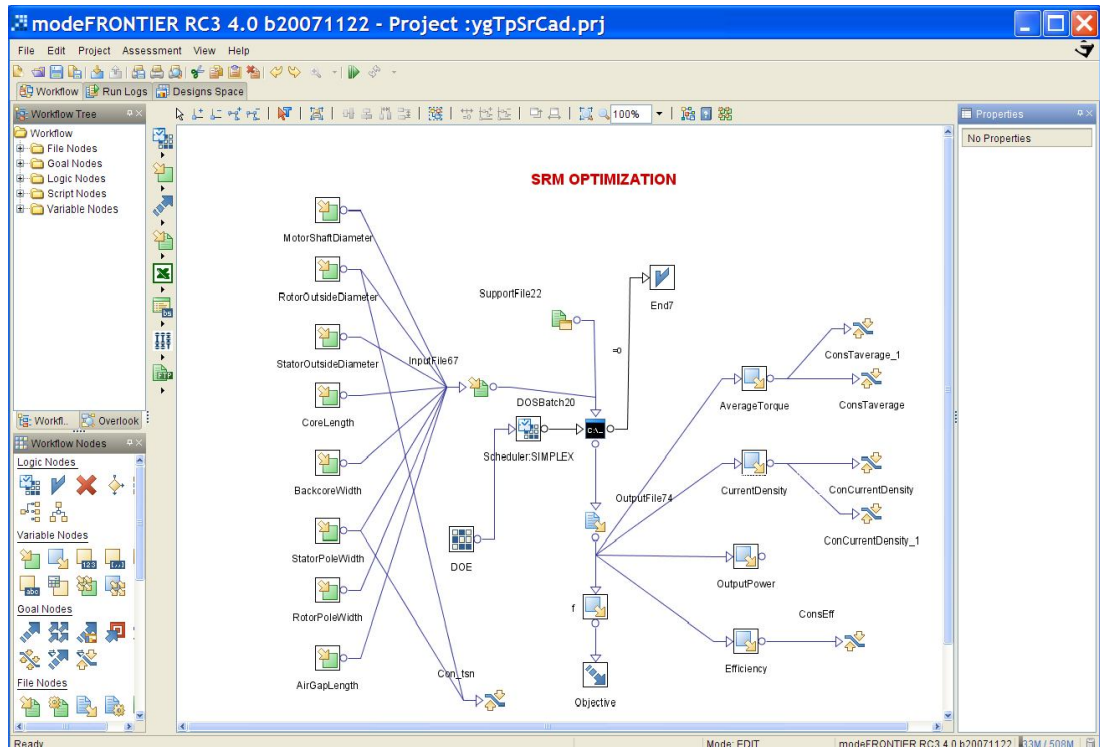


Figure 4.42 Designed Project at modeFRONTIER

4.8.5.2. Design Steps of modeFRONTIER Project

In this section, to design a modeFRONTIER project is introduced step by step. Since the modeFRONTIER is a generic tool, so user can change the design specifications. These steps are given for this purpose.

- 1) After opening the modeFRONTIER workflow, the independent input parameters for the SRM is added to project as the Input Variable. As stated before the input variables are:

d_u : inner diameter of rotor
 d_i : outer diameter of rotor
 D_o : outer diameter of stator
 L_c : core length
 y_b : back core width
 t_s : stator pole width
 t_r : rotor tooth width
 g : airgap length

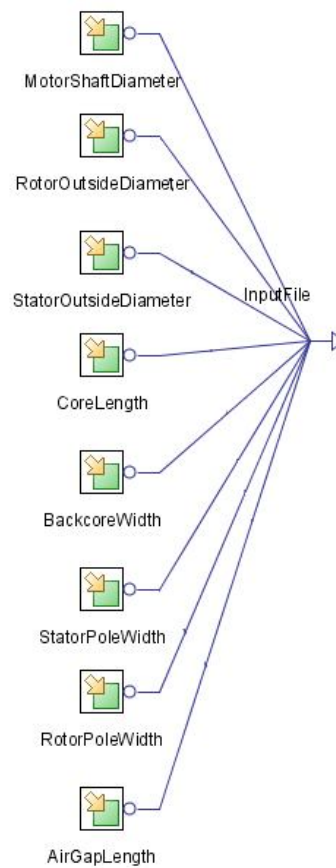


Figure 4.43 Input Variables

- 2) Since the input variables are the input of the SRCAD, these variables should be written into a file. For this reason, in the modeFRONTIER "Input Files" are used to keep data. In this project, "SRM_input.dat" is used as the Input File. This file is dynamically generated at run-time, inserting into its fixed structure the values of one or more scalar values. In the figure, the arrangement of the input file format for the variables is shown.

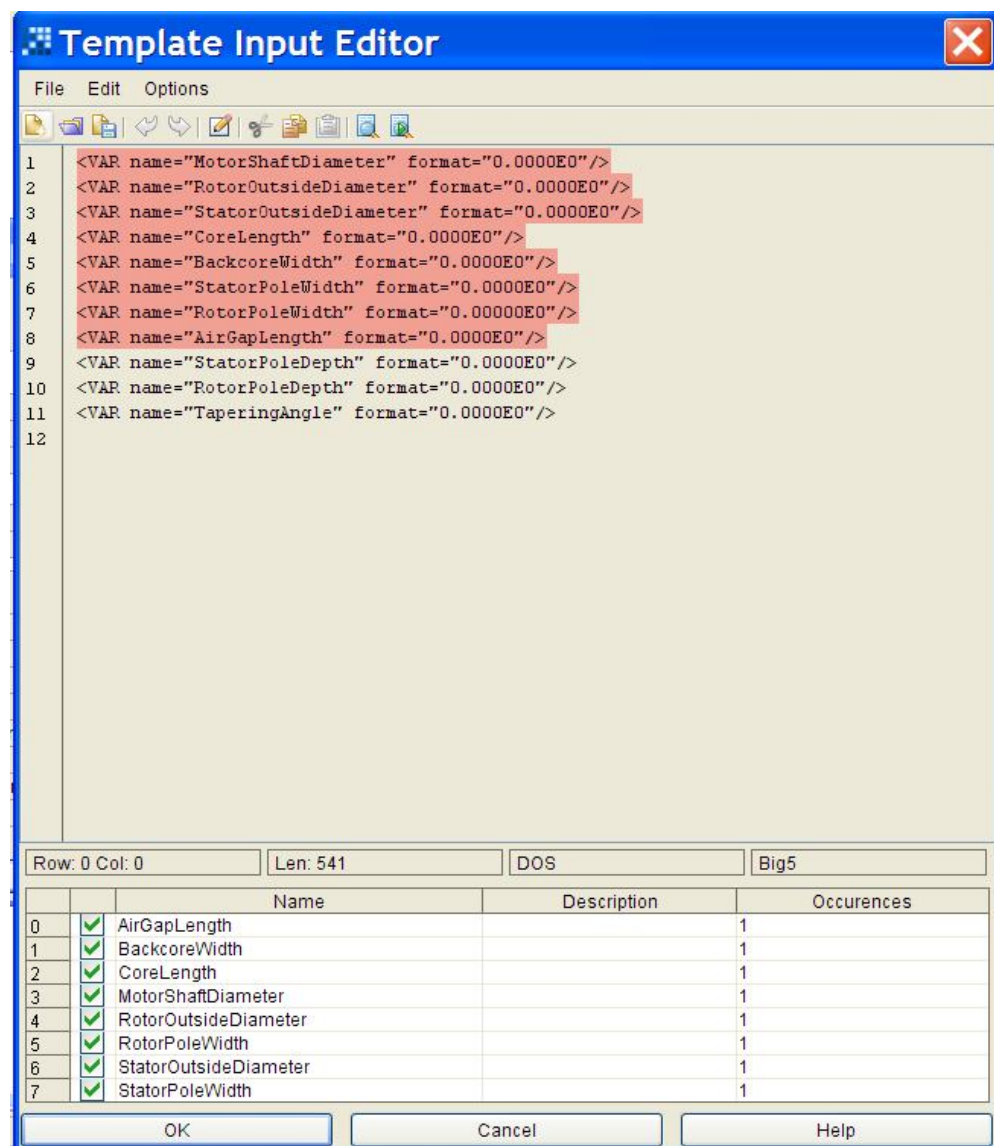


Figure 4.44 Input File Editor

- 3) In this project, DosBatch program is chosen to call the SRCAD performance calculations program. "CalculateSrm.exe" is the program, which computes the performance of the SRM according to Input File, "SRM_input.dat".

