

# **TRACTION MACHINE WINDING AND MAGNETDESIGN FOR ELECTRIC VEHICLES**

A thesis submitted to the University of Manchester for the degree of  
Master of Philosophy  
In the faculty of Engineering and Physical Sciences

**2016**

**By  
Xin Niu**

**School of Electrical and Electronic Engineering**

# CONTENTS

LIST OF TABLES	5
LIST OF FIGURES	6
TABLE OF ABBREVIATION	9
NOMENCLATURE	10
ABSTRACT	11
DECLARATION	12
COPYRIGHT STATEMENT	13
ACKNOWLEDGEMENTS	14

## CHAPTER 1

### BACKGROUND OF PERMANENT MAGNET (PM) MACHINE DESIGN

1.1	Introduction	15
1.2	PM machine Design	16
1.3	Aims and objectives	18

## CHAPTER 2

### WINDING DESIGN

2.1	Literature Review on Multiphase Design	22
2.2	Machine Model Analysis	24
2.2.1	Terms in model analysis	24
2.2.2	Machine Model Analysis	25
2.2.3	Searching Coil	26
2.2.3.1	Model Development	26
2.2.3.2	Model Analysis	28
2.2.3.3	Data Processing	29
2.2.3.4	Estimated Result	31
2.2.3.5	Search Coil Test	33
2.2.3.6	Search Coil Conclusion	34
2.2.4	Concentrated Winding Model	35
2.2.5	Distributed Winding Models	36
2.2.6	Model Analysis and Outputs Comparison	39
2.2.6.1	Comparison of Flux Linking and Back EMF Waveform	40

2.2.6.2 Torque Analysis	45
2.3 Summary	49
CHAPTER 3	
ANALYSIS OF MACHINE MODEL WITH BREAD LOAF MAGNET DESIGN	
3.1 Introduction	50
3.2 Different rotor Topologies	50
3.2.1 Mechanical Design of Magnet	52
3.2.2 Model Defining and Software Program	53
3.3 Electromagnetic Capability	56
3.3.1 Cogging Torque	56
3.3.2 Open Circuit EMF	63
3.3.3 Electromagnetic Torque Capability	71
3.3.4 Demagnetization Limits	78
3.4 Excitation Torque Analysis	90
3.4.1 Excitation Torque with Sinusoidal Current	90
3.4.2 Results for Excitation Torque	92
3.5 Summary	100
CHAPTER 4	
IRON LOSS ANALYSIS	
4.1 Literature Review	101
4.1.1 Equations for Alternating Loss.	111
4.1.2 Equations for Rotational Loss.	113
4.2 Implementation	114
4.2.1 Equation and parameter	114
4.2.2 Simulation control	115
4.2.3 Solution analysis	117
4.2.4 Data manipulation	119
4.3 Machine iron loss analysis	121
4.4 Iron loss Calculation	125
4.5 Summary	127

## CHAPTER 5

### STUDY INTO THE CHOICE OF PHASE NUMBER FOR A MULTIPHASE, CONCENTRATED WINDING MACHINE

5.1	Introduction	128
5.2	The Influence of Phase Number for Sinusoidal Drive Systems	129
5.3	Analysis of Test Machine for 3 and 9-Phase Connection	132
5.3.1	Back-EMF	132
5.3.2	Electromagnetic torque.	133
5.4	Other Phase Connections	135
5.4.1	Back-EMF and Torque Performance	135
5.4.2	Iron-Loss Considerations	135
5.5	Summary	142

## CHAPTER 6

### CONCLUSIONS AND RECOMENDATIONS FOR FURTHER WORK

6.1	Conclusion	143
6.2	Recommendation for further work.	145

REFERENCES	145
------------	-----

APPENDIX I	Software coding	156
APPENDIX II	Code description and demagnetization check	201
APPENDIX II	Flux Results	271

Word count of main text: 26,167



## **LIST OF TABLES**

Table 1.1 Data for different types of machines.	16
Table 2.1 Table of reviewed reference for multiphase design.	21-23
Table 2.2 Calculation for the phase difference.	34
Table 2.3 Table of slot area and current injected	46
Table 3.1 Magnet size for different magnet designs	52
Table 3.2 Machine model dimensions.	57
Table 3.3 Part of result recorded in cogging torque test.	60
Table 3.4 Universal array for storing peak torque data.	74
Table 3.5 Universal array for storing peak torque location.	75
Table 3.6 The case table for calculating angle of flux density.	83
Table 3.7 Demagnetization table 1.	87
Table 3.8 Demagnetization table 2.	88
Table 3.9 Demagnetization table 3.	88
Table 3.10 Demagnetization table 4.	88
Table 3.11 Demagnetization table 5.	89
Table 3.12 Demagnetization table 6.	89
Table 3.13 Demagnetization table 7.	89
Table 3.14 Demagnetization table 8.	90
Table 4.1 Table of reference reviewed on Iron loss calculation, Part 1.	109
Table 4.2 Table of reference reviewed on Iron loss calculation, Part 2.	110
Table 4.3 Alternating loss equation, part 1.	111
Table 4.4 Alternating loss equation, part 2.	112
Table 4.5 Rotating loss equation.	113
Table 4.6 Implemented alternating loss equation.	114
Table 4.7 Implemented rotating loss equation	114
Table 4.8 Diagram of variable array.	116

## LIST OF FIGURES

Fig. 2.1	An example of machine model in FEMM.	25
Fig. 2.2	MATLAB programme data processing diagram.	26
Fig. 2.3	FEMM model of the generator, with 16 poles, 18 slots.	27
Fig. 2.4	An example of flux density plot in FEMM.	28
Fig. 2.5	Plot of simulation result.	30
Fig. 2.6	Estimated back EMF waveform for search coil.	32
Fig. 2.7	Phase shift of back EMF of search coils.	32
Fig. 2.8	Search coil test setup.	33
Fig. 2.9	Back EMF waveform.	34
Fig. 2.10	3-phase concentrated winding.	35
Fig. 2.11	FEMM model of 32 poles short-pitched PM machine.	36
Fig. 2.12	Stator winding.	37
Fig. 2.13	Basic tooth and slot block.	37
Fig. 2.14	FEMM models.	38
Fig. 2.15	Balanced star connection.	39
Fig. 2.16	Plot of flux linking the phase windings.	41
Fig. 2.17	Plot of flux linking the phase windings.	42
Fig. 2.18	Plot of flux linking the phase windings.	43
Fig. 2.19	Rotating process for concentrated winding.	45
Fig. 2.20	Rotating process for distributed winding.	45
Fig. 2.21	Electromagnetic torque when DC current injected of 3-phase PM machine with concentrated winding.	47
Fig. 2.22	Electromagnetic torque when DC current injected of 3-phase PM machine with distributed winding, 1 slot per pole per phase.	47
Fig. 2.23	Electromagnetic torque when DC current injected of 3-phase machine with distributed winding, 2 slots per pole per phase.	48
Fig. 3.1	Rotor magnet designs.	51
Fig. 3.2	Iterative process of coding function verification	53
Fig. 3.3	An example of line selecting in result file.	55
Fig. 3.4	Brief function of a MATLAB code.	56
Fig. 3.5	Winding schematic for 3-phase machine model.	58
Fig. 3.6	Plots for cogging torque.	61-62

Fig. 3.7 Plot of flux linking the coil in one tooth for different magnet designs	64
Fig. 3.8 Plot of phase EMF for different magnet designs.	66
Fig. 3.9 Harmonic spectrum and its parameter table for different magnet design.	67
Fig. 3.10 Harmonic spectrum and its parameter table for different magnet design.	68
Fig. 3.11 Harmonic spectrum and its parameter table for different magnet design.	69
Fig. 3.12 Harmonic spectrum and its parameter table for different magnet design.	70
Fig. 3.13 Harmonic spectrum and its parameter table for different magnet design.	71
Fig. 3.14 Peak torque analyzing program.	73
Fig. 3.15 Winding schematic for 9-phase machine model.	74
Fig. 3.16 Torque versus current characteristic for different magnet designs.	77
Fig. 3.17 Torque versus current characteristic for different magnet design.	76
Fig. 3.18 The recoil process of a magnet after demagnetization.	79
Fig. 3.19 Flow chart for result file analyzing program .	81
Fig. 3.20 Implemented calculations in Cartesian coordinate system.	83
Fig. 3.21 FEA field solutions of bread loaf design.	85
Fig. 3.22 Flow chart and example of code.	92
Fig. 3.23 Total torque versus electrical angle.	93
Fig. 3.24 Excitation torque versus electrical angle.	93
Fig. 3.25 Electro-magnetic torque magnitude spectrum.	94
Fig. 3.26 Unit phase current and emf for 3-phase model.	96
Fig. 3.27 Unit phase current and emf for 9-phase model.	96
Fig. 3.28 Unit torque comparison between 3-phase and 9-phase model.	97
Fig. 3.29 Magnitude spectrum of estimated unit power or unit torque for two machine models.	98
Fig. 3.30 Flow chart of automatic analyzing program.	99
Fig. 4.1 Diagram of particle energy state changing in a magnetic field.	105
Fig. 4.2 Hysteresis Loop of ferromagnetic material.	105
Fig. 4.3 Diagram of EMF introduced by field	106
Fig. 4.4 Diagrams of eddy current: (a) in a block and (b) in laminated material.	107
Fig. 4.5 Scenario for simulation program.	117
Fig. 4.6 Flow chart for example simulation control program.	118
Fig. 4.7 Concept for solution analysis program.	119
Fig. 4.8 Format of solution files when opened as text file.	120
Fig. 4.9 Brief flow chart for solution analysis program	120

Fig. 4.10	Concept for data manipulation program.	121
Fig. 4.11	Flow chart for an example data manipulation program.	122
Fig. 4.12	Diagram of testing blocks.	124
Fig. 4.13	Examples of flux plot for detecting rotating component.	125
Fig. 4.14	Flux plot and FFT table for different testing block 1.	125
Fig. 4.15	Flux plot and FFT table for different testing block 3.	126
Fig. 4.16	Linear fitting for P/f data for Core loss coefficient obtain.	127
Fig. 4.17	Calculated iron loss with different approaches compared with experimental data.	128
Fig. 5.1	Electro-magnetic field distribution of test machine 3-phase configuration.	133
Fig. 5.2	Electro-magnetic field distribution of test machine 9-phase configuration.	133
Fig. 5.3	Normalised induced back-EMF per phase for the 3-phase (3-teeth) and 9-phase (1 tooth coil) test machine.	134
Fig. 5.4	Normalised torque versus electric angle for the 3-phase and 9-phase test machine with sinusoidal phase current.	134
Fig. 5.5	Models and full-load field solutions.	137
Fig. 5.6	Models and full-load field solutions.	138
Fig. 5.7	Back-EMF and torque results.	139
Fig. 5.8	Back-EMF and torque results.	140
Fig. 5.9	Average torque versus phase number.	141
Fig. 5.10	Iron loss versus speed results.	141

## TABLE OF ABBREVIATIONS

Abbreviations	Definition
PM	Permanent Magnet
EV	Electric Vehicle
HEV	Hybrid Electric Vehicle
DC	Direct Current
AC	Alternating Current
IM	Induction Machine
PM	Permanent Magnet
SR	Switch Reluctance
BLPM	Brushless Permanent Magnet
FEA	Finite Element Analysis
MMF	Magnetomotive Force
TM	Traction Machine
ICE	Internal Combustion Engine
SC	Super Capacitors

## NOMENCLATURE

Symbol	Definition	Unit
A	Area	$m^2$
B	Magnetic flux density	Tesla
e	EMF	V
f	Frequency	Hz
H	Magnetic flux density	$A \cdot m^{-1}$
I	Current	A
J	Current density	$A/m^2$
M, N	Number of slots	--
P	Pole pair or power	-- or Watt
R	Resistance	$\Omega$
t	Time	s
$\alpha, \beta, \theta$	Angle or phase angle	Degree
$\varphi$	Magnetic flux linking	Wb
$\omega$	Angular speed	Degree/s
$\sigma$	Resistivity or conductivity	$\Omega^{-1} \cdot m^{-1}$ or $S / m$

## **ABSTRACT**

Work had been established for traction machine design aspects in this research. The effect of multiphase design for Permanent Magnet (PM) machine was investigated. The electromagnetic characteristics of both 3-phase and 9-phase machine, along with different magnet designs, were simulated and analyzed by using the program developed during the process. The software used were FEMM and MATLAB. The iron loss for different designs was established, based on the analytical flux density obtained by 2-D stepping FEA method. The harmonic of flux waveform and rotating field were also considered for difference areas in the machine models. The prediction was compared with experimental data collected in open circuit. The simulation result shown that there was a minimum 4% torque gain and noticeable less torque ripples for 9-phase machine, comparing with 3-phase one, with the same excitation phase current. The embedded magnet rotor design was suggested to monitor the demagnetization of each magnet closely, since some area of the magnet could be demagnetized even when the working point of magnet was well distance away from the nonlinear region of its characteristic. There were about 6% less iron loss was produced in 9-phase model than 3-phase model. The implemented method for calculating iron loss was more accurate within 3500 rpm rotor speed comparing with other approaches.

## **DECLARATION**

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute or learning.



## COPYRIGHT STATEMENT

- i. The author of this thesis (including any appendices and/or schedules to this thesis) owns any copyright in it (the “copyright”) and he has given The University of Manchester the right to use such Copyright for any administrative, promotional, educational and/or teaching purposes.
- ii. Copies of this thesis, either in full or in extracts, may be made **only** in accordance with the regulations of the John Rylands University Library of Manchester. Details of these regulations may be obtained from the Librarian. This page must form part of any such copies made.
- iii. The ownership of any patents, designs, trade marks and any and all other intellectual property rights except for the Copyright (the “Intellectual Property Rights”) and any reproductions of copyright works, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property Rights and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of relevant Intellectual Property Rights and/or Reproductions.
- iv. Further information on the conditions under which disclosure, publication and exploitation of this thesis, the Copyright and any Intellectual Property Rights and/or Reproductions described in it may take place is available from the Head of School or Electrical and Electronic Engineering (or the Vice-President) and the Dean of the Faculty of Life Sciences, for Faculty of Life Sciences’ candidates.

## **ACKNOWLEDGEMENTS**

After few years long run, the commitment for myself has been fulfilled. Sincerely thanks to my supervisors Dr. Nigel Schofield who support and encourage me through these years, especially after he changed position in work. Without his help, I definitely will not reach this far. I need to apologize for all the troubles and problems I brought, for which he spent so much time to help me with patience. Also thanks to another supervisor of mine, Dr. Roger Shuttleworth, for his support and understanding while I had personally issues.

Can not forgot the relaxing moment while taking break at the tea corner with my friends who taught me so much which I can not learn from textbooks, including English pronunciation. Tea drinkers and coffee drinkers but no one tried the fridge juice. Good time always passes so fast and now we all part away for better future. Wish you guys all the best and let us have a drink together next time when we met. Talk and relax like what we used to do at the tea corner.

# **CHAPTER 1**

## **BACKGROUND OF PERMANENT MAGNET (PM) MACHINE DESIGN**

### **1.1 Introduction**

Since the first industrial revolution, the industrialisation has been accomplished in most of the countries and it brought us into modernization with the most convenient life the ancient people could never imagine [1]. But with rising environment issue recently, i.e. globe warming and industry pollution and over-exploitation on natural resources, the sustainable development is required for the good of all the people living on this planet. As the agreement made on reduction of carbon dioxide (CO<sub>2</sub>) emission, the transport sector is stressed on. Currently, most of the transportations are still relies on fossil fuels and combustion process, which conflict with the environment in quite few aspects: fossil fuel is un-renewable resource, small scale combustion process has low efficient in power conversion because of the heat losses and the exhaust causes air pollution and also contains CO<sub>2</sub> [2, 3]. In UK, about 30% of the all energy demand was consumed by road transportation. When the time stepped into 21<sup>st</sup> century, more and more electricity is produced by large scale power station and renewable energy plants, i.e. wind farm, solar power station and hydro power station [4]. The renewable electricity and the development of battery technology in recent twenty years proved the possibility of replacing the tradition transportation with electricity powered transportation: Electric Vehicle (EV), to achieve zero emission and environment friendly. The electric drive system has two main advantages comparing to fuel-consumed engine: three times more efficient and the renewable energy source.

There two solutions of electric powered vehicles: battery powered EV or Hybrid Electric Vehicle (HEV). For a HEV, as a compensate design for the limit of battery energy

storability, the combustion engine is still used to generate the energy required to drive the vehicle [3]. Although the fuel economy of HEV is improved dramatically, the battery powered EV is the irreplaceable solution for zero emission and maximized efficiency. The power ratings of EVs has been from few tens watts to few hundred kilowatts since 2009 with 40 mile range [2]. Now days, as the battery storability increased, the range of EVs has been improved to 209 km or 130 miles [5].

There are four types of machines possible for traction: Direct Current (DC) machine, Alternating Current (AC) Induction Machine (IM), Permanent Magnet (PM) machine and Switch Reluctance (SR) machine. Table 1.1 [6] shows the comparison of different types of machines. Due to the limitation of the battery and weight of the vehicle, PM machine is chosen for EVs for its high energy efficiency, high torque, high power density, low weight and the capability to regenerating electricity back to the battery when braking [7-11].

Machine type	Torque per unit stator volume (kNm/m <sup>3</sup> )	Weight (%)	Overall Efficiency	Cost (%)
DC	3.5	100	78%	100
AC Induction	7.4	50	84%	100
PM	23.7	25	90%	150
SRM	6.4	40	85%	150

Table 1.1 Data for different types of machines [6, 11, 118].

## 1.2 PM machine for traction applications

There are many papers writing about the PM machine design for traction applications. The improvement over the conventional Brushless PM machine (BLPM) including two parts: stator winding slot and rotor magnet. For the stator design, the benefits of modular 3-phase BLPM, which has special designed stator tooth and winding and as the result, lower in pole number and the fractional number of slots per pole leading to lower cogging torque is discussed in reference [10]. For the rotor design, the permanent magnet size is considered theoretically for traction motor in reference [4] and Interior Permanent Magnet (IPM) technology [12], enclosing the magnets in the rotor rather than mounting on the surface, provides more mechanical robustness and extended flux wakening region. There

were concentrated and distributed winding designs for stator. Detail and comparison of these two designs would be introduced in Chapter 2.

Finite Element Analysis (FEA) method has been extensively used [7, 13] for preliminary design. The torque output of the machine related to the size, electrical loading and average air-gap flux density [6]. There are also analytical design methods and harmonic analysis methods [6].

As the power converters developing and some of the specialised application requirements cannot be met by conventional 3-phase machine, there is more and more considerations about multiphase PM machine, because there are several advantages over 3-phase PM machine: lowered the phase current and DC-link current harmonic which will reduce via heat and improve the thermal environment, higher torque-ampere ratio for same size of machine and increase in the reliability [11, 14].

In reference [11], the existing techniques and major advantages has been reviewed and summarised of multiphase induction and synchronous machine, used for variable speed application. The mechanical difficult in stator winding design for multiphase machine has also been pointed out: with increasing phase number, the more slots required to realize a sinusoidal magnetomotive force (MMF) distribution for machine driving mechanism.

The torque pulsation problem for a multiphase PM machine is studied in reference [15, 16] and in which also state a fractional-slot design will reduce the torque pulsation in 5-phase, 7-phase and 9-phase IPM machines based on the results of FEA. Reference [8] represents a practical example of 5-five phase radial flux motor for a direct-driven system, which can achieve high torque at low speed.

For designing a BLPM machine for EV, the reliability was most import for safety reason and the multiphase design could be an option, and also the other advantages would benefit the overall driving system for EV. Analyzing on different PM machine using FEA method would be the first stage of the design work. A temperature monitoring system is also required due to the thermal effect to the battery, permanent magnet, winding and control power electronics in the traction driving system.

### 1.3 Aims and objectives

The aim of this thesis is to determine how the machine winding configuration, particularly the number of phases, and the magnet design can be used to improve the performance of surface permanent magnet electrical machines for road vehicles.

1. To compare distributed and concentrated windings using finite element analysis, in order to understand the advantages of concentrated windings for traction applications.
2. To evaluate five common magnet shapes and identify the performance of each in terms of back emf, cogging torque, output torque and risk of demagnetisation.
3. To extend the FE modelling software to account for rotational loss in order to improve the accuracy of iron loss calculations.
4. To compare designs with different numbers of phases, ranging from 3 and 18, in terms of torque and loss values.

Chapter 2 will discuss the use of phase number as a design consideration and review recent research in the field of multiphase machines. It will discuss the analysis methods and techniques relevant to the design of brushless permanent magnet machines in particular a public domain finite element software FEMM.

Chapter 3 discusses the code developed to automate the characterisation of a surface magnet brushless permanent magnet machine in terms of back-EMF, excitation torque, cogging torque and magnet demagnetisation. The developed code uses the MATLAB platform for implementation and also multiply field solutions and subsequent post-processing of a machine design.

Many finite element solvers implement some form of iron-loss calculation, however, there is no such feature in FEMM. Further, while other solvers implement calculation routines, these generally yield a single iron-loss value per solution from which an understanding of the factors influencing iron-loss is hard to deduce. Chapter 4 develops an iron-loss procedure for implementation via FEMM that yields a total iron-loss value per field solution, but also stores data specific to highlighted areas of the machine that is then post-processed to help with the understanding of iron-loss trends.

Chapter 5 utilises the FEMM routines discussed in Chapters 3 and 4 to study the impact of phase number on the performance of concentrated winding machines with a suggestion that, given the advent of power electronic converters, phase number may become a design variable in the future.

Finally, Chapter 6 summarises and concludes this research study and presents ideas for future research.

## **CHAPTER 2**

### **WINDING DESIGN**

#### **2.1 Literature review on multiphase design**

For EV applications, the requirements for the traction motor including high power density, high efficiency, low weight, fast torque response, wide range of operation speed and most importantly reliability[27,28].The BLPM machine was gain increasingly attention because of its high efficiency and power density for same machine volume, less complexity in machine structure and fault tolerant capability [6,11].

In recent years, the multiphase design brought more advantages for the BLPM machine: enhanced reliability, improved torque characteristic, capability in expanding speed range, less losses in stator [27-32].There many research works published in recent years and they could be divided into two main areas: multiphase machine design [38-44]and multiphase machine drive [32-37].Although the papers [32-37] were focus on different aspects of multiphase machine, the advantages of multiphase design were described in detail. The multiphase structure provides lower winding voltage and leads to lower power rating for the inverters, relative less winding copper loses and potential capability of field-weakening related to the current rating. The harmonic, torque pulsation and noise in DC link were also analyzed correlated to permanent magnet design on rotor, stator slot design and type of winding. Fault tolerance of multiphase machine was stressed because of the importance of reliability for EV.

The advantages of multiphase PM machine make it as a suitable candidate for EV application. The idea of 5-phase, 6-phase, 9-phase and 12-phase machine modelling and design were proposed and there are few of them were built, concerning about higher harmonic in higher phase machine, i.e. 12-phase, and the compensation of performance



gain and machine complexity [31,38,41,42]. In [31] a model of 9-phase Interior Permanent Magnet (IPM) machine was built and the phase voltages were testified when driving the prototype machine. The harmonic of the 4-pole, 9-phase machine were analysis.

The 9-phase machine design in this research is focus on the impact from the DC link and the machine was designed for certain machine torque-speed characteristic. Less noise in DC link and less ripple peak torque was expected and the harmonic was an important design aspect to consider. In Table 2.1, [36,41,44,42,57,56] were found helpful during research, because similar rotor design, same analysis method and different phase winding consideration. S: surface mount; I: interior; CW: concentrated winding; DW: distributed winding; F: full pitched; FS: fractional-slot, DL: double layer; SL: single layer; PTC: Predictive Torque Control method.

Reference number	Phase number	Slot_number/ Pole_number	Magnet layout	Winding	Analysis method	Prototype machine built	Results compare	Comments
[32]	2,3,5	2,4	Surface mount rotor	2-phase; 3-phase; 5-phase	FEA	yes	---	2-phase 180° max torque, less peak current; 3-, 5-phase has less torque ripple; 5-phase has largest flux density or iron utilization
[44]	5	80/16;80/14	S rotor	CW	FEA	No	Simulation	Modified fractional-pitch winding, analysis only
[28]	5	60/12	S stator	DL	Numerical; Matlab	Yes	Validated	Develop a numerical model for a machine (how to build dynamic model)
[1]	--	--	--	--	Numerical	No	--	General multiphase machine winding
[2]	--	--	--	--	--	--	--	Wave model
[33]	4 star 3	96/--	S rotor	FS,CW	PWM VSI	--	--	Discuss lower torque ripple and fault tolerant
[36]	3,5	5/6	S R;I R	--	Numerical	No	Simulation	Model only,fault tolerate with comparable power density
[3]	--	--	--	--	Simulink	No	-	General dynamic model, energy point of view.
[27]	5	--	--	--	Numerical d/q	No	Simulation	MPM on EV A&D; Y&star connection -- speed range
[42]	3,4,4,5,5,6	24/20,16/14,16/18,20/18,20/22,24/22	S rotor	FS, CW	FEM, Maxwell's EQ	Yes	--	Loss in support structure, no model, measurements comparison
[4]	3	Multi set	--	--	Numerical ,	No	--	Dynamic model

		winding			FEM			
[5]	5	30/1	S rotor	DL, CW	Numerical , FEA	No	--	Numerical model validate by FEA, bearingless lead to simpler structure.
[34]	5	20/18	S rotor	DL;SL	Numerical	Yes	Yes	Current control improve 5-phase machine torque output; fault- tolerant operation
[11]	--	--	--	--	--	--	--	Summaries development in multiphase machine,drive and their applications. Design model and control methods.
[6]	4,5,6,78, 9,10,12	--	S rotor	CW	Numerical	No	Analysis	Compare design, loses, drive and cost for different multiphase number. Aerospace application.
[7]	6	--	--	Duplex set in 3	Numerical	No	--	d-q model; mutual inductance, phase and torque analysis.
[40]	3,4,5,6	24/20,16/14, 20/18,24/22	S rotor	FSCW, DL, SL	FEA	Yes	Simulation	Eddy currentloses for different clamp rings. Only 3/12/10 tested.
[41]	3,5	30/3,30/16, 30/32	--	FSCW	Numerical, FEM	No	Simulation	Harmonic analysis; Rotor loses caused by slot harmonic do not improve with increasing phase number
[8]	5	10/16	S rotor	--	Numerical, FEM	No	Simulation	Optimize design aspect for best machine output (efficiency, power or acceleration)
[43]	--	--	--	--	Numerical, FEM	Yes	Not tested	Redundant coil reduce switching loses, not much helpful for overall system
[30]	3,6,9,12	--	--	--	Numerical	No	Simulation	Analysis the harmonic and loses for 6-phase machine withdifferent winding layout.less field loses in 6-.
[16]	3,5	15/4,20/4	I rotor	FS,DW, SL,DL	FEA	Yes	Simulation and practical	Torque pulsation lower in 5-FSDL 15/4 machine,better than 5-FSSL 20/4 or 3-FSDL 15/4.
[29]	--	--	--	--	Numerical d-q	--	--	Literature of advantage of multiphase

								machine:reliability,torqueperformance and torque control.
[9]	-	--	--	FS	Numerical	No	--	Drive current harmonics analysis; rectangular excited multiphase motor has more flux.
[15]	5,7,9	21/--,27/--,28/--,36/--	I rotor	FS,DL	FEA	No	Simulation	Impact of stator winding to torque pulsation. 9-phase IPM motors has lowest torque pulsation.
[31]	9	32/4	I rotor	DW,DL	Numerical d-q	No	Simulation	d-q model for 9-phase IPM; fault tolerant capability and torque ripple reduction.
[10]	12	--	--	--	FEA	no	simulation	Stator mmf will be reduced by short-pitched winding in 12-phase asynchronous generator.
[36]	5	--	--	--	Numerical	Yes	--	Modular PM machine and drive design for phase current independent control and fault tolerate test.
[39]	6	24/22	--	--	FEM	yes	no	Flux density analysis for a 6-phase disc PM.
[55]	3,4,5,6	10/12,14/16,18/20,22/24	--	--	FEM	yes	no	Do not compare with similar phase machine.
[56]	3	12/10	S rotor, I rotor	FS	Numerical	no	simulation	Compare torque and faulty operation with different winding and rotor design(SPM IPM)
[35]	3	12/10	I rotor	FS,DL, nonoverla-pped	Numerical	yes	--	Analyze faulty condition: average torque, thermal load, torque ripple.
[37]	5	--	--	--	PTC	--	--	EV,5-phase machine drive realization.
[57]	3,15	60/4	S rotor, I rotor	--	Numerical, FEM	yes	practical	Capability of multiphase configuration and higher torque of Active stator machine.

Table 2.1 Table of reviewed reference for multiphase design.

## 2.2 Machine model analysis

For the interests in the machine design aspects, 3-phase Brushless Permanent Magnet (BLPM) machine models would be developed and analyzed in this chapter. The conventional 3-phase system provided a baseline for comparing distributed and concentrated windings; A 9-phase concentrated winding with the 3-phase equivalent would be compared in the next chapter. Although the concentrated winding was more common [13], because the affect of the harmonic field was not significant, the outputs of distributed winding were compared with the output of concentrated winding. At this stage, the effects of different winding schemes were focus on and other aspects of the machine design were not discussed, for example material, size, air-gap, back iron and tooth shape. For consistency, only the slots on the stator were modified according to the winding schemes. The outputs of the models, i.e. flux linkage, back EMF and torque were obtained and compared through the simulation.

### 2.2.1 Terms in model analysis

The permanent magnet provides a salient field in the air-gap of a machine [13]. It was also sensitive to temperature and working load. Under some of the extreme condition, i.e. over temperature and overload, it would be demagnetised which was irreversible process and affect the performance of the machine [13].

For N poles M slots permanent magnet machine, the pole pair,  $P = \frac{N}{2}$ , the mechanical angle between two teeth,  $\theta_m = \frac{360}{M}$  degrees and the electrical angle,  $\theta_e = \frac{360}{M} \times P$  degrees.

Current density, J, could be calculated as in equation 2.1:

$$J = \frac{NI}{A} \quad (2.1)$$

where I is the current carries in the conductor; A is the area of the slot and N the number of turns in slot. To avoid insulation breakdown due to overheating, the current density in the conductor should not greater than 8 A/mm<sup>2</sup> [17] and for most machine design, the value should be less than 5 A/mm<sup>2</sup> [18] in air cooling condition. For this machine models analysis, the current density was picked as 4A/mm<sup>2</sup>, which was lower than 5A/mm<sup>2</sup>. Further more, when the current was inject in the winding coil, for most of the case,

the material of stator, i.e. pure iron, would saturated [7, 13, 17-21] before thermal problem rising up.

### 2.2.2 Software tools

**AutoCAD:** AutoCAD is popular software for mechanical design. The original machine drawings of the machine models were accomplished in AutoCAD.

**Finite Element Method Magnetics (FEMM):** FEMM was the software employed to assist with the analysis of machine models using Finite Element Analysis (FEA) method [7]. It is very powerful in solving magnetic and electrostatic problems in 2-D dimension. FEMM could also import the machine model from a drawing file generated by AutoCAD. By modifying the property of the dots, lines and blocks, machine models could be setup easily. Fig 2.1 shows a machine model with 16 poles, 18 slots. Although FEMM was not good at 3D or dynamic model modelling, it was competent for pure electromagnetic field problems.

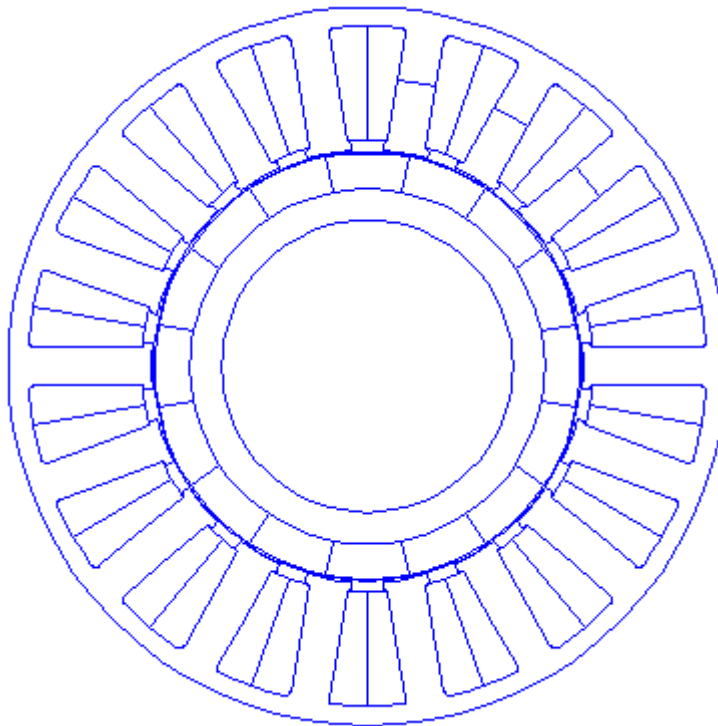


Fig. 2.1 An example of machine model in FEMM.

Lua script with FEMM: Lua script is a coding programme associate with a FEMM model. It has full range of command for defining, modifying and analyzing the model. It was used for data recording and rotor model rotating.

MATLAB: MATLAB is a mathematical tool for data analysis. A MATLAB programme was written to accelerate the speed of data processing. A few functions in MATLAB were used for analyzing the data record by Lua script and also deriving the back Electromotive force (EMF) from the data. These functions included data input, array calculation and plot graph. Fig.2.2 shows the progress of data processing using MATLAB programme.

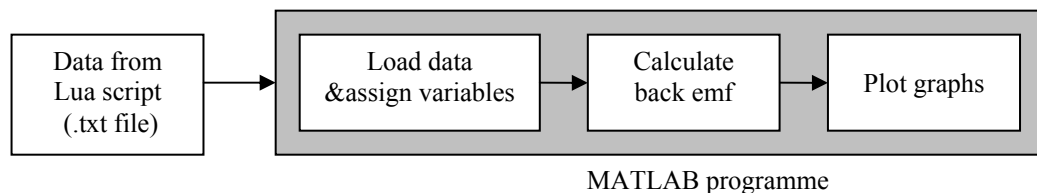


Fig. 2.2 MATLAB programme data processing diagram.

### 2.2.3 Verify design method using FEMM via search coil test (MSc taught exercise)

To exam the reliability of FEMM, a search coil test had been established. The accuracy of software was very important in machine design and simulation, i.e. how good the simulated result tie with practical result. In this section, simulated results and experimental measurements were compared to ensure machine modelling and understand of FEMM were correct. The Back EMF of the search coil would be simulated in FEMM first, and then the 10 turns coils would be winded on the machine and the output would be measured.

#### 2.2.3.1 Model employed

The machine model was a small brushless permanent magnet generator, 20 Watts, which was given from the assignment called ‘Design of a brushless permanent magnet generator and rectification stage’. Dimensions was shown in I.15 in Appendix I. Along with the FEMM model, the actual machine was also provided and as a result, this particular model was selected at the first place. By modifying the definition of the block properties, the accomplished model was shown in Fig 2.3. The curves crossing the teeth were used for deriving the flux in the teeth or flux linking the coils under unsaturated condition.

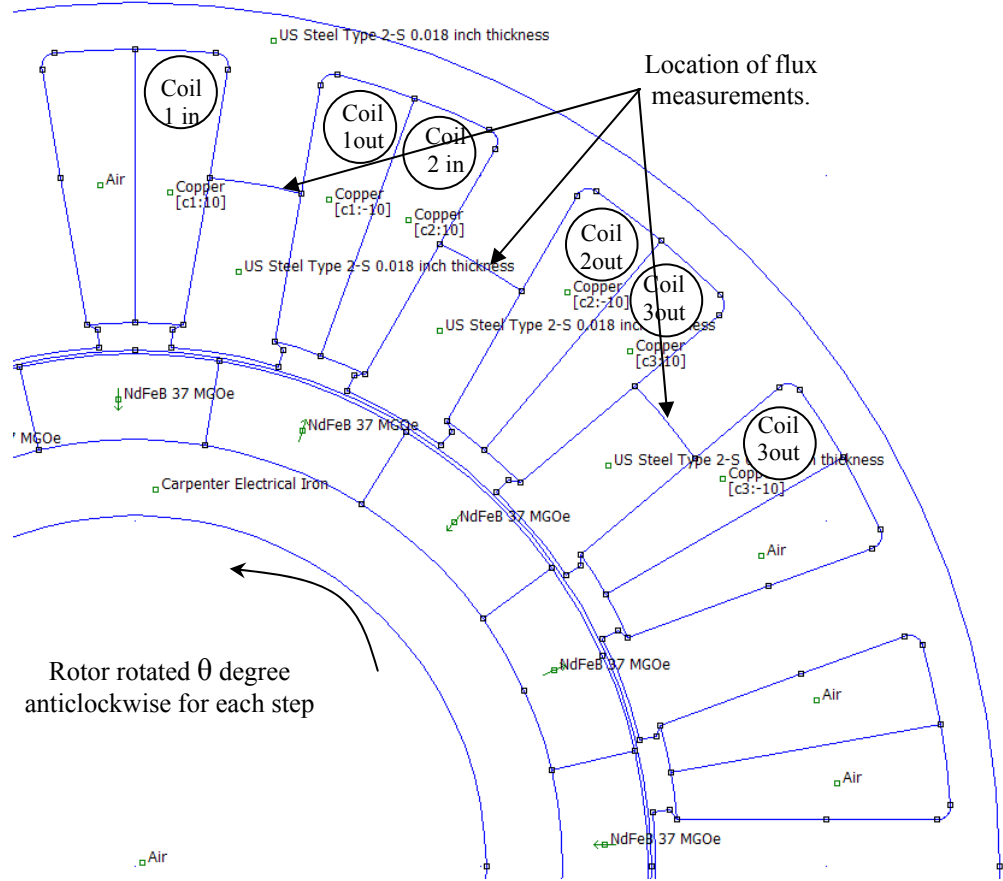


Fig. 2.3 FEMM model of the generator, with 16 poles, 18 slots.

Although the coils were defined in the model, no back EMF could be measured across the coils, because of zero resistance along the coils. The back EMF could be derived from the flux linkage changes of the coils according to Faraday's law. The back EMF generated by coils in a changing magnetic field is defined by:

$$e = \frac{d\phi}{dt} \quad (2.2)$$

where  $\phi$  is the total magnetic flux linking the coil in Webers (Wb) and  $t$  is the time.

Detail calculation would be shown in section 2.3.3.

The plot of flux density for the model was shown in Fig 2.4 below. It was useful to check the mechanical design to avoid the magnetic saturation in the iron. The saturation area would likely occur the flux leakage[18, 19]. The B-H characteristic of steel was selected from FEMM material library and the material would start to become saturated when flux density reached 2 Tesla (T). In this section, the saturated was checked visually via flux density plot.

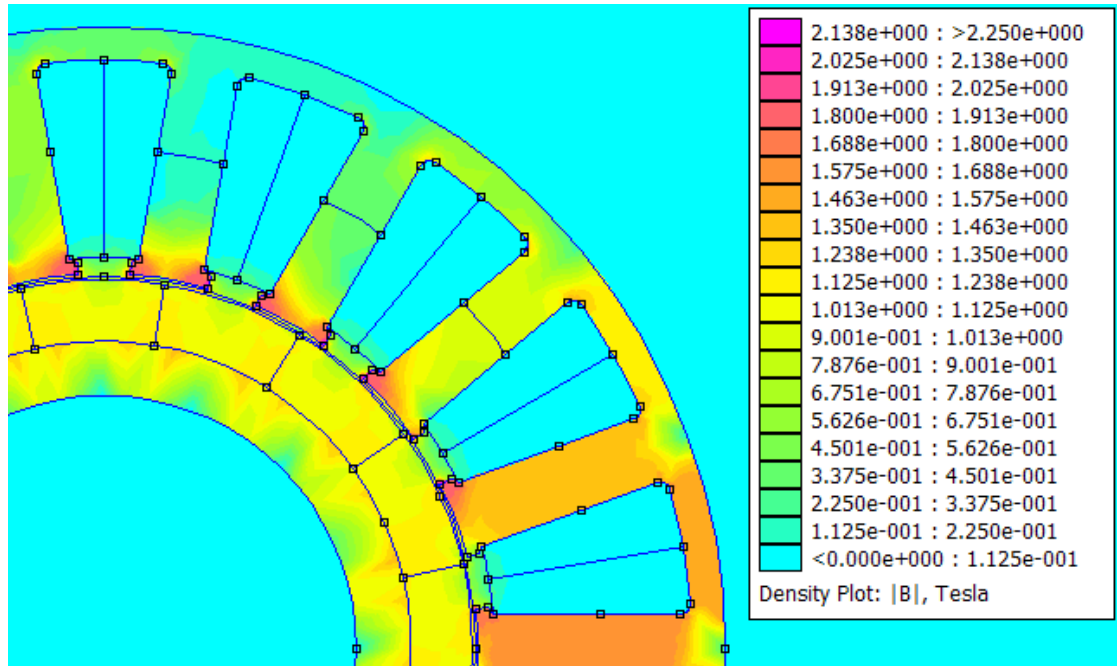


Fig. 2.4 An example of flux density plot in FEMM.

### 2.2.3.2 Model analysis

A Lua script programme was written for data collection, referring to the FEMM manual. It was combined with functions of model rotation, variable measuring and data recording. The automatic code accelerated the speed of data collection dramatically, comparing with manual operation. I.1 in appendix I shows the main functions of the Lua script, including rotating, object selecting, integration and data recoding. By running the Lua script programme, the flux linkages in the teeth of the generator model were recorded while the rotor rotating  $\theta$  degree anticlockwise for each step for 360 degree in electrical angle. For this test, there were 50 steps within 360 degrees and the angular velocity was 7.2 degree per step. Assuming there was 1 second between steps, the angular velocity became 7.2 degree per second. The output of the code was shown in I.13 in appendix I, listing the amount of flux in three different locations. The flux at time 0, before the model rotating, was measured manually to complete the output plot in analysis.



### 2.2.3.3 Data processing

The back EMF generated by coils in a changing magnetic field was defined by Faraday's law, equation 2.2. Because the flux linkage  $\phi$  was the sum of the magnetic flux for a single turn coil,  $\phi$ , the equation became:

$$e = N \frac{d\phi}{dt} = N \left( \frac{d\phi}{d\theta} \right) \frac{d\theta}{dt} = N \omega_r \frac{d\phi}{d\theta} \quad (2.3)$$

$e$  is the back EMF of the coil, unit Volt;  $N$  is the number of turns of the coil;  $\phi$  is the magnetic flux per coil in the tooth, unit Wb;  $\theta$  is the mechanical position of the coil or tooth, unit radian;  $\omega_r$  is the mechanical speed of the coil or tooth, unit rad/s.

As a result, the Back EMF per turn per unit speed could be stated as  $\frac{d\phi}{d\theta}$  (rate of change in flux linkage for each step), and it could be calculated using the data shown in I.13 in appendix I. Lua script output file. For example, if there was 1 turn on tooth 1 and the rotor rotated at unit speed 1 radian per second mechanical, the back EMF at 2 second was:

$$e = 1 \times 1 \times \frac{-2.17 \times 10^{-6} - 1.37 \times 10^{-6}}{(1.8 - 0.9) \div 360 \times 2\pi} = 2.25 \times 10^{-4} (V).$$

A MATLAB code was programmed to analyze the data recorded by Lua script. This programme was also designed for later development: programme blocks and variables could be modified to fit any dimension of output data. Code examples were shown in I.2 in appendix I, including data file realisation, calculating and graph plotting.

By running the MATLAB code, the flux and back EMF were plotted in Fig 2.5. The maximum value of the back EMF for 1 turn coil could be achieved was  $2.275 \times 10^{-4}$  V per unit speed. This value was obtained from calculated back EMF data using MATLAB.

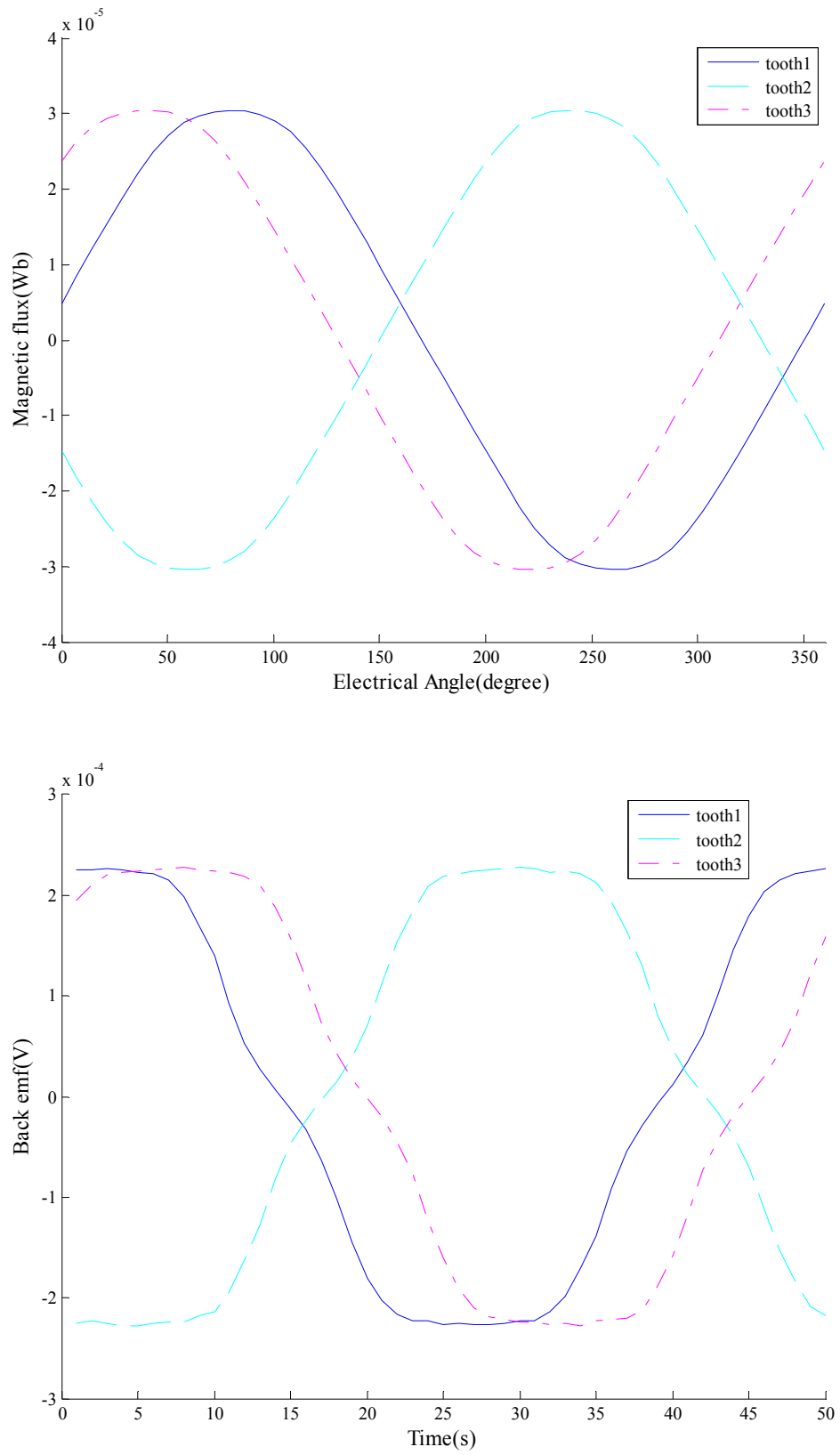


Fig. 2.5 Plot of simulation result: Flux(top) and Back EMF(bottom).

#### 2.2.3.4 Estimated result

To estimate the back EMF for search coil, the mechanical speed of the rotor and the number of turns of the coil required for the calculation using equation 2.3. If the angular velocity was 970.5 rpm (round per minute) and the search coil was 10 turns, the maximum back EMF could be calculated as following:

$$e_{\max} = 10 \times \frac{970.5 \times 2\pi}{60} \times 2.275 \times 10^{-4} = 0.2312(V)$$

Similar to the calculation above, the estimated back EMF waveform for ten turns coil at 970.5 rpm could be plotted as shown in Fig 2.6. At time 14.3715s, the back EMF of coil 1 was 0 and 42.17s and 19.3s for coil 2 and 3 respectively. The phase shift in electrical degrees between coils could be calculated as:

$$phase_{1-2} = \frac{14.3715 - 42.17}{50} \times 360 = -200 \text{ or } 160 \text{ (degree),}$$

$$phase_{1-3} = \frac{14.3715 - 19.93}{50} \times 360 = -40 \text{ (degree).}$$

So in electrical, the phase shift between coil 1 and coil 2 was 160 degree and -40 degree between coil 1 and coil 3.

From slot distribution side to analyze the phase shift, the generator had 18 teeth and 16 poles, so the mechanical angle between teeth was 20 degrees and the electrical angle would be 160 degrees. The anticlockwise was the positive direction and the phase shifts of the back EMF between coil 1 and coil 2 should be -20 or 160 degrees and -40 or 320 degree electrical between coil 1 and coil 3. Fig 2.7 shows the phase shift drawn with MATLAB.

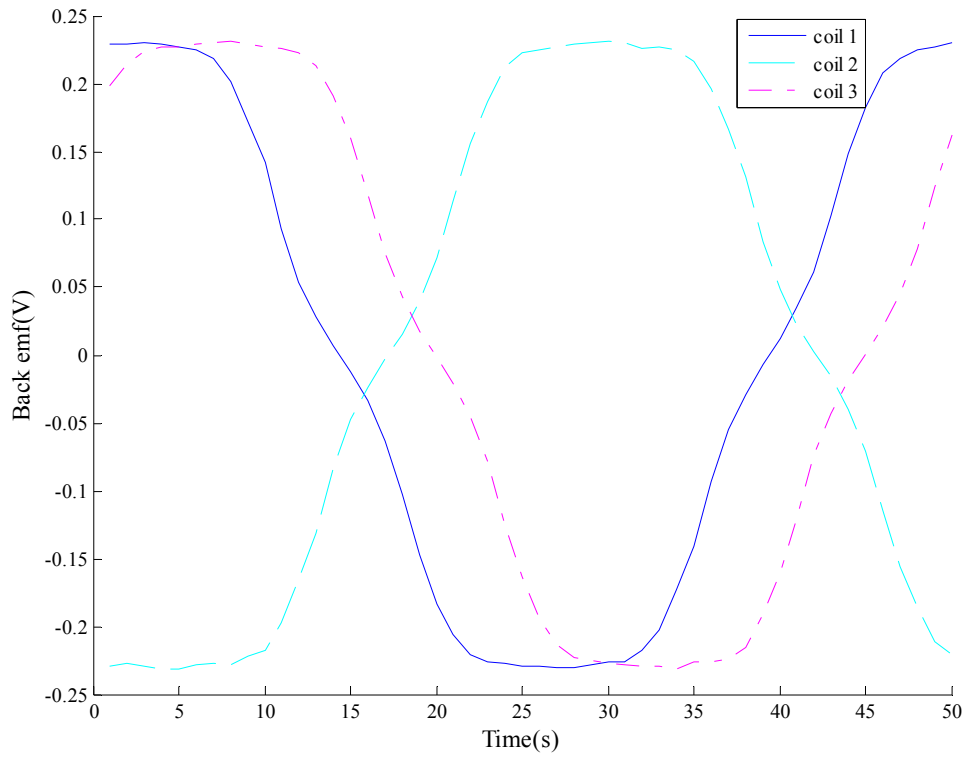


Fig. 2.6 Estimated back EMF waveform for search coil.

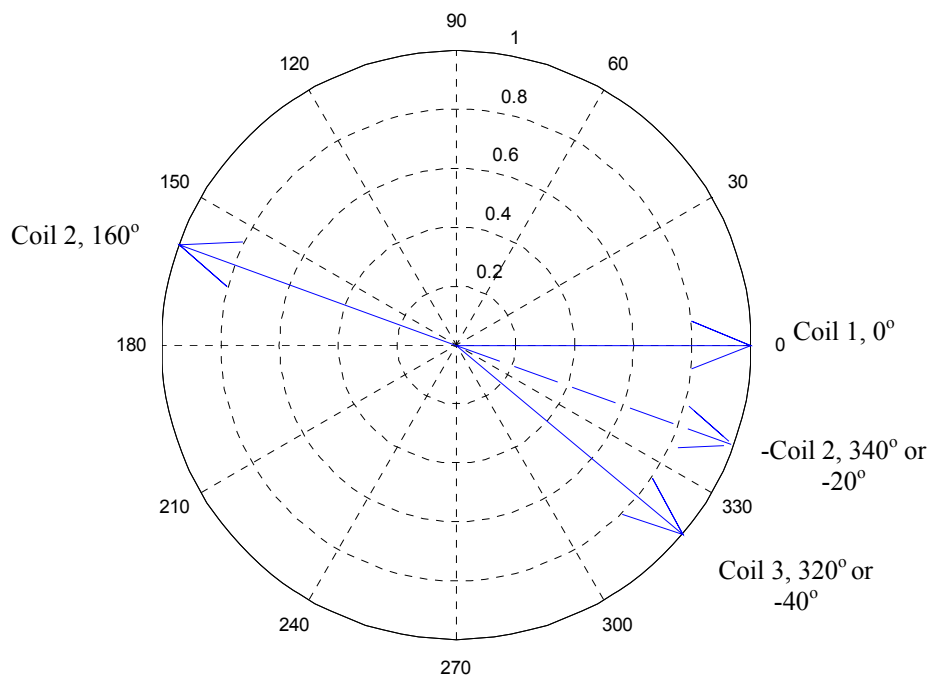


Fig.2.7 Phase shift of back EMF of search coils.

### 2.2.3.5 Search coil test

An electrical drill was used to rotate the rotor of the generator at 970.5rpm and an oscilloscope was used to measure the back EMF of three sets of open-circuit coil. 10 turns of copper coil were wound on tooth 1, tooth 2 and tooth 3 respectively and the coils are connected to different channel of the oscilloscope as Fig 2.8 shown below. 'A' 'B' and 'C' indicated different coils and '+' and '-' shown direction of winding. The drill was kept running at constant speed for a period for data collection.

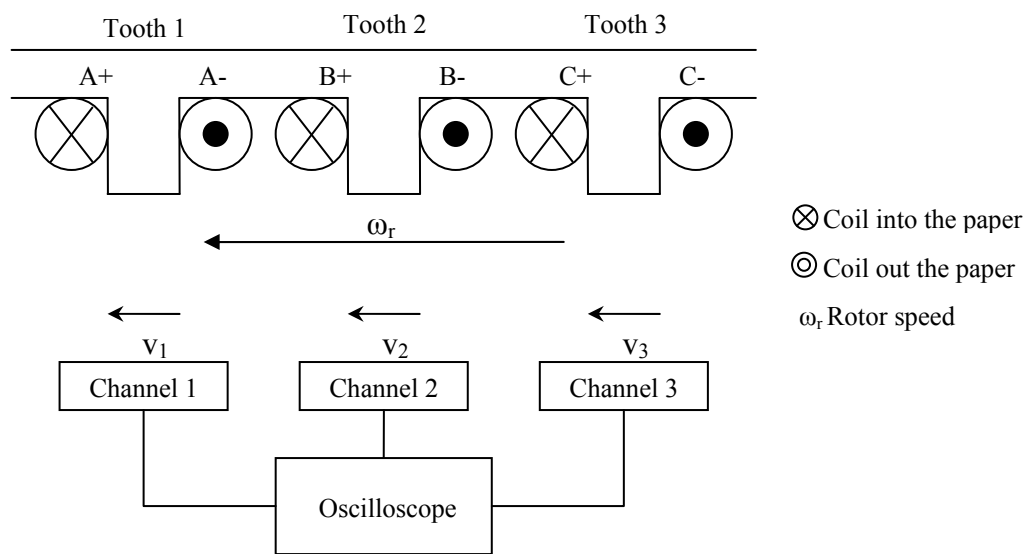


Fig. 2.8 Search coil test setup.

In order to import the data using Matlab, the format of the data files was converted to excel worksheet, 'xls'. Data from oscilloscope are plotted using Matlab, shown in Fig 2.9 and the phase angles between each output can be obtained using those data. Table 2.2 shows the calculation about the phase between the coils. The method was the same as used in section 2.3.4, the 'zero' back EMF points were selected and phase angles were determined by the subtraction of time difference. The measured emf may look higher than predicted value, because the spikes at the peak of emf waveform or the inconsistent rotor speed.

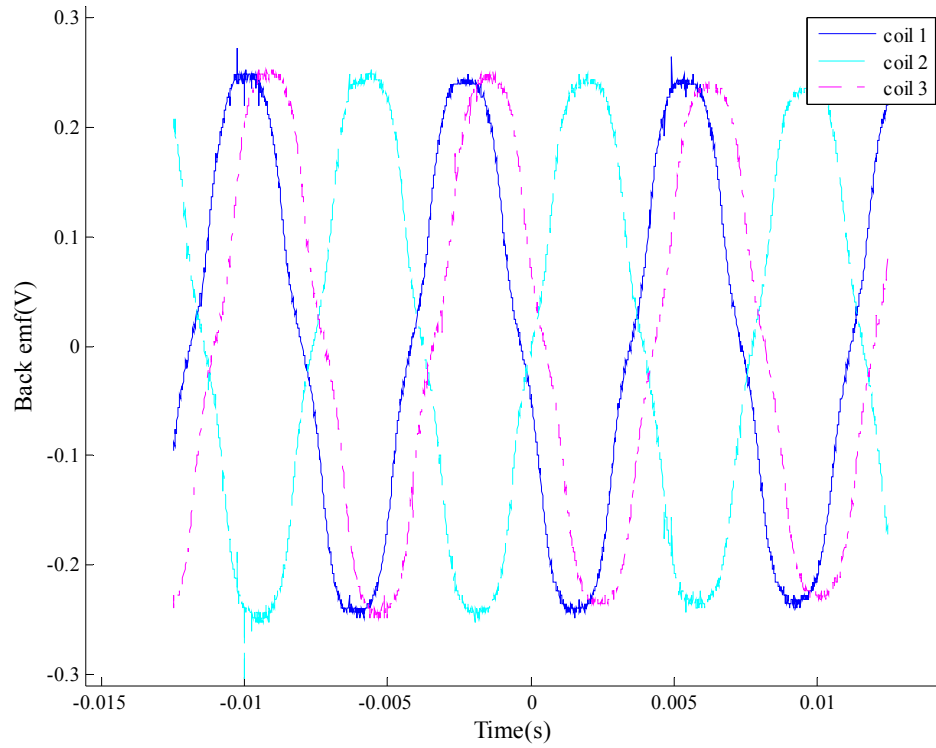


Fig. 2.9 Back EMF waveform.

Sequence number of zero emf	coil 1 (s)	coil 2 (s)	coil 3 (s)	period (coil 1) (s)	c1_c2 (s)	phase shift (c1_c2) (degree)	c1_c3 (s)	phase shift (c1_c3) (degree)
1	-0.02293	-0.02161	-0.02062		-0.01254		-0.00231	-36.62629
2	-0.01153	-0.01039	-0.00884	0.0228	-0.01273	-198.8285	-0.00269	-42.6514
3	-0.00021	0.0012	0.0021	0.02264	-0.01249	-201.841	-0.00231	-36.62629
4	0.01136	0.01228	0.01375	0.02314	-0.01264	-198.0357	-0.00239	-37.89474
5	0.02248	0.024	0.02486	0.02224		-200.414	-0.00238	-37.73618

Table 2.2 Calculation for the phase difference.

#### 2.2.3.6 Search coil conclusion

Comparing the estimated waveform with the experimental back EMF plot, the maximum value of both were about 2.4 Volts and the experimental phase shift was less than 2 degree different with the estimated data. So the search coil proved the reliability of FEMM model. As seen in Fig 2.11, the back EMF was slightly greater than estimated waveform, the phase

shift was not exact and the noise combined with the waveform. This was because the speed of drill was not constant and the coil was measuring without load.

#### 2.2.4 Concentrated winding model

In next two sections, concentrated and distributed windings were compared. The model used in this test was 32 poles 36 teeth/slots PM machine, so the slot pitch,  $\alpha$  and pole pitch,  $\beta$ , could be derived using the number of poles as following:

the slot angle,  $\beta = \frac{360}{36} = 10$  (degree), the pole pitch,  $\partial = \frac{360}{32} = 11.25$  (degree).

The 3-phase concentrated winding scheme was shown in Fig 2.10 which was short pitched because the slot angle was less than one pole pitch. Dimensions was shown in Table I.15 in Appendix I.

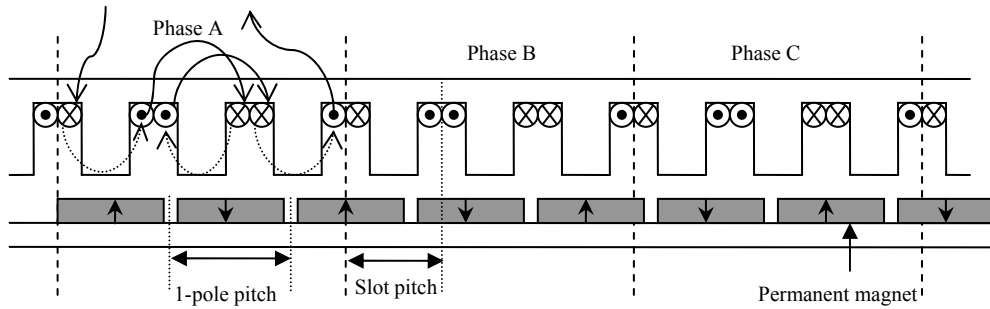


Fig. 2.10 3-phase concentrated winding.

Apart from changing the properties in FEMM model manually, there were alternative ways to modify the model by which either using commands in Lua script or edit the FEMM file with Notepad. Using notepad to edit the FEMM file was more convenient once first similar model set up, especially for large amount of blocks. I.14 in appendix I shown below was an example of model block properties. The block list contained all the information needed to build a machine model. It included block location, material information, grouping, circuit and number of turn for phase winding.

By loading the available drawing from AutoCAD and modifying the block properties, the FEMM model for the 3-phase concentrated machine, shown in Fig 2.11, was accomplished for further analysis. The slots in the stator were divided to two to meet the double-layer winding.





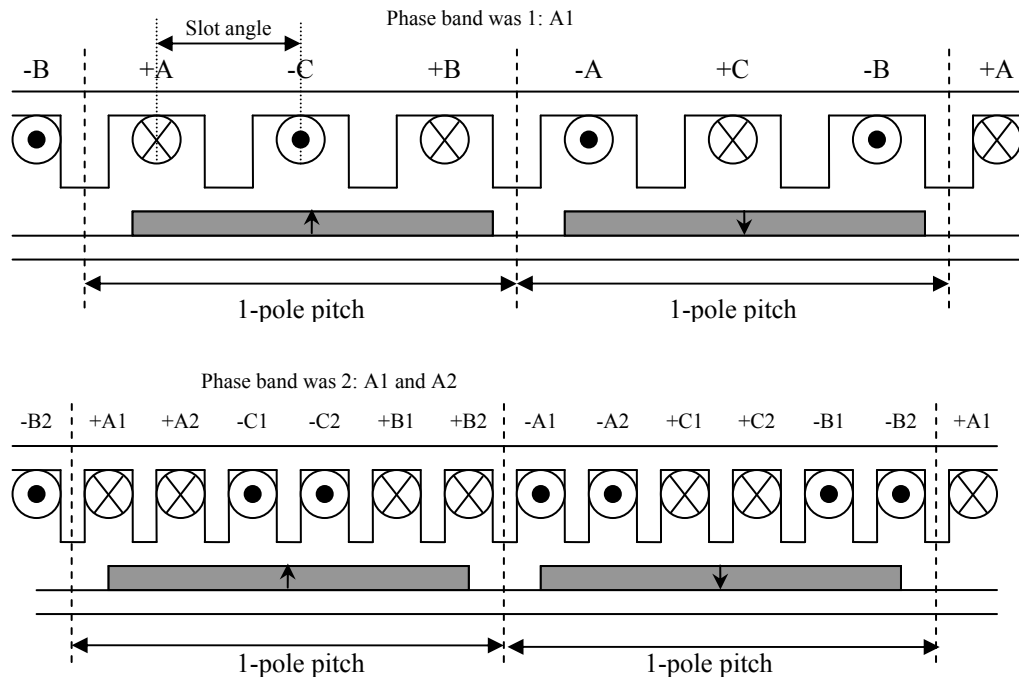


Fig. 2.12 Stator winding for 1 slot per pole per phase distributed winding (top) and 2 slot per pole per phase distributed winding (bottom).

To create the machine model, a basic stator block was drawn in AutoCAD. By using the 'Polar Array' function, the basic block could be duplicated around a specified centre point. The basic block was shown in Fig 2.13. With modified block properties, the accomplished distributed winded machine models were shown in Fig 2.13. Noted that the phase winding were defined as several open circuit. Due to the size matter, only parts of the machine models shown in Fig 2.14. 1 slot per pole per phase distributed winding could be difficult to build and 2 slot per pole per phase distributed winding was not realistic since the size of slot.

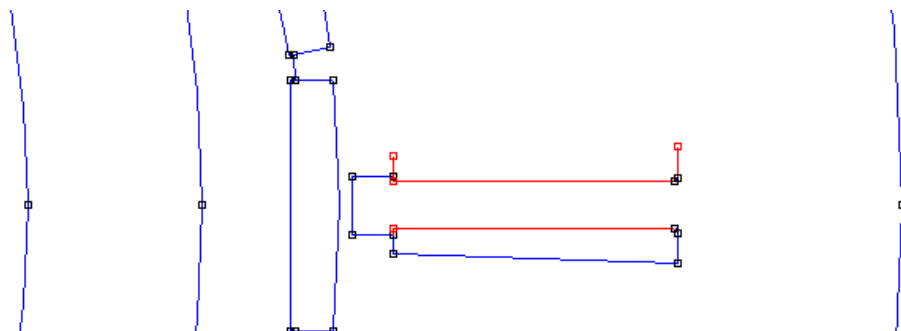


Fig 2.13 Basic tooth and slot block.

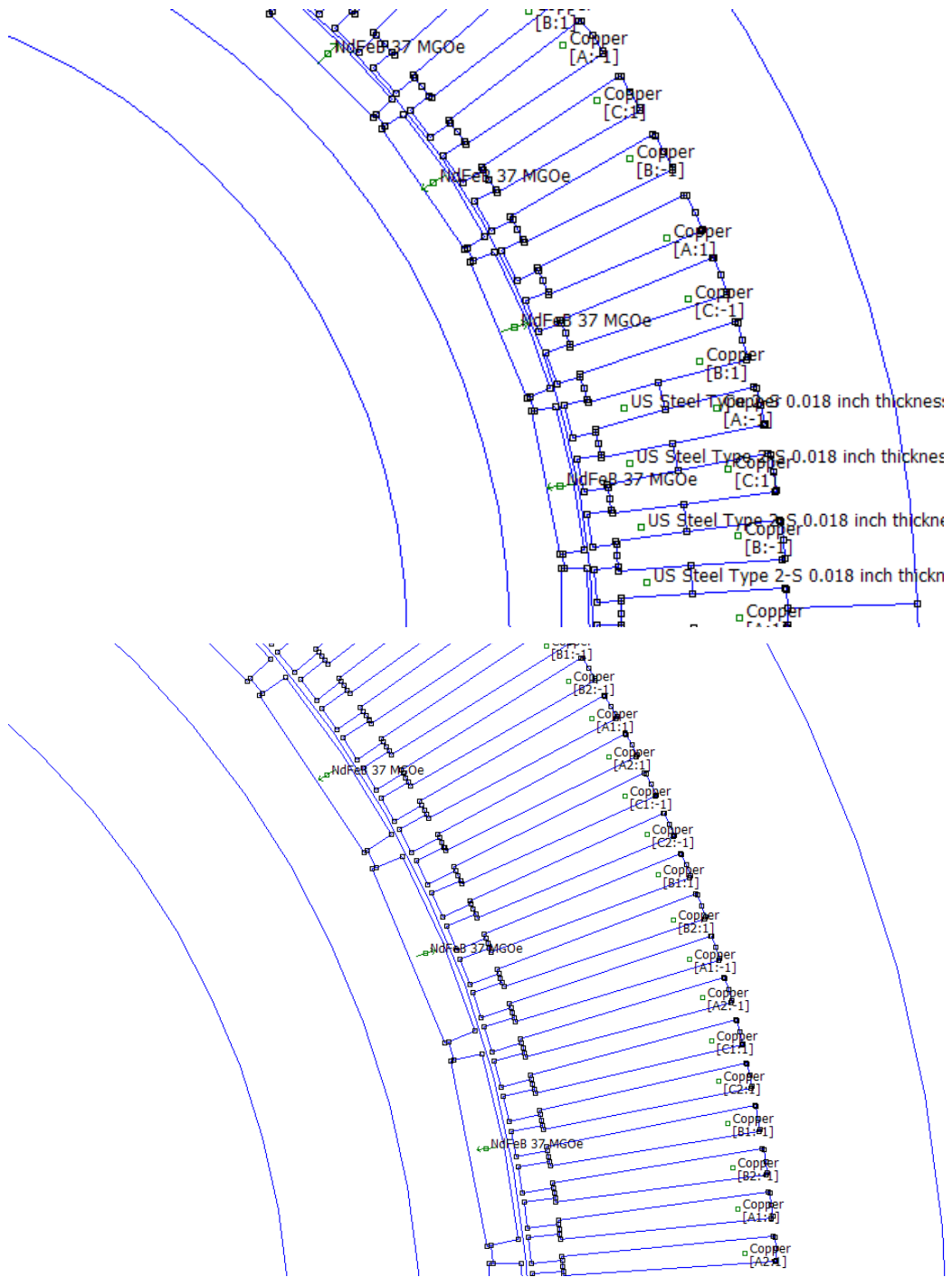


Fig 2.14 FEMM models for 1 slot per pole per phase distributed winding (top) and 2 slot per pole per phase distributed winding (bottom).

### 2.2.6 Model analysis and outputs comparison

The Lua script was modified to analyze the models for two electrical circles, i.e. from 0 degree to 720 degree. The 'circuit properties' command was used for determine the overall flux linking and torque. I.3 in appendix I shows an example code for circuit analysis. The MATLAB code was also used to process the data and plot the waveform. The analysis for the torque required injecting current into the winding and so the process was divided to two parts: Flux based test and current based test.

According to equation 2.3, the back EMF was direct proportional to the rate of change in flux linking the phase winding per angular displacement,  $\frac{d\phi}{d\theta}$ . For model simulation, the angular displacement was constant, so the waveform of flux linking difference was mapped with the back EMF. The rate of change in flux linking  $d\phi$  could be derived by subtracting the flux linking the winding at previous position from present position, similarly with displacement,  $d\theta$ . The 'diff' function in MATLAB would use for handling subtracting calculations, shown in I.2 in appendix I.

For the torque analysis, the windings were connected in 'star' form, shown in Fig 2.14. The Direct Current (DC) torque outputs were measured by Maxwell's stress tensor integral when the current density was  $4\text{A/mm}^2$  and  $8\text{A/mm}^2$  for both 1slot and 2 slots per pole per phase models. The reason for inject DC current rather than Alternating Current (AC) current was to avoid the introduction of harmonic by driving current and to stress on the effect from different winding schemes and stator design.

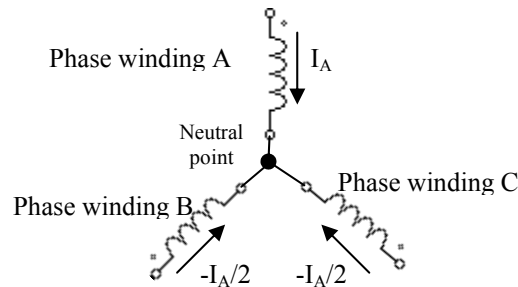


Fig. 2.15 Balanced star connection.

#### 2.2.6.1 Comparison of flux linking and back EMF waveform

The flux linking the winding coil for the concentrated winding 3-phase AC machine model was recorded using Lua script and the data were processed using MATLAB programme. Fig 2.15 shows the flux linking the winding coils and the rate of change in flux linking versus the electrical angular position. Noted that, the back EMF waveform would have the same shape with waveform of the rate of change in flux linking the phase windings,  $\frac{d\phi}{d\theta}$ , and different amplitude, as described in previous section. With similar process, the outputs of the machine with concentrated winding were plotted in Fig 2.16 and Fig 2.17. The flux linkage was expected to be sinusoidal for concentrated winding and more triangle looking for distributed winding. The simulated waveforms matched with expectation.

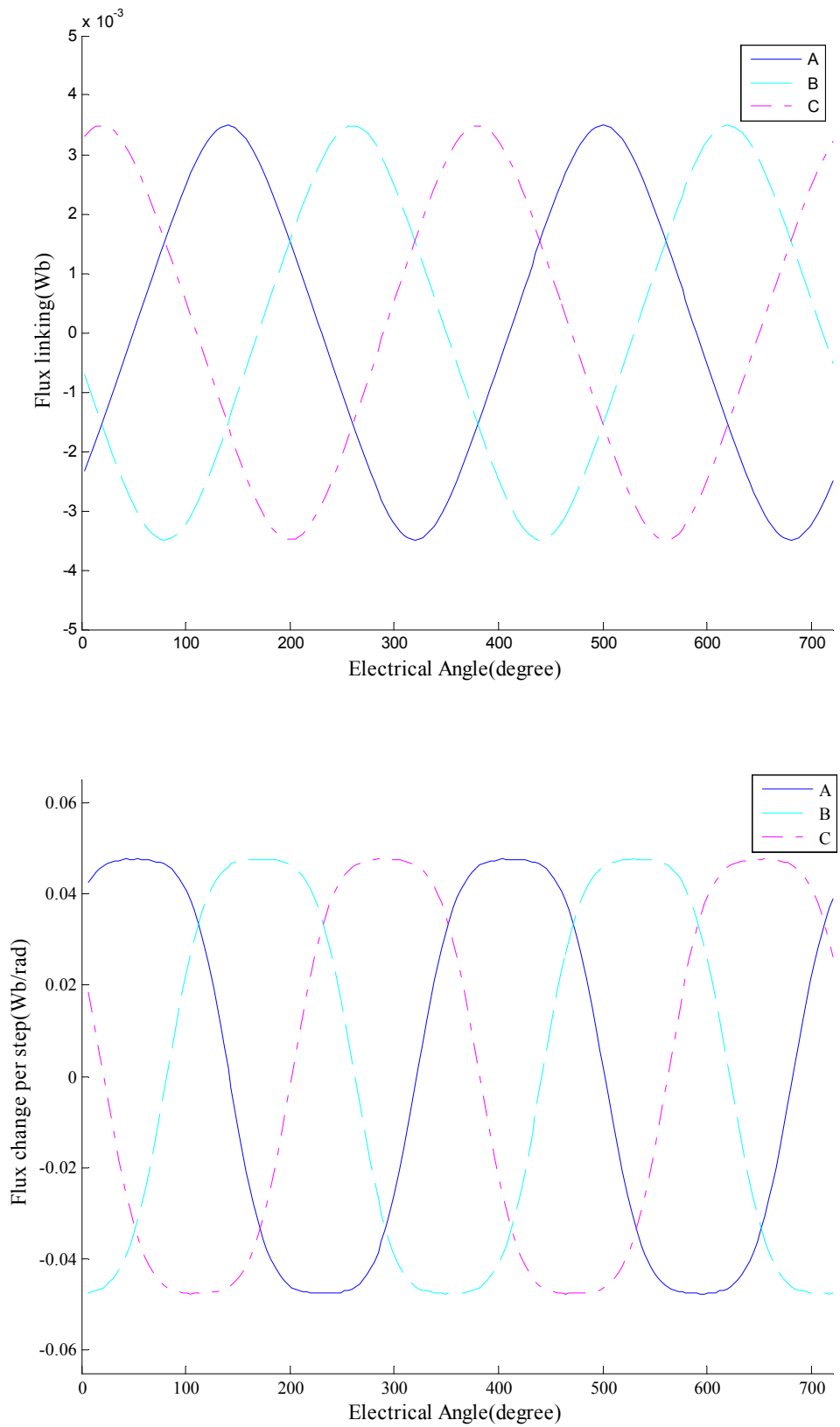


Fig. 2.16 Plot of flux linking the phase windings A,B and C (top) and  $\frac{d\phi}{d\theta}$  (bottom) for 3-phase PM machine with short pitched concentrated winding.

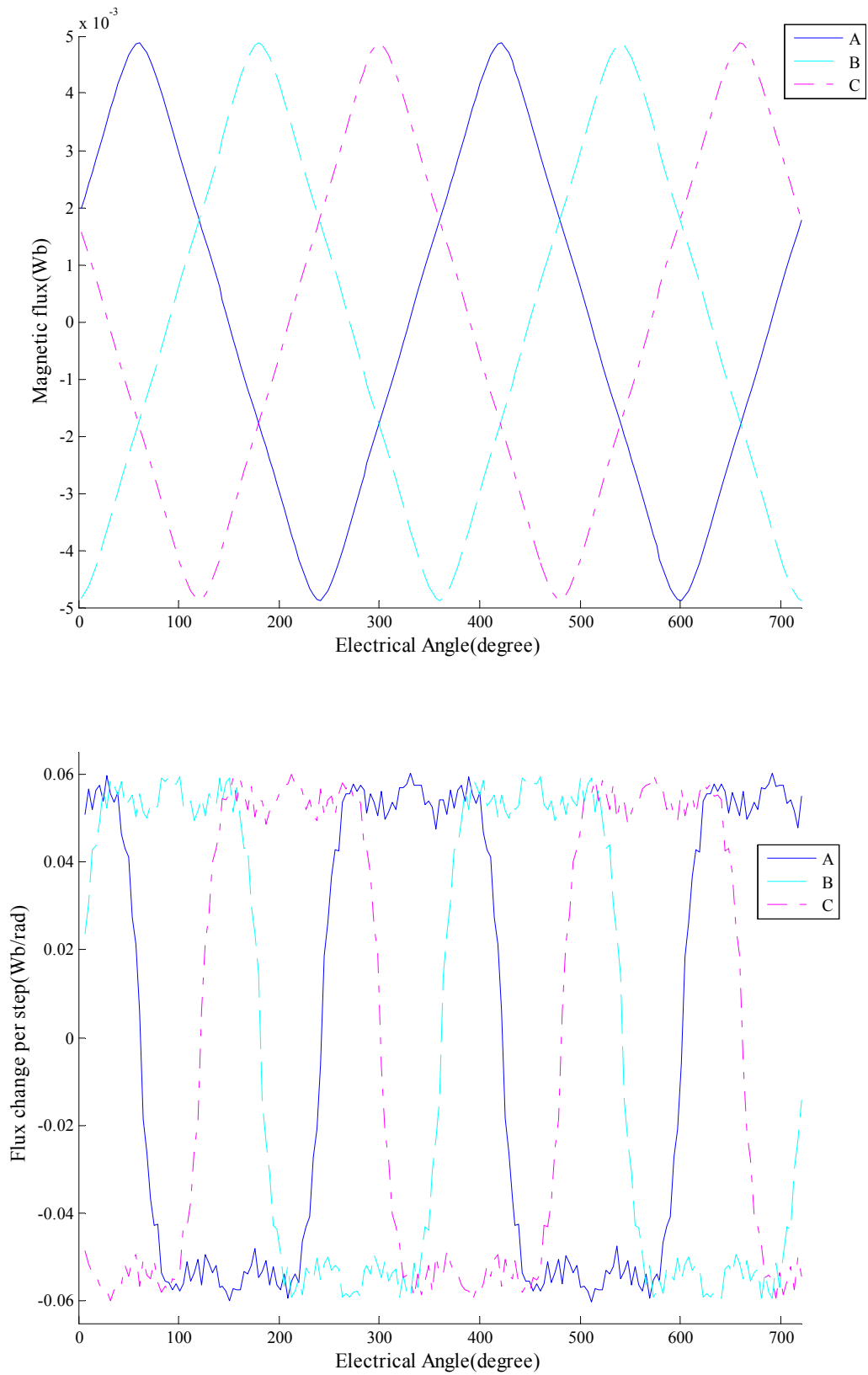


Fig. 2.17 Plot of flux linking the phase windings A,B and C (top) and  $\frac{d\phi}{d\theta}$  (bottom) for 3-phase PM machine with full pitched distributed winding with phase band 1 (1 slot per pole per phase).

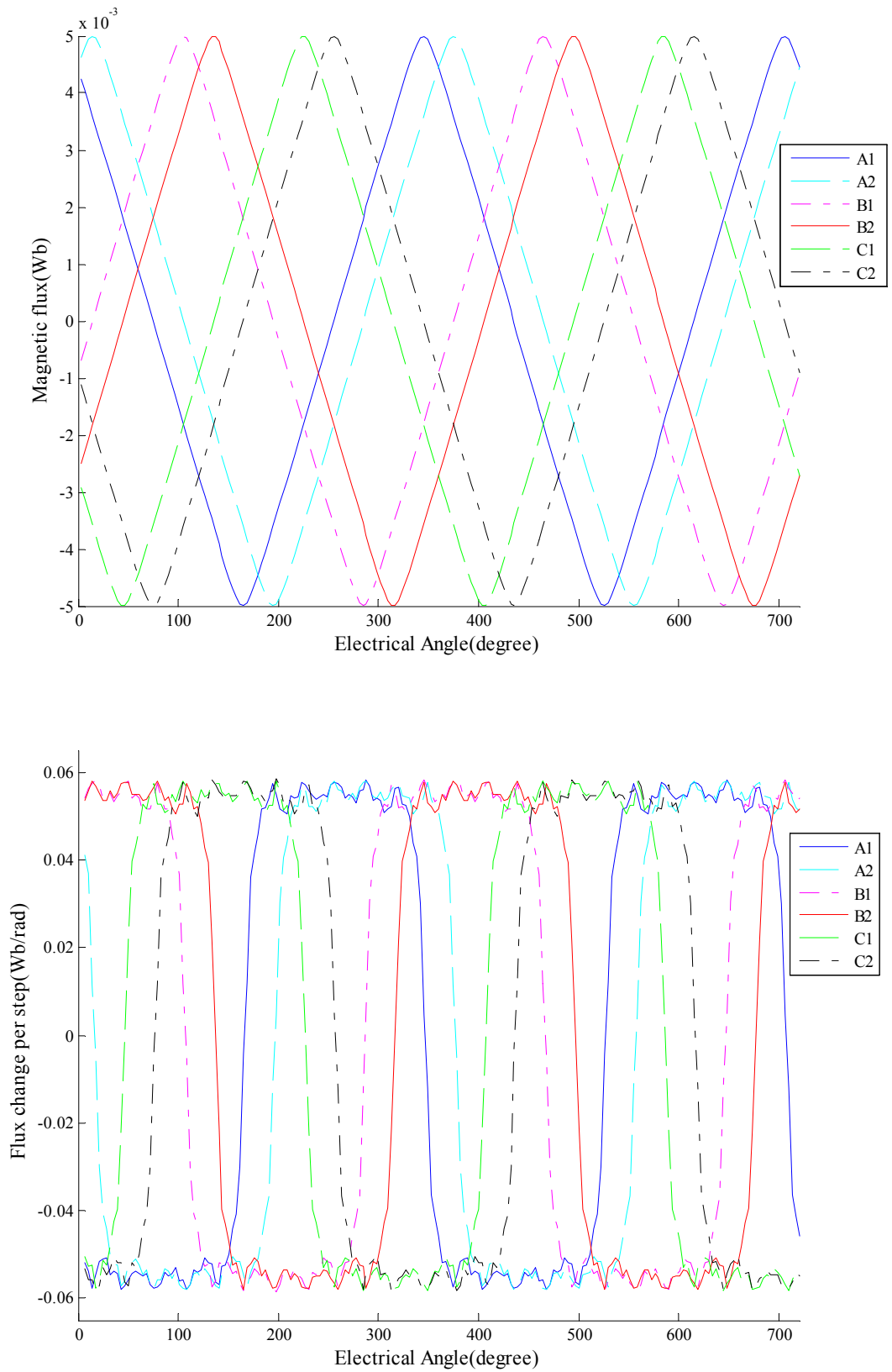


Fig. 2.18 Plot of flux linking the phase windings A,B and C (top) and  $\frac{d\phi}{d\theta}$  (bottom) for 3-phase PM machine with full pitched distributed winding with phase band 2 (2 slots per pole per phase).

For a PM machine model, the explanation of how a sinusoidal waveforms is formed could be refer to the winding scheme. Comparing the flux linking per phase winding, the peak flux linking in concentrated winding was less than the flux linking for distributed winding. Referring to Fig 2.10, there were less than three magnets providing the magnetic field for phase A. Assuming there was no flux loss, no gap between magnets and the flux provided from one magnet was 1. The maximum flux linking,  $\phi$ , or total number of magnets affecting phase A would be

$$\phi_1 = 1 - \frac{\partial \times 3 - \beta \times 3}{\partial} = 1 - \frac{11.25 \times 3 - 10 \times 3}{11.25} = \frac{2}{3} \approx 0.667.$$

Because both of the distributed winding machines were full pitched, shown in Fig 2.12, the maximum possible flux linking for phase winding A or A1 and A2 was 1. As a result, the maximum flux linking in distributed winding was about 33.3% more than it was in concentrated winding.

The shape of flux linking waveforms from those two winding schemes was also different. As shown in Fig 2.18, when the rotor rotated from position 1 to position 2 for the concentrated winding, the flux linking phase winding A, maximum values in this case, was maintain the same. This was the reason why the peak of the sinusoidal [13, 17] waveforms looked more smooth than the flux waveform of distributed winding. The distributed winding did not have the capability to maintain a value of flux linking and the waveforms were sharper. As shown in Fig 2.19, after the resultant flux equalling to zero, the position 1, the flux linking phase A winding changed sharply until the maximum flux linking position 2, by reducing flux upwards and increasing in downwards.

The waveforms for flux changing in distributed winding were also steeper than concentrated winding, shown in 2.19, 2.20 and 2.21 bottom. The longer period of peak back EMF may benefit the output rating, i.e. continuous torque of the machines with distributed winding. Noted that the wobbles on the peak of rate of flux changing, or back EMF waveform, for both winding schemes, were caused by the slot air-gaps [13].

It was the same for the waveform shape of 1 slot per pole per phase and 2 slots. Only difference was 2 slots per pole per phase had a duplicated set of waveforms shifted by a slot pitch, shown in Fig 2.17.



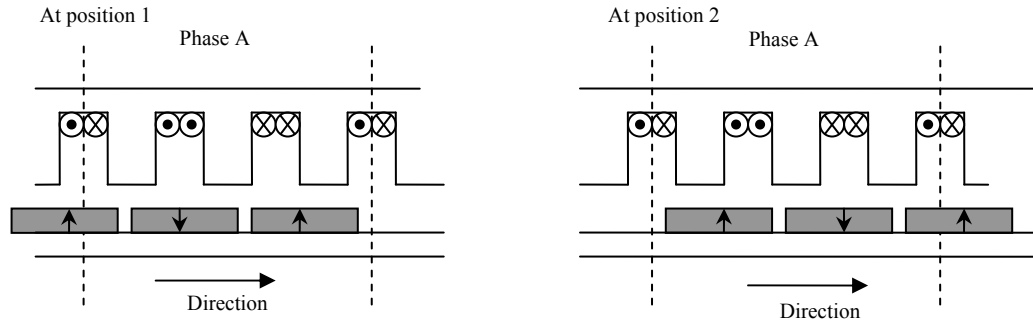


Fig. 2.19 Rotating process for concentrated winding.

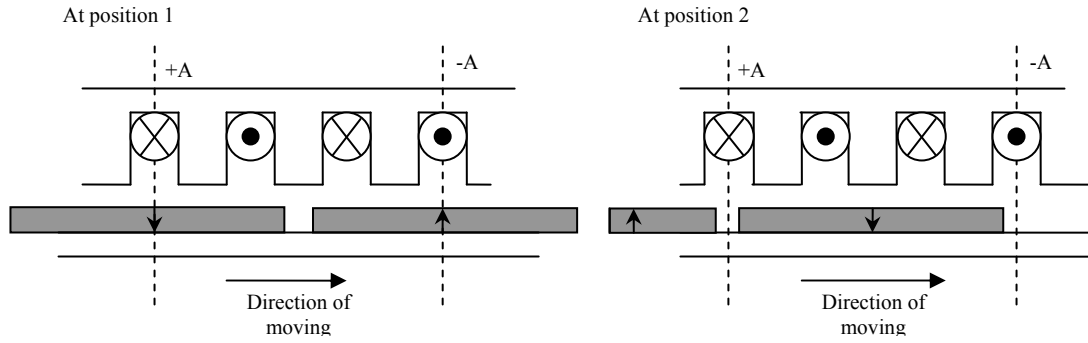


Fig. 2.20 Rotating process for distributed winding.

#### 2.2.6.2 Torque analysis

The slot areas were obtained in AutoCAD using 'pedit' and 'area' commands and the result table was shown in table 2.3. Referring to equation 2.1, when slot area was  $116.87 \text{ mm}^2$  and the current density desired was  $4\text{A/mm}^2$ , the current required to inject into the single coil phase winding A,  $I_A$ , was calculated as below:

$$J = \frac{NI}{A} \Rightarrow I_A = \frac{J_A \times A}{N} = \frac{4 \times 116.87}{1} = 233.74(\text{A}).$$

The current in phase B and C was

$$I_B = I_C = -\frac{I_A}{2} = -\frac{233.74}{2} = -116.28(\text{A}).$$

Winding types	Slot area (mm <sup>2</sup> )	Current in A(A) when 4A/mm <sup>2</sup>	Current in B,C(A) when 4A/mm <sup>2</sup>	Current in A(A) when 8A/mm <sup>2</sup>	Current in B,C(A) when 8A/mm <sup>2</sup>
Concentrated winding	116.87	233.74	-116.87	467.48	-233.74
Distributed winding 1 slot	55.61	222.44	-111.22	444.8	-222.4
Distributed winding 2 slots	27.8	111.2	-55.6	222.4	-111.2

Table 2.3 Table of slot area and current injected.

By modifying Lua script, the torque information was collected by line integral on the defined contour, derived from Maxwell's stress tensor. The data were recorded for two electrical periods. The results for different machines were plotted in Fig 2.20 Fig 2.21 and Fig 2.26. The torque waveform of concentrated winding was typical[13, 17, 18] smooth and sinusoidal waveform. Under the same current density, the waveforms of distributed windings were in distorted sinusoidal waveforms, especially for 1 slot per pole per phase. In a PM machine, the cogging torque was existed because of the air-gap between permanent magnets on the surface of the rotor or stator. The effect of the cogging torque was made more extreme by the full-pitch slot design in distributed winding, referring to Fig 2.23 position 2. It was not a good design with such distorted torque output, although it rated highest torque among the three models. The output of 2 slots per pole per phase model was more sinusoidal because of the compensation of the second set of winding. Increasing in phase band was not always reducing the distortion because of the harmonic introduced [13]. For the same model under different current densities, the torque increased directly proportional to the current density.

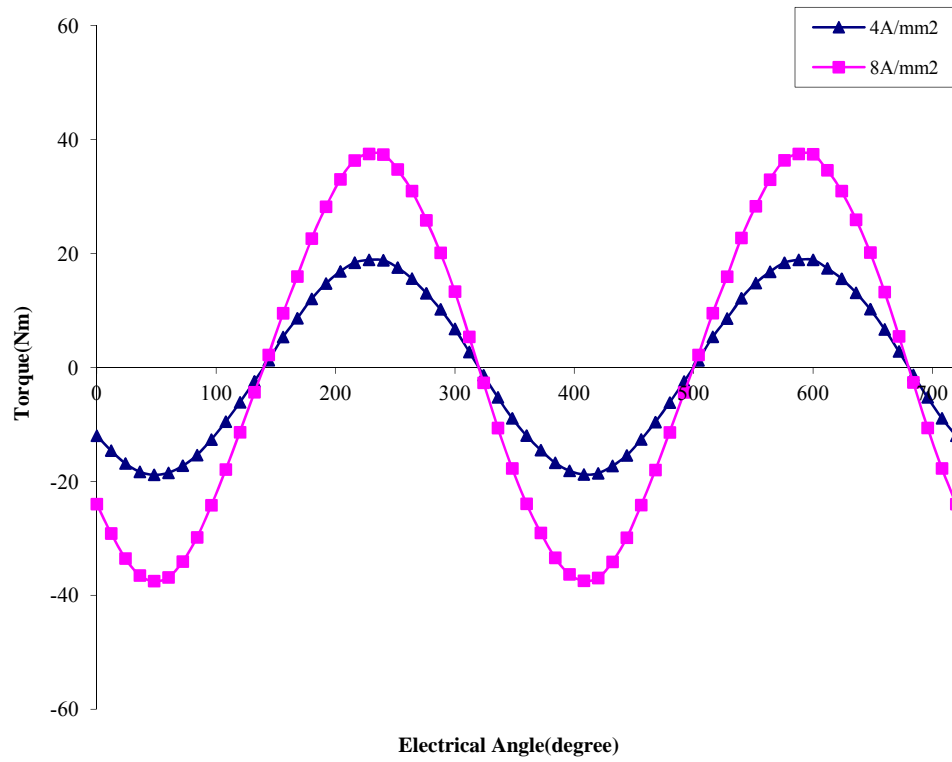


Fig. 2.21 Electromagnetic torque when DC current injected of 3-phase PM machine with concentrated winding.

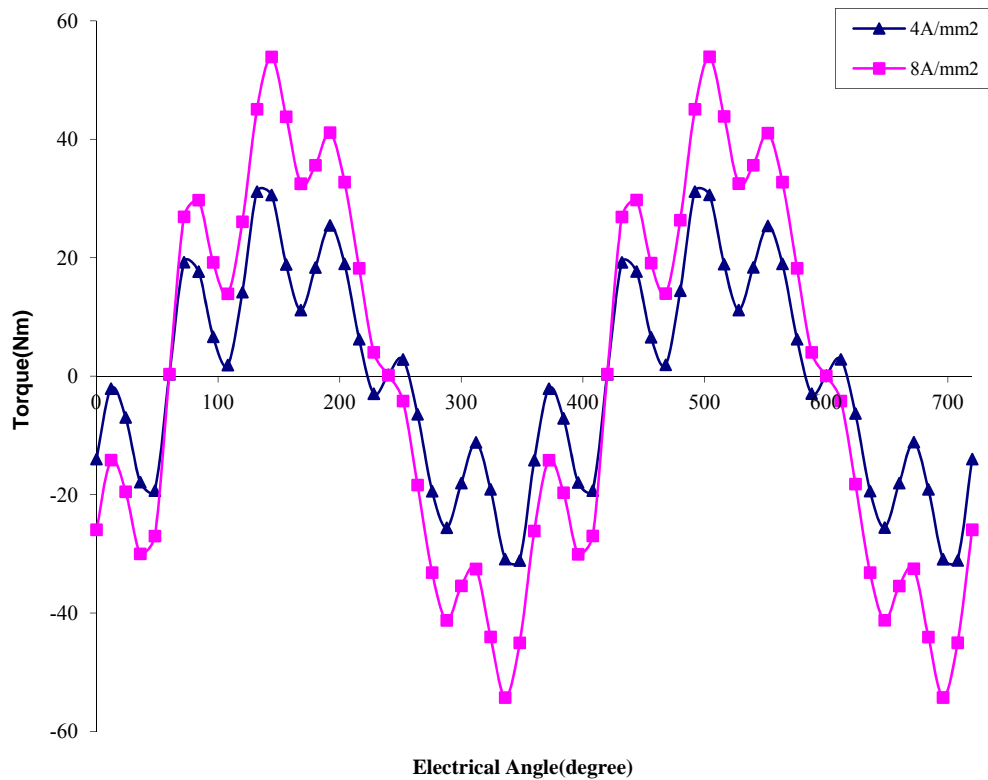


Fig. 2.22 Electromagnetic torque when DC current injected of 3-phase PM machine with distributed winding, 1 slot per pole per phase.

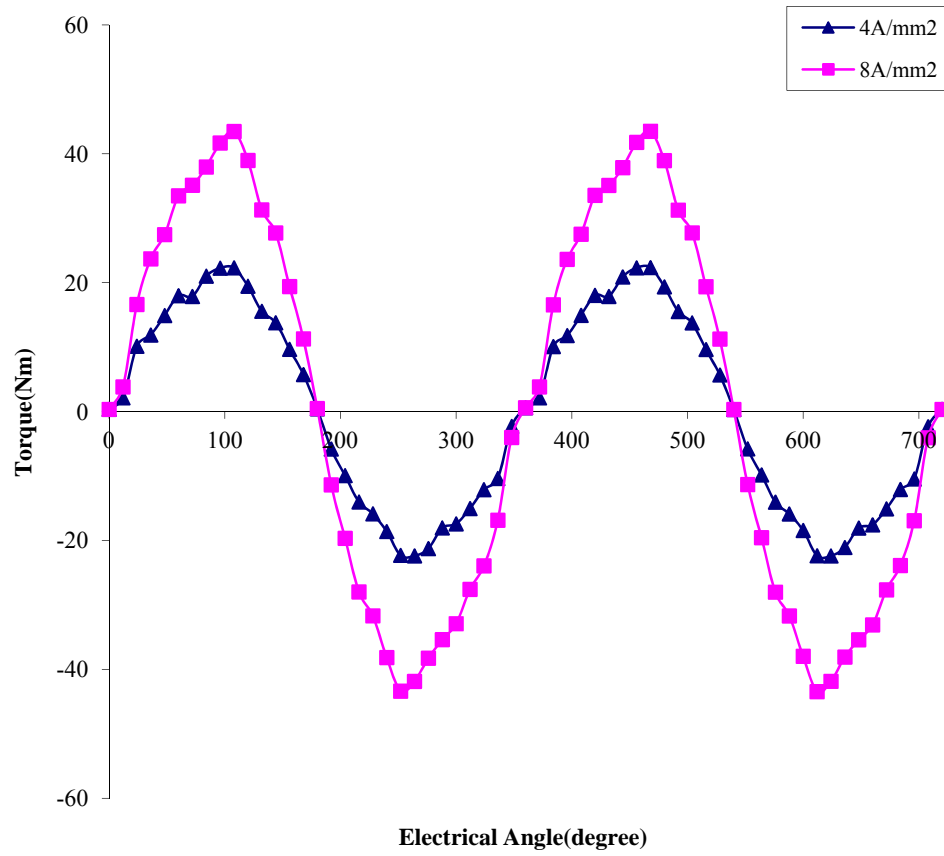


Fig. 2.23 Electromagnetic torque when DC current injected of 3-phase PM machine with distributed winding, 2 slots per pole per phase.

## 2.3 Summary

The search coil test verified the simulation tool FEMM experimentally. It provided a realizable method in machine designing, and most importantly for this research, which was mainly focused on the model simulation. More functions of FEMM would be introduced in next Chapter.

Comparing the charts for flux change per step, the flux change or back EMF for short pitched concentrated winding was smooth and sinusoidal. On the contrary, the waveforms of full pitched distributed winding machine were not smooth. In machine drive point of view, the short pitched and concentrated winding design may bring some benefits to the drive circuit. However, the higher peak value and density of back EMF from distributed winding machine, which led to higher peak power and average power density in driving mode, would benefit the machine output performance and may make this type of machine a better option for traction applications. The reluctance torque when DC current injected were shown in torque plots. The shoulders of peak torque or torque pulsation of distributed winding machines may be caused by their full pitched winding design. For the multi-slot design, 2 slots per pole per phase in this case, the torque pulsation was significantly reduced with the penalty of decreasing in torque. Note that, the peak torque of 2 slots per pole per phase distributed winding machine was still higher than the concentrated winding machine.

The concentrated winding was selected for rest of this thesis, because it produce less torque ripple which would benefit the traction application. The trapezoidal emf waveform could be separated into few sinusoidal signal and the power electronics to created the signal would be simplified.

## **CHAPTER 3**

# **ANALYSIS OF MACHINE MODEL WITH DIFFERENT MAGNET DESIGNS**

### **3.1 Introduction**

Characteristic of an electric machine would be defined by its design parameters, i.e. magnet shape, tooth design and winding. In this section, machine models with different designs were simulated using MATLAB based on Lua scripts used in last section. The electromagnetic capability would be investigated in detail after data of each model were collected. Based on the simulation results, different designs were compared and also ideal torque output was estimated for bread-loaf machine design.

### **3.2 Different rotor topologies**

For a Permanent-magnet (PM) machine, the design of the magnets which were mounted on the rotor has significant impact on the electrical characteristic, e.g. cogging torque, EMF, and demagnetization of the magnet. In this section, five different rotor magnet designs were investigated and the capability of each machine which implements those designs was simulated using FEMM. For more sensible comparison concerns, the stator design was maintain the same and also the size of the magnet was kept relatively the same as bread loaf design. These five rotor magnet designs were shown in Fig. 3.1, which were:

- (a) Bread loaf design
- (b) Embedded bread loaf design
- (c) Full pole-pitch and flat bottom design
- (d) Full pole-pitch arc design
- (e) Full pole-pitch pseudo-radial magnetization design

About these rotor magnet designs, experiments established for first two sets were focus on the steel saturation and magnet demagnetization and the studies for the other three sets were laid stressed on possible performance gain or benefit in electromagnetic aspect by changing the shape of magnets. The bread loaf design was existing PM machine design, which has high current-torque capability and the rotor magnets used in design (a) and (b)

were the same and the only difference between them were how they implement on the rotor: surface mount for design (a) and insert for design (b). The difference among design (a), (b) and (c) was material filled between magnets, i.e. (a) filled with air, (b) filled with iron and (c) filled with magnet. Design (d) and (e) were developed by preserving air gap and the average depth of magnet in design (c) was used as magnet depth. The cross section area and volume of the magnet for each design were shown in Table 3.1. More mechanical strength would be provided by slotting the magnet onto the rotor and difference in electrical capability would be studied in later section. The volume of the magnets in the other three designs were slightly larger than bread loaf shape, hence there was possible emf and torque gain and it would be verified. The fixed magnetization direction of the full arc design would produce more trapezoidal looking emf. The torque performance and the impacts in electromagnetic aspect by changing the fixed magnetization direction of the magnets to radial would be discussed.

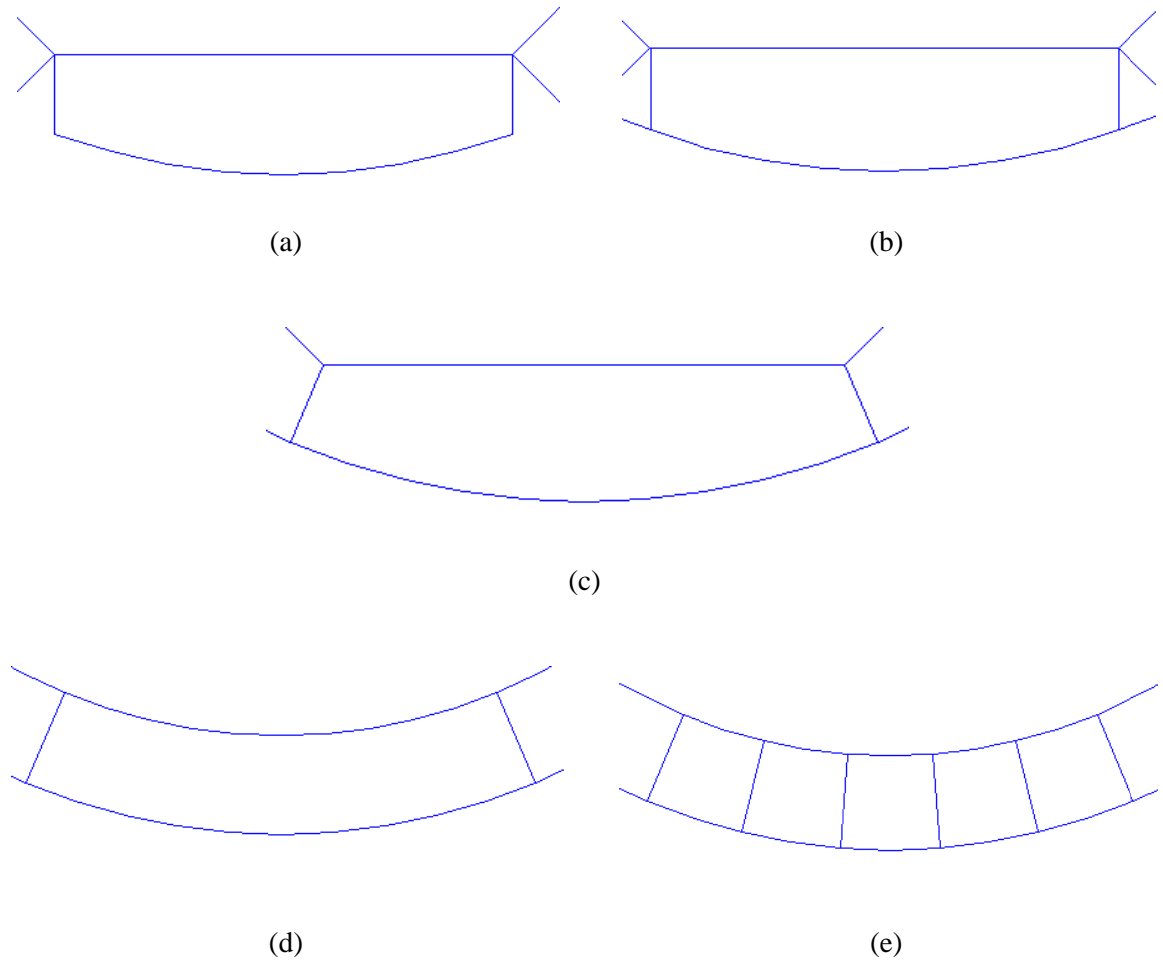


Fig. 3.1 Rotor magnet designs: (a) Bread loaf magnet design; (b) Embedded bread loaf design; (c) Full pole-pitch and flat bottom design; (d) Full pole-pitch arc design; (e) Full pole-pitch pseudo-radial magnetization design.

Magnet designs	Cross section area ( $10^{-4} \text{ m}^2$ )	Volum ( $10^{-5} \text{ m}^3$ )
Bread loaf	2.16909	1.24077
Full-/pole-pitch flat bottom	2.27148	1.29929
Full pole-pitch arc	2.17675	1.2451

Table 3.1 Magnet size for different magnet designs.

### 3.2.1 Mechanical design of magnet

All the model designs were drawing in AutoCAD and exported to FEMM afterwards for later modification and examination. Based on the bread loaf magnet design, shown in Fig. 3.1 (a), the different magnet designs were modified using AutoCAD. For inserted magnet design, a circle centred as the magnets with radius of 45 mm was added to the drawing to replace top surface of the magnets, to maintain similar air gap with other designs. After imported into FEMM, The material between the magnets was assigned as Transil steel, which was same as the material of the rotor. The analysis was executed to check any unsigned blocks in the model and the magnetic field solution of the model was useful to spot any mistakes made in assigning material. Separating the gap between magnets into two equally parts and merging with the magnet next to the gap respectively, the design of full/pole-pitch and flat bottom magnet, shown in Fig 3.1 (c), was completed and noted that the total volume of the magnet was slightly larger than bread loaf shape magnet and this would be counted into discussion when comparison was made. Although for all models, the main objective was to investigate the impacts due to shape change, the volume of the magnet was roughly maintain the same to avoid significant effect due to size of the magnet. The sizes of magnet were shown in Table 3.1. Hence, for the full pole-pitch arc design, shown in Fig 3.1 (d), the depth of magnet was set as average depth of bread-loaf shape magnet. After dividing full pole-pitch magnet into five equality segments, radial distribute angles were assign to magnet blocks to create a unified radial magnetization direction for magnets for full pole-pitch pseudo-radial magnetization design, shown in Fig. 3.1 (e).



### 3.2.2 Model defining and software program

Using the code to execute the model simulation was essential to this research, hence the development of program code was with extra caution. Even the functions were stated in FEMM manual, it was necessary to verify before applying into the code. Therefore, each step of code development was kept as simple as possible and only the verified function or code segment was implemented into simulation program. This iterative process helped to reduce the possibility of error and dysfunction of simulation program. In this section, the execution effect of the functions, which would be using in simulation program, were tested by creating several small testing programs and every individual function was tested manually in FEMM before embedding into program code. To verify the coding functions, the outcomes of the code were comparing with manual operation using commands in pull-down list in FEMM window. The iterative process of function verification was shown in Fig. 3.2 and the also the functions could be tested in ‘Lua Console’ in FEMM preliminary.

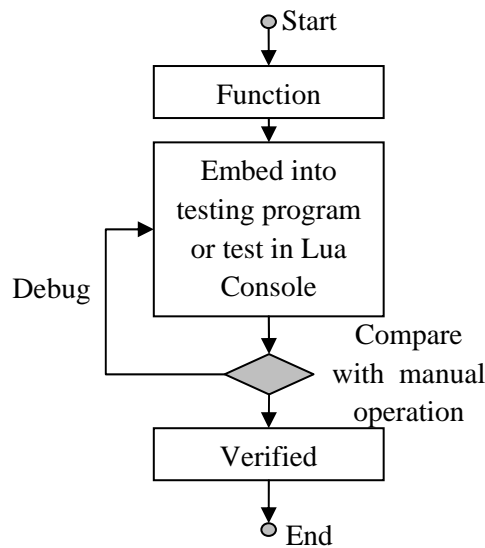


Fig. 3.2 Iterative process of coding function verification.

Many useful measurement features were included in FEMM, which were mainly functional in FEMM result file. These features could be separated into two types: line integral and area integral, depending on the object selected. With the depth of a model and 2-D layout in FEMM, a line would be associated with a cross section and an area would be a block. To be convenient, a name called block integral was used instead of area integral. For constrain with this research, line integral was used to obtain flux across an area and torque on surface. The block integral was used extract information of total flux, cross section area,

block volume and current. The flux measurement was the most important for this research among these features and it would be described in detail here. The flux measurement in FEMM, for both line and block integral, was calculated based on the nodes information. The bread-loaf magnet model with nine tooth was used as example in this section and the data were extract when the model was open circuit. By selecting two points on the model in ‘segment selecting mode’, shown in Fig. 3.3(a), a red line was created across a tooth of the model.

It was straight forward to obtain measurement in results file by using ‘Integrate’ command from pull-down menu in FEMM and the flux would be returned when the appropriated mode was selected. Examples of flux information extracting dialog was shown in Fig. 3.3(b), for both line integral and block integral. Although the integral function in FEMM would satisfied in most cases, individual mesh element would be examined for some particular cases in this research, i.e. demagnetization and iron saturation. An area of the model was zoomed in and filled with meshes, shown in Fig. 3.3(c). The bold triangle ‘ $ijk$ ’ was one mesh element in the block and the coordinate,  $(X,Y)$ , and vector potential,  $A$ , for each node could be obtained from the result file. The calculation for flux in line integral was based on the nodes, which was on the line or nearest to the selected line. The flux or flux density,  $B$ , for this case, calculation for a block was derived by summing up the flux density for each mesh element within the block and the flux density for each mesh element could be calculated by using Eq. 3.1 - Eq. 3.3 [6, 13], where  $A_v$  was the area of a mesh element,  $A$  was vector potential and  $X$  and  $Y$  were coordinate of a node. The implement of these equations would be introduced when demagnetization and saturation of material was involved.

$$B_x = \left[ \frac{(X_k - X_j) A_i}{2A_v} \right] + \left[ \frac{(X_i - X_k) A_j}{2A_v} \right] + \left[ \frac{(X_j - X_i) A_k}{2A_v} \right] \quad (3.1)$$

$$B_y = \left[ \frac{(Y_j - Y_k) A_i}{2A_v} \right] + \left[ \frac{(Y_k - Y_i) A_j}{2A_v} \right] + \left[ \frac{(Y_i - Y_j) A_k}{2A_v} \right] \quad (3.2)$$

$$|B| = \sqrt{B_x^2 + B_y^2} \quad (3.3)$$

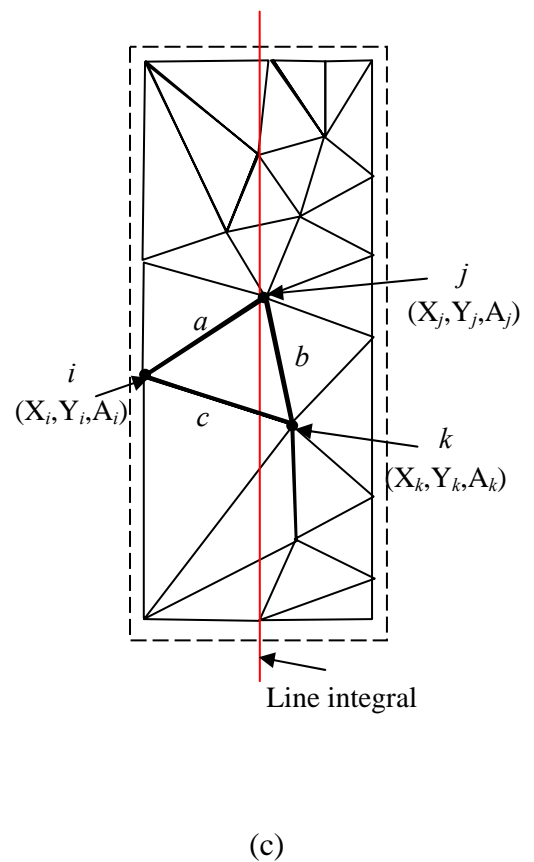
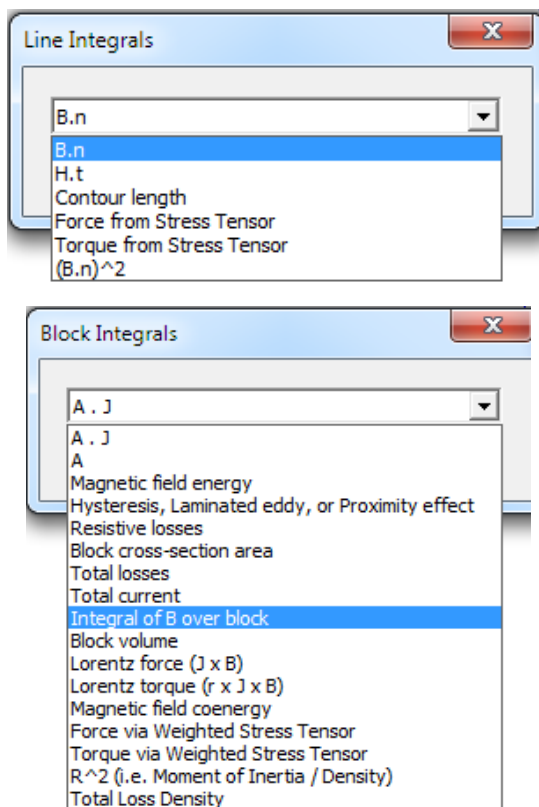
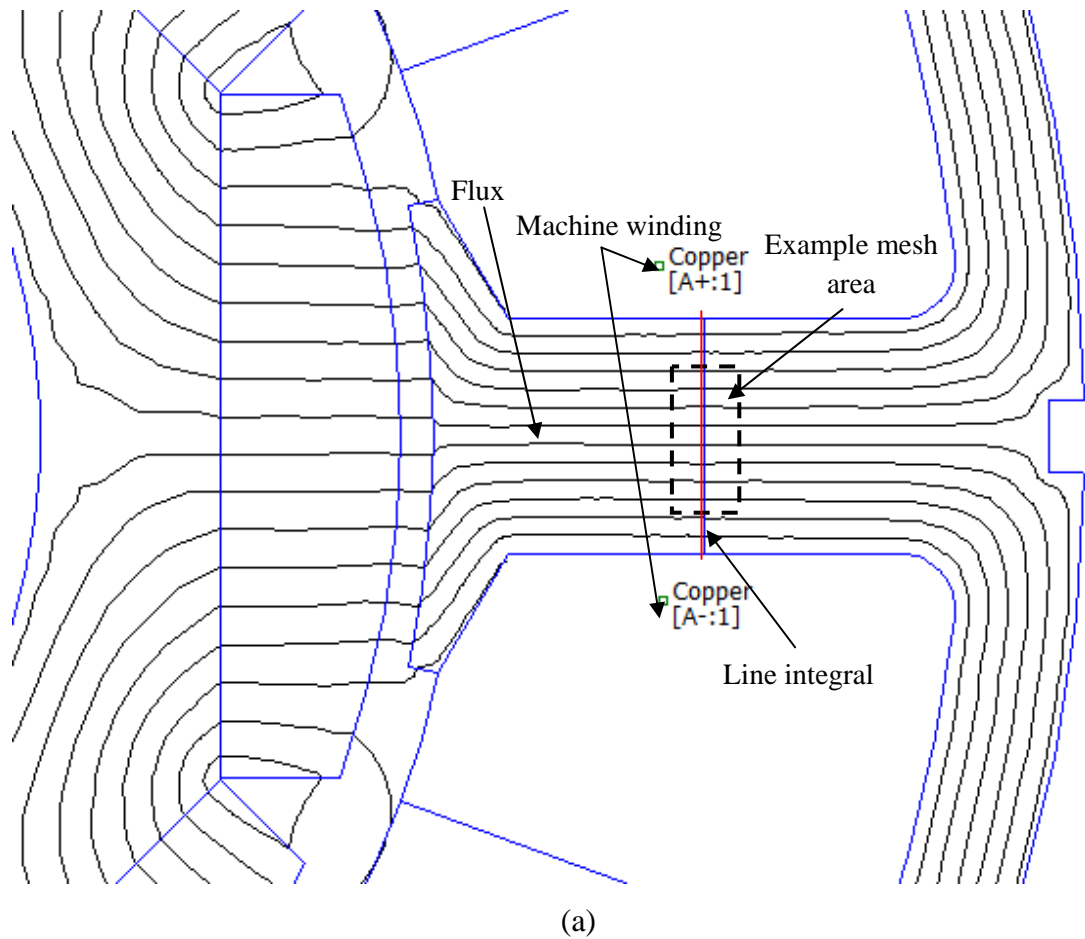


Fig. 3.3 An example of line selecting in result file (a), measurement extracting dialog (b) and detail mesh element for an area on tooth.

When the model became complicated, the manually operation in FEMM window could not meet the requirement of speed and convenience for model analysis. Therefore, the automatic analyzing program was developed both in Lua script and MATLAB. The analyzing program was a combination of Lua script and MATLAB function. The brief structure and file handling for each part was shown in Fig. 3.4. The model simulation was controlled by ‘Lua script’ and the result file generated by FEMM, based on the model file, was the data source for the rest of process. The data were analyzed and recorded by MATLAB and Excel was employed demonstrating the result.

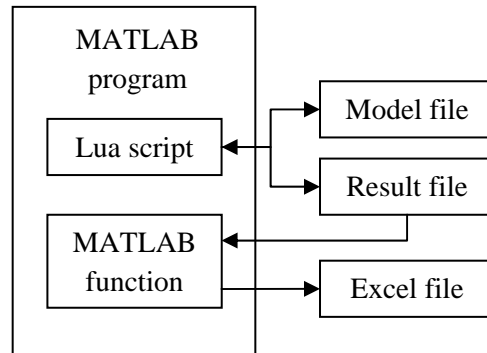


Fig. 3.4 Brief function of a MATLAB code.

The Lua script was substitution for manual operation in FEMM and it was consisted of varies commands, which would be called FEMM function here. The script was execute line by line from top to bottom. Hence FEMM was not a sequence-tolerated program, especially in object selecting, and the commands should be executed in precise sequence or it would cause faulty in program and unexpected behaviours. A Lua script could achieve most of the requirement for this research, which were including model modifying, model analyzing, data extracting and data recording. Comparing to collect data from each static solution manually, the automatic code would accelerate the speed of data collection dramatically.

### 3.3 Electromagnetic capability

#### 3.3.1 Cogging torque

Cogging torque was existed in permanent magnet (PM) machine, due to the air gap between magnets, pole pair and stator slot design. It was a periodical torque ripples when PM machine rotated without any current supplied and it has more significant impact at low

speed, so PM machine designs were trying to minimize it. In this section, the cogging torque was investigated for each magnet design for both 3-phase winding and 9-phase winding model when it was open circuit, i.e. no current flow in machine winding. The dimension of machine was shown in Table 3.2. The winding schematic used for this task was concentrated 3-phase winding and STAR connection for end winding, which was for 3-phase machine model and shown in Fig. 3.5 (a) and the STAR connection of winding circuit was shown in Fig. 3.5 (b). Although circuit definition was not required for extracting data in an open circuit model, i.e. no current flow in winding, the circuits or windings were defined in machine model for usage of commands for data extraction.

The end winding was Star connection for this 3-phase model, hence the phase difference between each phase was 120 degree and the direction of current flowing was opposite current flowing out. Current flowing into the circuit could be expressed as:

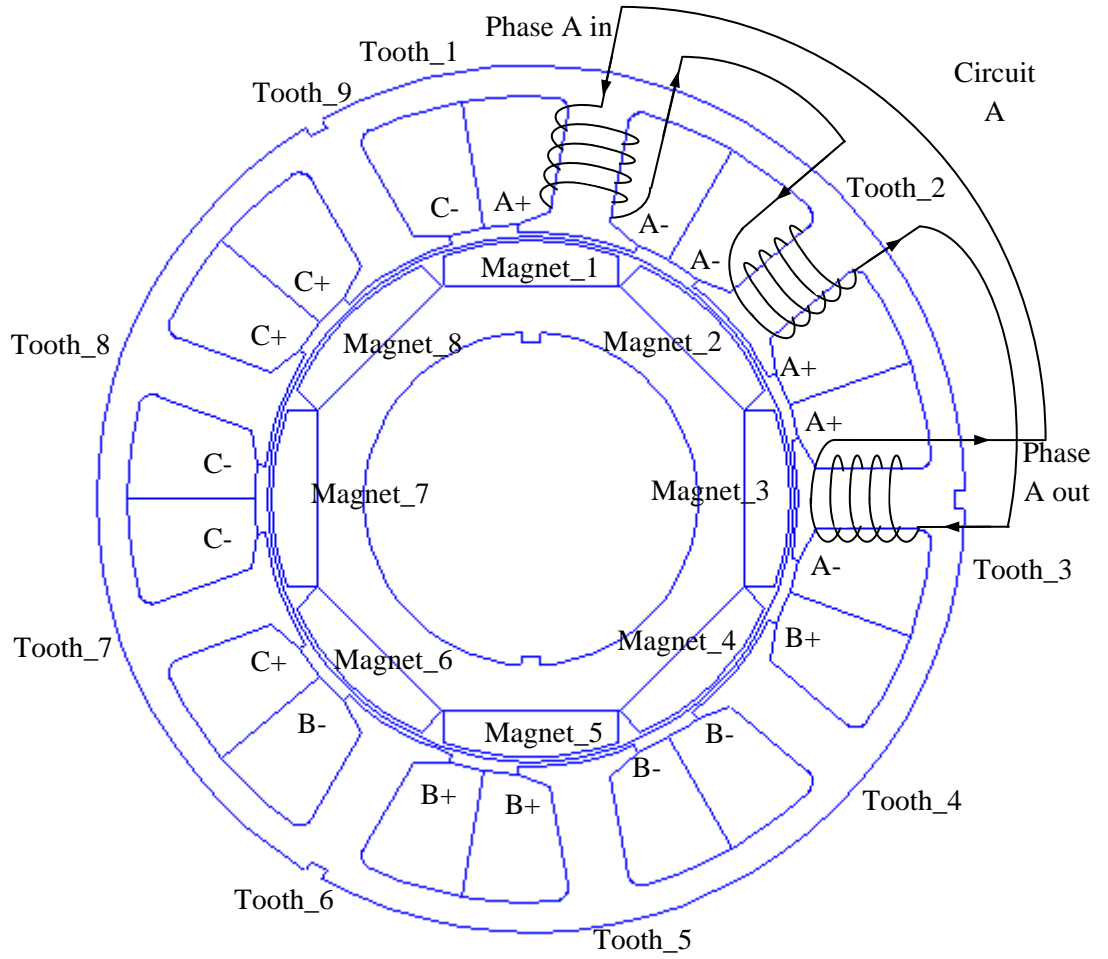
$$current\ A_{in} = current\ peak \cdot \cos(0^\circ), \quad (3.4)$$

$$current\ B_{in} = current\ peak \cdot \cos(0^\circ + 120^\circ), \quad (3.5)$$

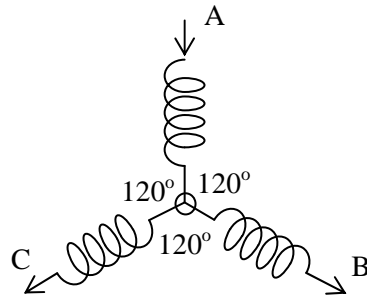
$$current\ C_{in} = current\ peak \cdot \cos(0^\circ - 120^\circ). \quad (3.6)$$

Rotor radius (m)	0.045
stator radius (m)	0.0755
Air gap (m)	0.0015
Axial length (m)	0.0572
Slot area (m <sup>2</sup> )	0.00066713

Table 3.2 Machine model dimensions.



(a)



(b)

Fig. 3.5 Winding schematic for 3-phase machine model (a) and STAR connect of the circuits (b).

To investigate the characteristic of a machine, data for a full electrical cycle were required. An electrical cycle was not necessary to be a mechanical revolution, i.e. rotating *360 degree*. In this research, the models were all symmetric and the electrical cycle was defined by pole pairs of magnet. Using model shown in Fig. 3.5 (a) as example, there were eight magnet mounted on the rotor of the machine. For a permanent magnet (PM), the magnetic

field direction it provided was depending on the way it was magnetized during manufacture process. This direction was called magnetization direction and there were two directions for this example: radial toward to the centre and radial outwards. A magnet pole pair was consisted with two magnets with opposite magnetization direction and next to each other. There was four pole pairs in this case and it took a quarter of revolution for machine model to complete a electrical cycle. Also electrical cycle would be repeat another three times if the model was rotated full revolution. Hence the relation between mechanical angle and electrical angle could be described by using a equation:

$$\text{mechanical angle} = \frac{\text{electrical angle}}{\text{pole pair}}, \quad (3.7)$$

where the unit for angle was *degree*. As mentioned in Lua script development, the angular displacement was mechanical angle for each step and in order to collect data in detail, the decision on precision of electrical step was required in program development. If precision of *1 degree* in electrical angle was chosen for a simulation, the actual angle displacement would be could *0.25 degree* mechanically and the total number of steps for a full electrical cycle should be:

$$\text{Total step} = \frac{\text{angle in full electrical cycle}}{\text{precision}} = \frac{360}{1} = 360 (\text{steps}), \quad (3.8)$$

where the unit for angle was *degree* and *degree per step* for precision. To implement the full electrical cycle into MATLAB program, the upper limit of loop number was set to the total number of steps. Sometimes, the rotation was extended to more than one electrical cycle for collecting extra data and data manipulation purpose.

Data of torque could be obtained using line integral and the difference from flux linkage extracting was the function mode. Instead selecting a line crossing a tooth, the contour in the middle of the airgap was selected. Due to the magnitude of cogging torque, extra contours were created surrounding measuring contour, to force the mesh size to be unified and hence torque data could be more accurate. The simulation program for extracting torque information when it was open circuit was development by combining full cycle loop control and other functions mentioned earlier in MATLAB and the flow chart was similar with the one shown in Fig. II.10 (a). By change the target model files, cogging torque data was collect for five models with different magnet designs which was shown in Fig. 3.1. Since the simulation condition was open circuit, other useful data were also collected, i.e. flux linkage in each tooth. The tooth on stator was names from one to nine, shown in Fig. II.10 (a).As more variables were required to be recorded for each loop, the length of row

of universal array was expended to fit the size of local array. An example result of bread loaf magnet design was partially shown in Table 3.3 and the structure of universal array also could be observed. The cogging torque was low as expected. The full results were included in Appendix III.1 and the codes used to obtain data in open circuit for bread loaf magnet design were shown in Appendix I.5.

Time (step)	electrical angle (degree)	Cogging torque (Nm)	Mechanical angle (degree)	Flux linkage tooth 1 (Wb)	Flux linkage tooth 2 (Wb)	...
0	0	-0.00649991	0	0.000955	-0.001100	...
1	1	0.024888965	-0.25	0.000967	-0.001104	...
2	2	0.053541741	-0.5	0.000978	-0.001107	...
3	3	0.06363916	-0.75	0.000988	-0.001110	...
4	4	0.073286601	-1	0.000998	-0.001113	...
5	5	0.073212102	-1.25	0.001008	-0.001116	...
:	:	:	:	:	:	:

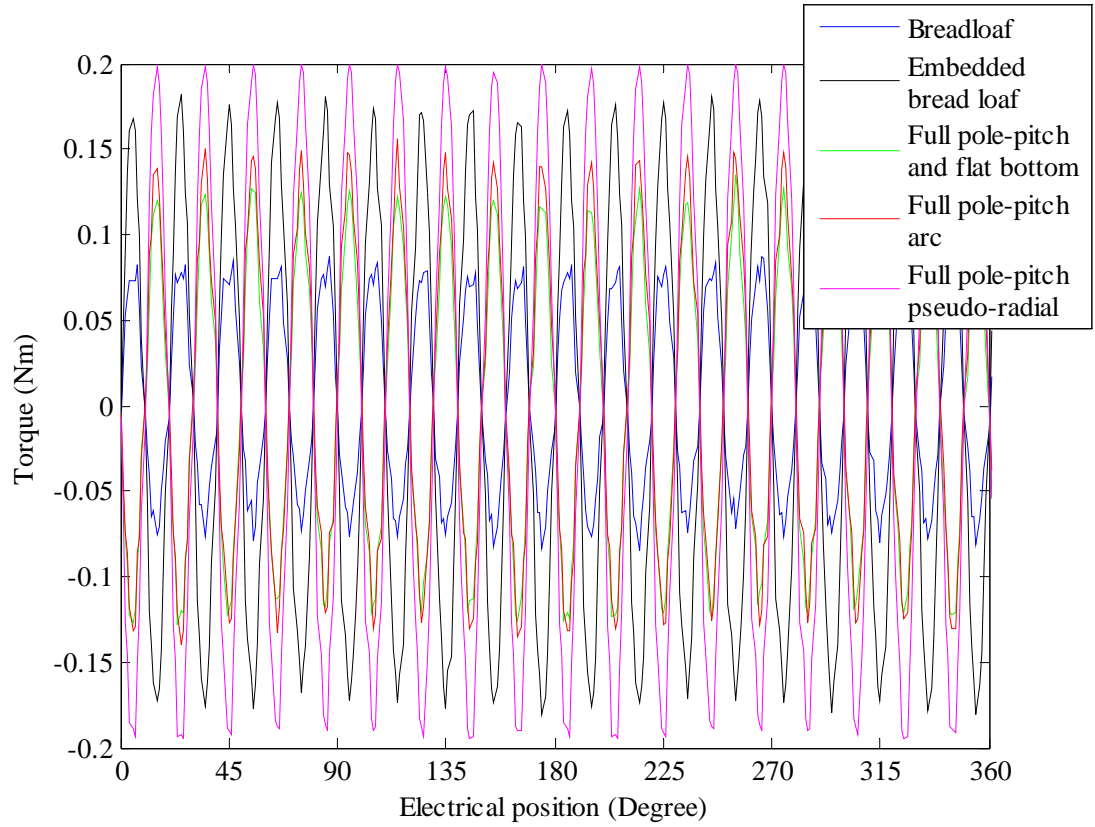
Table 3.3 Part of result recorded in cogging torque test.

The cogging data was process using MATLAB and the they were plotted on the same diagram for a full electrical cycle, shown in Fig. 3.6 (a). Since the data stored separately for different magnet designs, a piece of MATLAB code was programmed for manipulating the data, i.e. merge desired data into one data file. There were also some other functions in this code and it was employed to process the data in torque capability section, which would be introduced later in detail. The full data process program was shown in Appendix I.6. For easier comparison, a zoomed in version of cogging torque were plotted in Fig. 3.6 (b), which shown the detail of cogging torque from  $0$  to  $60$  degree in electrical angle. The magnitude of the cogging torque for different magnet design was less than  $0.2$  Nm and the cogging torque from bread loaf design was lowest.

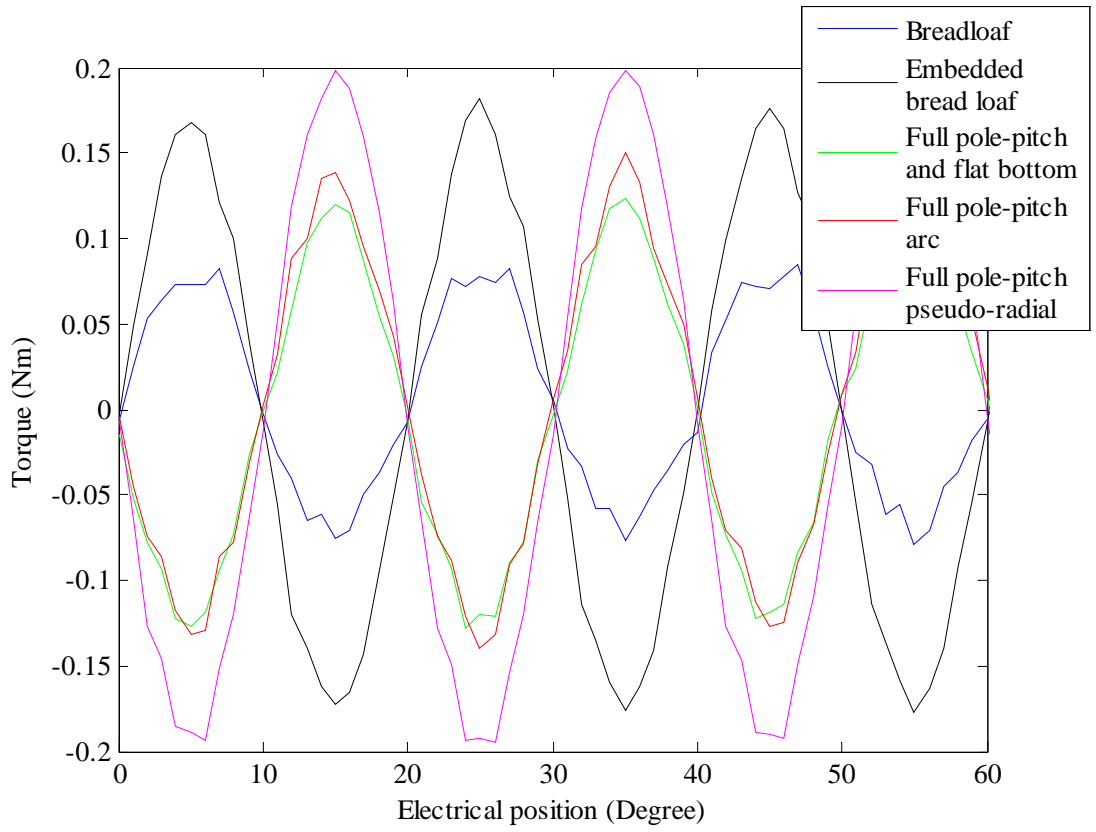
The lowest cogging torque means Bread loaf magnet design was the best design in term of cogging torque, among five magnet designs. Filling the gap between magnets with steel would have significant impact on cogging torque: doubled the value of the non-filling one. To state in another way, the embedded bread loaf design produced more cogging than bread loaf design. The radius distributed magnetization would not reduce the cogging torque, according to the results. It produced the largest cogging torque in this test, nearly two times more than bread loaf design. Two full pole-pitch designs shared similar performance in cogging torque and slightly less from flat bottom magnet design. All five designs shared same electrical period which was  $20$  degree. The shape of waveform at the



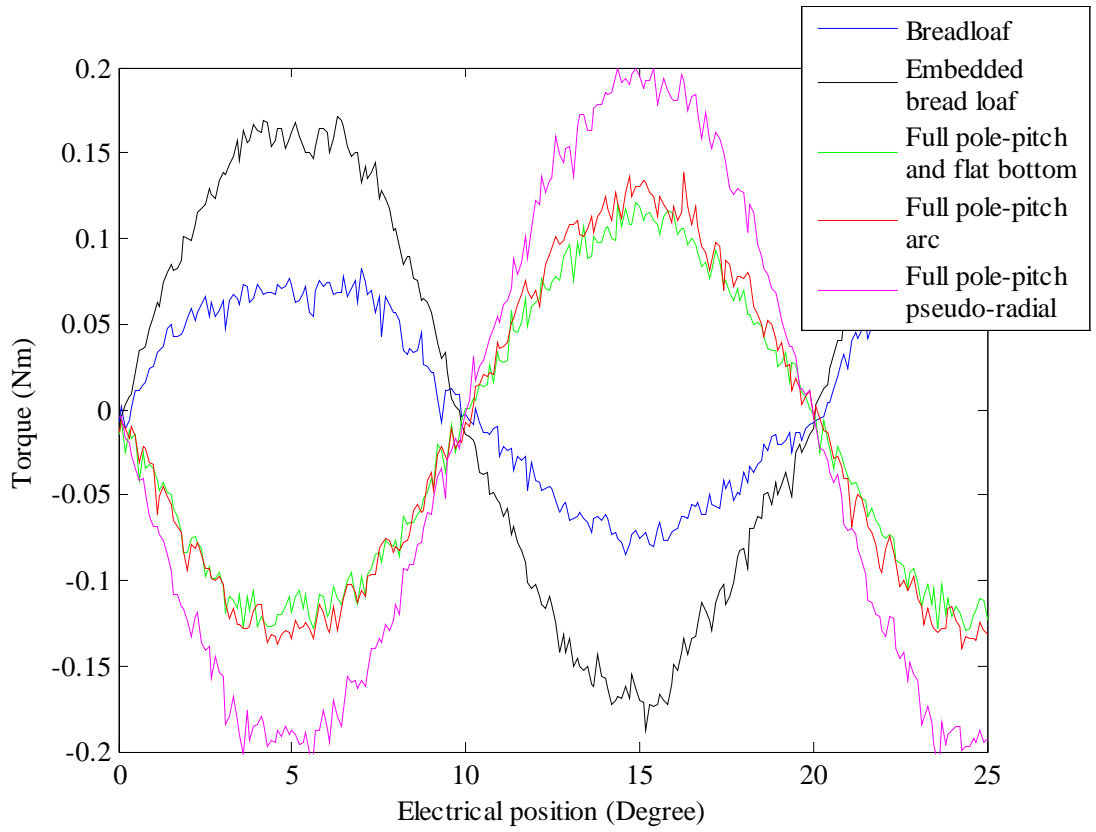
peaks was not consistent, especially at negative peak and this may be caused by lack of data points in the area. Hence another simulation was taken place with finer precision and the results were shown in Fig. 3.6 (c). With more data points, the shape of cogging torque waveforms were improved and the magnitude of both positive and negative peak was matched. Also, the period of cogging torque were the same for all designs as expected, which was because magnet pole pair and stator tooth were maintained the same.



(a)



(b)



(c)

Fig. 3.6 Plots for cogging torque: (a) cogging torque for one electric cycle; (b) zoomed in cogging torque and (c) cogging torque plot with fine precision.

### 3.3.2 Open circuit EMF

When a PM machine was rotated by external force, there would be current generated in winding. This current was caused by the motion of winding, i.e. conductor moving in electrical field in perpendicular direction and referring to Faraday's law The induced voltage was called back Electromotive Force (EMF). Back EMF generated in winding circuit of a PM machine could indicate the coincident voltage required when the same machine was desired to be driven. As an important characteristic of a PM machine, phase back EMF was investigated and compared across different magnet designs. By using flux data extracted from open circuit simulation in the section of investigating cogging torque and also recalling the equation for calculating back EMF in Chapter 2:

$$e = N \frac{d\phi}{dt} = N \left( \frac{d\phi}{d\theta} \right) \frac{d\theta}{dt} = N \omega_r \frac{d\phi}{d\theta}, \quad (3.9)$$

where  $e$  was the back EMF with unit Volt;  $N$  was the number of turns of the coil;  $\phi$  was the magnetic flux per coil in tooth and unit was  $Wb$ ;  $\theta$  was the mechanical position of the coil or tooth and unit was *radian (rad)*;  $\omega_r$  was the mechanical speed of the coil or tooth and unit was *radian per second (rad/s)*.

With data collected during open circuit test, flux linkage in each tooth for different design could be gathered in one data file by using a piece of MATLAB code, which was shown in Appendix I.7. The calculating process for EMF was included in this data analyzing program. During data analyzing process, universal array from multiple locations was loaded and flux information were pulled together with corresponding index which was indicate magnet design. By implementing the EMF equation, the EMF was derived from flux linking the coil. Since each circuit of a 9-phase model was wound on each tooth, the phase EMF was calculated by using flux linkage in a single tooth. For 3-phase model with concentrated winding shown in Fig. 3.5 (a), the flux linking phase winding would be:

$$\phi_{phase} = \phi_{tooth1} - \phi_{tooth2} + \phi_{tooth3}. \quad (3.10)$$

The subtraction for the second coil was because of the winding direction was opposite with other two. By substituting it into Eq. 3.7 and simplifying, the phase EMF for 3-phase winding could be calculated by summing up the EMF in each coil of a phase:

$$e_{phase} = e_{tooth1} - e_{tooth2} + e_{tooth3}. \quad (3.11)$$

To represent the results, plot function was employed in MATLAB. The flux linkage in single tooth was shown in Fig. 3.7 and the phase EMF for both 3-phase and 9-phase windings were shown in Fig. 3.8. In terms of flux linkage in a tooth, the waveforms shared similar sinusoidal shape for magnet designs. The full pole-pitch and flat bottom design was

the highest and this was caused by the largest total volume of the magnet among five designs. The bread loaf design achieved slightly high flux linkage than full pole-pitch arc design due to the similar size of magnet. The embedded design were performed worse than bread loaf design even both of them sharing the same magnet.

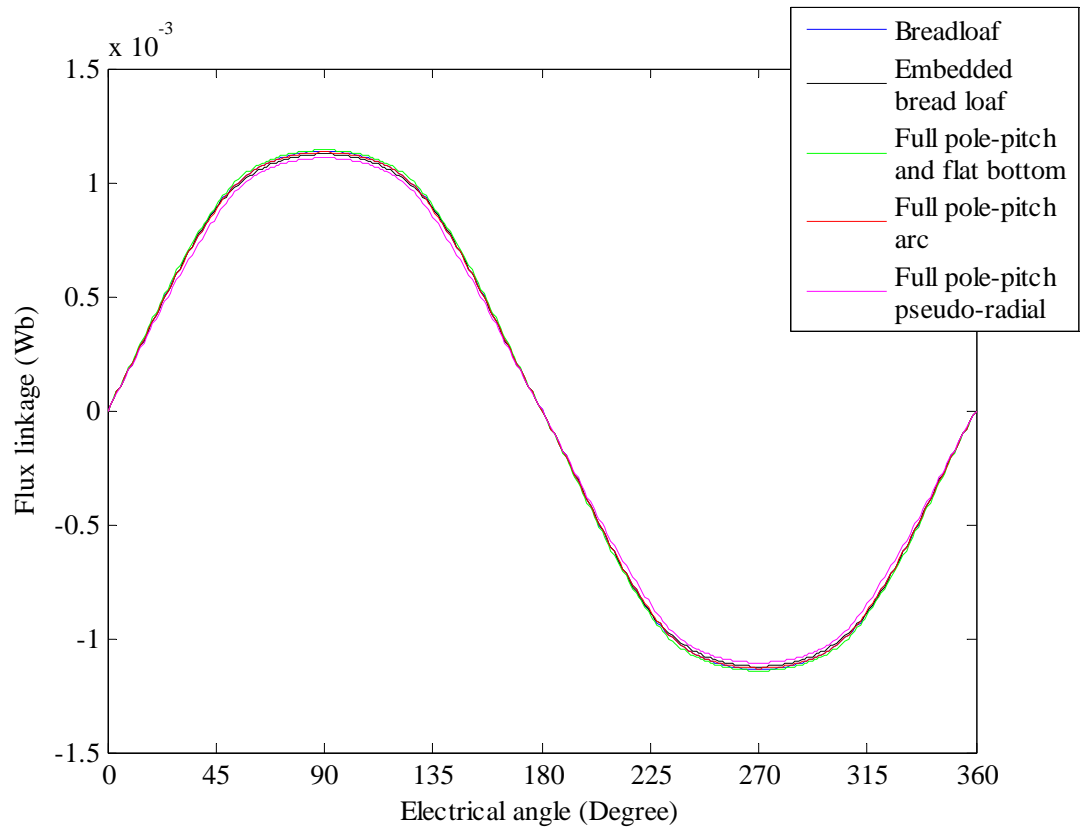


Fig. 3.7 Plot of flux linking the coil in one tooth for different magnet designs.

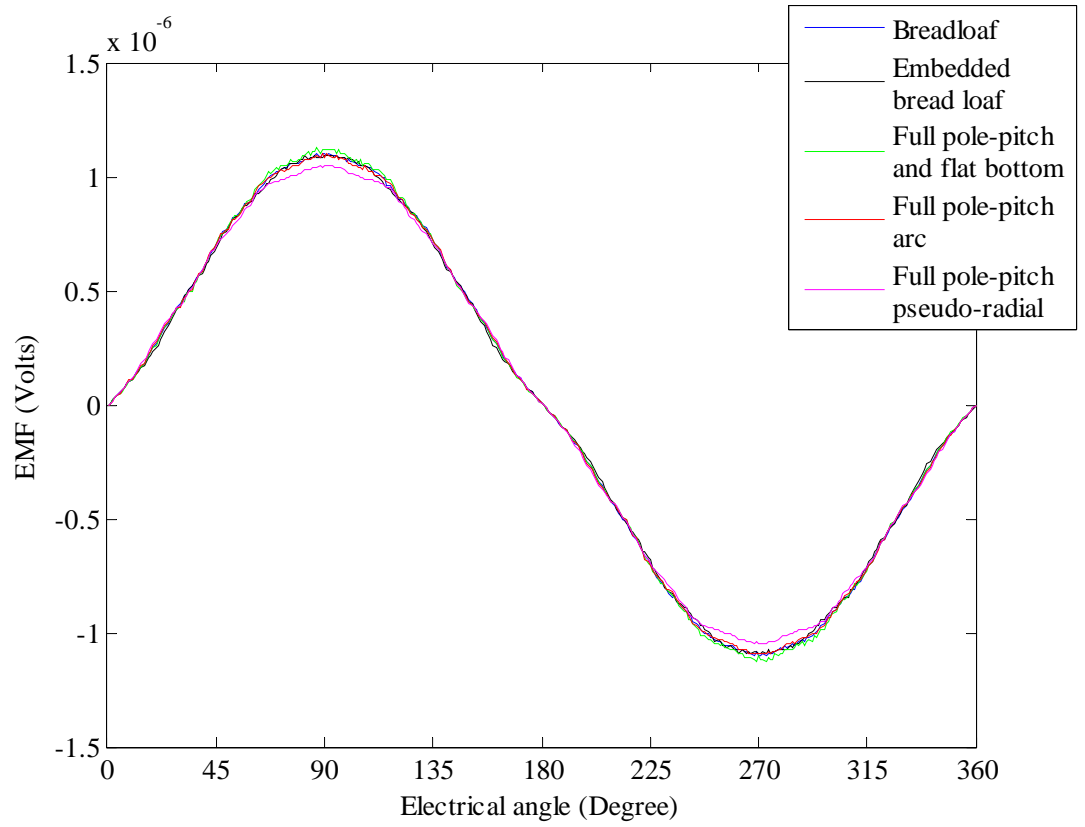
The difference in flux linkage was caused by the material between two magnets. Since steel was better conductor for flux than air, in embedded design, a small amount of flux could flow through the steel between magnets into rotor core, instead flowing through the airgap between rotor and stator tooth. There was noticeable less flux linkage for pole-pitch pseudo-radial design and it could be caused by its radial distributed demagnetization, which reduced amount of flux flowing through the airgap. The order of performance was inherited in EMF plot where the highest EMF was produced by full pole pitch and flat bottom design and full pole pitch pseudo-radial produced the lowest, for both 3-phase and 9-phase windings, because the EMF was related to the rate of change in flux linkage and the angular velocity was keep the same through the simulation, which was  $0.0175 \text{ Hz}$ . The similar EMF were produced by other three designs. The phase EMF waveform for 9-phase model was trapezoidal and it was sinusoidal for 3-phase model. The peak EMF of 3-phase model was three times much as 9-phase. To investigate for more detail, the frequency

analysis was taken place for EMF waveforms, by using Fast Fourier transform (FFT) in MATLAB.

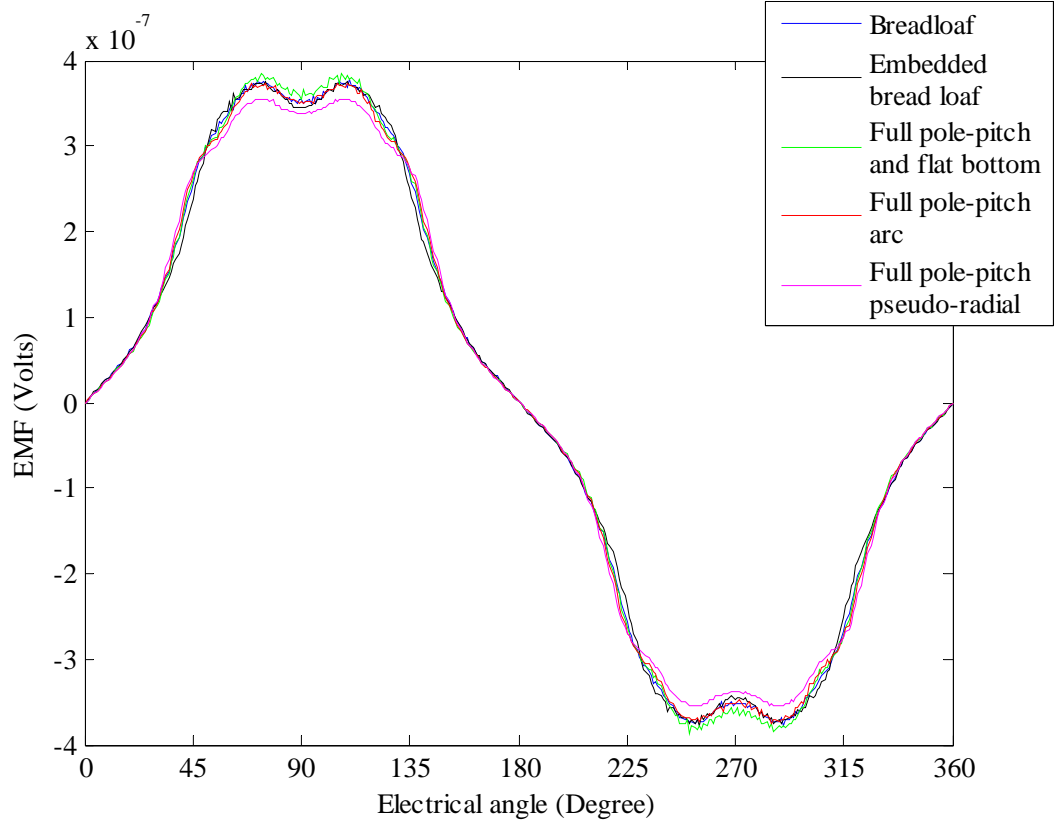
FFT function was widely used in discrete signal analysis and with its implementation, the time-domain problem could be analyzed in frequency-domain. A complex signal could be expressed as harmonic frequency spectrum and contribution of each frequency with proper program setup. The spectrum FFT function produced also indicated where power of the signal was spread at. In this research, the FFT function was employed to analyze the periodical waveforms and obtain the frequency related parameter and magnitude of each harmonic. These parameters could be used to simulate the original waveform for later analysis. Before implementing FFT function, a test was established to ensure it was functional as expected which would be described in detail in Chapter 4. By using analyzing program with FFT function embedded, the harmonic spectrums of EMF waveforms of different magnet designs were shown in Fig. 3.8 and tables which contained magnitude and phase angle in radian were shown in Fig. 3.19, Fig. 3.20 and Fig. 3.21, along with the table in which contained magnitude and phase shifting information. The flux and EMF analyzing program with FFT function was shown in Appendix I.7.

Comparing the spectrums of two designs using bread loaf magnet, the magnitude in bread loaf design was more concentrated in low order of harmonic and slightly more magnitude was spread towards high order of harmonic for embedded design. Generally speaking, the magnitude in higher order would have less impact on overall waveform, i.e. phase EMF in this case, because when the order of harmonic was high enough, the magnitude at that order would become noise which affects the system in negative way, i.e. creating heat. Similar distribution was observed in 9-phase winding and the amplitude of 5th order harmonic was similar to 3rd order harmonic, according to Fig. 3.11 (b) and Fig. 3.12 (a). In the situation of machine driving, higher order harmonic in voltage could introduce more complexity into driving circuit. Hence the bread loaf design was better than embedded design in terms of EMF. On the other hand, the loss in flux linkage and slightly higher harmonic in EMF may be compensated by the embedded design, which would provide more mechanical strength to hold the magnet on rotor.

In this part of research, the difference between different magnet designs was not significant, apart from Full pole-pitch pseudo-radial magnetization design. The amount of flux linkage and EMF were reduced with noticeable amount for radial magnetization. More comparison in performance for different magnet designs would be established in next section.

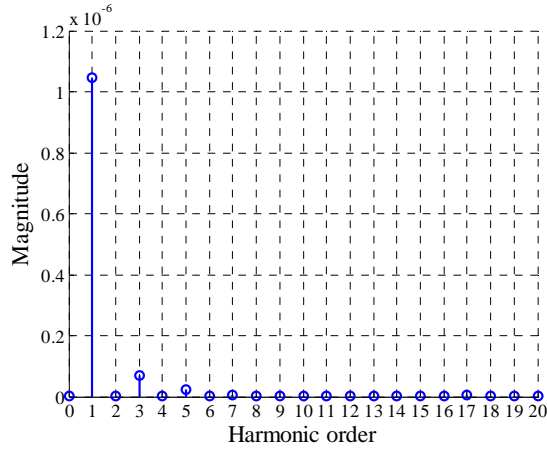


(a)



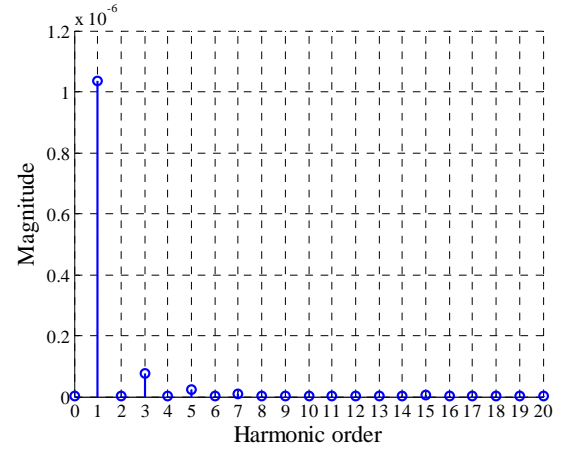
(b)

Fig. 3.8 Plot of phase EMF for different magnet designs: (a) 3-phase and (b) 9-phase.



Harmonic order	Amplitude (10-6)	Phase
0	0.000	3.142
1	1.044	-1.580
2	0.000	1.926
3	0.067	1.546
4	0.000	-0.767
5	0.023	1.524
6	0.000	-1.404
7	0.004	1.515
8	0.000	1.612
9	0.003	1.478
10	0.000	-1.600
11	0.002	1.475
12	0.000	3.047
13	0.002	1.465
14	0.000	0.364
15	0.001	1.411
16	0.000	-2.787
17	0.007	-1.725
18	0.000	0.084
19	0.001	1.075
20	0.000	2.964

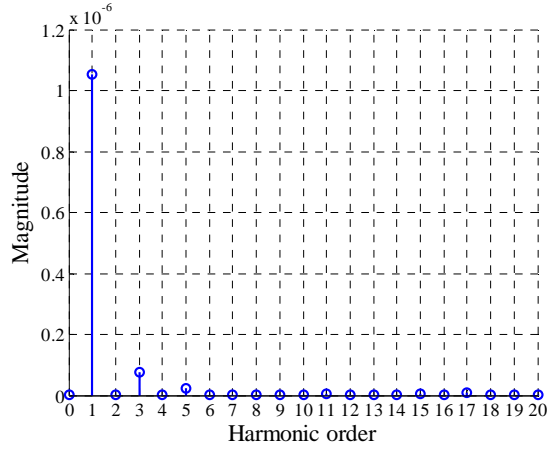
(a)



Harmonic order	Amplitude (10-6)	Phase
0	0.000	0.000
1	1.035	-1.580
2	0.000	2.815
3	0.074	1.546
4	0.000	-0.548
5	0.023	1.524
6	0.000	-0.953
7	0.008	1.510
8	0.000	1.298
9	0.002	1.483
10	0.000	-1.597
11	0.000	-1.698
12	0.000	1.905
13	0.002	1.476
14	0.000	2.192
15	0.007	-1.691
16	0.000	-1.454
17	0.003	-1.681
18	0.000	1.153
19	0.003	1.379
20	0.000	1.999

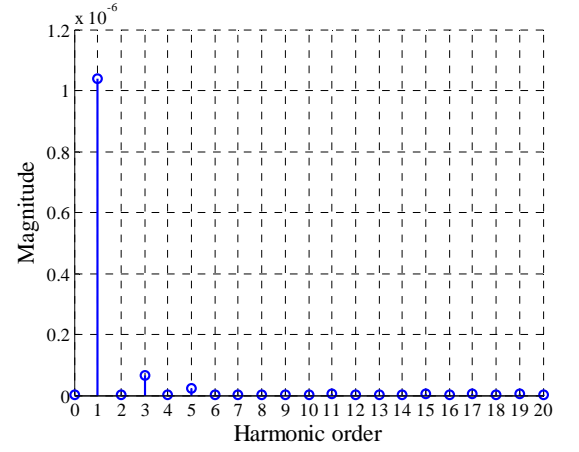
(b)

Fig. 3.9 Harmonic spectrum and its parameter table for different magnet design: (a) Bread loaf design in 3-phase model; (b) Embedded bread loaf design in 3-phase model. Phase in radians.



Harmonic order	Amplitude (10-6)	Phase
0	0.000	3.142
1	1.054	-1.580
2	0.000	-0.498
3	0.077	1.545
4	0.000	1.735
5	0.022	1.531
6	0.000	-2.159
7	0.003	1.470
8	0.000	0.363
9	0.003	1.533
10	0.000	2.311
11	0.004	1.464
12	0.000	-2.082
13	0.003	1.424
14	0.000	2.044
15	0.006	1.454
16	0.000	-0.803
17	0.007	-1.754
18	0.000	-2.198
19	0.002	-1.730
20	0.000	2.910

(a)

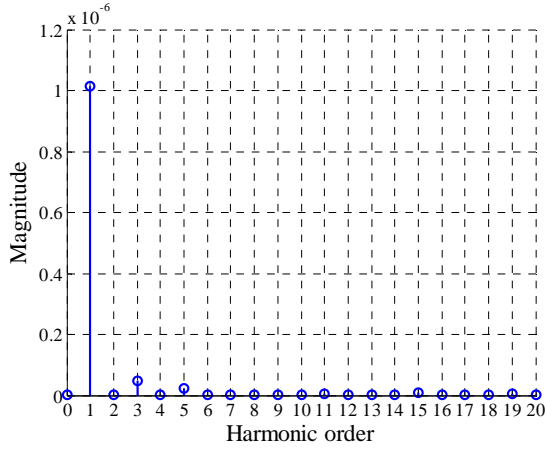


Harmonic order	Amplitude (10-6)	Phase
0	0.000	0.000
1	1.041	-1.580
2	0.000	1.109
3	0.065	1.546
4	0.000	0.629
5	0.023	1.529
6	0.000	-2.322
7	0.002	1.484
8	0.000	1.593
9	0.003	1.487
10	0.000	1.188
11	0.004	1.464
12	0.000	-1.303
13	0.002	1.477
14	0.000	1.617
15	0.007	1.444
16	0.000	-1.687
17	0.006	-1.756
18	0.000	1.047
19	0.004	-1.793
20	0.000	0.822

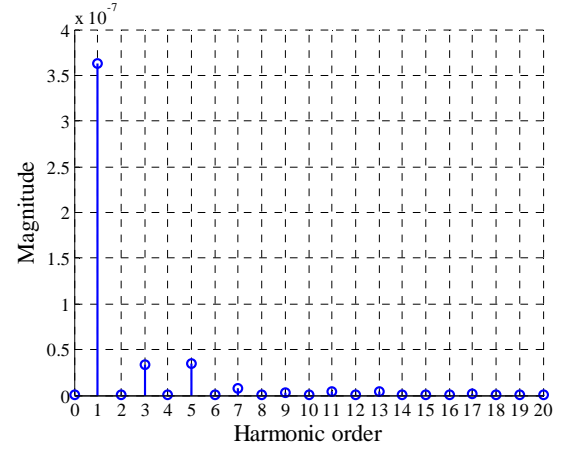
(b)

Fig. 3.10 Harmonic spectrum and its parameter table for different magnet design: (a) Full pole-pitch and flat bottom design in 3-phase; (b) Full pole-pitch arc design in 3-phase model. Phase in radians.



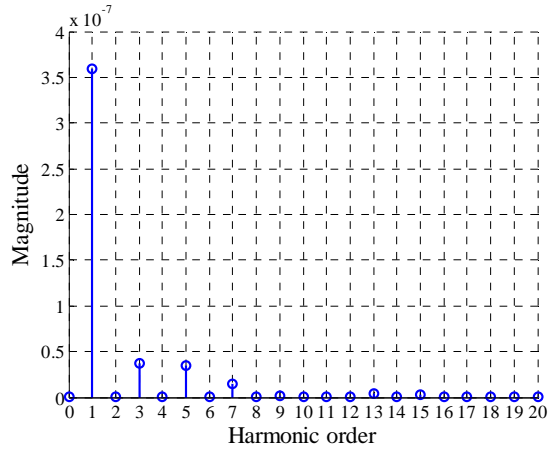


(a)



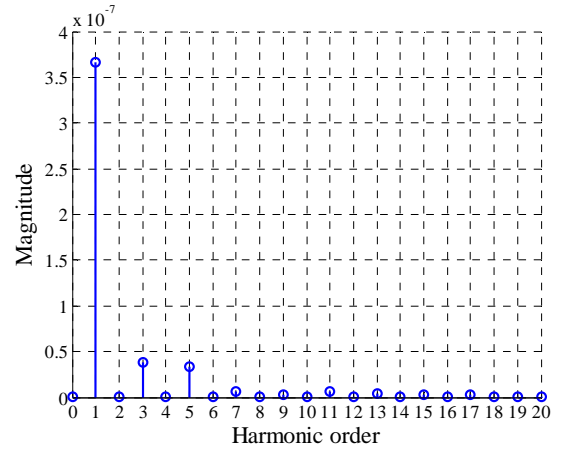
(b)

Fig. 3.11 Harmonic spectrum and its parameter table for different magnet design: (a) Full pole-pitch pseudo-radial magnetization design in 3-phase model; (b) Bread loaf design in 9-phase model. Phase in radians.



Harmonic order	Amplitude (10 <sup>-7</sup> )	Phase
0	0.000	0.000
1	3.594	-1.580
2	0.000	1.964
3	0.372	1.545
4	0.000	-1.689
5	0.347	1.527
6	0.000	1.189
7	0.145	-1.631
8	0.000	0.123
9	0.021	-1.658
10	0.000	-3.038
11	0.006	1.476
12	0.000	-2.280
13	0.037	1.444
14	0.001	0.797
15	0.034	-1.726
16	0.000	-2.971
17	0.011	-1.646
18	0.000	2.868
19	0.009	1.354
20	0.000	-2.686

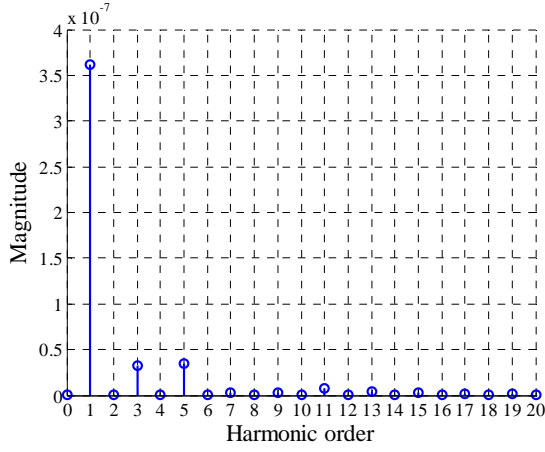
(a)



Harmonic order	Amplitude (10 <sup>-7</sup> )	Phase
0	0.000	0.000
1	3.661	-1.580
2	0.000	1.315
3	0.387	1.545
4	0.000	0.117
5	0.339	1.527
6	0.000	2.626
7	0.062	-1.627
8	0.000	-2.891
9	0.027	-1.654
10	0.000	-0.629
11	0.070	-1.672
12	0.001	1.762
13	0.041	1.438
14	0.000	2.882
15	0.027	1.459
16	0.000	-1.314
17	0.025	-1.735
18	0.000	-1.420
19	0.009	-1.722
20	0.001	2.239

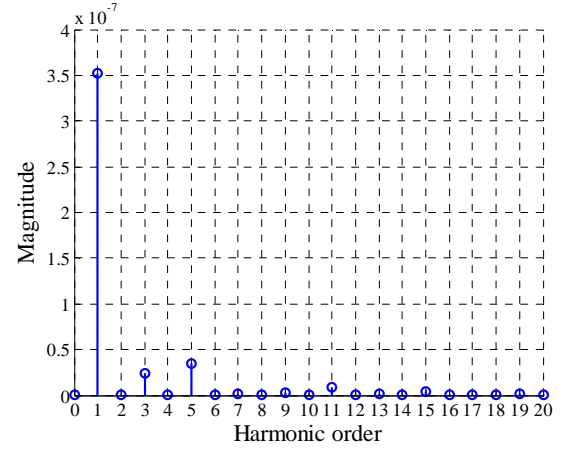
(b)

Fig. 3.12 Harmonic spectrum and its parameter table for different magnet design:  
(a) Embedded bread loaf design in 9-phase model; (b) Full pole-pitch and flat bottom design in 9-phase model. Phase in radian.



Harmonic order	Amplitude (10 <sup>-7</sup> )	Phase
0	0.000	3.142
1	3.614	-1.580
2	0.000	2.421
3	0.326	1.545
4	0.000	-0.145
5	0.346	1.527
6	0.000	-0.353
7	0.033	-1.627
8	0.000	-1.500
9	0.028	-1.657
10	0.000	-0.881
11	0.079	-1.672
12	0.000	-3.108
13	0.037	1.439
14	0.000	0.970
15	0.032	1.459
16	0.001	-1.418
17	0.021	-1.668
18	0.002	1.447
19	0.013	-1.853
20	0.001	-3.067

(a)



Harmonic order	Amplitude (10 <sup>-7</sup> )	Phase
0	0.000	0.000
1	3.516	-1.580
2	0.000	-1.076
3	0.244	1.545
4	0.000	1.256
5	0.344	1.527
6	0.000	2.854
7	0.015	1.493
8	0.000	2.504
9	0.027	-1.658
10	0.000	-0.426
11	0.093	-1.669
12	0.000	1.636
13	0.023	1.452
14	0.000	2.045
15	0.042	1.444
16	0.000	-0.158
17	0.011	-1.705
18	0.000	-1.695
19	0.018	-1.684
20	0.000	1.979

(b)

Fig. 3.13 Harmonic spectrum and its parameter table for different magnet design: (a) Full pole-pitch arc design in 9-phase model; (b) Full pole-pitch pseudo-radial magnetization design in 9-phase model. Phase in radian.

### 3.3.3 Electromagnetic torque capability

Torque capability was one of the most important parameter in terms of machine designing, because it indicate the mechanical power of a electrical machine, which could be expressed as:

$$P = T \cdot \omega, \quad (3.12)$$

where  $T$  was the torque with unit  $Nm$ ;  $\omega$  was the angular velocity with unit  $rad/s$  and the unit for power was  $Watt$ . Torque capability were limited by several aspects, including thermal environment, mechanical strength and electromagnetic capability. In this research, the electromagnetic capability was focused on and it could be represented by torque capability, demagnetization of magnet and saturation. The objective of analyzing torque capability was to find that the peak electromagnetic torque for each machine model could generated with particular driving environment, in order to compare impact of magnet designs and phase winding. The testing strategy was that a range of Direct Current (DC) were applied to phase of STAR connection of the winding circuit and machine model was simulate for a full electrical cycle, to looking for the peak torque for each current. The peak torque and the angular location where the peak torque was achieved would be gathered for further analyzing different magnet designs and phase windings. The peak torque position could be used to derive the relation between rotor position and driving current in machine excitation mode, i.e. phase difference between electromagnetic field and current. Theoretically, the maximum force would be produced when the direction of field and current was perpendicular to each other, according to Lorentz force law. As the state of electromagnetic field was complicated, the practical phase between maximum of the field and maximum of the current for generating peak torque was necessary.