DosBatch program takes the input variables and calls the "CalculateSrm.exe" to find the performance according to inputs.

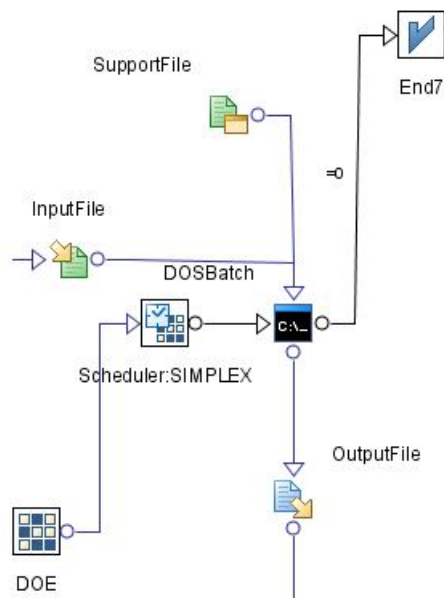


Figure 4.45 Application Program

- 4) In this step, output parameters of the SRCAD are added to design. In addition, for keeping these parameters an Output File is used. "SRM_output.dat" is chosen as the Output File. This file is written by SRCAD program, CalculateSrm.exe. The output parameters are:

Average Torque
Current Density
Output Power
Efficiency
Weight

Weight is also object of the design.

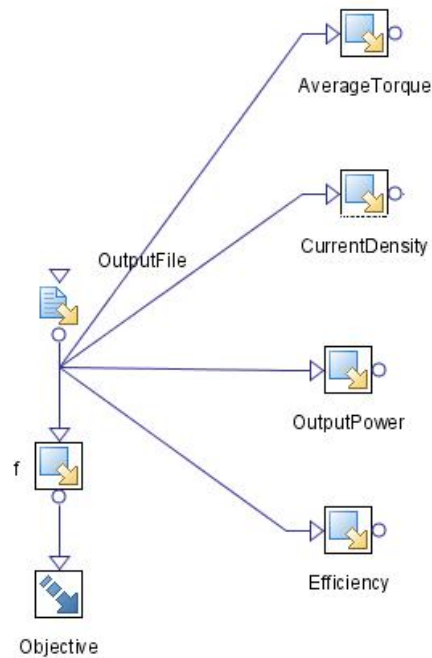


Figure 4.46 Output Variables

- 5) The constraints of the system are added to design like seen in Figure 4.47. These constraints are explained in Section 4.8.3.

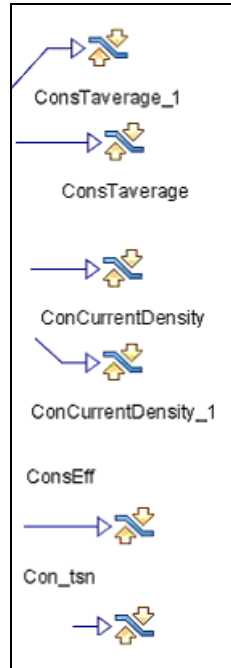


Figure 4.47 Constraints

6) In this step, Scheduler Node is chosen. The Scheduler Node determines which designs need to be evaluated. The strategy used to select the DOE designs of interest, and to generate new designs is determined by the scheduling algorithm selected. In our study for optimization, "Simplex" optimization method is chosen. Main features of Simplex Algorithm are:

- a. Obeys boundary constraints on continuous variables.
- b. Allows user defined discretization (base).
- c. Enforces user defined constraints by objective function penalization.
- d. The $n+1$ independent points of the initial simplex can be evaluated concurrently.

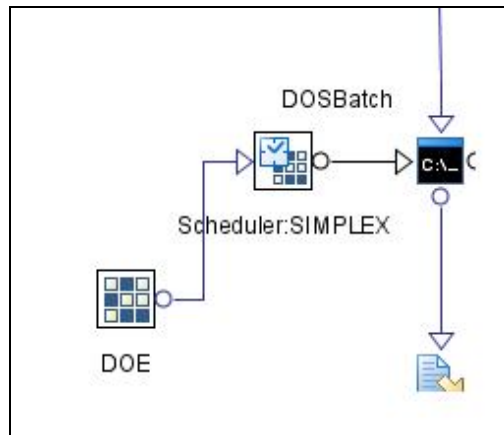


Figure 4.48 Optimization Algorithm

- 7) By using DOE properties, user can add initial parameters for the design.

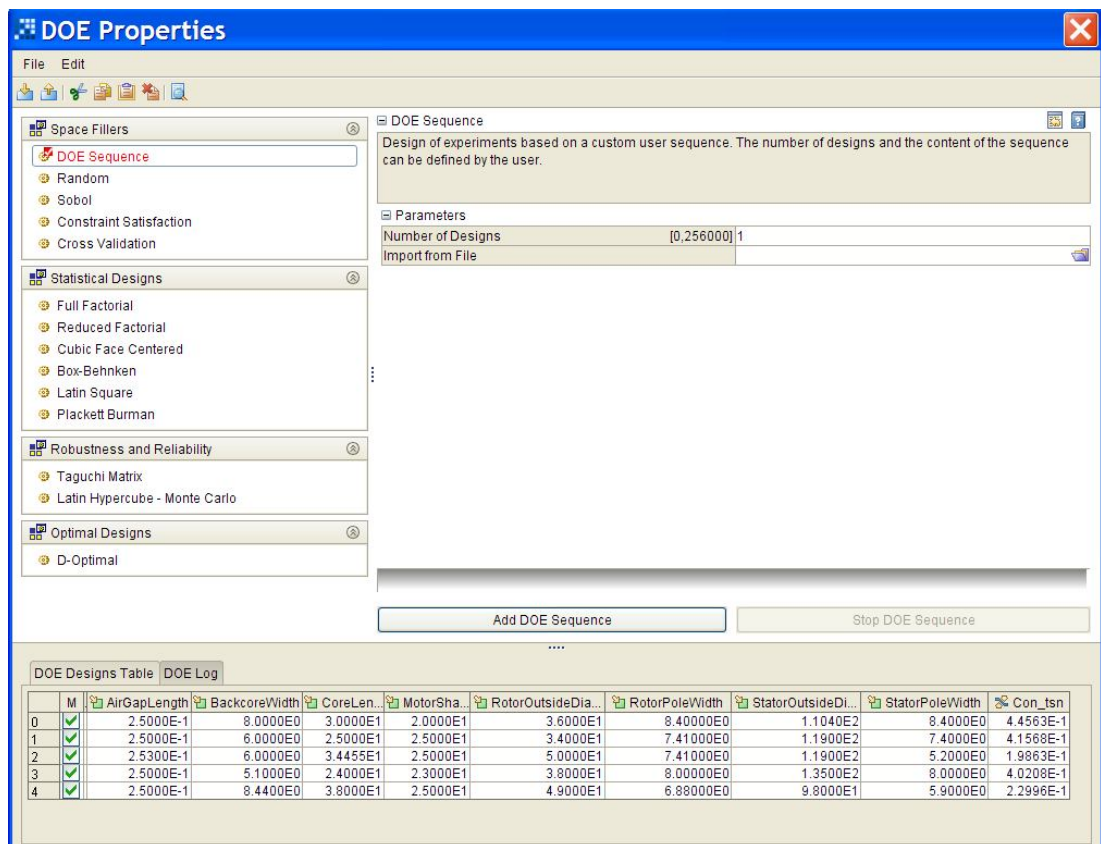


Figure 4.49 DOE Properties

4.8.6. Optimization Result

SRCAD is tested on motor SR2, properties of which is given in Section 4.7, to confirm the ability of reaching optimum designs. Several Optimization Algorithms are tested and the results are compared with one another. These designs are done at the low operating speed of 520 rpm.