The process of peak torque analysis by using MATLAB code was shown in Fig. 3.14 (a) and the example code of searching peak torque was shown in Fig. 3.14 (b). Data extracting program for 3-phase winding was shown in Appendix I.8 and 9-phase in Appendix I.9. The peak torque analyzing and graph generating program included in Appendix I.6. The structure of universal array for storing peak torque information was shown in Table 3.4 and for location of peak torque was shown in Table 3.5. In peak torque array, the excitation torque or electric torque, which correlated with the current in machine winding, was derived by subtracting the cogging torque from total torque generated by DC current, in order to eliminate the effect on torque capability from cogging. To note that, the data collection was separated with peak torque analyzing program for convenience of process control, i.e. the analyzing process was taken place after the torque-current data were collect for all five magnet designs and phase windings. The peak torque testing range was between 0 to 3000 Ampere turn ( $A-t$ ) peak current with increments of 100  $A-t$ , plus 3500  $A-t$  and 4000  $A-t$  to extended the range which were not likely happened in practice. 4000  $A-t$  was equal to  $12A/mm^2$  and 3000  $A-t$  was  $9A/mm^2$  and both were conceptual. The phase current for 3-phase winding could be derived by substituting peak current into Eq.3.3-3.6. There was one circuit on each tooth. Current in 9-phase model, shown in Fig. 3.15 and the phase

current could be derived using the peak phase current and *40 degree* electrical phase difference between each current. Therefore, current flowing into the circuit could for 9-phase winding could be expressed as:

$$\text{current } A_{in} = \text{current peak} \cdot \cos(0^\circ), \quad 3.13$$

$$\text{current } B_{in} = \text{current peak} \cdot \cos(0^\circ + 40^\circ), \quad 3.14$$

$$\text{current } C_{in} = \text{current peak} \cdot \cos(0^\circ + 80^\circ), \quad 3.15$$

$$\text{current } D_{in} = \text{current peak} \cdot \cos(0^\circ + 120^\circ), \quad 3.16$$

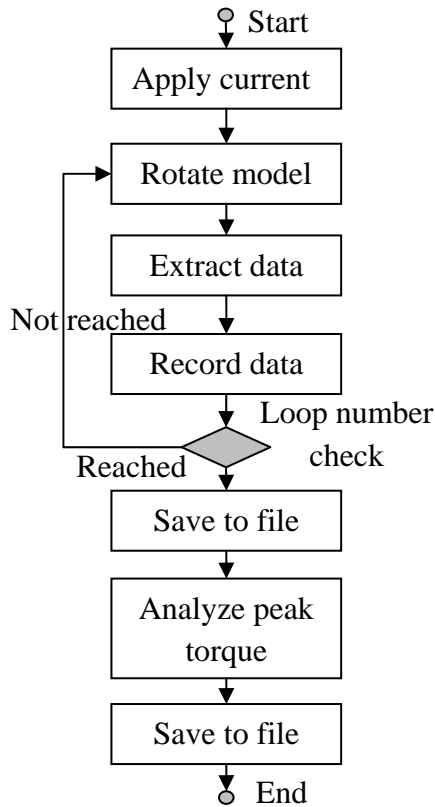
$$\text{current } E_{in} = \text{current peak} \cdot \cos(0^\circ + 160^\circ), \quad 3.17$$

$$\text{current } F_{in} = \text{current peak} \cdot \cos(0^\circ + 200^\circ), \quad 3.18$$

$$\text{current } G_{in} = \text{current peak} \cdot \cos(0^\circ + 240^\circ), \quad 3.19$$

$$\text{current } H_{in} = \text{current peak} \cdot \cos(0^\circ + 280^\circ), \quad 3.20$$

$$\text{current } I_{in} = \text{current peak} \cdot \cos(0^\circ + 320^\circ). \quad 3.21$$



(a)

```

a = max(torque_EM);
b = min(torque_EM);
a_index = find(torque_EM == a);
b_index = find(torque_EM == b);

array_mm_breadloaf(current_count,2) ...
= array_torque_breadloaf(a_index,1);

array_mm_breadloaf(current_count,4) ...
= array_torque_breadloaf(b_index,1);

array_mm_breadloaf(current_count,3)= a;
array_mm_breadloaf(current_count,5)= b;

```

(b)

Fig. 3.14 Peak torque analyzing program: (a) Flow chart of analyzing process and (b) Example code for peak torque tracking.

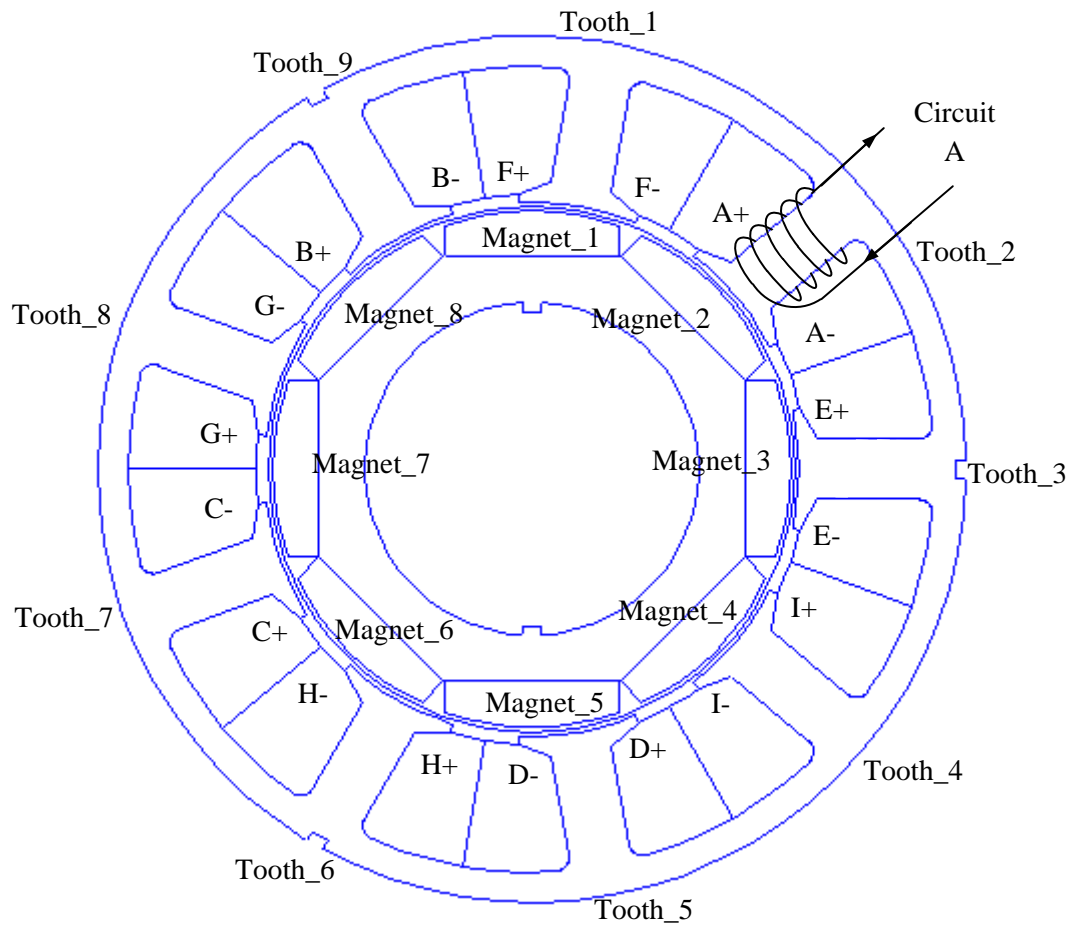


Fig. 3.15 Winding schematic for 9-phase machine model.

Electric position (Degree)	Cogging torque (Nm)	3-phase								9-phase
		Torque_sum (Nm)				Torque_electic (Nm)				Torque_su
		100 A-t	200 A-t	...	4000 A-t	100 A-t	200 A-t	...	4000 A-t	
0										
1										
⋮										
369										

Table 3.4 Universal array for storing peak torque data.

	3-Phase				9 Phase			
Current (A-t)	Maximum Torque postion (dgree)	Maximum Torque (Nm)	Minimum Torque postion (dgree)	Minimum Torque (Nm)	Maximum Torque postion (dgree)	Maximum Torque (Nm)	Minimum Torque postion (dgree)	Minimum Torque (Nm)
100								
200								
⋮								
4000								

Table 3.5 Universal array for storing peak torque location.

Electromagnetic torque capability for five different magnet design was presented in Fig. 3.16. The straight line fitting for first five data points of bread loaf design were also added to both 3-phase and 9-phase plot for locating when would the torque plots lose linearity, shown in Fig. 16 (a) and Fig. 3.16 (b) respectively. As current increasing in circuit, the electromagnetic field became stronger hence the torque would be higher. When stator steel became saturated by some stage, the increase in current would not yield higher torque. The losing of linearity in torque characteristic plot was meant that some field energy was lost during torque production and the loss was caused by some saturated area on stator, in this pure electromagnetic simulation. For both 3-phase and 9-phase, all five magnetic designs lost linearity at about  $1000\text{ A-t}$ , which was because the same stator design was shared among models. To note that, torque characteristic of 9-phase lost its linearity slightly early than 3-phase models and which meant saturation area would appear earlier during the test. Among the five magnet designs with 3-phase winding, using peak current  $1000\text{ A-t}$  as example, the full pole-pitch and flat bottom design performed the best in torque and full pole-pitch pseudo-radial design was the worst. There was about  $1\text{ Nm}$  or  $5\%$  more torque generated in the best case than the worst.

The other three designs produced similar torque with same current input and among which the torque of the embedded bread loaf design was the lowest. The bread loaf design produced slightly more than full pole-pitch arc and the difference was less than  $1\%$ . For 9-phase winding, the difference in torque was similar to 3-phase torque plot when the peak current was  $1000\text{ A-t}$  in circuit. The torque performance gap between full pole-pitch and flat bottom design and bread loaf design was narrowed. There was similar improvement for full pole-pitch arc design. On the contrary, the difference between embedded bread loaf design and full pole-pitch and flat bottom design was widened. This could mean that the bread loaf design and full pole-pitch arc design were better design for 9-phase winding and

the embedded bread loaf design was not. The peak torque characteristic for both 3-phase and 9-phase winding was represented in the same plot, shown in Fig. 3.17. There was noticeable improvement in torque production for 9-phase winding when peak phase current was 1000  $A-t$ . The improvements were 1.2 Nm, 1.04 Nm, 1.1 Nm, 1.2 Nm and 1.2 Nm for Bread loaf design, Embedded bread loaf design, Full pole-pitch and flat bottom design, Full pole-pitch arc design and Full pole-pitch pseudo-radial magnetization design respectively. 1000  $A-t$  was set as maximum peak current in further research in machine model analysis.

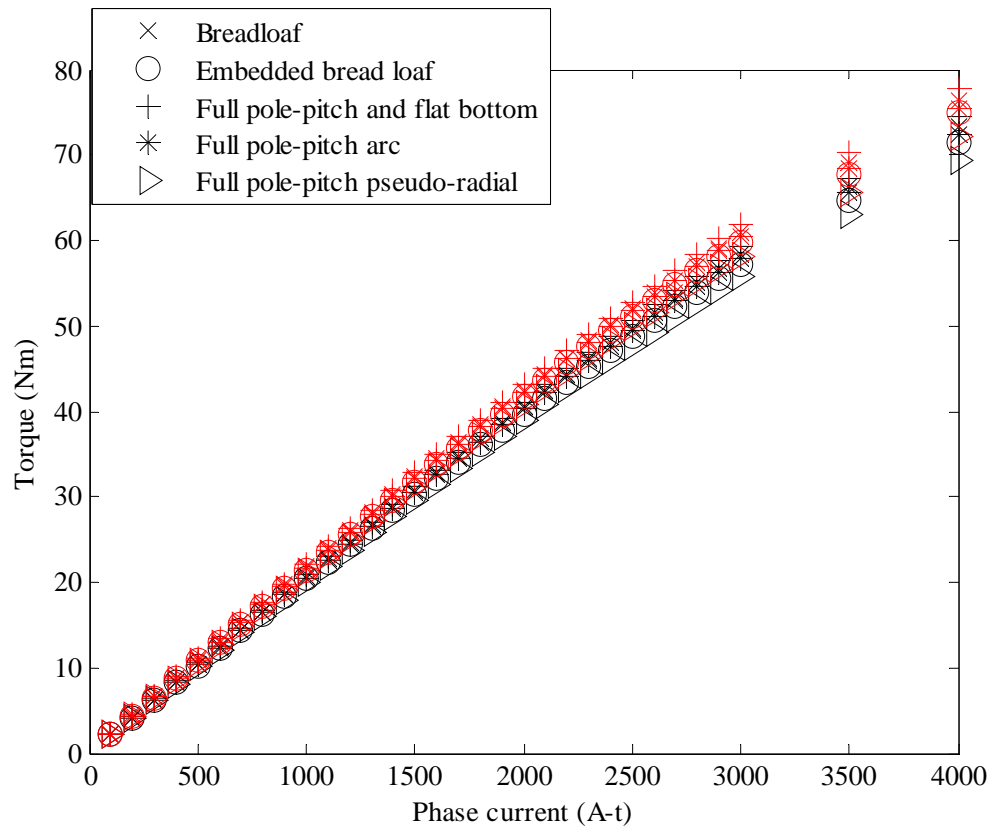
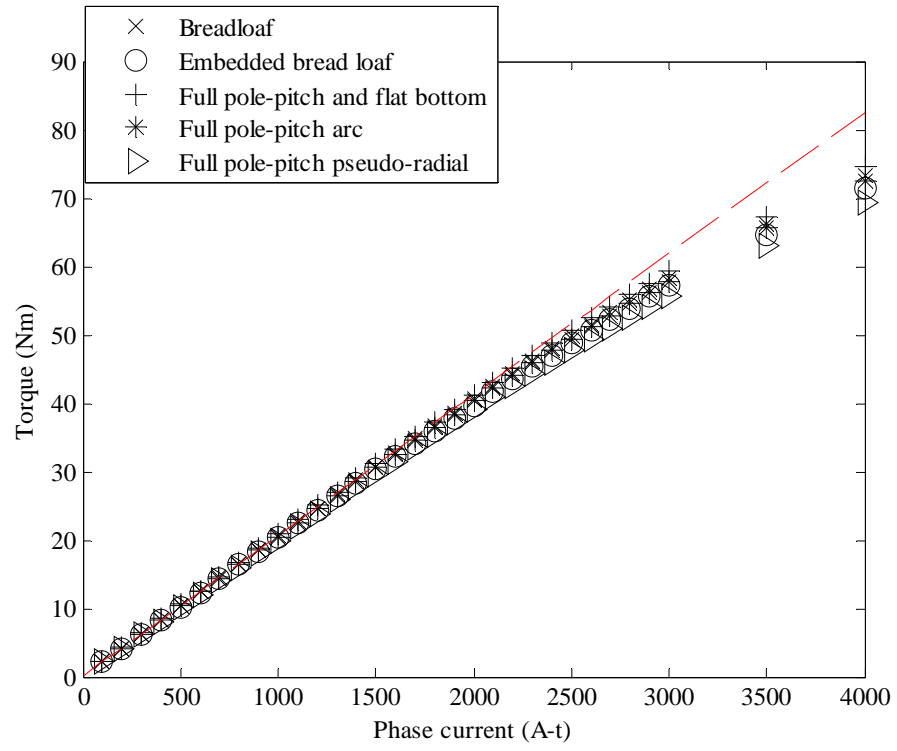
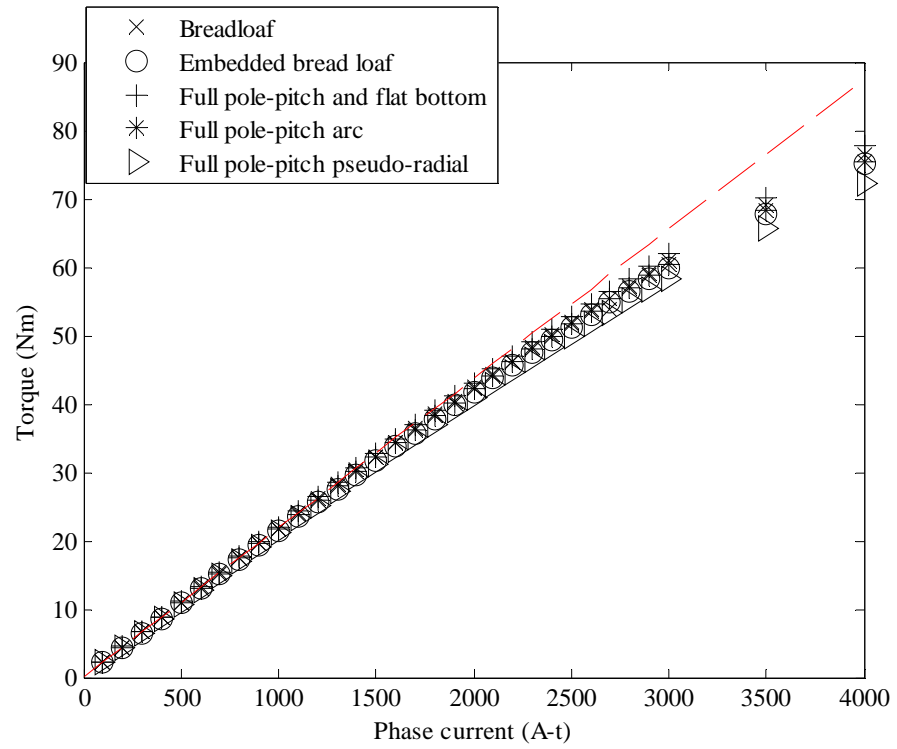


Fig. 3.17 Torque versus current characteristic for different magnet design with both 3-phase (black marks) and 9-phase (red marks) windings





(a)



(b)

Fig. 3.16 Torque versus current characteristic for different magnet designs: 3-phase (a) and 9-phase (b).

### 3.3.4 Demagnetization limits

A permanent magnet was normally made from rare earth material and it could store magnetic energy permanently by processing it in a strong electromagnetic field. After magnetization process, the magnet would maintain a consistent magnetic field surround it. The strength of the magnetic field would be weaken if physical impact, extreme temperature or demagnetizing by another field occurred to magnet. The defected magnet with weaken field would cause negative impact on the performance of a electric machine. Hence the operating environment was very important when permanent magnet was involved.

The demagnetization caused by winding field was typical incident in PM machine operating. The magnetic recoil process of a magnet was shown in Fig. 3.18 to stress the situation. Point A was located on the characteristic line of magnet and the magnet was pro-magnetized in first quadrant. Since the direction of flux density of field was the same as magnetization direction, there would not affect on the characteristic of magnet. As the strength of external field increase in opposite direction, the dynamic working point was pushed into second quadrant and point B was typical working point in practical. If the field increase further and the working point of magnet passed point C, i.e. reach the non-linear part of the characteristic at point D, the magnet would be demagnetized and as a result the working point would follow a new characteristic line permanently. When the external field was weaken, instead of returning to point B, the working point would be at point E. In practice, when electric machine was operating, heat was generated inevitably due to the efficiency of energy converting. Actually, the point of losing linearity of the magnet could be pushed further up due to the thermal effect. Hence a limit was set on magnet characteristic line to indicate where the possible demagnetization would occurred. Using the linear part characteristic of 'NdFeB 37 MGO', the value of limited was set to  $0.2 T$ .

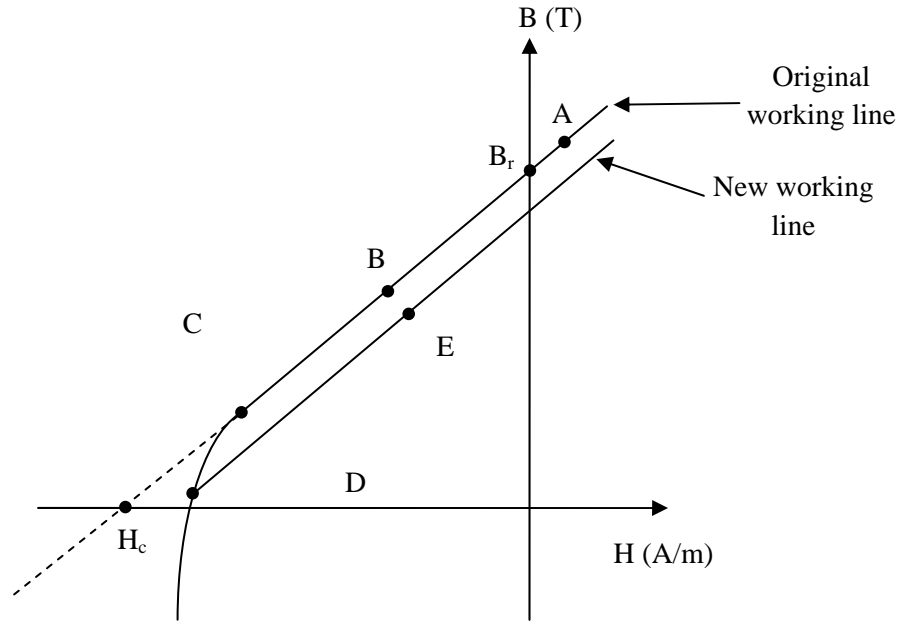


Fig. 3.18 The recoil process of a magnet after demagnetization.

The characteristic of the magnet which passed the limit was not in scale of this research and therefore characteristic was assuming to be linear. In FEMM model, the flux density,  $B$ , could be obtained based on FEA method and the working point of magnet could be represented on working line and any possible of demagnetization could be observed. The linear characteristic could be described as a linear equation:

$$B = \frac{B_r}{H_c} H + B_r, \quad (3.22)$$

where  $B_r$  was Residual magnetism and  $H_c$  was Coercive force of which the value was  $9.5 \times 10^5 \text{ A} \cdot \text{m}$  from FEMM library. Since the magnetic permeability was

$$\mu = \frac{B}{H} = \frac{B_r}{H_c} \quad (3.23)$$

and

$$\mu = \mu_r \cdot \mu_o, \quad (3.24)$$

where the unit of  $\mu$  was  $\text{Wb} / (\text{A} \cdot \text{m})$ ,  $\mu_r$  was relative permeability with value of 1.048  $\text{Wb} / (\text{A} \cdot \text{m})$  in FEMM library and  $\mu_o$  was the vacuum permeability with value of  $4 \pi \times 10^{-7} \text{ Wb} / (\text{A} \cdot \text{m})$ , Residual magnetism,  $B_r$  could be derived as

$$\begin{aligned}
B_r &= \mu \cdot H_c \\
&= \mu_r \cdot \mu_o \cdot H_c \\
&= 1.048 \times 4\pi \times 10^{-7} \times 9.5 \times 10^5 \\
&\approx 1.25 (T)
\end{aligned} \tag{3.25}$$

and the linear characteristic of magnet would become

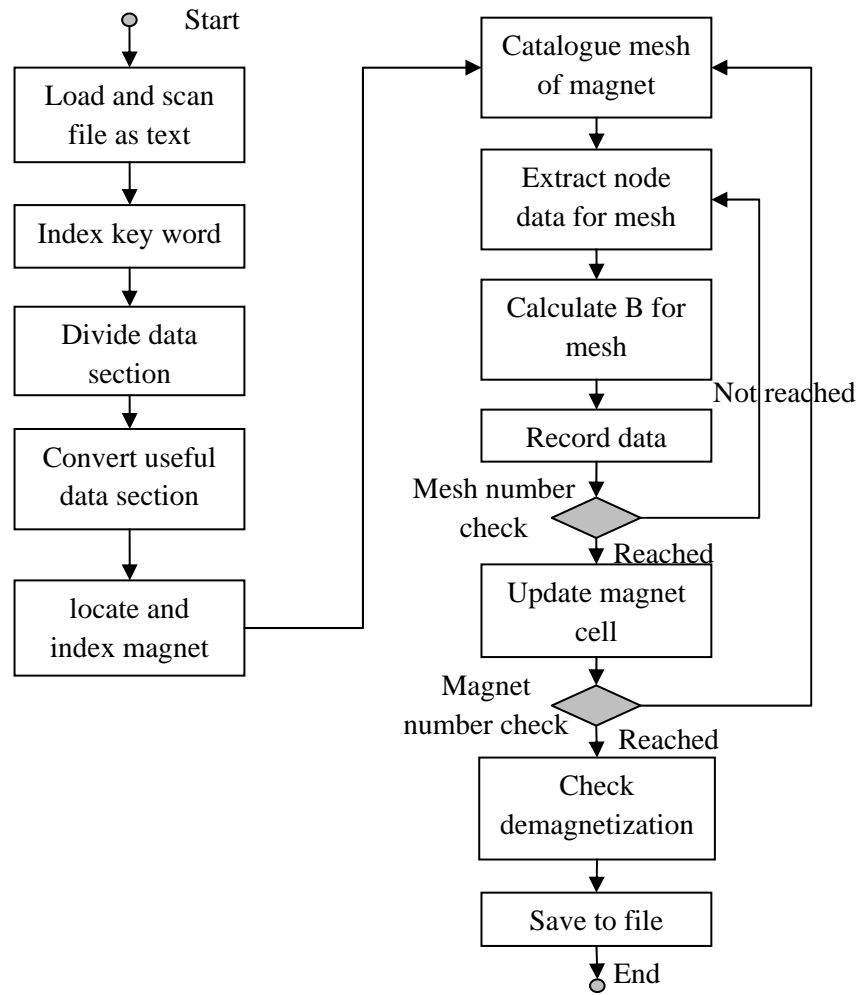
$$B = \frac{1.25}{9.5 \times 10^5} \cdot H + 1.25 = 1.317 \times 10^{-6} \cdot H + 1.25. \tag{3.26}$$

With this linear relation of magnet, any part of the magnet could be checked for demagnetization, even in scale of mesh element. The demagnetization was a process of a magnet who was losing its magnetic characteristic and the process could not be completed instantly. Within the process, some part of magnet could be demagnetized prior the change of overall characteristic was noticed. Hence, the inspecting working point in different areas was necessary. In following part of this section, the electromagnetic state of each mesh element of magnet was investigated when the maximum peak phase current, i.e. *1000 A-t* was applied to stator winding. The working point of each element was plot on characteristic line for each magnet for inspecting by using MATLAB program.

Recalling Eq. 3.1-3.3, the flux density for a mesh element could derived with element detail, i.e. coordinate, vector potential and mesh area. To access the detail information of mesh elements, the result file of FEMM model would be analyzed by using MATLAB. A program was developed to handle the information contained in result file and the flow chart was shown in Fig. 3.19. The result file was treated as pure text, or *string* in MATLAB to ensure the complete set of data was recognized and the most importantly, to preserve the structure of data. The text data was loaded into temporary location for further analysis, i.e. workspace. Next step was to identify data section and text searching could be employed to locate the title of data section.

In result file, there were three types of content: text, symbol and number. Since the whole result file was stored as text in workspace, the effect from other types was illumined and the title of data section could identified and indexed. As mentioned previously, the structure of result file was consistent and the index of certain data was maintained for same model file through the simulation, i.e. block information, unless the model was modified. Three sections of most important data, block information, node information and mesh index, were extracted from text version of result file, then converted as number and store in separated arrays with original structure for further manipulating. Locating magnet blocks

was relative simple task to achieve. In block array extracted from result file, the index of magnet blocks could obtain by searching the material number in material type column.



(a)

Mesh	Node_1			Node_2			Node_3			Calculated data			
index	$X_i$	$Y_i$	$A_i$	$X_j$	$Y_j$	$A_j$	$X_k$	$Y_k$	$A_k$	$B_x$	$B_y$	$B_{abs}$	$B_{angle}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

(b)

Fig. 3.19 Flow chart for result file analyzing program (a) and magnet array structure (b).

In the node array, each row contained node coordinate and vector potential and hence the row number would become the node index. A mesh element was consisted with three nodes and the node index of each node included in a row of mesh array. The block of which this mesh element was belong to was also store in the same row. Using the block index of magnet as searching criteria, mesh element which were created within the block could be catalogued. By replacing the node index number in mesh array with node

information, a complete magnet array which contained detail mesh element information could be created in a magnet cell. cell was a higher level of array, which could handle more complex data without computational error. A magnet cell which stored all magnet mesh information and one array for each magnet block could be created by repeating the searching and replace process. The flow chart of file analyzing program was shown in Fig. 3.19 (a) and the structure of magnet array was shown in Fig. 3.19 (b).

To implement the flux density calculation for mesh elements, last piece of information was required: the area of the a mesh element. With the coordinate of three nodes, the side length of mesh triangle could be derived in Cartesian coordinate system. Using side  $ij$  as an example, shown in Fig. 3.20, and by projecting point  $i$  and point  $j$  onto axis, the length difference in x and y component could be described respectively as

$$io = X_j - X_i, \quad (3.27)$$

$$jo = Y_j - Y_i. \quad (3.28)$$

The length of  $a$  could be calculated as

$$a = ij = \sqrt{io^2 + jo^2} = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2} \quad (3.29)$$

and hence the expression of length of  $b$  and  $c$  were

$$b = \sqrt{(X_k - X_j)^2 + (Y_k - Y_j)^2}, \quad (3.30)$$

$$c = \sqrt{(X_k - X_i)^2 + (Y_k - Y_i)^2}. \quad (3.31)$$

Then the area of a triangle  $ijk$  could be derived by using Heron's Formula:

$$\begin{aligned} A &= \sqrt{s \cdot (s - a) \cdot (s - b) \cdot (s - c)} \\ &= \sqrt{\frac{a+b+c}{2} \cdot \left(\frac{a+b+c}{2} - a\right) \cdot \left(\frac{a+b+c}{2} - b\right) \cdot \left(\frac{a+b+c}{2} - c\right)} \\ &= \frac{\sqrt{(a+b+c) \cdot (-a+b+c) \cdot (a-b+c) \cdot (a+b-c)}}{4}. \end{aligned} \quad (3.32)$$

The equations implemented for calculating the flux density of a mesh element were Eq. 3.29-Eq. 3.32 and Eq.3.1-Eq. 3.3 and the flux density was absolute value. A analyzing program with result file processing was shown in Appendix I.10. The angle of flux density could be calculated separately by using axial component of flux density,  $B_x$  and  $B_y$ . The trigonometry function arctangent was used to determine the angle between two perpendicular lines. To cover the full range of flux density angle, a table of logic was shown in table 3.6. For a magnet, its magnetization direction should dominating the flux

angle of flux density of mesh element. The calculation of flux density in magnetization direction was also implemented in analyzing program for investigating the detail of demagnetization, also this part of program would be used for calculate flux density in radial direction for steel later in this research. It was easier to derive flux density in magnetization direction by using two component of flux density of mesh than using flux density itself, since the chance of error introduction was reduce by using the same source of parameters and demagnetization angle could be used straight forward without considering the angular relation between flux density and dominating direction. By projecting two components of flux density on magnetization direction, shown in Fig. 3.20 (b), the flux density in dominating direction could be derived by using

$$B_{dom} = B_{x\_dom} + B_{y\_dom} = B_x \cdot \cos \angle B_{dom} + B_y \cdot \sin \angle B_{dom} . \quad (3.33)$$

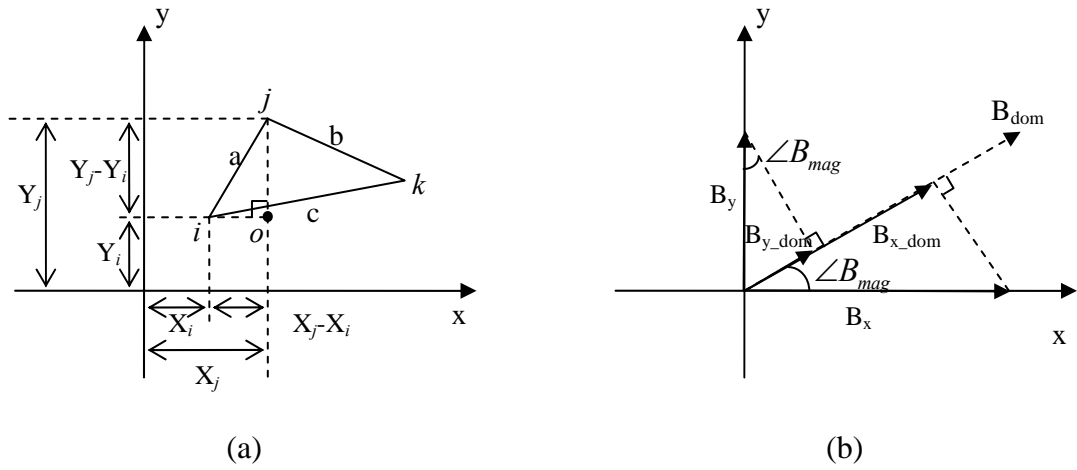


Fig. 3.20 Implemented calculations in Cartesian coordinate system: (a) Side length and (b) Flux density in demagnetized direction.

Case number	Case condition	Angle of flux density (rad)
1	$B_x = 0$	$\angle B = 0$
2	$B_x \geq 0$ and $B_y \geq 0$	$\angle B = \arctan\left(\frac{B_y}{B_x}\right)$
3	$B_x < 0$	$\angle B = \arctan\left(\frac{B_y}{B_x}\right) + \pi$
4	$B_x \geq 0$ and $B_y < 0$	$\angle B = \arctan\left(\frac{B_y}{B_x}\right) + 2\pi$

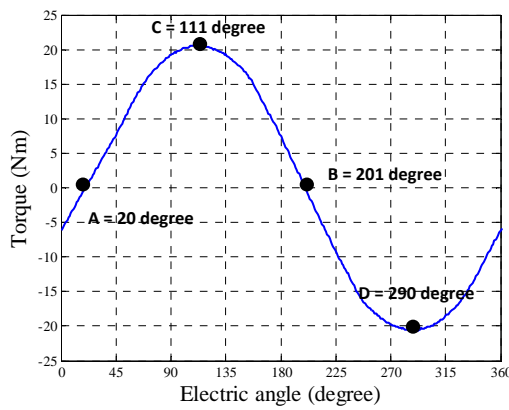
Table.3.6 The case table for calculating angle of flux density.

Within a cycle of peak torque test, there were four positions where the demagnetization could occur. Two of them were at peak torque position and the other two were at zero torque position. During the rotating of rotor, winding field was fixed since the phase current was remained the same. The angular position of rotor indicated the angle between winding field and magnetic field of magnet. When the direction of both fields was overlapped, peak torque was produced at peak torque position and both pro-magnetization and demagnetization would happen. At zero torque position, two fields were perpendicular to each other and where the magnetization unlikely to be happened. It would be sufficient for cover demagnetization check by examining at these four position. The peak torque position for each magnet design at particular peak phase current was extract from previous simulation. The zero torque positions was obtained by manually checking the electrical torque plot and the testing positions were shown in (a) of Fig. II.12, II.17, II.22, II.27, II.32, II.37, II.42, II.47, II.52 and II.57. The demagnetization check was established on five magnets design models when the peak phase current was  $1000\text{ A-t}$ , for both 3-phase and 9-phase winding.  $1000\text{ A-t}$  was selected because it below machine saturation limit as shown in Fig. 3.17 and the current density was  $3\text{ A/mm}^2$ . The number of element of which working point was lower than  $0.2\text{ T}$  were logged in Table 3.7-3.14. Note that the number listed in Table 3.7-3.14 were number of meshes and the area of demagnetization may be not preserved by this number, because mesh size could not be controlled at this level in FEMM.

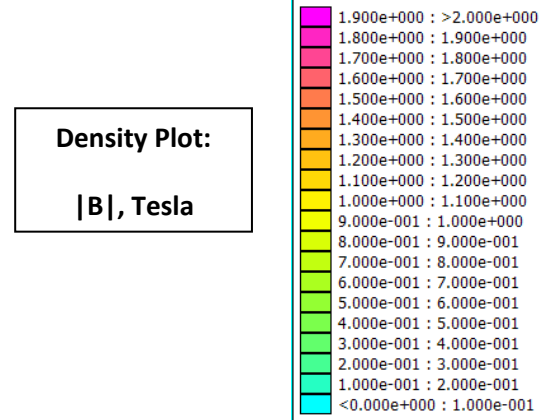
The size of demagnetization area was check visually and a number of 30 was selected as significant. The main part of demagnetization check program was result file analyzing and simple model rotating command for a step was added in the front of the program, to create and analyze the snapshot of the model at testing position. The full program for checking 3-phase models was shown in Appendix I.10. The variables of program was changed manually according to the testing model. The parameters was including name of model file, angular position of testing point and peak phase current. Also the current profile was different between 3-phase and 9-phase winding. While logging working points of magnet, magnet map was also created to ensure the data was corresponding to the correct magnet for each magnet design. The solution density plots for each testing position were shown in (c), (d), (e) and (f) of Fig. II.12, II.17, II.22, II.27, II.32, II.37, II.42, II.47, II.52 and II.57. The areas with possible demagnetization were also indicated with circle. By using the linear relation in Eq. 3.26, working point for every mesh element could be determined and



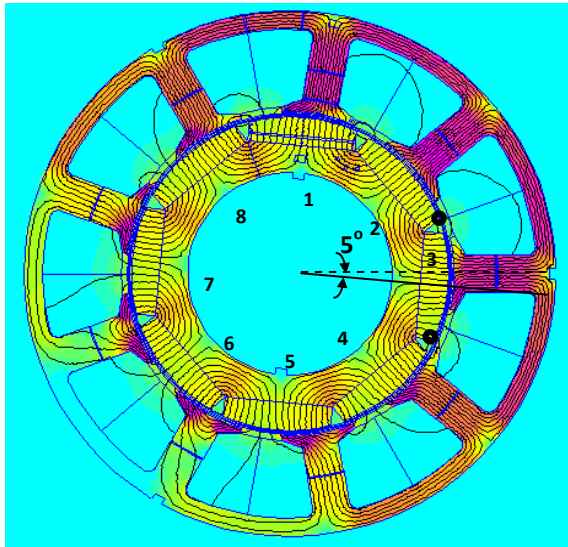
they were plotted for each magnet in Fig. II.13-II.16, II.18-II.21, II.23-II.26, II.28-II.31, II.33-II.36, II.38-II.41, II.43-II.46, II.48-II.51, II.53-II.56 and II.58-II.61.



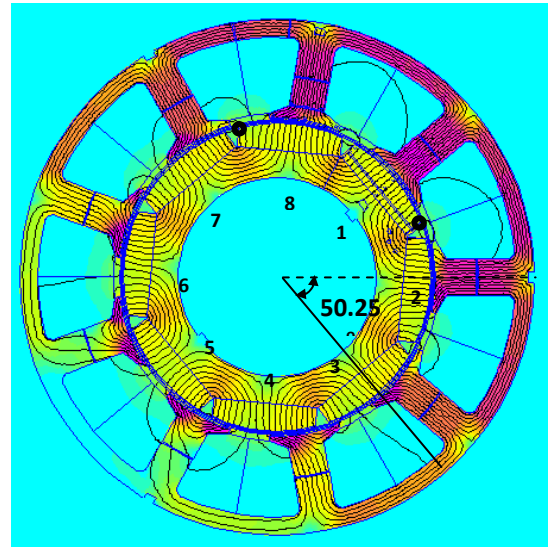
(a) Demagnetization checking position.



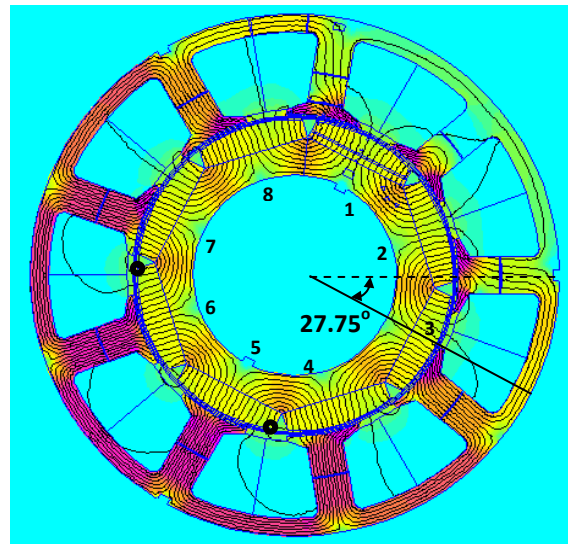
(b) Legend of field solution.



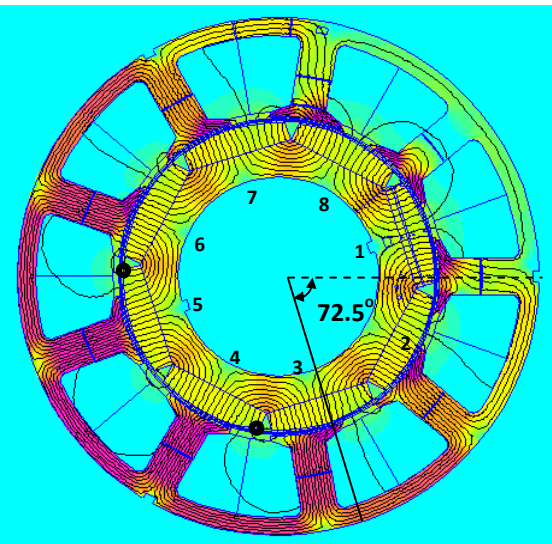
(c) FEA field solution for position A.



(d) FEA field solution for position B.



(e) FEA field solution for position C.



(f) FEA field solution for position D.

Fig. 3.21 FEA field solutions of bread loaf design for each position when peak phase current of 3-phase winding was 1000A-t, with possible demagnetized meshes circled.

The straight line was the B-H characteristic of magnet based on linear relation and the bigger size of cross mark was indicated the average of all mesh elements within a magnet, or the magnet working point. For 3-phase concentrated winding, demagnetization of embedded magnet design model was the worst at every position among the different magnet designs. The number of demagnetizing mesh was significant higher than other designs and was not evenly distributed for each magnet. At point B, the situation was similar with it was at point A: demagnetizing elements was concentrated at the corners and the number in each of three magnets was two times as in others. The difference was that the three magnets was *magnet 1*, *magnet 2* and *magnet 8*. From the result of mesh element checking at point A and B, the demagnetizing elements were assemble in the magnets aligned with the tooth of phase winding A. The number of demagnetizing elements in each magnet of *magnet 1*, *magnet 2* and *magnet 3*, was as twice as much it was in other individual magnet.

There was noticeable more demagnetizing mesh elements in three of magnets at each position and which was likely to be a pattern. By cross checking with the solution density plot, the demagnetizing meshes were concentrated on one side of the rotor and the location was opposite phase winding A for testing position A and B. At peak torque position, the concentrated area was on one side and it was opposite the juncture of phase winding B and C. Also the area was spread to four magnets instead of three. The number of demagnetizing mesh elements at peak torque position was about 20 more than it was at zero torque on average. The best designs in demagnetization checking were bread loaf design and full pole-pitch pseudo-radial magnetization design, especially the second one in which case there was none demagnetizing mesh. There only two meshes was on risk of demagnetization and their locations followed the same distribution pattern with embedded design. The number of demagnetizing meshes was similar for full pole-pitch and flat bottom design and full pole-pitch arc design and the second one was slightly better.

Due to the size of mesh element, difference in such small margin may mean no difference in demagnetizing region and it was not sufficient for comparing two designs. The distribution of demagnetizing meshes for full pole-pitch design could be observed and it was similar with the pattern of both bread loaf designs and introduced one magnet shift. Unlike bread loaf magnet designs, number of demagnetizing meshes decreased slightly at peak torque position, comparing it was at zero torque position. For the 9-phase winding, there no difference in demagnetization state from 3-phase winding for all five magnet

designs at zero torque positions and it was the same at peak torque position for bread loaf design and full pole-pitch pseudo-radial magnetization design. There was small increase in number of demagnetizing mesh elements for full pole-pitch and flat bottom and full pole-pitch arc, comparing the number in 3-phase winding. By checking the location of possible demagnetized mesh elements, the area of demagnetization was concentrated at two corners of each magnet, which were partially exposed to airgap.

In working point plots, although the working point of the magnet was lay with a stable distance away from the demagnetizing limit, demagnetization did occur in some mesh elements, i.e. embedded magnet design. The state of magnetization unlikely to be happened at zero torque position was not true, but the reduction in number of element was noticeable, especially for embedded bread loaf magnet design, comparing with the number at peak torque position. There was no significant difference in number of demagnetizing mesh elements by changing from 3-phase to 9-phase winding. The embedded bread loaf design was questionable since the dogmatization was much worse than other designs. Although there was no demagnetization for full pole-pitch pseudo-radial magnetization design through the checking, the design was not favourable comparing the performance in EMF and torque production with bread loaf design.

At point A	1000 A-t peak current in 3-phase winding				
	Bread Loaf	Embedded bread loaf	Full pole-pitch and flat bottom	Full pole-pitch arc	Full pole-pitch pseudo-radial magnetization
Magnet_1	0	86	20	13	0
Magnet_2	1	95	27	21	0
Magnet_3	1	98	9	8	0
Magnet_4	0	50	13	15	0
Magnet_5	0	42	8	8	0
Magnet_6	0	41	10	6	0
Magnet_7	0	41	12	9	0
Magnet_8	0	51	33	17	0

Table.3.7 Demagnetization table 1.

At point B	1000 A-t peak current in 3-phase winding				
	Bread Loaf	Embedded bread loaf	Full pole-pitch and flat bottom	Full pole-pitch arc	Full pole-pitch pseudo-radial magnetization
Magnet_1	1	87	27	19	0
Magnet_2	0	98	9	9	0
Magnet_3	0	50	13	12	0
Magnet_4	0	42	8	8	0
Magnet_5	0	41	10	8	0
Magnet_6	0	41	12	9	0
Magnet_7	0	51	33	18	0
Magnet_8	1	100	20	10	0

Table.3.8 Demagnetization table 2.

At point C	1000 A-t peak current in 3-phase winding				
	Bread Loaf	Embedded bread loaf	Full pole-pitch and flat bottom	Full pole-pitch arc	Full pole-pitch pseudo-radial magnetization
Magnet_1	0	67	9	8	0
Magnet_2	0	71	8	8	0
Magnet_3	0	78	9	12	0
Magnet_4	0	123	8	8	0
Magnet_5	1	128	16	20	0
Magnet_6	1	128	12	16	0
Magnet_7	0	91	13	13	0
Magnet_8	0	71	8	8	0

Table.3.9 Demagnetization table 3.

At point D	1000 A-t peak current in 3-phase winding				
	Bread Loaf	Embedded bread loaf	Full pole-pitch and flat bottom	Full pole-pitch arc	Full pole-pitch pseudo-radial magnetization
Magnet_1	0	59	11	9	0
Magnet_2	0	71	14	8	0
Magnet_3	0	104	23	15	0
Magnet_4	1	128	20	14	0
Magnet_5	1	127	26	17	0
Magnet_6	0	109	8	8	0
Magnet_7	0	72	9	11	0
Magnet_8	0	68	8	8	0

Table.3.10 Demagnetization table 4.

At point A	1000 A-t peak current in 9-phase winding				
	Bread Loaf	Embedded bread loaf	Full pole-pitch and flat bottom	Full pole-pitch arc	Full pole-pitch pseudo-radial magnetization
Magnet_1	0	86	20	13	0
Magnet_2	1	95	27	21	0
Magnet_3	1	98	9	8	0
Magnet_4	0	50	13	15	0
Magnet_5	0	42	8	8	0
Magnet_6	0	41	10	9	0
Magnet_7	0	41	12	9	0
Magnet_8	0	51	33	17	0

Table.3.11 Demagnetization table 5.

At point B	1000 A-t peak current in 9-phase winding				
	Bread Loaf	Embedded bread loaf	Full pole-pitch and flat bottom	Full pole-pitch arc	Full pole-pitch pseudo-radial magnetization
Magnet_1	1	87	27	19	0
Magnet_2	1	98	9	21	0
Magnet_3	0	50	13	12	0
Magnet_4	0	42	8	8	0
Magnet_5	0	41	10	8	0
Magnet_6	0	41	12	8	0
Magnet_7	0	51	33	18	0
Magnet_8	0	100	20	10	0

Table.3.12 Demagnetization table 6.

At point C	1000 A-t peak current in 9-phase winding				
	Bread Loaf	Embedded bread loaf	Full pole-pitch and flat bottom	Full pole-pitch arc	Full pole-pitch pseudo-radial magnetization
Magnet_1	0	61	8	8	0
Magnet_2	0	67	10	8	0
Magnet_3	0	71	12	8	0
Magnet_4	0	103	34	15	0
Magnet_5	1	128	20	12	0
Magnet_6	0	127	27	20	0
Magnet_7	0	109	8	9	0
Magnet_8	0	72	13	15	0

Table.3.13 Demagnetization table 7.

At point D	1000 A-t peak current in 9-phase winding				
	Bread Loaf	Embedded bread loaf	Full pole-pitch and flat bottom	Full pole-pitch arc	Full pole-pitch pseudo-radial magnetization
Magnet_1	0	66	8	8	0
Magnet_2	0	78	11	8	0
Magnet_3	0	123	9	10	0
Magnet_4	2	128	16	21	0
Magnet_5	1	128	12	12	0
Magnet_6	1	91	16	16	0
Magnet_7	0	91	9	20	0
Magnet_8	0	71	9	8	0

Table.3.14 Demagnetization table 8.

### 3.4 Excitation torque analysis

#### 3.4.1 Excitation torque with sinusoidal current

Torque is one of the important specifications for a machine and so for the machine modelling. In this part of the work, excitation torque was estimated by using measurements from machine models. Several pieces of MATLAB code was developed to control the rotation of the machine model, measure necessary parameters for torque estimation and analysis measured data. The peak torque position was obtained in previous test with cogging torque component removed and the positions was used to define the phase shifting of current supplied in phase winding. To achieve peak torque production, the phase angle between current and rotor angular position should be maintained through simulation, i.e. current would be a equation of rotor angular position and shift its phase angle with the rotor. The torque produced by 3-phase and 9-phase winding models were compared and these analytical measurement were proved by estimated torque which were calculated from analytical phase EMF measurements. The potential torque improvement for 9-phase winding design over 3-phase winding could be estimated.

Since sinusoidal EMF was observed in open circuit test for both 3-phase and 9-phase winding, sinusoidal current was employed for excitation torque test. There were few facts to consider when excitation current was created. One was the angular distance between peak torque position and reference position which was the position before rotor rotating. Another one was the phase shift between phase winding and finally, the positive direction for model rotating was anticlockwise in FEMM. Summarizing the conditions mentioned above, the excitation current could be derived with

$$current\ Ain = current\ peak \cdot \cos(-\theta + \theta_{peak}) , \quad 3.34$$

$$current\ Bin = current\ peak \cdot \cos(-\theta + \theta_{peak} + 120^\circ) , \quad 3.35$$

$$current\ Cin = current\ peak \cdot \cos(-\theta + \theta_{peak} - 120^\circ) \quad 3.36$$

for 3-phase model and where  $\theta$  was angular displacement per step and  $\theta_{peak}$  was the peak torque position for excitation and current peak was 1000 A-t. For 9-phase model, the current equations were

$$current\ Ain = current\ peak \cdot \cos(-\theta + \theta_{peak}) , \quad 3.37$$

$$current\ Bin = current\ peak \cdot \cos(-\theta + \theta_{peak} + 40^\circ) , \quad 3.38$$

$$current\ Cin = current\ peak \cdot \cos(-\theta + \theta_{peak} + 80^\circ) , \quad 3.39$$

$$current\ Din = current\ peak \cdot \cos(-\theta + \theta_{peak} + 120^\circ) , \quad 3.40$$

$$current\ Ein = current\ peak \cdot \cos(-\theta + \theta_{peak} + 160^\circ) , \quad 3.41$$

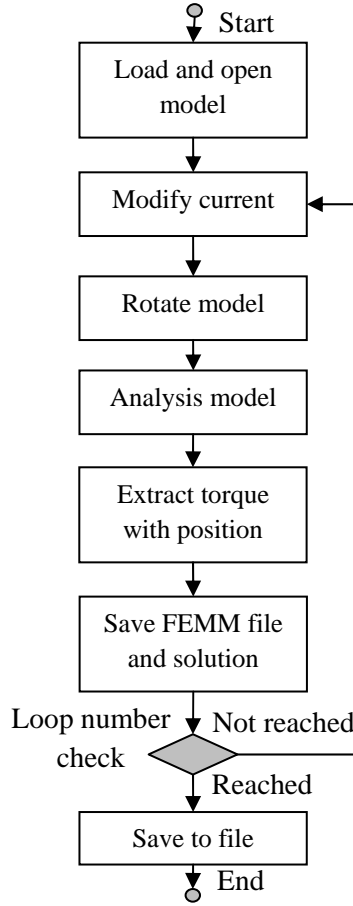
$$current\ Fin = current\ peak \cdot \cos(-\theta + \theta_{peak} + 200^\circ) , \quad 3.42$$

$$current\ Gin = current\ peak \cdot \cos(-\theta + \theta_{peak} + 240^\circ) , \quad 3.43$$

$$current\ Hin = current\ peak \cdot \cos(-\theta + \theta_{peak} + 280^\circ) , \quad 3.44$$

$$current\ Iin = current\ peak \cdot \cos(-\theta + \theta_{peak} + 320^\circ) . \quad 3.45$$

The current out of a circuit was also need to be assigned in FEMM, which was minus current flow in. For excitation tasks, the current equations was embedded into the model analyzing loop to achieve a sinusoidal current. A few of commands from Lua script were employed to accomplish phase current modification in machine model winding before rotating the model. The rotation and analysis process would be continued until the model rotated for a whole cycle. The data was save at end of the process for further analysis. A flow chart was created to guide the programming for this test, shown in Fig. 3.22(a) and the current implementation was shown in Fig. 3.22 (b). The 3-phase excitation torque analyzing program was included in Appendix I.11. The 9-phase version would be covered by automatic analyzing program at the end of this chapter.



(a)

```

current_peak = 1000;
current_Ain = current_peak
*cos((-electrcial_tooth_angle + 111
- (360/3)*0 )/360*2*pi);

current_Aout = -current_peak
*cos((-electrcial_tooth_angle + 111
- (360/3)*0 )/360*2*pi);

current_Bin = current_peak
*cos((-electrcial_tooth_angle + 111
-(360/3)*1 )/360*2*pi);

current_Bout = - current_peak
*cos((-electrcial_tooth_angle + 111
-(360/3)*1 )/360*2*pi);

current_Cin = current_peak
*cos((-electrcial_tooth_angle + 111
-(360/3)*2 )/360*2*pi);

current_Cout = -current_peak
*cos((-electrcial_tooth_angle + 111
-(360/3)*2 )/360*2*pi);
  
```

(b)

Fig. 3.22 Flow chart and example of code. (a) Flow chart for excitation torque (a) and implementation of phase current for 3-phase.

### 3.4.2 Results for excitation torque

With torque data collected by using excitation torque analyzing program, total torque of 3-phase model could be plot together with 9-phase torque, shown in Fig. 3.23. The waveforms of torque was noisy because of cogging torque. After removing cogging torque for both waveform, the excitation torque was much clearer, shown in Fig. 3.24. The average of 3-phase excitation torque was 20.72 Nm and it was 21.66Nm for 9-phase. The 4% difference matched the case in peak electric torque test, established previously. The variation of 9-phase excitation torque was 0.2 Nm versus 0.8 Nm from 3-phase. The characteristic of higher excitation torque and less variation could benefit some particular system, i.e. electric vehicle in which high torque with low ripple was required. The harmonic components were also check by FFT function and the harmonic spectrums and the tables was shown in Fig. 3.25 for both 3-phase and 9-phase model.



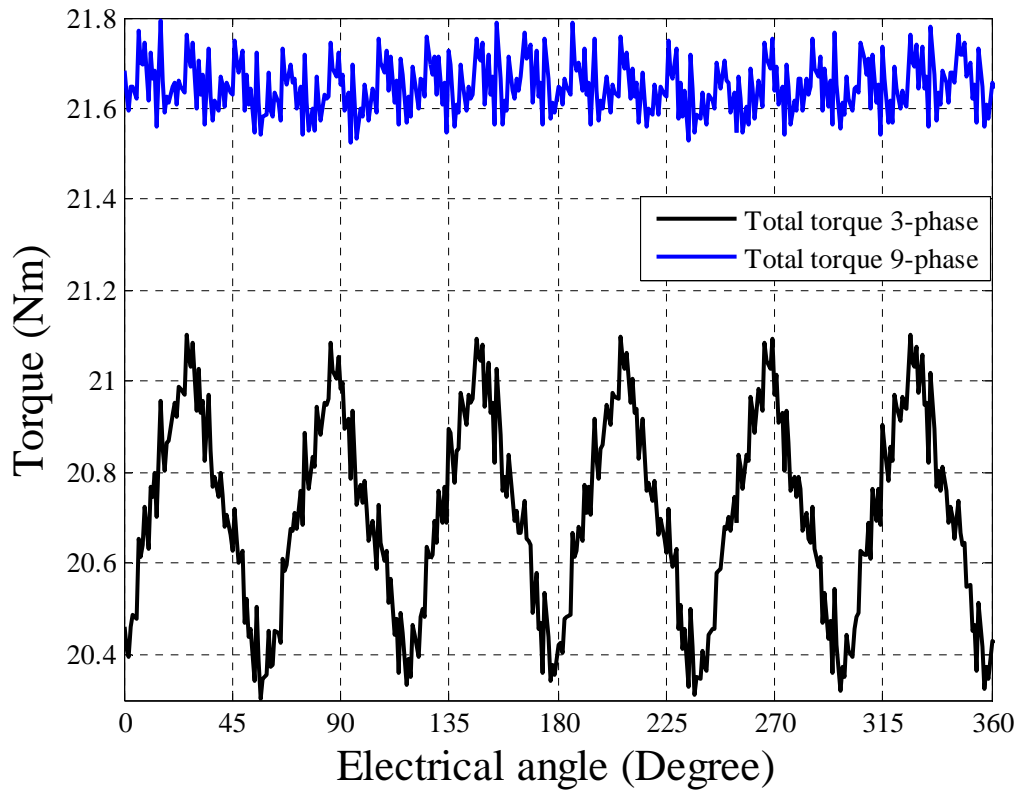


Fig. 3.23 Total torque versus electrical angle for 3-phase and 9-phase models with 1000 A-t (peak) sinusoidal phase current.

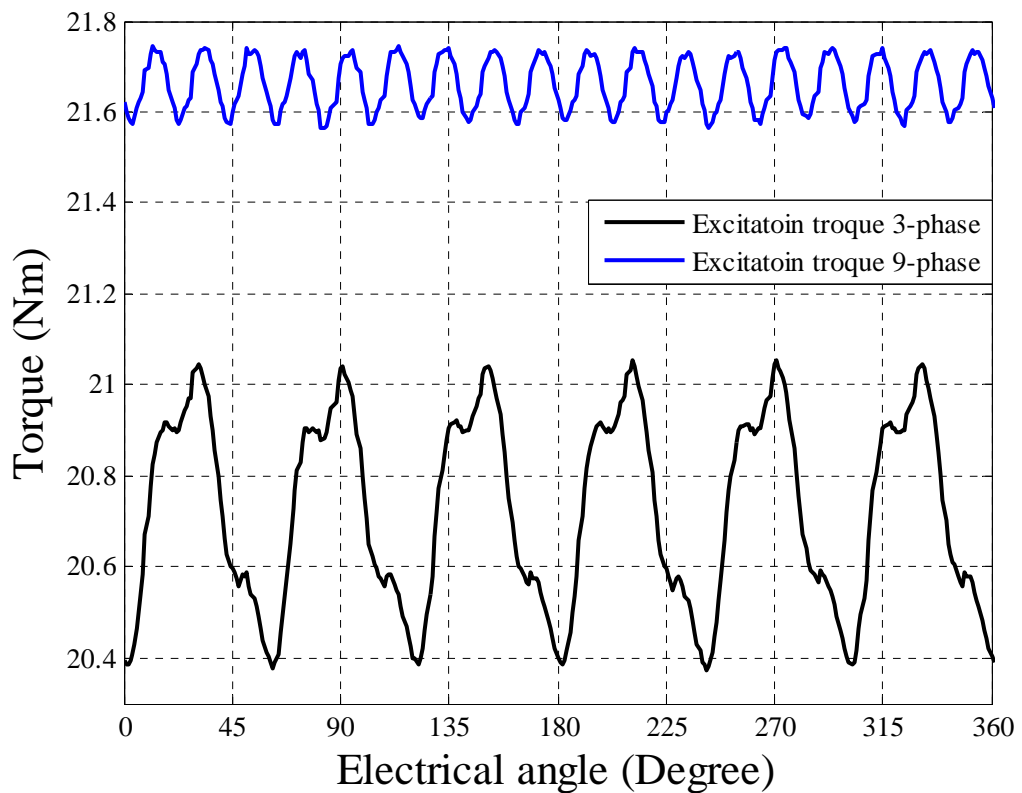
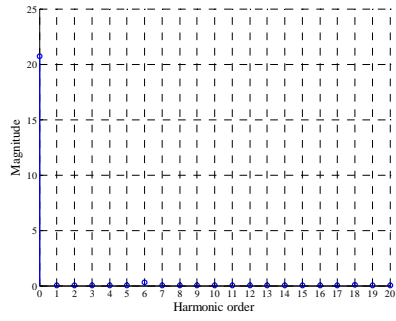
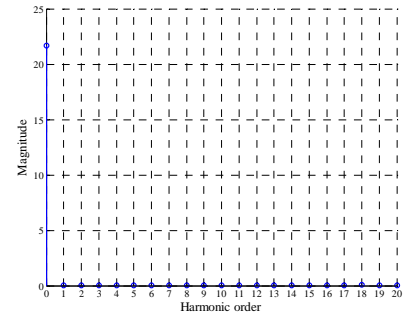


Fig. 3.24 Excitation torque versus electrical angle for 3-phase and 9-phase models with 1000 A-t (peak) sinusoidal phase current.



Harmonic order	Amplitude	Phase
0	20.726	0.000
1	0.001	0.603
2	0.002	1.006
3	0.002	-1.596
4	0.003	-1.730
5	0.002	1.245
6	0.284	-2.844
7	0.001	-1.648
8	0.000	-1.589
9	0.000	-2.641
10	0.001	0.418
11	0.000	3.043
12	0.017	2.873
13	0.000	-2.632
14	0.000	1.342
15	0.000	1.134
16	0.001	-2.022
17	0.001	-2.292
18	0.083	2.128
19	0.000	2.655
20	0.001	-2.255

(a) 3-phase excitation torque



Harmonic order	Amplitude	Phase
0	21.662	0.000
1	0.000	0.504
2	0.002	1.182
3	0.002	-1.794
4	0.003	-1.651
5	0.002	1.184
6	0.001	1.121
7	0.001	-1.759
8	0.000	-1.829
9	0.000	-2.387
10	0.001	0.235
11	0.000	1.931
12	0.001	-0.815
13	0.000	-2.677
14	0.000	1.885
15	0.000	0.943
16	0.001	-2.093
17	0.001	-2.531
18	0.080	2.054
19	0.000	2.476
20	0.001	-2.245

(b) 9-phase excitation torque

Fig. 3.25 Electro-magnetic torque magnitude spectrum.

Another approach to obtain excitation torque without establishing excitation torque analysis by MATLAB program. It was numerical method which was based on usage of back EMF of machine model. The power of a electric machine could be expressed in mechanical and electrical. The equations were

$$P_m = T \cdot \omega \quad \text{Eq. 3.46}$$

and

$$P_e = V \cdot I, \quad \text{Eq. 3.47}$$

where  $P_m$  was mechanical power and  $P_e$  was electrical power with unit of *Watt*;  $T$  was torque with unit *Nm*;  $\omega$  was angular velocity with unit *rad/s*;  $V$  was supplied voltage with unit *Volt* and  $I$  was supplied current with unit *A*. When the loss was ignored, mechanical power would equal to electrical power. Hence

$$T \cdot \omega = V \cdot I. \quad \text{Eq. 3.46}$$

When the angular velocity was unit angular velocity, i.e. *1 degree/s*, the torque produce would be

$$T = V \cdot I. \quad \text{Eq. 3.46}$$

Using phase EMF in 9-phase model as unit EMF, the phase EMF of 3-phase model could derived by dividing 3-phase EMF by 9-phase EMF, shown in Fig. 3.26 and unit EMF of 9-phase was shown in Fig. 3.27. The unit phase current was also shown with the EMF plots. Substituting phase EMF and current into Eq. 3.46 and computing for all phases, the excitation torque could be estimated, shown in Fig. 3.28. The average estimated unit torque were 4.091 Nm for 3-phase and 4.262 Nm for 9-phase per unit speed. 9-phase machine were expected has at least 4% gain in torque over 3-phase machine when both machines were driven by same phase current. This estimated result matched with model simulation and the improvement of torque ripple was also noticeable according table of magnitude spectrums, shown in Fig. 3.29. Same with the model simulation results, 6th order harmonic was not present in 9-phase machine.

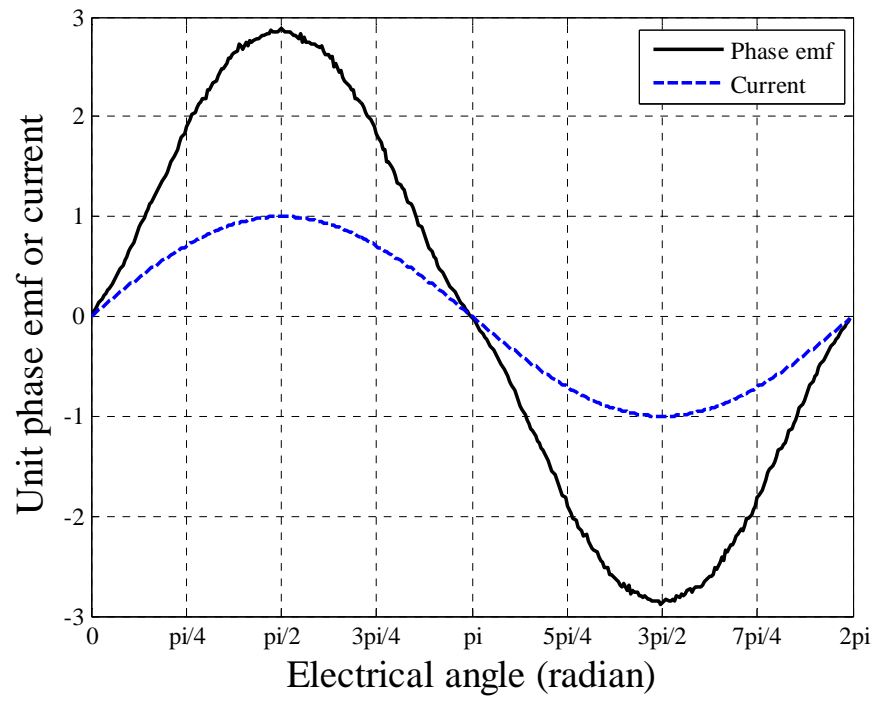


Fig. 3.26. Unit phase current and emf for 3-phase model.

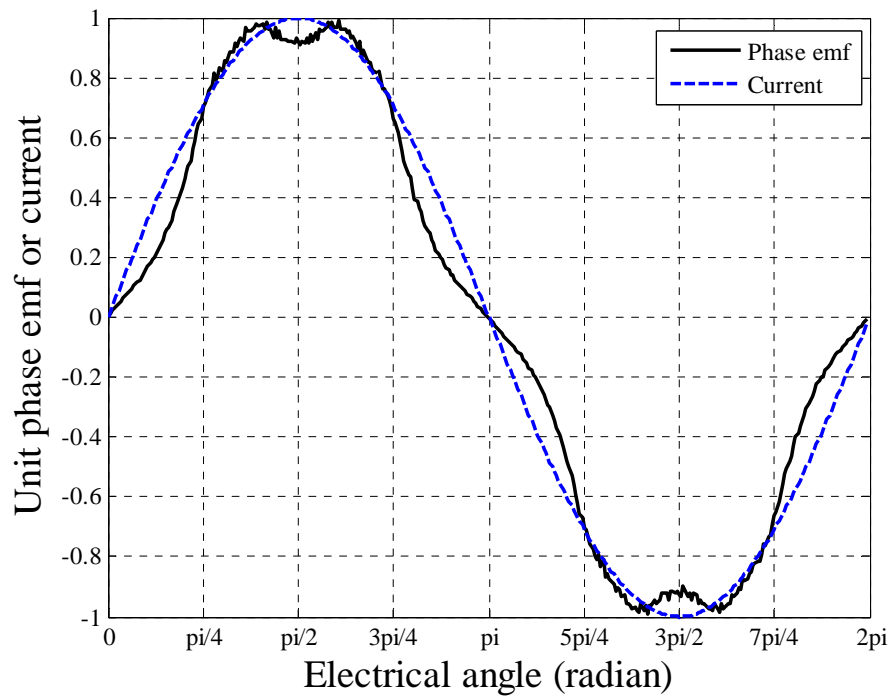


Fig. 3.27. Unit phase current and emf for 9-phase model.

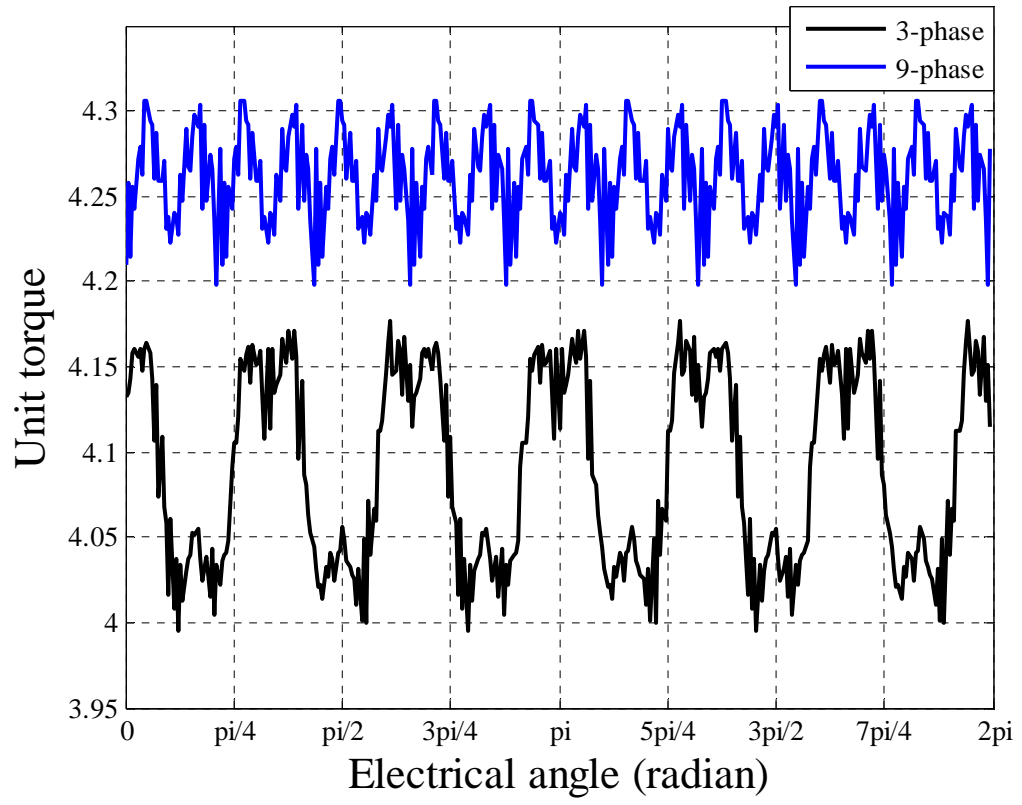
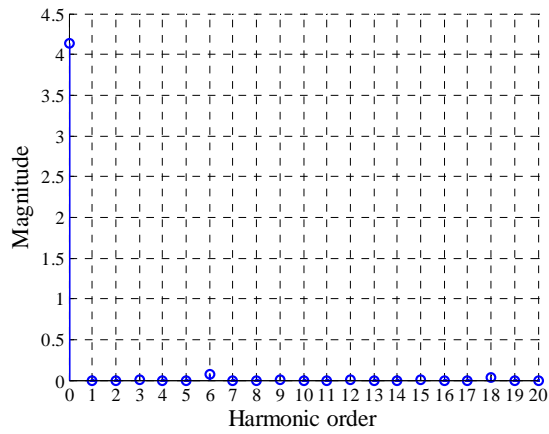
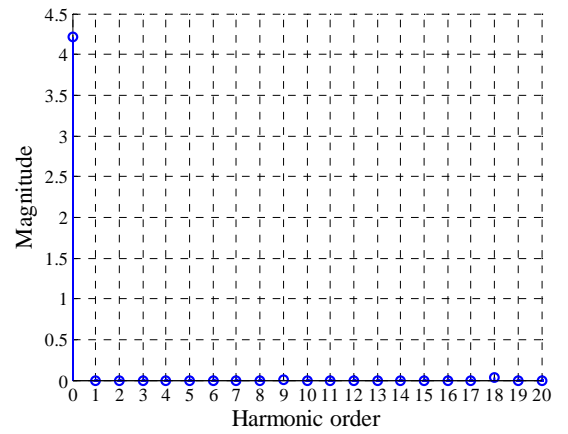


Fig. 3.28. Unit torque comparison between 3-phase and 9-phase model.



Harmonic order	Amplitude	Phase
0	4.091	0.000
1	0.000	-0.781
2	0.000	-0.713
3	0.000	0.372
4	0.000	2.660
5	0.000	3.056
6	0.072	0.036
7	0.000	-0.409
8	0.000	-0.921
9	0.001	1.007
10	0.000	-2.416
11	0.000	-3.000
12	0.000	-1.897
13	0.000	2.426
14	0.000	0.835
15	0.001	2.516
16	0.000	-1.708
17	0.000	0.070
18	0.028	-2.977
19	0.000	0.381
20	0.000	-2.551

(a) 3-phase



Harmonic order	Amplitude	Phase
0	4.262	0.000
1	0.000	-0.614
2	0.000	-1.209
3	0.000	2.407
4	0.000	1.402
5	0.000	1.376
6	0.000	2.842
7	0.000	2.058
8	0.000	1.354
9	0.001	-2.065
10	0.000	1.408
11	0.000	-2.828
12	0.000	2.424
13	0.000	-0.275
14	0.000	3.051
15	0.000	2.327
16	0.000	2.339
17	0.000	1.031
18	0.029	-3.013
19	0.000	-1.145
20	0.000	-0.670

(b) 9-phase

Fig. 3.29. Magnitude spectrum of estimated unit power or unit torque for two machine models. Phase in radian.

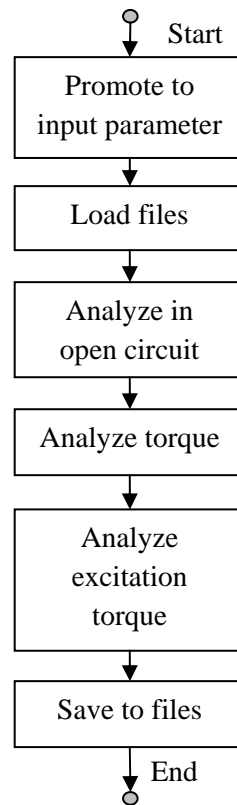


Fig. 3.30 Flow chart of automatic analyzing program.

### 3.5 Summary

According to the research established in this chapter, the bread loaf magnet design was the best among five designs, in terms of electromagnetic capability. Although in torque capability analysis the bread loaf design was not best, it was balanced in demagnetization analysis. The embedded bread loaf design was worth mentioning because it was the worst in demagnetization test. Hence it was suggested the demagnetizing state needed to be established in mesh element level in designing an embedded magnet machine, since the working point of the magnet would not indicate the demagnetizing state in particular region. The full pole-pitch arc design performed similarly with bread loaf design through the tests. The design could be adopted if the cost of manufacturing was not considered. Also for full pole-pitch and flat bottom design, since it produced the highest electrical torque during the test with slightly more volume. Full pole-pitch pseudo-radial magnetization design was not considered because it was difficult to manufacture and was the worst case in most of the test. With bread loaf design, the excitation torque was compared between 3-phase model and 9-phase model. 9-phase machine could be a better option than 3-phase machine in traction application, i.e. Electric vehicle, for the reason of higher torque output with same peak phase current, less variation in torque and lower rating requirement for driving circuit when similar output was required.

In software development, several program codes were created for different tasks in this chapter. It was integrated and developed further to simplify the analyzing process for a machine design. The automatic program was developed to investigate the electrical capability for a machine design with sinusoidal current and interference was not required during the analyzing. The flow chart of this analyzing program was shown in Fig. 3.30. To execute the automatic analyzing program, model file and current profile should be in the same folder as the analyzing program. The input required at the beginning of the program was number of phases, pole pair, precision of stepping, peak phase current and model file name. It would load files according to the analyzing task and save data with different names for further manipulating, i.e. check demagnetization in saved array. Also after each task, the saved data was exported to excel for quick access. The automatic analyzing program was shown in Appendix I.12.



## CHAPTER 4

### IRON LOSS ANALYSIS

#### 4.1 Literature Review

Many researchers have been taken place in recent years, to determine the energy loss of ferromagnetic material conducting flux, either in magnetic or electromagnetic field. From simple transformer to more complicated electric machine, by using different method, the prediction of the iron loss become more and more accurate. In this chapter, different approach to deduce iron loss were reviewed and discussed and selected methods were implemented for prototype machine.

Despite the mechanical related loss, i.e. windage and bearing friction, the losses in AC machine could be mainly accounted into two parts: copper loss, iron loss or core loss[63,114,115]. Copper loss is caused by resistance of winding material when it is conducting current and the energy loss was dissipated as heat which could be measured in Joules. The iron loss is the energy loss of ferromagnetic material when it conducting flux in a electromagnetic field. The loss was depended on property of the material and variation of electromagnetic field. Iron loss could be separated into three parts: hysteresis loss, eddy current loss and excess loss[63,114,115].

To understand hysteresis loss, fundamental physics about ferromagnetic material was useful. Using iron as an example, an iron block would be magnetic natural before any external magnetic or electric field applied as shown in Fig. 4.1(a), although individual particle carries minor charge and the direction and the magnitude of relative magnetic energy was complicated to state in vector format. The detail of material formation was not within the scope of this research and diagrams shown in Fig. 4.1 were for demonstrating how the particle react to the varying magnetic field. The most importantly, the iron block was in the state of magnetic relax and the resultant magnetic energy is zero. When a field applied, as shown in Fig. 4.1(b), the magnetic status of the particles would not be changed instantaneously with the external field or would not be magnetized by the field

instantaneously to be precise. Energy and time was required for the magnetization process to be accomplished on each particle and the magnetized iron block was shown in Fig 4.1(c).

When the direction of external field was changed, i.e. to the opposite direction of previous external field, the iron particles with settled magnetic state would experiencing resistance in external field direction, shown in Fig. 4.1 (d). The magnetic direction of particles would gradually shift toward the external field direction, as the field was continuously working on each particle. Similar to the process previously, it took time and energy to change the settled magnetic state to new states shown in Fig. 4.1 (e). The delay for the particle changing its states under external magnetic field was the fundamental of hysteresis of ferromagnetic material. The energy which was used for shifting particle direction and magnetic magnitude was the hysteresis loss. To represent this process, a hysteresis loop was shown in Fig. 4.2 where  $B$  was the flux density,  $H$  was magnetic field strength.

To start at point  $O$ , as field strength increase, ferromagnetic material in the field became magnetized and the flux density increase linearly toward to point  $a$ . At certain stage, the ferromagnetic material would not conducting anymore flux and reach its saturation state, i.e. further increase of field strength would not yield higher flux density. The demagnetizing process would follow the upper curve toward to point  $b$  when field strength direction reversed. eventually ferromagnetic material would be saturated again with negative flux. When the field strength increased again, the magnetizing process would follow the lower curve of the loop and reach point  $a$  to became saturated again. This loop was hysteresis loop and the area it disclosed indicated the amount of hysteresis loss. Also, the B-H characteristic of ferromagnetic material was not linear apart from magnetizing from magnetic neutral state.

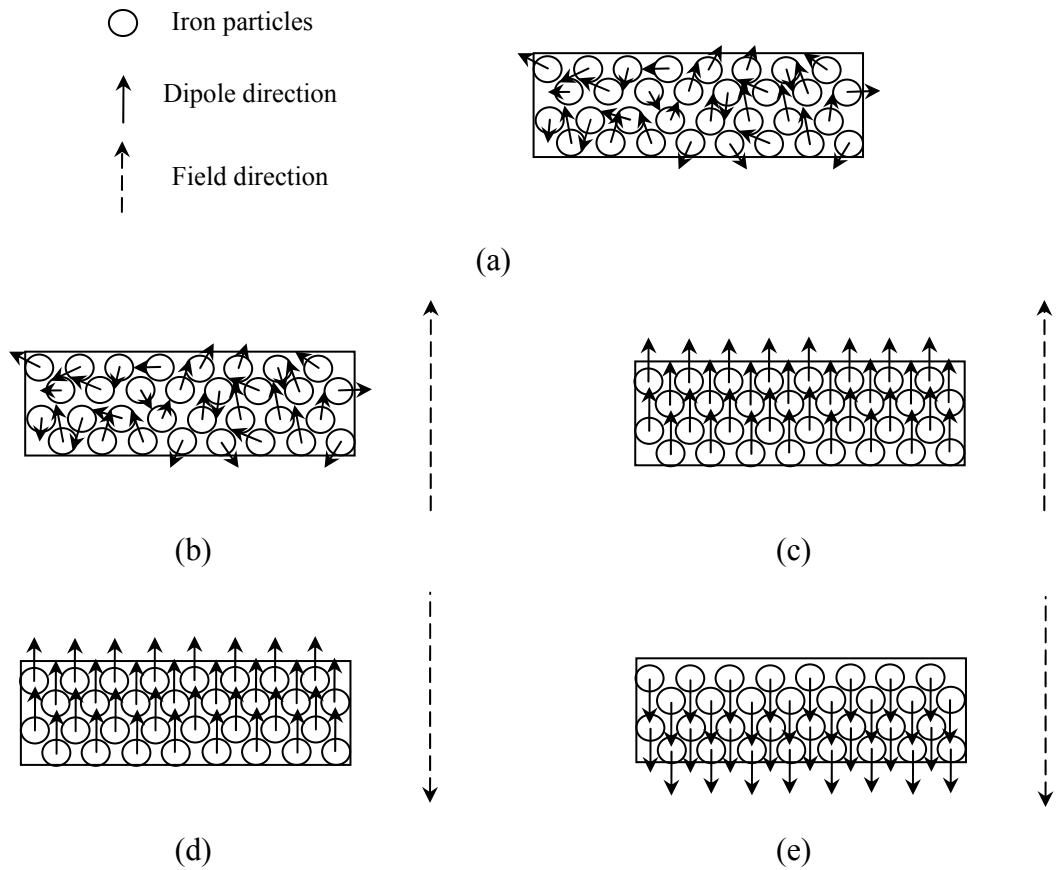


Fig. 4.1 Diagram of particle energy state changing in a magnetic field.

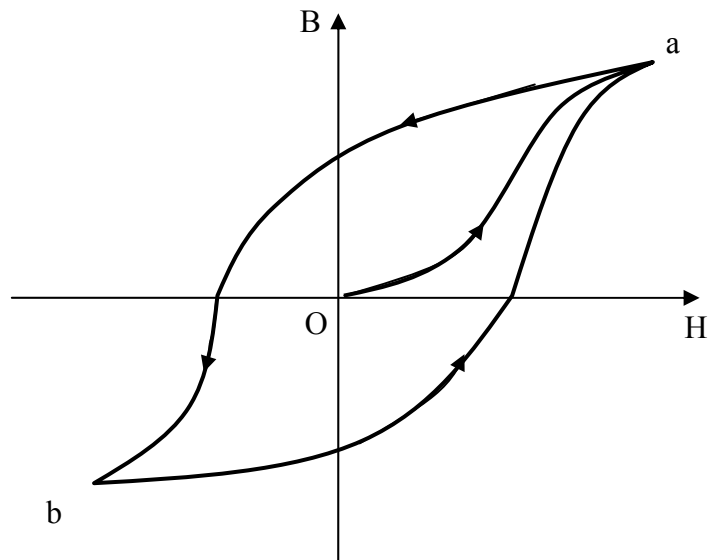


Fig. 4.2 Hysteresis Loop of ferromagnetic material.

According to Faraday's Law, Electromotive Force (EMF) would induced when a varying field passing a coil, which would be either electromagnetic or magnetic, shown in Fig. 4.3. Hence a current would conduct in the coil. If the field decrease, the direction of the current would reversed. In this case, the current was conducting in a coil and the conducting object

was not necessary to be a coil. In Fig. 4.4 (a), a rectangular ferromagnetic material, i.e. a iron block, was winded with coil. When current flow in the coil, electromagnetic field would be created. If the current was varying and as a result field strength was varying. EMF would be induced within the iron block. Because the iron block was conducting material, a small amount of current would conducting and flow in rectangular loop on the cross section of the iron block, on which was act as the coil in Fig. 4.3. This minor current was called eddy current. The magnitude and direction of the eddy current was depended on the variation of field, i.e. flux density,  $B$ . According to Lenz's Law, once eddy current was conducting on the cross section of the iron block, it would created a field opposite the varying field introduced by current in coil. Hence flux density in the iron block was less than it should be.

The reduction of flux density caused by eddy current yielded power loss of a system, i.e. an electric machine, and this energy loss was called eddy current loss. Eddy current loss could be determine by using the time based varying flux density which would be introduced later in this section. Eddy current loss could not be eliminated since the ferromagnetic material was used for conducting flux and it conducted current. In modern machine design, laminated ferromagnetic material was used instead of block, shown in Fig. 4.4, to minimized eddy current loss. Due to the area of cross section of lamination, the smaller area limited eddy current path and caused the increases in resistance of minor loop, which reduced eddy current.

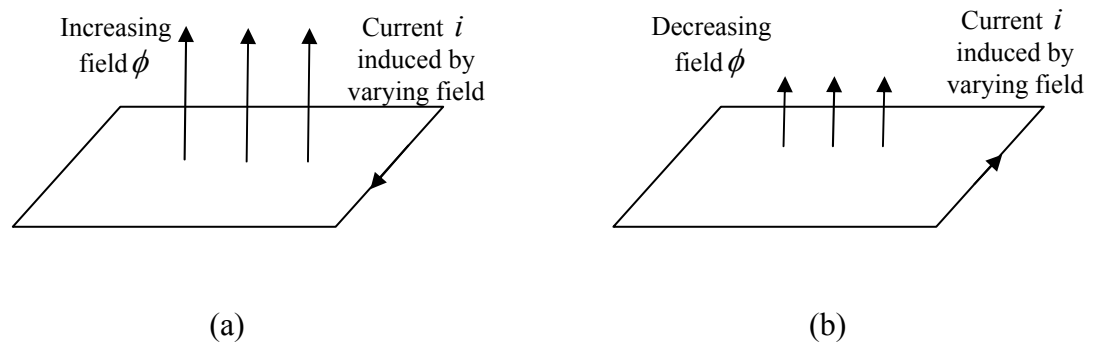


Fig. 4.3. Diagram of EMF introduced by field. (a) increasing field and (b) decreasing field.

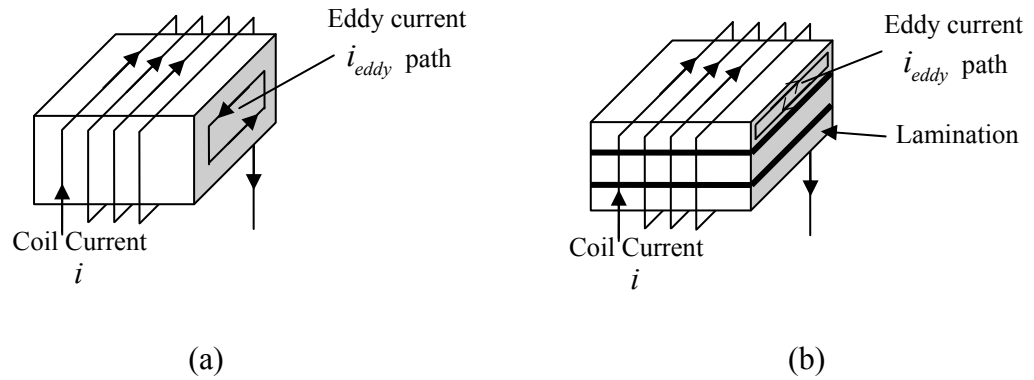


Fig. 4.4 Diagrams of eddy current: (a) in a block and (b) in laminated material.

Excess loss was the energy loss for two magnetic field carriers reach their new equilibrium after two field interacting with each other. It could be explained with an example. There two objects which carried magnetic energy were infinite far away from each other. They were in their own field and stable. When they approach toward each other and their fields were starting interacting at some stage. The field would naturally prevent the approaching and after the field became stabilised or interrelated, the interaction would prevent two objects departing. The excess loss was produced in the process of field interacting. A practical example would be cogging in permanent magnet (PM) machine.

In machine design aspect, the iron loss become important part when the motor driving by current with different waveforms and frequencies [105]. In this part of research, the iron loss was focused on and different methods to calculate were reviewed and compared. According to the fundamental of losses mentioned above, each of the loss production was related to the external field and flux density must be a major part of iron loss calculation.

Many researches were established on iron loss predication since it related to the efficient of a electric machine and also the reduction in loss would improve thermal environment [84]. The most importantly, the iron loss accounted a big margin of the total losses[71,74,105], hence the iron loss prediction was inevitable for every machine design engineer. The development in early days was focus on characteristic of ferromagnetic material, i.e. hysteresis loop [107]. The iron loss was determined by Steinmetz original equation, in which was including two terms: hysteretic loss and eddy current loss. The calculation was depending on coefficients derived from experimental data. It was the very first paper stress on the importance of hysteresis loop for ferromagnetic material. Because only loss due to alternating field was considered and excess loss and harmonic content was not counted, this method was not that accurate[63].The conventional method still provided a quick look of loss in a machine.

Iron loss prediction was well established in transformer area where the alternating field

was dominated, using Epstein test, i.e. transformer. The most common approach was using FEA method [69,71] and indicate difference in flux density at different area[58,63,71] and obtain the waveform. 2-D FEA method was widely used in machine modelling and it was proved that 2-D static FEA method was accurate for simulating flux distribution [70]. Based on the flux density waveforms, iron loss was calculated in three parts: hysteresis loss, eddy current loss and excess loss [71]. The accuracy of iron loss prediction was varied depending on if there were hysteresis loop was employed, there were flux density harmonic component consideration and different correction mechanism. The approach only depending on peak flux density [61] and assuming the waveform was pure sinusoidal would have relative poor performance in prediction [54]. The correction for hysteresis loss was mainly due to minor hysteresis loop [67, 71]. Using FEA method to obtain flux density waveform and separate the waveform into sinusoidal component for each harmonic [70]

$$P_t = P_h + P_d = P_h + P_c + P_{exc} \quad (4.1)$$

where

$$P_h = k_h f B_m^\alpha K(B_m) \quad (4.2)$$

$$P_c = \frac{\sigma d^2}{12} \frac{1}{T} \int_T B(t)^2 dt \quad (4.3)$$

$$P_{exc} = \frac{ke}{T} \int_T |B(t)|^{1.5} dt \quad (4.4)$$

Other methods was modelling the loss relayed on experimental data [66], using detail of hysteresis loop [67] which would not considered in this research and build hysteresis model, surface model on restrictive assumption [62].

The iron loss estimated based on sole alternating field was proved to be significant less than experiment measurement and the predict was improve by introducing the rotation loss due to rotational field [64, 65, 102]. Comparing with the measurement results, the conventional method which was based on peak flux density would predict half of the total loss and it improved with methods considering of harmonic components in flux density waveform. The improvement was mainly on the loss due to alternating field. The iron loss would be improved further when rotational loss was counted [64,102]. The loss prediction considering both loss due to alternating field (alternating loss) and rotational field (rotational loss) would be express as[64,102]

$$W_{i-rot} = W_{h-alt} + W_{e-alt} + W_{add-rot} \quad (4.5)$$

$$W_{add-rot} = \gamma \frac{L_{short}}{L_{long}} (W_{h-alt} + W_{e-alt}) \quad (4.6)$$

where  $W_{i-rot}$  is total iron loss under rotational field and  $W_{h-alt}$  is the hysteresis loss caused

by alternating field,  $W_{e-alt}$  is the eddy current loss;  $W_{add-rot}$  is rotational loss.  $L_{short}$  and  $L_{long}$  are the short and long axis of flux vector locus;  $\gamma$  was a constant and it was the rate of iron loss increment under rotational field to alternating loss which was required experimental data and it was difficult to obtain [64]. Similar approaches of rotational loss were also found in [65], which introduce correction based on flux vector locus and the loss factor  $\gamma$ .

The rotation loss was additive part for alternating loss equation. The area where the rotational field presents could be found by flux density plot of radial direction versus tangential direct. When the plot was narrow and nearly a line, the ratio of short axis and long axis of the ellipse was small and affect of rotational field was minimal. When the plot was a loop or circle, the rotational field need to be counted in, in order to improve the iron loss prediction. According to equation 4.6, when both of the ratio of short axis and long axis and  $\gamma$  were equal to 1, the alternating loss could be as much as alternating loss, which was 50% of the total loss. Hence the rotational loss should always account into iron loss calculation. Due to uncertainty of parameters in the rotational loss equation, it suggest that by simple adding the loss, which was assuming produced in alternating field [67], due to the tangential component and radial component of flux vector in a rotational field as correction for rotational loss would introduce error to the predication.

The iron loss methods could be catalogued into two schemes by the amount of knowledge about the ferromagnetic material [105].

- Mathematical approach based on Steinmetz Equation [107] and other improved equations [71,104,106,108,109,110,111].
- Hysteresis model approach based on experimental measurements of the material [112,113].

The first approach provided a good estimation of iron loss for sinusoidal or non-sinusoidal flux density in ferromagnetic material with small amount of information about the materials themselves and the coefficients using in those equations could be obtained by re-plotting manufacture data sheets [104,114]. The limitation of this approach was also obvious, which was the frequency range of material loss on the data sheets [110, 111]. Apart from the frequency content, the errors may be broadly introduced into mathematical approach because the assumptions made during the iron loss prediction, e.g. the linear mechanism to obtain coefficients and unified flux density areas. Although Steinmetz Equation was greatly improved later [70,104,106,108,109,110,111] by introduce corrections and improved methodology of iron loss calculation, to fill the 20% difference gap [107] between estimated loss and measured loss was still the desire by machine design engineers.

For the Hysteresis model approach, it provides a method to achieve higher accuracy in iron loss prediction and overcame the limitation on frequency range [105,111,112]. The drawback for this approach was the complexity of the model and required more information about the Ferro- or Ferromagnetic materials to be able to determine the coefficients for the model [105].

For this particular iron loss prediction, the Steinmetz equation based approach is appropriate, because the detailed knowledge about 'Transil335' British steel were not able to obtained and there not enough information to determine coefficient of a Hysteresis model. The most importantly, it is not appropriate to treat electromagnetic complex system as a 'black-box' and there no benefit in machine designing, machine control and problem locating in machine design point of view. Referring to the previous development papers which were listed in Table 4.1 and 4.2 [58-97] (also previous iron loss iron loss review [106]), the different techniques Steinmetz equation based mathematical approach were listed in Table 4.3.

Methods of Bertotti's and Howe's were proposed for iron loss calculation for this study, because of the validation of these two methods were sensible and many aspects of machine iron loss were discussed, which include flux density superposition location and harmonic content. Also FEA was employed by both methods for field solution. For handling iron loss in a rotational field, a paper [63,64,102] was recommended.



Reference number	Machine type	Analysis method	Hysteresis loop consideration	Practical Validation	Comments
[58]	PM	Num.	No	No	Method compare with FEA analysis. How good the FEA method is when H loop was ignored?
[59]	PM	Num.	No	Yes	Loss coefficient variation in time and frequency domain. Harmonic part;No FEA compare.
[60]	IM	Num.	No	No	Single line BH curves for different frequency.
[61]	PM	Num.	No	No	Focused on other iron loss parts.
[62]	PM	Num., FEA	yes	yes	Loss surface model compare with harmonic model
[63]	PM	FEA	yes	Yes	Magnets on stator.
[64]	PM	FEA	No	Yes	Core iron loss due to harmonics and flux vector rotation.
[65]	IM	Num., FEA	No	No	Ironloss coefficient estimation without considering B.
[66]	SR	FEA	No	No	No even consideration about ironloss-temp.
[67]	PM	FEA	Yes	No	Focus on effect of ironloss in motor control.
[68]	PM,SR	FEA	Yes	No	The hysteresis loop are negligible for unload machine. Ripples in control current may introduce more iron loss.
[69]	IM	FEA	No	No	Hysteresis loop peak value used for 3D model
[70]	PM	FEA	No	Yes	Core losses analysis during high speed operation.
[71]	PMDC	FEA	Yes	Yes	Compare ironloss under different conditions:open-circuit and full load.
[72]	PM	FEA	No	Yes	Eddy current losses, stationary +Ref.8
[73]	DC	Num., FEA	No	No	Increases of eddy current loss due to PWM switch is significant compare to AC supply. Mentioned nonlinear B-H curve in ref[9].
[73]	PM	FEA	Yes	Yes	Iron loss prediction model.
[75]	PM	FEA	Yes	Yes	Ironloss calculation using System simulator-FEM software. PWM drive, FEM software FLUX; 'Electrical sheet'. Non sinusoidal voltage cause additional current harmonics.
[76]	PM	FEA	Yes	Yes	Rotor eddy current loss was marginally influence the torque behaviour and rotor design (solid or laminated) was more significant on machine efficiency for Fractional slot concentrated winding PM.
[77]	--	FEA	No	Yes	Polarized flux density is significant in core loss prediction. Standard prediction has major error in predicting loss in rotational field.

Table 4.1 Table of reference reviewed on Iron loss calculation, Part 1.

Reference number	Machine type	Analysis method	Hysteresis loop consideration	Practical Validation	Comments
[78]	PM	FEA	No	No	Simulated the iron and magnet loss for different magnet design.
[79]	IM	Num., FEA	No	No	Slot shape design is relevant to ironloss due to the harmonics.
[80]	PM	FEA	No	Yes	Treat ironloss as block box. Prediction based on approximation.
[81]	IM	FEA	Yes	Yes	A model for estimating accurate high frequency harmonic ironloss, based on Eddy current in 'electrical steel sheets'.
[82]	SR	FEA	No	No	Calculated ironloss based on FEA.(fft,super elements)
[83]	IM,PM	Num.	Yes	No	Ironloss equivalent current expressed in complex inductance. Rotating field.
[84]	PM	FEA	Yes	Yes	Investigated coreloss due to hysteresis loop and non-sinusoidal flux will cause substantial harmonics. 'Epstein square'
[85]	PM	--	No	Yes	Soft magnetic composites has greater ironloss density than silicon iron steel.
[86]	DC	Num.	Yes	Yes	Discussed various losses in machine.
[87]	General	FEA	Yes	Yes	Ironloss using stepping FEA with approx of excess loss.
[88]	IM	3D-FEA	No	Yes	Electrical steel sheets skin effect and excess loss.
[89]	IM	FEA	Yes	Yes	FEA including ironloss and excluding.
[90]	SR	FEA	Yes	No	Dynamic model of SR. equations for different loss parts.
[91]	PM	FEA	No	Yes	Eddy currents in magnets are resistance limited and not in back iron. Isolated magnet machine has less eddy current density.
[92]	PM	FEA, Num.	No	No	Two term expression with 3D eddy current analysis is suitable for estimate higher order harmonic iron losses.
[93]	--	Num.	Yes	yes	Dynamic hysteresis analysis and possible loss reduction.
[94]	PM	FEA	Yes	No	Ironloss does not affect operating point. Hysteresis losses are negligible from field solution .
[95]	PM	FEA	No	Yes	Ironloss become more significant in field weakening region.
[96]	PM	Num.	Yes	Yes	Iron loss estimation method
[97]	PM	FEA	No	Yes	Reduce ironloss by varying magnets and slot design.

Table 4.2 Table of reference reviewed on Iron loss calculation, Part 2.

#### 4.1.1 Measured results and comparison of equations for alternating losses.

Method	Equation	Implementation
Classic Steinmetz [107, 114, 115]	$H = H_1 + H_2$ $H = \eta B^{1.6} + \varepsilon NB^2$	Experiments carried out on transformer. Flux density waveform restrict within sine wave. The coefficients $\eta$ and $\varepsilon$ were obtained by linear fitting with experimental results.
Slemon [74,75]	$P_c = P_t V_t \bar{P}_{et} + \rho_i V_y \bar{P}_{ey}$ $P_c = P_t k_h \omega_s \left[ V_t \hat{B}_t^n + V_y \hat{B}_y^n \right]$ $+ \frac{4}{\pi} \rho_i k_e \omega_s \left[ \frac{2r}{pw_t} V_t^2 + \frac{V_y \hat{B}_y^2}{\alpha} \right]$	SMPM machine model was separated into two areas (tooth and yoke) in which the flux densities were considered unified. Non-sinusoidal and sinusoidal waveforms of flux density were considered. Coefficients were calculated with the physical dimensions of the machine and its speed. Harmonic contents not included.
Bertotti [92,109]	$W_t = W_h + W_c + W_e$ $W_{hr} = [1 + c(r-1)] W_{ha}$ $W_c = \left( \frac{\pi^2 d^2 f^2}{6 \rho \delta} \right) \sum_{n=1}^{\infty} n^2 [B_{nx}^2 + B_{ny}^2]$	Experiment carried on Induction machine and Epstein test for loss measurements. Model was separated into three areas. Assumption was made on excess loss and equal to classic loss (core loss). Harmonic contents were included. 20% different between prediction and measurements. FEA

Table 4.3 Alternating loss equation, part 1.

<p>Howe [58,63,71]</p>	$P_t = P_h + P_d ;$ $P_h = k_h f B_m^\alpha K(B_m),$ <p>where <math>K(B_m) = 1 + \frac{0.65}{B_m} \sum_{i=1}^n \Delta B_i</math></p> $P_d = P_c + P_{exc} ;$ $P_c = \frac{\sigma d^2}{12} \frac{1}{T} \int_T B(t)^2 dt ;$ $P_{exc} = \frac{ke}{T} \int_T  B(t) ^{1.5} dt$	<p>SMPM machine model was separated into five areas. Driving model also built. Some coefficient obtained from single-sheet ‘dc’ test. Harmonic contents were included.</p> <p>Induction waveform minor loops considered and correction was given when minor loop existed.</p> <p>FEA</p>
<p>Modified Steinmetz Equation (MSE)[104]</p>	$\dot{B} = \frac{1}{\Delta B} \oint \frac{dB}{dt} dB ,$ $\Delta B = B_{\max} - B_{\min} ,$ $\dot{B} = \frac{1}{\Delta B} \int_0^T \left( \frac{dB}{dt} \right)^2 dt$ $f_{eq} = \frac{2}{\Delta B^2 \pi^2} \int_0^T \left( \frac{dB}{dt} \right)^2 dt$ $w_v = C_m f_{eq}^{\alpha-1} \hat{B}^\beta , \text{ with } \hat{B} = \frac{\Delta B}{2}$ $\rho_v = \left( C_m f_{eq}^{\alpha-1} \hat{B}^\beta \right) f_r$	<p>Switch Reluctance Machine. Average flux density over whole machine and no energy separation. Dynamic hysteresis models by replace the frequency in classic Steinmetz equation by physical loss parameter dM/dt. dM/dt is proportional to the rate of change of induction dB/dt.</p> <p>No flux density area and no separation of different losses.</p> <p>FEA</p>

Table 4.4 Alternating loss equation, part 2.

#### 4.1.2 Equations for rotational loss.

M.Enokizono [99]	$P_r = \frac{2\pi}{\rho T^2} \int_0^T (H \times B) dt$ $P_r = \frac{1}{\rho T} \int_0^T \left( H_x \frac{dB_x}{dt} + H_y \frac{dB_y}{dt} \right) dt$	<p>P is the mass density; T is the length of period. Rotational loss caused by phase difference between magnetic flux density (B<sub>x</sub>, B<sub>y</sub>) and magnetic field strength (H<sub>x</sub>, H<sub>y</sub>). P<sub>t</sub> is the total power loss in x and y direction. For pure rotational field, P<sub>r</sub> = P<sub>t</sub>.</p> <p>Based on experimental measurement of B and H.</p>
K.Atallah [101]	$P_r = \frac{\omega}{T} \int_T (\vec{H} \times \vec{B})_z dt$ $= \frac{1}{T} \int_T \frac{d\theta}{dt} (\vec{H} \times \vec{B})_z dt$ $P_\alpha = \frac{1}{T} \int_T  \vec{H}  \frac{d \vec{B} }{dt} \cos \alpha dt$	<p>dθ/dt is the instantaneous angular velocity of flux density vector, different from Enokizono's method, which depends on angular velocity ω. P<sub>a</sub> is the power loss due to alternating flux density.</p> <p>Measurements of B and H? H is the winding field strength.</p>
Bertotti; L.ma, M. Sanada [102]	$W_{i-rot} = W_{h-alt} + W_{e-alt} + W_{add-rot}$ $W_{add-rot} = \gamma \frac{L_{short}}{L_{long}} (W_{h-alt} + W_{e-alt})$ $= \gamma \delta_e (W_{h-alt} + W_{e-alt})$ $W_{i-rot} = \sum_{k=1}^n \sum_{i=1}^l g_i \left\{ \begin{array}{l} \varepsilon_h \left( \frac{kf}{100} \right) B_{mk-i}^\alpha \\ + \varepsilon_e \left( \frac{kf}{100} \right)^2 B_{mk-i}^2 \end{array} \right\}$ $\times (1 + \gamma \delta_{ek-i})$ $B_{mk-i} = \frac{L_{long-k-i}}{2}$	<p>W<sub>i-rot</sub> is total iron loss under rotational field and W<sub>h-alt</sub> is the hysteresis loss caused by alternating field, W<sub>e-alt</sub> is the eddy current loss; W<sub>add-rot</sub> is rotational loss. L<sub>short</sub> and L<sub>long</sub> are the short and long axis; γ is the rate of iron loss increment under rotational field to alternating loss. FEA method, no need for experimental measurements for B and H. Suitable to use.</p> <p>The Bertotti's computation is more detail in numerical modelling.</p>
Carlos A.Hernandez- Aramburo [65]	$P_{iron} = P_{hys} + P_{eddy} + P_{anom}$ $P_{hys} = k_{1hys} f B_{pk}^{k_{2hys}} \left( 1 + \frac{k_{3hys}}{B_{pk}} \sum_i^N \Delta B_i \right)$ $P_{eddy} = \frac{\sigma \delta^2}{12} \frac{1}{T} \int_T \left( \frac{dB}{dt} \right)^2 dt ;$ $P_{anom} = \frac{k_{anom}}{T} \int_T \left  \frac{dB}{dt} \right ^{\frac{3}{2}} dt$	<p>Similar to Bertotti.</p> <p>N is number of minor loops; k<sub>1hys</sub> is the hysteresis loss linear constant, k<sub>2hys</sub> is hysteresis loss exponential constant and k<sub>3hys</sub> is hysteresis loss minor loops constant; k<sub>anom</sub> is anomalous loss constant.</p> <p>Although very detail in loss and additional correction were given, those constants are not provided.</p>

Table 4.5 Rotating loss equation.

## 4.2 Implementation

### 4.2.1 Equation and parameter

After consideration with the material and method, Howe's method was selected for calculating iron loss in alternating field [58,63,71]. This was due to the availability of material data. Derived from table 4.4, the equation was modified as table shown below.

Howe [1] [6] [14]	$P_{total} = P_{eddy} + P_{exc} + P_{hys}$ <p>And when <math>B(t) = A \sin(2\pi f \cdot t + \varphi)</math></p> $P_{eddy} = \frac{\sigma}{12} \cdot \frac{d^2}{\delta} \cdot \frac{1}{T} \cdot \int_T \left( \frac{dB}{dt} \right)^2 \cdot dt$ $= \frac{\sigma}{24} \cdot \frac{d^2}{\delta} \cdot \omega^2 A^2$ $= \frac{\sigma}{24} \cdot \frac{d^2}{\delta} \cdot (2\pi f)^2 A^2$ $P_{exc} = \frac{k_e}{T} \int_T \left  \frac{dB}{dt} \right ^{1.5} dt = 8.67 \times k_e f^{1.5} B_m^{1.5}$ <p>for a sinusoidal flux density waveform;</p> $P_{hys} = k_h f B_m^\alpha$	<p>where A is the magnitude peak flux density for nth order harmonic; <math>\omega</math> or <math>2\pi f</math> is the harmonic number; <math>\varphi</math> is the phase shift; <math>k_h = 0.0155</math> ;</p> <p><math>\alpha = 2.45</math> ; <math>k_e = 1 \times 10^{-4}</math> ; resistivity <math>\sigma = 0.5 \times 10^{-6} \Omega^{-1} \cdot m^{-1}</math> or conductivity <math>\sigma = 2 \times 10^6 S / m</math> ;</p> <p><math>d = 3.5 \times 10^{-4} m</math> ;</p> <p><math>\delta = 7.65 \times 10^3 kg / m^3</math> ; <math>T = 2\pi</math> ;</p> <p><math>a = 2.12471</math> ; <math>b = -0.19674</math> ;</p> <p><math>C_h = 0.025199</math> ; <math>C_e = 0.000107</math> .</p>
Table 4.6 Implemented alternating loss equation.		

Bertotti's method was intended to deduce the iron loss when rotational field was represented. To produce preliminary comparison and simplified the calculation, two components of rotational losses were summed up, of which direction were perpendicular to each other. Referring to the conclusion of reference M.Enokizono [99], the rotational iron loss derived from equation should be greater than the sum of two components. Further implementation of Bertotti's method would be taken out for more accurate iron loss prediction. The equation included in Bertotti's method was shown in table below and the total power loss,  $W_{i-rot}$  was separated into sum of alternation loss and rotational loss:

Bertotti; L.ma, M. Sanada [102]	$W_{i-rot} = W_{h-alt} + W_{e-alt} + W_{add-rot}$ $W_{add-rot} = \gamma \frac{L_{short}}{L_{long}} (W_{h-alt} + W_{e-alt})$	<p><math>W_{i-rot}</math> is total iron loss under rotational field and <math>W_{h-alt}</math> is the hysteresis loss caused by alternating field, <math>W_{e-alt}</math> is the eddy current loss; <math>W_{add-rot}</math> is rotational loss. <math>L_{short}</math> and <math>L_{long}</math> are the short and long axis; <math>\gamma</math> is the rate of iron loss increment under rotational field to alternating loss.</p>
Table 4.7 Implemented rotating loss equation.		

#### 4.2.2 Simulation control

Model simulation was implemented using MATLAB program work along with Lua script from FEMM. In old version of FEMM, Lua script executed alone and it was difficult to accomplish some complicate tasks, i.e. save solution file for each program loop. If data of certain position required within simulation circle, Lua script needed to be rerun with corresponding parameters. In version 4.2, Octave package was introduced, which provide a way to run Lua script within MATLAB environment, and it simplified the model simulation greatly. The programs to establish machine model simulation were programmed using MATLAB and could be separated to three parts: simulation control, results analyzing and data manipulation. In these programs, the functions were divided into two parts: Lua script manage FEMM model and its solution files and MATLAB commands associated with data and other parameters.

Codes for simulation control were designed for manage the model running during the tests, e.g. model modification, rotated speed, location to load or save FEMM files and variables to record. The scenario of this part of programme was shown in Fig. 4.5 below. The simulation program should be able to read and load the FEMM model file and rotate machine model at desired speed. At the same time, the solution file from FEMM solver should be saved for each step during the rotation. Furthermore, data from simulation were recorded synchronously for reference purposes, i.e. angular position.

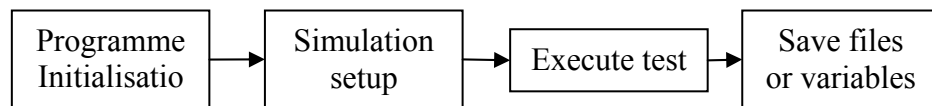


Fig. 4.5 Scenario for simulation program.

In the stage of initialisation, storage for parameters required a cleanup and prevent any unexpected effect occurred by data from previous test. In order to call Lua script in MATLAB, a path of relevant functions was added in this stage. When setup the simulation model, Lua scripts were used to modify the model with desired parameters, including rotor position of the model. Storage arrays were also setup for recording the variables during the program execution. After model set up, a for loop were created to rotated the rotor of machine model with equal angle increment for a full electric cycle. Within each executed loop, preliminary data about machine model was recorded into variable array and the solution file of the machine model was saved for further investigation at the end of each

loop. The name of these solution files were same in the increment sequence associate with the loop number or rotor position of machine model. For some tasks, further model modification may be required for each loop using Lua scripts: changing current, changing winding and position of reading taken. At the end of simulation control program, variable arrays were saved as data file for further manipulation. Table. 4.8 was a structure diagram of a variable array and for different tasks, the simulation control program would be modified to accomplish the tests and an example flow chart of simulation control program was shown in Fig. 4.6.

Loop number	Angular position	Torque	Current	...
1				
2				
3				
⋮				

Table 4.8 Diagram of variable array.

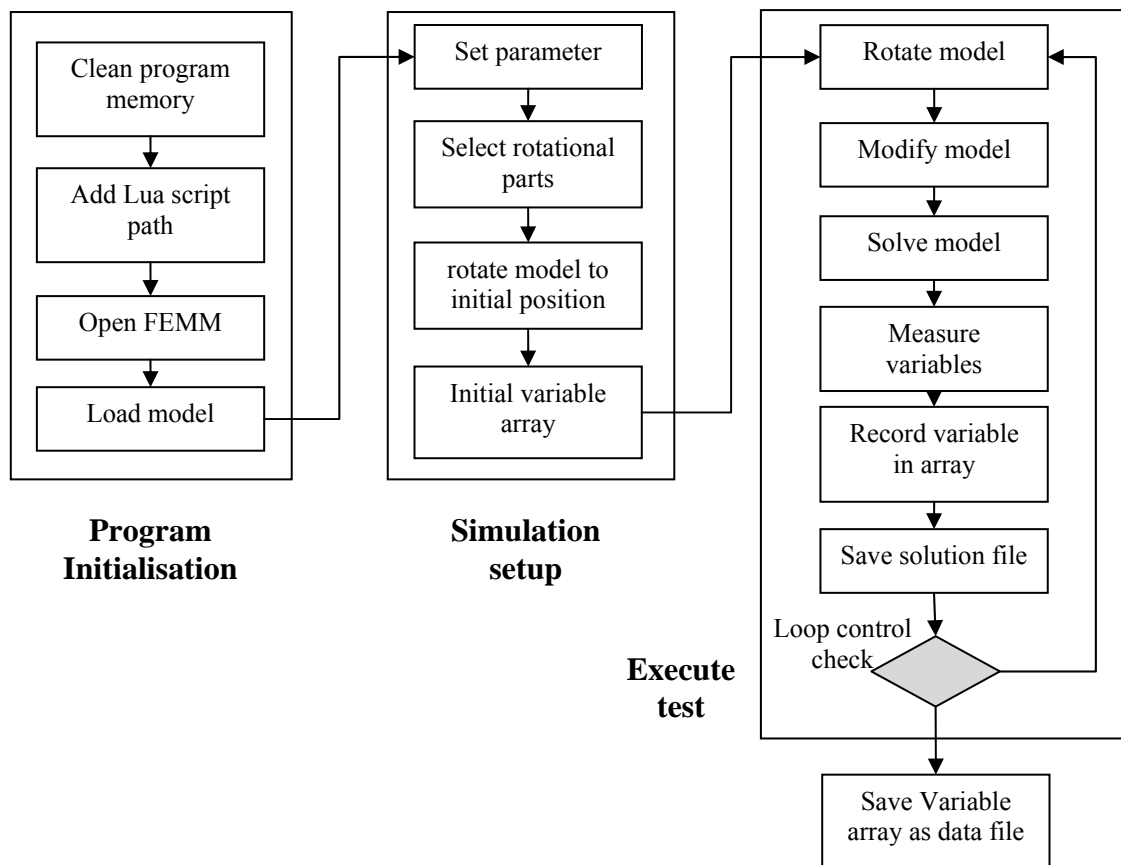


Fig. 4.6 Flow chart for example simulation control program.



### 4.2.3 Solution analysis

With static solutions of machine model, generated by model control program, further analysis were required to obtain information about machine model during its testing cycle. The objectives of this part of program were converting solution files to user-friendly data array and catalogue the data for later use. Fig. 4.7 shows the concept of solution analysis program and it could be treated as a black box. This program should return necessary data for machine model calculation, i.e. current, flux density and torque, after processing all the solution files for whole testing cycle. There would be one or few electrical cycles contained in testing cycle, depending on the specific test.

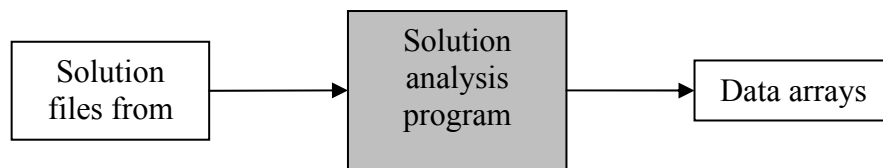


Fig. 4.7 Concept for solution analysis program.

FEMM is a aggregate of material database and electromagnetic algorithm and does not contain any information about the machine model. The information about machine model were all contained in its model file and solution file with same layout, i.e. dimensions, node coordinator, material type and mesh detail. The consist format of FEMM files when they were opened as text file, also provides an alternative way to modify machine model. The only difference between two types of FEMM files was the last section, 'solution', which was only contained in solution files and not in model file.

The first task of solution analysis program was indicate different sections and located the desired parts. Then the useful information about the model should be extracted and saved to data array in MATLAB. Normally, there were number of solution files for a machine model simulation and the data extracted from each file also should be stored into arrays. In Fig. 4.9, a brief flow chart for solution analysis program. The initialization of was similar to solution analysis program was similar to simulation control program, the process were: memory cleanup, adding lua script, open FEMM and initial data arrays. Unlike the simulation control program, the file loading step was executed within the loop of program to ensure that all the solution files would be processed. The rotor position associated file name was also guaranteed the sequence of file processing. There were two ways to extracting data from solution files: reading from the content of solution files and using Lua script to obtain data. The difference between these two methods was that the content

cataloguing method was more fundamental than the Lua script approach. The Lua script method was focus on obtaining the processed data from machine model, i.e. flux density of particular surface of the model. On other the other hand, content cataloguing method only provided row information about the model, i.e. mesh details. These two methods were complementary to each other and for different tasks, either or both of these methods were employed. To extract data from content of solution files, these solution files were converted to the format which was readable for MATLAB at first. Then the starting row number of desired data, i.e. solution section, was indicated using search command in MATLAB. The next task would be cataloguing the numerical data from different sections with right titles and store them in data arrays. This data storing process was also similar to simulation control program, in which the row number was associated with the loop number of solution analysis program. The Lua script approach to obtain data was more straight forward and the only information required were the coordinate of the node, which used to built surface and block of the machine model. At the end of analyzing program, the data arrays was save as data file for further manipulation. Fig. 4.8 shows the detailed flow chart for the solution analysis program with both data extracting method implemented. This analyzing program would become more complicated as the simulation proceeding.

Model setting
:
Block definition
:
BH Curve data
:
Circuit Property
:
Points Property
:
Segments Property
:
Blocks Property
:
Solutions
:

Fig. 4.8 Format of solution files when opened as text file.

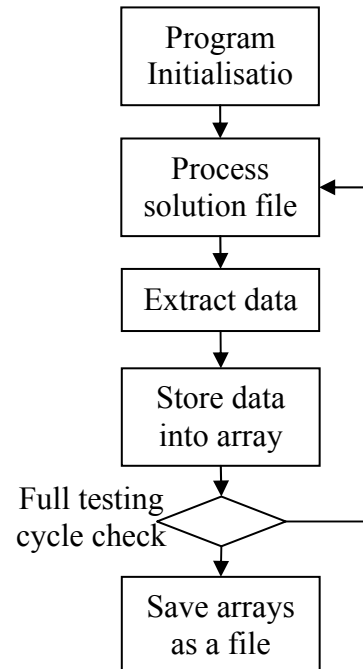


Fig. 4.9 Brief flow chart for solution analysis program.

#### 4.2.4 Data manipulation

Graphic way to express numerical data is normally a good way to looking for trends or patterns of testing object, so a data manipulating program was created to convert numerical data. The task of this program was gathering all the data extracted from FEMM solution files, and re-representing in graphic format, i.e. plots, for which later observation and comparison. The concept for data manipulation program was shown in Fig. 4.10 below. Tables also used when detail information about the plots required in some cases.

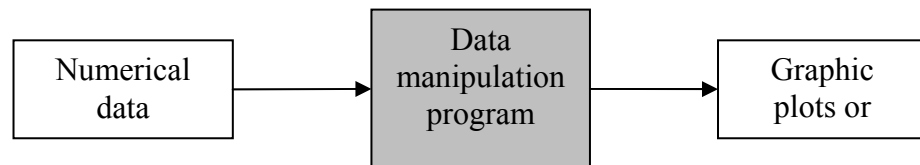


Fig. 4.10 Concept for data manipulation program.

With the data saved by solution analysis program, the main issue was to locate the desired data block and plot them out using MATLAB. After load the data, all data blocks could be assigned referring to data title in solution analysis, so all individual data blocks could be recognized and used for data plotting. For some situations, manipulation was required for the data, i.e. unit correction. The plots were generated and saved in the same format for particular test, in order to convenience the process after. Loop process also employed when multiple plots required. In Fig. 4.11, the flow chart for an example of data manipulation program was shown.

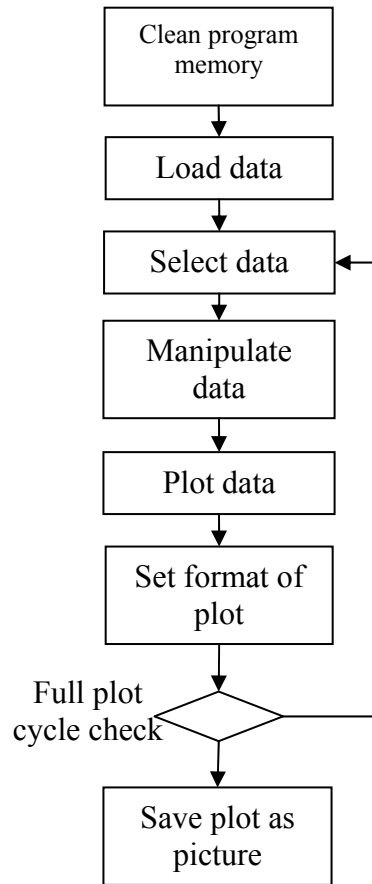


Fig. 4.11 Flow chart for an example data manipulation program.

Fast Fourier transform (FFT) was employed when further analysis was required for graphic plots, for example, to realize the frequency content and magnitude information of a flux density waveform. FFT transform a series of time-domain data to frequency-domain series, using Discrete Fourier Transform (DFT) algorithm. The output of FFT in MATLAB was a series of complex numbers correlating to the input series sequence. These imaginary numbers contain both amplitude and phase information of a sinusoidal component of original data series and the harmonic order is related to the sequence of the imaginary series. With FFT function, a data series could be re-represent into sum of sinusoidal expressions, which helps understanding the periodical content of data series. Furthermore, the sinusoidal expression could be used in further simulation to regenerate data series.

### 4.3 Machine iron loss analysis

The iron loss analysis was established for different parts of machine separately. To identify the flux density in different parts, several testing blocks was created crossing the model , which were shown in Fig. 4.12. The testing blocks were aimed to indicate different areas with different alternating and rotating flux contribution [58,63,71]. By employing the analyzing program developed previously, the flux density in each testing block could be extracted while stepping FEMM model. The result file analyzing program for this section was updated since the testing blocks were re-indexed individually, shown in Appendix I.9. The full set of flux density results in open circuit, 1000A-t 3-phase model and 1000A-t 9-phase model were shown in Appendix III.2, III.3, III.4 respectively.

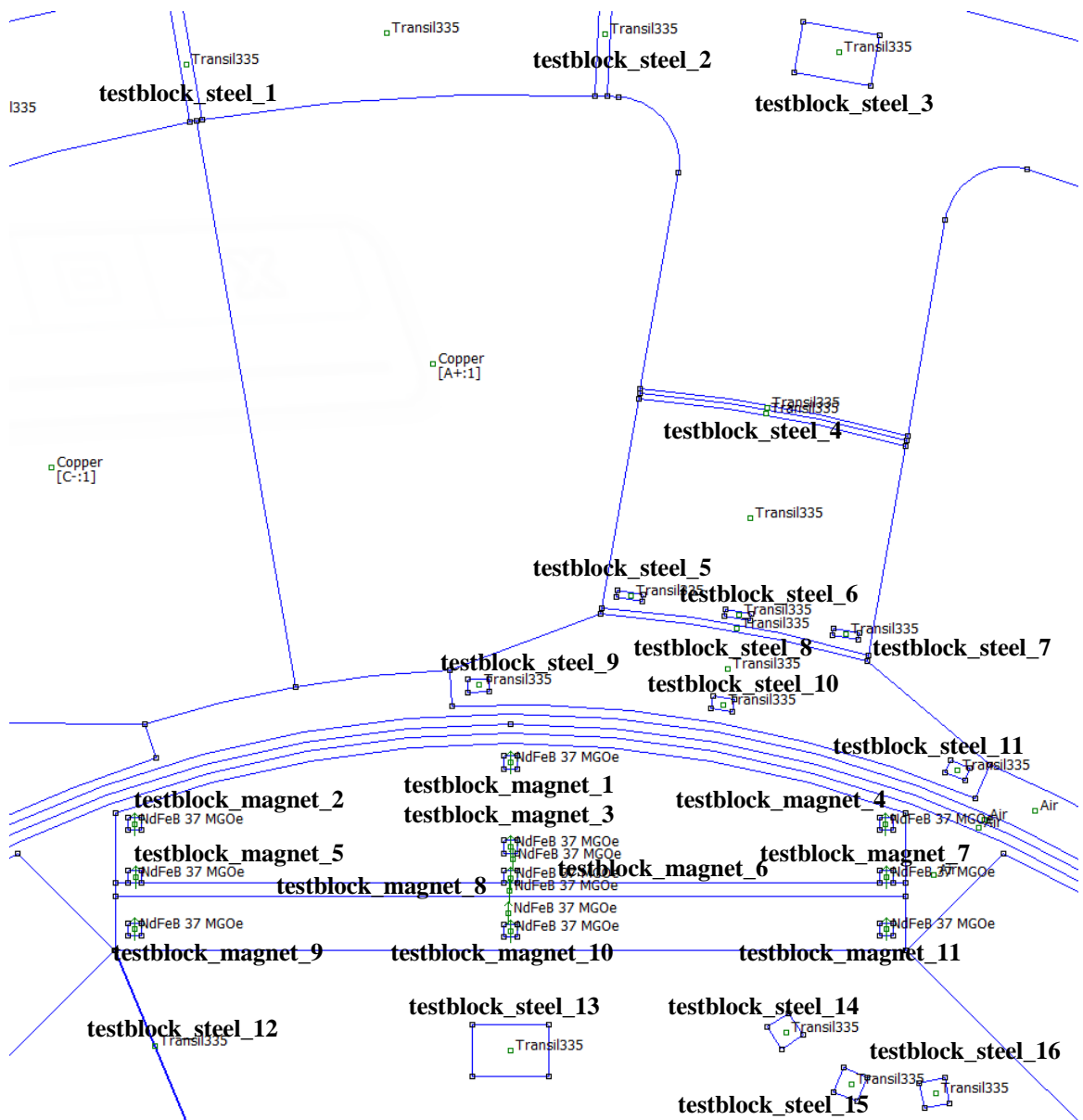
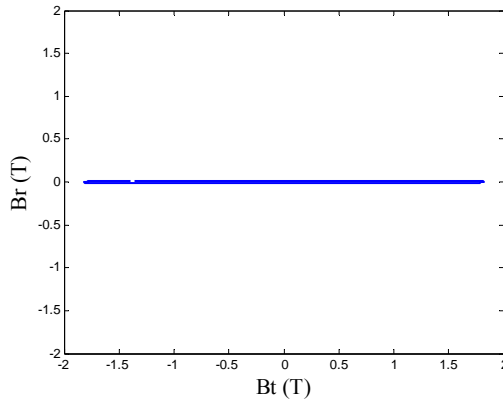


Fig. 4.12 Diagram of testing blocks.

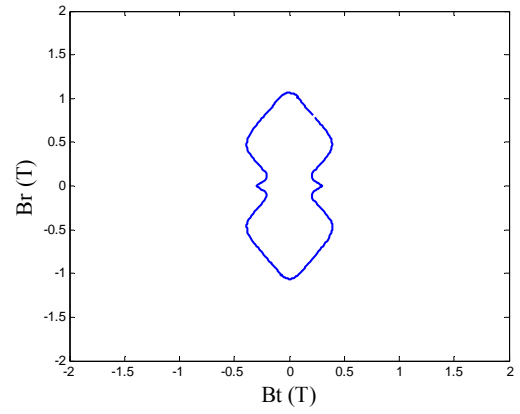
The flux density plot in most of testing blocks on stator were sinusoidal and DC on rotor. By reporting the flux density into two components, the rotational field were reviewed, shown in Fig. II.1-II.3 in Appendix III. Hence the area which effected by rotating field could be defined on machine model, for example, steel testing block 3,5,6,7,8,9,10 and 10 and the area they were represented. For the testing blocks on rotor, testing block 14, 15and 16, the flux density variation was not as large as it was in stator, but was not negligible comparing with the DC component. To able apply the rotational loss equation on rotor, the DC component may need to take out in order to derive the parameter. The implement of rotational loss on these three blocks needed to be investigated further.

For this research, rotational field was assumed not affect the testing block on rotor, as other rotor testing blocks. Examples of flux plot using radial and tangential component were shown in Fig.4.13. There was alternating component flux in testing block 1 in steel and both alternating and rotating component in testing block 3. Hence, the flux analysis for testing blocks with rotating component presented was divided into two parts : radial direction and tangential direction. The flux analysis results of these examples were shown in Fig. 4.14 and Fig. 4.15. Note that the period of flux waveform was 360 degree for testing blocks on stator and halved on rotor. This was because in one electric cycle, the testing block on stator experiencing field provided by two magnets with opposite magnetization direction. On the other hand, the testing blocks on rotor constantly under the field associated between two magnet, i.e. the dominating direction of the field would not change. Hence, the period was halved on rotor.

The situation was similar with open circuit when excitation current was applied. The ratio of long axis and short axis could be derived manually via these plot for later rotational calculation.

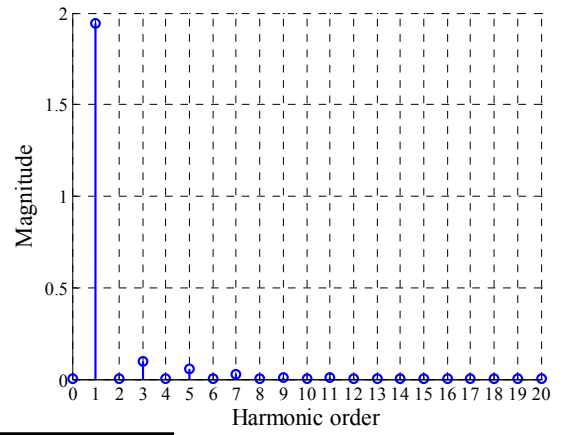
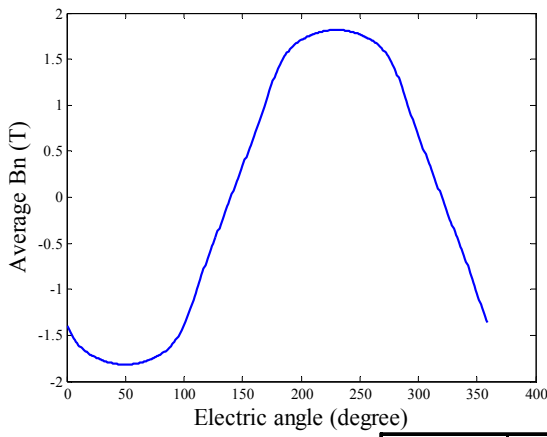


(a) Testing block 1.



(b) Testing block 6.

Fig. 4.13 Examples of flux plot for detecting rotating component.



Harmonic order	Amplitude	Phase
0	0.000	3.1416
1	1.946	2.2698
2	0.000	-2.0712
3	0.098	-2.6149
4	0.000	-0.6592
5	0.058	1.9194
6	0.000	1.6116
7	0.025	-2.9646
8	0.000	2.2615
9	0.008	1.5633
10	0.000	2.3339
11	0.010	2.9640
12	0.000	1.6215
13	0.002	-1.9315
14	0.000	2.8926
15	0.001	2.6528
16	0.000	1.8965
17	0.001	-2.2291
18	0.000	1.7880
19	0.000	2.6274
20	0.000	1.9377

Fig. 4.14 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 1. Phase in radian.

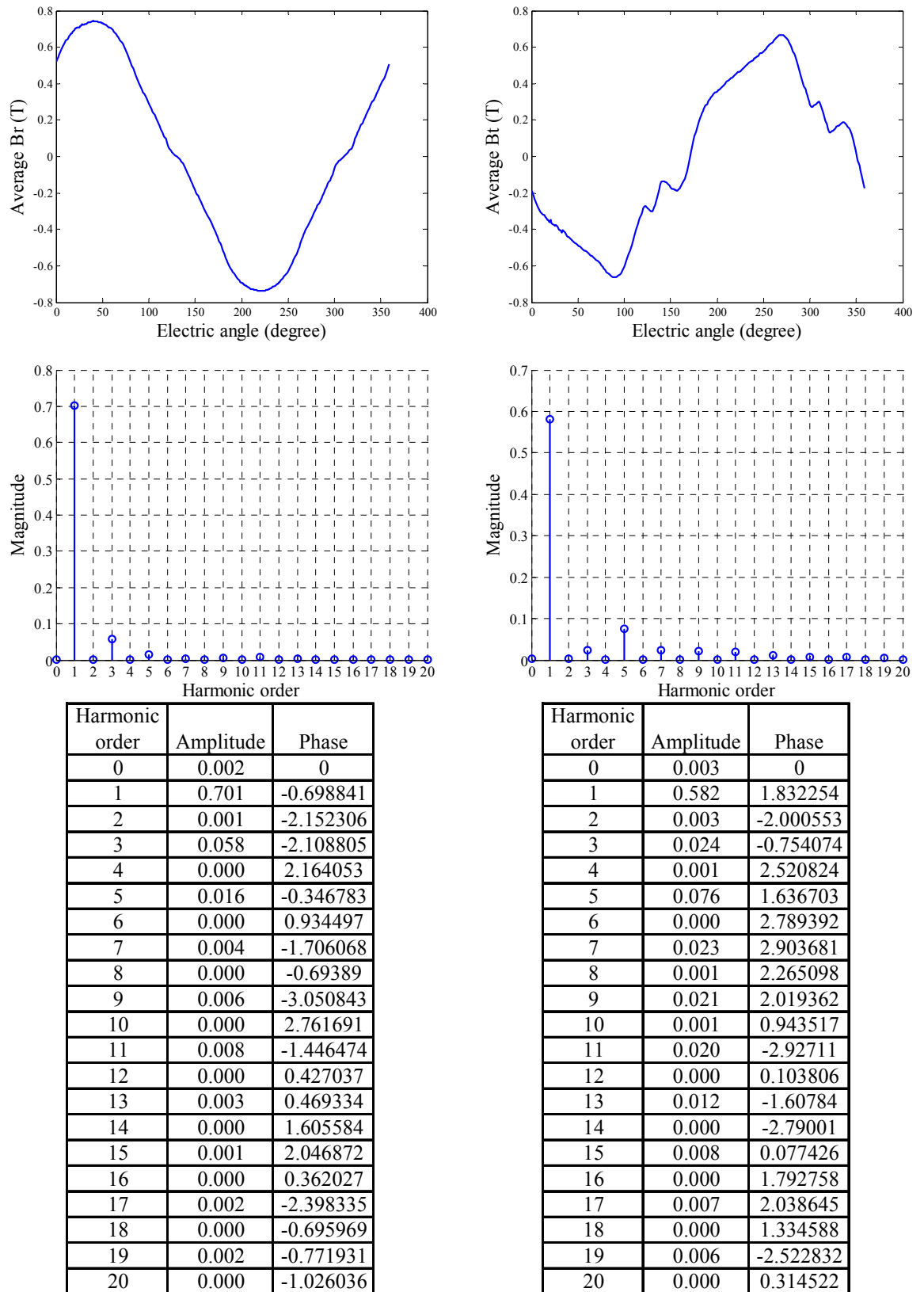


Fig. 4.15 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 3. Phase in radian.



#### 4.4 Iron Loss Calculation

To establish loss calculation, the Steinmetz core loss coefficients [106] was obtained by linear fitting of data points extract from Tansil 335 steel data sheets. By dividing frequency for both side of the iron loss equation (4.7) could be derived into (4.8).

$$W = C_h B_p^{a+bB_p} f + C_e B_p^2 f^2 \quad (4.7)$$

$$\frac{P}{f} = C_h B_p^{a+bB_p} + C_e B_p^2 f \quad (4.8)$$

The linear fitting for P/f data was shown in Fig. 4.16 and the coefficients derived from linear fitting was:  $a = 2.12471$ ,  $b = -0.19674$ ,  $C_h = 0.025199$  and  $C_e = 0.000107$ . These parameter was used with Slemon's and Bertotti's method to estimate the iron loss for comparison, which was shown in Fig. 4.17.

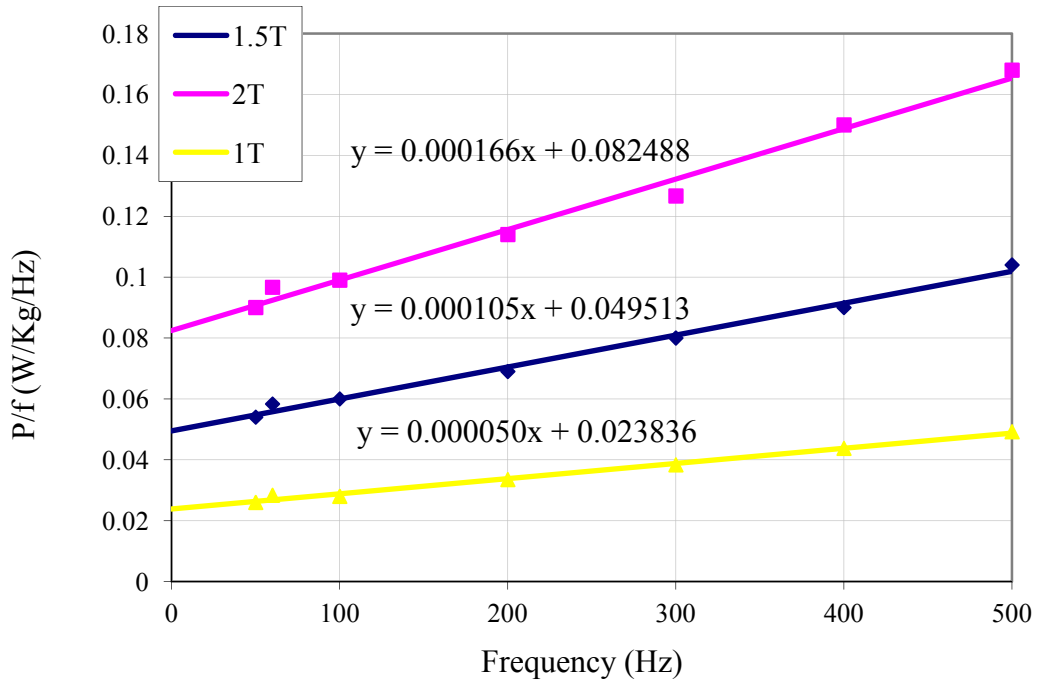


Fig. 4.16 Linear fitting for P/f data for Core loss coefficient obtain.

The alternating iron loss could be derived by using equation 4.1-4.4. Because single valued characteristic was implemented in FEMM, the hysteresis loop and its minor loop effect was neglected. Hence the hysteresis loss equation (4.2) was simplified as

$$P_{hys} = k_h f B_m^\alpha \quad (4.9)$$

where  $B_m$  and  $f$  were the amplitude and frequency of flux density;  $k_h$  and  $\alpha$  were parameters derived from single-sheet test. The eddy current loss was calculated by using equation (4.3) and when

$$B(t) = A \cos(\omega t + \varphi) = A \cos(2\pi f \cdot t + \varphi), \quad (4.10)$$

equation (4.3) was become

$$P_{eddy} = \frac{\sigma}{12} \cdot \frac{d^2}{\delta} \cdot \frac{1}{T} \cdot A^2 \omega^2 \left\{ \left[ \frac{1}{2} t \right]^T - \left[ \frac{1}{4\omega} \sin(2\omega t + 2\varphi) \right]^T \right\} \quad (4.11)$$

and in this case  $T = 2\pi$ , after simplified, it was transformed as

$$P_{eddy} = \frac{\sigma}{12} \cdot \frac{d^2}{\delta} \cdot \frac{1}{T} \cdot A^2 \omega^2 \left\{ \pi - \frac{1}{4\omega} [\sin(4\pi\omega + 2\varphi) - \sin(2\varphi)] \right\} \quad (4.12)$$

where  $A$  is the magnitude peak flux density for  $n$ th order harmonic;  $\omega$  or  $2\pi f$  is the harmonic number;  $\varphi$  is the phase shift;  $k_h = 0.0155$ ;  $\alpha = 2.45$ ; resistivity  $\sigma = 0.5 \times 10^{-6} \Omega^{-1} \cdot m^{-1}$  or conductivity  $\sigma = 2 \times 10^6 S/m$ ;  $d = 3.5 \times 10^{-4} m$ ;  $\delta = 7.65 \times 10^3 kg/m^3$ ;  $T = 2\pi$ .

The excess loss could derived by equation (4.4) and it would become

$$P_{exc} = 8.67 \times k_e f^{1.5} B_m^{1.5} \quad (4.13)$$

for a sinusoidal, i.e. cosine in this case, flux density waveform [14]. The calculated iron loss was shown in Fig. 16 and it was more accurate than Berotti's [106] and Slemom's [108] methods, but the estimated loss was still distance from experimental data [118]. As speed increase, the implemented method became relatively worse than the other two methods, because the lack of correction terms for hysteresis loss. According work published [7], half of the iron loss could be contributed by rotational loss, since the alternating loss was only sensitive to peak and rate of change of flux density. Even accounted in each harmonic of flux density waveform, the error between estimated iron loss and experimental one was still large, i.e. 40%, in open circuit. When rotational loss was counted into estimation, the predication was noticeably improved and leave it as the best method among these three. The rotational part of loss was calculated when assuming constant  $\gamma$  was equal to 1 in equation 4.6 and apply to the area where rotational field was presented, i.e. root of tooth and tooth tip. The ratio of short axis and long axis of flux vector ellipse was derived for each harmonic. Even counted loss for each harmonic, there still significant error with estimated loss. This prove that the iron loss prediction method which heavily relied on peak flux density was not accurate for loss estimation in practice, especially without correction terms.

The iron loss calculation was set up on spread sheet for easy data accessing and better graphic representation. The full spread sheet for iron loss calculation was included as digital media file attach with this thesis, along with other program and data were used in this research.

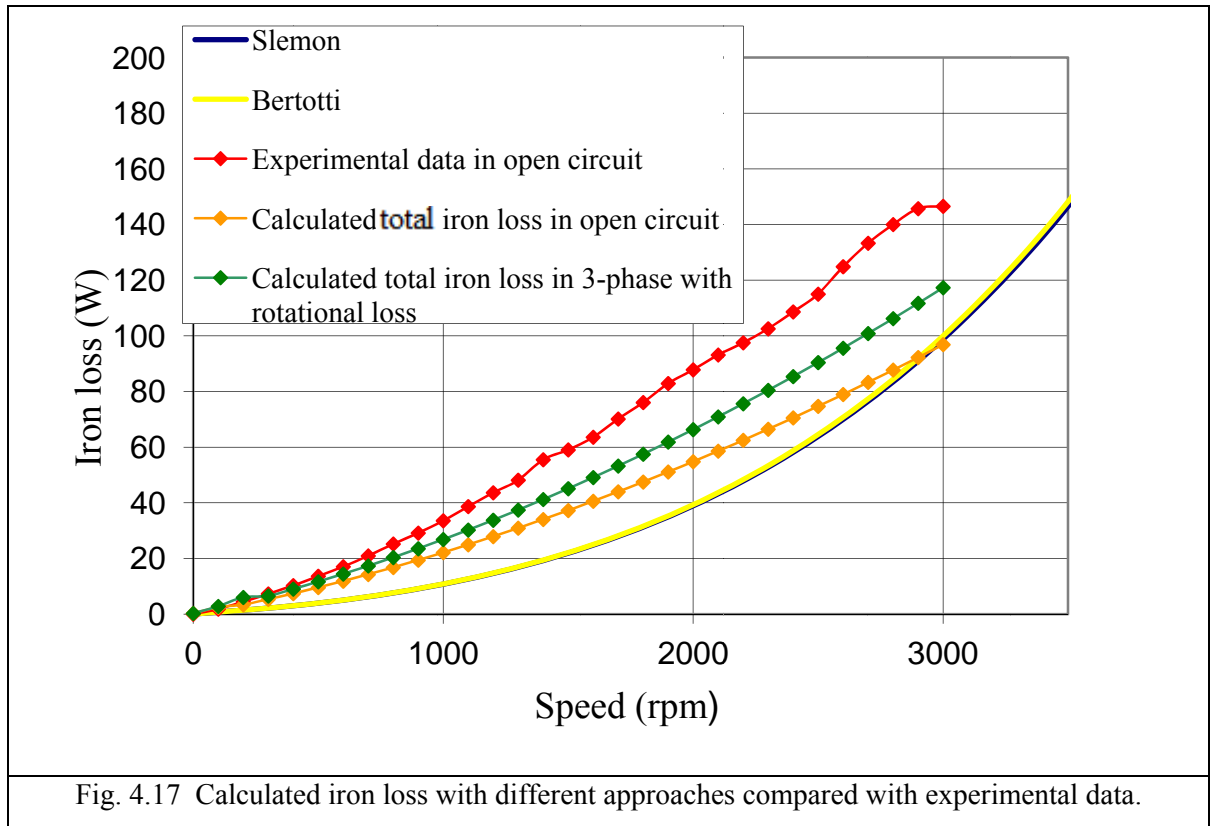


Fig. 4.17 Calculated iron loss with different approaches compared with experimental data.

#### 4.5 Summary

By comparing different ironloss prediction method, Howe's method was selected which was suit the situation of this research. The area dividing method was improved based on whether rotating flux was presented. The ironloss calculation then established based on flux spectrum of different area and rotating component correction. The predicted value reached 75% of experimental data in the worst case and even only with alternating flux, there was still significant improvement over Berotti's [106] and Slemom's [108] methods. The analyzing software developed during the work related to this Chapter was also packaged and could be used in any other machine model simulation, which was based on flux density. In next Chapter, this analyzing software was employed for the tasks with different phase winding options were considered.

## **CHAPTER5**

# **STUDY INTO THE CHOICE OF PHASE NUMBER FOR A MULTIPHASE, CONCENTRATED WINDING MACHINE**

### **5.1 Introduction**

To have greater impact on application it was desirable to integrate machines and their associated power electronic converters into complicated systems that share component count to minimise cost. for example integrated starter/alternator systems that were now becoming common for small urban vehicles [28]. However, while compactness of systems was achieved, reliability of the power conversion components could be compromised by the arduous nature of system operating environments. This Chapter investigated the impact of using higher phase numbers for electromagnetic machines and their associated power electronic converters targeting improved specific and volumetric power density and increased lifetime.

A key aspect of electric vehicle design was the traction system comprising of a traction machine, power electronic converter and energy source (or sources).For EV applications, traction machine requirements include high torque and/or power density, high efficiency, low mass, wide operating speed range and, importantly, reliability [116, 117]. The brushless permanent magnet (BLPM) machine had gained popularity because of these attributes. Additionally, the emphasis for future designs would be continuing improvements in component integration and lifetime, factors that ultimately impact on system cost and application uptake.

In a wider applications context, the next five years, a substantial increase in the number of all electric vehicles (EVs), hybrid-electric vehicles (HEVs), more electric aircraft, electric ship propulsion, renewable (wind) and other industrial applications would be seen of which

were using energy/power conversion via electromagnetic machines and power electronic converters operating as variable speed drive systems.

Power electronic converters were used to convert DC to AC or vice versa, and were, from an industrial perspective, a relatively new technology, i.e. with an established track record of only 20 or so years. New system concepts have flourished during the past 20 years, for example in the automotive sector, where HEVs have typical installed power electronic converters of the order of 50 kW in high volumes, reference the much publicised Toyota Prius.

This Chapter would discuss the use of electric machines with phase numbers higher than three and then compare the performance of a test BLPM machine design that may be connected in either a 3-phase or 9-phase configuration. Thus, the test machine facilitates a comparison to be made between the more conventional 3-phase topology referenced as a benchmark traction machine, with that of a machine connected and controlled in a 9-phase configuration and intermediate (5 and 7) and higher phase numbers (greater than 9). The study would discuss the improvement in torque capability and present the comparison in iron loss for comparable load and copper loss conditions.

## 5.2 The Influence of Phase Number for Sinusoidal Drive Systems

Considering a 3-phase BLPM drive system in which the power electronic converter control scheme maintained sinusoidal currents in the machine windings that are in-phase with the respective machine back EMF and, neglecting windage and frictional losses, the machine peak mechanical power output was:

$$\hat{P}_3 = 3E_{RMS}I_{RMS} = \frac{3}{2}\hat{E}_3\hat{I}_3 \quad (5.1)$$

where  $\hat{E}$ ,  $\hat{I}$  were the peak of the 3-phase, phase back-EMF and phase current respectively. Generalizing (1) for poly-phase machines yielded:

$$\hat{P}_n = \frac{n}{2}\hat{E}_n\hat{I}_n \quad (5.2)$$

where n is the number of phases, and was greater than 3.

The following discussion is for a 2-level, half bridge converter and star-connected machine. The three-phase power electronic converter switching devices must be suitably rated so as

to be capable of blocking twice the machine peak phase back-EMF and supply the machine peak phase current. Hence, the converter volt-ampere (VA) rating for a three-phase drive system was:

$$VA_{RATING} = 6_{DEVICES} (2\hat{E}) \hat{I}_3 \quad (5.3)$$

Again, generalizing for a poly-phase converter power electronic converter yielded:

$$VA_{RATING} = 2n (2\hat{E}) \hat{I}_n \quad (5.4)$$

Comparing the BLPM machine and power electronic converter peak power capabilities yields a figure indicative of the system utilization:

$$\frac{\hat{P}}{VA_{RATING}} = \frac{1}{8} \quad (5.5)$$

which was independent of the number of phases.

Thus, the utilization of a sinusoidal poly-phase brushless drive system, normalized to a reference 3-phase system, is unity. This finding assumes that within a given stator volume, a sinusoidal winding distribution may be realised as the number of phases increase. This may be an acceptable assumption for larger machines where it was clearly beneficial to exploit high phase number. However, was not as valid for smaller machines (of the 10-50kW range) due to the physical realisation of practical coils.

Considering the thermal aspects of the machine design with phase number. This was more easily illustrated by way of an example, thus a 3- and 9-phase design solution would be considered. In chapter 4, and the femm modelling in this chapter, the 3-phase machine has series-connected coils, to give the same phase current as the 9-phase machine, but a different supply voltage. In this section only, parallel-connection of the 3-phase coils is considered to give the same phase voltage as the 9-phase machine. Here, for the same converter supply voltage and machine volts/rpm specification, the number of turns for both 3- and 9-phase windings are equal. Hence, the phase resistance of the 3-phase winding is 3-times that of the 9-phase winding:

$$R_3 = 3 R_9 \quad (5.6)$$

The continuous thermal rating of both the 3- and 9-phase machines would be determined by the machine internal loss density, since the flux levels in both stators were similar then the iron loss for both machines could be assumed to be equal. Hence, the winding copper

loss will dictate the relative continuous rating of the two machines. For sinusoidal excitation, the winding copper loss was determined from:

$$\hat{P}_{CU-n} = n \left( \frac{\hat{I}_n}{\sqrt{2}} \right)^2 R_n = \frac{n}{2} \hat{I}_n^2 R_n \quad (5.7)$$

Therefore,

$$\hat{P}_{CU-3} = \frac{3}{2} \hat{I}_3^2 R_3 \quad (5.8)$$

$$\hat{P}_{CU-9} = \frac{9}{2} \hat{I}_9^2 R_9 \quad (5.9)$$

for the 3- and 9-phase machines respectively.

Substituting (5.6) into (5.9) and equating the copper loss for the two topologies yielded the relative phase current magnitudes for equal copper and hence machine loss dissipation:

$$\frac{3}{2} \hat{I}_3^2 R_3 = \frac{9}{2} \hat{I}_9^2 R_9 \quad (5.10)$$

$$\hat{I}_9 = \frac{\hat{I}_3}{3} \quad (5.11)$$

Therefore, substitution of (5.11) into (5.1) and (5.3) yields the same machine and power electronic converter peak power capability. Eqn.(11) is also the condition for equal winding mmfs in the stator core. Generalizing the above argument for poly-phase machines, it can be shown that :

$$R_n = \frac{n}{3} R_3 \quad (5.12)$$

$$\hat{I}_n = \frac{3}{n} \hat{I}_3 \quad (5.13)$$

Consequently, for sinusoidal brushless drive systems, there appeared to be no benefit in power output to be gained by increasing the system phase number above 3. There were hardware disadvantages arising from the increased number of cable connections and current measurement requirements, plus increased software control complexity with increasing phase number, number of devices, gate drivers etc.

Apart from these points, an alternative argument could be considered. Firstly, there appears to be no detriment in power output by considering higher phase numbers. Secondly, with power electronic converter controlled machines there no longer need to strive to achieve a sinusoidal winding/current MMF. By appropriate PM design, a simple concentrated coil may develop a sinusoidal back-EMF. Thirdly, the power electronic converter could be improved by using more, lower current silicon devices which may be better distributed

around the module heat sink to improve thermal performance and hence reduce heatsink requirements.

These potential benefits will be explored by virtue of a comparison between the performance capability of a BLPM machine that has a particular magnetic implementation that lends itself to be configured for as either a 3- or 9-phase machine.

## 5.3 Analysis of Test Machine for 3 and 9-Phase Connection

### 5.3.1 Back-EMF

The test machine was consisted of 9 stator teeth and 8 rotor poles implemented via bread-loaf shaped sintered NdFeB permanent magnets. The magnetic design minimises cogging torque while yielding a near sinusoidal back-EMF waveform for the 3-phase winding connection. Fig. 5.1 shows an example machine field distribution and coil connection scheme for one phase of the 3-phase configuration. Note that each phase consisted of 3-series connected coils, where the centre coil was reverse wound, as shown in Fig. 5.1. The other 2 phases follow on consecutive teeth. Fig. 5.2 shows an example machine field distribution and coil connection scheme for one phase of the 9-phase machine. Fig. 5.3 shows the normalised induced back-EMF per phase for the 3- and 9-phase configurations where it could be seen that the series combination of 3 coils for the 3-phase winding reduces a number of EMF harmonics (namely 5th and 7th) that are present in the individual coils.

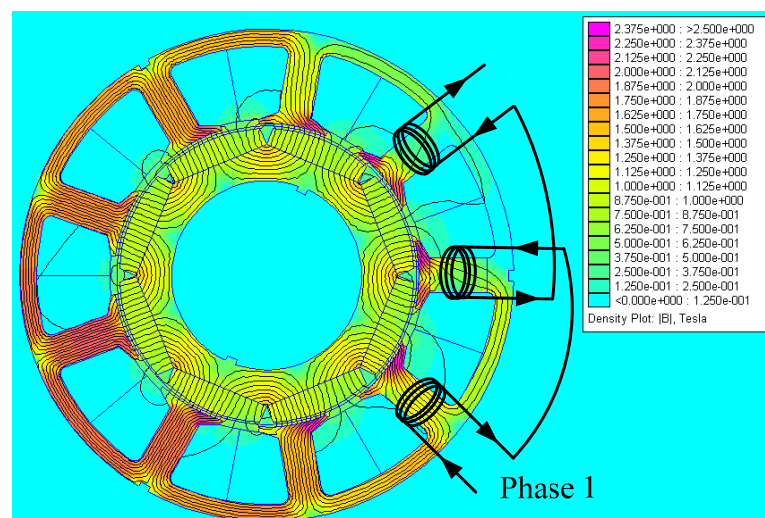




Fig. 5.1. Electro-magnetic field distribution of test machine - 3-phase configuration.

### 5.3.2 Electromagnetic torque.

Excitation of the machine windings with sinusoidal currents controlled would be in-phase with the respective machine back EMFs (i.e.  $I_d = 0$ , torque proportional to  $I_q$ ) yielded the normalised torque versus rotor angle for the test machine, as shown in Fig. 5.4. Here there was an increase of 4.5% in the mean torque per Amp for the 9-phase configuration arising from the winding distribution factor together with a reduction in excitation torque ripple from 3.31% for the 3-phase to 0.85% for the 9-phase case, as discussed in the previous Chapters.

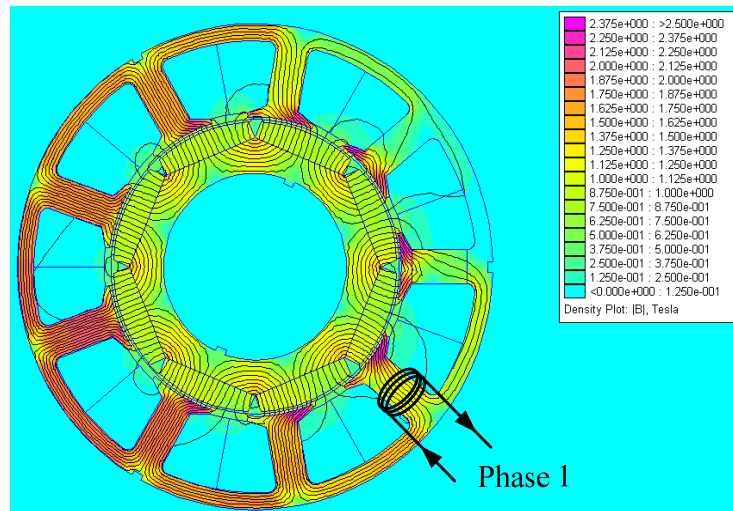


Fig. 5.2. Electro-magnetic field distribution of test machine - 9-phase configuration.

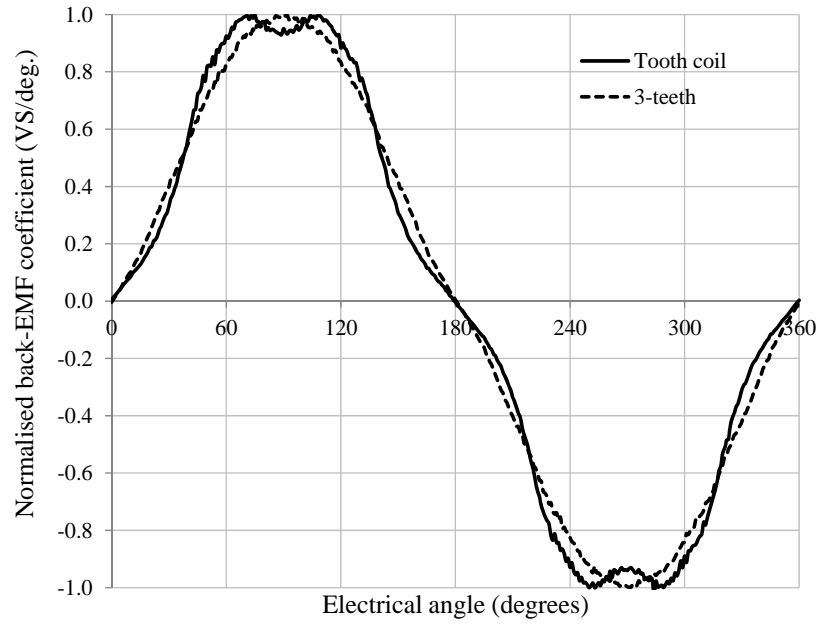


Fig. 5.3. Normalised induced back-EMF per phase for the 3-phase (3-teeth) and 9-phase (1 tooth coil) test machine.

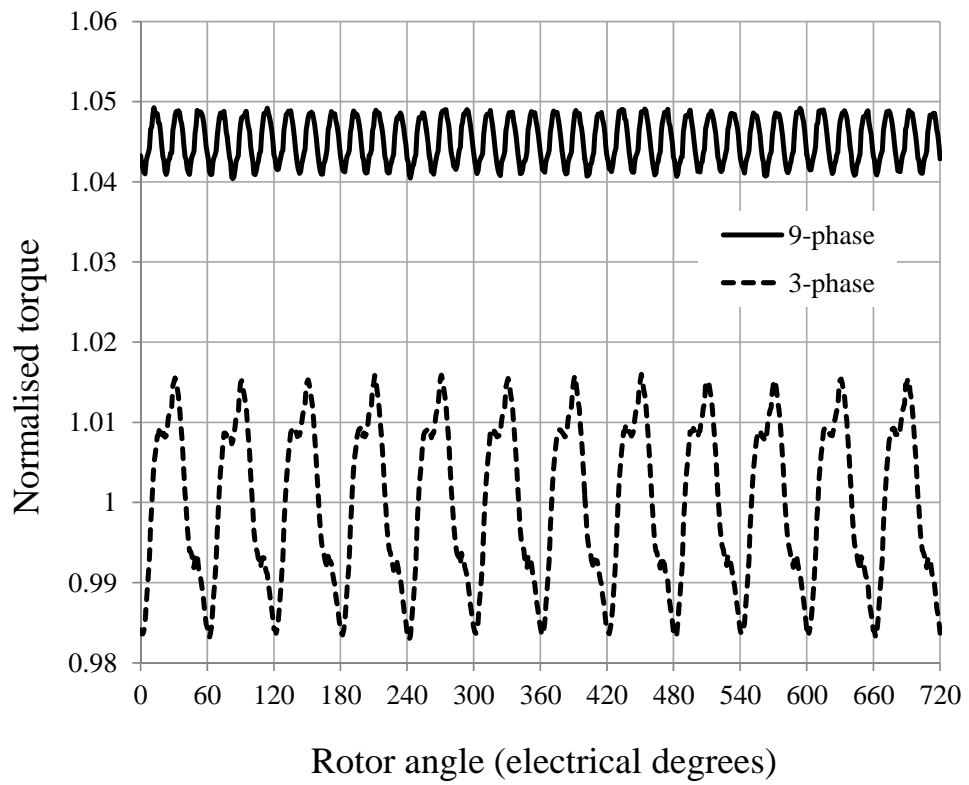


Fig. 5.4. Normalised torque versus electric angle for the 3-phase and 9-phase test machine with sinusoidal phase current.

## 5.4 Other Phase Connections

The benchmark machine used for the 3- and 9-phase study was modified to investigate the impact of higher phase number. To maintain a consistent comparison, i.e. similar copper loss or slot area, the rotor design and airgap dimensions were changed. Further, the machine copper loss was maintained constant and equal to that of the 3- and 9-phase machine for each phase design. Finally, the stator soft magnetic area (and hence volume) was maintained such that for each machine design considered the total available slot area for copper remained constant. Within these considerations, Figs. 5.5 and 5.6 show the main designs considered. Tooth/slot ratio and tooth tip shaping was not optimised for each design.

#### **5.4.1 Back-EMF and Torque Performance**

Figs. 5.7 and 5.8 shown the phase back-EMF and torque versus angle data for each phase design considered. For the concentrated winding topologies there were benefits in higher phase number illustrated by the 5-, 7- and 9-phase design, due to the winding connection for the 9-phase design, as discussed previously, and contribution to torque from the back EMF harmonics for the 5- and 7-phase designs. One benefit is an increase in average torque per ampere of 11%, 7% and 4% for the 5-, 7- and 9-phase designs when compared to the base 3-phase design. The second benefit was an improved excitation torque ripple for the 9-phase and higher. Indeed, for traction machine drives where the electric machine drives through a gear-stage, the designer may consider a lower average torque per ampere if the peak-to peak torque ripple is reduced since this ultimately leads to gear fatigue and gear-stage failure.

For phase numbers above 9, there was no benefit in torque capability due primarily from the continual depreciation of the fundamental component of the phase back EMF, as illustrated in Fig. 5.8.

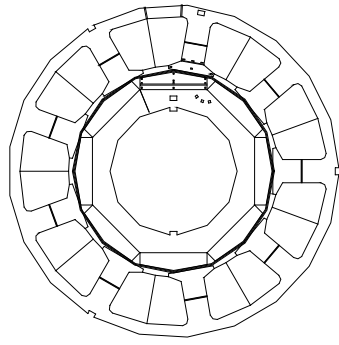
#### **5.4.2 Iron-Loss Considerations**

Figs. 5.10 shown the iron-loss profile for each machine design versus rotor speed. The results of estimated iron loss were all less than the open circuit loss measure in experiment

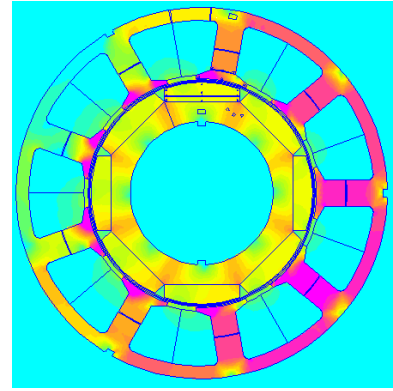
[118], which data was obtained from published work. There were large error even after the rotational loss was counted for each harmonic and this may because the estimation was heavily depend on the peak flux density and lack of hysteresis correction. Despite the difference with experimental data, the iron loss generated by 3-phase, 9-phase and 11-phase model with 1000 A-t excitation current were similar with loss in open circuit. Rest of the model tested, i.e. 5-phase, 7-phase and 13-phase, produced noticeable less loss, especially 5-phase model. At same excitation current, 3-phase model produced more loss than 9-phase model by a small margin. 11-phase model had the highest iron loss among all models.

According to the peak value flux density in field solution of each model, the model produce similar iron loss with open circuit, should have similar flux density distribution, hence the implemented estimation method yielded not much difference in iron loss for 3-phase, 9-phase and 11-phase model. On the other hand, 5-phase, 7-phase and 13-phase may have more evenly distributed flux in stator, possibly led to less flux variation.

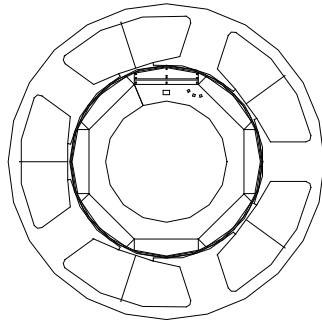
Also there was little increase in iron loss when the phase current was approximately doubled for 7-phase model, so it appeared that there was little correlation between iron loss and excitation current. More study was required and saturation check was considered in future study. In terms of iron loss, 5-phase model was the best design. It produced significant less iron loss than 3-phase and 9-phase model. The 7-phase design may be replacement of 5-phase after validated in saturation test.



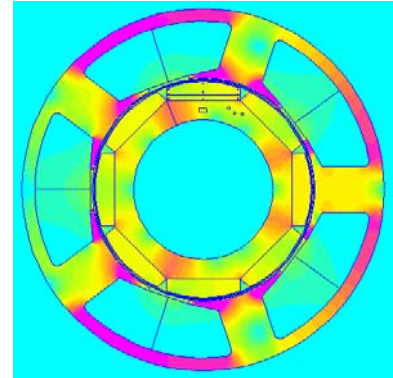
(a) 3-phase model



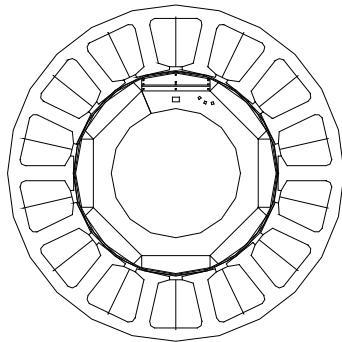
(b) 3-phase field solution



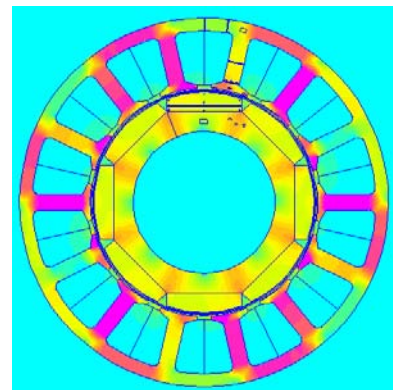
(c) 5-phase model



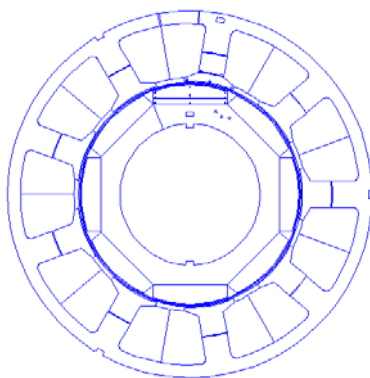
(d) 5-phase field solution



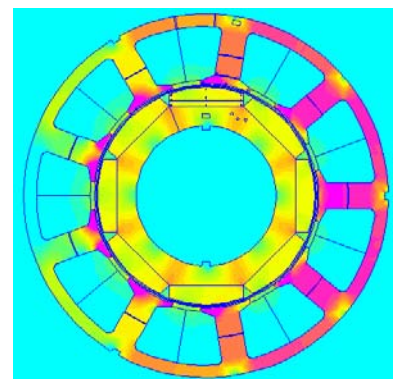
(e) 7-phase model



(f) 7-phase field solution

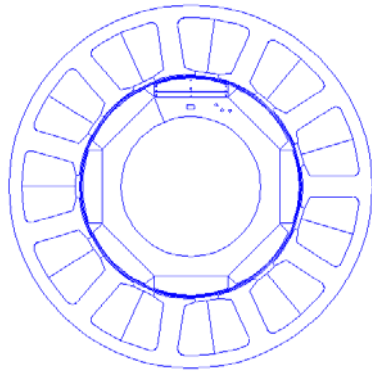


(g) 9-phase model

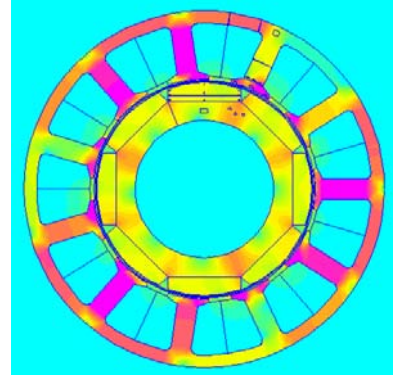


(h) 9-phase field solution

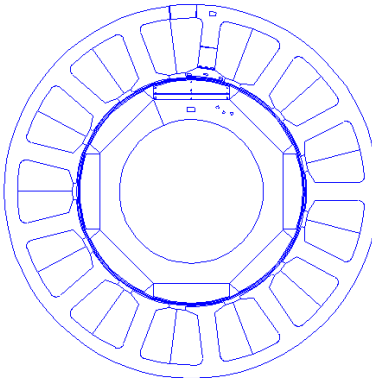
Fig. 5.5 Models and full-load field solutions.