4.8.6.1. Results of Simplex Algorithm

For the optimization of SR2, firstly Simplex Algorithm is used for different initial designs. Although Simplex algorithm is tested for single and three phase induction machine, and observed that it is not suitable for the optimization of electrical machines, it is decided to test this for the SR motor to observe whether it is suitable or not. The details about the Simplex Algorithm is given in the Appendix B.

The results are listed in Table 4.9. Initial designs are indicated by capital letter I and final results are indicated by capital letter F.

Three different initial vectors are chosen for the Simplex algorithm. I1 is the same size as the SR2 and the other two vectors have smaller size than the SR2.

The optimization conditions are specified below.

Parameters of Simplex Algorithm:

Maximum Number of Iterations: 1000

Final Termination Accuracy: 1.0E-5

Constraint Penalty Policy: Automatic

Table 4.9 Initial and Final Design of SR2 by using Simplex Algorithm

Design Vector	I1	F1	I2	F2	I3	F3
Advance Angle(degree)	10.000	11.530	7.620	10.159	7.500	9.803
Air Gap Length(mm)	0.250	0.250	0.250	0.250	0.250	0.297
Backcore Width(mm)	8.000	5.586	6.000	8.136	7.860	4.787
Core Length(mm)	30.000	26.900	24.260	27.130	27.630	29.405
Rotor Inner Diameter(mm)	20.000	20.880	20.000	30.138	20.000	37.096
Rotor Outside Diameter(mm)	36.000	41.840	34.075	37.176	44.520	37.834
Rotor Pole Width(mm)	8.400	7.150	7.410	8.090	9.322	6.217
Stator Outside Diameter(mm)	110.40	113.24	118.43	122.52	100.00	109.32
Stator Pole Width(mm)	8.400	7.391	7.400	6.528	9.311	8.446
Average Torque(Nm)	1.385*	1.670	1.082*	1.670	1.793*	1.670
Current Density(A/mm ²)	7.54	7.700	5.41	7.674	10.734*	7.700
Efficiency(%)	38.14*	42.720	40.93	43.893	40.99	42.079
Output Power(W)	75.45	90.950	58.91	90.950	97.63	90.950
Weight(kg)	2.035	2.072	1.979	2.174	1.569	2.123
Norm. Stator Pole Width	0.440	0.337	0.414	0.335	0.399	0.426
Number of Iterations	---	699	---	756	---	728

In Table 4.9, it can be observed that for the initial design I1, the average torque and efficiency is lower than the desired values. "*" means constraints are not satisfied. After 699 steps of iteration of the optimization process, final optimum value is reached. It is seen that for the F1, all the constraints are satisfied and also weight converges to a minimum. The final design is about 2 % heavier than the reference motor (I1).

For Initial Design 1:

- Average Torque : 1.385 Nm, Not satisfied, percentage error is 18.5 %, this error is high, it is not acceptable.
- Current Density : 7.54 A/mm², satisfied.
- Efficiency : 38.14 % , Not satisfied, percentage error is 4.65 %, error is in the acceptable range.
- Normalized Stator Pole Width : 0.44, satisfied.
- Weight : 2.035 kg

For Final Design 1:

- Average Torque : 1.67 Nm, satisfied, since in the modeFRONTIER constraint is chosen in the range 1.67 Nm - 1.73 Nm.
- Current Density : 7.7 A/mm², satisfied.
- Efficiency : 42.72 % , satisfied.
- Normalized Stator Pole Width : 0.337, satisfied.
- Weight : 2.072 kg, converged since the weight is 1.81% larger than the reference motor weight.
- Number of Iterations : 699
- Elapsed Time :1267 seconds

For the initial design I2, average torque is lower than the desired values. But at the end of the 756 iterations, simplex method converges and also satisfies the constraints. This final design is about 6 % larger than the reference motor weight.

For Initial Design 2:

- Average Torque : 1.082 Nm, Not satisfied, percentage error is 36.4 %, this error is high, it is not acceptable.
- Current Density : 5.41 A/mm², satisfied.
- Efficiency : 40.93 % , satisfied.
- Normalized Stator Pole Width : 0.414, satisfied.
- Weight : 1.979 kg

For Final Design 2:

- Average Torque : 1.67 Nm, satisfied, since in the modeFRONTIER constraint is chosen in the range 1.67 Nm - 1.73 Nm.
- Current Density : 7.67 A/mm², satisfied.
- Efficiency : 43.89 % , satisfied.
- Normalized Stator Pole Width : 0.335, satisfied.
- Weight : 2.174 kg, converged since the weight is 6.83% larger than the reference motor weight.
- Number of Iterations : 756
- Elapsed Time :1371 seconds

For the initial design I3, in this case average torque and current density is higher than the desired values. The initial weight is about 75 % of the reference motor weight. At the end of the 728 iterations, it also converges to a minimum value by satisfying the constraints. This time the final design is about 5 % larger than the reference motor in weight.

For Initial Design 3:

- Average Torque : 1.79 Nm, Not satisfied, percentage error is 5.3 %, error is the acceptable range.
- Current Density : 10.734 A/mm², Not satisfied percentage error is 39.4 %, this error is high, it is not acceptable.
- Efficiency : 40.99 % , satisfied.
- Normalized Stator Pole Width : 0.399, satisfied.
- Weight : 1.569 kg

For Final Design 3:

- Average Torque : 1.67 Nm, satisfied, since in the modeFRONTIER constraint is chosen in the range 1.67 Nm - 1.73 Nm.
- Current Density : 7.7 A/mm², satisfied.
- Efficiency : 42.08 % , satisfied.
- Normalized Stator Pole Width : 0.426, satisfied.
- Weight : 2.123 kg, converged since the weight is 4.32% larger than the reference motor weight.
- Number of Iterations : 728
- Elapsed Time : 1321 seconds

By looking the results, it can be said that program is successful for optimizing the motor weight while satisfying the constraints. If we compare the results according to weight, it can be observed that although the initial designs do not satisfy the constraints, they all converge to a minima. Further experiments with larger initial designs than the reference motor are highly desirable. However, it appears that this algorithm gives successful results for the SR motor optimization problem.

4.8.6.2. Results of MOSA Algorithm

For the optimization of SR2, MOSA (Multi Objective Simulated Annealing) Algorithm is used for different initial designs. The same initial designs that are used for the Simplex algorithm are used again. The results are listed in Table 4.10.

The original Simulated Annealing algorithm for optimization is a scalar-objective algorithm. MOSA is a genuine multi-objective optimization algorithm. The evolution of the system is controlled by an external parameter, the temperature (T); an initial configuration undergoes a random perturbation, and the difference in "energy" (E) between the new state and the old one is evaluated. At high T nearly every new configuration is accepted; at low T only the favourable transitions are allowed. E is suitably defined in the multi-objective ambit, involving the concepts of Pareto optimality and dominance. The temperature temporal evolution is governed by the temperature scheduler:

after an initial hot phase ($T > 0$), a cold phase follows ($T = 0$).

The hot phase increases the algorithm robustness, while the cold phase supports the algorithm convergence. The random spatial perturbation is controlled by the perturbation length scheduler[22].

The optimization conditions are specified below.