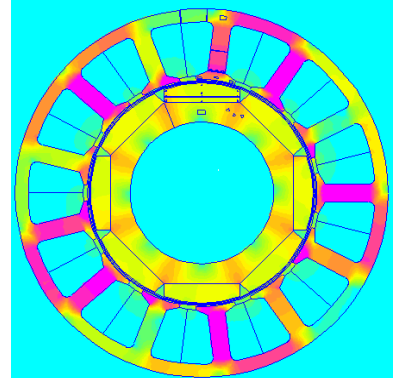
(a) 11-phase model



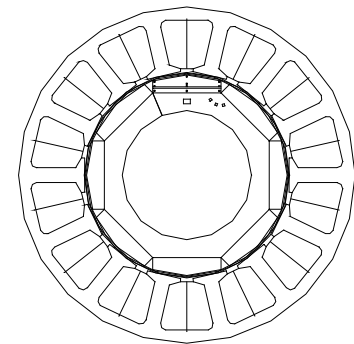
(b) 11-phase field solution



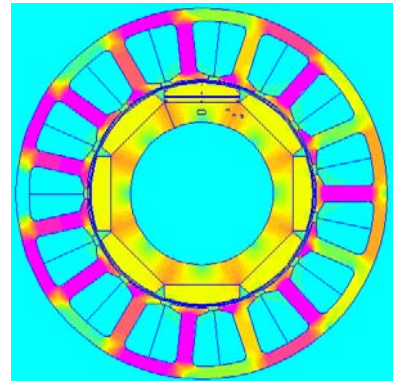
(c) 13-phase model



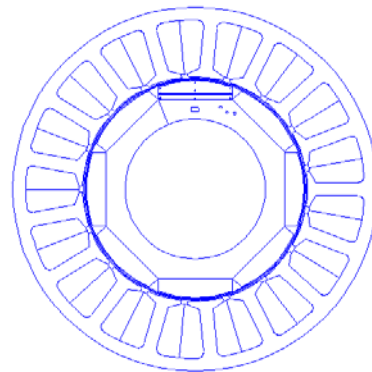
(d) 13-phase field solution



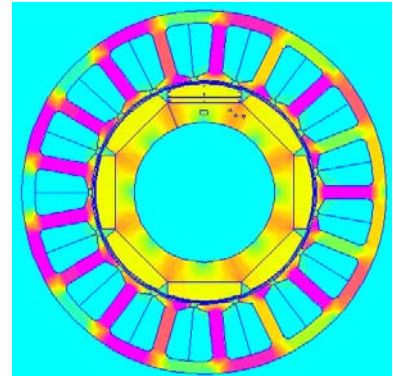
(e) 15-phase model



(f) 15-phase field solution

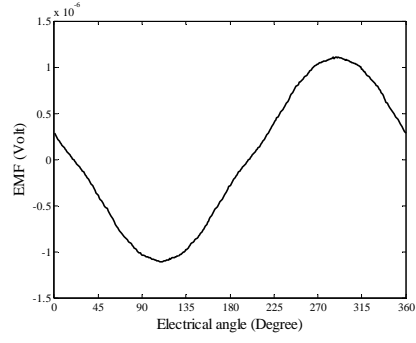


(g) 17-phase model

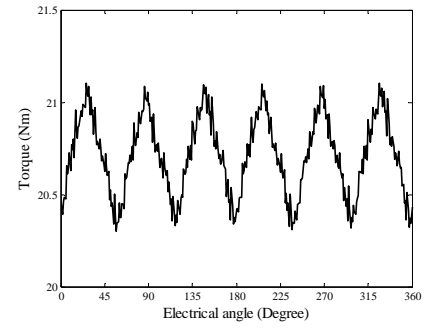


(h) 17-phase field solution

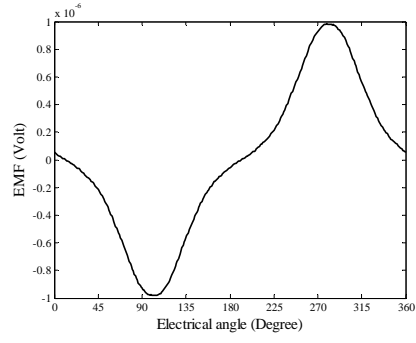
Fig. 5.6 Models and full-load field solutions.



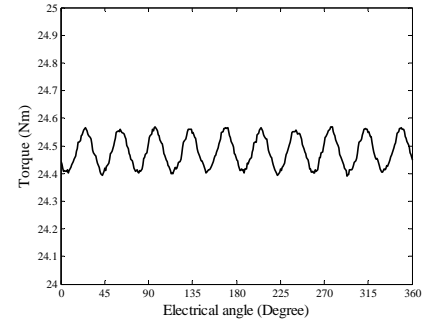
(a) 3-phase open-circuit back-EMF



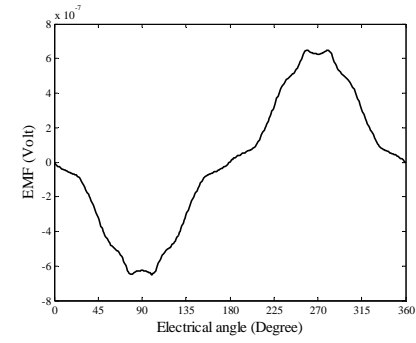
(b) 3-phase torque versus angle



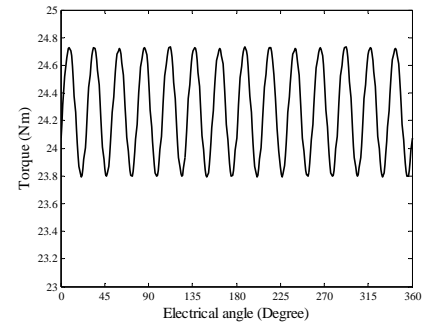
(c) 5-phase open-circuit back-EMF



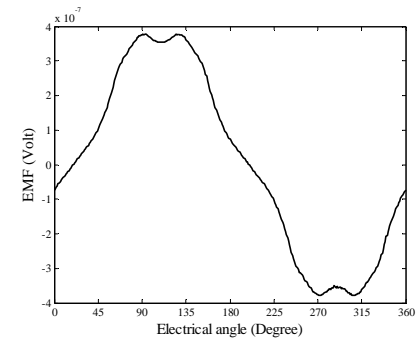
(d) 5-phase torque versus angle



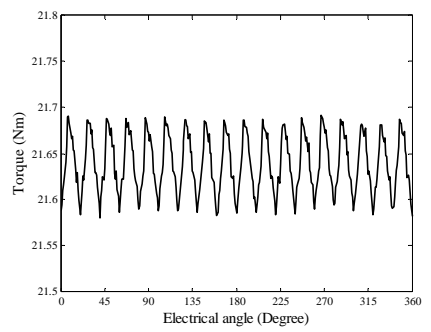
(e) 7-phase open-circuit back-EMF



(f) 7-phase torque versus angle

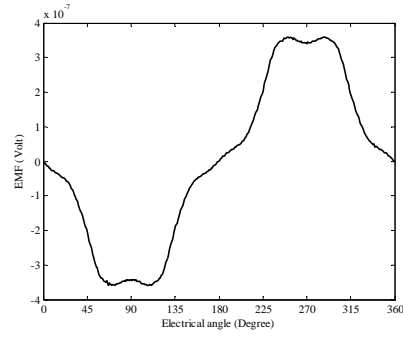


(g) 9-phase open-circuit back-EMF

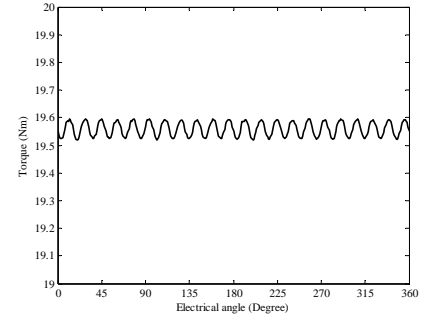


(h) 9-phase torque versus angle

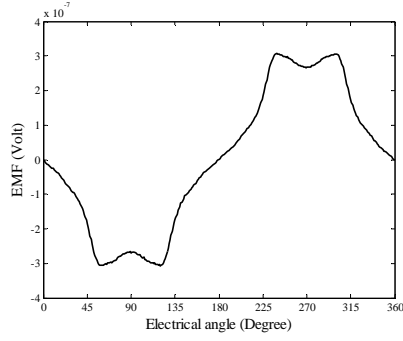
Fig. 5.7 Back-EMF and torque results.



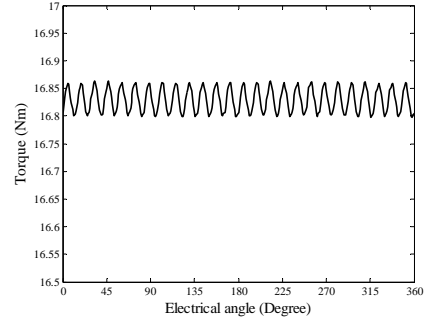
(a) 11-phase open-circuit back-EMF



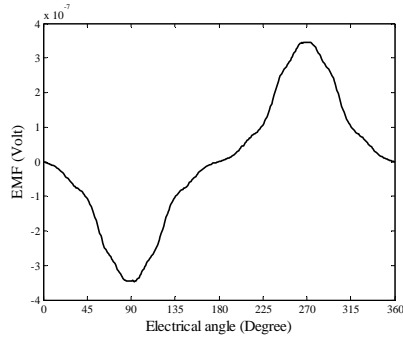
(b) 11-phase torque versus angle



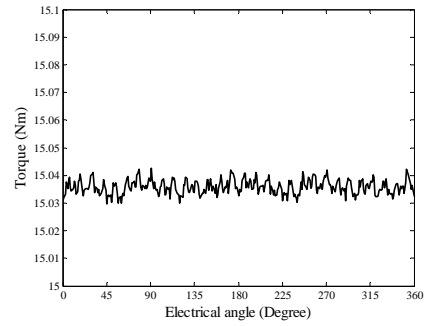
(c) 13-phase open-circuit back-EMF



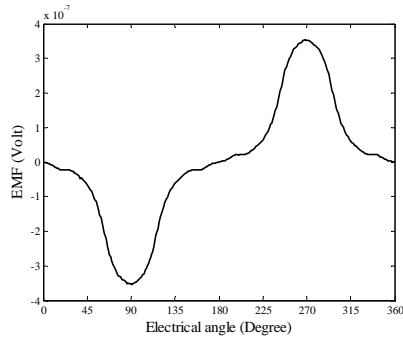
(d) 13-phase torque versus angle



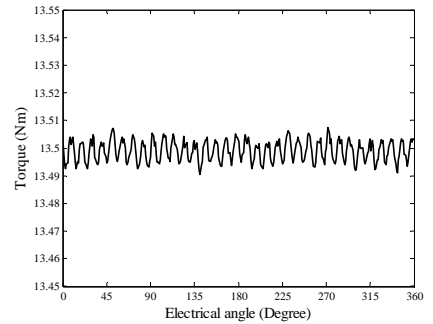
(e) 15-phase open-circuit back-EMF



(f) 15-phase torque versus angle



(g) 17-phase open-circuit back-EMF



(h) 17-phase torque versus angle

Fig. 5.8 Back-EMF and torque results.



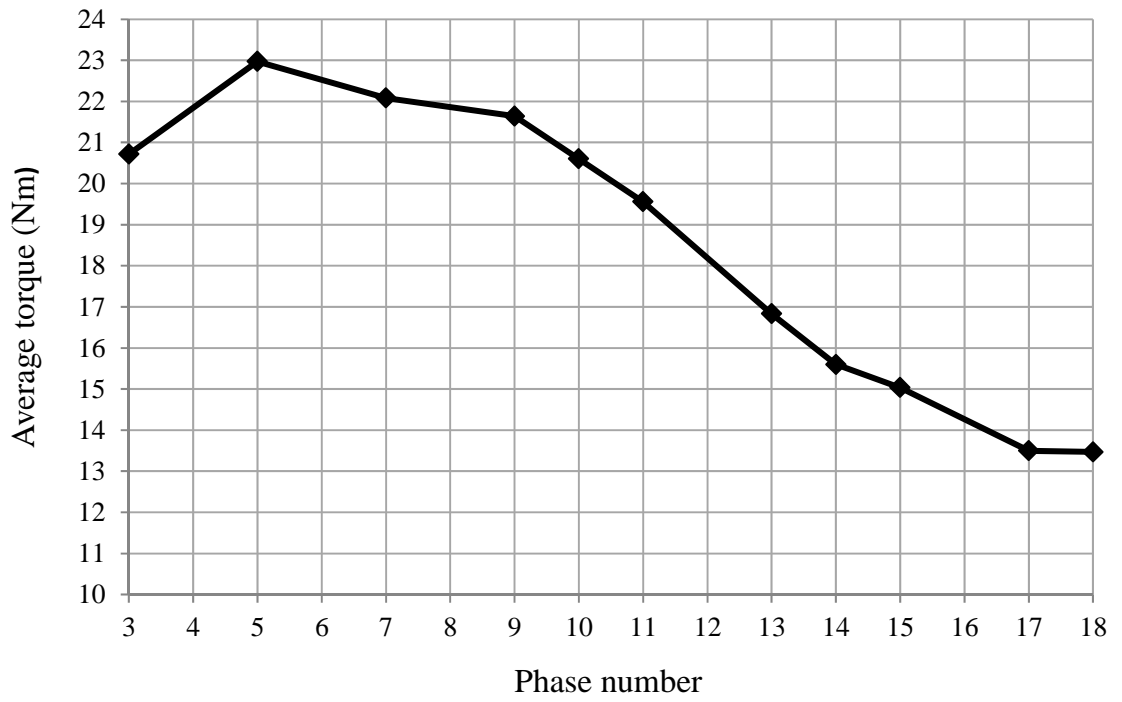


Fig. 5.9 Average torque versus phase number.

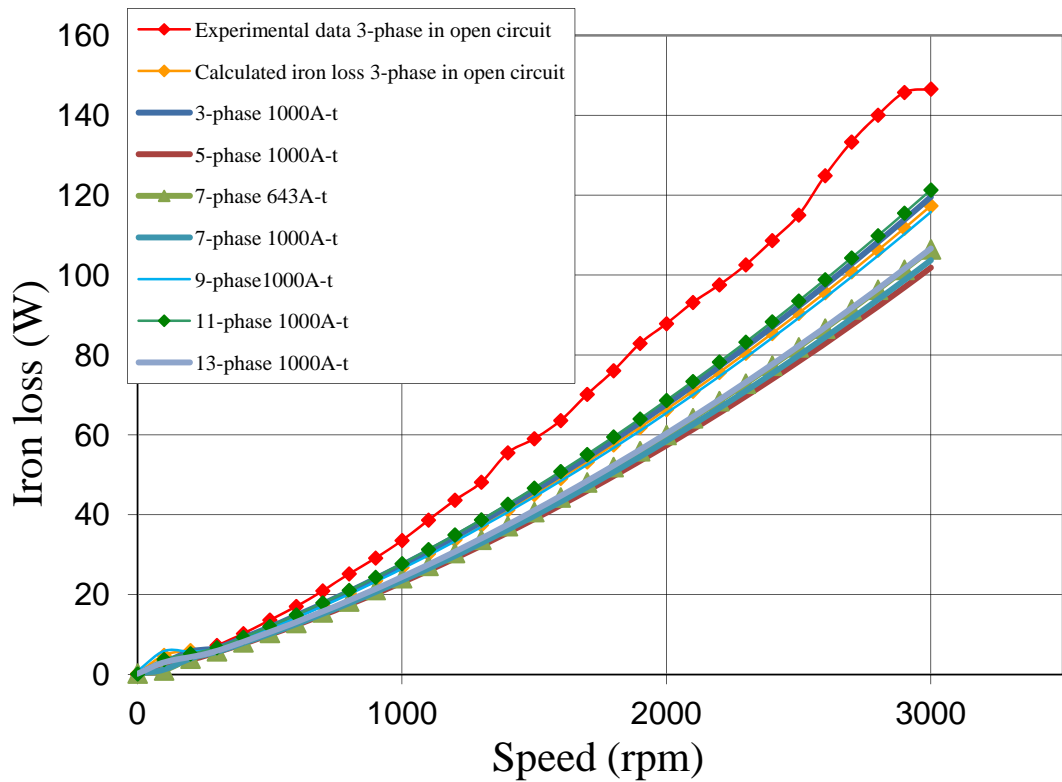


Fig. 5.10 Iron loss versus speed results.

## 5.5 Summary

As the power ratings of traction drive systems continue to increase, the benefits of using phase numbers higher than three start to outweigh perceived disadvantages. With automated, high volume manufacture the ultimate route for production of these systems, the increase in system component count and complexity become less of an issue. This Chapter provides a comparison between a standard 3-phase machine design rewound as a 9-phase design, utilising the same magnet and copper volume. The study then expands the study by considering 5, 7 phases and 11, 13 and 15 phases.

The results highlight the improvement in torque capability per unit current and torque quality showing that an optimum lies somewhere between 3 and 9 phase, but that capability reduces above 9 phases due to the degradation in machine back-EMF shape from an ideal sinusoid. There is no advantage for 9 phase design in iron loss, comparing with 5 phase. With slight less peak torque production, the 5 phase model was the better design in terms of performance, at this stage of research. The 9 phase design may be used situational where fault tolerate and power rating was crucial.

These observations have to be taken with some caution and not to be expressed as generalised conclusions. However, the improvement in performance for a concentrated wound machine by the appropriate choice of phase number is an important conclusion and one that should be considered by future designers of variable speed drive systems.

## **CHAPTER6**

# **CONCLUSIONS AND RECOMENDATIONS FOR FURTHER WORK**

### **6.1Conclusions**

Work had been established for traction machine design aspects in this research. The effect of multiphase design for Permanent Magnet (PM) machine was investigated. The electromagnetic characteristics of both 3-phase and 9-phase machine, along with different magnet designs, were simulated and analyzed by using the program developed during the process. The software used were FEMM and MATLAB. The iron loss for different designs was established, based on the analytical flux density obtained by 2-D stepping FEA method. The harmonic of flux waveform and rotating field were also considered for difference areas in the machine models. The prediction was compared with experimental data collected [118] in open circuit. The simulation result shown that there was a minimum 4% torque gain and noticeable less torque ripples for 9-phase machine, comparing with 3-phase one, with the same excitation phase current. The embedded magnet rotor design was suggested to monitor the demagnetization of each magnet closely, since some area of the magnet could be demagnetized even when the working point of magnet was well distance away from the nonlinear region of its characteristic. There were about 6% less iron loss was produced in 9-phase model than 3-phase model. The implemented method for calculating iron loss was more accurate within 3500 rpm rotor speed comparing with other approaches.

As the power ratings of traction drive systems continue to increase, the benefits of using phase numbers higher than 3 start to outweigh perceived disadvantages. Indeed, with automated, high volume manufacture the ultimate route for production of these systems, the increase in system component count and complexity become less of an issue. This

thesis provides a comparison between a standard 3-phase machine design rewound as a 9-phase design, utilising the same magnets and copper volume. The study is then expanded considering 5-phases and 7- phases, and a number of options about 9-phase.

The results highlight the improvement in torque capability per unit current and torque quality showing that an optimum lies somewhere between 3 and 9 phase, but that capability reduces above 9 phases due to the degradation in machine back-EMF shape from an ideal sinusoid. Although, the study at this stage shows that there was no improvement in iron loss for the phase designs move to 9, the result would be justified while future study established.

These observations have to be taken with some caution and not to be expressed as generalised conclusions. However, the improvement in performance for a concentrated wound machine by the appropriate choice of phase number is an important conclusion and one that should be considered by future designers of variable speed drive systems.

## **6.2 Recommendations for further work**

Due to the lack of correction in iron loss calculation, the non-linear steel characteristic would be better to implement to improve the accuracy of iron loss estimation. Using testing blocks to represent the area with difference flux density waveform was rough and large error would introduce. Hence, more testing blocks in different area will improve accuracy as well or detail into each elements. But the FEMM program would regenerate the mesh for each static solution and it would be difficult to fix a mesh. Creating layers in model, i.e. separating whole model into large number of small blocks, by user would be a possible solution. Although 9-phase design was slight better design in terms of torque and loss, more examples were required to be tested before generalising the conclusions.

## REFERENCE

- [1] Junior high school text book, "World History", 1999-2000.
- [2] Putrus, G.A.; Suwanapingkarl, P.; Johnston, D.; Bentley, E.C.; Narayana, M.; ,  
"Impact of electric vehicles on power distribution networks," *Vehicle Power and Propulsion Conference, 2009. VPPC '09. IEEE* , vol., no., pp.827-831, 7-10 Sept. 2009  
doi: 10.1109/VPPC.2009.5289760
- [3] Shams-Zahraei, M.; Kouzani, A.Z.; , "A study on plug-in hybrid electric vehicles,"  
*TENCON 2009 - 2009 IEEE Region 10 Conference* , vol., no., pp.1-5, 23-26 Jan. 2009  
doi: 10.1109/TENCON.2009.5395851
- [4] International Energy Agency, "Hybrid and electric vehicles the electric drive establishes a market foothold", February 2009.
- [5] Smith Electric vehicles, "Product full specification", September 2010,  
<http://www.smithelectricvehicles.com/>
- [6] Prof A C Smith and Dr. N. Schofield, Lecture notes for module EEEN60195-Design of Electrical Machines, February 2010.
- [7] Mi, C.C.; , "Analytical design of permanent-magnet traction-drive motors,"  
*Magnetics, IEEE Transactions on* , vol.42, no.7, pp. 1861- 1866, July 2006  
doi: 10.1109/TMAG.2006.874511
- [8] Patterson, D.J.; , "High efficiency permanent magnet drive systems for electric vehicles," *Industrial Electronics, Control and Instrumentation, 1997. IECON 97. 23rd International Conference on* , vol.2, no., pp.391-396 vol.2, 9-14 Nov 1997  
doi: 10.1109/IECON.1997.671766
- [9] Mellor, P.H.; Wrobel, R.; McNeill, N.; Drury, D.; , "Impact of winding and rotor design on efficiency and torque ripple in brushless AC permanent magnet traction motors," *Power Electronics, Machines and Drives, 2008. PEMD 2008. 4th IET Conference on* , vol., no., pp.240-244, 2-4 April 2008
- [10] Wang, J.; Atallah, K.; Zhu, Z.Q.; Howe, D.; , "Modular 3-Phase Permanent Magnet Brushless Machines for In-Wheel Applications," *Vehicle Power and Propulsion Conference, 2006. VPPC '06. IEEE* , vol., no., pp.1-6, 6-8 Sept. 2006  
doi: 10.1109/VPPC.2006.364384
- [11] Levi, E.; , "Multiphase Electric Machines for Variable-Speed Applications,"  
*Industrial Electronics, IEEE Transactions on* , vol.55, no.5, pp.1893-1909, May

2008

doi: 10.1109/TIE.2008.918488

- [12] Rahman, M.A.; Masrur, M.A.; , "Advances on IPM technology for hybrid electric vehicles," *Vehicle Power and Propulsion Conference, 2009. VPPC '09. IEEE* , vol., no., pp.92-97, 7-10 Sept. 2009  
doi: 10.1109/VPPC.2009.5289864
- [13] Prof A C Smith and Dr. N. Schofield, Lecture notes for module EEEN60193-Modelling of machines, October 2009.
- [14] Simoes, M.G.; Vieira, P., Jr.; , "A high-torque low-speed multiphase brushless machine-a perspective application for electric vehicles," *Industrial Electronics, IEEE Transactions on* , vol.49, no.5, pp. 1154- 1164, Oct 2002  
doi: 10.1109/TIE.2002.803241
- [15] Parsa, L.; Taehyung Kim; , "Reducing Torque Pulsation of Multi-Phase Interior Permanent Magnet Machines," *Industry Applications Conference, 2006. 41st IAS Annual Meeting. Conference Record of the 2006 IEEE* , vol.4, no., pp.1978-1983, 8-12 Oct. 2006  
doi: 10.1109/IAS.2006.256806
- [16] Parsa, L.; Toliyat, H. A.; Goodarzi, A.; , "Five-Phase Interior Permanent-Magnet Motors With Low Torque Pulsation," *Industry Applications, IEEE Transactions on* , vol.43, no.1, pp.40-46, Jan.-feb. 2007  
doi: 10.1109/TIA.2006.887235
- [17] Gordon R.Slemon and A.Straughen, "Electric machines", Published Reading, Mass. ; London : Addison-Wesley 1980
- [18] Gordon R.Slemon, "Electric machines and drives", Published Reading, Mass.; Wokingham : Addison-Wesley 1992
- [19] M.G. Say, "Alternating current machines fifth edition", Published London: Pitman 1983.
- [20] B. J. Chalmers, A. C. Williamson, "A.C. Machines Electromagnetics and Design", Published Taunton: Research Studies 1991.
- [21] H. C. J. de Jong, "AC motor design with conventional and converter supplies", Published Oxford: Clarendon Press 1976.
- [22] Dr. N. Schofield, "Electric Vehicle Traction Machine and Generator Design Considerations", presentation in seminar: Developments in Electric Propulsion Technology, 20/21 September 2010 ▪ Helsingborg, Sweden.

- [23] Putrus, G.A.; Suwanapingkarl, P.; Johnston, D.; Bentley, E.C.; Narayana, M.; ,  
"Impact of electric vehicles on power distribution networks," *Vehicle Power and Propulsion Conference, 2009. VPPC '09. IEEE* , vol., no., pp.827-831, 7-10 Sept. 2009
- [24] Shams-Zahraei, M.; Kouzani, A.Z.; , "A study on plug-in hybrid electric vehicles,"  
*TENCON 2009 - 2009 IEEE Region 10 Conference* , vol., no., pp.1-5, 23-26 Jan. 2009
- [25] Jarushi, A. M.; Schofield, N.; , "Battery and supercapacitor combination for a series hybrid electric vehicle," *Power Electronics, Machines and Drives (PEMD 2010), 5th IET International Conference on* , vol., no., pp.1-6, 19-21 April 2010
- [26] N. Jinrui, W. Zhifu, R. Qinglian, "Simulation and analyses of Performance of a Pure Electric Vehicle with a Super-capacitor", *Vehicle Power and Propulsion Conference, 2006. VPPC'06, IEEE, Sept. 2006*, pp. 1-6.
- [27] S. Sadeghi, L. Parsa, and H. A. Toliyat, "Extending speed range of five-phase PM machines by changing the stator windings connections," in *Electric Machines & Drives Conference (IEMDC), 2011 IEEE International*, 2011, pp. 1540-1545.
- [28] M. G. Simoes and P. Vieira, Jr., "A high torque low-speed multi-phase brushless machine a perspective application for electric vehicles," in *Industrial Electronics Society, 2000. IECON 2000. 26th Annual Conference of the IEEE*, 2000, pp. 1395-1400 vol.2.
- [29] L. Parsa, "On advantages of multi-phase machines," in *Industrial Electronics Society, 2005. IECON 2005. 31st Annual Conference of IEEE*, 2005, p. 6 pp.
- [30] K. S. Khan, W. M. Arshad, and S. Kanerva, "On performance figures of multiphase machines," in *Electrical Machines, 2008. ICEM 2008. 18th International Conference on*, 2008, pp. 1-5.
- [31] A. Gautam, S. Karugaba, and J. Ojo, "Modeling of nine-phase interior permanent magnet machines (IPM) including harmonic effects," in *Electric Machines & Drives Conference (IEMDC), 2011 IEEE International*, 2011, pp. 681-686.
- [32] S. Waikar, T. Gopalarathnam, H. A. Toliyat, and J. C. Moreira, "Evaluation of multiphase brushless permanent magnet (BPM) motors using finite element method (FEM) and experiments," in *Applied Power Electronics Conference and Exposition, 1999. APEC '99. Fourteenth Annual*, 1999, pp. 396-402 vol.1.
- [33] F. Scuiller, J. Charpentier, and E. Semail, "Multi-star multi-phase winding for a high power naval propulsion machine with low ripple torques and high fault tolerant

- ability," in *Vehicle Power and Propulsion Conference (VPPC), 2010 IEEE*, 2010, pp. 1-5.
- [34] N. Bianchi, S. Bolognani, and M. D. Pre, "Impact of Stator Winding of a Five-Phase Permanent-Magnet Motor on Postfault Operations," *Industrial Electronics, IEEE Transactions on*, vol. 55, pp. 1978-1987, 2008.
- [35] M. Barcaro, N. Bianchi, and F. Magnussen, "Faulty Operations of a PM Fractional-Slot Machine With a Dual Three-Phase Winding," *Industrial Electronics, IEEE Transactions on*, vol. 58, pp. 3825-3832, 2011.
- [36] O. Wen and T. A. Lipo, "Multiphase Modular Permanent Magnet Drive System Design and Realization," in *Electric Machines & Drives Conference, 2007. IEMDC '07. IEEE International*, 2007, pp. 787-792.
- [37] J. Riveros, B. Bogado, J. Prieto, F. Barrero, S. Toral, and M. Jones, "Multiphase machines in propulsion drives of electric vehicles," in *Power Electronics and Motion Control Conference (EPE/PEMC), 2010 14th International*, 2010, pp. T5-201-T5-206.
- [38] O. Wen, S. Huang, A. Good, and T. A. Lipo, "Modular permanent magnet machine based on soft magnetic composite," in *Electrical Machines and Systems, 2005. ICEMS 2005. Proceedings of the Eighth International Conference on*, 2005, pp. 235-239 Vol. 1.
- [39] X. Wang, Q. Wang, and Y. Wang, "Analysis of the magnetic field of the six-phase disc PMSM and effect of air gap length on the air gap magnetic field," in *Electrical Machines and Systems, 2008. ICEMS 2008. International Conference on*, 2008, pp. 3212-3215.
- [40] A. M. El-Refaie, M. R. Shah, J. P. Alexander, S. Galioto, H. Kum-Kang, and W. D. Gerstler, "Rotor end losses in multi-phase fractional-slot concentrated-winding permanent magnet synchronous machines," in *Energy Conversion Congress and Exposition (ECCE), 2010 IEEE*, 2010, pp. 1312-1320.
- [41] E. Fornasiero, N. Bianchi, and S. Bolognani, "Rotor losses in fractional-slot three-phase and five-phase PM machines," in *Electrical Machines (ICEM), 2010 XIX International Conference on*, 2010, pp. 1-5.
- [42] M. R. Shah and A. M. El-Refaie, "End Effects in Multiphase Fractional Slot Concentrated-Winding Surface Permanent Magnet Synchronous Machines," *Energy Conversion, IEEE Transactions on*, vol. 25, pp. 1001-1009, 2010.



- [43] C. Ho-Yong, P. Sun Jung, L. Man Li, Y. K. Kong, and J. G. Bin, "Efficiency compensation of multi-phase PM motor with redundant structure," in *Electrical Machines (ICEM), 2010 XIX International Conference on*, 2010, pp. 1-3.
- [44] F. Scuiller, J. F. Charpentier, E. Semail, and S. Clenet, "Comparison of two 5-phase Permanent Magnet machine winding configurations. Application on naval propulsion specifications," in *Electric Machines & Drives Conference, 2007. IEMDC '07. IEEE International*, 2007, pp. 34-39.
- [45] H. Oraee, "Prediction of transient behaviour of electrical machine windings," in *Electrotechnical Conference, 1991. Proceedings., 6th Mediterranean*, 1991, pp. 1433-1435 vol.2.
- [46] C. Kral, A. Haumer, and F. Pirker, "A Modelica Library for the Simulation of Electrical Asymmetries in Multiphase Machines - The ExtendedMachines Library," in *Diagnostics for Electric Machines, Power Electronics and Drives*, 2007. SDEMPED 2007.
- [47] R. Zanasi and F. Grossi, "Optimal rotor flux shape for multi-phase permanent magnet synchronous motors," in *Power Electronics and Motion Control Conference, 2008. EPE-PEMC 2008. 13th*, 2008, pp. 874-879.
- [48] Contin, A. Grava, A. Tassarolo, and G. Zocco, "A novel modeling approach to a multi-phase, high power synchronous machine," in *Power Electronics, Electrical Drives, Automation and Motion, 2006. SPEEDAM 2006. International Symposium on*, 2006, pp. 428-433.
- [49] K. Min, H. Jin, J. Hai-bo, and Y. Jia-qiang, "Principle and simulation of a 5-phase bearingless permanent magnet-type synchronous motor," in *Electrical Machines and Systems, 2008. ICEMS 2008. International Conference on*, 2008, pp. 1148-1152.
- [50] J. J. Wolmarans, H. Polinder, J. A. Ferreira, and D. Clarenbach, "Selecting an optimum number of system phases for an integrated, fault tolerant permanent magnet machine and drive," in *Power Electronics and Applications, 2009. EPE '09. 13th European Conference on*, 2009, pp. 1-10.
- [51] T. J. E. Miller and M. I. McGilp, "Analysis of multi-phase permanent-magnet synchronous machines," in *Electrical Machines and Systems, 2009. ICEMS 2009. International Conference on*, 2009, pp. 1-6.
- [52] S. Sadeghi and L. Parsa, "Multiobjective Design Optimization of Five-Phase Halbach Array Permanent-Magnet Machine," *Magnetics, IEEE Transactions on*, vol. 47, pp. 1658-1666, 2011.

- [53] Y. Kats, "Adjustable-speed drives with multiphase motors," in *Electric Machines and Drives Conference Record*, 1997. IEEE International, 1997, pp. TC2/4.1-TC2/4.3.
- [54] L. Cheng, W. Xiangheng, and N. Puqi, "The analysis of stator MMF and the calculation of rotor losses in a 12-phase high speed asynchronous generator," in *Electrical Machines and Systems*, 2005. ICEMS 2005. Proceedings of the Eighth International Conference on, 2005, pp. 690-694 Vol. 1.
- [55] S. Thomas, Z. Q. Zhu, R. L. Owen, G. W. Jewell, and D. Howe, "Multi-Phase Flux-Switching Permanent Magnet Brushless Machine for Aerospace Application," in *Industry Applications Society Annual Meeting*, 2008. IAS '08. IEEE, 2008, pp. 1-8.
- [56] M. Barcaro, N. Bianchi, and F. Magnussen, "Analysis and tests of a dual three-phase 12-slot 10-pole permanent magnet motor," in *Energy Conversion Congress and Exposition*, 2009. ECCE 2009. IEEE, 2009, pp. 3587-3594.
- [57] N. T. Irenji, "Comparison of an Active Stator and a 3-phase synchronous machine," in *Power Electronics and Applications (EPE 2011)*, Proceedings of the 2011-14th European Conference on, 2011, pp. 1-8.
- [58] Y. Amara, J. Wang, and D. Howe, "Stator iron loss of tubular permanent magnet machines," in *Industry Applications Conference*, 2004. 39th IAS Annual Meeting. Conference Record of the 2004 IEEE, 2004, pp. 2119-2124 vol.3.
- [59] S. Jang-Ho, C. Tae-Kyung, L. Cheol-Gyun, J. Sang-Yong, and J. Hyun-Kyo, "Harmonic Iron Loss Analysis of Electrical Machines for High-Speed Operation Considering Driving Condition," *Magnetics, IEEE Transactions on*, vol. 45, pp. 4656-4659, 2009.
- [60] R. Fang, "An improved iron loss calculation method for high-speed induction machine fed by nonsinusoidal voltage sources," in *Electrical Machines and Systems*, 2005. ICEMS 2005. Proceedings of the Eighth International Conference on, 2005, pp. 886-888 Vol. 2.
- [61] W. Haitian, L. Ge, C. Liang, H. Xuan, and W. Ruyi, "Iron Loss Reduction Method for Permanent Magnet Machine with Beveling Magnet," in *Power and Energy Engineering Conference (APPEEC)*, 2010 Asia-Pacific, pp. 1-4.
- [62] X. Jannot, J. C. Vannier, A. Kedous-Lebouc, C. Marchand, M. Gabsi, and J. Saint-Michel, "Analytical computation of stator iron losses in interior permanent-magnet synchronous machine," in *Electrical Machines (ICEM)*, 2010 XIX International Conference on, pp. 1-6.

- [63] Y. Pang, Z. Q. Zhu, D. Howe, S. Iwasaki, R. Deodhar, and A. Pride, "Investigation of iron loss in flux-switching PM machines," in *Power Electronics, Machines and Drives*, 2008. PEMD 2008. 4th IET Conference on, 2008, pp. 460-464.
- [64] M. Lei, M. Sanada, S. Morimoto, and Y. Takeda, "Prediction of iron loss in rotating machines with rotational loss included," *Magnetics, IEEE Transactions on*, vol. 39, pp. 2036-2041, 2003.
- [65] C. A. Hernandez-Aramburo, T. C. Green, and A. C. Smith, "Estimating rotational iron losses in an induction machine," *Magnetics, IEEE Transactions on*, vol. 39, pp. 3527-3533, 2003.
- [66] D. J. Powell, G. W. Jewell, S. D. Calverley, and D. Howe, "Iron loss in a modular rotor switched reluctance machine for the "More-Electric" aero-engine," *Magnetics, IEEE Transactions on*, vol. 41, pp. 3934-3936, 2005.
- [67] K. Yamazaki, "Torque and efficiency calculation of an interior permanent magnet motor considering harmonic iron losses of both the stator and rotor," *Magnetics, IEEE Transactions on*, vol. 39, pp. 1460-1463, 2003.
- [68] N. Sadowski, M. Lajoie-Mazenc, J. P. A. Bastos, M. V. Ferreira da Luz, and P. Kuo-Peng, "Evaluation and analysis of iron losses in electrical machines using the rain-flow method," *Magnetics, IEEE Transactions on*, vol. 36, pp. 1923-1926, 2000.
- [69] S. L. Ho, W. N. Fu, and H. C. Wong, "Estimation of stray losses of skewed rotor induction motors using coupled 2-D and 3-D time stepping finite element methods," *Magnetics, IEEE Transactions on*, vol. 34, pp. 3102-3105, 1998.
- [70] K. J. Tseng and S. B. Wee, "Analysis of flux distribution and core losses in interior permanent magnet motor," *Energy Conversion, IEEE Transactions on*, vol. 14, pp. 969-975, 1999.
- [71] K. Atallah, Z. Q. Zhu, and D. Howe, "An improved method for predicting iron losses in brushless permanent magnet DC drives," *Magnetics, IEEE Transactions on*, vol. 28, pp. 2997-2999, 1992.
- [72] W. Jiabin, T. Ibrahim, and D. Howe, "Prediction and Measurement of Iron Loss in a Short-Stroke, Single-Phase, Tubular Permanent Magnet Machine," *Magnetics, IEEE Transactions on*, vol. 46, pp. 1315-1318, June 2010
- [73] L. Ruifang, C. C. Mi, and D. W. Gao, "Modeling of Iron Losses of Electrical Machines and Transformers Fed by PWM Inverters," in *Power Engineering Society General Meeting*, 2007. IEEE, 2007, pp. 1-7.

- [74] M. Chunting, G. R. Slemon, and R. Bonert, "Modeling of iron losses of permanent-magnet synchronous motors," *Industry Applications, IEEE Transactions on*, vol. 39, pp. 734-742, 2003.
- [75] G. Ugalde, G. Almandoz, J. Poza, and A. Gonzalez, "Computation of iron losses in permanent magnet machines by multi-domain simulations," in *Power Electronics and Applications, 2009. EPE '09. 13th European Conference on*, 2009, pp. 1-10.
- [76] A. Cavagnino, M. Lazzari, A. Miotto, A. Tenconi, and S. Vaschetto, "Impact of the rotor back-iron resistivity on the rotor eddy-current losses in fractional-slot concentrated windings PM machines," in *Energy Conversion Congress and Exposition (ECCE), 2011 IEEE*, pp. 1604-1611.
- [77] N. Stranges and R. D. Findlay, "Importance of rotational iron loss data for accurate prediction of rotating machine core losses," in *Industry Applications Society Annual Meeting, 1994., Conference Record of the 1994 IEEE*, 1994, pp. 123-127 vol.1.
- [78] Z. Azar, L. J. Wu, D. Evans, and Z. Q. Zhu, "Influence of rotor configuration on iron and magnet losses of fractional-slot IPM machines," in *Power Electronics, Machines and Drives (PEMD 2010), 5th IET International Conference on*, pp. 1-6.
- [79] S. Jelassi, R. Romary, and J. F. Brudny, "Slot design for dynamic iron loss AC rotating electrical machine reduction," in *Energy Conversion Congress and Exposition (ECCE), 2010 IEEE*, pp. 563-569.
- [80] P. H. Mellor, R. Wrobel, and D. Holliday, "A computationally efficient iron loss model for brushless AC machines that caters for rated flux and field weakened operation," in *Electric Machines and Drives Conference, 2009. IEMDC '09. IEEE International*, 2009, pp. 490-494.
- [81] K. Yamazaki and N. Fukushima, "Experimental validation of iron loss model for rotating machines based on direct eddy current analysis of electrical steel sheets," in *Electric Machines and Drives Conference, 2009. IEMDC '09. IEEE International*, 2009, pp. 851-857.
- [82] F. Leonardi, T. Matsuo, and T. A. Lipo, "Iron loss calculation for synchronous reluctance machines," in *Power Electronics, Drives and Energy Systems for Industrial Growth, 1996., Proceedings of the 1996 International Conference on*, 1996, pp. 307-312 vol.1.
- [83] D. Buecherl and H. Herzog, "Iron loss modeling by complex inductances for steady state simulation of electrical machines," in *Power Electronics Electrical Drives*

- Automation and Motion (SPEEDAM), 2010 International Symposium on, pp. 878-883.
- [84] J. A. Walker, D. G. Dorrell, and E. Ritchie, "Iron Losses in Electrical Machines Due to Non Sinusoidal Alternating Fluxes," in *Power Electronics and Drive Systems, 2007. PEDS '07. 7th International Conference on*, 2007, pp. 974-980.
  - [85] A. G. Jack, B. C. Mecrow, and P. G. Dickinson, "Iron loss in machines with powdered iron stators," in *Electric Machines and Drives, 1999. International Conference IEMD '99*, 1999, pp. 48-50.
  - [86] B. G. Lamme, "Iron Losses in Direct-Current Machines," *American Institute of Electrical Engineers, Transactions of the*, vol. XXXV, pp. 261-299, 1916.
  - [87] K. Yamazaki and N. Fukushima, "Iron Loss Model for Rotating Machines Using Direct Eddy Current Analysis in Electrical Steel Sheets," *Energy Conversion, IEEE Transactions on*, vol. 25, pp. 633-641, Sept. 2010.
  - [88] K. Yamazaki, M. Tanida, and H. Satomi, "Iron Loss Analysis of Rotating Machines Considering Skin Effect and Excess Loss in Electrical Steel Sheets," in *Electromagnetic Field Computation, 2006 12th Biennial IEEE Conference on*, 2006, pp. 322-322.
  - [89] E. Dlala, A. Belahcen, and A. Arkkio, "On the Importance of Incorporating Iron Losses in the Magnetic Field Solution of Electrical Machines," *Magnetics, IEEE Transactions on*, vol. 46, pp. 3101-3104, Aug. 2010.
  - [90] J. T. Charton, J. Corda, J. M. Stephenson, and S. P. Randall, "Dynamic modelling of switched reluctance machines with iron losses and phase interactions," *Electric Power Applications, IEE Proceedings -*, vol. 153, pp. 327-336, 2006.
  - [91] G. Ugalde, Z. Q. Zhu, J. Poza, and A. Gonzalez, "Analysis of rotor eddy current loss in fractional slot permanent magnet machine with solid rotor back-iron," in *Electrical Machines (ICEM), 2010 XIX International Conference on*, pp. 1-6.
  - [92] K. Yamazaki and N. Fukushima, "Iron-Loss Modeling for Rotating Machines: Comparison Between Bertotti's Three-Term Expression and 3-D Eddy-Current Analysis," *Magnetics, IEEE Transactions on*, vol. 46, pp. 3121-3124, Aug. 2010.
  - [93] L. Li, A. Kedous-Lebouc, A. Foggia, and J. C. Mipo, "An iron loss model (loss surface) for FeCo sheet and its application in machine design," in *Electrical Machines (ICEM), 2010 XIX International Conference on*, pp. 1-6.
  - [94] P. Rasilo, A. Belahcen, and A. Arkkio, "Importance of Iron-Loss Modeling in Simulation of Wound-Field Synchronous Machines," *Magnetics, IEEE Transactions on*, vol. PP, pp. 1-1, Sept. 2012.

- [95] Z. Q. Zhu, Y. S. Chen, and D. Howe, "Iron loss in permanent-magnet brushless AC machines under maximum torque per ampere and flux weakening control," *Magnetics, IEEE Transactions on*, vol. 38, pp. 3285-3287, Sep 2002.
- [96] H. Domeki, Y. Ishihara, C. Kaido, Y. Kawase, S. Kitamura, T. Shimomura, N. Takahashi, T. Yamada, and K. Yamazaki, "Investigation of benchmark model for estimating iron loss in rotating machine," *Magnetics, IEEE Transactions on*, vol. 40, pp. 794-797, 2004.
- [97] M. Chunting, G. R. Slemon, and R. Bonert, "Minimization of iron losses of permanent magnet synchronous machines," in *Electrical Machines and Systems, 2001. ICEMS 2001. Proceedings of the Fifth International Conference on*, 2001, pp. 818-823 vol.2.
- [98] A. Boglieni, O. Bottauscio, M. Chiampi, M. Pastorelli, and M. Repetto, "Computation and measurement of iron losses under PWM supply conditions," *Magnetics, IEEE Transactions on*, vol. 32, pp. 4302-4304, 1996.
- [99] Enokizono, M., T. Suzuki, et al. (1990). "Rotational power loss of silicon steel sheet." *Magnetics, IEEE Transactions on* 26(5): 2562-2564.
- [100] Enokizono, M., G. Shirakawa, et al. (1991). "Two-Dimensional Magnetic Properties of Silicon-Steel Sheet." *Magnetics in Japan, IEEE Translation Journal on* 6(11): 937-946.
- [101] Atallah, K. and D. Howe (1993). "Calculation of the rotational power loss in electrical steel laminations from measured  $\vec{H}$  and  $\vec{B}$  ." *Magnetics, IEEE Transactions on* 29(6): 3547-3549.
- [102] Ma, L., M. Sanada, et al. (2003). "Iron loss prediction considering the rotational field and flux density harmonics in IPMSM and SynRM." *Electric Power Applications, IEE Proceedings -* 150(6): 747-751.
- [103] Hernandez-Aramburo, C. A., T. C. Green, et al. (2003). "Estimating rotational iron losses in an induction machine." *Magnetics, IEEE Transactions on* 39(6): 3527-3533.
- [104] J. Reinert, A. Brockmeyer, and R. W. A. A. De Doncker, "Calculation of losses in ferro- and ferrimagnetic materials based on the modified Steinmetz equation," *Industry Applications, IEEE Transactions on*, vol. 37, pp. 1055-1061, 2001.
- [105] A. Krings and J. Soulard, "Overview and Comparison of Iron Loss Models for Electrical Machines," *Journal of Electrical Engineering*, 2010.
- [106] L. Jeli, T. Abdallah, and C. R. Sullivan, "Improved calculation of core loss with nonsinusoidal waveforms," in *Industry Applications Conference, 2001. Thirty-Sixth*

*IAS Annual Meeting. Conference Record of the 2001 IEEE*, 2001, pp. 2203-2210  
vol.4.

- [107] C. P. Steinmetz, "On the law of hysteresis," *Proceedings of the IEEE*, vol. 72, pp. 197-221, 1984.
- [108] G. R. Slemon and X. Liu, "Core losses in permanent magnet motors," *Magnetics, IEEE Transactions on*, vol. 26, pp. 1653-1655, 1990.
- [109] G. Bertotti, A. Boglietti, M. Chiampi, D. Chiarabaglio, F. Fiorillo, and M. Lazzari, "An improved estimation of iron losses in rotating electrical machines," *Magnetics, IEEE Transactions on*, vol. 27, pp. 5007-5009, 1991.
- [110] K. Venkatachalam, C. R. Sullivan, T. Abdallah, and H. Tacca, "Accurate prediction of ferrite core loss with nonsinusoidal waveforms using only Steinmetz parameters," in *Computers in Power Electronics, 2002. Proceedings. 2002 IEEE Workshop on*, 2002, pp. 36-41.
- [111] W. A. Roshen, "A Practical, Accurate and Very General Core Loss Model for Nonsinusoidal Waveforms," *Power Electronics, IEEE Transactions on*, vol. 22, pp. 30-40, 2007.
- [112] I. Mayergoyz, "Mathematical models of hysteresis," *Magnetics, IEEE Transactions on*, vol. 22, pp. 603-608, 1986.
- [113] L. R. Dupre, O. Bottauscio, M. Chiampi, M. Repetto, and J. A. A. Melkebeek, "Modeling of electromagnetic phenomena in soft magnetic materials under unidirectional time periodic flux excitations," *Magnetics, IEEE Transactions on*, vol. 35, pp. 4171-4184, 1999.
- [114] J. R. Hendershot and T. Miller, "*Design of Brushless Permanent-magnet Motors*," 1994.
- [115] G.R. Slemon, A. Straughen, "Electric Machines," Book, 1982.
- [116] UK transport greenhouse gas emissions, <http://www.cabinetoffice.gov.uk/>, last accessed April 2014.
- [117] Data available on [www.parliament.uk/briefing-papers/POST-PN-365.pdf/](http://www.parliament.uk/briefing-papers/POST-PN-365.pdf/), last accessed April 2014.
- [118] Schofield, N., Mellor, P.H., and Howe, D.: 'High efficiency, PM brushless machine-based drive-train for a single-seat, three-wheel, all-electric vehicle' Proc. 32nd Int. Symp. on Automotive Technology and Automation (ISATA), Dedicated Conference on Clean Power Sources and Environmental Implications, Paper 99CPE072, pp.1-8, Vienna, 14-18 June 1999.
- [119] Power Conversion Group, School of Electrical and Electronic Engineering, University of Manchester, "Design of an Electrical Power-Train for the Formula Student Team" , Fourth Year Project Final Report ,May 2010.

# Appendix I

## Software coding

### I.1 Codes of Lua script, multi-data recoding.

```

.....
mi_seteditmode("group")
mi_selectgroup(1000)
mi_moverotate(0,0,step)

mi_analyze(1)
mi_loadsolution()
-- select the arc cross the middle of the first stator tooth
    mo_selectpoint(20,1.3)
    mo_selectpoint(20,-1.3)
-- flux linkage of 1st tooth
    flux_link_tooth1,x1 = mo_lineintegral(0);

mi_analyze(1)
mi_loadsolution()
-- select the arc cross the middle of the second stator tooth
    mo_selectpoint(18,8.1)
    mo_selectpoint(19,5.6)
-- flux linkage of 2nd tooth
    flux_link_tooth2,x2 = mo_lineintegral(0);
.....
    handle = fopen("test.txt","a")
    write(handle,time," \t", pm_angle, " \t",electrcial_tooth_angle," \t", flux_link_tooth1, " \t",
flux_link_tooth2, " \t", flux_link_tooth3," ", "\n")
    fclose(handle)
    mo_close()
end

```

### I.2 Examples of MATLAB code.

<pre> %load data disp(['loading data']) X=load('C:\.....txt') t=X(:,1); .....  %Calculate %rate of change:flux dB1=diff((y1'))'./dx1; % Calculate %Back-emf, e e1=N*w.*dB1; ..... </pre>	<pre> %assign variables %select data flux linkage for each tooth gap(universal) y1=X(:,4); ..... %number of turns of the coil N=10; .....  %plot %back emf-electrical angle Fig plot(x,e1,'o'); title('Back-emf, e for arc1&amp;2') xlabel('time, t'); ylabel('Back-emf, e'); ..... </pre>
--	--



### I.3 An example code for analyzing the circuit.

```
.....
mi_seteditmode("group")
mi_selectgroup(1000)
mi_moverotate(0,0,step)
    mi_analyze(1);
    mi_loadsolution();
    A1,A2,A3=mo_getcircuitproperties("A");
.....
    flux_link_backiron,x = mo_lineintegral(0);
.....
```

### I.4 Testing Lua script with loop.

File name: chpt3\_t3.2.lua

-- this is to calculate the flux linkage waveform of one coil and tooth, by the permanent magnet field only.  
-- stator windings are not fed. Armature reaction is not considered.  
-- note: to be able to run this code, all the parts of rotor must be indentified by number of 'group 1000' from properties of selected block

```
open("chpt3_3.6_test.FEM")
mi_saveas("temp1.fem")

--Correct initial position of the first permanent magnet, at 0 degree.
pm_angle=5
mi_seteditmode("group")
mi_selectgroup(1000)
mi_moverotate(0,0,pm_angle)
--n: how many points/steps on the linkage flux wave

n=72

electrical_step=360/n
polepair=1

-- rotation step of rotor given in degree
step= - electrical_step/polepair

m=10
for time= 1,m, 1 do

    if time==1 then
        handle = openfile("chpt3_t3.2.txt","a")
        write(handle,"time", "    \t", "pm_angle", "    \t", "electrcial_tooth_angle", "    \t",
"flux_link_tooth1", "    \t", "flux_link_tooth2", "    \t", "flux_link_tooth3", "    \t",
"flux_link_tooth4", "    \t", "flux_link_tooth5", "    \t", "flux_link_tooth6", "    \t",
\t", "flux_link_tooth7", "    \t", "flux_link_tooth8", "    \t", "flux_link_tooth9", "    \t",
\t", "flux_link_backiron_mid", "    \t", "flux_link_backiron_side", "    \t", "Torque1", "    \t", "
", "\n")
        closefile(handle)
    end
end
```

```

--angle position of the first magnet after the rotation.
pm_angle=pm_angle+step
--relative angle position of the first stator tooth corresponding to the first PM after
each rotation.
mechanical_tooth_angle=0-pm_angle
electrcial_tooth_angle=mechanical_tooth_angle*polepair

mi_seteditmode("group")
mi_selectgroup(1000)
mi_moverotate(0,0,step)

mi_analyze(1)
mi_loadsolution()
-- select the arc cross the middle of the first stator tooth,x axis+, clockwise
mo_selectpoint(58.52,5.52)
mo_selectpoint(58.52,-5.25)
-- flux linkage of 1st tooth
flux_link_tooth1,x1 = mo_lineintegral(0);
mo_clearcontour()

-- select the arc cross the middle of the second stator tooth
mo_selectpoint(48.2,-33.60)
mo_selectpoint(41.4573,-41.64)
-- flux linkage of 2nd tooth
flux_link_tooth2,x2 = mo_lineintegral(0);
mo_clearcontour()

-- select the arc cross the middle of the backiron slot mid
mo_selectpoint(-13.11,74.35)
mo_selectpoint(-12.18,69.08)
-- flux linkage of 1st tooth
flux_link_backiron_mid,x10 = mo_lineintegral(0);

mo_clearcontour()
-- select the arc cross the middle of the backiron slot side
mo_selectpoint(4.51,75.365)
mo_selectpoint(4.19,70.02)
-- flux linkage of 1st tooth
flux_link_backiron_side,x11 = mo_lineintegral(0);

mo_clearcontour()
-- select the arc in airgap clockwise
mo_selectpoint(45.75,0)
mo_selectpoint(32.35,-32.35)
mo_selectpoint(0,-45.75)
mo_selectpoint(-32.35,-32.35)
mo_selectpoint(-45.75,0)
mo_selectpoint(-32.35,32.35)
mo_selectpoint(0,45.75)

```

```

        mo_selectpoint(32.35,32.35)
        mo_selectpoint(45.75,0)
-- flux linkage of 1st tooth
        Torque1,x12 = mo_lineintegral(4);

        handle = fopen("chpt3_t3.2.txt","a")
        write(handle,time,"      \t", pm_angle, "      \t", electrical_tooth_angle, "      \t",
flux_link_tooth1, "      \t", flux_link_tooth2, "      \t",flux_link_backiron_mid, "
\t",flux_link_backiron_side, "      \t",Torque1,"      \t","      \n")
        fclose(handle)

    mo_close()

end

```

## I.5 Code used for extracting cogging torque.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Author:Xin Niu, PC Group, EEE, the University of Manchester
%Date: 21/05/2012
%Software: MATLAB R2009a Version 7.8.0.347 32-bit
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%versions update:
%130612:one step rotation and circuit property modification.
%200712:1.synchronize peak current with rotor position in order to
maintain
%      peak torque.
%230712:1.change code to 'rotate-analyze-savetofile'
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;
clear all;
%rotate model change current
addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
opendocument('model_original_mod_3phase_210812_mesh.FEM');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%model rotation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
current_peak = 0;
current_Ain = current_peak *sin(90/360*2*pi);
current_Aout = -current_peak *sin(90/360*2*pi);
current_Bin = current_peak *sin(( 90-120 )/360*2*pi);
current_Bout = - current_peak *sin(( 90-120 )/360*2*pi);
current_Cin = current_peak *sin(( 90+120 )/360*2*pi);
current_Cout = -current_peak *sin(( 90+120 )/360*2*pi);

n = 360;
electrical_step = 360/n;
polepair = 4;
step= - electrical_step/polepair;
m = 360 + n + 1;
%rotate to starting position (peak torque position-step)
pm_angle_P1 = (0+ electrical_step) /polepair;
mi_seteditmode('group');
mi_selectgroup(1000);

```

```

mi_moverotate(0,0,pm_angle_P1);
mi_clearselected()
name_temp = ['Temp_open.FEM'];
mi_saveas(name_temp);
pm_angle = pm_angle_P1;

mi_modifycircprop('A+', 1, current_Ain);
mi_modifycircprop('A-', 1, current_Aout);
mi_modifycircprop('B+', 1, current_Bin);
mi_modifycircprop('B-', 1, current_Bout);
mi_modifycircprop('C+', 1, current_Cin);
mi_modifycircprop('C-', 1, current_Cout);

torque_array = zeros(m,30);
line_integral_array = zeros(1,18);
for time= 1:m

    mi_seteditmode('group');
    mi_selectgroup(1000);
    mi_moverotate(0,0,step);
    mi_clearselected()

    pm_angle = pm_angle+step;
    mechanical_tooth_angle=0-pm_angle;
    electrcial_tooth_angle= mechanical_tooth_angle*polepair;

    %step from 0
    %filename_femm = ['motoring_step_' num2str(time-1) '.FEM'];
    %mi_saveas(filename_femm);
    mi_analyze(1);
    mi_loadsolution()

    mo_selectpoint(4.99232,58.5464);
    mo_selectpoint(15.3328,56.7231);
    line_integral_array(1:2) = mo_lineintegral(0);
    mo_clearcontour();

    mo_selectpoint(41.4572,41.6401);
    mo_selectpoint(48.2065,33.5967);
    line_integral_array(3:4) = mo_lineintegral(0);
    mo_clearcontour();

    mo_selectpoint(58.5239,5.25);
    mo_selectpoint(58.5239,-5.25);
    line_integral_array(5:6) = mo_lineintegral(0);
    mo_clearcontour();

    mo_selectpoint(48.2065,-33.5967);
    mo_selectpoint(41.4573,-41.6402);
    line_integral_array(7:8) = mo_lineintegral(0);
    mo_clearcontour();

    mo_selectpoint(15.3328,-56.7232);
    mo_selectpoint(4.99233,-58.5465);
    line_integral_array(9:10) = mo_lineintegral(0);
    mo_clearcontour();

```

```

mo_selectpoint(-24.7153,-53.3082);
mo_selectpoint(-33.8086,-48.0582);
line_integral_array(11:12) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-53.1989,-24.9498);
mo_selectpoint(-56.7901,-15.083);
line_integral_array(13:14) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-56.7901,15.083);
mo_selectpoint(-53.1989,24.9497);
line_integral_array(15:16) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-33.8086,48.0582);
mo_selectpoint(-24.7153,53.3081);
line_integral_array(17:18) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(45.75,0)
mo_selectpoint(32.35,-32.35)
mo_selectpoint(0,-45.75)
mo_selectpoint(-32.35,-32.35)
mo_selectpoint(-45.75,0)
mo_selectpoint(-32.35,32.35)
mo_selectpoint(0,45.75)
mo_selectpoint(32.35,32.35)
mo_selectpoint(45.75,0)
T = mo_lineintegral(4);
torque_array(time,:) = [time-1 electrcial_tooth_angle T(1) T(2)
pm_angle ...
    (electrcial_tooth_angle - 205 ) ...
    current_Ain current_Aout current_Bin current_Bout current_Cin
current_Cout ...
    line_integral_array];
mo_clearcontour()

mo_close()

end
filename_data = ['data_fix_open_mesh.mat'];
save(filename_data, 'torque_array');

```

## I.6 Code used to process torque data collected in Section 3.3.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Author:      Xin Niu
%Date:        04/05/2013
%Software:    MATLAB, Version 7.1.246(R14), Service pack 3, August02, 2005
%process data from FEMM
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Update:
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

close all
clear all

counter_current = 32;
counter_step = 370;

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Magnet design_Breadloaf
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
array_torque_breadloaf = zeros(370,130); %step|cogging|**total
torque**|**excit torque**|
array_mm_breadloaf = zeros(32,9); %current|max|min
current_table = [100:100:3000 3500 4000]';
load('Z:\Torque_current_rerun\torque_current_breadloaf\data_fix_open_mesh
.mat');
array_torque_breadloaf(:,1) = torque_array(1:370,2); %electric
position
array_torque_breadloaf(:,2) = torque_array(1:370,3); %cogging torque
array_mm_breadloaf(:,1) = current_table;
clear torque_array;

for current_count = 1: counter_current
    %3phase
    filename_load = ['data_fix_3phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_breadloaf\'
filename_load] );
    array_torque_breadloaf(:,current_count+2) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) -
array_torque_breadloaf(:,2); %define EM_torque
    array_torque_breadloaf(:,current_count+2+32)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_breadloaf(current_count,2) =
array_torque_breadloaf(a_index,1);
    array_mm_breadloaf(current_count,4) =
array_torque_breadloaf(b_index,1);
    array_mm_breadloaf(current_count,3) = a;
    array_mm_breadloaf(current_count,5) = b;

    clear filename_load torque_array torque_EM a b;

    %9phase
    filename_load = ['data_fix_9phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_breadloaf\'
filename_load] );
    array_torque_breadloaf(:,current_count+2+64) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) -
array_torque_breadloaf(:,2); %define EM_torque
    array_torque_breadloaf(:,current_count+2+32+64)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_breadloaf(current_count,6) =
array_torque_breadloaf(a_index,1);
    array_mm_breadloaf(current_count,8) =
array_torque_breadloaf(b_index,1);
    array_mm_breadloaf(current_count,7) = a;
    array_mm_breadloaf(current_count,9) = b;

    clear filename_load torque_array torque_EM a b;

```

```

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Magnet design_breadloaf_filled
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
array_torque_filled = zeros(370,130); %step|cogging|**total
torque**|**excit torque**|
array_mm_filled = zeros(32,9); %current|max|min
current_table = [100:100:3000 3500 4000]';
load('Z:\Torque_current_rerun\torque_current_filled\data_fix_open_mesh.ma
t');
array_torque_filled(:,1) = torque_array(1:370,2); %electric position
array_torque_filled(:,2) = torque_array(1:370,3); %cogging torque
array_mm_filled(:,1) = current_table;
clear torque_array;

for current_count = 1: counter_current
    %3phase
    filename_load = ['data_fix_3phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_filled\'
filename_load] );
    array_torque_filled(:,current_count+2) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) - array_torque_filled(:,2); %define
EM_torque
    array_torque_filled(:,current_count+2+32)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_filled(current_count,2) = array_torque_filled(a_index,1);
    array_mm_filled(current_count,4) = array_torque_filled(b_index,1);
    array_mm_filled(current_count,3) = a;
    array_mm_filled(current_count,5) = b;

    clear filename_load torque_array torque_EM a b;

    %9phase
    filename_load = ['data_fix_9phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_filled\'
filename_load] );
    array_torque_filled(:,current_count+2+64) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) - array_torque_filled(:,2); %define
EM_torque
    array_torque_filled(:,current_count+2+32+64)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_filled(current_count,6) = array_torque_filled(a_index,1);
    array_mm_filled(current_count,8) = array_torque_filled(b_index,1);
    array_mm_filled(current_count,7) = a;
    array_mm_filled(current_count,9) = b;

    clear filename_load torque_array torque_EM a b;

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Magnet design_Full pole-pitch arc

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
array_torque_radius = zeros(370,130); %step|cogging|**total
torque**|**excit torque**|
array_mm_radius = zeros(32,9); %current|max|min
current_table = [100:100:3000 3500 4000]';
load('Z:\Torque_current_rerun\torque_current_radius\data_fix_open_mesh.ma
t');
array_torque_radius(:,1) = torque_array(1:370,2); %electric position
array_torque_radius(:,2) = torque_array(1:370,3); %cogging torque
array_mm_radius(:,1) = current_table;
clear torque_array;

for current_count = 1: counter_current
    %3phase
    filename_load = ['data_fix_3phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_radius\'
filename_load] );
    array_torque_radius(:,current_count+2) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) - array_torque_radius(:,2); %define
EM_torque
    array_torque_radius(:,current_count+2+32)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_radius(current_count,2) = array_torque_radius(a_index,1);
    array_mm_radius(current_count,4) = array_torque_radius(b_index,1);
    array_mm_radius(current_count,3) = a;
    array_mm_radius(current_count,5) = b;

    clear filename_load torque_array torque_EM a b;

    %9phase
    filename_load = ['data_fix_9phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_radius\'
filename_load] );
    array_torque_radius(:,current_count+2+64) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) - array_torque_radius(:,2); %define
EM_torque
    array_torque_radius(:,current_count+2+32+64)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_radius(current_count,6) = array_torque_radius(a_index,1);
    array_mm_radius(current_count,8) = array_torque_radius(b_index,1);
    array_mm_radius(current_count,7) = a;
    array_mm_radius(current_count,9) = b;

    clear filename_load torque_array torque_EM a b;

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Magnet design_Full pole-pitch pseudo-radial
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
array_torque_segment = zeros(370,130); %step|cogging|**total
torque**|**excit torque**|

```



```

array_mm_segment = zeros(32,9); %current|max|min
current_table = [100:100:3000 3500 4000]';
load('Z:\Torque_current_rerun\torque_current_radius_segment\data_fix_open
_mesh.mat');
array_torque_segment(:,1) = torque_array(1:370,2); %electric position
array_torque_segment(:,2) = torque_array(1:370,3); %cogging torque
array_mm_segment(:,1) = current_table;
clear torque_array;

for current_count = 1: counter_current
    %3phase
    filename_load = ['data_fix_3phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_radius_segment\'
filename_load] );
    array_torque_segment(:,current_count+2) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) -
array_torque_segment(:,2); %define EM_torque
    array_torque_segment(:,current_count+2+32)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_segment(current_count,2) = array_torque_segment(a_index,1);
    array_mm_segment(current_count,4) = array_torque_segment(b_index,1);
    array_mm_segment(current_count,3) = a;
    array_mm_segment(current_count,5) = b;

    clear filename_load torque_array torque_EM a b;

    %9phase
    filename_load = ['data_fix_9phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_radius_segment\'
filename_load] );
    array_torque_segment(:,current_count+2+64) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) -
array_torque_segment(:,2); %define EM_torque
    array_torque_segment(:,current_count+2+32+64)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_segment(current_count,6) = array_torque_segment(a_index,1);
    array_mm_segment(current_count,8) = array_torque_segment(b_index,1);
    array_mm_segment(current_count,7) = a;
    array_mm_segment(current_count,9) = b;

    clear filename_load torque_array torque_EM a b;

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Magnet design_Full/pole-pitch and flat bottom
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
array_torque_smooth = zeros(370,130); %step|cogging|**total
torque**|**excit torque**|
array_mm_smooth = zeros(32,9); %current|max|min
current_table = [100:100:3000 3500 4000]';
load('Z:\Torque_current_rerun\torque_current_smooth\data_fix_open_mesh.ma
t');

```

```

array_torque_smooth(:,1) = torque_array(1:370,2);    %electric position
array_torque_smooth(:,2) = torque_array(1:370,3);    %cogging torque
array_mm_smooth(:,1) = current_table;
clear torque_array;

for current_count = 1: counter_current
    %3phase
    filename_load = ['data_fix_3phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_smooth\'
filename_load] );
    array_torque_smooth(:,current_count+2) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) - array_torque_smooth(:,2); %define
EM_torque
    array_torque_smooth(:,current_count+2+32)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_smooth(current_count,2) = array_torque_smooth(a_index,1);
    array_mm_smooth(current_count,4) = array_torque_smooth(b_index,1);
    array_mm_smooth(current_count,3) = a;
    array_mm_smooth(current_count,5) = b;

    clear filename_load torque_array torque_EM a b;

    %9phase
    filename_load = ['data_fix_9phase_'
num2str(current_table(current_count)) '.mat'];
    load(['Z:\Torque_current_rerun\torque_current_smooth\'
filename_load] );
    array_torque_smooth(:,current_count+2+64) = torque_array(1:370,3);
    torque_EM = torque_array(1:370,3) - array_torque_smooth(:,2); %define
EM_torque
    array_torque_smooth(:,current_count+2+32+64)= torque_EM;

    a = max(torque_EM);
    b = min(torque_EM);
    a_index = find(torque_EM == a);
    b_index = find(torque_EM == b);
    array_mm_smooth(current_count,6) = array_torque_smooth(a_index,1);
    array_mm_smooth(current_count,8) = array_torque_smooth(b_index,1);
    array_mm_smooth(current_count,7) = a;
    array_mm_smooth(current_count,9) = b;

    clear filename_load torque_array torque_EM a b;

end

%Re-meshed cogging torque
%load('Z:\Torque_current_rerun\torque_current_breadloaf\data_fix_open_mes
h.mat');
%load('Z:\Torque_current_rerun\torque_current_filled\data_fix_open_mesh.m
at');
%load('Z:\Torque_current_rerun\torque_current_radius\data_fix_open_mesh.m
at');
%load('Z:\Torque_current_rerun\torque_current_radius_segment\data_fix_ope
n_mesh.mat');
%load('Z:\Torque_current_rerun\torque_current_smooth\data_fix_open_mesh.m
at');

```

```

save('torque_analysis.mat', 'array_torque_breadloaf',
'array_torque_filled', ...
'array_torque_radius', 'array_torque_segment',
'array_torque_smooth', ...
'array_mm_breadloaf', 'array_mm_filled', 'array_mm_radius', ...
'array_mm_segment', 'array_mm_smooth');

```

### I.7 Code used to process flux and EMF data in Section 3.3.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%process data from FEMM
%emf.
%MATLAB, Version 7.1.246(R14), Service pack 3, August02, 2005
%Xin Niu
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all
clear all
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%load data
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
load('Z:\Torque_current_rerun\flux_open.mat');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%assign variables
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Mechanical angle
angle_me=flux_breadloaf(:,5);

%electrical angle
x1=flux_breadloaf(:,2);
x2=flux_filled(:,2);
x3=flux_smooth(:,2);
x4=flux_radius(:,2);
x5=flux_segment(:,2);

%flux linkage for first tooth
y11=flux_breadloaf(:,13);
y12=flux_breadloaf(:,15);
y13=flux_breadloaf(:,17);

y21=flux_filled(:,13);
y22=flux_filled(:,15);
y23=flux_filled(:,17);

y31=flux_smooth(:,13);
y32=flux_smooth(:,15);
y33=flux_smooth(:,17);

y41=flux_radius(:,13);
y42=flux_radius(:,15);
y43=flux_radius(:,17);

y51=flux_segment(:,13);
y52=flux_segment(:,15);

```

```

y53=flux_segment(:,17);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Obtained variables
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%t or step number
tt=(1:length(x1))';
%electrical angle in radius
x1_rd=2*pi*x1*360;
x2_rd=2*pi*x2*360;
x3_rd=2*pi*x3*360;
x4_rd=2*pi*x4*360;
x5_rd=2*pi*x5*360;
%number of turns
N=1;
%electrical angle advanced for each step
dx1=(diff(x1'))';
dx2=(diff(x2'))';
dx3=(diff(x3'))';
dx4=(diff(x4'))';
dx5=(diff(x5'))';
%time difference
dtt=(diff(tt'))';
%angular speed electrical in radius per second(or step)
w1=dx1./dtt/360*2*pi;
w2=dx2./dtt/360*2*pi;
w3=dx3./dtt/360*2*pi;
w4=dx4./dtt/360*2*pi;
w5=dx5./dtt/360*2*pi;

%rate of change in flux and emf
dB11=diff((y11'))'./dx1;
dB12=diff((y12'))'./dx1;
dB13=diff((y13'))'./dx1;

dB21=diff((y21'))'./dx2;
dB22=diff((y22'))'./dx2;
dB23=diff((y23'))'./dx2;

dB31=diff((y31'))'./dx3;
dB32=diff((y32'))'./dx3;
dB33=diff((y33'))'./dx3;

dB41=diff((y41'))'./dx4;
dB42=diff((y42'))'./dx4;
dB43=diff((y43'))'./dx4;

dB51=diff((y51'))'./dx5;
dB52=diff((y52'))'./dx5;
dB53=diff((y53'))'./dx5;

emf_9phase_11=N*w1.*dB11;
emf_9phase_12=N*w1.*dB12;
emf_9phase_13=N*w1.*dB13;

emf_9phase_21=N*w2.*dB21;
emf_9phase_22=N*w2.*dB22;
emf_9phase_23=N*w2.*dB23;

emf_9phase_31=N*w3.*dB31;
emf_9phase_32=N*w3.*dB32;
emf_9phase_33=N*w3.*dB33;

```

```

emf_9phase_41=N*w4.*dB41;
emf_9phase_42=N*w4.*dB42;
emf_9phase_43=N*w4.*dB43;

emf_9phase_51=N*w5.*dB51;
emf_9phase_52=N*w5.*dB52;
emf_9phase_53=N*w5.*dB53;

%manipulating emf data for 3phase oncentrated winding.
emf_3phase_1 = emf_9phase_11 - emf_9phase_12 + emf_9phase_13;
emf_3phase_2 = emf_9phase_21 - emf_9phase_22 + emf_9phase_23;
emf_3phase_3 = emf_9phase_31 - emf_9phase_32 + emf_9phase_33;
emf_3phase_4 = emf_9phase_41 - emf_9phase_42 + emf_9phase_43;
emf_3phase_5 = emf_9phase_51 - emf_9phase_52 + emf_9phase_53;

%rescale time axis
x=x1(1:length(dx1));

%rms voltage
rms1=sqrt(sum(emf_9phase_11.^2)/max(x));
rms2=sqrt(sum(emf_9phase_21.^2)/max(x));
rms3=sqrt(sum(emf_9phase_31.^2)/max(x));
rms4=sqrt(sum(emf_9phase_41.^2)/max(x));
rms5=sqrt(sum(emf_9phase_51.^2)/max(x));
disp(['rms1=',num2str(rms1)]);
disp(['rms2=',num2str(rms2)]);
disp(['rms3=',num2str(rms3)]);
disp(['rms4=',num2str(rms4)]);
disp(['rms5=',num2str(rms5)]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%plot the flux linkage(universal)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure
plot(x1,y11, 'b');
hold on
plot(x2,y21, 'k');
hold on
plot(x3,y31, 'g');
hold on
plot(x4,y41, 'r');
hold on
plot(x5,y51, 'm');
%xlim([-1 1100]);
%ylim([-1.3e-007 1.3e-007]);
xlabel('Electrical angle (Degree)','FontName','Times New Roman');
ylabel('Flux linkage (Wb)','FontName','Times New Roman');
set(gca,'FontName','Times New Roman');
%set(gca,'XTick',[0 45 90 135 180 225 270 315 360],'FontName','Times New Roman');
f = legend('Breadloaf','Embedded bread loaf','Full pole-pitch and flat bottom','Full pole-pitch arc','Full pole-pitch pseudo-radial');
set(f,'Location','NorthEast');

%fit to 1cycle(flux=0@310degree,row 311)
figure
plot(x1(1:361),y11(311:671), 'b');
hold on
plot(x2(1:361),y21(311:671), 'k');
hold on
plot(x3(1:361),y31(311:671), 'g');
hold on

```

```

plot(x4(1:361),y41(311:671), 'r');
hold on
plot(x5(1:361),y51(311:671), 'm');
xlim([0 360]);
%ylim([-1.3e-003 1.3e-003]);
xlabel('Electrical angle (Degree)','FontName','Times New Roman');
ylabel('Flux linkage (Wb)','FontName','Times New Roman');
%set(gca,'FontName','Times New Roman');
set(gca,'XTick',[0 45 90 135 180 225 270 315 360],'FontName','Times New Roman');
f = legend('Breadloaf','Embedded bread loaf','Full pole-pitch and flat bottom','Full pole-pitch arc','Full pole-pitch pseudo-radial');
set(f,'Location','NorthEast');

%fft for flux linkage 1 cycle.need to change value of z and t to get all 5
%spectrums and tables.
t = x1(1:360);
z = y11(311:670);
N = length(z);
Z = fft(z);
Z_magnitude = abs(Z);
Z_phase = angle(Z);
Z_magnitude_abs = Z_magnitude / (N/2); %absolute magnitude obtained by
k = 0:N-1;
T = 360;
ho = k; %creat harmonic order range
cutoff = ceil(N/2);
Z_cutoff = Z_magnitude_abs(1:cutoff);
ho_cutoff = ho(1:cutoff);
figure
stem(ho_cutoff, Z_cutoff,'LineWidth',2);
xlabel('Harmonic order','FontSize',16,'FontName','Times New Roman')
ylabel('Magnitude','FontSize',16,'FontName','Times New Roman');
xlim([0 20]);
%ylim([0 100]);
set(gca,'XTick',[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20],...
'FontSize',12,'FontName','Times New Roman');
grid on
box off

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%plot phase emf 9 phase
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure
plot(x,emf_9phase_11, 'b');
hold on
plot(x,emf_9phase_21, 'k');
hold on
plot(x,emf_9phase_31, 'g');
hold on
plot(x,emf_9phase_41, 'r');
hold on
plot(x,emf_9phase_51, 'm');
%xlim([-1 1100]);
%ylim([-1.3e-007 1.3e-007]);
xlabel('Electrical angle (Degree)','FontName','Times New Roman');
ylabel('EMF (Volts)','FontName','Times New Roman');
set(gca,'FontName','Times New Roman');
%set(gca,'XTick',[0 45 90 135 180 225 270 315 360],'FontName','Times New Roman');
f = legend('Breadloaf','Embedded bread loaf','Full pole-pitch and flat bottom','Full pole-pitch arc','Full pole-pitch pseudo-radial');
set(f,'Location','NorthEast');

```

```

%fit to 1cycle(emf=0@219degree,row 220)
figure
plot(x(1:361),emf_9phase_11(220:580), 'b');
hold on
plot(x(1:361),emf_9phase_21(220:580), 'k');
hold on
plot(x(1:361),emf_9phase_31(220:580), 'g');
hold on
plot(x(1:361),emf_9phase_41(220:580), 'r');
hold on
plot(x(1:361),emf_9phase_51(220:580), 'm');
xlim([0 360]);
%ylim([-1.3e-003 1.3e-003]);
xlabel('Electrical angle (Degree)','FontName','Times New Roman');
ylabel('EMF (Volts)','FontName','Times New Roman');
%set(gca,'FontName','Times New Roman');
set(gca,'XTick',[0 45 90 135 180 225 270 315 360],'FontName','Times New
Roman');
f = legend('Breadloaf','Embedded bread loaf','Full pole-pitch and flat
bottom','Full pole-pitch arc','Full pole-pitch pseudo-radial');
set(f,'Location','NorthEast');

%fft analysis phase emf 9phase.need to change value of z and t to get all
5
%spectrums and tables.
t = x1(1:360);
z = emf_9phase_11(220:579);
N = length(z);
Z = fft(z);
Z_magnitude = abs(Z);
Z_phase = angle(Z);
Z_magnitude_abs = Z_magnitude / (N/2); %absolute magnitude obtained by
k = 0:N-1;
T = 360;
ho = k; %creat harmonic order range
cutoff = ceil(N/2);
Z_cutoff = Z_magnitude_abs(1:cutoff);
ho_cutoff = ho(1:cutoff);
figure
stem(ho_cutoff, Z_cutoff,'LineWidth',2);
xlabel('Harmonic order','FontSize',16,'FontName','Times New Roman')
ylabel('Magnitude','FontSize',16,'FontName','Times New Roman');
xlim([0 20]);
%ylim([0 100]);
set(gca,'XTick',[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20],...
'FontSize',12,'FontName','Times New Roman');
grid on
box off

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%plot phase emf 3 phase
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure
plot(x,emf_3phase_1, 'b');
hold on
plot(x,emf_3phase_2, 'k');
hold on
plot(x,emf_3phase_3, 'g');
hold on
plot(x,emf_3phase_4, 'r');
hold on
plot(x,emf_3phase_5, 'm');
%xlim([-1 1100]);

```

```

%ylim([-1.3e-007 1.3e-007]);
xlabel('Electrical angle (Degree)','FontName','Times New Roman');
ylabel('EMF (Volts)','FontName','Times New Roman');
set(gca,'FontName','Times New Roman');
%set(gca,'XTick',[0 45 90 135 180 225 270 315 360],'FontName','Times New Roman');
f = legend('Breadloaf','Embedded bread loaf','Full pole-pitch and flat bottom','Full pole-pitch arc','Full pole-pitch pseudo-radial');
set(f,'Location','NorthEast');

%fit to 1cycle(emf=0@199degree,row 200)
figure
plot(x(1:361),emf_3phase_1(200:560), 'b');
hold on
plot(x(1:361),emf_3phase_2(200:560), 'k');
hold on
plot(x(1:361),emf_3phase_3(200:560), 'g');
hold on
plot(x(1:361),emf_3phase_4(200:560), 'r');
hold on
plot(x(1:361),emf_3phase_5(200:560), 'm');
xlim([0 360]);
%ylim([-1.3e-003 1.3e-003]);
xlabel('Electrical angle (Degree)','FontName','Times New Roman');
ylabel('EMF (Volts)','FontName','Times New Roman');
%set(gca,'FontName','Times New Roman');
set(gca,'XTick',[0 45 90 135 180 225 270 315 360],'FontName','Times New Roman');
f = legend('Breadloaf','Embedded bread loaf','Full pole-pitch and flat bottom','Full pole-pitch arc','Full pole-pitch pseudo-radial');
set(f,'Location','NorthEast');

%fft analysis phase emf 3-phase.need to change value of z and t to get all 5
%spectrums and tables.
t = x(1:360);
z = emf_3phase_1(200:559);
N = length(z);
Z = fft(z);
Z_magnitude = abs(Z);
Z_phase = angle(Z);
Z_magnitude_abs = Z_magnitude / (N/2); %absolute magnitude obtained by
k = 0:N-1;
T = 360;
ho = k; %creat harmonic order range
cutoff = ceil(N/2);
Z_cutoff = Z_magnitude_abs(1:cutoff);
ho_cutoff = ho(1:cutoff);
figure
stem(ho_cutoff, Z_cutoff,'LineWidth',2);
xlabel('Harmonic order','FontSize',16,'FontName','Times New Roman')
ylabel('Magnitude','FontSize',16,'FontName','Times New Roman');
xlim([0 20]);
%ylim([0 100]);
set(gca,'XTick',[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20],...
'FontSize',12,'FontName','Times New Roman');
grid on
box off

```

## I.8 Peak torque extracting program for 3-phase winding.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```



```

%Author:Xin Niu, PC Group, EEE, the University of Manchester
%Date: 21/05/2012
%Software: MATLAB R2009a Version 7.8.0.347 32-bit
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%versions update:
%130612:one step rotation and circuit property modeification.
%200712:1.synchronize peak current with rotor position in order to
maintain
%      peak torque.
%230712:1.change code to 'rotate-analyze-savetofile'
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;
clear all;
%rotate model change current
addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
opendocument('model_original_mod_3phase_210812.FEM');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%model rotation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
current_peak = 1000;
current_Ain = current_peak *sin(90/360*2*pi);
current_Aout = -current_peak *sin(90/360*2*pi);
current_Bin = current_peak *sin(( 90-120 )/360*2*pi);
current_Bout = - current_peak *sin(( 90-120 )/360*2*pi);
current_Cin = current_peak *sin(( 90+120 )/360*2*pi);
current_Cout = -current_peak *sin(( 90+120 )/360*2*pi);

n = 360;
electical_step = 360/n;
polepair = 4;
step= - electical_step/polepair;
m = n + 10;
%rotate to starting position (peak torque position-step)
pm_angle_P1 = (0+ electical_step) /polepair;
mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,pm_angle_P1);
mi_clearselected()
name_temp = ['Temp_3phase_' num2str(current_peak) '.FEM'];
mi_saveas(name_temp);
pm_angle = pm_angle_P1;

mi_modifycircprop('A+', 1, current_Ain);
mi_modifycircprop('A-', 1, current_Aout);
mi_modifycircprop('B+', 1, current_Bin);
mi_modifycircprop('B-', 1, current_Bout);
mi_modifycircprop('C+', 1, current_Cin);
mi_modifycircprop('C-', 1, current_Cout);

torque_array = zeros(m,30);
line_integral_array = zeros(1,18);
for time= 1:m

    mi_seteditmode('group');
    mi_selectgroup(1000);
    mi_moverotate(0,0,step);
    mi_clearselected()

```

```

pm_angle = pm_angle+step;
mechanical_tooth_angle=0-pm_angle;
electrcial_tooth_angle= mechanical_tooth_angle*polepair;

%step from 0
%filename_femm = ['motoring_step_' num2str(time-1) '.FEM'];
%mi_saveas(filename_femm);
mi_analyze(1);
mi_loadsolution()

mo_selectpoint(4.99232,58.5464);
mo_selectpoint(15.3328,56.7231);
line_integral_array(1:2) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(41.4572,41.6401);
mo_selectpoint(48.2065,33.5967);
line_integral_array(3:4) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(58.5239,5.25);
mo_selectpoint(58.5239,-5.25);
line_integral_array(5:6) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(48.2065,-33.5967);
mo_selectpoint(41.4573,-41.6402);
line_integral_array(7:8) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(15.3328,-56.7232);
mo_selectpoint(4.99233,-58.5465);
line_integral_array(9:10) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-24.7153,-53.3082);
mo_selectpoint(-33.8086,-48.0582);
line_integral_array(11:12) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-53.1989,-24.9498);
mo_selectpoint(-56.7901,-15.083);
line_integral_array(13:14) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-56.7901,15.083);
mo_selectpoint(-53.1989,24.9497);
line_integral_array(15:16) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-33.8086,48.0582);
mo_selectpoint(-24.7153,53.3081);
line_integral_array(17:18) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(45.75,0)
mo_selectpoint(32.35,-32.35)
mo_selectpoint(0,-45.75)
mo_selectpoint(-32.35,-32.35)
mo_selectpoint(-45.75,0)

```

```

mo_selectpoint(-32.35,32.35)
mo_selectpoint(0,45.75)
mo_selectpoint(32.35,32.35)
mo_selectpoint(45.75,0)
T = mo_lineintegral(4);
torque_array(time,:) = [time-1 electrcial_tooth_angle T(1) T(2)
pm_angle ...
    (electrcial_tooth_angle - 205 ) ...
    current_Ain current_Aout current_Bin current_Bout current_Cin
current_Cout ...
    line_integral_array];
mo_clearcontour()

mo_close()

end
filename_data = ['data_fix_3phase_' num2str(current_peak) '.mat'];
save(filename_data, 'torque_array');

```

## I.9 Peak torque extracting program for 9-phase winding.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Author:Xin Niu, PC Group, EEE, the University of Manchester
%Date: 21/05/2012
%Software: MATLAB R2009a Version 7.8.0.347 32-bit
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%versions update:
%130612:one step rotation and circuit property modeification.
%200712:1.synchronize peak current with rotor position in order to
maintain
%      peak torque.
%230712:1.change code to 'rotate-analyze-savetofile'
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;
clear all;
%rotate model change current
addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
opendocument('model_original_mod_9phase_210812.FEM');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%model rotation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
current_peak = 900;
current_Ain = current_peak *sin(( 90-40*0 )/360*2*pi);
current_Aout = -current_peak *sin(( 90-40*0 )/360*2*pi);
current_Bin = current_peak *sin(( 90-40*1 )/360*2*pi);
current_Bout = - current_peak *sin(( 90-40*1 )/360*2*pi);
current_Cin = current_peak *sin(( 90-40*2 )/360*2*pi);
current_Cout = -current_peak *sin(( 90-40*2 )/360*2*pi);
current_Din = current_peak *sin(( 90-40*3 )/360*2*pi);
current_Dout = -current_peak *sin(( 90-40*3 )/360*2*pi);
current_Ein = current_peak *sin(( 90-40*4 )/360*2*pi);
current_Eout = - current_peak *sin(( 90-40*4 )/360*2*pi);
current_Fin = current_peak *sin(( 90-40*5 )/360*2*pi);
current_Fout = -current_peak *sin((90-40*5 )/360*2*pi);
current_Gin = current_peak *sin(( 90-40*6 )/360*2*pi);
current_Gout = -current_peak *sin(( 90-40*6 )/360*2*pi);
current_Hin = current_peak *sin(( 90-40*7 )/360*2*pi);
current_Hout = - current_peak *sin(( 90-40*7 )/360*2*pi);

```

```

current_Iin = current_peak *sin(( 90-40*8)/360*2*pi);
current_Iout = -current_peak *sin(( 90-40*8 )/360*2*pi);

n = 360;
electrical_step = 360/n;
polepair = 4;
step= - electrical_step/polepair;
m = n + 10;
%rotate to starting position (peak torque position-step)
pm_angle_P1 = (0+ electrical_step) /polepair;
mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,pm_angle_P1);
mi_clearselected()
name_temp = ['Temp_9phase_' num2str(current_peak) '.FEM'];
mi_saveas(name_temp);
pm_angle = pm_angle_P1;

mi_modifycircprop('A+', 1, current_Ain);
mi_modifycircprop('A-', 1, current_Aout);
mi_modifycircprop('B+', 1, current_Bin);
mi_modifycircprop('B-', 1, current_Bout);
mi_modifycircprop('C+', 1, current_Cin);
mi_modifycircprop('C-', 1, current_Cout);
mi_modifycircprop('D+', 1, current_Din);
mi_modifycircprop('D-', 1, current_Dout);
mi_modifycircprop('E+', 1, current_Ein);
mi_modifycircprop('E-', 1, current_Eout);
mi_modifycircprop('F+', 1, current_Fin);
mi_modifycircprop('F-', 1, current_Fout);
mi_modifycircprop('G+', 1, current_Gin);
mi_modifycircprop('G-', 1, current_Gout);
mi_modifycircprop('H+', 1, current_Hin);
mi_modifycircprop('H-', 1, current_Hout);
mi_modifycircprop('I+', 1, current_Iin);
mi_modifycircprop('I-', 1, current_Iout);

torque_array = zeros(m,42);
line_integral_array = zeros(1,18);
for time= 1:m

    mi_seteditmode('group');
    mi_selectgroup(1000);
    mi_moverotate(0,0,step);
    mi_clearselected()

    pm_angle = pm_angle+step;
    mechanical_tooth_angle=0-pm_angle;
    electrcial_tooth_angle= mechanical_tooth_angle*polepair;

    %step from 0
    %filename_femm = ['motoring_step_' num2str(time-1) '.FEM'];
    %mi_saveas(filename_femm);
    mi_analyze(1);
    mi_loadsolution()

    mo_selectpoint(4.99232,58.5464);
    mo_selectpoint(15.3328,56.7231);
    line_integral_array(1:2) = mo_lineintegral(0);
    mo_clearcontour();

```

```

mo_selectpoint(41.4572,41.6401);
mo_selectpoint(48.2065,33.5967);
line_integral_array(3:4) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(58.5239,5.25);
mo_selectpoint(58.5239,-5.25);
line_integral_array(5:6) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(48.2065,-33.5967);
mo_selectpoint(41.4573,-41.6402);
line_integral_array(7:8) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(15.3328,-56.7232);
mo_selectpoint(4.99233,-58.5465);
line_integral_array(9:10) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-24.7153,-53.3082);
mo_selectpoint(-33.8086,-48.0582);
line_integral_array(11:12) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-53.1989,-24.9498);
mo_selectpoint(-56.7901,-15.083);
line_integral_array(13:14) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-56.7901,15.083);
mo_selectpoint(-53.1989,24.9497);
line_integral_array(15:16) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(-33.8086,48.0582);
mo_selectpoint(-24.7153,53.3081);
line_integral_array(17:18) = mo_lineintegral(0);
mo_clearcontour();

mo_selectpoint(45.75,0)
mo_selectpoint(32.35,-32.35)
mo_selectpoint(0,-45.75)
mo_selectpoint(-32.35,-32.35)
mo_selectpoint(-45.75,0)
mo_selectpoint(-32.35,32.35)
mo_selectpoint(0,45.75)
mo_selectpoint(32.35,32.35)
mo_selectpoint(45.75,0)
T = mo_lineintegral(4);
mo_clearcontour();
torque_array(time,:) = [time-1 electrcial_tooth_angle T(1) T(2)
pm_angle ...
    (electrcial_tooth_angle - 205 ) ...
    current_Ain current_Aout current_Bin current_Bout current_Cin
current_Cout ...
    current_Din current_Dout current_Ein current_Eout current_Fin
current_Fout ...

```

```

        current_Gin current_Gout current_Hin current_Hout current_Iin
current_Iout ...
        line_integral_array];

mo_clearcontour()

mo_close()

end
filename_data = ['data_fix_9phase_' num2str(current_peak) '.mat'];
save(filename_data, 'torque_array');

```

## I.10 Analyzing program for demagnetization checking.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Author:Xin Niu, PC Group, EEE, the University of Manchester
%Date: 21/05/2012
%Software: MATLAB R2009a Version 7.8.0.347 32-bit
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%versions update:
%130612:one step rotation and circuit property modeification.
%200712:1.synchronize peak current with rotor position in order to
maintain
%      peak torque.
%230712:1.change code to 'rotate-analyze-savetofile'
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;
clear all;
%rotate model change current
addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
model_name = 'model_radius_segment_3phase';
opendocument([model_name '.FEM']);
electrcial_tooth_angle = 281;
    current_peak = 1000;
    current_Ain = current_peak *sin(90/360*2*pi);
    current_Aout = -current_peak *sin(90/360*2*pi);
    current_Bin = current_peak *sin(( 90-120 )/360*2*pi);
    current_Bout = - current_peak *sin(( 90-120 )/360*2*pi);
    current_Cin = current_peak *sin(( 90+120 )/360*2*pi);
    current_Cout = -current_peak *sin(( 90+120 )/360*2*pi);

%prompt = 'What peak current do you desired? in AT?? \n';

polepair = 4;
mechanical_tooth_angle = electrcial_tooth_angle/polepair;
pm_angle = 0 - mechanical_tooth_angle;

name_temp = ['snapshot_' model_name '_' num2str(current_peak) '_'
num2str(electrcial_tooth_angle) '.FEM'];
mi_saveas(name_temp);

mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,pm_angle);
mi_analyze(1);

```

```

name_load = ['snapshot_' model_name '_' num2str(current_peak) '_'
num2str(electrcial_tooth_angle) '.ans'];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%.ans file read and identification
%convert string cell to data array
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%fid = fopen('model_original_mod.ans');
%fid = fopen('test_mesh1.ans');
%fid = fopen('test_mesh2.ans');
%fid = fopen('demagnetization1.ans');
%fid = fopen('snapshot_17_phase_1500_259.ans');
fid = fopen(name_load);
%fid = fopen('snapshot_17_phase_1000_101.ans');
%fid = fopen('demagnetization1.ans');

C = textscan(fid, '%s %s %s %s %s %s %s %s %s %s');
fclose(fid)

%Check title and locate useful data:[NumBlockLabels] and [Solution]
title1 = C{1,1};
title2 = C{1,2};
TF_blocksearch = strcmp(title1, '[NumBlockLabels]');
index_block = find(TF_blocksearch == 1);

TF_solutionsearch = strcmp(title1, '[Solution]');
index_solution = find(TF_solutionsearch == 1);
%block:convert cellto data array
counter_block = (index_solution-index_block-1);
%block_count = zeros(counter_block ,1);
block = zeros(counter_block ,9);

for block_count = 1 : counter_block
    block((block_count),:) = [ ...
        str2num(str2mat(C{1}(index_block+block_count))), ...
        str2num(str2mat(C{2}(index_block+block_count))), ...
        str2num(str2mat(C{3}(index_block+block_count))), ...
        str2num(str2mat(C{4}(index_block+block_count))), ...
        str2num(str2mat(C{5}(index_block+block_count))), ...
        str2num(str2mat(C{6}(index_block+block_count))), ...
        str2num(str2mat(C{7}(index_block+block_count))), ...
        str2num(str2mat(C{8}(index_block+block_count))), ...
        str2num(str2mat(C{9}(index_block+block_count))) ...
    ];

end

%search for empty cell within solution section
%'[Solution]' '' <-----start from this row(index_Solution)
%'65' '' <-----start from this row(index_node)
%'node_x' 'node_y' 'A'
TF_empty = strcmp(title2((index_solution+1):end), '');
index_empty = find(TF_empty == 1);

%nodes list start@index_solutionsearch+index_empty(1)
index_node = index_solution+index_empty(1);
%mesh info start@index_solutionsearch+index_empty(2)
index_mesh = index_solution+index_empty(2);
%end of mesh info@index_solutionsearch+index_empty(3)
index_end = index_solution+index_empty(3);

```

```

%node:convert celltodata array
counter_node = (index_mesh - index_node - 1);
%node_count = zeros(counter_node, 1);
node = zeros(counter_node, 3);
for node_count = 1 : counter_node
    node(node_count, :) = [ ...
        str2num(str2mat(C{1}(index_node+node_count))), ...
        str2num(str2mat(C{2}(index_node+node_count))), ...
        str2num(str2mat(C{3}(index_node+node_count))) ...
    ];
end

%mesh:convert celltodata array
counter_mesh = (index_end - index_mesh - 1);
%mesh_count = zeros(counter_mesh, 1);
mesh = zeros(counter_mesh, 4);
for mesh_count = 1 : counter_mesh
    mesh(mesh_count, :) = [ ...
        str2num(str2mat(C{1}(index_mesh+mesh_count))), ...
        str2num(str2mat(C{2}(index_mesh+mesh_count))), ...
        str2num(str2mat(C{3}(index_mesh+mesh_count))), ...
        str2num(str2mat(C{4}(index_mesh+mesh_count))) ...
    ];
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%locate correlated data(index block,node and mesh)
%spareate into array according to index
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%search material prop list: magnet is material No.2
index_block_magnet = find(block(:,3)==2);

counter_magnet_cell = length(index_block_magnet);
magnet_cell = cell(length(index_block_magnet),1);
%magnet_cell_count = zeros(counter_magnet_cell, 1);
area_mesh_magnet_cell = cell(length(index_block_magnet),1);
B_cell = cell(length(index_block_magnet),1);
B_H_curve_cell = cell(length(index_block_magnet),1);
for magnet_cell_count = 1 : counter_magnet_cell
    %search magnet(nth)data in mash list(block(4,:));
    %block number in mesh list <---> block index-1
    index_mesh_magnet = ...
        find(mesh(:,4)==(index_block_magnet(magnet_cell_count)-1));

    counter_magnet = length(index_mesh_magnet);
    %magnet_count = zeros(counter_magnet, 1);
    mesh_magnet = zeros(counter_magnet, 4);
    magnet = zeros(counter_magnet,9);
    area_mesh_magnet = zeros(counter_magnet, 1);
    B = zeros(counter_magnet, 1);
    H = zeros(counter_magnet, 1);
    Bx = zeros(counter_magnet, 1);
    By = zeros(counter_magnet, 1);
    B_angle = zeros(counter_magnet, 1);
    B_H_curve = zeros(counter_magnet, 4);
    angle_magnetization = zeros(counter_magnet, 1);
    Bx_n = zeros(counter_magnet, 1);
    By_n = zeros(counter_magnet, 1);
    B_n = zeros(counter_magnet, 1);
    H_xaxis = zeros(counter_magnet, 1);
    %put node info into magnet,a row for each count.
    for magnet_count = 1 : counter_magnet
        mesh_magnet(magnet_count, :) = ...
            mesh(index_mesh_magnet(magnet_count),:);
    end
end

```



```

%node index -1 -->mesh(1/2/3,:)
magnet(magnet_count,:) = [ ...
    node(mesh_magnet((magnet_count), 1)+1, :), ...
    node(mesh_magnet((magnet_count), 2)+1, :), ...
    node(mesh_magnet((magnet_count), 3)+1, :), ...
];

%'magnet' contain node list of nth count and will be rewritten
%in next count.calculation for|B|of magnet mesh element better
%insert here,within the loop,save the extraction from a cell
%with another for loop.
%unit of a,b,c in mm, area in m^2
a = sqrt((magnet(magnet_count,7)-magnet(magnet_count,1)).^2 ...
    +(magnet(magnet_count,8)-magnet(magnet_count,2)).^2);
b = sqrt((magnet(magnet_count,4)-magnet(magnet_count,1)).^2 ...
    +(magnet(magnet_count,5)-magnet(magnet_count,2)).^2);
c = sqrt((magnet(magnet_count,7)-magnet(magnet_count,4)).^2 ...
    +(magnet(magnet_count,8)-magnet(magnet_count,5)).^2);
area_mesh_magnet(magnet_count) = ...
    (sqrt((a+b+c)*(-a+b+c)*(a-b+c)*(a+b-c))./4)./1000000;

Bx(magnet_count) = ...
    (magnet(magnet_count,4)- magnet(magnet_count,1))./1000 ...
    * magnet(magnet_count,9)/(2*area_mesh_magnet(magnet_count))...
    + (magnet(magnet_count,7)- magnet(magnet_count,4))./1000 ...
    * magnet(magnet_count,3)/(2*area_mesh_magnet(magnet_count))...
    +(magnet(magnet_count,1)- magnet(magnet_count,7))./1000 ...
    * magnet(magnet_count,6)/(2*area_mesh_magnet(magnet_count));
By(magnet_count) = ...
    (magnet(magnet_count,5)- magnet(magnet_count,2))./1000 ...
    * magnet(magnet_count,9)/(2*area_mesh_magnet(magnet_count))...
    + (magnet(magnet_count,8)- magnet(magnet_count,5))./1000 ...
    * magnet(magnet_count,3)/(2*area_mesh_magnet(magnet_count))...
    +(magnet(magnet_count,2)- magnet(magnet_count,8))./1000 ...
    * magnet(magnet_count,6)/(2*area_mesh_magnet(magnet_count));
B(magnet_count) = ...
    sqrt((Bx(magnet_count)).^2 +(By(magnet_count)).^2);

%work out the angle between Bx and By.
%logic:
%1.Bx=0angle = 90
%2.Bx<0(sec or third)    angle = atan + 180
%3.Bx>=0&By>=0(first)   angle = atan
%4.Bx>=0&By<0(fourth)   angle = atan + 360
if Bx(magnet_count) == 0
    B_angle(magnet_count) = 90;
elseif Bx(magnet_count) < 0
    B_angle(magnet_count) = ...
        atan(By(magnet_count)/Bx(magnet_count))/(2*pi)*360 + 180;
elseif By(magnet_count) >= 0
    B_angle(magnet_count) = ...
        atan(By(magnet_count)/Bx(magnet_count))/(2*pi)*360;
else
    B_angle(magnet_count) = ...
        atan(By(magnet_count)/Bx(magnet_count))/(2*pi)*360 + 360;
end

H(magnet_count) = (B(magnet_count) - 1.25)/(1.25/950000);

%angle of magnetization(other material return 0) and Bn

```

```

    %Bx_n=Bx * cos(angle_magnetization)
    %By_n=By * sin(angle_magnetization)
    angle_magnetization(magnet_count)= ...
        block(index_block_magnet(magnet_cell_count),6);
    Bx_n(magnet_count) = ...
        Bx(magnet_count) *
        cos(angle_magnetization(magnet_count)/360*2*(pi));
    By_n(magnet_count) = ...
        By(magnet_count) *
        sin(angle_magnetization(magnet_count)/360*2*(pi));

    B_n(magnet_count)= ...
        Bx_n(magnet_count) + By_n(magnet_count);
    H_xaxis(magnet_count) = ...
        (B_n(magnet_count) - 1.25)/(1.25/950000);

    B_H_curve(magnet_count,:) = [H(magnet_count), B(magnet_count), ...
        H_xaxis(magnet_count), B_n(magnet_count)];
%B_H_curve_cell header: H | B | H_xaxis | B_n
%H = (B - 1.25)/(1.25/950000) | B = sqrt(Bx.^2 +By.^2) | H_xaxis =
%(B_n - 1.25)/(1.25/950000) | B_n = Bx_n + By_n
    end

    B_cell{magnet_cell_count} = [Bx, By, B, B_angle,
    angle_magnetization, ...
        Bx_n, By_n, B_n];
    B_H_curve_cell{magnet_cell_count} = B_H_curve;
    magnet_cell{magnet_cell_count} = magnet;
    area_mesh_magnet_cell{magnet_cell_count} = area_mesh_magnet;

end

%pick up and mesh less than 0.2T in a pole pair(possible demagnetized)
dm_magnet_cell = cell(length(B_cell),1);

for i = 1: length(B_cell)
    index_dm = find(B_cell{i}(:,8)< 0.2);
    TF_dm = isempty(index_dm);
    if TF_dm == 1
        dm_mesh = [];
    else
        dm_mesh = zeros(length(index_dm),17);
        for counter = 1:length(index_dm)
            dm_mesh(counter,:) = [magnet_cell{i}(index_dm(counter),:)
            B_cell{i}(index_dm(counter),:)];
        end
    end
    dm_magnet_cell{i} = dm_mesh;
end

clear C;
name_savedata = ['data_snapshot_' model_name '_' num2str(current_peak)
    '_' num2str(electrcial_tooth_angle)];
save (name_savedata);

closefemm;
%save test.mat
%d = {'x1','y1','A1','x2','y2','A2','x3','y3','A3','Bx','By','B_abs', ...
    %     'B_angle','PM_angle','Bx_c','By_c','B.n'};
%xlswrite('test.xls',d);
%hold on
%xlswrite('test.xls', magnet_cell{1}, 1, 'A2');

```

```
%hold on
%xlswrite('test.xls', B_cell{1}, 1, 'J2');
```

## I.11 Analyzing program for analyzing excitation torque for 3-phase.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Author:Xin Niu, PC Group, EEE, the University of Manchester
%Date: 21/05/2012
%Software: MATLAB R2009a Version 7.8.0.347 32-bit
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%versions update:
%130612:one step rotation and circuit property modeification.
%200712:1.synchronize peak current with rotor position in order to
maintain
%      peak torque.
%230712:1.change code to 'rotate-analyze-savetofile'
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;
clear all;
%rotate model change current
addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
opendocument('model_original_mod_3phase_210812.FEM');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%model rotation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n = 36;
electrical_step = 360/n;
polepair = 4;
step= - electrical_step/polepair;
m = n+2;
%rotate to starting position (peak torque position-step)
pm_angle_P1 = (-0 + electrical_step) /polepair;
mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,pm_angle_P1);
mi_clearselected()
mi_saveas('motoring_step_P1.FEM');
pm_angle = pm_angle_P1;

torque_array = zeros(m,12);
for time= 1:m

    mi_seteditmode('group');
    mi_selectgroup(1000);
    mi_moverotate(0,0,step);
    mi_clearselected()

    pm_angle = pm_angle+step;
    mechanical_tooth_angle=0-pm_angle;
    electrcial_tooth_angle= mechanical_tooth_angle*polepair;
    current_peak = 1000;
    current_Ain = current_peak *cos((-electrcial_tooth_angle + 111 -
(360/3)*0 )/360*2*pi);
    current_Aout = -current_peak *cos((-electrcial_tooth_angle + 111 -
(360/3)*0 )/360*2*pi);
```

```

    current_Bin = current_peak*cos((-electrcial_tooth_angle + 111 -
(360/3)*1 )/360*2*pi);
    current_Bout = - current_peak *cos((-electrcial_tooth_angle + 111 -
(360/3)*1 )/360*2*pi);
    current_Cin = current_peak *cos((-electrcial_tooth_angle + 111 -
(360/3)*2 )/360*2*pi);
    current_Cout = -current_peak *cos((-electrcial_tooth_angle + 111 -
(360/3)*2 )/360*2*pi);

    mi_modifycircprop('A+', 1, current_Ain);
    mi_modifycircprop('A-', 1, current_Aout);
    mi_modifycircprop('B+', 1, current_Bin);
    mi_modifycircprop('B-', 1, current_Bout);
    mi_modifycircprop('C+', 1, current_Cin);
    mi_modifycircprop('C-', 1, current_Cout);

    %step from 0
    filename_femm = ['motoring_step_' num2str(time-1) '.FEM'];
    mi_saveas(filename_femm);
    mi_analyze(1);
    mi_loadsolution()

    mo_selectpoint(45.75,0)
    mo_selectpoint(32.35,-32.35)
    mo_selectpoint(0,-45.75)
    mo_selectpoint(-32.35,-32.35)
    mo_selectpoint(-45.75,0)
    mo_selectpoint(-32.35,32.35)
    mo_selectpoint(0,45.75)
    mo_selectpoint(32.35,32.35)
    mo_selectpoint(45.75,0)
    T = mo_lineintegral(4);
    torque_array(time,:) = [time-1 electrcial_tooth_angle T(1) T(2)
pm_angle ...
    (electrcial_tooth_angle - 19 ) ...
    current_Ain current_Aout current_Bin current_Bout current_Cin
current_Cout];

    mi_clearselected()

    mo_close()

end
filename_data = ['torque_' num2str(current_peak) '.mat'];
save(filename_data, 'torque_array');

```

## I.9 Automatic analyzing program for electrical capability research.

```

clear all;
close all;
prompt = 'How many phase does your machine have?please type in from
3,5,7,9,10,11,13,14,17. \n';
phase_number = input(prompt);
prompt = 'How many pole pair does your machine have? \n';
polepair = input(prompt);
prompt = 'What precision do you desired? in degree/step electric? \n';
electrical_step = input(prompt);
prompt = 'What peak current do you desired? in AT?? \n';
current_peak = input(prompt);

```

```

prompt = 'Please type in your machine model file name?Please type the
full name and single quote your model \n';
model_name = input(prompt);

addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
opendocument([model_name '.FEM']);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%model rotation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

n = 360/electrical_step;
step= - electrical_step/polepair;
m = n + 10;
%rotate to starting position
pm_angle_P1 = (0+ electrical_step) /polepair;
mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,pm_angle_P1);
mi_clearselected()
name_temp = ['Temp_open_' model_name '.FEM'];
mi_saveas(name_temp);
pm_angle = pm_angle_P1;

%current profile.
run(['current_profile_' num2str(phase_number) '_phase_open']);

array_torque_current_open = zeros(m,54);
array_flux_linkage = zeros(m,24*2*3+3);
circuit_prop_array = zeros(1,24*2*3);
for time= 1:m

    mi_seteditmode('group');
    mi_selectgroup(1000);
    mi_moverotate(0,0,step);
    mi_clearselected()

    pm_angle = pm_angle+step;
    mechanical_tooth_angle=0-pm_angle;
    electrcial_tooth_angle= mechanical_tooth_angle*polepair;

    %step from 0
    %filename_femm = ['motoring_step_' num2str(time-1) '.FEM'];
    %mi_saveas(filename_femm);
    %Tooth flux linkage measurements
    mi_analyze(1);
    mi_loadsolution()

    %Torque measurements
    mo_selectpoint(45.75,0)
    mo_selectpoint(32.35,-32.35)
    mo_selectpoint(0,-45.75)
    mo_selectpoint(-32.35,-32.35)
    mo_selectpoint(-45.75,0)
    mo_selectpoint(-32.35,32.35)
    mo_selectpoint(0,45.75)
    mo_selectpoint(32.35,32.35)

```

```

mo_selectpoint(45.75,0)
T = mo_lineintegral(4);
mo_clearcontour()

array_torque_current_open(time,:) = [time-1 electrcial_tooth_angle
T(1) T(2) pm_angle ...
    (electrcial_tooth_angle) ...
    current_Ain current_Aout current_Bin current_Bout current_Cin
current_Cout ...
    current_Din current_Dout current_Ein current_Eout current_Fin
current_Fout ...
    current_Gin current_Gout current_Hin current_Hout current_Iin
current_Iout ...
    current_Jin current_Jout current_Kin current_Kout current_Lin
current_Lout ...
    current_Min current_Mout current_Nin current_Nout current_Oin
current_Oout ...
    current_Pin current_Pout current_Qin current_Qout current_Rin
current_Rout ...
    current_Sin current_Sout current_Tin current_Tout current_Uin
current_Uout ...
    current_Vin current_Vout current_Win current_Wout current_Xin
current_Xout];

%Circuit flux linkage(Current&Volt&FLux)
circuit_prop_array(1:3) = mo_getcircuitproperties('A+');
circuit_prop_array(4:6) = mo_getcircuitproperties('A-');
circuit_prop_array(7:9) = mo_getcircuitproperties('B+');
circuit_prop_array(10:12) = mo_getcircuitproperties('B-');
circuit_prop_array(13:15) = mo_getcircuitproperties('C+');
circuit_prop_array(16:18) = mo_getcircuitproperties('C-');
circuit_prop_array(19:21) = mo_getcircuitproperties('D+');
circuit_prop_array(22:24) = mo_getcircuitproperties('D-');
circuit_prop_array(25:27) = mo_getcircuitproperties('E+');
circuit_prop_array(28:30) = mo_getcircuitproperties('E-');
circuit_prop_array(31:33) = mo_getcircuitproperties('F+');
circuit_prop_array(34:36) = mo_getcircuitproperties('F-');
circuit_prop_array(37:39) = mo_getcircuitproperties('G+');
circuit_prop_array(40:42) = mo_getcircuitproperties('G-');
circuit_prop_array(43:45) = mo_getcircuitproperties('H+');
circuit_prop_array(46:48) = mo_getcircuitproperties('H-');
circuit_prop_array(49:51) = mo_getcircuitproperties('I+');
circuit_prop_array(52:54) = mo_getcircuitproperties('I-');
circuit_prop_array(55:57) = mo_getcircuitproperties('J+');
circuit_prop_array(58:60) = mo_getcircuitproperties('J-');
circuit_prop_array(61:63) = mo_getcircuitproperties('K+');
circuit_prop_array(64:66) = mo_getcircuitproperties('K-');
circuit_prop_array(67:69) = mo_getcircuitproperties('L+');
circuit_prop_array(70:72) = mo_getcircuitproperties('L-');
circuit_prop_array(73:75) = mo_getcircuitproperties('M+');
circuit_prop_array(76:78) = mo_getcircuitproperties('M-');
circuit_prop_array(79:81) = mo_getcircuitproperties('N+');
circuit_prop_array(82:84) = mo_getcircuitproperties('N-');
circuit_prop_array(85:87) = mo_getcircuitproperties('O+');
circuit_prop_array(88:90) = mo_getcircuitproperties('O-');
circuit_prop_array(91:93) = mo_getcircuitproperties('P+');
circuit_prop_array(94:96) = mo_getcircuitproperties('P-');
circuit_prop_array(97:99) = mo_getcircuitproperties('Q+');
circuit_prop_array(100:102) = mo_getcircuitproperties('Q-');
circuit_prop_array(103:105) = mo_getcircuitproperties('R+');
circuit_prop_array(106:108) = mo_getcircuitproperties('R-');
circuit_prop_array(109:111) = mo_getcircuitproperties('S+');

```

```

circuit_prop_array(112:114) = mo_getcircuitproperties('S-');
circuit_prop_array(115:117) = mo_getcircuitproperties('T+');
circuit_prop_array(118:120) = mo_getcircuitproperties('T-');
circuit_prop_array(121:123) = mo_getcircuitproperties('U+');
circuit_prop_array(124:126) = mo_getcircuitproperties('U-');
circuit_prop_array(127:129) = mo_getcircuitproperties('V+');
circuit_prop_array(130:132) = mo_getcircuitproperties('V-');
circuit_prop_array(133:135) = mo_getcircuitproperties('W+');
circuit_prop_array(136:138) = mo_getcircuitproperties('W-');
circuit_prop_array(139:141) = mo_getcircuitproperties('X+');
circuit_prop_array(142:144) = mo_getcircuitproperties('X-');

array_flux_linkage(time,:) = [time-1 electrcial_tooth_angle
pm_angle ...

                                circuit_prop_array];

mo_close()

end

%Calculate EMF in with two methods:line_integral and circuit property
t = array_flux_linkage(:,1);
angle_electric = array_flux_linkage(:,2);
angle_mechanical = 0 - array_flux_linkage(:,3);
N = 1;
flux_circuit = array_flux_linkage(:,6) - array_flux_linkage(:,9);
%rate of change:time
dt=(diff(t'))';
%rate of change: electrical angle
dx1=(diff(angle_electric'))';
%angular velocity
w=dx1./dt/360*2*pi;

%rate of change:flux density
dB_circuit = diff((flux_circuit'))'./dx1;

%Back-emf, e
emf_circuit=N*w.*dB_circuit;

emf_x = angle_electric(1:length(dx1));
array_emf = [emf_x emf_circuit];
filename_data = ['data_fix_' model_name '_' num2str(current_peak_open)
'.mat'];
save(filename_data,
'array_torque_current_open','array_flux_linkage','array_emf');

d = {'e_angle','emf_circuit','cogging torque','Time','e_angle',
'pm_angle', ...
'Current','Volts','Phi_A+','Current','Volts','Phi_A-
','Current','Volts','Phi_B+','Current','Volts','Phi_B-', ...
'Current','Volts','Phi_C+','Current','Volts','Phi_C-
','Current','Volts','Phi_D+','Current','Volts','Phi_D-',...
'Current','Volts','Phi_E+','Current','Volts','Phi_E-
','Current','Volts','Phi_F+','Current','Volts','Phi_F-',...
'Current','Volts','Phi_G+','Current','Volts','Phi_G-
','Current','Volts','Phi_H+','Current','Volts','Phi_H-',...
'Current','Volts','Phi_I+','Current','Volts','Phi_I-
','Current','Volts','Phi_J+','Current','Volts','Phi_J-',...
'Current','Volts','Phi_K+','Current','Volts','Phi_K-
','Current','Volts','Phi_L+','Current','Volts','Phi_L-',...

```

```

        'Current','Volts','Phi_M+', 'Current','Volts','Phi_M-
    ', 'Current','Volts','Phi_N+', 'Current','Volts','Phi_N-',...
        'Current','Volts','Phi_O+', 'Current','Volts','Phi_O-
    ', 'Current','Volts','Phi_P+', 'Current','Volts','Phi_P-',...
        'Current','Volts','Phi_Q+', 'Current','Volts','Phi_Q-
    ', 'Current','Volts','Phi_R+', 'Current','Volts','Phi_R-',...
        'Current','Volts','Phi_S+', 'Current','Volts','Phi_S-
    ', 'Current','Volts','Phi_T+', 'Current','Volts','Phi_T-',...
        'Current','Volts','Phi_U+', 'Current','Volts','Phi_U-
    ', 'Current','Volts','Phi_V+', 'Current','Volts','Phi_V-',...
        'Current','Volts','Phi_W+', 'Current','Volts','Phi_W-
    ', 'Current','Volts','Phi_X+', 'Current','Volts','Phi_X-'};

xlswrite(['data_open_' model_name '.xls'],d);
hold on
xlswrite(['data_open_' model_name '.xls'], array_emf, 1, 'A2');
hold on
xlswrite(['data_open_' model_name '.xls'], array_torque_current_open(:,3),
1, 'C2');
hold on
xlswrite(['data_open_' model_name '.xls'], array_flux_linkage, 1, 'D2');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Change Ipeak to 1000At and motoring the model with decreate Sinsoidal
%current
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;
openfemm;
opendocument([model_name '.FEM']);

n = 360/electrical_step;
step= - electrical_step/polepair;
m = n + 10;
%rotate to starting position (peak torque position-step)
pm_angle_P1 = (0+ electrical_step) /polepair;
mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,pm_angle_P1);
mi_clearselected()
name_temp = ['Temp_fix_' model_name '_' num2str(current_peak) '.FEM'];
mi_saveas(name_temp);
pm_angle = pm_angle_P1;

run(['current_profile_' num2str(phase_number) '_phase_fix']);

array_torque_fix_1000 = zeros(m,54);
for time= 1:m

    mi_seteditmode('group');
    mi_selectgroup(1000);
    mi_moverotate(0,0,step);
    mi_clearselected()

    pm_angle = pm_angle+step;
    mechanical_tooth_angle=0-pm_angle;
    electrcial_tooth_angle= mechanical_tooth_angle*polepair;

    %step from 0
    %filename_femm = ['motoring_step_' num2str(time-1) '.FEM'];
    %mi_saveas(filename_femm);
    %Tooth flux linkage measurements
    mi_analyze(1);

```



```

mi_loadsolution()

%Torque measurements
mo_selectpoint(45.75,0)
mo_selectpoint(32.35,-32.35)
mo_selectpoint(0,-45.75)
mo_selectpoint(-32.35,-32.35)
mo_selectpoint(-45.75,0)
mo_selectpoint(-32.35,32.35)
mo_selectpoint(0,45.75)
mo_selectpoint(32.35,32.35)
mo_selectpoint(45.75,0)
T = mo_lineintegral(4);
mo_clearcontour()

array_torque_fix_1000(time,:) = [time-1 electrcial_tooth_angle T(1)
T(2) pm_angle ...
    (electrcial_tooth_angle) ...
    current_Ain current_Aout current_Bin current_Bout current_Cin
current_Cout ...
    current_Din current_Dout current_Ein current_Eout current_Fin
current_Fout ...
    current_Gin current_Gout current_Hin current_Hout current_Iin
current_Iout ...
    current_Jin current_Jout current_Kin current_Kout current_Lin
current_Lout ...
    current_Min current_Mout current_Nin current_Nout current_Oin
current_Oout ...
    current_Pin current_Pout current_Qin current_Qout current_Rin
current_Rout ...
    current_Sin current_Sout current_Tin current_Tout current_Uin
current_Uout ...
    current_Vin current_Vout current_Win current_Wout current_Xin
current_Xout];

mo_close()

end
filename_data = ['data_fix_' model_name '_' num2str(current_peak) '.mat'];
save(filename_data, 'array_torque_fix_1000');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Find the Tpeak position and use it motoring the model
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;
torque_EM = array_torque_fix_1000(:,3) - array_torque_current_open(:,3);
a = max(torque_EM);
b = min(torque_EM);
a_index = find(torque_EM == a);
b_index = find(torque_EM == b);
position_a = array_torque_fix_1000(a_index,2);
Position_b = array_torque_fix_1000(b_index,2);
peak_info = [position_a a Position_b b];
openfemm;
opendocument([model_name '.FEM']);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%model rotation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n = 360/electical_step;
step= - electrical_step/polepair;
m = n + 10;
%rotate to starting position (peak torque position-step)

```

```

pm_angle_P1 = (0+ electrical_step) /polepair;
mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,pm_angle_P1);
mi_clearselected()
name_temp = ['Temp_motoring_' model_name '_' num2str(current_peak)
'.FEM'];
mi_saveas(name_temp);
pm_angle = pm_angle_P1;

torque_array = zeros(m,54);
for time= 1:m

    mi_seteditmode('group');
    mi_selectgroup(1000);
    mi_moverotate(0,0,step);
    mi_clearselected()

    pm_angle = pm_angle+step;
    mechanical_tooth_angle=0-pm_angle;
    electrcial_tooth_angle= mechanical_tooth_angle*polepair;
    run(['current_profile_' num2str(phase_number) '_phase_motoring']);

    %step from 0
    %filename_femm = ['motoring_step_' num2str(time-1) '.FEM'];
    %mi_saveas(filename_femm);
    mi_analyze(1);
    mi_loadsolution()

    mo_selectpoint(45.75,0)
    mo_selectpoint(32.35,-32.35)
    mo_selectpoint(0,-45.75)
    mo_selectpoint(-32.35,-32.35)
    mo_selectpoint(-45.75,0)
    mo_selectpoint(-32.35,32.35)
    mo_selectpoint(0,45.75)
    mo_selectpoint(32.35,32.35)
    mo_selectpoint(45.75,0)
    T = mo_lineintegral(4);
    torque_array(time,:) = [time-1 electrcial_tooth_angle T(1) T(2)
pm_angle ...
    (electrcial_tooth_angle - position_a ) ...
    current_Ain current_Aout current_Bin current_Bout current_Cin
current_Cout ...
    current_Din current_Dout current_Ein current_Eout current_Fin
current_Fout ...
    current_Gin current_Gout current_Hin current_Hout current_Iin
current_Iout ...
    current_Jin current_Jout current_Kin current_Kout current_Lin
current_Lout ...
    current_Min current_Mout current_Nin current_Nout current_Oin
current_Oout ...
    current_Pin current_Pout current_Qin current_Qout current_Rin
current_Rout ...
    current_Sin current_Sout current_Tin current_Tout current_Uin
current_Uout ...
    current_Vin current_Vout current_Win current_Wout current_Xin
current_Xout];

    mo_clearcontour()

    mo_close()

```

```

end
filename_data = ['torque_motoring_' model_name '_' num2str(current_peak)
'.mat'];
save(filename_data, 'torque_array');
d1 = {'Position A', 'Tpeak_motoring', 'Position B', 'Tpeak_generating'};
d2 = {'e_angle', 'Torque'};
d3 = {'Time', 'e_angle', 'torque_motoring', '', 'pm_angle', 'e_angle-
position_a', ...
'current_A+', 'current_A-', 'current_B+', 'current_B-
', 'current_C+', 'current_C-', ...
'current_D+', 'current_D-', 'current_E+', 'current_E-
', 'current_F+', 'current_F-', ...
'current_G+', 'current_G-', 'current_H+', 'current_H-
', 'current_I+', 'current_I-', ...
'current_J+', 'current_J-', 'current_K+', 'current_K-
', 'current_L+', 'current_L-', ...
'current_M+', 'current_M-', 'current_N+', 'current_N-
', 'current_O+', 'current_O-', ...
'current_P+', 'current_P-', 'current_Q+', 'current_Q-
', 'current_R+', 'current_R-', ...
'current_S+', 'current_S-', 'current_T+', 'current_T-
', 'current_U+', 'current_U-', ...
'current_V+', 'current_V-', 'current_W+', 'current_W-
', 'current_X+', 'current_X-'};

xlswrite(['data_motoring_' model_name '_' num2str(current_peak)
'.xls'],d1);
hold on
xlswrite(['data_motoring_' model_name '_' num2str(current_peak)
'.xls'],peak_info, 1, 'A2');
hold on
xlswrite(['data_motoring_' model_name '_' num2str(current_peak)
'.xls'],d2, 1, 'A4');
hold on
xlswrite(['data_motoring_' model_name '_' num2str(current_peak) '.xls'],
array_torque_fix_1000(:,2), 1, 'A5');
hold on
xlswrite(['data_motoring_' model_name '_' num2str(current_peak) '.xls'],
array_torque_fix_1000(:,3), 1, 'B5');
hold on
xlswrite(['data_motoring_' model_name '_' num2str(current_peak)
'.xls'],d3, 1, 'D4');
hold on
xlswrite(['data_motoring_' model_name '_' num2str(current_peak) '.xls'],
torque_array, 1, 'D5');

```

## I.12 Analyzing program for testing blocks.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Author:Xin Niu, PC Group, EEE, the University of Manchester
%Date: 21/05/2012
%Software: MATLAB R2009a Version 7.8.0.347 32-bit
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%versions update:
%earlier version:
%1.femm .ans field detection and data conversion;
%2.block,node and mesh list extraction;3.mesh elements list extraxtion
%('magnet'=[x1,y1,A1,x2,y2,A2,x3,y3,A3]);

```

```

%4.calculation for B field from mesh and creat'B_cell'=
%[Bx,By,B,B_angle,angle_magnetization,Bx_n,By_n,B_n] to store info.
%5.fix calculation of Bx,By in unit length.
%
%180612:1.scaled x-axis in BH plot(10^3);2.add plot for each magnet
%190612:1.fix the +-sign of Bx,By in magnetization direction.
%200612:1.add average function for B.
%160712:1.coordinate of test block;2.B block intergral,B_av calculation
%       and B_av_angle for test block.
%190712:1.add torque measuring;2.varied current supply to maintain peak
%       torque.
%260712:1.simplify the code,no need for mesh info;2.delete magnet info,
%       replaced with the testing blocks on magnets.3.correct mistake in
%       caculating 'B_av_dom_magnet'.
%170812:1.added programme block for searching 'NumPoints' and index the
%       points which form the arc or contour;2.added lineintegral for all
%       the side of each tesing block,in clockwise direction from left
%       side in reference position(point@left bottom corner is point_1);
3.
%       model reference file: 'model_original_mod_9phase_190812;4. fix
%       rotational angle for testing blocks on rotor.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%analysis
clear all;
close all;

addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
step = 360/1440;
offset = 0;
counter_step = 5761;
%using one cell for each testing block
testblock_magnet_cell = cell(11,1);
testblock_steel_cell = cell(16,1);
torque_array = zeros(counter_step,5);
line_integral_array = zeros(1,8);
for step_count= 1:counter_step

    filename_load = ['motoring_step_' num2str(step_count-1) '.ans'];
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %process in each step
    %.ans file read and identification
    %convert string cell to data array
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %fid = fopen('model_original_mod.ans');
    %fid = fopen('test_mesh1.ans');
    %fid = fopen('test_mesh2.ans');
    %fid = fopen('demagnetization1.ans');
    %fid = fopen('noAcopper.ans');
    fid = fopen(filename_load);
    %fid = fopen('demagnetization1.ans');

    C = textscan(fid, '%s %s %s %s %s %s %s %s %s %s');
    fclose(fid)

    %Check title and locate useful data:[NumBlockLabels] and [Solution]
    title1 = C{1,1};
    title2 = C{1,2};
    %block index
    TF_blocksearch = strcmp(title1, '[NumBlockLabels]');

```

```

index_block = find(TF_blocksearch == 1);

%solution index
TF_solutionsearch = strcmp(title1, '[Solution]');
index_solution = find(TF_solutionsearch == 1);

%points index
TF_pointsearch = strcmp(title1, '[NumPoints]');
index_point = find(TF_pointsearch == 1);

%block:convert celltodata array
counter_block = (index_solution-index_block-1);
%block_count = zeros(counter_block ,1);
block = zeros(counter_block ,9);

for block_count = 1 : counter_block
    block((block_count),:) = [ ...
        str2num(str2mat(C{1}(index_block+block_count))), ...
        str2num(str2mat(C{2}(index_block+block_count))), ...
        str2num(str2mat(C{3}(index_block+block_count))), ...
        str2num(str2mat(C{4}(index_block+block_count))), ...
        str2num(str2mat(C{5}(index_block+block_count))), ...
        str2num(str2mat(C{6}(index_block+block_count))), ...
        str2num(str2mat(C{7}(index_block+block_count))), ...
        str2num(str2mat(C{8}(index_block+block_count))), ...
        str2num(str2mat(C{9}(index_block+block_count))) ...
    ];

end

%point:convert celltodata array
counter_point = (index_solution - index_point - 1);
point = zeros(counter_point ,4);
for point_count = 1 : counter_point
    point((point_count),:) = [ ...
        str2num(str2mat(C{1}(index_point+point_count))), ...
        str2num(str2mat(C{2}(index_point+point_count))), ...
        str2num(str2mat(C{3}(index_point+point_count))), ...
        str2num(str2mat(C{4}(index_point+point_count))), ...
    ];
end

%point:index measuring points for each testing block

%11 teting blocks on magnet
%format of index array:
%test_block_1--->   x1  y1  x2  y2  x3  y3  x4  y4
%test_block_2--->   x1  y1  x2  y2  x3  y3  x4  y4
index_point_measuring_magnet = [ ...
    point(222,1) point(222,2) point(224,1) point(224,2) point(223,1)
point(223,2) point(221,1) point(221,2); ...
    point(214,1) point(214,2) point(220,1) point(220,2) point(219,1)
point(219,2) point(213,1) point(213,2); ...
    point(204,1) point(204,2) point(202,1) point(202,2) point(201,1)
point(201,2) point(203,1) point(203,2); ...
    point(215,1) point(215,2) point(217,1) point(217,2) point(218,1)
point(218,2) point(216,1) point(216,2); ...
    point(233,1) point(233,2) point(228,1) point(228,2) point(227,1)
point(227,2) point(234,1) point(234,2); ...
    point(229,1) point(229,2) point(208,1) point(208,2) point(207,1)
point(207,2) point(230,1) point(230,2); ...

```

```

        point(234,1) point(234,2) point(195,1) point(195,2) point(196,1)
point(196,2) point(231,1) point(231,2); ...
        point(226,1) point(226,2) point(206,1) point(206,2) point(205,1)
point(205,2) point(225,1) point(225,2); ...
        point(211,1) point(211,2) point(212,1) point(212,2) point(194,1)
point(194,2) point(193,1) point(193,2); ...
        point(199,1) point(199,2) point(200,1) point(200,2) point(198,1)
point(198,2) point(197,1) point(197,2); ...
        point(191,1) point(191,2) point(192,1) point(192,2) point(210,1)
point(210,1) point(209,1) point(209,2)];

%16 tesing block on steel
index_point_measuring_steel = [ ...
        point(157,1) point(157,2) point(158,1) point(158,2) point(156,1)
point(156,2) point(155,1) point(155,2); ...
        point(151,1) point(151,2) point(152,1) point(152,2) point(154,1)
point(154,2) point(153,1) point(153,2); ...
        point(159,1) point(159,2) point(160,1) point(160,2) point(162,1)
point(162,2) point(161,1) point(161,2); ...
        point(328,1) point(328,2) point(280,1) point(280,2) point(279,1)
point(279,2) point(327,1) point(327,2); ...
        point(237,1) point(237,2) point(239,1) point(239,2) point(240,1)
point(240,2) point(238,1) point(238,2); ...
        point(243,1) point(243,2) point(244,1) point(244,2) point(246,1)
point(246,2) point(245,1) point(245,2); ...
        point(248,1) point(248,2) point(247,1) point(247,2) point(241,1)
point(241,2) point(242,1) point(242,2); ...
        point(91, 1) point(91, 2) point(336,1) point(336,2) point(335,1)
point(335,2) point(97, 1) point(97, 2); ...
        point(163,1) point(163,2) point(164,1) point(164,2) point(166,1)
point(166,2) point(165,1) point(165,2); ...
        point(167,1) point(167,2) point(168,1) point(168,2) point(170,1)
point(170,2) point(169,1) point(169,2); ...
        point(173,1) point(173,2) point(174,1) point(174,2) point(172,1)
point(172,2) point(171,1) point(171,2); ...
        point(236,1) point(236,2) point(24, 1) point(24, 2) point(35, 1)
point(35, 2) point(235,1) point(235,2); ...
        point(175,1) point(175,2) point(177,1) point(177,2) point(178,1)
point(178,2) point(176,1) point(176,2); ...
        point(190,1) point(190,2) point(189,1) point(189,2) point(187,1)
point(187,2) point(188,1) point(188,2); ...
        point(181,1) point(181,2) point(179,1) point(179,2) point(180,1)
point(180,2) point(182,1) point(182,2); ...
        point(185,1) point(185,2) point(183,1) point(183,2) point(184,1)
point(184,2) point(186,1) point(186,2)];

opendocument(filename_load);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%obtain B_ave using lua blockintegral(8,9,5,10)
%testing block(11) for magnet
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

counter_testblock_magnet = 11;

for testblock_magnet_count = 1:counter_testblock_magnet
    %get coordinates of testing blocks
    %steel testing blocks start from 75th row in block list
    x_magnet = block((63 + testblock_magnet_count-1),1);

```

```

y_magnet = block((63 + testblock_magnet_count-1),2);
mo_selectblock(x_magnet,y_magnet);
Bx_block_magnet = mo_blockintegral(8);           %T.m^3
By_block_magnet = mo_blockintegral(9);           %T.m^3
Carea_block_magnet = mo_blockintegral(5);
volume_block_magnet = mo_blockintegral(10);
mo_clearblock();
Bx_av_magnet = Bx_block_magnet / volume_block_magnet;
By_av_magnet = By_block_magnet / volume_block_magnet;
B_av_magnet = sqrt((Bx_av_magnet)^2 ...
    +(By_av_magnet)^2);

    if Bx_av_magnet == 0
        B_av_angel_magnet = 90;
    elseif Bx_av_magnet < 0
        B_av_angel_magnet = ...
            atan(By_av_magnet/Bx_av_magnet)/(2*pi)*360 + 180;
    elseif By_av_magnet >= 0
        B_av_angel_magnet = ...
            atan(By_av_magnet /
Bx_av_magnet)/(2*pi)*360;
    else
        B_av_angel_magnet = ...
            atan(By_av_magnet/Bx_av_magnet)/(2*pi)*360
+ 360;
    end

angle_magnetization_magnet = block((63 + testblock_magnet_count-
1),6);

    B_av_dom_magnet = ...
        Bx_av_magnet *
cos(angle_magnetization_magnet/360*2*(pi))+ ...
        By_av_magnet * sin(angle_magnetization_magnet/360*2*(pi));

%write lineintegral info into testblock_magnet_cell
%format:
%side formed by point_2 and point 1 of testing block

mo_selectpoint(index_point_measuring_magnet(testblock_magnet_count,3),ind
ex_point_measuring_magnet(testblock_magnet_count,4));

mo_selectpoint(index_point_measuring_magnet(testblock_magnet_count,1),ind
ex_point_measuring_magnet(testblock_magnet_count,2));
    line_integral_array(1:2) = mo_lineintegral(0);
    mo_clearcontour();
    %side formed by point_2 and point_3 of testing block

mo_selectpoint(index_point_measuring_magnet(testblock_magnet_count,3),ind
ex_point_measuring_magnet(testblock_magnet_count,4));

mo_selectpoint(index_point_measuring_magnet(testblock_magnet_count,5),ind
ex_point_measuring_magnet(testblock_magnet_count,6));
    line_integral_array(3:4) = mo_lineintegral(0);
    mo_clearcontour();
    %side formed by point_3 and point_4 of testing block

mo_selectpoint(index_point_measuring_magnet(testblock_magnet_count,5),ind
ex_point_measuring_magnet(testblock_magnet_count,6));

```

```

mo_selectpoint(index_point_measuring_magnet(testblock_magnet_count,7),ind
ex_point_measuring_magnet(testblock_magnet_count,8));
    line_integral_array(5:6) = mo_lineintegral(0);
    mo_clearcontour();
    %side formed by point_1 and point_4 of testing block

mo_selectpoint(index_point_measuring_magnet(testblock_magnet_count,1),ind
ex_point_measuring_magnet(testblock_magnet_count,2));

mo_selectpoint(index_point_measuring_magnet(testblock_magnet_count,7),ind
ex_point_measuring_magnet(testblock_magnet_count,8));
    line_integral_array(7:8) = mo_lineintegral(0);
    mo_clearcontour();

    testblock_magnet_cell{testblock_magnet_count}(step_count,:) = ...
        [step_count (step_count-1)*step (step_count-1 +
offset)*step ...
        x_magnet y_magnet Carea_block_magnet Bx_block_magnet
By_block_magnet ...
        volume_block_magnet B_av_magnet B_av_angel_magnet ...
        angle_magnetization_magnet B_av_dom_magnet
line_integral_array];

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%obtain B_ave using lua blockintegral(8,9,5,10)
%testing block(16) for steel
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

counter_testblock_steel = 16;

for testblock_steel_count = 1:counter_testblock_steel

    %get coordinates of testing blocks
    %steel testing blocks start from 75th row in block list
    x_steel = block((75 + testblock_steel_count-1),1);
    y_steel = block((75 + testblock_steel_count-1),2);
    mo_selectblock(x_steel,y_steel);
    Bx_block_steel = mo_blockintegral(8);
    By_block_steel = mo_blockintegral(9);
    Carea_block_steel = mo_blockintegral(5);
    volume_block_steel = mo_blockintegral(10);
    mo_clearblock();
    Bx_av_steel = Bx_block_steel / volume_block_steel;
    By_av_steel = By_block_steel / volume_block_steel;
    B_av_steel = sqrt((Bx_av_steel)^2+ ...
        (By_av_steel)^2);

    if Bx_av_steel == 0
        B_av_angel_steel = 90;
    elseif Bx_av_steel < 0

```



```

        B_av_angel_steel = ...
        atan(By_av_steel/Bx_av_steel)/(2*pi)*360 + 180;
    elseif By_av_steel >= 0
        B_av_angel_steel = ...
        atan(By_av_steel / Bx_av_steel)/(2*pi)*360;
    else
        B_av_angel_steel = ...
        atan(By_av_steel/Bx_av_steel)/(2*pi)*360 +
360;

    end

    %write lineintegral info into testblock_steel_cell
    %format:
    %side formed by point_2 and point 1 of testing block

mo_selectpoint(index_point_measuring_steel(testblock_steel_count,3),index
_point_measuring_steel(testblock_steel_count,4));

mo_selectpoint(index_point_measuring_steel(testblock_steel_count,1),index
_point_measuring_steel(testblock_steel_count,2));
    line_integral_array(1:2) = mo_lineintegral(0);
    mo_clearcontour();
    %side formed by point_2 and point_3 of testing block

mo_selectpoint(index_point_measuring_steel(testblock_steel_count,3),index
_point_measuring_steel(testblock_steel_count,4));

mo_selectpoint(index_point_measuring_steel(testblock_steel_count,5),index
_point_measuring_steel(testblock_steel_count,6));
    line_integral_array(3:4) = mo_lineintegral(0);
    mo_clearcontour();
    %side formed by point_3 and point_4 of testing block

mo_selectpoint(index_point_measuring_steel(testblock_steel_count,5),index
_point_measuring_steel(testblock_steel_count,6));

mo_selectpoint(index_point_measuring_steel(testblock_steel_count,7),index
_point_measuring_steel(testblock_steel_count,8));
    line_integral_array(5:6) = mo_lineintegral(0);
    mo_clearcontour();
    %side formed by point_1 and point_4 of testing block

mo_selectpoint(index_point_measuring_steel(testblock_steel_count,1),index
_point_measuring_steel(testblock_steel_count,2));

mo_selectpoint(index_point_measuring_steel(testblock_steel_count,7),index
_point_measuring_steel(testblock_steel_count,8));
    line_integral_array(7:8) = mo_lineintegral(0);
    mo_clearcontour();

    %      testblock_steel_array(testblock_steel_count, : ) = ...
    %      [x y Carea_block Bx_block By_block volume_block B_av_lua
B_av_angel];
    testblock_steel_cell{testblock_steel_count}(step_count,:) = ...
        [step_count (step_count-1)*step (step_count-1+offset)*step ...
        x_steel y_steel Carea_block_steel Bx_block_steel
By_block_steel ...
        volume_block_steel B_av_steel B_av_angel_steel
line_integral_array];

    end

    %torque
    mo_selectpoint(45.75,0)

```

```

mo_selectpoint(32.35,-32.35)
mo_selectpoint(0,-45.75)
mo_selectpoint(-32.35,-32.35)
mo_selectpoint(-45.75,0)
mo_selectpoint(-32.35,32.35)
mo_selectpoint(0,45.75)
mo_selectpoint(32.35,32.35)
mo_selectpoint(45.75,0)
T = mo_lineintegral(4);

torque_array(step_count,:) = [step_count (step_count-1) (step_count-
1)*step ...
    T(1) T(2)];

mo_clearcontour()
mo_close()

end

%rotationary angle fix for testing block on rotor
%magnet 1-11 and steel 12-16

correction_vector_angle_magnet =
zeros(length(testblock_magnet_cell{1}(:,11)),11);
counter_correction_column_magnet = length(testblock_magnet_cell{1}(:,11));
counter_correction_row_magnet = 11;

for correction_count_column_magnet = 1:counter_correction_row_magnet
    for correction_count_row_magnet = 1:counter_correction_column_magnet
        correction_vector_angle_magnet_element =
testblock_magnet_cell{correction_count_column_magnet}(correction_count_ro
w_magnet,11)...
        +
testblock_magnet_cell{correction_count_column_magnet}(correction_count_ro
w_magnet,2)/4;
        if correction_vector_angle_magnet_element >= 360
            correction_vector_angle_magnet_element =
correction_vector_angle_magnet_element- 360;
        else
            correction_vector_angle_magnet_element =
correction_vector_angle_magnet_element;
        end

correction_vector_angle_magnet(correction_count_row_magnet,correction_cou
nt_column_magnet) ...
        = correction_vector_angle_magnet_element;
    end
end

correction_vector_angle_steel =
zeros(length(testblock_steel_cell{1}(:,11)),16);
counter_correction_column_steel = length(testblock_steel_cell{1}(:,11));
counter_correction_row_steel = 16;
for correction_count_column_steel = 1:11
    correction_vector_angle_steel(:, correction_count_column_steel) =
testblock_steel_cell{correction_count_column_steel}(:,11);
end
for correction_count_column_steel = 12:counter_correction_row_steel

```

```

        for correction_count_row_steel = 1:counter_correction_column_steel
            correction_vector_angle_steel_element =
testblock_steel_cell{correction_count_column_steel}(correction_count_row_
steel,11)...
            +
testblock_steel_cell{correction_count_column_steel}(correction_count_row_
steel,2)/4;
            if correction_vector_angle_steel_element >= 360
                correction_vector_angle_steel_element =
correction_vector_angle_steel_element- 360;
            else
                correction_vector_angle_steel_element =
correction_vector_angle_steel_element;
            end

correction_vector_angle_steel(correction_count_row_steel,correction_count
_column_steel) ...
                = correction_vector_angle_steel_element;
        end
    end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%save the data to data file 'xxx.mat'
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
save('final_1000');

%d = {'x1','y1','A1','x2','y2','A2','x3','y3','A3','Bx','By','B_abs', ...
%     'B_angle','PM_angle','Bx_c','By_c','B.n','area_meshelement'};
%xlswrite('test_aveB.xls',d);
%hold on
%xlswrite('test_aveB.xls', magnet_cell{1}, 1, 'A2');
%hold on
%xlswrite('test_aveB.xls', B_cell{1}, 1, 'J2');

```

### I.13 Lua script output file.

Time (s)	pm_angle (degree)	electrcial tooth angle(degree)	flux tooth 1(Wb)	flux tooth 2(Wb)	flux tooth 3(Wb)
0	0	360	4.91E-06	-1.47E-05	2.38E-05
1	0.9	352.8	1.37E-06	-1.12E-05	2.07E-05
2	1.8	345.6	-2.17E-06	-7.69E-06	1.74E-05
3	2.7	338.4	-5.74E-06	-4.15E-06	1.39E-05
4	3.6	331.2	-9.28E-06	-5.79E-07	1.04E-05
5	4.5	324.0	-1.28E-05	2.99E-06	6.90E-06

### I.14 An example of FEMM model block properties.

	location		material	unused	circuit	Angle	Group	turns
Rotor	1.30	1.50	1	-1	0	0	0	1
	4.20	69.00	2	-1	0	0	0	1
	3.70	76.70	2	-1	0	0	0	1
Air gap	8.10	83.60	1	-1	0	0	0	1
Magnet	81.50	-0.20	5	-1	0	0	1000	1
	80.00	-15.80	5	-1	0	168.75	1000	1
	75.30	-30.70	5	-1	0	337.5	1000	1
	67.90	-45.20	5	-1	0	146.25	1000	1
	57.60	-57.60	5	-1	0	315	1000	1
	-----							

### I.15 Models dimensions.

Model	Tooth width (mm)	Airgap (mm)	Slot area (mm <sup>2</sup> )	Rotor outside radius (mm)	Axial length (mm)	Magnet power rating (W)
16 poles 18 slots	2.7	0.225	32.9	14.825	6.9	20
32 poles 36 slots	9.2	0.225	116.8	14.825	6.9	20
32 poles 96 slots	2.73	0.225	55.6	14.825	6.9	20
32 poles 192 slots	1.36	0.225	27.8	14.825	6.9	20
8 poles 9 slots	10.5	1.5	667.1	45	57.2	3000

## Appendix II

### Code description and demonization check

#### II.1 Description of FEMM and MATLAB usage.

FEMM is a simple and straight forward tool to use for designer and it is efficient tools for solving two dimension (2-D)magnetic and electromagnetic problems using Finite element Analysis (FEA) method. FEA is a popular mathematic method for solving complicated electromagnetic problem, such as machine model development. It created triangle meshes to fill up a large object following particular algorithm and the total flux of a object could be calculated based on the position and vector potential information of the mesh node. Although the algorithm was not within the scale of this research, more and finer meshes did generated in the area where the flux density was considered as high. An example magnet filled with mesh using FEA method was shown in Fig. II.1 and the meshes was dense at the corner of the magnet. Theoretically, it would be more accurate if more meshes created in one area, but the drawback was the volume of computation was also increasing with the number of meshes created. Therefore, different program based on FEA method were employed in modern design work, to analyze on computer instead of establishing manually on paper [6], which accelerated the speed of analysis process. Also, convenient ways to obtain measurement from the model was provided by software. The detail of using such software, i.e. FEMM in this research, was introduced within this chapter. For a FEMM model, to limit the area of mesh generating, a boundary normally would be defined outer the model and at the same time, this outer boundary shall not constrain the electromagnetic field. An example of such a boundary was shown in Fig. II.2.

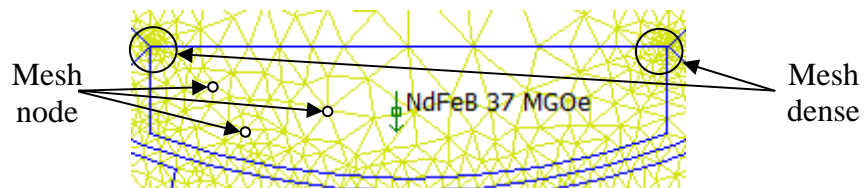


Fig. II.1 Magnet model using FEA method.

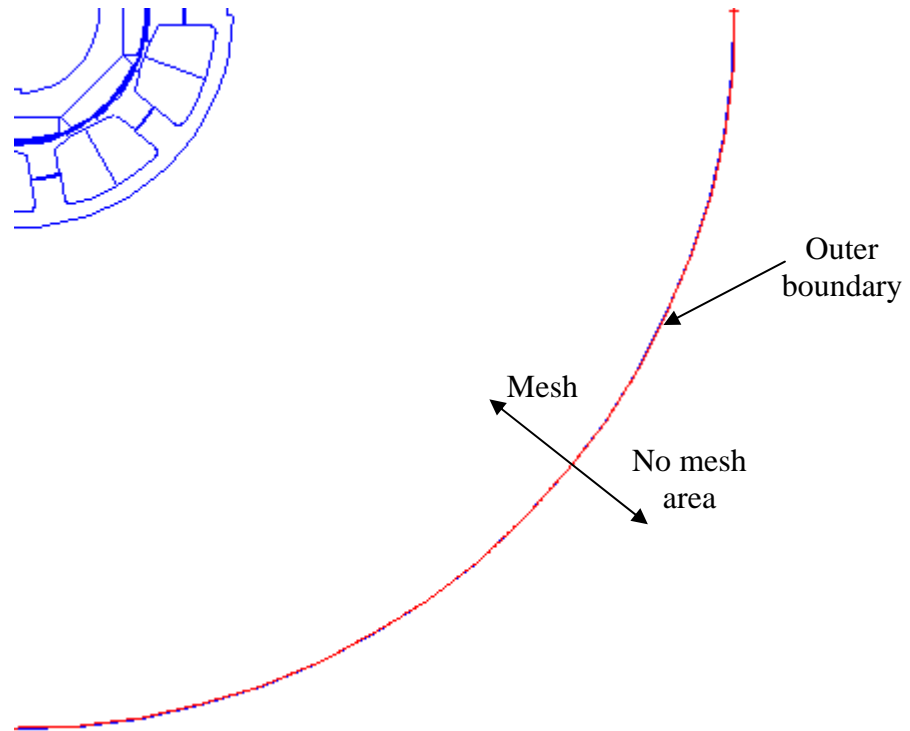


Fig. II.2 An example of boundary of a FEMM model.

There were two type of files FEMM would generate for a model: the model file of which file name was ending with '.FEM' and the result file was ending with '.ans'. The result file was the static field solution for the model file and it could be created by analyzing the model file using 'Analyze' command in pull-down menu in FEMM. It is import to define objects precisely in FEMM model file, because any piece of information would affect the result, i.e. the sequence of selecting during measurement would affect the sign of result. It would cause error when analyze the model, if any mistake had been made in model file. Also the sequence of creating objects in FEMM would affect the location of parameters and information in fundamental list of model file. To access the model file list, the files need to be opened with a text editor, i.e. Notepad which was a text editor embedded in Windows operating system. Both model and result files both could be viewed and edited using Notepad. Notepad provided a convenient way to access the variables in FEMM model file and its solution file and this method was the fundamental concept for later analysis. The model file contains all the parameter and objects information. The information could be displayed visually in a window if opened with FEMM itself and described numerically as a list. It was equivalent to modify parameter in the list to changing via a FEMM window and for some particular tasks, i.e. modifying current in circuit, changing the parameter in the list was more convenient. The selection of modifying method was situational for the model file. Similar to the model file, the results file not only contain the information of a model, but also the electromagnetic solution using FEA

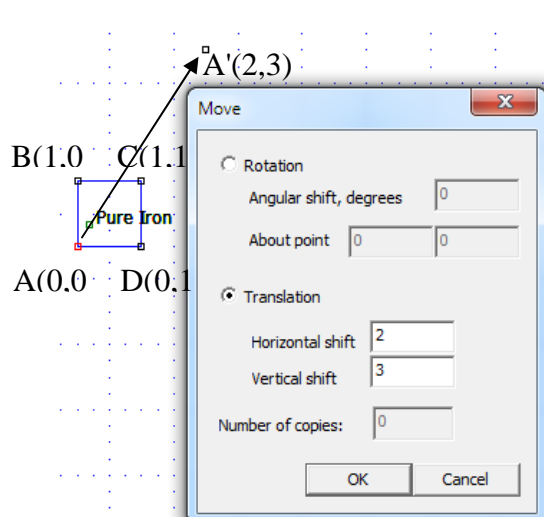
method, i.e. mesh information. Therefore, the size of result file was normally larger than the model file. The result file was the key to this research and excessive usage of it would be discussed in later chapter when iron loss calculation was involved. There were several sections in FEMM result file, if viewed by Notepad. The most valuable part of was the last section of the list: 'Solution', which was used for demagnetization check later in this chapter.

In FEMM, the fundamental structure of every object was node, which was a point with precise coordinate. Cartesian coordinate system was used in FEMM and any points or nodes could be determined as one set of unique coordinate:  $(x, y)$  according to the centre point  $(0, 0)$ . This coordinate system provided tight trace of any model parts which were formed by points and lines between points in FEMM, and also the fundamental control of the program. Based on node, line and block could be created by connecting multiple nodes, i.e. straight line between two nodes and assign a material block in a closed area. The most convenient way to create an object was to operate in FEMM window. In Fig. II.3, a square pure Iron block was created by using four nodes. With known position, in Fig. II.3(a), four nodes: ABCD were placed on an 'Magnetics Problem' sheet while FEMM was in 'operate on nodes' mode. To create a line between nodes, the mode of FEMM should be changed to 'operate on segments' and then select starting node of the line and ending node in sequence. If a curve was required, the mode should be set as 'operate on arc segments'. By repeat the selection process, i.e.  $AB$ ,  $BC$ ,  $CD$  and  $DA$ , a block was formed by four lines and was ready to be assigned as material block. Similar to created a node, defining a block was to place a node within the block when 'operate on block labels' mode. The type of material and other definitions of the block could be modified after the properties window had been opened in FEMM. There were many material in FEMM library and the coefficients of them were accessible. A custom material could be also defined in FEMM by typing in its characteristic data and coefficients.

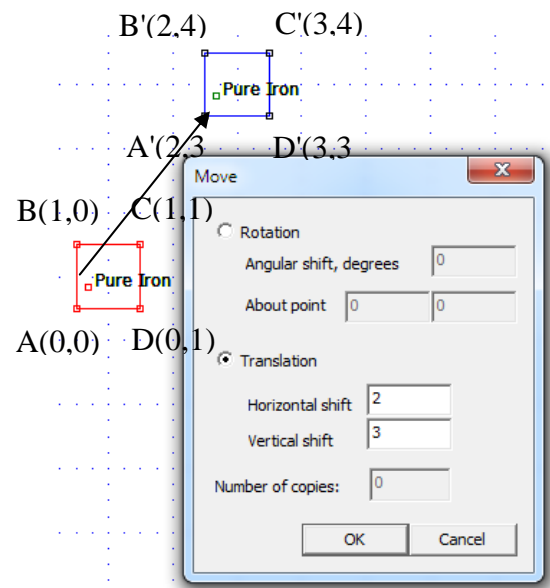
For demonstration purpose, the block in Fig. II.3 was assigned as pure iron. While editing the FEMM mode, the most important thing was selecting the right mode, i.e. nodes mode or segments mode, as FEMM would not recognise which kind of object was operating on. As a result, the simplicity of FEMM potentially brought more controllability in future programming. Once the object was created, the modification could not be accomplished by simply 'drag and drop' in FEMM, again, the simplicity of FEMM. There three ways to modify the object in FEMM, location to be precise in this case. The first way was achieve the modification by bring dialog window via 'edit' pull-down menu and an example was shown in Fig. II.3(a). Node A was moved to A' by shifting 2mm horizontally and 3mm

vertically. Another example of moving an object was shown in Fig. II.3(b) and in this case, the iron block ABCD was shifted 2mm horizontally and 3mm vertically to new location, A'B'C'D', using 'move dialog'. To create grouped a object, all the components of this object should be set as same group number in property dialog, including nodes, lines and block labels. The objects with same group number would be recognized as one group in FEMM and any modification would be applied to each components, i.e. location in Fig. II.3(b). This grouping facility of FEMM was very useful for simulate complicated model which required movement.

With the grouped iron block, rotating feature was also tested, shown in Fig. II.3(c) and it was key to simulate a machine motion. It was noticed that the positive position for angular position variation was anticlockwise in FEMM and this would affect rotational part of simulation program design. Apart from editing in FEMM window, the second way was to find the node information in fundamental list of model file and change the correlated coordinate to desired value. This method could be more difficult especially when the model become complicated and indentifying the right information field of the node could be time consuming. Alternatively, also the most applicable route to edit an objects in FEMM, in both simple and complex cases, was using functions in Lua script. With functions of Lua script, objects would be edited by FEMM if the command was appropriated and such a characteristic would benefit the simulation code development since direct operation on model file was not required. The detail of Lua script and its verification would be introduced in this section.

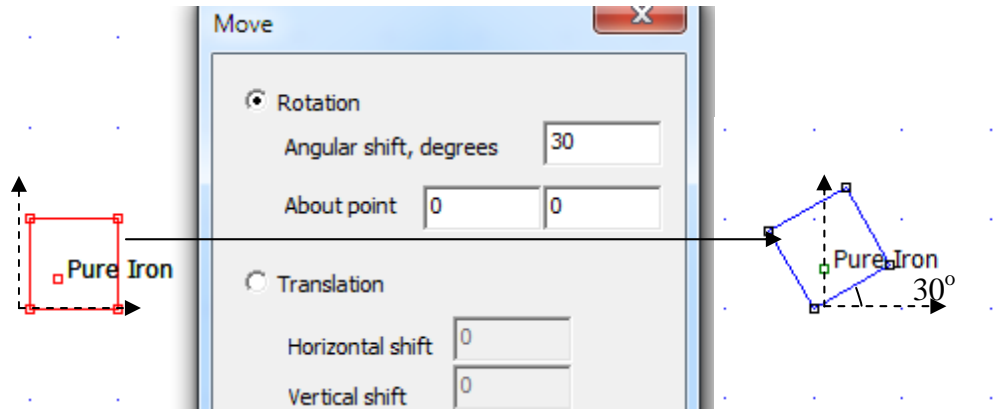


(a)



(b)

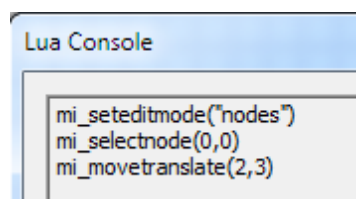




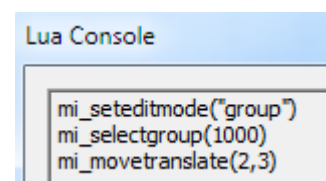
(c)

Fig. II.3 Example of moving individual point (a), moving grouped object (b) and rotate a grouped object (c).

To verify FEMM functions, the model modification in Fig. II.3 were repeated by using Lua script. The FEMM functions employed to accomplish the task were shown in Fig. II.4 and they were tested in 'Lua console'. The model moving and rotating task was as simple as manual operation once the function was selected correctly and script was debugged, shown in Fig. II.4 (a), (b) and (c). For creating a group object, it was more complicated than manual operation, since coordinates of every part of the iron block were required and also cross type selecting was not supported in FEMM. A example of group defining was shown in Fig. II.4 (d) and the objects were one node, one line and the block label. It would take more than ten lines of Lua script to achieve the grouping task for the iron block, which would be accomplished with few click in FEMM window. The manual operation was preferred in grouping task over the Lua script.



(a)



(b)

```

Lua Console
mi_seteditmode("group")
mi_selectgroup(1000)
mi_moverotate(0,0,30)

```

(c)

```

Lua Console
mi_seteditmode("nodes")
mi_selectnode(0,0)
mi_setgroup(1000)

mi_seteditmode("segments")
mi_selectsegment(0,0.5)
mi_setgroup(1000)

mi_seteditmode("blocks")
mi_selectlabel(0.32,0.33)
mi_setgroup(1000)

```

(d)

Fig. II.4 Lua script approach for basic model modification: moving a node (a), moving a block (b), rotating a block (c) and set group for objects (d) in FEMM.

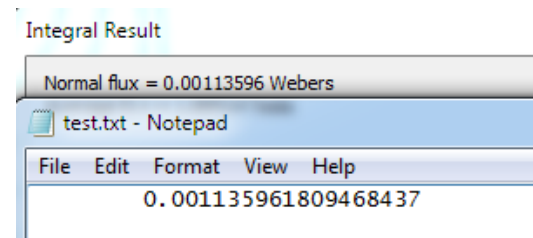
With verified model modifying functions, a simple Lua script was developed for verifying the data extracting and data recording, shown in Fig. II.5 (a), following similar iterative process shown in Fig. 3.2. This piece of program was designed to extract flux information using line integral. Using the model file used in Fig. 3.3(a), the first three lines of the coding was for file operating and was followed by line selecting and flux information extracting. Data recording to a text file was the end of testing program. The Lua script file was ended with '.lua' and it could be edited with Notepad. To execute a Lua script, 'Open Lua Script' should be chosen and the proper script was also needed to selected correctly via dialog window in FEMM. On default, the model file would be searched locally when a Lua script executed. The file operating commands in testing script was 'open', 'mi\_analyze' and 'mi\_loadsolution' and the verification was straight forward since the execution was easily observed: loading model file, analyzing model and loading result file respectively. The process of data recording was accomplished by using writing commands 'openfile', 'write' and 'closefile' and the value recorded was the same as manual operating, shown in Fig. II.6 (b).

```

open("chpt3_3.6_test.FEM")
mi_analyze(1)
mi_loadsolution()
mo_selectpoint(58.5,5.25)
mo_selectpoint(58.5,-5.25)
flux_link_tooth1,x1 = mo_lineintegral(0);
handle = openfile("test.txt","a")
    write(handle,"      \t",flux_link_tooth1,"
","\n")
    closefile(handle)

```

(a)



(b)

Fig. II.6 A testing Lua script (a) for flux linkage measurement and its result (b) comparing with manual obtain in FEMM.

After successfully recorded flux linkage for one static solution, more functions were embedded into the script: model rotating, to simulate a electric machine operating. In terms of programming, a loop of single static analyzing process would be ideal for the task, but testing combined functions in Lua script was essential for next step in software development. Hence, a piece of testing Lua script was written, shown in Fig. II.7 (a), with all necessary functions which were including rotating, object selecting, integration and data recoding. In the first section of the Lua script, the same model file which was used in Fig. 3.3 (a) was loaded and it was modified using rotating function. The result of rotating and line integral was shown in Fig. II.7 (b). The flux linkage was different from the model before rotating, of which was expected. The unit of flux linkage was Weber. The detail of script execution was described below. As mentioned earlier, the selecting mode was important for FEMM and therefore, '*mo\_seteditmode*' and '*mi\_selectgroup*' were used to set the type of selection and particular group for next processing. The type selection command was not essential in this script, because the mode of selection would be force to be group by '*mi\_selectgroup*' command and any pervious selections would be cleared. The double dash line would be used in front of any comments or unwanted commands and the line with it would not be executed.

Rotating command, '*mi\_moverotate*', was used to rotate an object about a centre point with certain degree, *15degree* in this case. The positive direction of rotating was anticlockwise and hence the angle in script was negative. Once the command '*mi\_analyze*' was called, a magnetic or electromagnetic field solution would be solved for the model by FEMM using FEA method and the solution would be saved as a result file. All the information required for investigating the machine static electromagnetic properties were contained in solution file, including physical geometry, circuit properties, flux and vector potential of mesh nodes. After FEMM finishing solving the problem, the solution file was required to be loaded before any further analysis established and the loading command '*mi\_loadsolution()*' would only load the solution file created by last '*mi\_analyze*' command, i.e. in Fig II.7 (a) '*mi\_loadsolution()*' was used following the analyze command. Another selecting command '*mi\_selectpoint*' was used to select the single point in model file or select points in series to form a line or a circle. During two-point selection, FEMM would search for the existing line between two points first and would created temperate

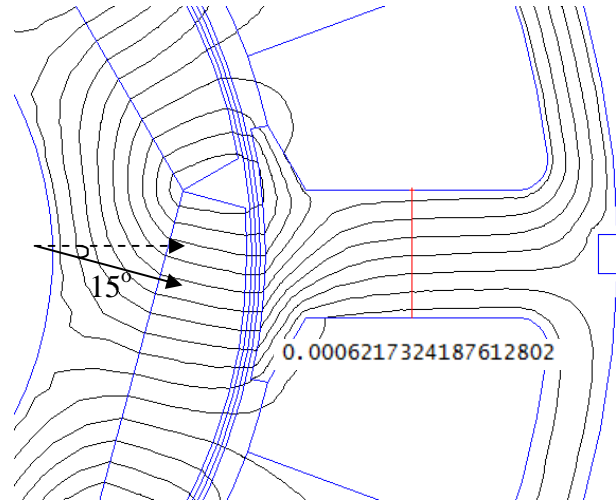
straight measuring line if there were not and any data extraction command executed would be solely about this straight line. In some case, multiple points were created along the curve to force the FEMM selecting the desired segments, i.e. when selected a circle for torque data extracting. In multiple-point selection, the sequence was very important and elbowed lines would be created if any mistake made. To be reminded that, the sequence of point selection would also affect the sign of calculated results. The sequence of selection was maintained through all tests, i.e. clockwise, in order to obtain consistent sign and direction of readings.

Depending on the geometry of examining objects, different data extracting commands were employed in this research, which were including '*mo\_getpointvalues*' for points, '*mo\_lineintegral*' for lines, '*mo\_blockintegral*' for blocks and '*mo\_getcircuitproperties*' for circuits. For each of these commands, there was corresponding mode to suit particular purpose of tasks, i.e. in Fig. II.7 (a), after selecting a line, line integral was called and the type of data extraction was '*0*', which meant FEMM would calculate total flux linkage and flux density along the selected line and return these two values as results. In order to storing the returned results, two variables were created respectively and in this case: '*flux\_link\_tooth1*' and '*x1*'. If only one variable was created, the Lua script would be stopped and return an error, because in Lua script, one variable could not handle two values. The line integral command could also be used for calculating the torque along a contour or obtaining area of the cross section by setting the type of data extraction. Those functions would be mentioned in later tasks. After data extracting, the final part of the program would be data recording command for this purpose was '*openfile*'. When '*openfile*' was called, FEMM would open a file of which name was set in parameter or would create a new file as named, when the mode of this I/O command was set to '"*a*"'. An ending flag for file operation '*closefile*' was always followed after data recorded. For simple data storing purpose, text file format was select to save the data extracted from pervious steps of program. Similarly to native Lua script I/O command, Lua script ran under FEMM shared same parameters in writing a file: '*/t*' to separate the data with a horizontal tab and '*/n*' to start a new line for another row to record. Within the output bracket, '*write()*', any names or variables needed to double quoted if they were desired to appear in output text file and value of variables did not need to.

```

open("chpt3_3.6_test.FEM")
mi_seteditmode("group")
mi_selectgroup(1000)
--Rotating model
mi_moverotate(0,0,-15)
mi_analyze(1)
mi_loadsolution()
--flux linkage of 1st tooth
mo_selectpoint(58.5,5.25)
mo_selectpoint(58.5,-5.25)
flux_link_tooth1,x1= mo_lineintegral(0);
handle = openfile("test.txt","a")
    write(handle,"
\t",flux_link_tooth1," ", "\n")
    closefile(handle)end

```



(a)

(b)

Fig. II.7 Example of Lua script: Flux linkage measurement.

The commands were covered so far could only achieve one static solution at a time. To simulate a machine model, solutions of an electric cycle with multiple steps were required. The loop control commands in Lua script could be used to set multiple step solutions in one program. An example code was shown in Fig. II.8 (a), using ‘for loop’ to rotate machine mode for  $m$  steps. The parameter and loop setup was the same as native Lua script and the parameters were initial value, end value and interval in sequence. Once a ‘for loop’ called, the program would execute the command lines between the next line of ‘for loop’ command and the last end flag which was indicated the end program. Combined with the rotate and data extracting commands, data required for analyzing machine under particular condition, i.e. certain current density and magnet design in this thesis, could be collected by one program and instead of multiple executing manually. More detailed example program would be introduced when machine model analysis was involved. The design of loop was shown in Fig. II.8 (b).