Parameters of MOSA Algorithm:

Number of Iterations: 1000

Initial Temperature (T): 0.1

Fraction of 'Hot' Iterations: 0.5

Minimum Perturbation Length: 0.05

Feasibility of Points: No Unfeasible Points

Random Generator Seed: 1

Table 4.10 Initial and Final Design of SR2 by using MOSA Algorithm

Design Vector	I1	F1	I2	F2	I3	F3
Advance Angle	10.000	9.042	7.620	7.987	7.500	7.634
Air Gap Length	0.250	0.262	0.250	0.265	0.250	0.256
Backcore Width	8.000	8.947	6.000	5.379	7.860	7.858
Core Length	30.000	28.216	24.260	26.745	27.630	24.942
Rotor Inner Diameter	20.000	19.699	20.000	21.864	20.000	15.103
Rotor Outside Diameter	36.000	35.120	34.075	38.144	44.520	47.826
Rotor Pole Width	8.400	7.487	7.410	8.139	9.322	9.277
Stator Outside Diameter	110.400	124.93	118.430	117.308	100.000	126.595
Stator Pole Width	8.400	6.639	7.400	7.277	9.311	10.122
Average Torque	1.385*	1.687	1.082*	1.710	1.793*	1.724
Current Density	7.54	7.440	5.41	7.542	10.734*	7.068
Efficiency	38.14*	43.923	40.93	43.910	40.99	45.452
Output Power	75.45	91.847	58.91	93.129	97.63	93.890
Weight	2.035	2.396	1.979	2.064	1.569	2.574
Norm. Stator Pole Width	0.440	0.361	0.414	0.364	0.399	0.404
Number of Iterations	---	16	---	511	---	128

In Table 4.10, as in the Simplex algorithm, the constraints for the I1,I2 and I3 are not satisfied. However, after 16 iterations of process, final design vector F1 is reached. If F1 is observed, it is seen that all the constraints are satisfied. But weight is larger than weight value found from the Simplex algorithm. The final design is about 18% heavier than the reference motor(I1).

For Final Design 1:

- Average Torque : 1.69 Nm, satisfied, since in the modeFRONTIER constraint is chosen in the range 1.67 Nm - 1.73 Nm.
- Current Density : 7.44 A/mm², satisfied.
- Efficiency : 43.93 % , satisfied.
- Normalized Stator Pole Width : 0.361, satisfied.
- Weight : 2.396 kg, Not converged since the weight is 18% larger than the reference motor weight.
- Number of Iterations : 16
- Elapsed Time : 30 seconds

However, if we consider the final design vector F2, it is seen that minimum weight value is obtained as well as all the constraints are satisfied. The final design is about 1.42 % larger than the reference motor weight. And, also after the iterations of 511 steps, this value is obtained.

For Final Design 2:

- Average Torque : 1.71 Nm, satisfied, since in the modeFRONTIER constraint is chosen in the range 1.67 Nm - 1.73 Nm.
- Current Density : 7.54 A/mm², satisfied.
- Efficiency : 43.91 % , satisfied.
- Normalized Stator Pole Width : 0.364, satisfied.
- Weight : 2.064 kg, converged since the weight is 1.42% larger than the reference motor weight.
- Number of Iterations : 511
- Elapsed Time : 922 seconds

For the F3, although, all the constraints are satisfied, weight is larger than the others. Weight of the final design is about 26% larger than the reference motor weight.

For Final Design 3:

- Average Torque : 1.72 Nm, satisfied, since in the modeFRONTIER constraint is chosen in the range 1.67 Nm - 1.73 Nm.
- Current Density : 7.07 A/mm², satisfied.
- Efficiency : 45.45 % , satisfied.
- Normalized Stator Pole Width : 0.404, satisfied.
- Weight : 2.574 kg, Not converged since the weight is 26% larger than the reference motor weight.
- Number of Iterations : 128
- Elapsed Time : 232 seconds

Finally, by looking the results, it can be said that MOSA algorithm is suitable for the optimization of SR motors.

4.8.7. Optimization Conclusion

After examining the final results in Table 4.9 and Table 4.10, it can be observed that initial designs are really important to get the optimum point for the optimization algorithms. For this reason user should try different initial designs. To optimize the SR motors, MOSA algorithm and Simplex algorithm can be chosen. MOSA algorithm is faster than the Simplex algorithm, however, both algorithms converge to local minima successfully.

4.9. GUI of SRCAD

4.9.1. Introduction

SRCAD is developed in CodeGear RAD Studio environment using the Pascal Delphi since the CodeGear RAD Studio environment helps the programmer to develop user-friendly interfaces in a fast way by supplying all necessary design tools such as forms, buttons, graphic plotters and list view. Therefore, SRCAD has a user friendly GUI that anyone can use it without knowing much about the software and hardware of the computer systems. In this manual, only the part according to Optimization is explained. The Analysis part is explained in [13].

4.9.2. Using the SRCAD Program

The main form of the SRCAD provides communication between the user and the program and can be seen in Figure 4.50 below. By using this form, user can reach all the facilities of the program. The Optimization part of the program is explained here.

4.9.2.1. Optimize Button

In the Main Form, firstly, the Solution Options and Physical Properties of the motor are written by the user. After then user can also choose B-H curve.

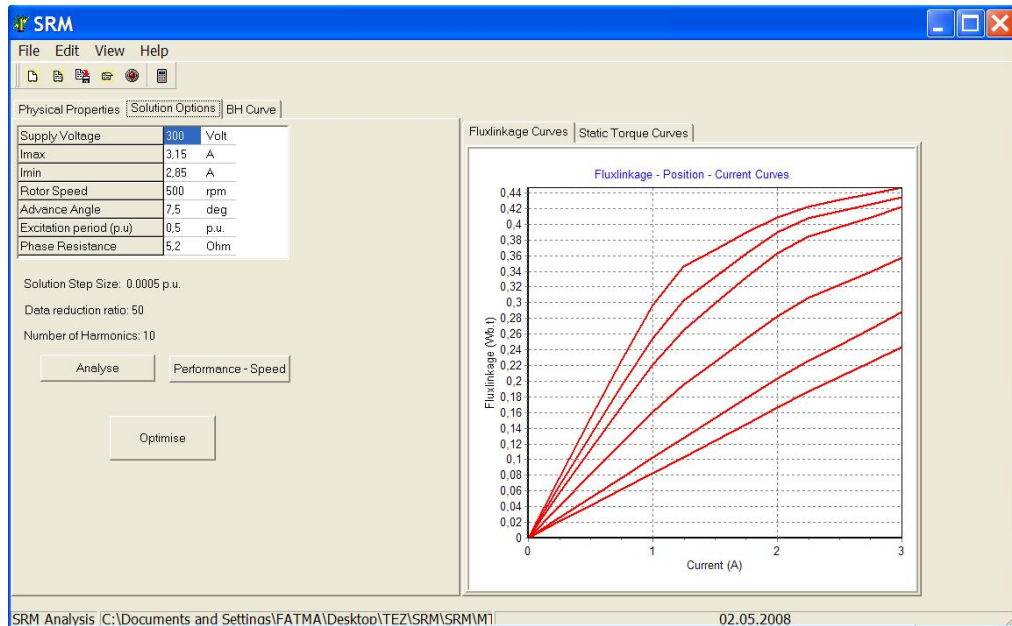



Figure 4.50 Main Form of SRCAD

For the optimization, Optimize button is used. If the user presses this button, Program writes all the user data a file, and then calls the optimization design program, modeFRONTIER.

4.9.2.2. Using modeFRONTIER Design

The user manual of the modeFRONTIER is given in reference [20].The project is ready for the optimization; user by only pressing the Run button can start the optimization. However, it is also possible for the user to change the design, initial parameters or optimization algorithm.

This  button is the Run button of the Program. Also, user can start the program form the main menu, Project/Run-Stop sub item.

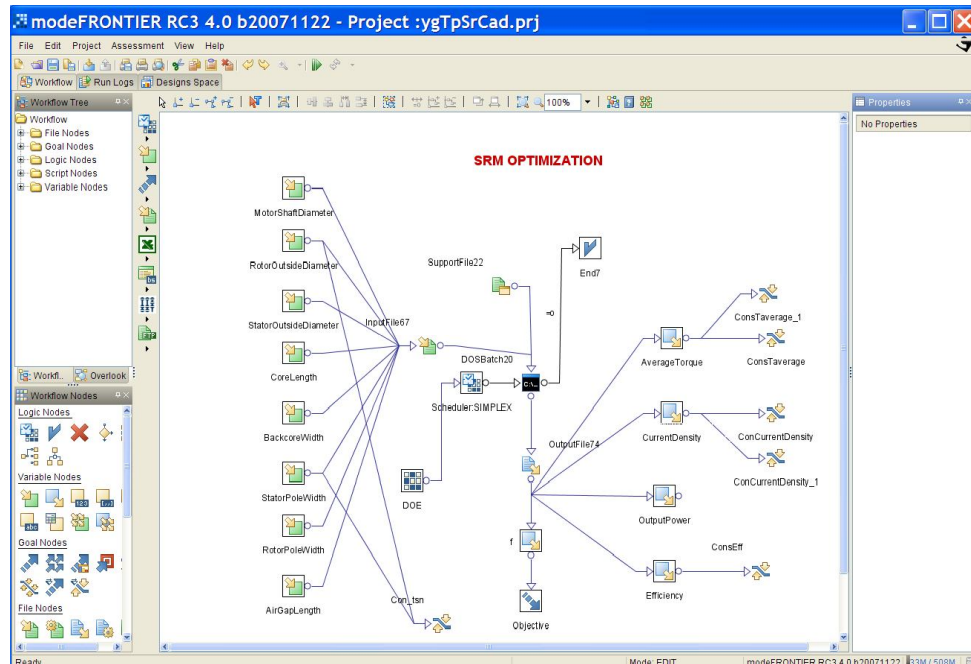


Figure 4.51 modeFRONTIER Design Form

After the ending of the optimization by using the “Designs Space” tab on the WorkFlow, user can easily observe the Outputs of the Optimization process, like in Figure 4.52.