```

m=n+10
for time= 1,m, 1 do
.....
end

```

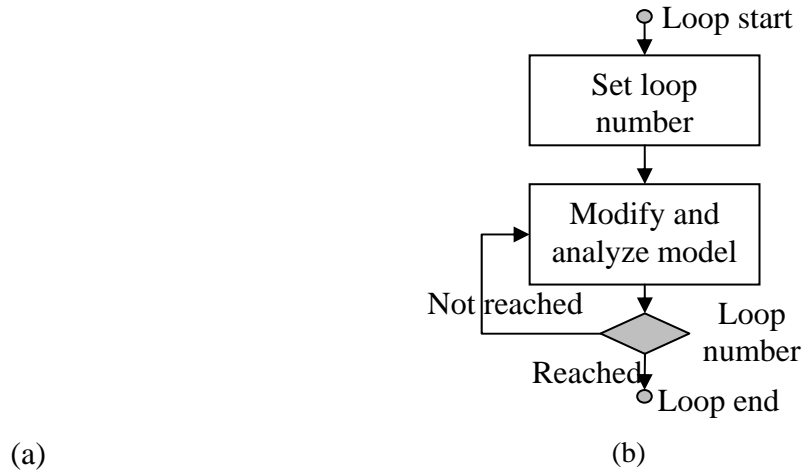


Fig. II.8 For loop in Lua script (a) and brief script design for multi-step machine simulation (b).

After implemented the loop into Lua script, the rotor of the model was rotated by a angle,  $\theta$  for each step, which was the rotation a electric machine simulation required. A testing Lua script was developed, shown in Appendix I.1. The flux linkages in the teeth of the machine model were recorded while the rotor rotating  $\theta$  degree anticlockwise for each step. For this test, *5 degree* step was used, in order to cover the model position in Fig. II.7 (b). The flux linkage for first five step were shown in Table II.1 and the value was match with pervious test. For this test, there was no particular time frame and the time was counted in step. Hence the angular velocity was *5 degree* per step and the angular speed could be derived if the time consumed for each step was given. For example, assuming there was *1 second* between steps, the angular velocity would be *5 degree* per second, i.e. *0.8 rpm*. To justified the flux linkage in table, there was less than *0.01%* difference between the data obtained by using manual operation and Lua script and it was negligible. The difference was caused by the small variation in mesh, which would be regenerated for each solution .

Time (step)	Mechanical angle (degree)	Flux linkage (Wb)
1	0	0.001135992030
2	5	0.001099602662
3	10	0.000954845563
4	15	0.000621733457
5	20	0.000203696568
6	25	-0.000203674177

Table II.2 The output table of Lua script with loop.

Although FEMM and Lua script were capable for accomplishing simulation task in pervious Chapter, MATLAB was employed later in the research to manage the simulation process, due to its advantage in array manipulating, loop control and mathematical calculation. Referring to Fig. 3.4, the type of files MATLAB program correlated with were shown. By executing an MATLAB program, i.e. an M-file, with all necessary command within it, the program would load corresponding FEMM model file, analyze the model for entire electric cycle in specified step set and store the results in both MATLAB data format and spreadsheet. The program could be either separated to accomplish a single task, i.e. obtain cogging torque, or integrated to analyze one model for multiple prospers. A full electric cycle is 360 degree for a machine model and it would be different according to its pole pair, which was the magnet design. To be able to cooperative with MATLAB program, FEMM version 4.2 above was required and noted that earlier versions of FEMM did not contain mathematical toolbox '*Octave*', in which case '*Octave*' needed to be install separately. It was a significant improvement to introduce the mathematical interface, since MATLAB provided a much convenient way to store data, modify variables and flow control, i.e. array and loop.

Lua script functions were required to simulate the machine model and it manage FEMM model directly, i.e. assigning material, changing circuit property and modifying model. Lua script itself worked as a fundamental program language in this case, which could be executed by using FEMM specified MATLAB functions via '*Octave*'. The FEMM specified MATLAB functions were usually contained in a folder under FEMM installation directory on hard drive and it should be included into MATLAB library prior of any FEMM related function was called. Although all the FEMM functions were used in MATLAB had already been convert to MATLAB functions, it was convenient to call those function as lua command or FEMM function, because function names were the same both in MATLAB and Lua script. With more familiar MATLAB loop control, full electric cycle simulation could be setup with less hassle. Although MATLAB plus Lua commands were enough for most of the tasks, for some certain task, i.e. demagnetization check, more fundamental method required to find mesh information for each element in FEMM model: opening FEMM results file with Notepad. FEMM model was consisted with number, values in both model file (.femm) and results file (.ans). All the information the simulation requires were in the result file.

The topology was that treated FEMM result file as text file, which was readable for MATLAB, and converted all the information into arrays using MATLAB. Then the information required could be found easily in array format using search function in

MATLAB. Examples of processing results files would be introduced in when demagnetization involved. Another advantage of using MATLAB was that it was a powerful mathematical tool and also provided graphical representation for analysis, which means data could be calculated and display the results within MATLAB. With all the advantage MATLAB mentioned above, it was possible to create set of program for certain multiple tasks, which combines model operation, data extraction, model analysis, graphic representation and data recording. This ultimate version of program would be introduced at the end of this chapter and the development of the program would be represented in each section in this chapter. For different tasks, as the model or parameters changing, MATLAB code did require modification to achieve different objectives. It was also very important to keep model test files in different folders and cataloguing different revision of the program as it developed. The folder index was used to prevent accidental file rewriting during each task and file name structure was used to keep track progress in programming. As shown in Fig. 3.4, sometimes numerical data would be stored in excel format for the purpose of easy accessing and better graphical representation.

According to FEMM manual, there would be another approach to collect data of flux linkage apart from line integral: using circuit property command. The line integral was operated on a cross section of the tooth and it would provide more detail information in one particular part of the model. The circuit in FEMM model was defined as blocks and the tooth of machine mode was clamped between two winding blocks. Hence the overall flux information could be more than the data extracted by line integral, due to the coverage. The difference in data extracting between two approaches was closely monitored in analysis and would be discussed further when it became significant. For a three phase FEMM model, both input and output circuits for circuit A, B and C were required to pre-defined in library in order to become assignable and the copper blocks between machine tooth were represented the winding around tooth. There were input and output for winding in practice and the same for FEMM model: To assign a circuit, circuit name needed to be selected from circuit library and for this task other parameters like number of turns and group number were kept as default, '1' and '0' respectively. In FEMM model, the winding direction was defined by the direction of current flow in series circuit, i.e. positive and negative current would represent winding in and out respectively. Although winding direction could be defined by positive or negative number of turns in circuit library, method of current sign defining winding was clear and simple and was kept consistently all through the simulation. An example of predefined '*circuit A*' was shown in Fig. 3.5. In this



example, the copper block was assigned as circuit 'A-' and bottom one was 'A+'. This meant copper wires went out from the paper in circuit 'A+' area and went into the paper in circuit 'A-' in practice and this was the concept to wind a tooth with copper wire in FEMM model. Also the slot was separated into two parts which indicate there were two sets of windings in one slot of machine model. The full concentrated 3-phase winding for this task was shown in Fig. II.4 and this winding schematic would be used for each magnet design. The current in circuits were zero as default and property of a circuit could be modified by MATLAB function '*mi\_modifycircprop()*', i.e. circuit, number of turn, value, depend on the task. There were six circuits for 3-phase machine model, which including current flow in and out for each phase.

During software development, parameters of circuit were assigned every time a program code was called to prevent parameter assigning by mistake from pervious task. For an open-circuit test, the current in all circuits should be zero amperes, so it was import to set the current to zero in MATLAB code when FEMM model was loaded. For a non-open-circuit test, although a fixed value of current would be sufficient for simulation setup, it was convenient to change the current assignment into expressions which were consisted with peak current and current phase information. Hence, for difference tasks, the circuit property control was simplified into two parameters, peak current and phase angle, and depending on the phase number and type of winding, the current expression could to be modified to be competent for further tasks easily. The current direction could be represented adding minus sign to the current when the current expression was kept the same.

Although FEMM functions were verified for machine simulation, to establish open circuit test using MATLAB, few verifications were still required before MABTLAB codes could be developed further. There were two main objectives for the verification:

- if Lua script correlated MATLAB function replace FEMM functions in Lua script;
- if machine model be rotated by using MATLAB.

The code development was divided into two steps: first step was to program a MATLAB code which could achieve static solution analysis and extract data; the next step was focused on adding loop control into tested program to accomplish one cycle machine simulation, i.e. rotating model while analyzing and data extracting. MATLAB code development was kept as simple as possible and the flow chart for static solution analysis program was shown in Fig. II.9 (a) and the loop control embedded was shown in Fig. II.10 (a). The output of MATLAB code could be justified when it compared with the output of

Lua script. The data recording was achieved using array in MATLAB while the simulation was running, which was the significant advantage over Lua script. Using program with loop as a instant, unlike data recording in Lua script, i.e. it was not possible to manipulate recorded data within Lua script, data could be stored in array and remain accessible at any time until array was over written or cleared with MATLAB. Also to note that, any model related process were still accomplish by using FEMM function, i.e. model modifying, model analyzing and data extracting.

Two flux linkage measurement program was developed according to the flow charts: signal static solution and multi-step simulation, shown in Fig. II.9 (b) and Fig. II.10 (b) respectively, which were corresponding to the Lua script verification in pervious section. In order to execute Lua script (FEMM functions) in MATLAB, a set of corresponding MATLAB functions were added to MATLAB library at beginning of any program in which FEMM function was embedded. Using the signal static program as example, the commands used for adding the path were '*addpath()*' and '*savepath*', which was part of program initialization and the purpose was to make FEMM function readable and functional in MATLAB. To verify FEMM functions called in MATLAB via Lua scripts toolbox, functions which were including model modifying, model analyzing data extracting were tested by executing line by line in 'Command Window' of MATLAB, of which was similar to Lua console in FEMM. By comparing the output and behaviour of FEMM manual operation, function validation could be justified. To prevent any effect by pervious executed program, MATLAB code was always start with a program initialization, i.e. clearing temporary data and close any program windows. The MATLAB code development was also followed the iterative process. To execute a Lua script, there were several operations was required manually, i.e. opening FEMM program, Lua script selecting. With MATLAB, the Lua script was integrated in MATLAB code and the manual operation in was not required and instead, it could be accomplished using MATLAB code: '*openfemm*' and '*opendocument()*'. Before analyzing the model, it was important to create a duplicate copy of original model file and set the new file as executing target, because it helped to preserve original model file from modification by FEMM. The file operating command was the same as Lua script, i.e. '*mi\_saveas()*' and as a result of this file operation, the original model was maintain the same through the research and model file organization was simplified. The reason for changing the target of manipulating was that model file would be updated and rewrote with any modification by FEMM when analyzing function was executed. Variables related to program would be assigned before analyzing, including current control parameters, i.e. peak current was set to zero ampere since this was open-

circuit test and there should be no supplied current in machine winding. Necessary modification on machine model was also required forehead of analyzing, i.e. circuit property and initial starting position.

The analyzing command was '*mi\_analyze()*' and when it executed, FEMM would solve electromagnetic field using FEA method at its position. Before extracting data, the solution file should be loaded first with '*mi\_loadsolution*' command, which was same as open results file manually. Without solution loading, MABLAB program would manipulate on model file and not be able to establish data extraction process, because there not sufficient information in model file to proceed, hence an error would interrupt program process. Similarly to Lua script, the data extracting commands, '*mo\_lineintegral()*', were tested by calculating the flux linkage and cogging torque. Comparing with results of manual data extraction from solution file, data obtained i.e. flux density and cogging torque, using MATLAB function executed in command window were the same, hence FEMM related functions could be verified and would be employed in further development confidently. Data recording also developed to save present work space, including variable, parameters and desired data, because it was very useful when review was required for data and made variables accessible to other program.

The importance of verifying MATLAB functions, it provided ground or confident for using MATLAB code instead of Lua script and also ensured the expected behaviours for each step a FEMM function was executed. An example MATLAB code was developed, shown in Fig. II.9 (b), for extracting flux information in one static solution and to cross check the output with Lua script verification in Fig II.7 (b). The objective of this piece of code was that it was able to rotate model by *15 degree* clockwise and to extract flux information for solution after model analyzing. The flux data were obtained after executing the MATLAB code and the value matched the output using Lua script, shown in Table II.3 (a).

The development next would be integrated a MATLAB loop into static solution analyzing program, of which could rotate the machine model for certain angular distance. To simulate a machine rotation, the rotor would maintain its centre position for rotating, so the rotation command '*mi\_moverotate*' from FEMM was employed. The base point of the rotation and the rotating angle could be setup within this command and the direction of rotation would be clockwise if the rotating angle was negative. For convenient, clockwise direction was selected as positive direct for rotor rotating. The sequence of rotating the rotor was that select edit mode, select object, rotate object. As mentioned in static analyzing, the manipulating target was changed to a copy of original model and this file

would be used for further modification. Model file would be rotated with same mechanical angle for every loop count and when same rotation command was exsiccated in second loop, the rotation would be built on the previous step, i.e. model rotated twice the mechanical angle comparing to original position. This angle increment was matched with 'for loop' counter and machine model could be rotated step by step along with the loop counter to achieve machine rotating simulation. For the sequence of code executing, the maximum value of loop counter and angular displacement were needed to be defined before entering loop process. The loop flow chart was shown in Fig. II.10 (a).

The most important and most useful part of the program was data recoding. To accomplish the recoding task, at least two arrays would be employed in a program: one universal array and one local array. The local array was a concept relative to universal and the variables stored inside would be only activated and updated under certain condition, i.e. within a loop. The local array was worked as a temporary storage for active data of variables in one loop and when next loop executed, the data from pervious loop would be rewritten or replaced. In contrast, the universal array was defined outside the loop and hence the loop execution was no effect to the data in which it stored. Data from each loop could be recorded without any loss if the local array could dump data to universal array before next loop. The flow chart for such a process was shown in Fig. II.11 (a). With the increment loop number and associated with row number or column number, the best solution to store data in local array was using one row or column for each loop, which restrict the size of local array to one row or one column. Using a one row local array as example, in loop one, the data would be stored in first row of universal array and in second row in loop two. A diagram to demonstrate this dumping process and the structure of universal array and local array were shown in Fig. II.11 (b). A benefit for associating the loop number with location was that it provided tight track of data recorded and it helped to simplify the process of data manipulating in later analysis, since the variable was fixed to particular column. In machine simulation point of view, the data could be collected step by step for a whole cycle and model could be analyzed after simulation.

To be able to store data for all loops, the size of the universal array was preset prior the loop execution depending on the number of loops or steps and also the number of variables required to store in local array. The number of row of the global array was same as maximum loop counter and the column was same as local array. Within each loop, data like torque and tooth flux density were calculated using '*mo\_lineintegral()*' command and were slot into local array with consistent column number. At the end of each loop, an expression was added to update corresponding row of global data array and data would be

fed into global array line by line until last loop was executed. With data extract from results file, the global array was filled at the end of the loop with total row number of maximum value of loop counter. The global array in workspace of MATLAB would be saved as standalone data file for further analysis along with other data arrays, i.e. torque array and flux array, at the end of MATLAB program. During data extracting, proper measuring lines was required to be selected in order to obtain data desired, i.e. a line across tooth for flux information and circle in airgap for torque. It was import to clear contour selection in the event of multiple object selecting. By implementing rotating function, data recording strategy and loop control, the multi-step simulation MATLAB program was developed and the first six lines of the data recoded in universal array was shown in Table. II.3 (b). The flux information were also matched with data extracted by using Lua script.

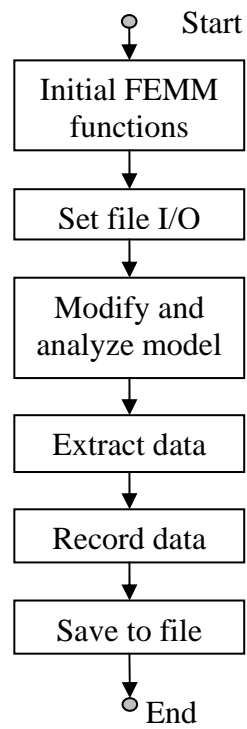
```
close all;
clear all;

addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
opendocument('chpt3_36_test.FEM');
mi_saveas('chpt3_14a.fem');

n = -15;

mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,n);

mi_analyze(1);
mi_loadsolution()
```



(a)

```

mo_selectpoint(58.5,5.25)
mo_selectpoint(58.5,-5.25)

flux_link_tooth1=
mo_lineintegral(0);
mo_close()

flux_link_tooth1(1);
save('chpt_314a');

```

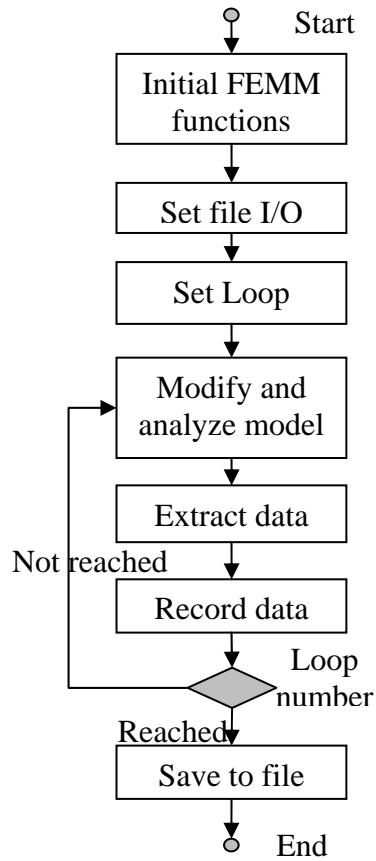
(b)

Fig. II.9 MATLAB program for flux measurement in single static solution: Flow chart (a) and code (b).

```

close all;
clear all;

```



(a)

```

addpath('c:\\femm42\\mfiles');
savepath;
openfemm;
opendocument('chpt3_36_test.FEM');
mi_saveas('chpt3_14a.fem');
pm_angle = 5;
n = 72;
step = 360/n;
m = 10;

mi_seteditmode('group');
mi_selectgroup(1000);
mi_moverotate(0,0,pm_angle);
mi_clearselected();

flux_array = zeros(m,3);
for time= 1:m
    Mechanical_angle= pm_angle+step;
    mi_seteditmode('group');
    mi_selectgroup(1000);
    mi_moverotate(0,0,step);
    mi_clearselected();

    mi_analyze(1);
    mi_loadsolution();
    mo_selectpoint(58.5,5.25)
    mo_selectpoint(58.5,-5.25)

    flux_link_tooth1=
    mo_lineintegral(0);