Designs Table											
ID	RID	StatorOutsideDiam...	StatorPoleWidth	AverageTorque	CurrentDensity	Efficiency	OutputPower	f	Objective	%ConCurrentDensity	%ConCurrentDensity_1
147		1.3974E2	3.0955E0	1.6075E0	7.7375E0	8.0195E1	8.7535E1	37.31648626	3.7316E1	7.7375E0	7.7375E0
148		1.3883E2	3.5440E0	1.7374E0	7.6863E0	8.0304E1	9.4607E1	36.28729665	3.6287E1	7.6863E0	7.6863E0
149		1.4188E2	3.3926E0	1.7625E0	7.5047E0	8.0484E1	9.5973E1	39.10203929	3.9102E1	7.5047E0	7.5047E0
150		1.3813E2	3.4176E0	1.6741E0	7.7675E0	8.0143E1	9.1160E1	35.73849573	3.5738E1	7.7675E0	7.7675E0
151		1.4028E2	3.5710E0	1.7616E0	7.7164E0	8.0360E1	9.5928E1	37.66702780	3.7667E1	7.7164E0	7.7164E0
152		1.3887E2	3.3539E0	1.6805E0	7.6937E0	8.0208E1	9.1510E1	36.37973200	3.6380E1	7.6937E0	7.6937E0
153		1.3638E2	3.2286E0	1.6373E0	7.9661E0	8.0341E1	8.9158E1	34.61888670	3.4619E1	7.9661E0	7.9661E0
154		1.4031E2	3.4846E0	1.7246E0	7.6081E0	8.0182E1	9.3911E1	37.52747842	3.7527E1	7.6081E0	7.6081E0
155		1.3812E2	3.5515E0	1.7516E0	7.8423E0	8.0423E1	9.5384E1	36.05228994	3.6052E1	7.8423E0	7.8423E0
156		1.3858E2	3.4873E0	1.7303E0	7.7774E0	8.0376E1	9.4223E1	36.32074244	3.6321E1	7.7774E0	7.7774E0
157		1.4023E2	3.3211E0	1.6590E0	7.4507E0	7.9896E1	9.0340E1	37.12008214	3.7120E1	7.4507E0	7.4507E0
158		1.3856E2	3.4683E0	1.7255E0	7.8343E0	8.0415E1	9.3960E1	36.36675927	3.6367E1	7.8343E0	7.8343E0
159		1.3849E2	3.3261E0	1.6574E0	7.8570E0	8.0251E1	9.0252E1	36.29921568	3.6299E1	7.8570E0	7.8570E0
160		1.3925E2	3.4592E0	1.7242E0	7.6656E0	8.0271E1	9.3892E1	36.69148250	3.6691E1	7.6656E0	7.6656E0
161		1.3851E2	3.6279E0	1.7482E0	7.6564E0	8.0256E1	9.5198E1	35.77113940	3.5771E1	7.6564E0	7.6564E0
162		1.3920E2	3.3469E0	1.6870E0	7.7446E0	8.0277E1	9.1866E1	36.85691381	3.6857E1	7.7446E0	7.7446E0
163		1.3867E2	3.5805E0	1.7524E0	7.7573E0	8.0338E1	9.5424E1	36.38427693	3.6384E1	7.7573E0	7.7573E0
164		1.3912E2	3.3776E0	1.6908E0	7.6991E0	8.0272E1	9.2071E1	36.58866328	3.6589E1	7.6991E0	7.6991E0
165		1.3733E2	3.3791E0	1.6860E0	7.8719E0	8.0340E1	9.1808E1	35.29991651	3.5300E1	7.8719E0	7.8719E0
166		1.3584E2	3.3264E0	1.6642E0	8.0000E0	8.0403E1	9.0625E1	34.21599704	3.4216E1	8.0000E0	8.0000E0
167		1.3796E2	3.5248E0	1.7230E0	7.7591E0	8.0280E1	9.3825E1	35.56268495	3.5563E1	7.7591E0	7.7591E0
168		1.3759E2	3.4289E0	1.6876E0	7.8601E0	8.0306E1	9.1900E1	35.44692314	3.5447E1	7.8601E0	7.8601E0
169		1.3735E2	3.5234E0	1.7195E0	7.8662E0	8.0307E1	9.3634E1	35.26961322	3.5270E1	7.8662E0	7.8662E0
170		1.3646E2	3.5963E0	1.7340E0	7.9514E0	8.0364E1	9.4426E1	34.61969850	3.4620E1	7.9514E0	7.9514E0
171		1.3699E2	3.6077E0	1.7455E0	7.9513E0	8.0415E1	9.5048E1	35.03404688	3.5034E1	7.9513E0	7.9513E0
172		1.3840E2	3.4173E0	1.6973E0	7.7522E0	8.0275E1	9.2425E1	36.04060845	3.6041E1	7.7522E0	7.7522E0
173		1.3726E2	3.4805E0	1.6885E0	7.7805E0	8.0122E1	9.1946E1	34.96612664	3.4966E1	7.7805E0	7.7805E0
174		1.3691E2	3.4598E0	1.6774E0	7.8286E0	8.0173E1	9.1343E1	34.67764795	3.4678E1	7.8286E0	7.8286E0
175		1.3618E2	3.3821E0	1.6575E0	7.9429E0	8.0256E1	9.0260E1	34.31483031	3.4315E1	7.9429E0	7.9429E0
176		1.3817E2	3.6035E0	1.7165E0	7.7593E0	8.0265E1	9.3470E1	35.78821787	3.5788E1	7.7593E0	7.7593E0
177		1.3655E2	3.5303E0	1.7031E0	7.9025E0	8.0237E1	9.2740E1	34.49211166	3.4492E1	7.9025E0	7.9025E0

Figure 4.52 Design Table

In the figure above, the design table is shown. The parameters with the orange color are unfeasible, but with the white ones are feasible outputs of the optimization.

4.10. Conclusion of SRCAD

In this study, SRCAD that was developed for analysis of SRM is tested with using commercial motor. After the test results it is seen that the performance prediction lie in range in 10 % with the actual motors.

The construction of optimum motor geometry, which satisfies required performance specifications and constraints, is a very important subject. In this study, the optimization of SRM is formulated as a weight minimization problem. In addition, a project is designed in modeFRONTIER. Different optimization algorithms are applied to the problem and most suitable one is chosen as the optimization algorithm. Simplex and MOSA algorithms are found to be suitable algorithms to the design of SR motors.

At last, using SRCAD and modeFRONTIER makes faster the design of SR motor and performance of the existing motor can be calculated easily.

CHAPTER 5

CONCLUSION

At the end of this study , a software tool that can be used for the design of single phase induction machine, three-phase induction machine and switched reluctance motor is developed.

The ability of the software package is tested for each type of the electrical machines. It is seen that error of performance calculations of the motors are in the acceptable ranges.

For the three phase induction motor optimization module of TPCAD, Augmented Lagrangian Method with Davidon-Fletcher and Powell minimization technique is used as the optimization algorithm. Optimization module is also tested by designing a commercial motor, and it is seen that optimization algorithm is succesfull for the minimization of weight of the motor.

Also, for the optimization of the three-phase induction machines an optimization tool called modeFRONTIER, is used. This software permits testing of various optimization algorithms. Simplex and NLPQLP optimization methods are tested for the three-phase induction machine design problem. It is seen that NLPQLP algorithm is suitable for the problem and it leads to an optimum design. It is also observed that TPCAD optimization module is three times faster than the modeFRONTIER NLPQLP algorithm. This is possibly because 3 initial designs are simultaneously used in NLPQLP to reach a succesful design. TPCAD obtains the optimum design in 6 minutes, but modeFRONTIER leads to optimum design approximately in 20 minutes.

For single phase induction machine design, again modeFRONTIER is used. Several algorithms are tested. Amongst these NLPQLP algorithm is found to be the most suitable optimization algorithm for the design of single-phase induction machine. NLPQLP algorithm leads to an optimum design approximately in 10 minutes.

For SR motors, a software module is developed for the prediction of flux-linkage characteristics and static torque characteristics, using the "Normalized Permeance Data". This software package makes possible prediction of any SR motor performance given its physical dimensions and materials used. Based on this module an SR motor performance prediction software tool is developed. This package is tested for accuracy on a test motor. Predicted characteristics are compared with the measured ones. It is seen that percentage errors of the predicted and measured characteristics of the test motor lies in the acceptable accuracy range.

For the optimum design of SR motors, an optimization software based on the analysis algorithm is developed. The optimization module uses modeFRONTIER software for optimization. Different optimization algorithms such as SIMPLEX, MOSA and NLPQLP are applied to the problem. But it is observed that MOSA and SIMPLEX algorithms are the algorithms most suitable for the SR motor design optimization. It can be observed that the design optimization takes in the order of 10 minutes with the modeFRONTIER.

In short an optimum design suite is developed for which can handle various motor types. As far as the author knows a design software which can also handle optimization is not available. The software package leads to optimal designs in the order of 10-20 minutes and is very suitable for the design office. Further development is necessary for the user interface and of course an additional module which can handle PM motors is highly desirable.

modeFRONTIER software is used for some of the optimization experiments. It is found that this software is robust and almost certainly provides a solution for design optimization even for the highly nonlinear problem here

REFERENCES

- [1] Ertan, H.B., "Analytical Prediction of Torque and Inductance Characteristics of Identically Slotted Doubly Salient Reluctance Motors", Proc. IEE, Vol 4, 1986, pp.230-236
- [2] Ertan, H.B., "Prediction of Torque and Permeance Characteristics of Asymmetrical Tooth Pairs", Proc. Of the International Conference on Electrical Machines (ICEM), Munich, 1986, Vol 1, pp.97-100
- [3] Ertan, H.B., "Prediction of Torque and Inductance Displacement Characteristics of Asymmetrically Slotted VR Motors Using a Simplified Model for Numerical Solution", IEEE Trans. On Magnetics, Vol 35, No 5, September 1999, pp.4247-4258
- [4] Krishnan, R., "Switched Reluctance Motor Drives: Modeling, Simulation, Analysis, Design, and Applications", CRC Press, 2001
- [5] Mukherji, K.C., Neville, S., "Magnetic Permeance of Identical Doubly Slotting: Deductions from Analysis by F.W. Carter", Proc. IEE, 1971, Vol 118, no 9, pp.1257-1268
- [6] Corda, J., Stephenson, J.M., "Analytical Estimation of the Minimum and Maximum Inductances of a Doubly Salient Motor", Proceedings of the International Conference on Stepping Motors and Systems, 1979, pp.50-59
- [7] Tohumcu, M., "Optimum Design of Switched Reluctance Motors", Ph.D. Thesis, Middle East Technical University, 1985
- [8] Kızılkaya, M.Ö., "A Software for Dynamic Simulation of Switched Reluctance Motor", M.Sc. Thesis, Middle East Technical University, December 2001
- [9] Jones, A. L., "Permeance Model and Reluctance Force Between Toothed Structures", Proc. Of 5th symposium on incremental motion control system and devices, University of Illinois, May 1976, paper H

- [10]** Chai, H.D., "Permeance Model and Reluctance Force Between Toothed Structures", Proc. Of 2nd symposium on incremental motion control system and devices, University of Illinois, May 1973, paper K
- [11]** Hamarat, S., "Design Optimization of Three-Phase Induction Motors", M.Sc. Thesis, Middle East Technical University, January 1997
- [12]** Aftahi, M., "Computer Aided Design of Single - Phase Capacitor - Type Induction Motors", M.Sc. Thesis, Middle East Technical University, January 1993
- [13]** Yalçiner, L.B., "A Software for Analysis and Design Optimization of Switched Reluctance Motor", M.Sc. Thesis, Middle East Technical University, April 2004
- [14]** Şahin, F., "Optimum Geometry for Torque Ripple Minimization of Switched Reluctance Motors", M.Sc. Thesis, Middle East Technical University, 1996
- [15]** Miller, T.J.E., "Optimal Design of Switched Reluctance Motors", IEEE Trans. On Industrial Electronics, Vol. 49, No 1, February 2002
- [16]** Güler, İ., "Optimum Design of Single Phase Capacitor Start Motors", M.Sc. Thesis, Middle East Technical University, September 1990
- [17]** Kızılkaya, M.Ö., "The Method for Calculation of the Frictional Torque of a Motor and Application of the Method to the SR Motor", January 2001
- [18]** Sheth N.K., Rajagopal K.R., "Calculation of the Flux-Linkage Characteristics of a Switched Reluctance Motor by Flux Tube Method", Magnetics, IEEE Transactions, Volume 41, Issue 10, October 2005, pp 4069 – 4071
- [19]** <http://en.wikipedia.org/wiki/ModeFRONTIER>
- [20]** <http://www.kxcad.net/ESTECO/modeFRONTIER320/html/userman/>

- [21]** Schittkowski K., "NLPQLP: A Fortran Implementation of a Sequential Quadratic Programming Algorithm with Distributed and Non-Monotone Line Search - User's Guide, Version 2.0", December 2004
- [22]** <http://www.kxcad.net/ESTECO/modeFRONTIER320/html/userman/files/Schedulers/MOSA.pdf>

APPENDIX A

ANALYSIS DATA OF TEST MOTORS

Analysis Data of 90L6B

Title	90L6B
Date	23/05/1996

No	Name	Description	Value	Unit
1	M	Number of phases	3	---
2	V	Phase volts impressed	220	Volt
3	f	Frequency	50	Hz
4	p	Number of poles	6	---
5	C	Total number of series connected conductor	720	---
6	P _{rated}	Rated output power	1100	W
7	P _{u_pit}	Per unit pitch	1.0	---
8	Γ	Coil extension	0.105	m
9	F	Stacking factor	0.96	---
10	KΦ	Reactance voltage drop correction factor	0.94	---
11	Kr1	Total flux factor	0.96	---
12	Kp1	Primary flux factor	0.94	---
13	T	Final temperature	35	°C
14	S1	Number of stator slots	36	---
15	S2	Number of rotor slots	46	---
16	Khi	High frequency loss constant	6.4	---
17	Krad	Radiator factor	1.0	---
18	α	Skew angle	11	°
19	qm	Number of parallel paths in main winding	1	---
20	A1m	Cross-sectional area of conductor	4.9E-7	m ²
21	Rhom	Resistivity of main winding conductor	1.72E-8	Ω/m
22	Tmain	Temperature at which main winding resistivity is measured	25	°C
23	Rhor	Resistivity of end ring and motor bar	2.8E-8	Ω/m
24	Tbar	Temperature at which bar resistivity is masured	25	°C
25	Ab	Cross-sectional area of a single bar	3.32E-5	m ²
26	Ar1	Cross-sectional area of the end ring (fan side)	1.48E-4	m ²
27	Dr1	Mean diameter of the end ring (fan side)	6.65E-2	m
28	Ar2	Cross-sectional area of the end ring (other side)	1.48E-4	m ²
29	Dr2	Mean diameter of the end ring (other side)	6.65E-2	m

30	RhoFe	Iron density	7750	kg/m ³
31	ns	Synchronous speed	1000	rpm
32	n	Speed	942	rpm
33		Friction	3.0	---
34		Windage	3.0	---
35		Throw of coils	6	---
36	SHL	Shaft length	0.308	m
37	SA1	Stator slot area	5.73E-5	m ²
38	D	Stator outside diameter	0.138	m
39	D1	Stator bore diameter	0.091	m
40	L1	Stator stack length	0.115	m
41	t1	Stator tooth width	4.33E-3	m
42	d1	Stator tooth depth	9.0E-3	m
43	w10	Stator slot opening	3.6E-3	m
44	d11	Stator tip depth	0.0	m
45	d10	Stator mouth depth	0.0	m
46	d14	---	9.0E-3	m
47	w11	Minimum slot width	3.6E-3	m
48	w13	Maximum slot width	6.0E-3	m
49	SA2	Rotor slot area	3.322E-5	m ²
50	D2	Rotor outside diameter	0.0956	m
51	Db	Rotor shaft diameter	0.033	m
52	L2	Rotor stack length	0.115	m
53	t2	Rotor tooth width	2.82E-3	m
54	d20	Rotor mouth depth	0.37E-3	m
55	w21	Rotor slot top width	3.1E-3	m
56	w23	Rotor slot bottom width	1.4E-3	m
57	d2	Rotor tooth depth	1.537E-2	m
58	df	Distance between fins	2.0E-2	m
59	hf	Height of fin	2.0E-2	m
60	lb	Length of terminal	5.0E-2	m
61	wb	Width of terminal	4.0E-2	m
62	hb	Height of terminal	2.0E-2	m
63	N	Number of fins	38	---

Analysis Data of 100L4A

Title	100L4A
Date	08/05/1996

No	Name	Description	Value	Unit
1	M	Number of phases	3	---
2	V	Phase volts impressed	220	Volt
3	f	Frequency	50	Hz
4	p	Number of poles	4	---
5	C	Total number of series connected conductor	516	---
6	P _{rated}	Rated output power	2200	W
7	P _{u_pit}	Per unit pitch	1.0	---
8	Γ	Coil extension	0.158	m
9	F	Stacking factor	0.96	---
10	KΦ	Reactance voltage drop correction factor	0.94	---
11	Kr1	Total flux factor	0.96	---
12	Kp1	Primary flux factor	0.96	---
13	T	Final temperature	30	°C
14	S1	Number of stator slots	36	---
15	S2	Number of rotor slots	28	---
16	Khi	High frequency loss constant	2.25	---
17	Krad	Radiator factor	3.0	---
18	α	Skew angle	11	°
19	qm	Number of parallel paths in main winding	1	---
20	A1m	Cross-sectional area of conductor	8.0E-7	m ²
21	Rhom	Resistivity of main winding conductor	1.72E-8	Ω/m
22	Tmain	Temperature at which main winding resistivity is measured	25	°C
23	Rhor	Resistivity of end ring and motor bar	2.8E-8	Ω/m
24	Tbar	Temperature at which bar resistivity is masured	25	°C
25	Ab	Cross-sectional area of a single bar	6.2E-5	m ²
26	Ar1	Cross-sectional area of the end ring (fan side)	1.3E-4	m ²
27	Dr1	Mean diameter of the end ring (fan side)	8.1E-2	m
28	Ar2	Cross-sectional area of the end ring (other side)	1.3E-4	m ²
29	Dr2	Mean diameter of the end ring (other side)	8.1E-2	m
30	RhoFe	Iron density	7750	kg/m ³
31	ns	Synchronous speed	1500	rpm
32	n	Speed	1410	rpm
33		Friction	10.0	---
34		Windage	10.0	---
35		Throw of coils	9	---
36	SHL	Shaft length	0.358	m
37	SA1	Stator slot area	7.43E-5	m ²

38	D	Stator outside diameter	0.155	m
39	D1	Stator bore diameter	0.104	m
40	L1	Stator stack length	0.100	m
41	t1	Stator tooth width	4.5E-3	m
42	d1	Stator tooth depth	1.06E-2	m
43	w10	Stator slot opening	3.8E-3	m
44	d11	Stator tip depth	0.0	m
45	d10	Stator mouth depth	0.0	m
46	d14	---	1.06E-3	m
47	w11	Minimum slot width	3.8E-3	m
48	w13	Maximum slot width	6.8E-3	m
49	SA2	Rotor slot area	6.17E-5	m ²
50	D2	Rotor outside diameter	0.1035	m
51	Db	Rotor shaft diameter	0.038	m
52	L2	Rotor stack length	0.1	m
53	t2	Rotor tooth width	5.72E-3	m
54	d20	Rotor mouth depth	0.7E-4	m
55	w21	Rotor slot top width	5.2E-3	m
56	w23	Rotor slot bottom width	2.2E-3	m
57	d2	Rotor tooth depth	1.74E-2	m
58	df	Distance between fins	2.0E-2	m
59	hf	Height of fin	2.0E-2	m
60	lb	Length of terminal	5.0E-2	m
61	wb	Width of terminal	4.0E-2	m
62	hb	Height of terminal	2.0E-2	m
63	N	Number of fins	38	---

Analysis Data of 112M4B

Title	112M4B
Date	07/05/1996

No	Name	Description	Value	Unit
1	M	Number of phases	3	---
2	V	Phase volts impressed	380	Volt
3	f	Frequency	50	Hz
4	p	Number of poles	4	---
5	C	Total number of series connected conductor	528	---
6	P _{rated}	Rated output power	40000	W
7	P _{u_pit}	Per unit pitch	1.0	---
8	Γ	Coil extension	0.16	m
9	F	Stacking factor	0.96	---
10	KΦ	Reactance voltage drop correction factor	0.94	---
11	Kr1	Total flux factor	0.96	---
12	Kp1	Primary flux factor	0.96	---
13	T	Final temperature	95	°C
14	S1	Number of stator slots	36	---
15	S2	Number of rotor slots	28	---
16	Khi	High frequency loss constant	1.3	---
17	Krad	Radiator factor	3.0	---
18	α	Skew angle	10.4	°
19	qm	Number of parallel paths in main winding	1	---
20	A1m	Cross-sectional area of conductor	6.6E-7	m ²
21	Rhom	Resistivity of main winding conductor	1.72E-8	Ω/m
22	Tmain	Temperature at which main winding resistivity is measured	25	°C
23	Rhor	Resistivity of end ring and motor bar	2.8E-8	Ω/m
24	Tbar	Temperature at which bar resistivity is masured	25	°C
25	Ab	Cross-sectional area of a single bar	6.9E-5	m ²
26	Ar1	Cross-sectional area of the end ring (fan side)	2.2E-4	m ²
27	Dr1	Mean diameter of the end ring (fan side)	8.9E-2	m
28	Ar2	Cross-sectional area of the end ring (other side)	2.2E-4	m ²
29	Dr2	Mean diameter of the end ring (other side)	8.9E-2	m
30	RhoFe	Iron density	7750	kg/m ³
31	ns	Synchronous speed	1500	rpm
32	n	Speed	1416.4	rpm
33		Friction	15.0	---
34		Windage	10.0	---
35		Throw of coils	9	---
36	SHL	Shaft length	0.385	m
37	SA1	Stator slot area	7.21E-5	m ²
38	D	Stator outside diameter	0.170	m

39	D1	Stator bore diameter	0.117	m
40	L1	Stator stack length	0.145	m
41	t1	Stator tooth width	4.69E-3	m
42	d1	Stator tooth depth	9.4E-3	m
43	w10	Stator slot opening	3.8E-3	m
44	d11	Stator tip depth	0.0	m
45	d10	Stator mouth depth	0.0	m
46	d14	---	9.4E-3	m
47	w11	Minimum slot width	3.8E-3	m
48	w13	Maximum slot width	7.2E-3	m
49	SA2	Rotor slot area	6.89E-5	m ²
50	D2	Rotor outside diameter	0.1165	m
51	Db	Rotor shaft diameter	0.038	m
52	L2	Rotor stack length	0.145	m
53	t2	Rotor tooth width	6.85E-3	m
54	d20	Rotor mouth depth	0.35E-3	m
55	w21	Rotor slot top width	5.5E-3	m
56	w23	Rotor slot bottom width	2.3E-3	m
57	d2	Rotor tooth depth	1.83E-2	m
58	df	Distance between fins	2.0E-2	m
59	hf	Height of fin	2.0E-2	m
60	lb	Length of terminal	5.0E-2	m
61	wb	Width of terminal	4.0E-2	m
62	hb	Height of terminal	2.0E-2	m
63	N	Number of fins	38	---

APPENDIX B

OPTIMIZATION ALGORITHMS IN modeFRONTIER

SIMPLEX METHOD

Simplex Definition

From a geometrical point of view, a simplex is a polyhedron containing $N + 1$ points in a N dimensional space, thus in two dimensions is a triangle, in three dimension is a tetrahedron, and so forth. When the points are equidistant, the simplex is said to be regular. The Simplex method compares the values of the objective function at the $N + 1$ vertices and then moves this polyhedron gradually towards the optimum point during the iterative process. For each iteration of the algorithm, it attempts to replace simplex vertices that yield high function values with vertices whose new function values are lower. The aim of this algorithm is to move the simplex, by replacing vertices, into the neighborhood of a minimizer. The movement of the simplex is given by three operations: Reflection, Expansion and Contraction.

Reflection

In the reflection movement, the worst vertex, corresponding to the worst value of the objective function among all the vertices, is reflected in the opposite face to obtain a new value. Since the direction of this movement is always away from the worst result, the reflection moves towards a favorable

direction. If the objective function does not have steep valleys, repetitive reflections will lead to the function optimum point.

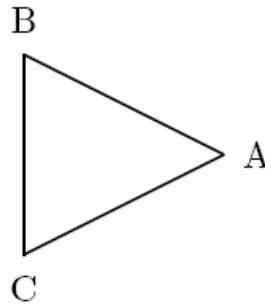


Figure B.1 Original Simplex (A,B,C) in a 2-dimension

Suppose that we have a 2-dimension simplex as in Figure B.1 and suppose that the vertex (A) is the highest and (C) the lowest. We can expect the point (D) obtained by reflecting the point (A) in the opposite face to have the smallest value. If this is the case, we can construct a new simplex by rejecting the point (A) from the original simplex and including the new point (D). In the Figure B.2 it is possible to see how the vertex (A) is reflected and moved to the opposite side of the original simplex. Mathematically the reflected point can be written as:

$$D = (1 + \alpha)H - \alpha A$$

where $\alpha > 0$ is a reflection coefficient, A is the vertex corresponding to the maximum function value and H is the centroid of all the points except A (in our example $H = (B + C)/2$).

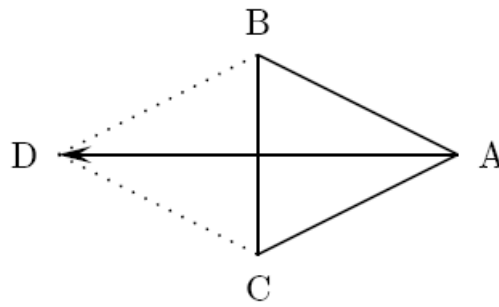


Figure B.2 Reflected Simplex (A,B,C) with the Reflected Point (D)

Expansion

When the reflection process gives a new minimum, we can generally expect to decrease the function value further by moving along the same direction expanding the simplex.

Suppose now that our 2-dimension simplex as in Figure B.2 has its minimum in the vertex (D). We can expect the point (E) obtained by expanding the simplex in the same direction gives a better value. If this is the case, we can construct a new simplex by rejecting the point (D) from the previous simplex and including the new point (E).

In the Figure B.3 it is possible to see how the vertex (E) is created by stretching the previous simplex.

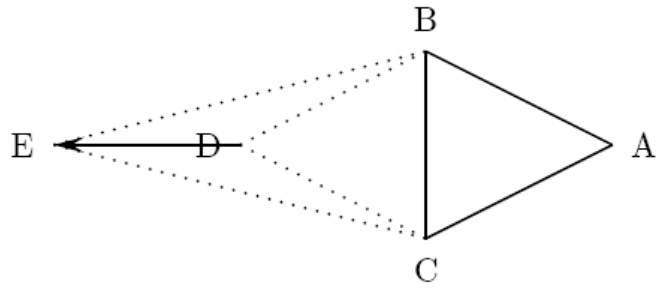


Figure B.3 Simplex Expansion (A,B,D) with the New Point (E)

Mathematically the expanded point can be written as:

$$E = \beta D + (1 - \beta)H$$

where $\beta > 1$ is a expansion coefficient, D is the vertex created by the reflection and H is the centroid of all the points except D (in our example $H = (B + C)/2$).

Contraction

When the reflection process gives a new point that is worse than all the other vertices except the worst, the simplex is contracted by exchanging the new point with the worst point.

If this is the case, we can construct a new simplex by rejecting the point (D) from the previous simplex and including the new point (E). In the Figure B.4 it is possible to see how the vertex (E) is created by shrinking the reflected simplex.

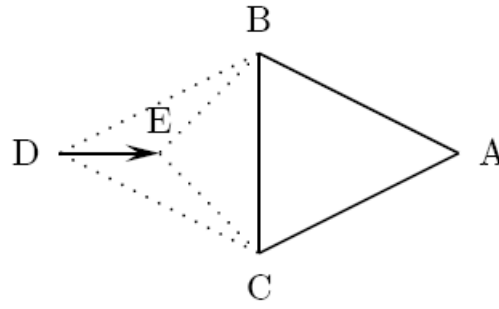


Figure B.4 Contracted Simplex (A,B,C) with the Contracted Point (E).

Mathematically the new point can be written as:

$$E = \beta D + (1 - \beta)H$$

where $\beta \in [0; 1]$ is a contraction coefficient, D is the vertex created by the reflection and H is the centroid of all the points except D (in our example $H = (B + C)/2$).

Constraint Violation Penalty

The original Nelder and Mead downhill method has been changed to consider not only the minimization of a function, but also the enforcement of feasibility. During the last few years, several methods have been proposed for handling constraints. The methods based on penalty functions are based on the concept that the objective function $f(x)$ is increased according to the intensity of the constraint violation. This is done to ensure that the vertices do not deviate too much from the constraints. Mathematically the penalty function can be written as:

$$P(x; \rho) = f(x) + \rho * c(x)$$

where ρ is the penalty parameter and $c(x)$ is the sum of all constraint violations. A drawback of this methodology is that the penalty parameter can be problem dependent and might draw the simplex to premature convergence. A way to overcome this problem is to adapt the penalty parameter to the present maximum and minimum function values.

$$\rho = f_{\max}(x) - f_{\min}(x)$$

Each function $c_i(x)$ can be transformed into a fuzzy introducing a tolerance parameter k in the i^{th} constraint definition. If $c_i(x) < 0$ then x represents a feasible value for the i^{th} constraint, if $c_i(x) > k$ then x is completely unfeasible. For $0 < c_i(x) < k$ the distinction between feasible and unfeasible values is fuzzy as shown in figure B.5.

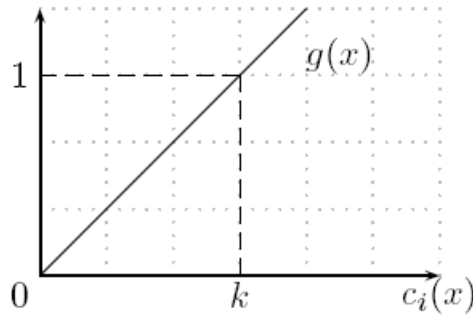


Figure B.5 The Tolerance Parameter k Transforms the Constraint Penalization into a Fuzzy Function $g(x)$

Termination Criteria

The termination criteria is delicate in any multidimensional algorithm. Since the simplex algorithm does not use derivatives, no termination criterion is available based on the gradient of the objective function. Hence, algorithm stops when it cannot find solution with improvements higher than some tolerance.