    flux_array(time,:) = [time
    Mechanical_angle
    flux_link_tooth1(1)];
    mi_clearselected();
    mo_close();
end
save('chpt_314b');
  
```

(b)

Fig. II.10 MATLAB program for flux measurement in multi-step simulation: Flow chart

(a) and code (b).

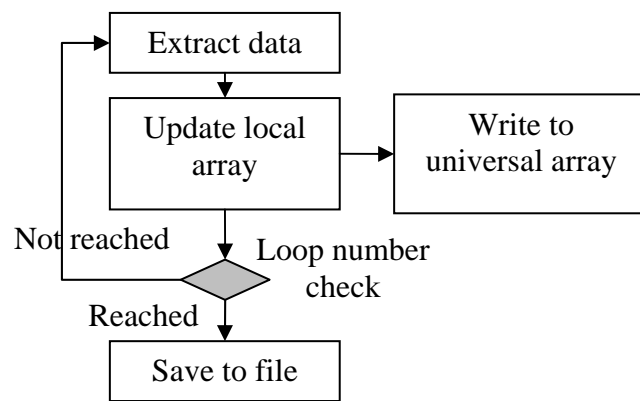
Mechanical angle (degree)	Flux linkage (Wb)
15	0.000621732419

(a)

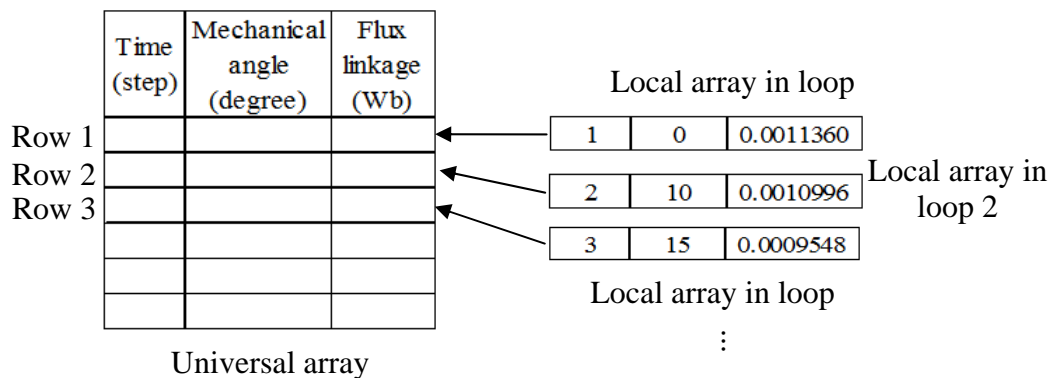
Time (step)	Mechanical angle (degree)	Flux linkage (Wb)
1	0	0.001135992029
2	5	0.001099602662
3	10	0.000954845563
4	15	0.000621733457
5	20	0.000203696568
6	25	-0.000203674177
⋮	⋮	⋮

(b)

Table. II.3 Data extracted using testing MATLAB code: single static solution (a) and multi-step simulation (b).



(a)

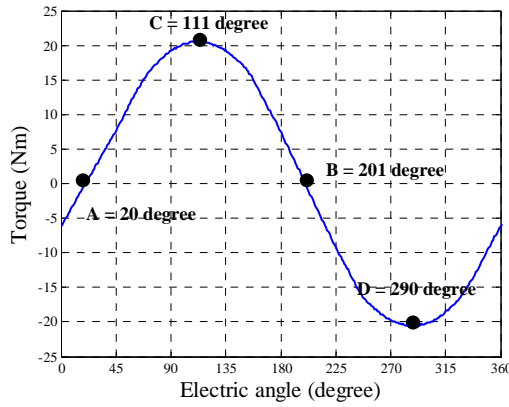


(b)

Fig. II.11 The design of data recoding array: the flow chart (a) and a diagram of data dumping (b).

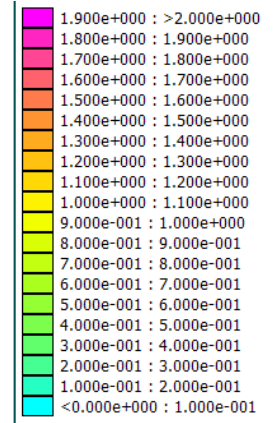


## II.2 static solutions and demagnetization check.

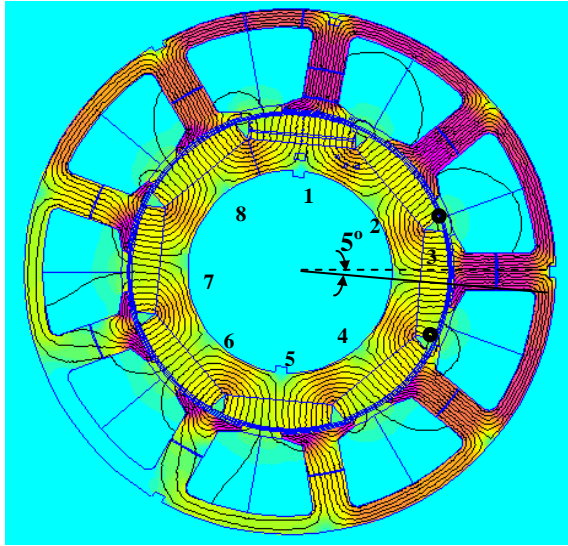


(a) Demagnetization checking position.

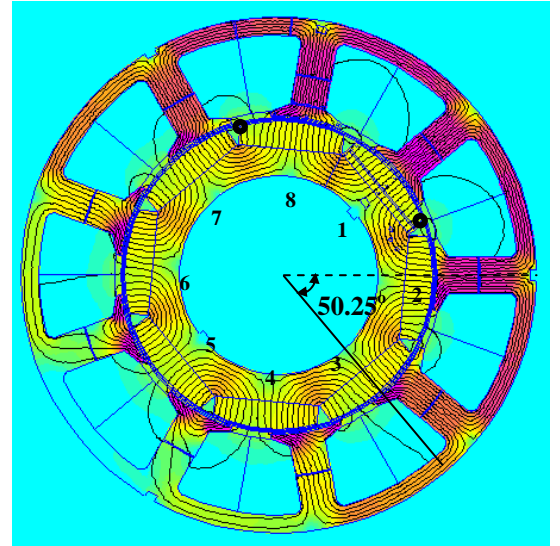
Density Plot:  
|B|, Tesla



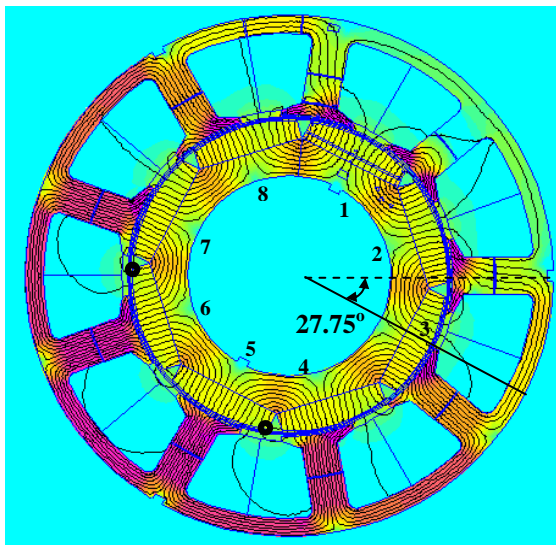
(b) Legend of field solution.



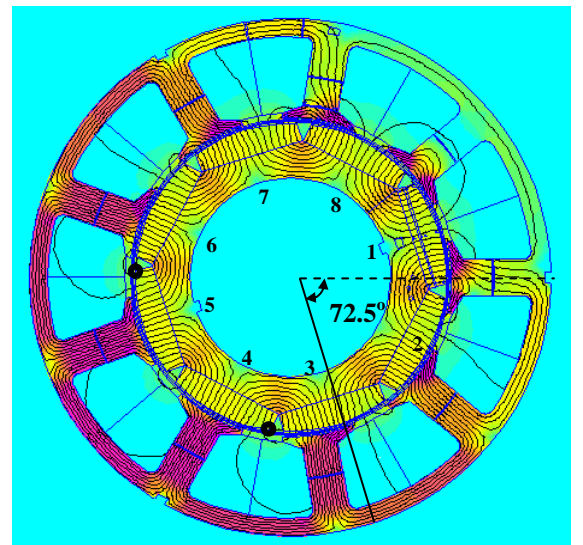
(c) FEA field solution for position A.



(d) FEA field solution for position B.

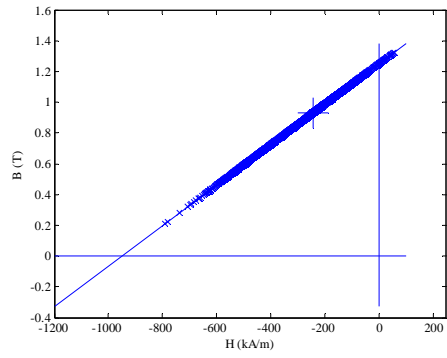


(e) FEA field solution for position C.

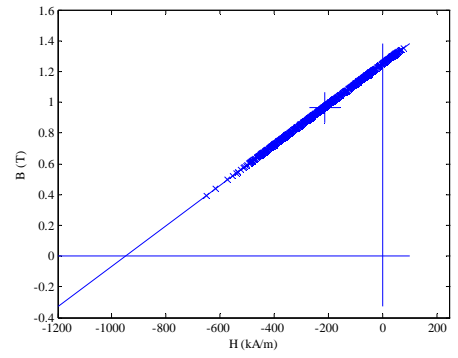


(f) FEA field solution for position D.

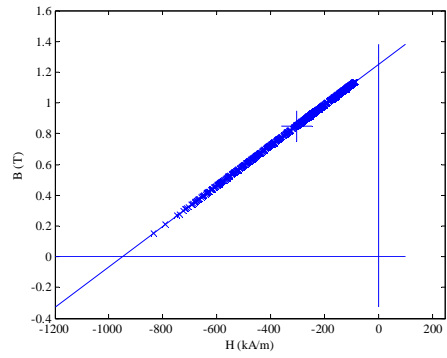
Fig. II.12 FEA field solutions of bread loaf design for each position when peak phase current of 3-phase winding was 1000A-t, with possible demagnetized meshes circled.



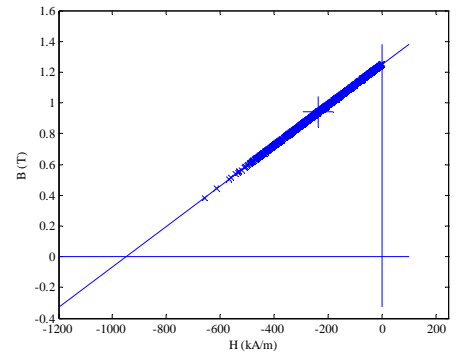
(a) Magnet\_1



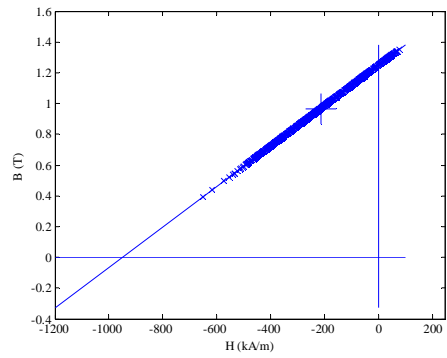
(b) Magnet\_5



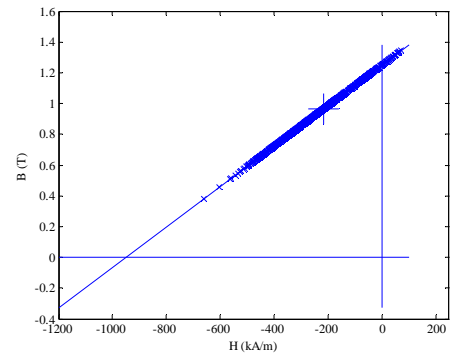
(c) Magnet\_2



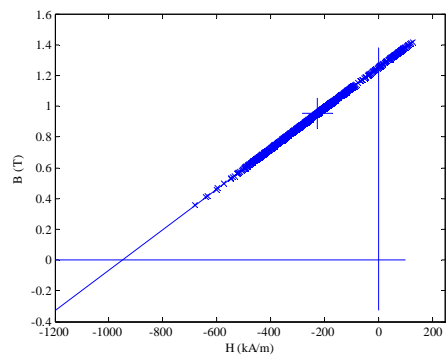
(d) Magnet\_6



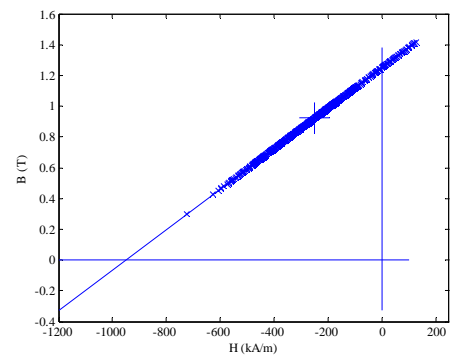
(e) Magnet\_3



(f) Magnet\_7

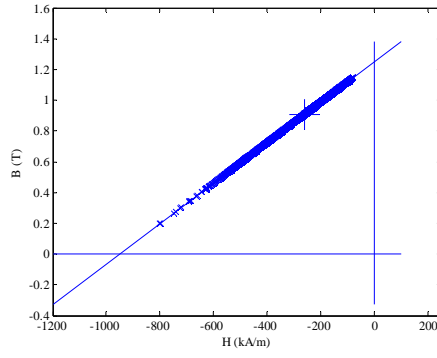


(h) Magnet\_4

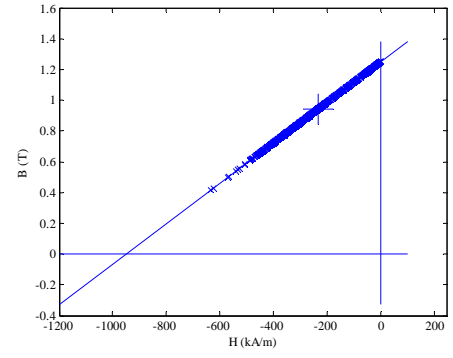


(g) Magnet\_8

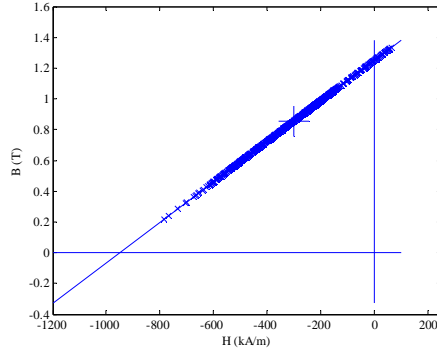
Fig. II.13 Working points plots for bread loaf magnet designed 3-phase model with 1000A-t peak phase current at position A.



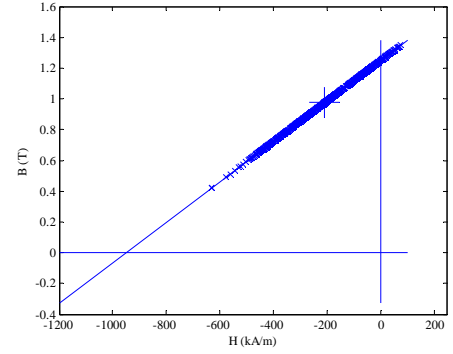
(a) Magnet\_1



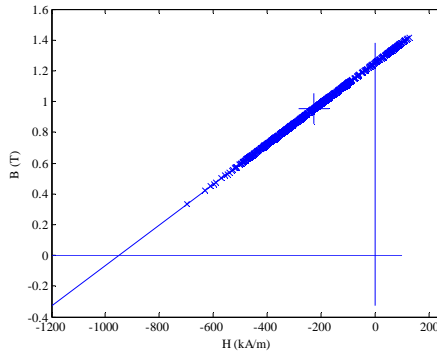
(b) Magnet\_5



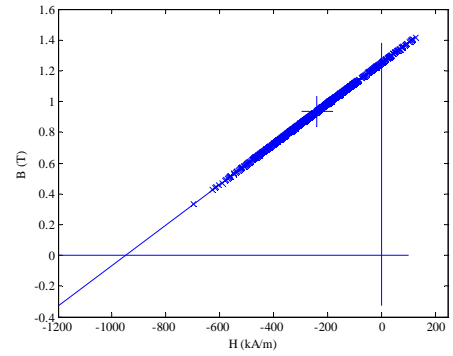
(c) Magnet\_2



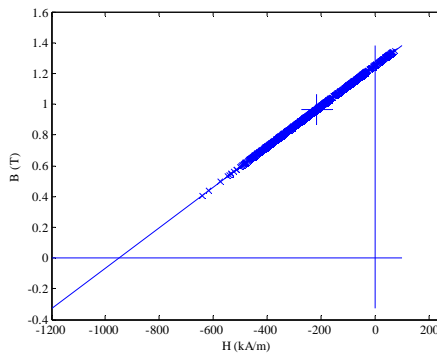
(d) Magnet\_6



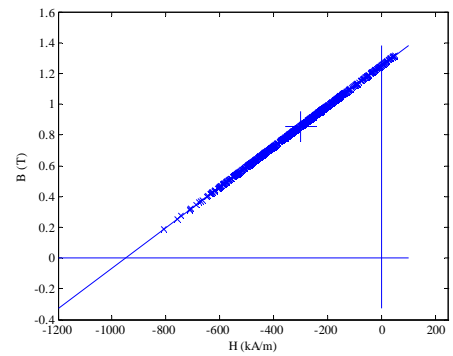
(e) Magnet\_3



(f) Magnet\_7

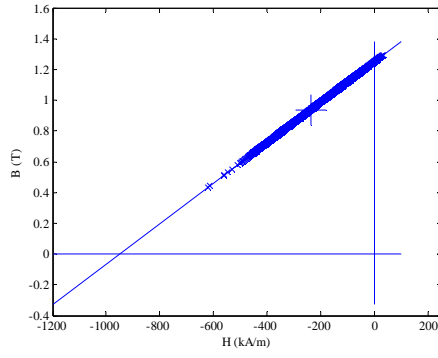


(g) Magnet\_4

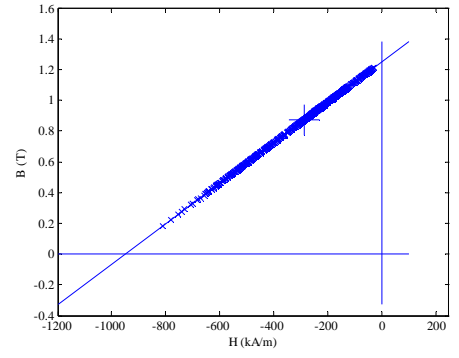


(h) Magnet\_8

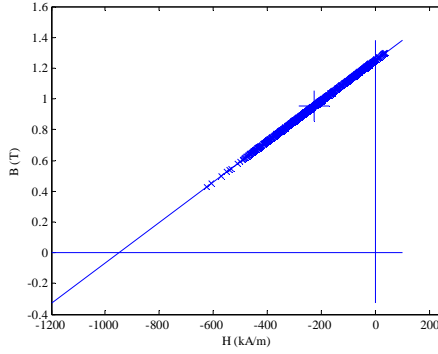
Fig. II.14 Working points plots for bread loaf magnet designed 3-phase model with 1000A-t peak phase current at position B.



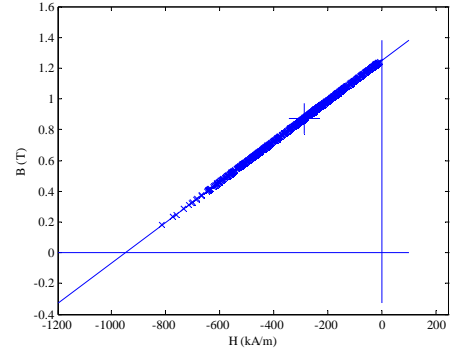
(a) Magnet\_1



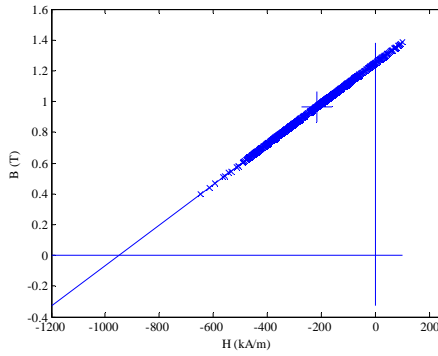
(b) Magnet\_5



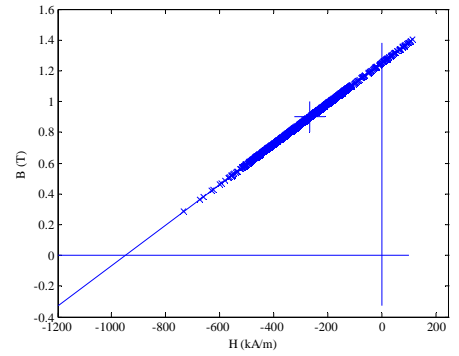
(c) Magnet\_2



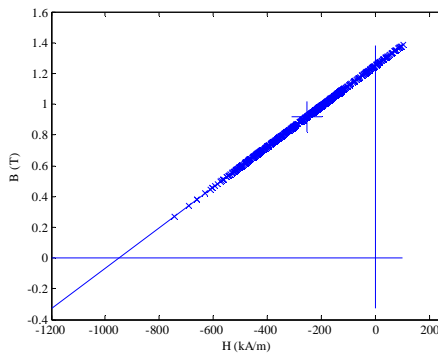
(d) Magnet\_6



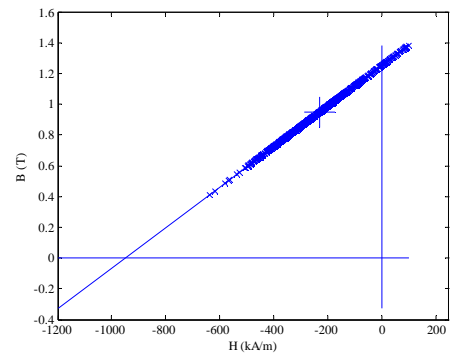
(e) Magnet\_3



(f) Magnet\_7

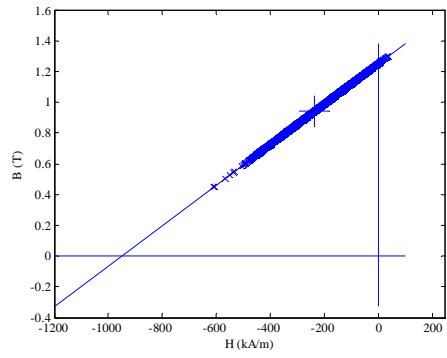


(g) Magnet\_4

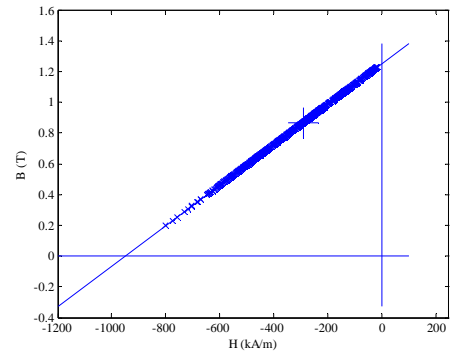


(h) Magnet\_8

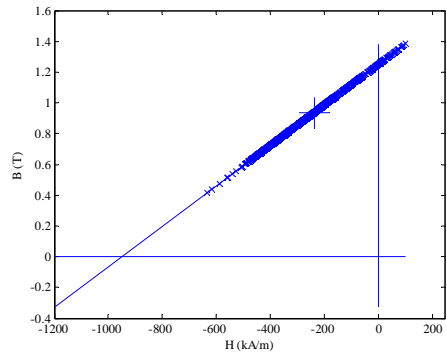
Fig. II.15 Working points plots for bread loaf magnet designed 3-phase model with 1000A-t peak phase current at position C.



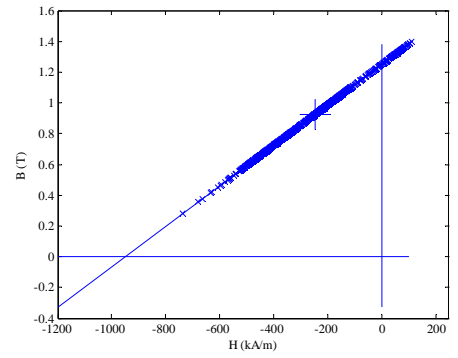
(a) Magnet\_1



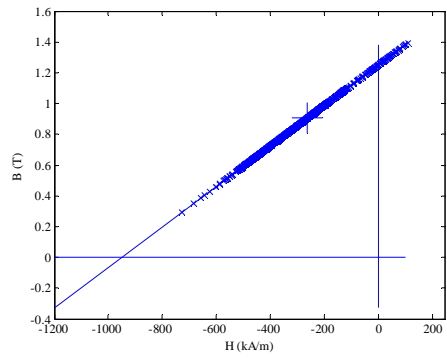
(b) Magnet\_5



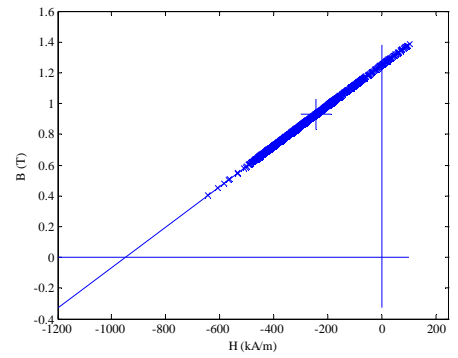
(c) Magnet\_2



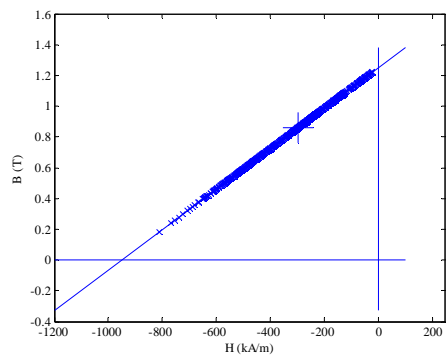
(d) Magnet\_6



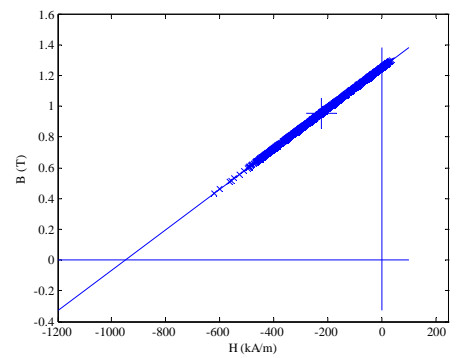
(e) Magnet\_3



(f) Magnet\_7

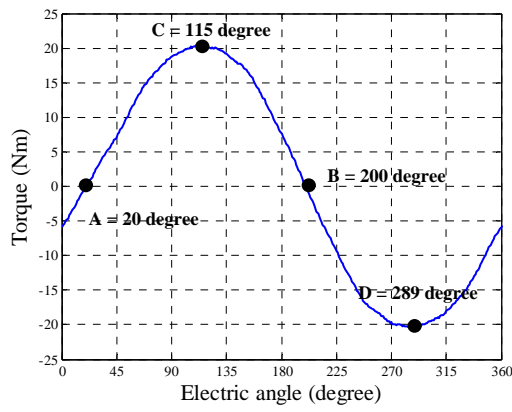


(g) Magnet\_4

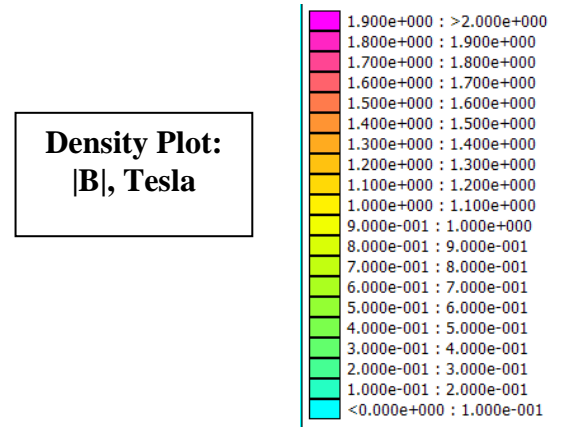


(h) Magnet\_8

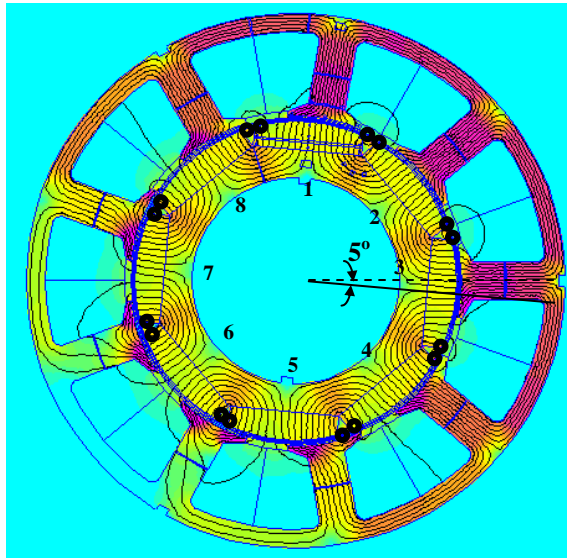
Fig. II.16 Working points plots for bread loaf magnet designed 3-phase model with 1000A-t peak phase current at position D.



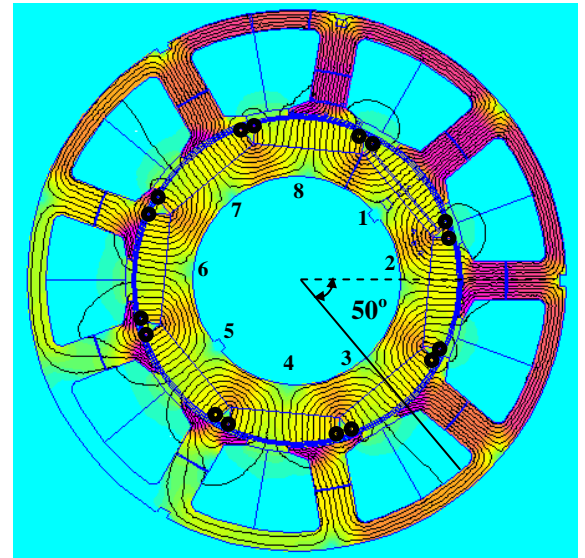
(a) Demagnetization checking position.



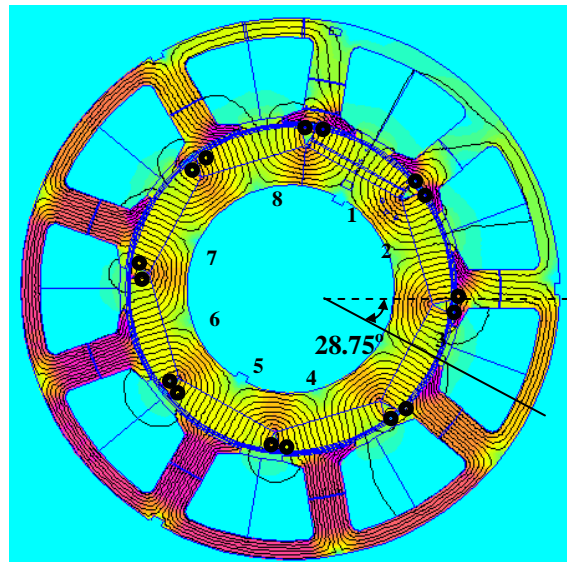
(b) Legend of field solution.



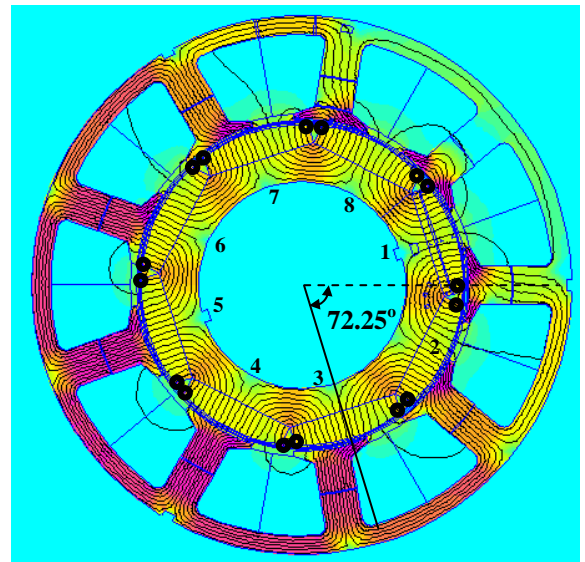
(c) FEA field solution for position A.



(d) FEA field solution for position B.

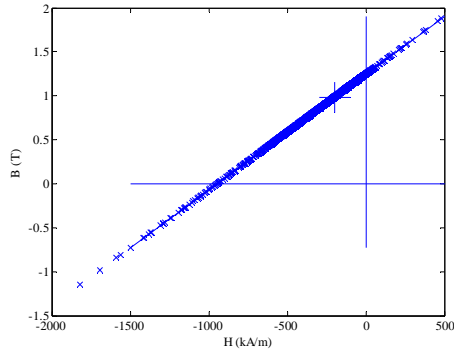


(e) FEA field solution for position C.

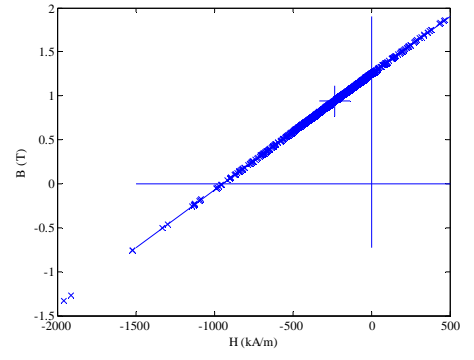


(f) FEA field solution for position D.

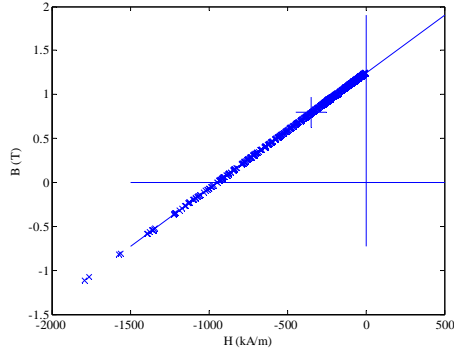
Fig. II.17 FEA field solutions of embedded bread loaf design when peak phase current of 3-phase winding was 1000A-t, with possible demagnetized meshes circled.



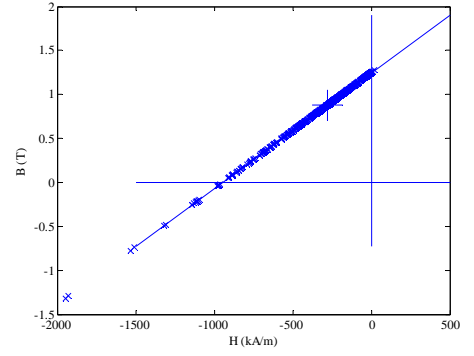
(a) Magnet\_1



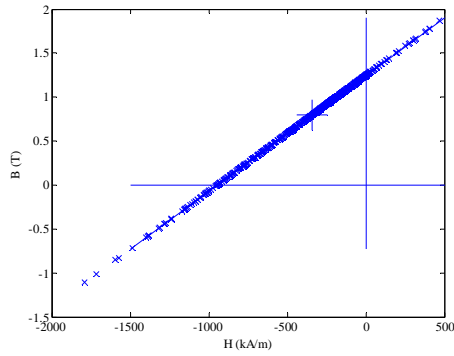
(b) Magnet\_5



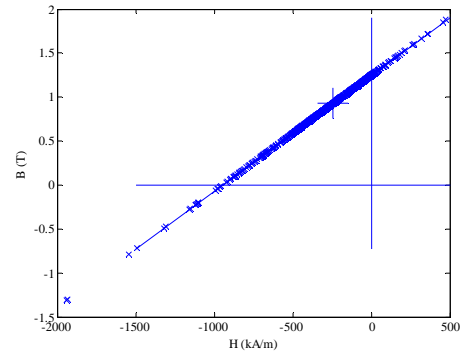
(c) Magnet\_2



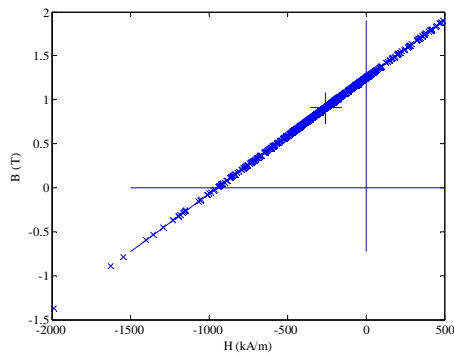
(d) Magnet\_6



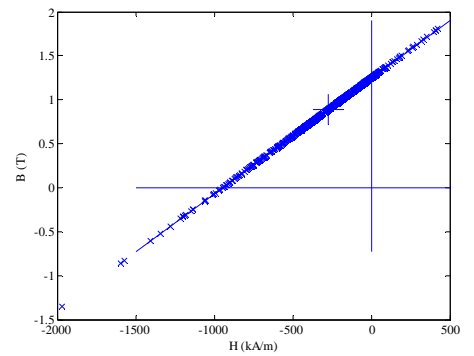
(e) Magnet\_3



(f) Magnet\_7

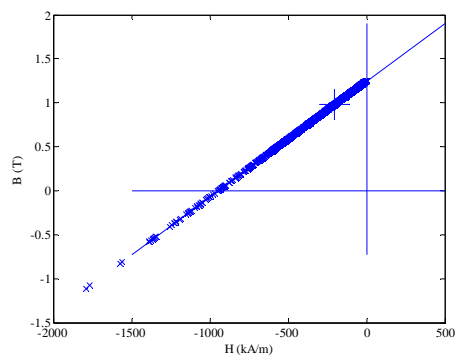


(h) Magnet\_4

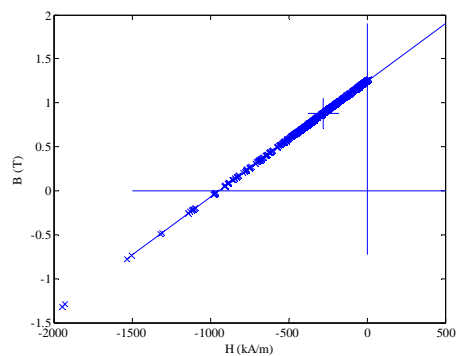


(g) Magnet\_8

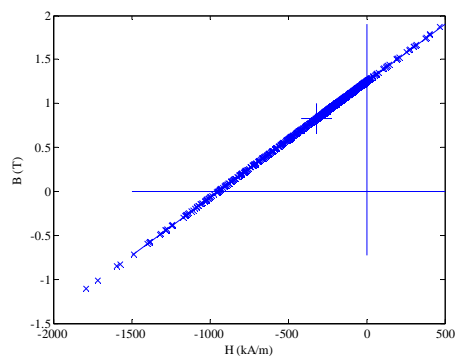
Fig. II.18 Working points plots for embedded bread loaf magnet designed 3-phase model with 1000A-t peak phase current at position A.



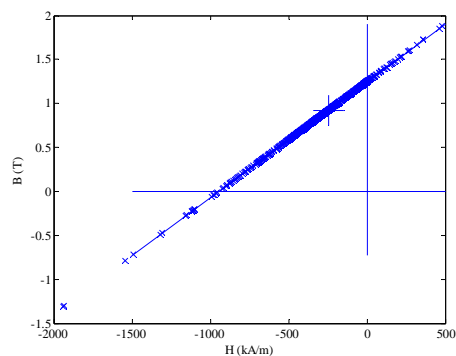
(a) Magnet\_1



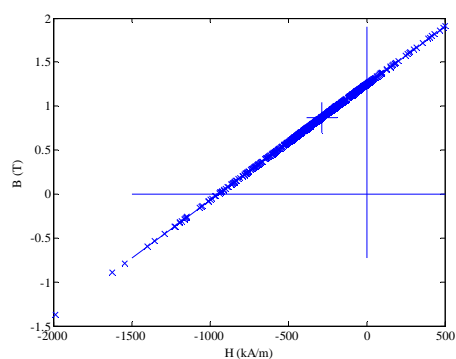
(b) Magnet\_5



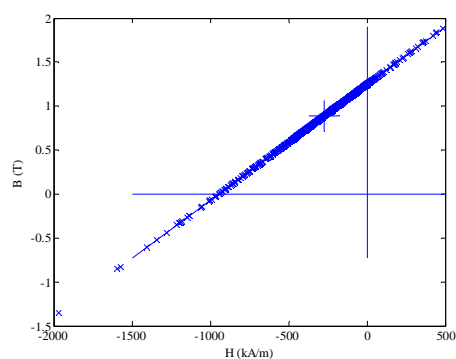
(c) Magnet\_2



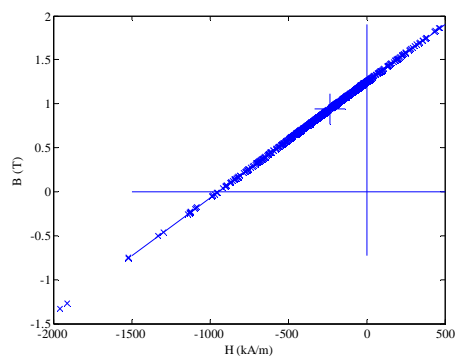
(d) Magnet\_6



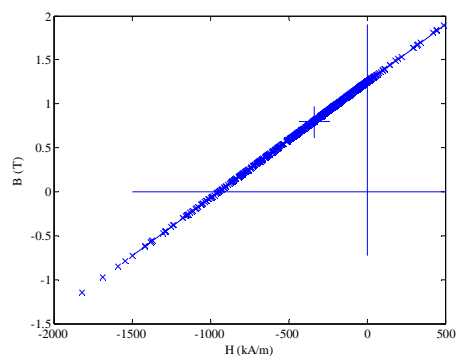
(e) Magnet\_3



(f) Magnet\_7



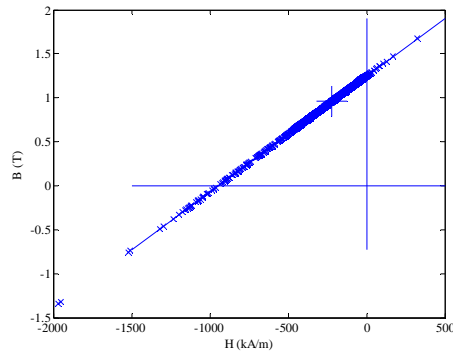
(g) Magnet\_4



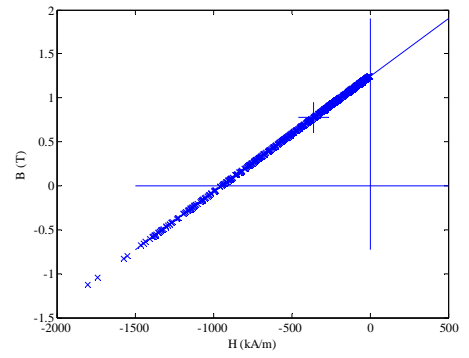
(h) Magnet\_8

Fig. II.19 Working points plots for embedded bread loaf designed 3-phase model with 1000A-t peak phase current at position B.

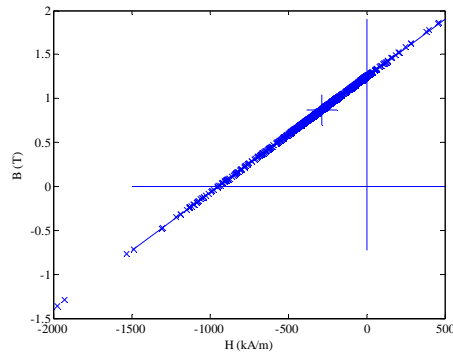




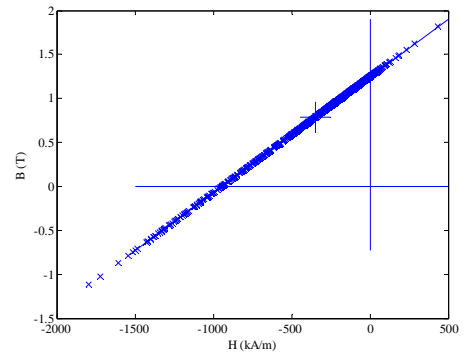
(a) Magnet\_1



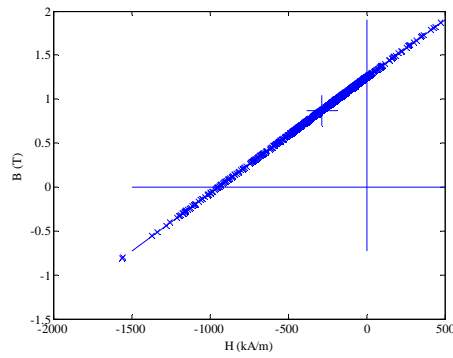
(b) Magnet\_5



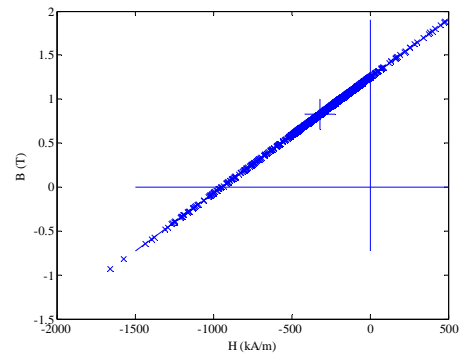
(c) Magnet\_2



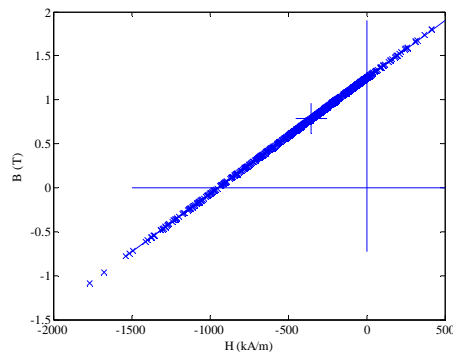
(d) Magnet\_6



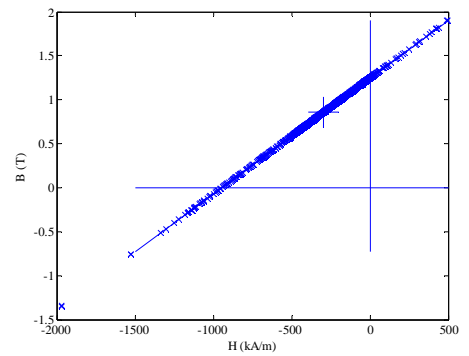
(e) Magnet\_3



(f) Magnet\_7

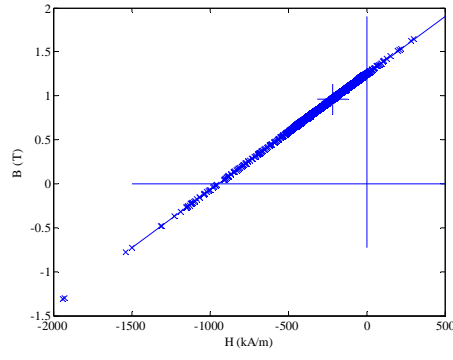


(g) Magnet\_4

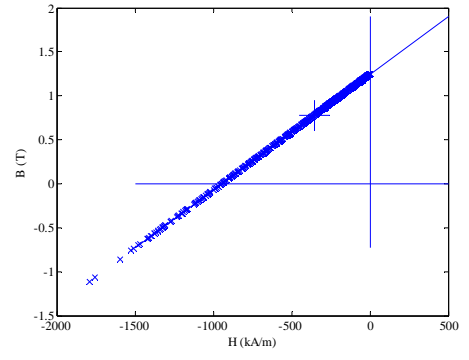


(h) Magnet\_8

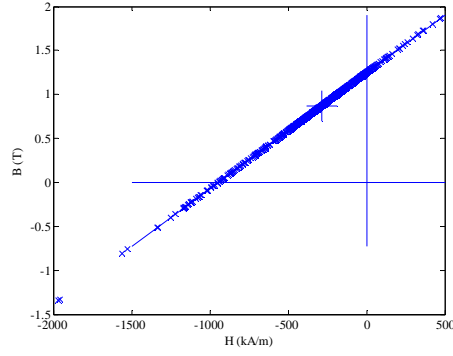
Fig. II.20 Working points plots for embedded bread loaf designed 3-phase model with 1000A-t peak phase current at position C.



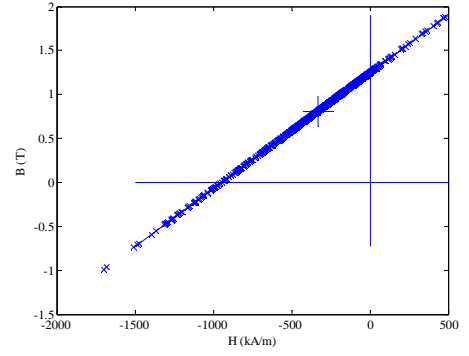
(a) Magnet\_1



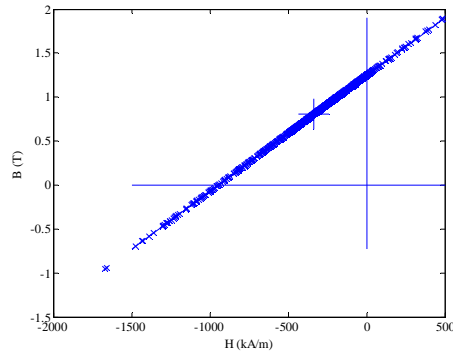
(b) Magnet\_5



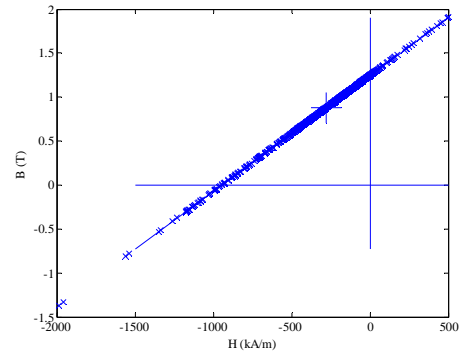
(c) Magnet\_2



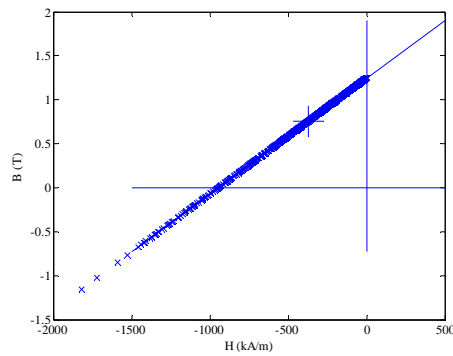
(d) Magnet\_6



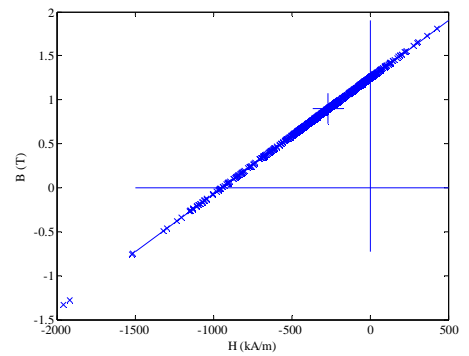
(e) Magnet\_3



(f) Magnet\_7

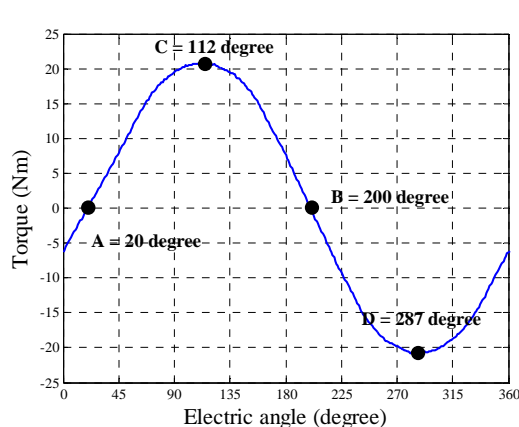


(g) Magnet\_4

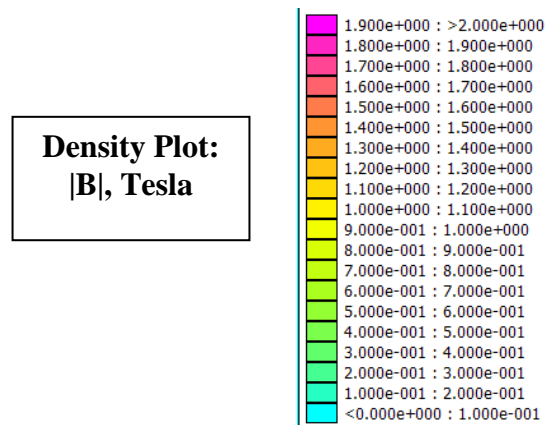


(h) Magnet\_8

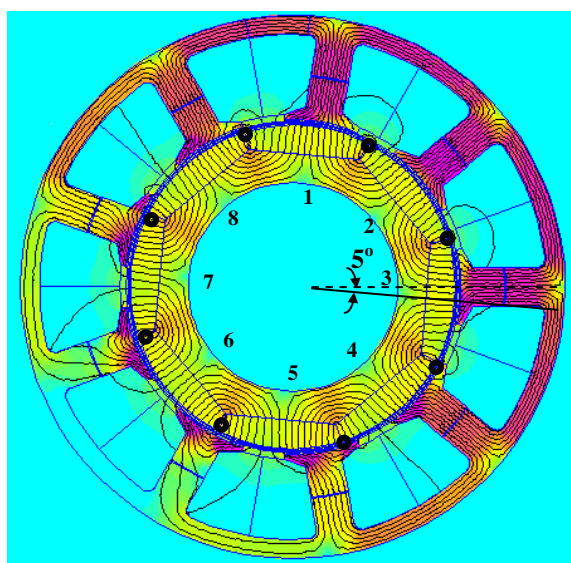
Fig. II.21 Working points plots for embedded bread loaf designed 3-phase model with 1000A-t peak phase current at position D.



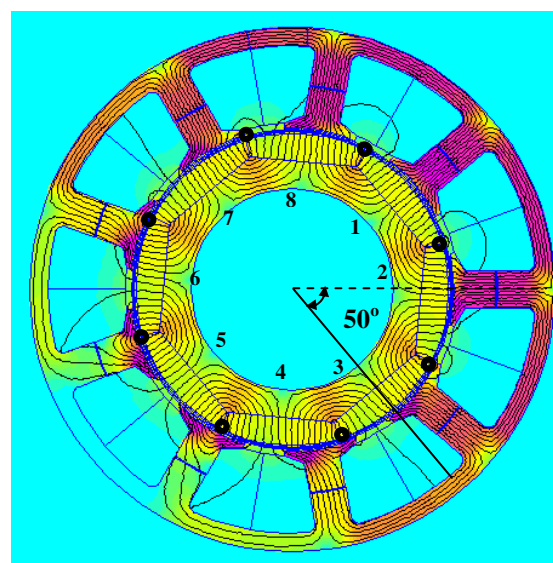
(a) Demagnetization checking position.



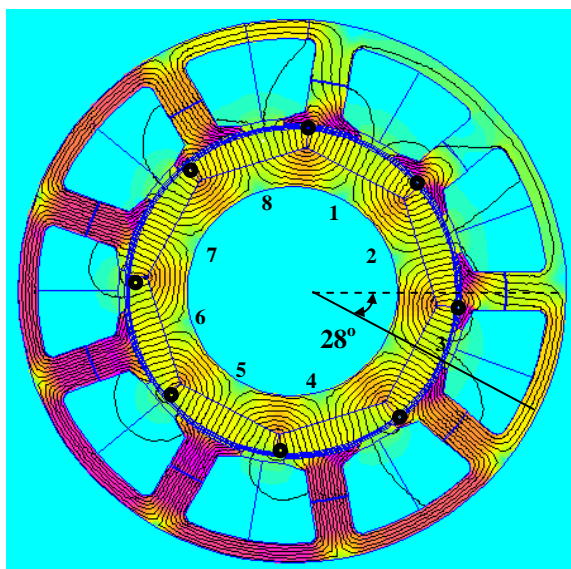
(b) Legend of field solution.



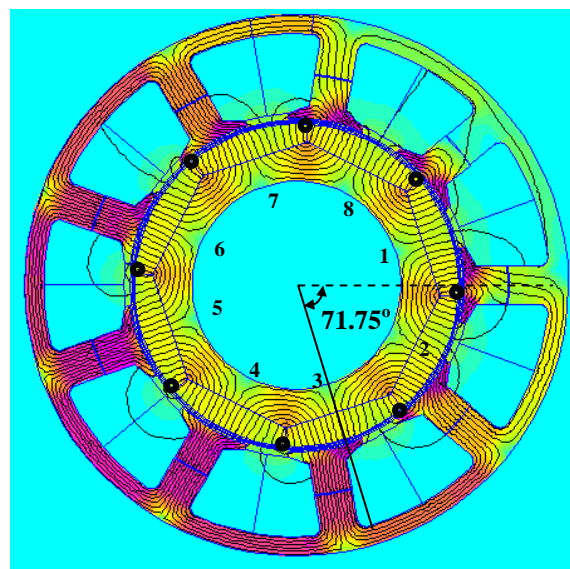
(c) FEA field solution for position A.



(d) FEA field solution for position B.

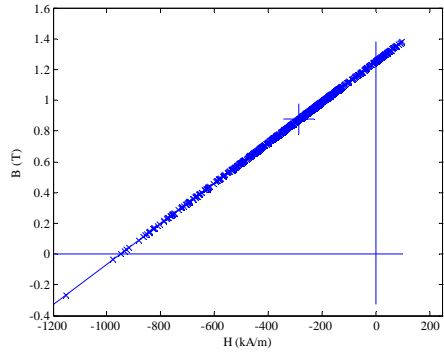


(e) FEA field solution for position C.

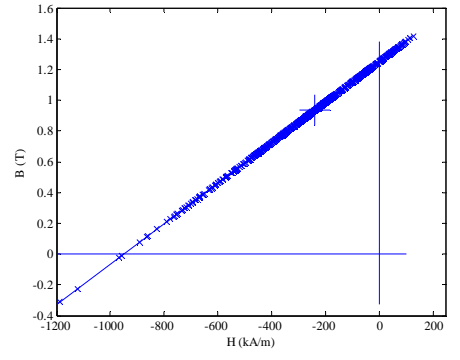


(f) FEA field solution for position D.

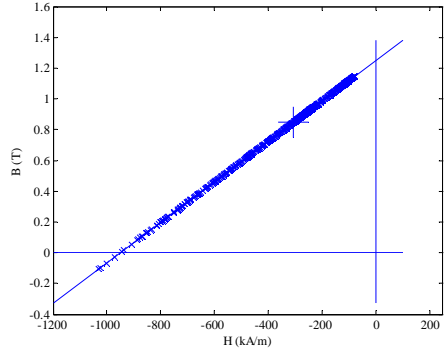
Fig. II.22 FEA field solutions of Full pole-pitch and flat bottom design when peak phase current of 3-phase winding was 1000A-t, with possible demagnetized meshes circled.



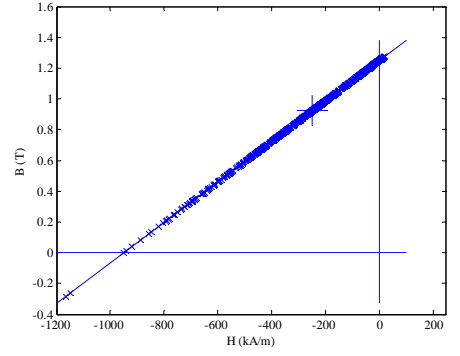
(a) Magnet\_1



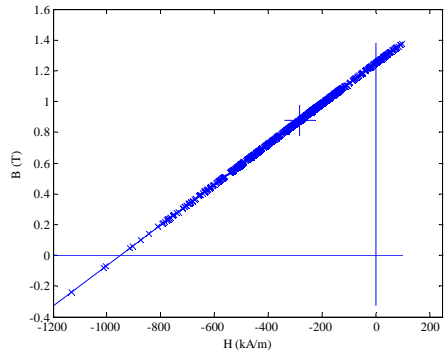
(b) Magnet\_5



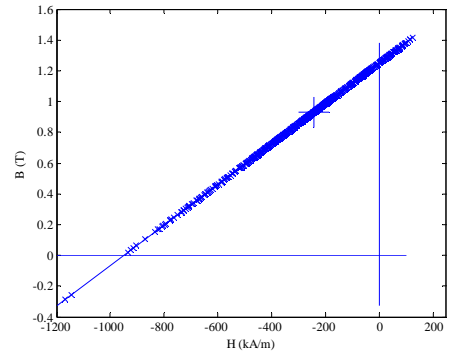
(c) Magnet\_2



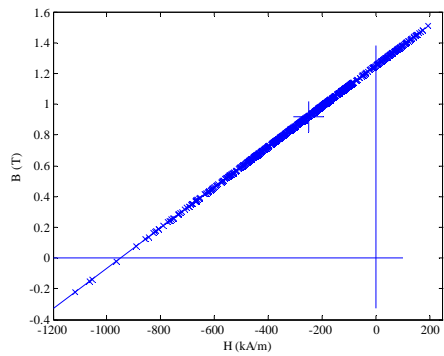
(d) Magnet\_6



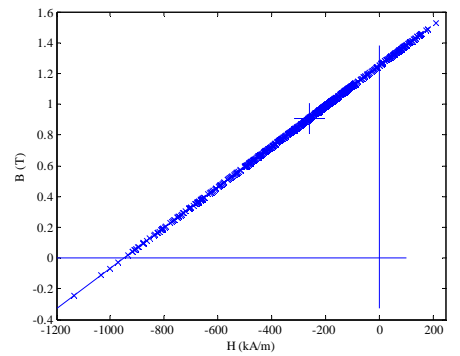
(e) Magnet\_3



(f) Magnet\_7

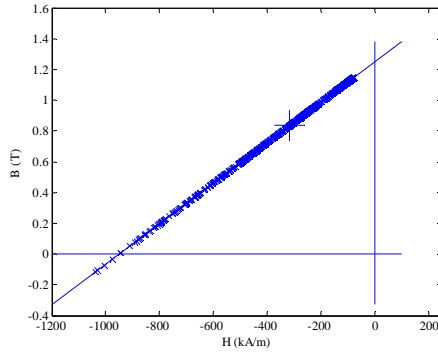


(h) Magnet\_4

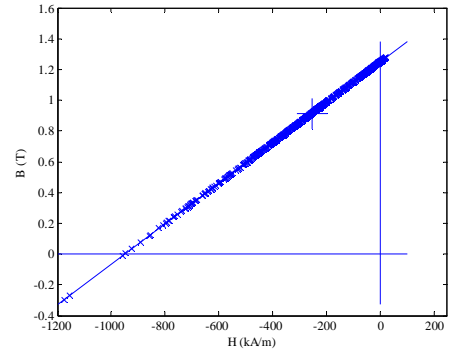


(g) Magnet\_8

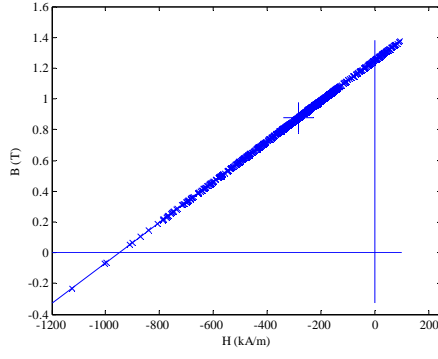
Fig. II.23 Working points plots for full pole-pitch and flat bottom magnet designed 3-phase model with 1000A-t peak phase current at position A.



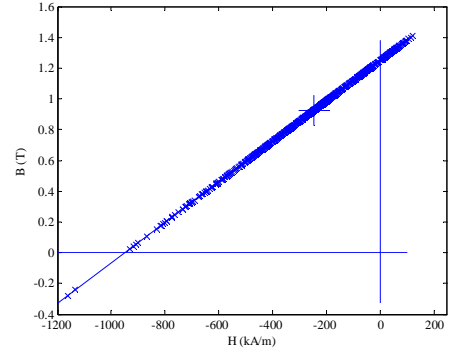
(a) Magnet\_1



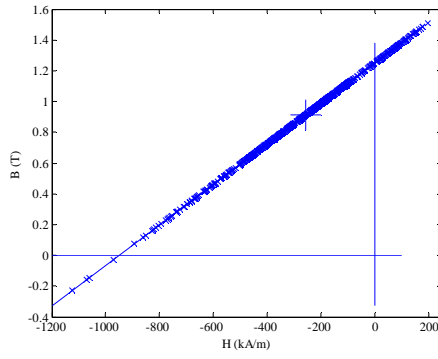
(b) Magnet\_5



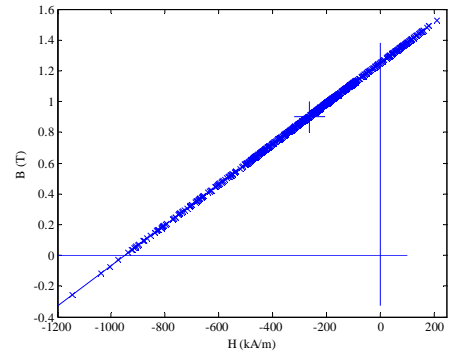
(c) Magnet\_2



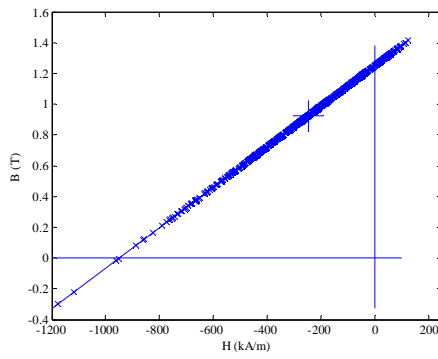
(d) Magnet\_6



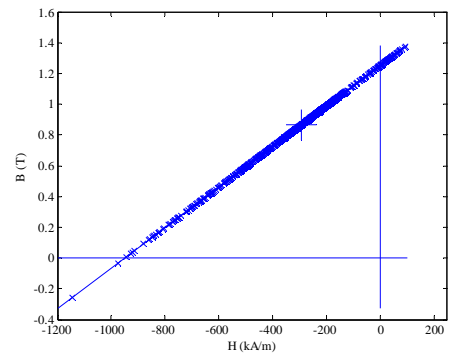
(e) Magnet\_3



(f) Magnet\_7

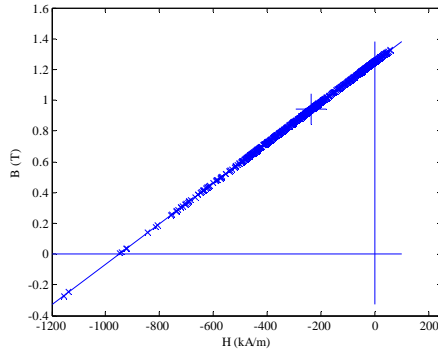


(h) Magnet\_4

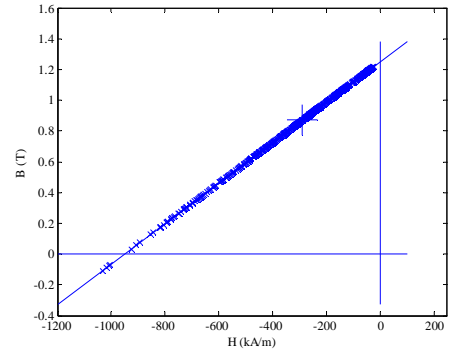


(g) Magnet\_8

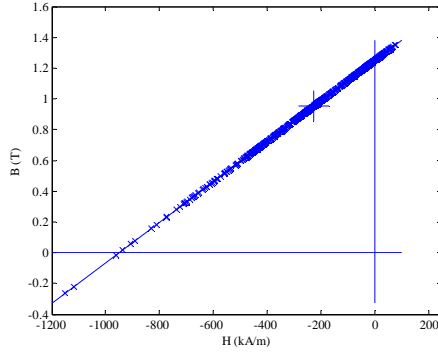
Fig. II.24 Working points plots for full pole-pitch and flat bottom magnet designed 3-phase model with 1000A-t peak phase current at position B.



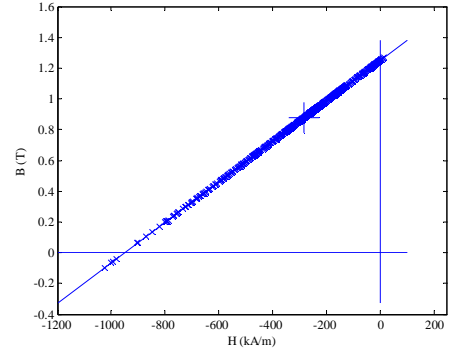
(a) Magnet\_1



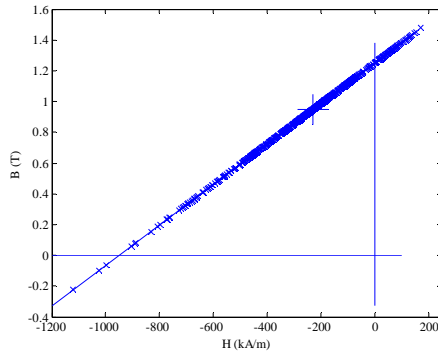
(b) Magnet\_5



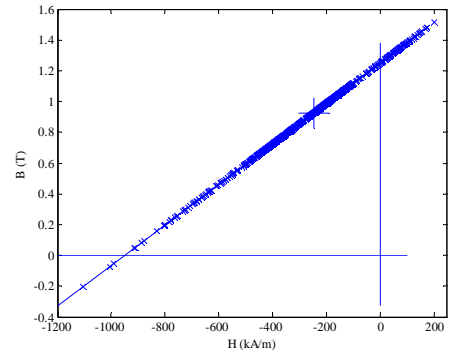
(c) Magnet\_2



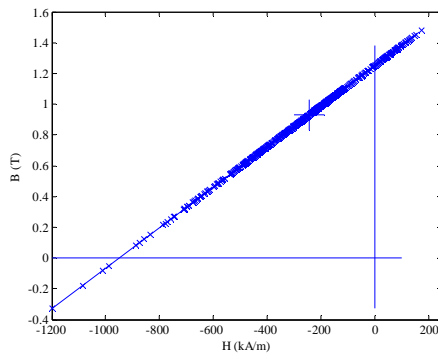
(d) Magnet\_6



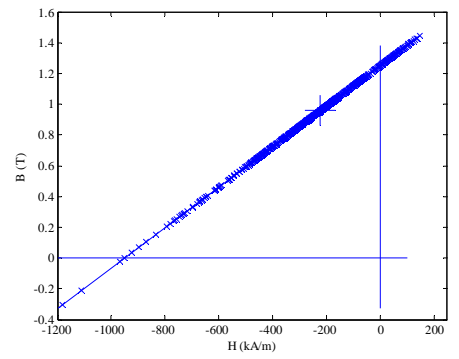
(e) Magnet\_3



(f) Magnet\_7

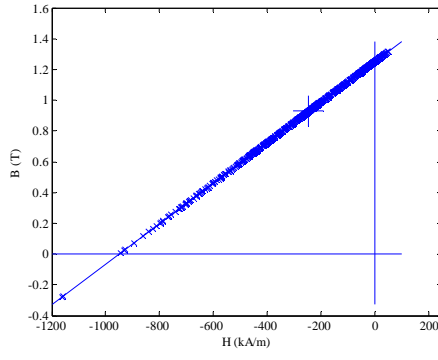


(h) Magnet\_4

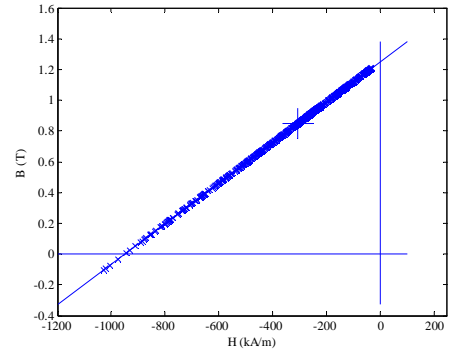


(g) Magnet\_8

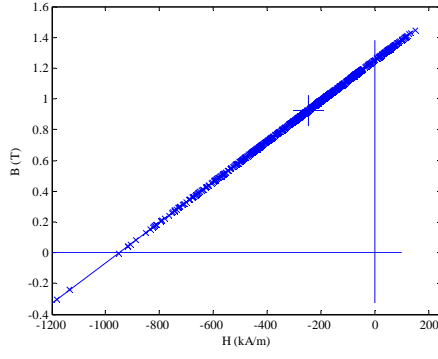
Fig. II.25 Working points plots for full pole-pitch and flat bottom magnet designed 3-phase model with 1000A-t peak phase current at position C.



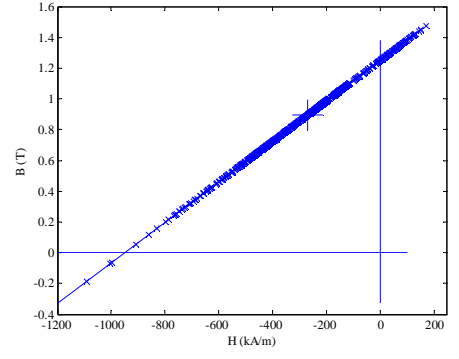
(a) Magnet\_1



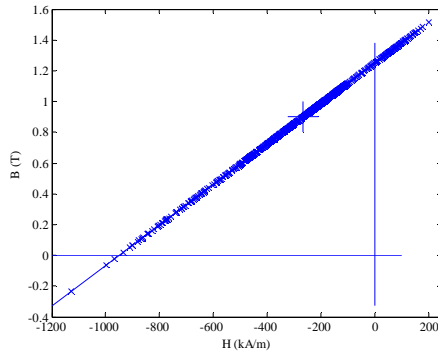
(b) Magnet\_5



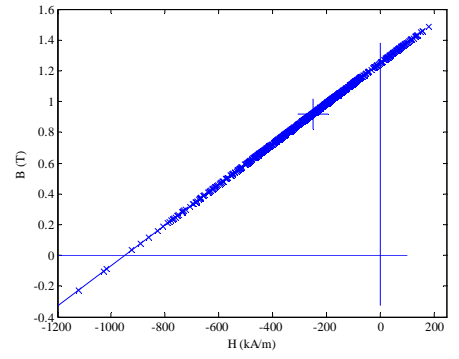
(c) Magnet\_2



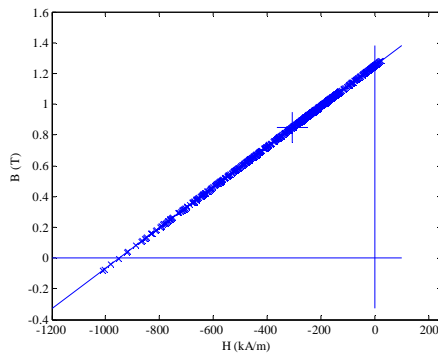
(d) Magnet\_6



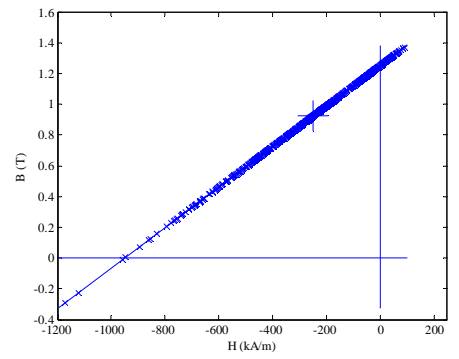
(e) Magnet\_3



(f) Magnet\_7

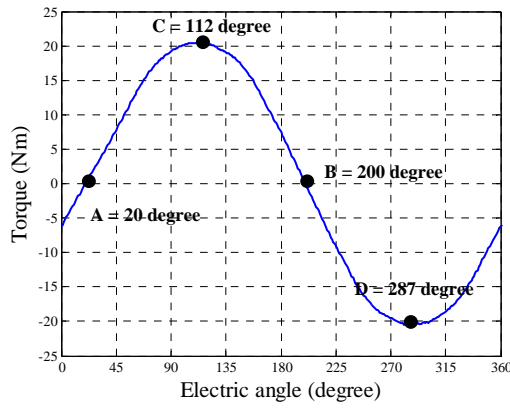


(h) Magnet\_4

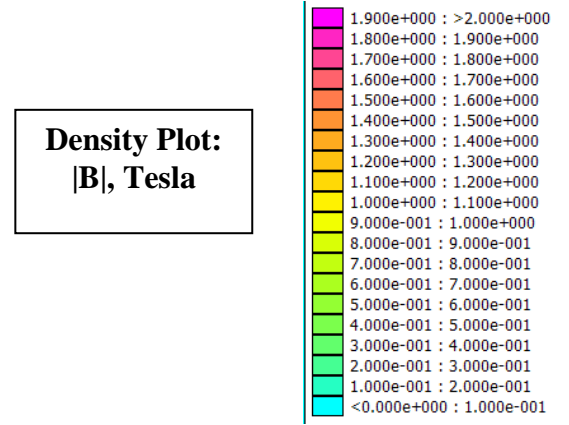


(g) Magnet\_8

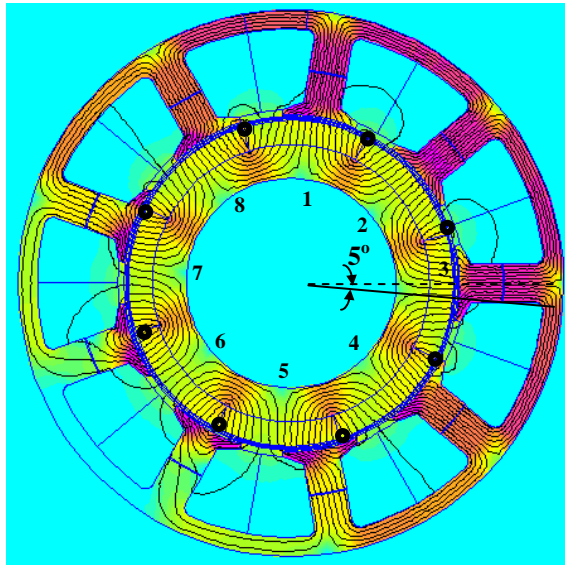
Fig. II.26 Working points plots for full pole-pitch and flat bottom magnet designed 3-phase model with 1000A-t peak phase current at position D.



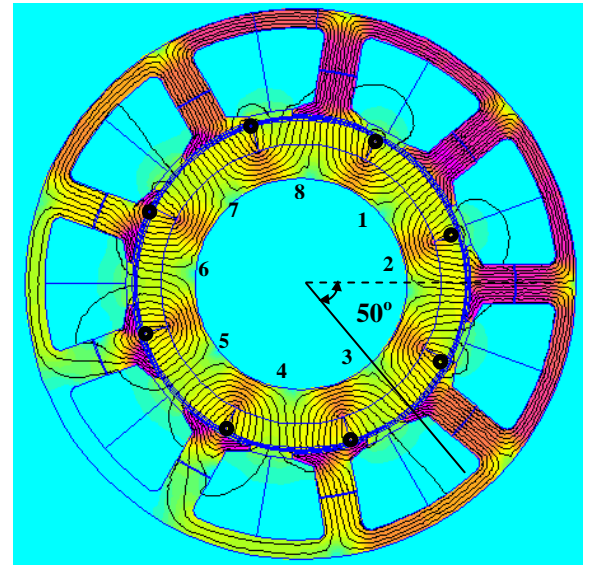
(a) Demagnetization checking position.



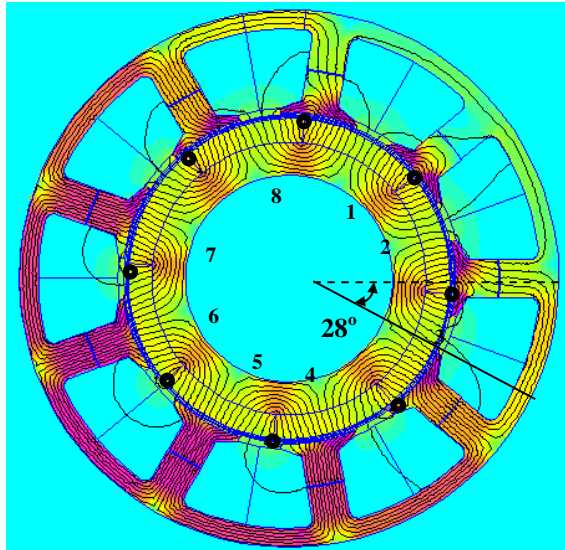
(b) Legend of field solution.



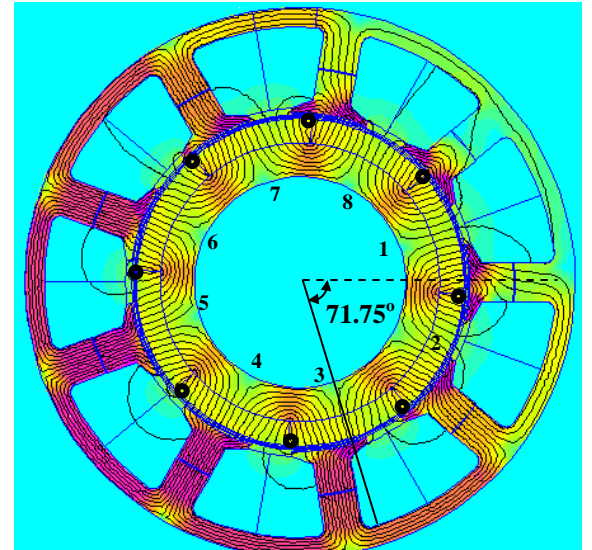
(c) FEA field solution for position A.



(d) FEA field solution for position B.



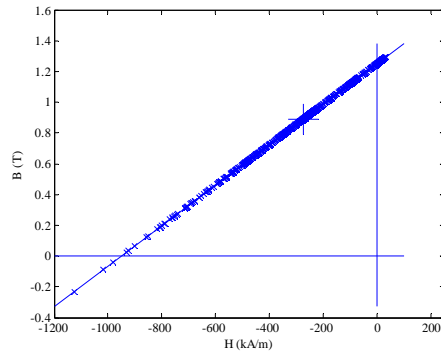
(e) FEA field solution for position C.



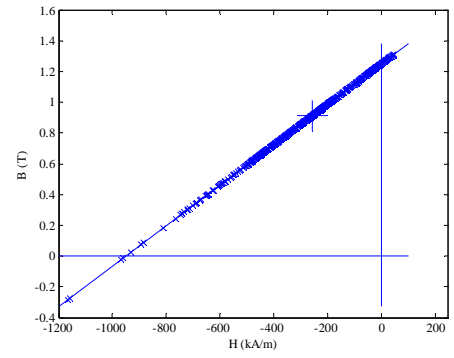
(f) FEA field solution for position D.

Fig. II.27 FEA field solutions of full pole-pitch arc design when peak phase current of 3-phase winding was 1000A-t, with possible demagnetized meshes circled.

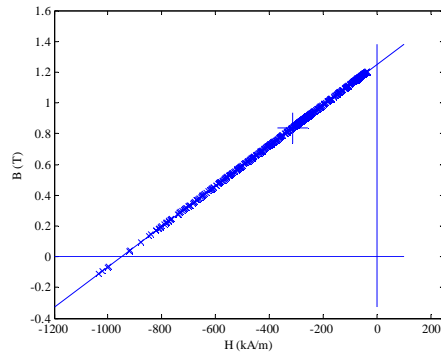




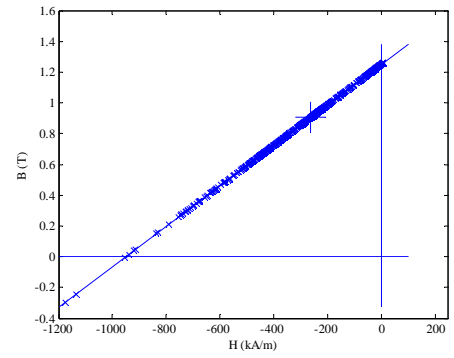
(a) Magnet\_1



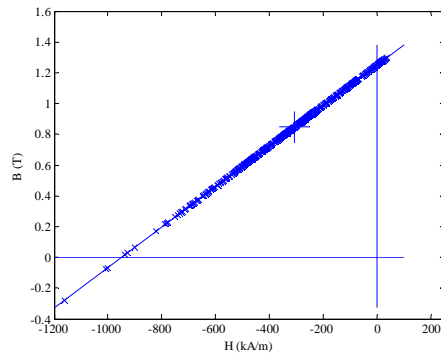
(b) Magnet\_5



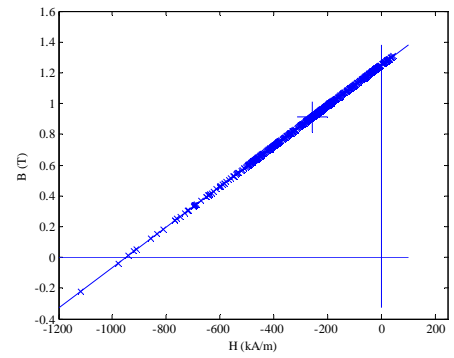
(c) Magnet\_2



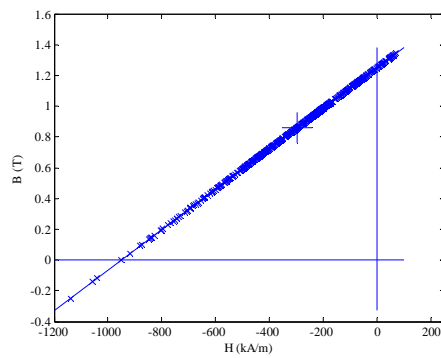
(d) Magnet\_6



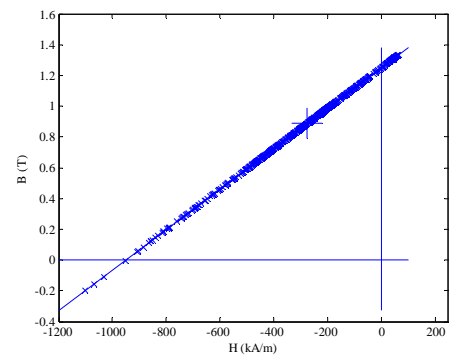
(e) Magnet\_3



(f) Magnet\_7

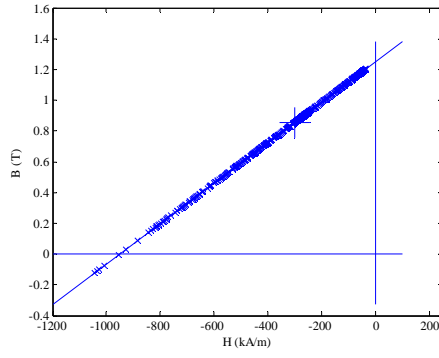


(h) Magnet\_4

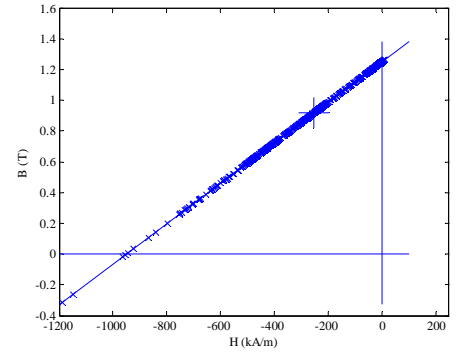


(g) Magnet\_8

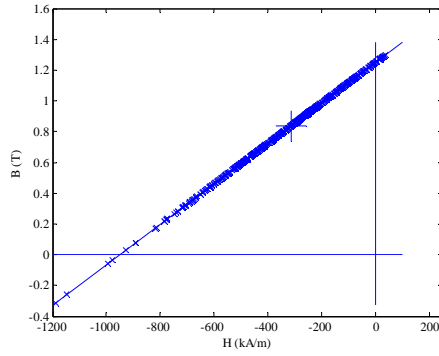
Fig. II.28 Working points plots for full pole-pitch arc designed 3-phase model with 1000A-t peak phase current at position A.



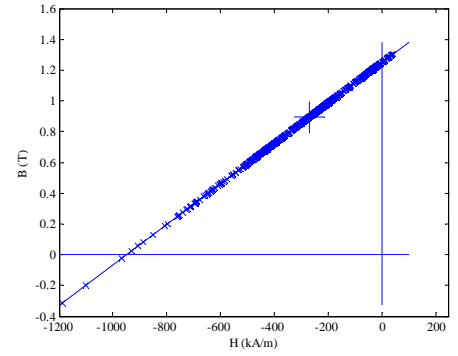
(a) Magnet\_1



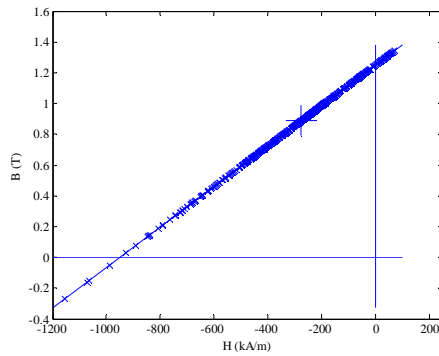
(b) Magnet\_5



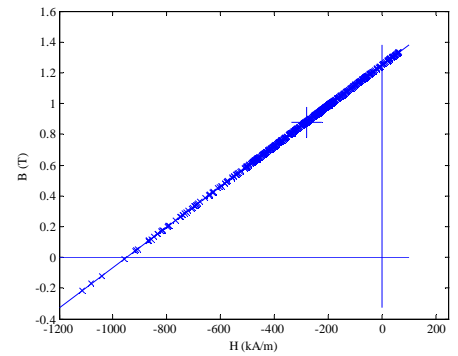
(c) Magnet\_2



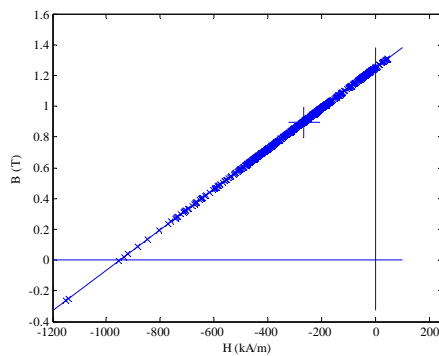
(d) Magnet\_6



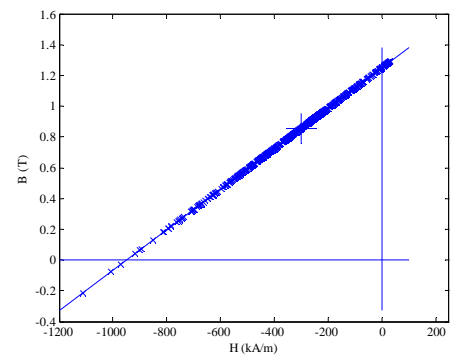
(e) Magnet\_3



(f) Magnet\_7

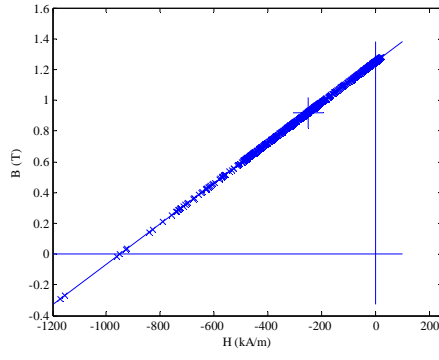


(h) Magnet\_4

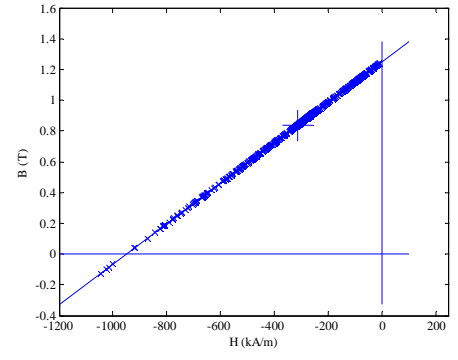


(g) Magnet\_8

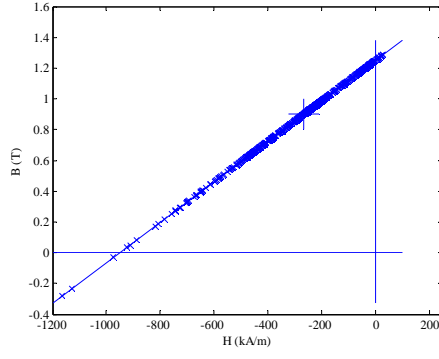
Fig. II.29 Working points plots for full pole-pitch arc designed 3-phase model with 1000A-t peak phase current at position B.



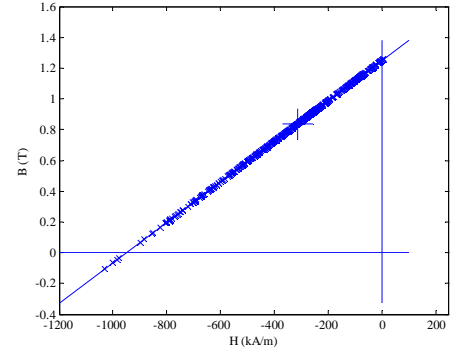
(a) Magnet\_1



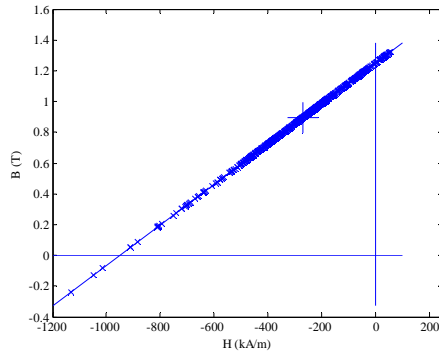
(b) Magnet\_5



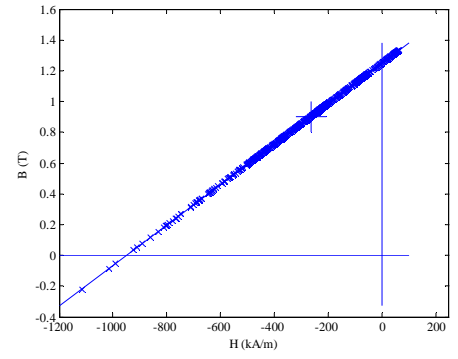
(c) Magnet\_2



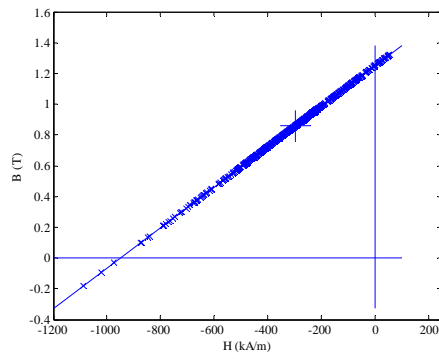
(d) Magnet\_6



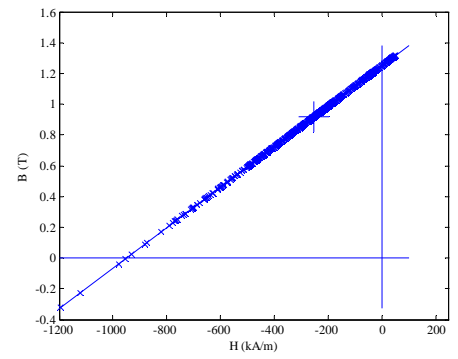
(e) Magnet\_3



(f) Magnet\_7

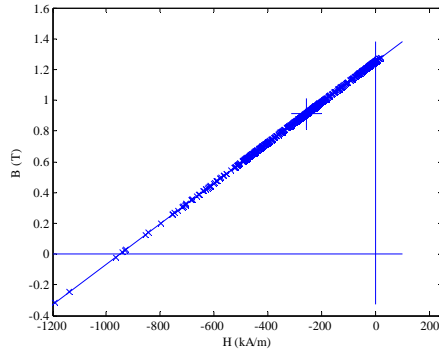


(h) Magnet\_4

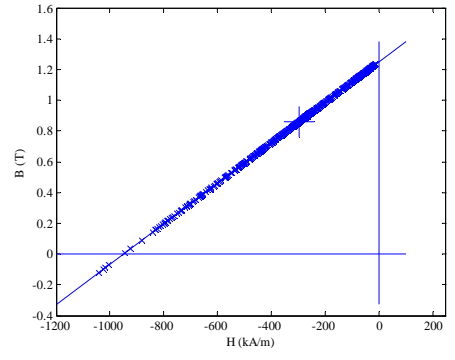


(g) Magnet\_8

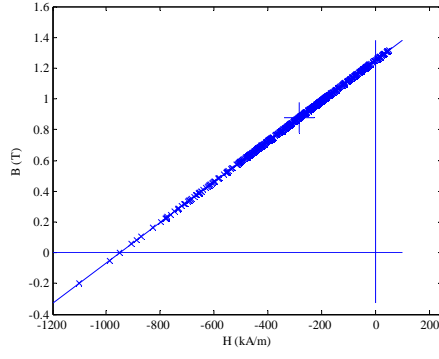
Fig. II.30 Working points plots for full pole-pitch arc designed 3-phase model with 1000A-t peak phase current at position C.



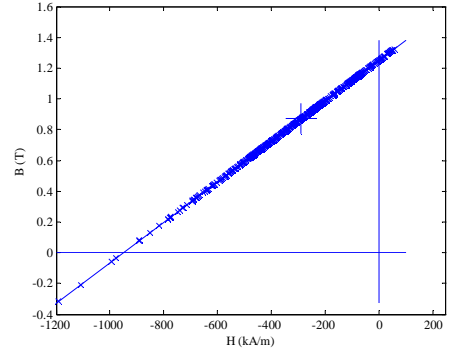
(a) Magnet\_1



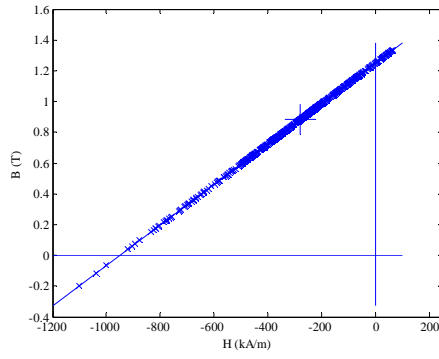
(b) Magnet\_5



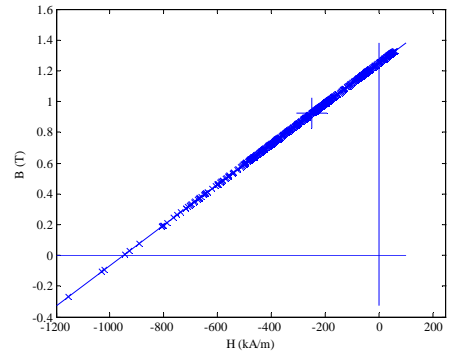
(c) Magnet\_2



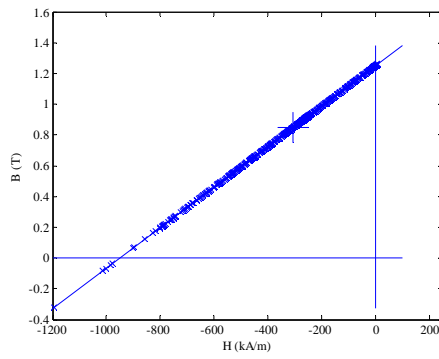
(d) Magnet\_6



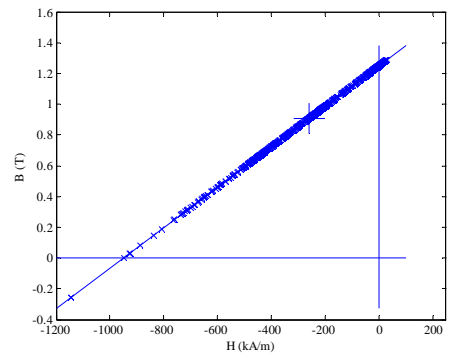
(e) Magnet\_3



(f) Magnet\_7

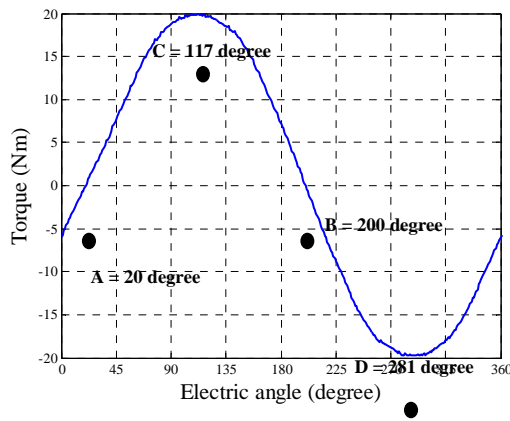


(h) Magnet\_4



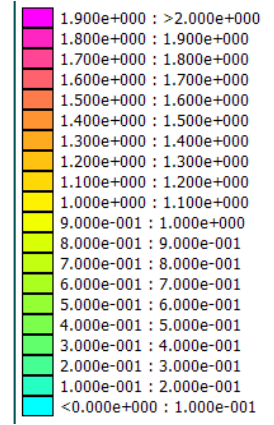
(g) Magnet\_8

Fig. II.31 Working points plots for full pole-pitch arc designed 3-phase model with 1000A-t peak phase current at position D.

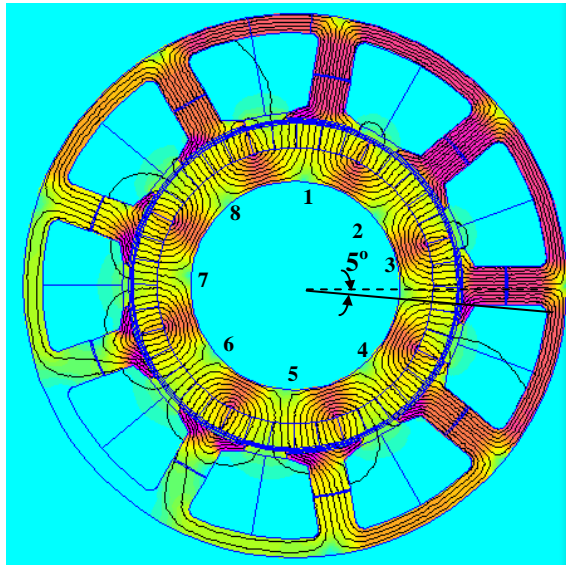


(a) Demagnetization checking position.

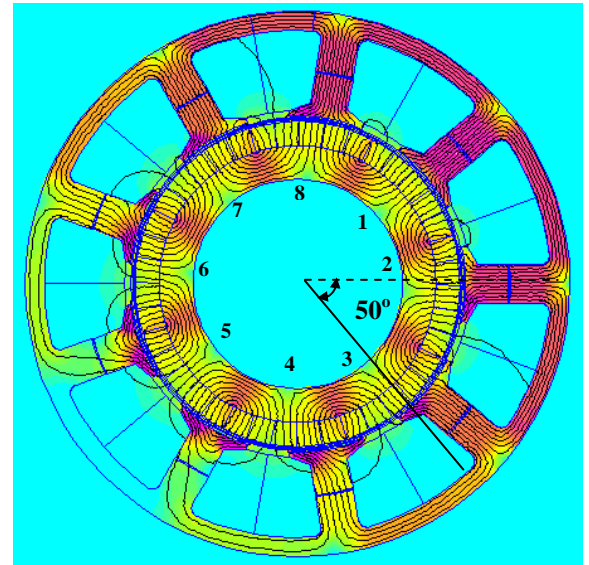
**Density Plot:**  
**|B|, Tesla**



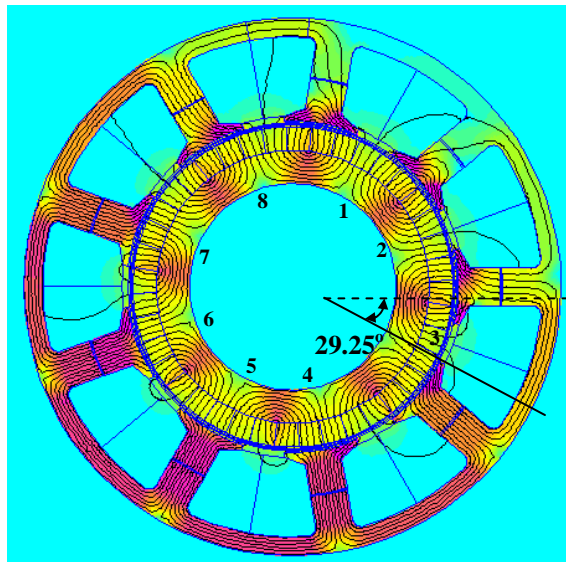
(b) Legend of field solution.



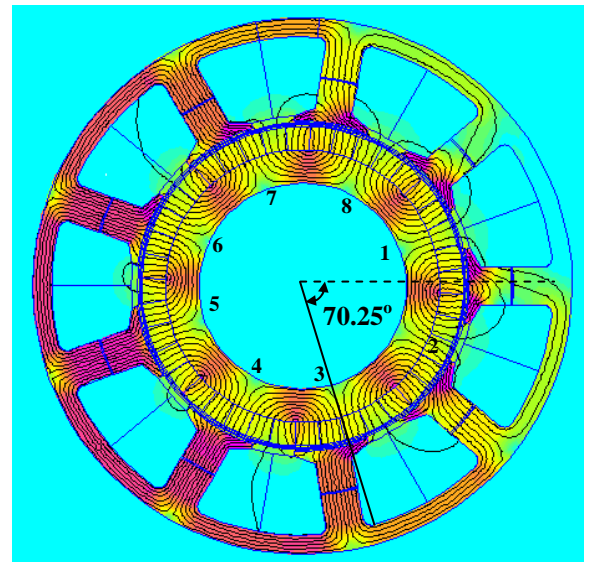
(c) FEA field solution for position A.



(d) FEA field solution for position B.

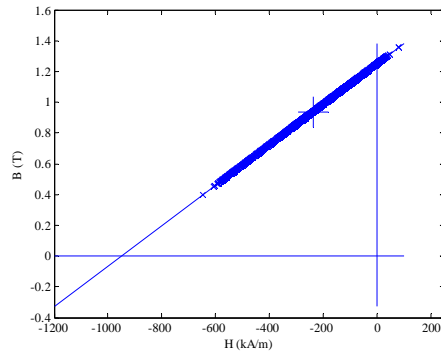


(e) FEA field solution for position C.

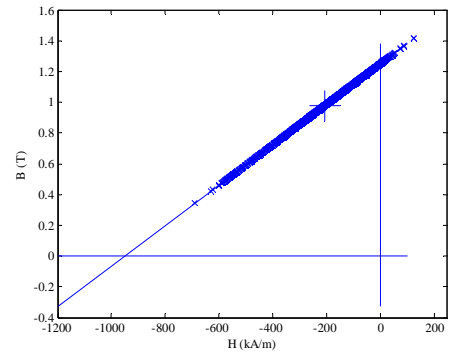


(f) FEA field solution for position D.

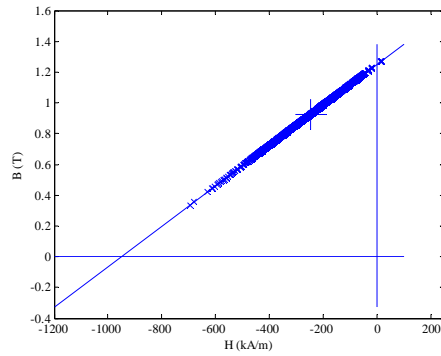
Fig. II.32 FEA field solutions of full pole-pitch pseudo-radial magnetization design when peak phase current of 3-phase winding was 1000A-t, with possible demagnetized meshes circled.



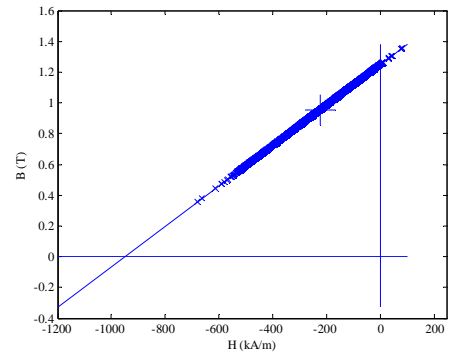
(a) Magnet\_1



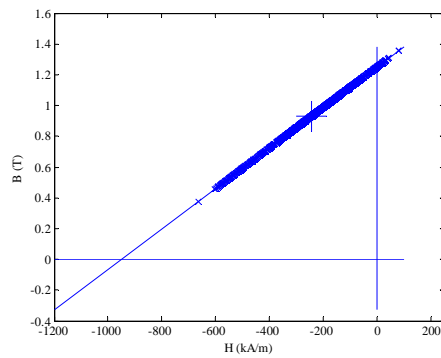
(b) Magnet\_5



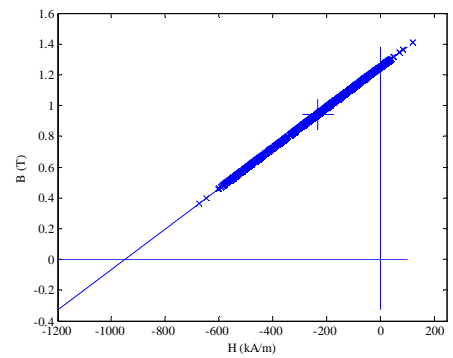
(c) Magnet\_2



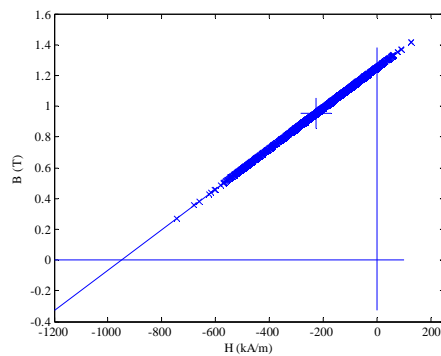
(d) Magnet\_6



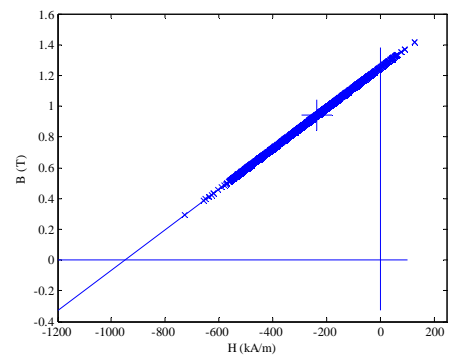
(e) Magnet\_3



(f) Magnet\_7

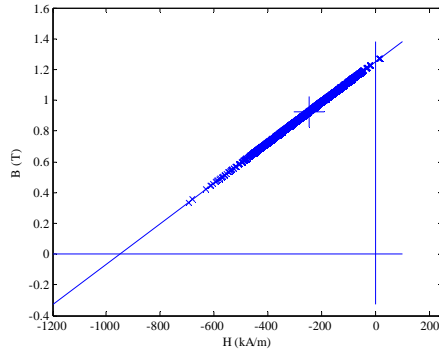


(h) Magnet\_4

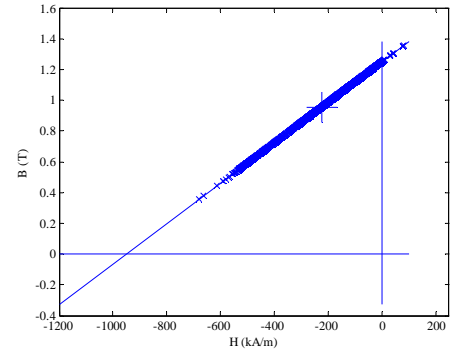


(g) Magnet\_8

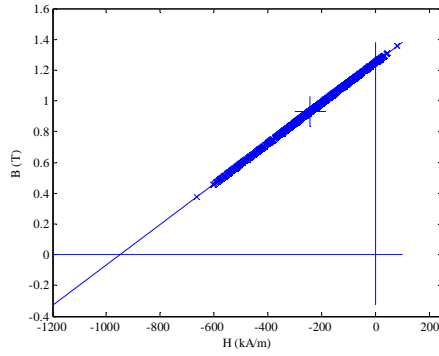
Fig. II.33 Working points plots for full pole-pitch pseudo-radial magnetization designed 3-phase model with 1000A-t peak phase current at position A.



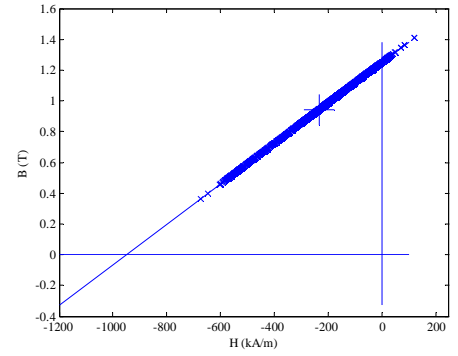
(a) Magnet\_1



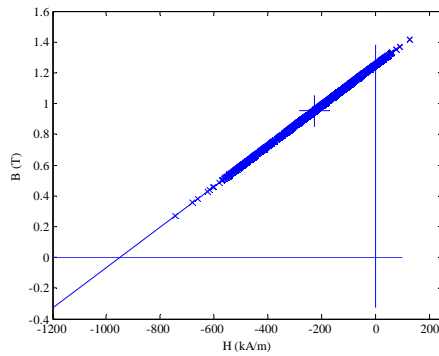
(b) Magnet\_5



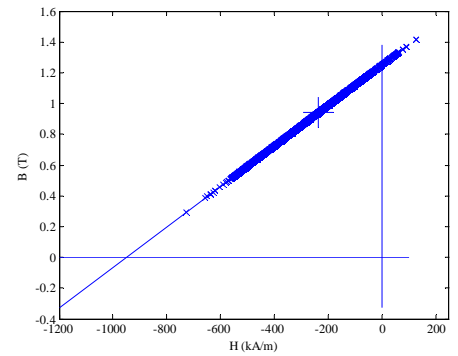
(c) Magnet\_2



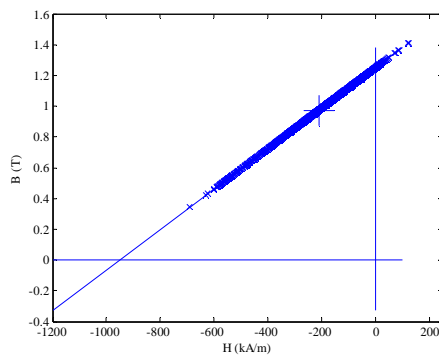
(d) Magnet\_6



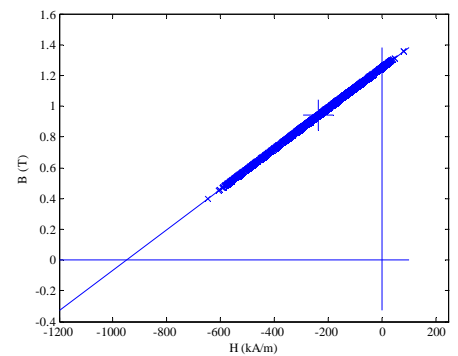
(e) Magnet\_3



(f) Magnet\_7

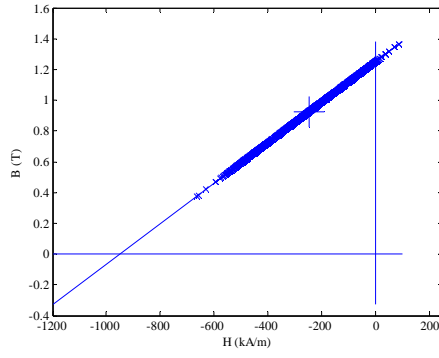


(h) Magnet\_4

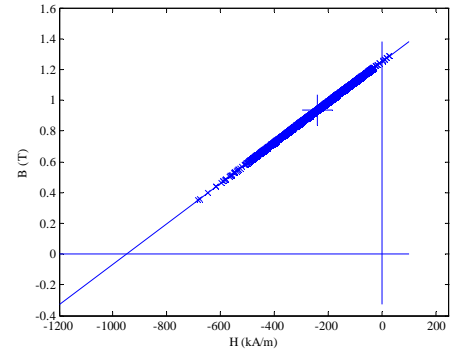


(g) Magnet\_8

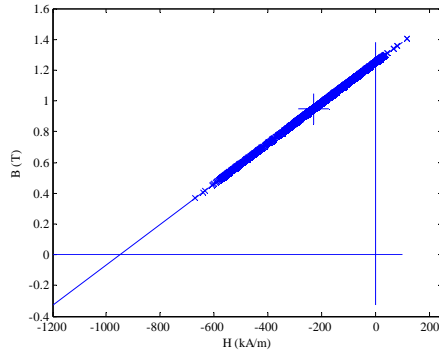
Fig. II.34 Working points plots for full pole-pitch pseudo-radial magnetization designed 3-phase model with 1000A-t peak phase current at position B.



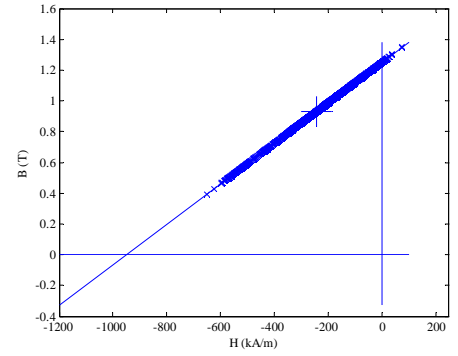
(a) Magnet\_1



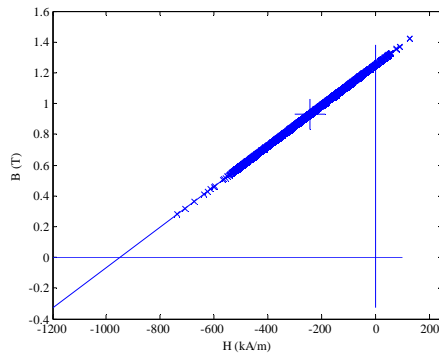
(b) Magnet\_5



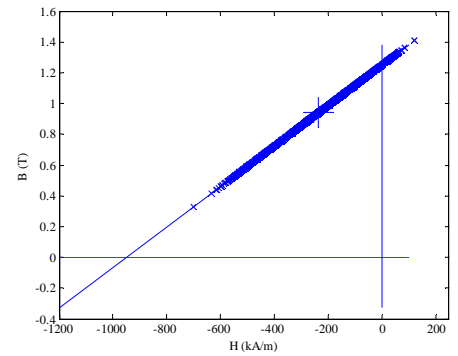
(c) Magnet\_2



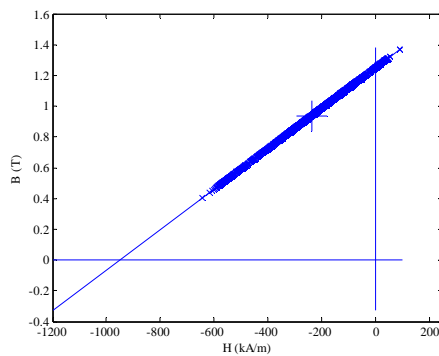
(d) Magnet\_6



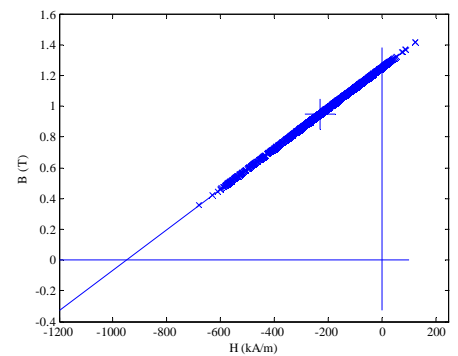
(e) Magnet\_3



(f) Magnet\_7



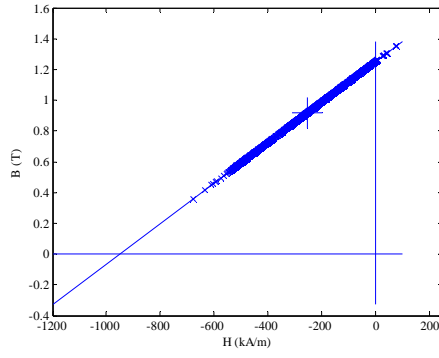
(h) Magnet\_4



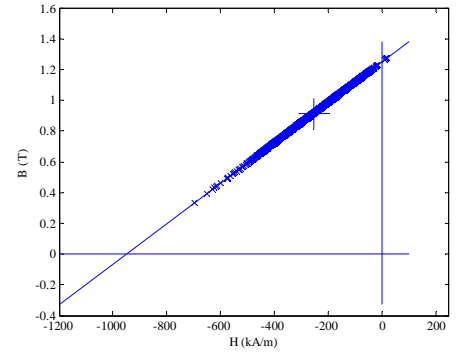
(g) Magnet\_8

Fig. II.35 Working points plots for full pole-pitch pseudo-radial magnetization designed 3-phase model with 1000A-t peak phase current at position C.

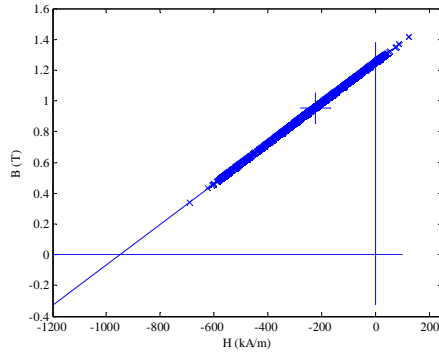




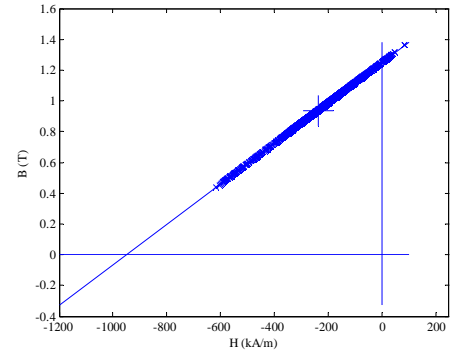
(a) Magnet\_1



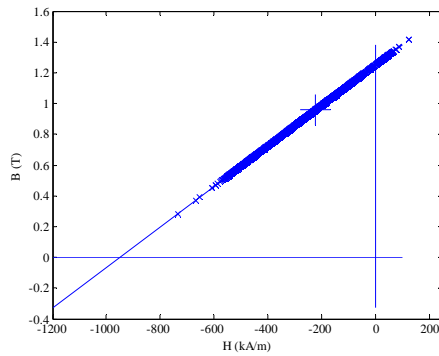
(b) Magnet\_5



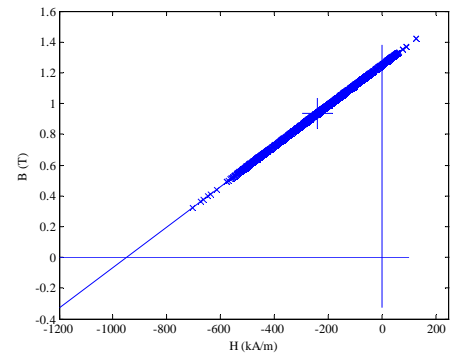
(c) Magnet\_2



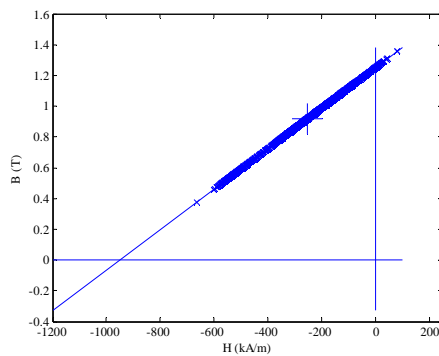
(d) Magnet\_6



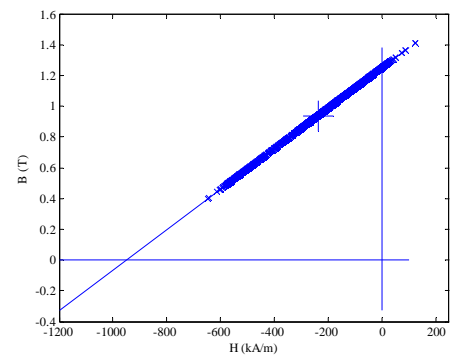
(e) Magnet\_3



(f) Magnet\_7

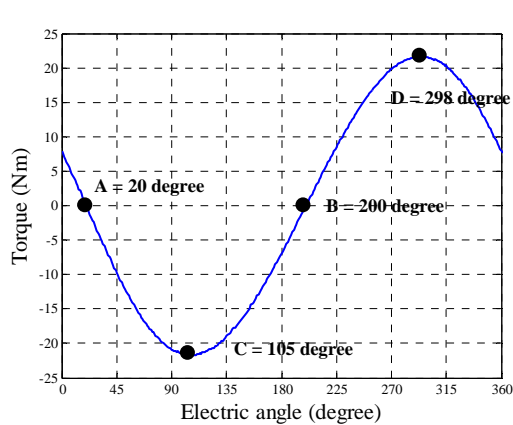


(h) Magnet\_4

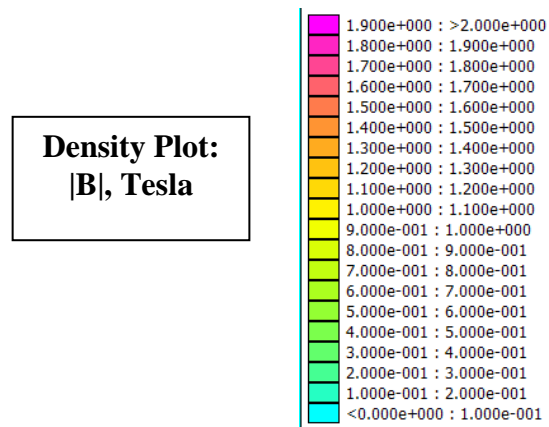


(g) Magnet\_8

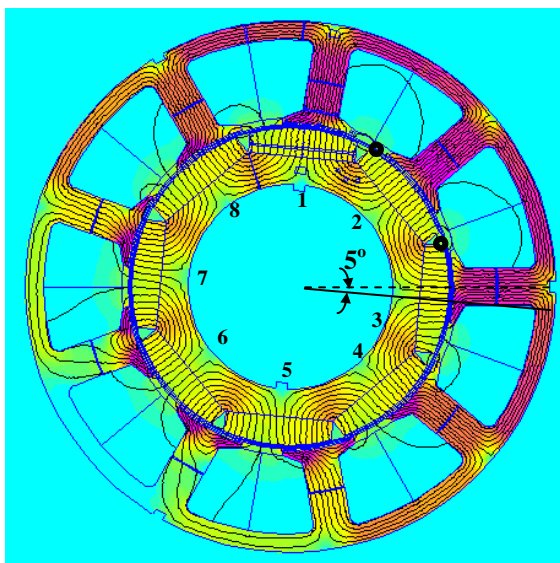
Fig. II.36 Working points plots for full pole-pitch pseudo-radial magnetization designed 3-phase model with 1000A-t peak phase current at position D.



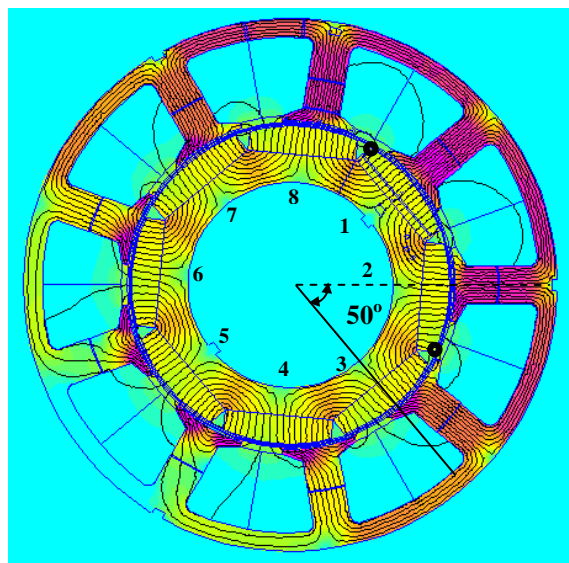
(a) Demagnetization checking position.



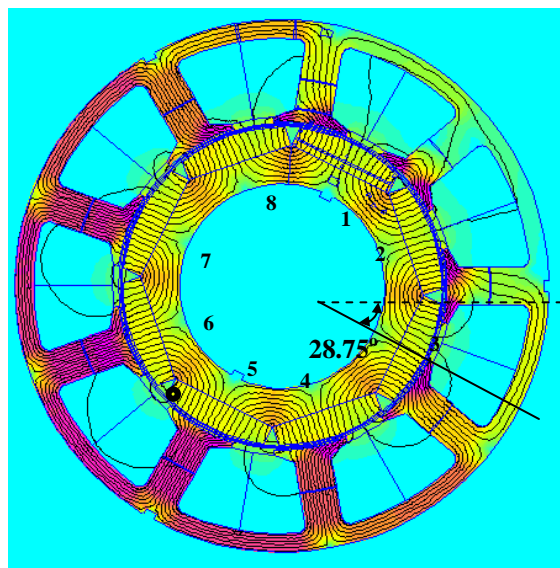
(b) Legend of field solution.



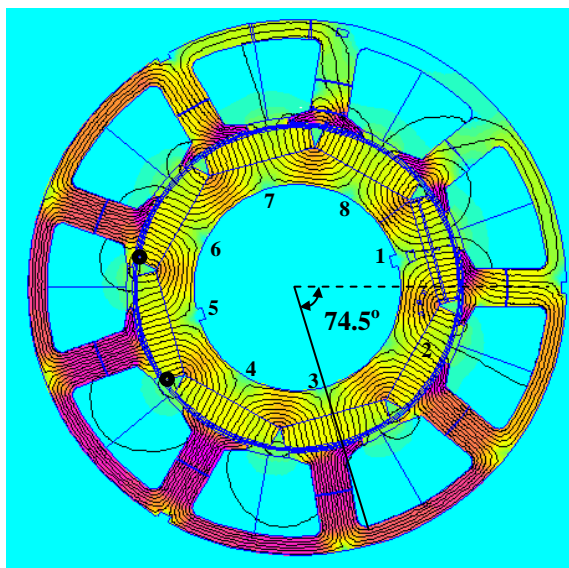
(c) FEA field solution for position A.



(d) FEA field solution for position B.

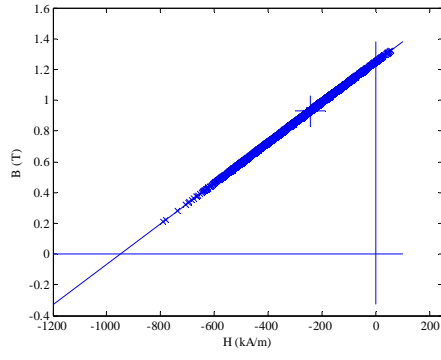


(e) FEA field solution for position C.

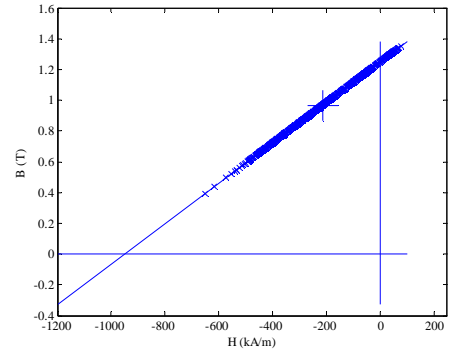


(f) FEA field solution for position D.

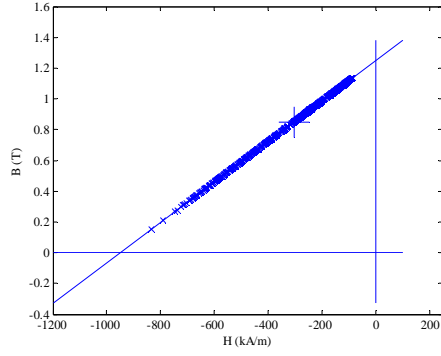
Fig. II.37 FEA field solutions of bread loaf design when peak phase current of 9-phase winding was 1000A-t, with possible demagnetized meshes circled.



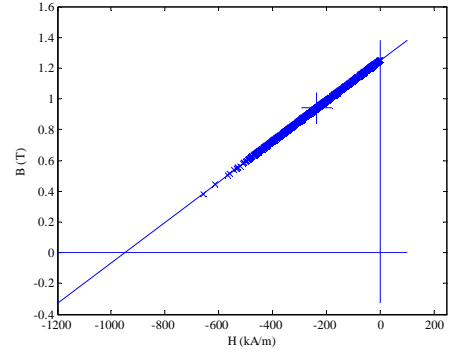
(a) Magnet\_1



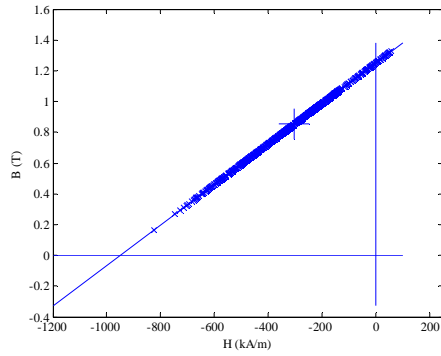
(b) Magnet\_5



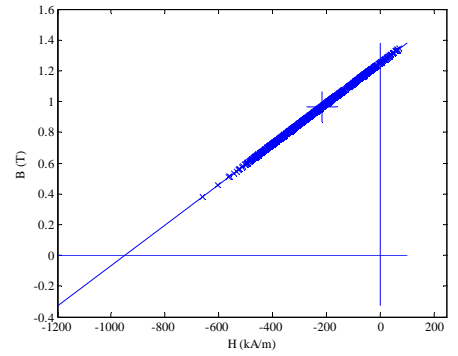
(c) Magnet\_2



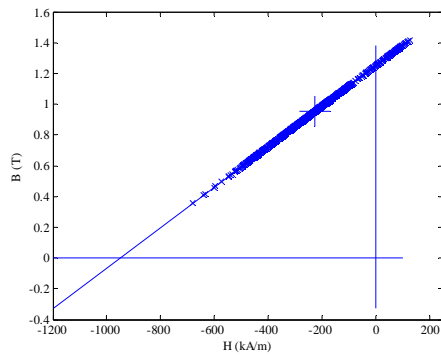
(d) Magnet\_6



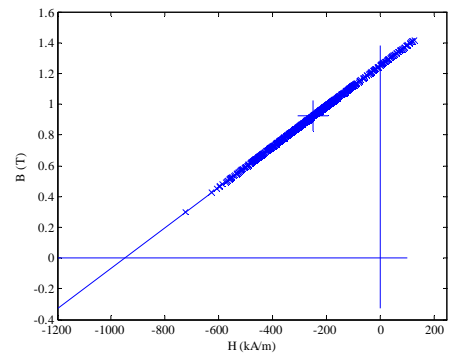
(e) Magnet\_3



(f) Magnet\_7

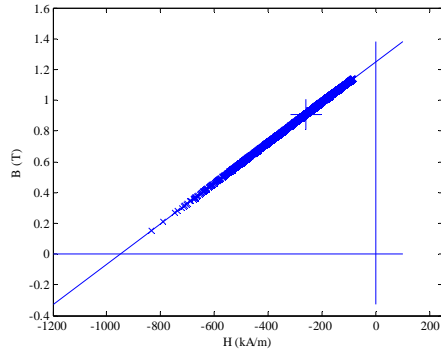


(h) Magnet\_4

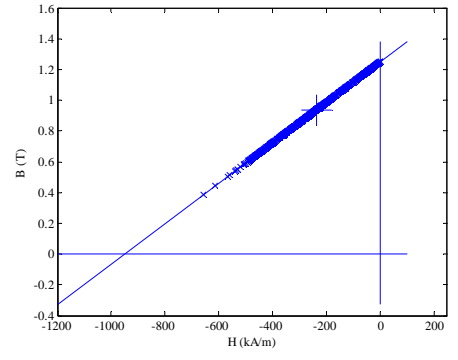


(g) Magnet\_8

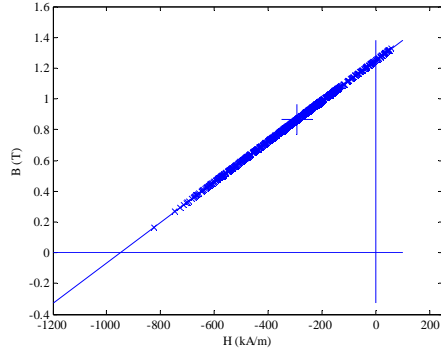
Fig. II.38 Working points plots for bread loaf designed 9-phase model with 1000A-t peak phase current at position A.



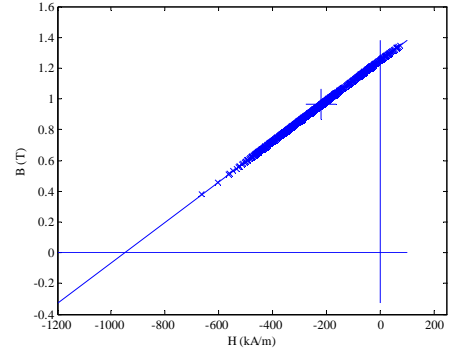
(a) Magnet\_1



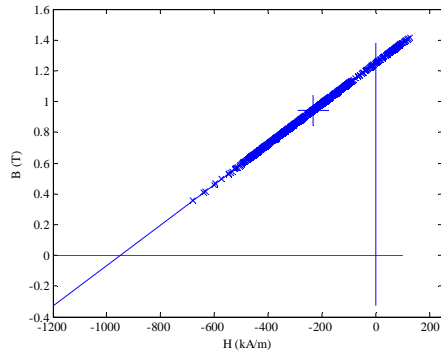
(b) Magnet\_5



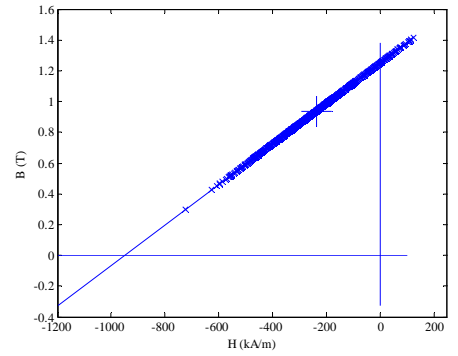
(c) Magnet\_2



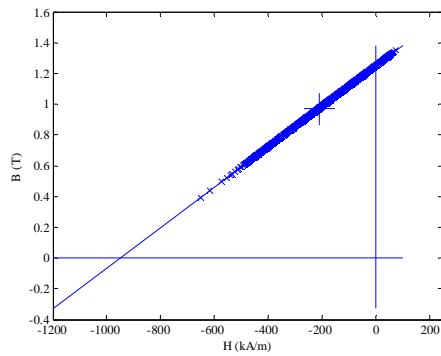
(d) Magnet\_6



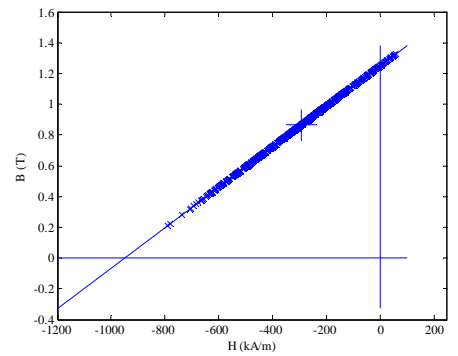
(e) Magnet\_3



(f) Magnet\_7

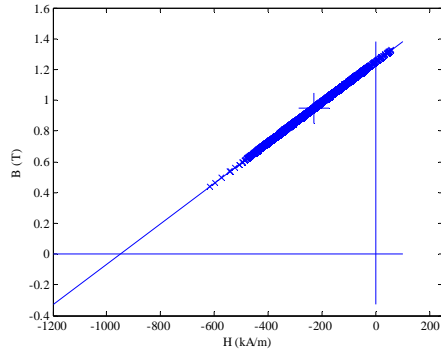


(h) Magnet\_4

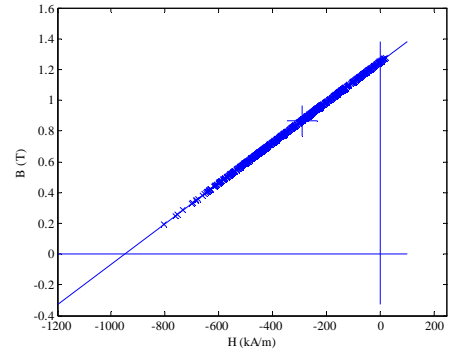


(g) Magnet\_8

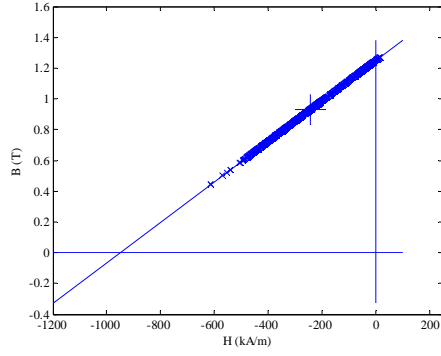
Fig. II.39 Working points plots for bread loaf designed 9-phase model with 1000A-t peak phase current at position B.



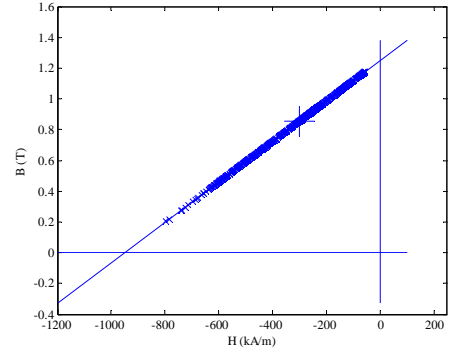
(a) Magnet\_1



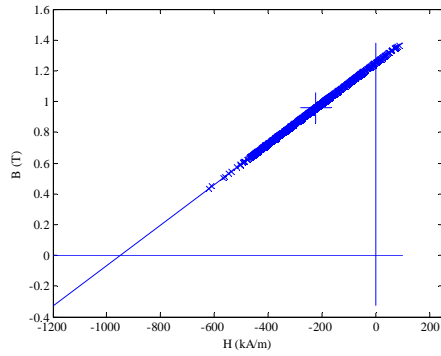
(b) Magnet\_5



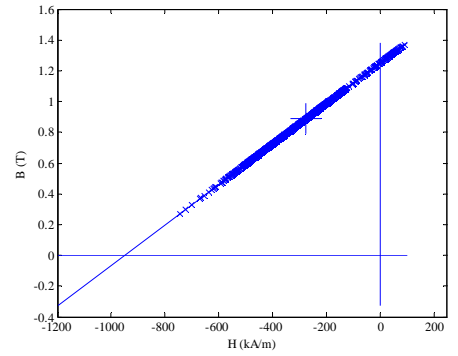
(c) Magnet\_2



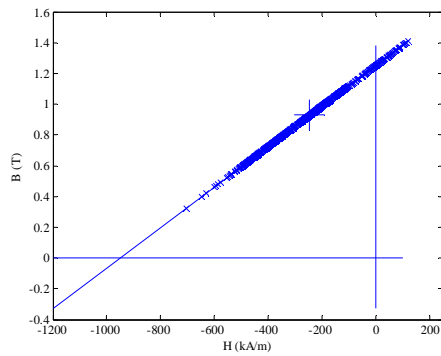
(d) Magnet\_6



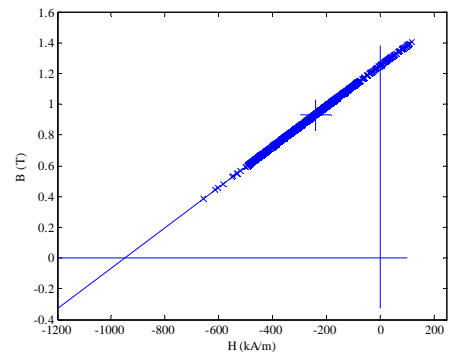
(e) Magnet\_3



(f) Magnet\_7

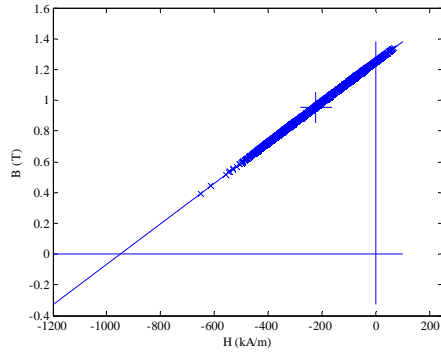


(h) Magnet\_4

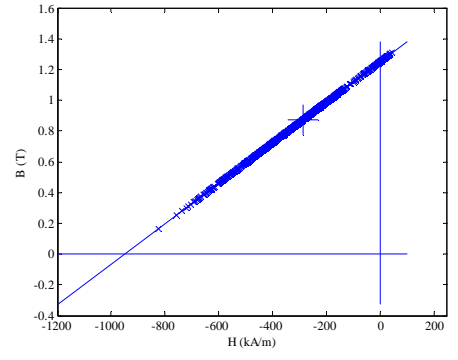


(g) Magnet\_8

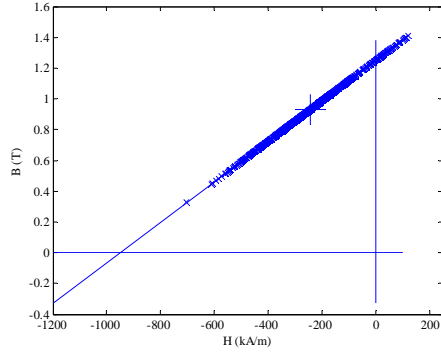
Fig. II.40 Working points plots for bread loaf designed 9-phase model with 1000A-t peak phase current at position C.



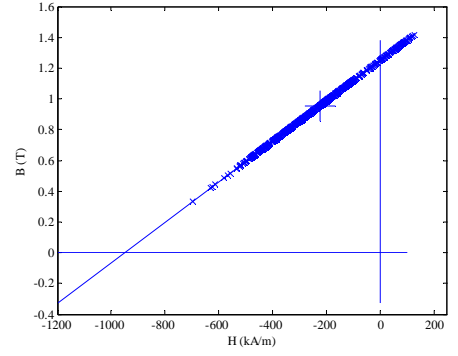
(a) Magnet\_1



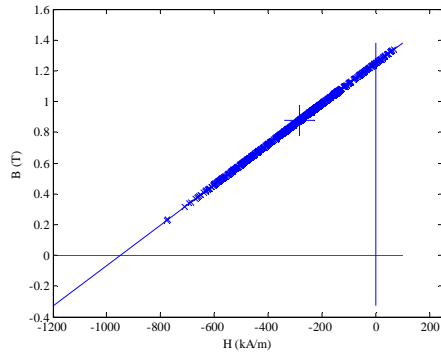
(b) Magnet\_5



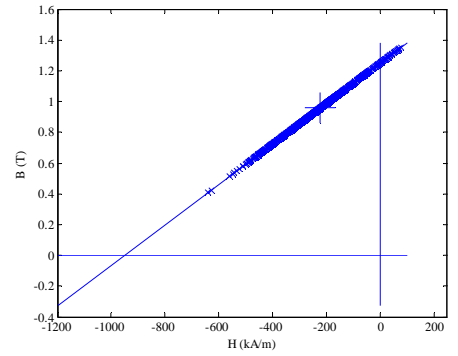
(c) Magnet\_2



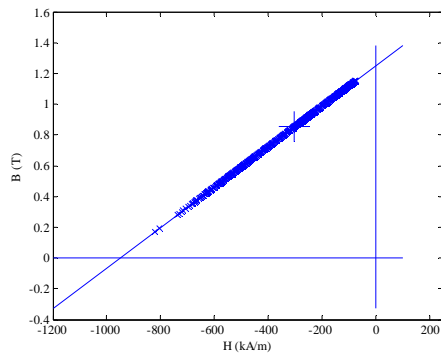
(d) Magnet\_6



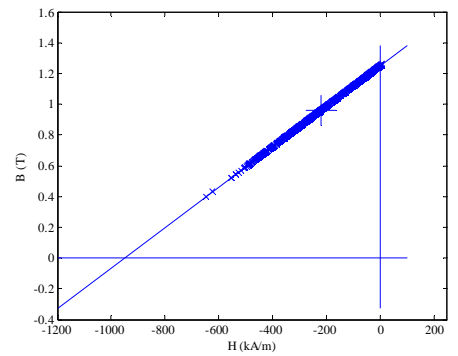
(e) Magnet\_3



(f) Magnet\_7

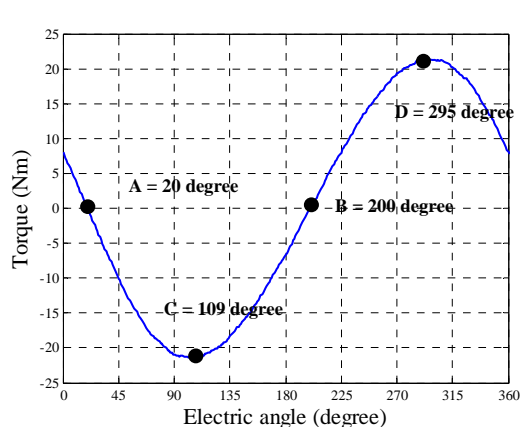


(h) Magnet\_4

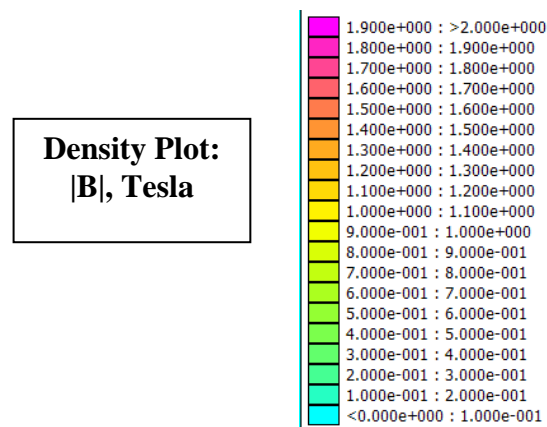


(g) Magnet\_8

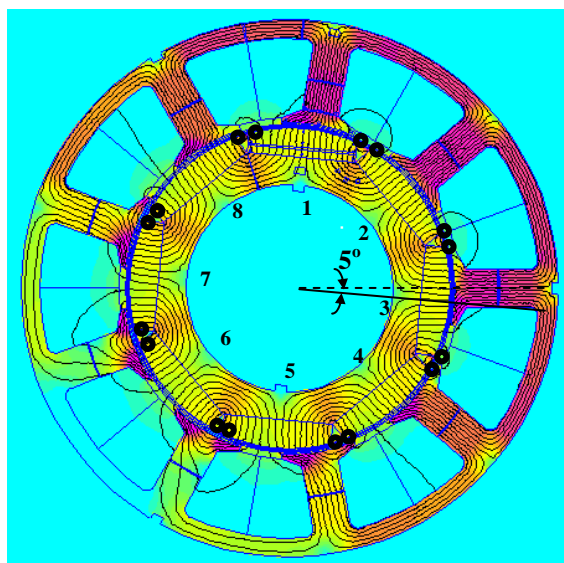
Fig. II.41 Working points plots for bread loaf designed 9-phase model with 1000A-t peak phase current at position D.



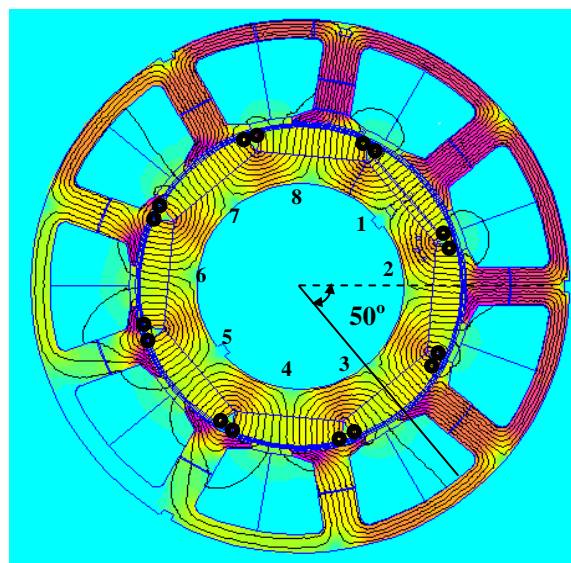
(a) Demagnetization checking position.



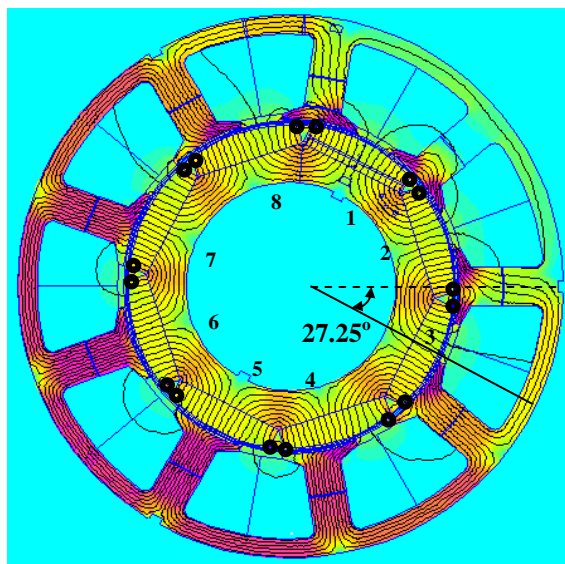
(b) Legend of field solution.



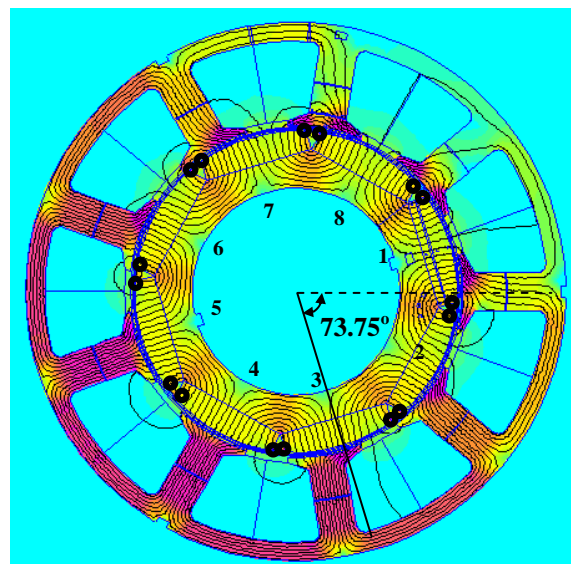
(c) FEA field solution for position A.



(d) FEA field solution for position B.

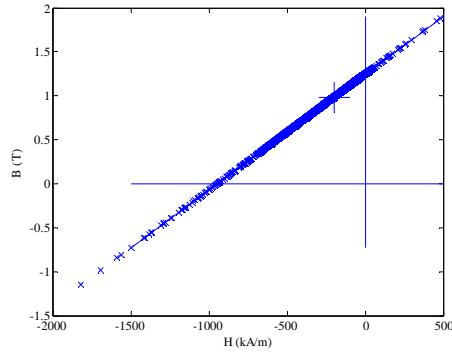


(e) FEA field solution for position C.

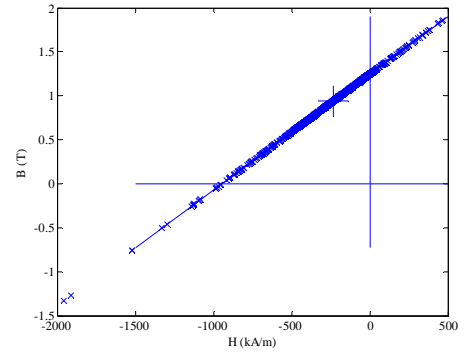


(f) FEA field solution for position D.

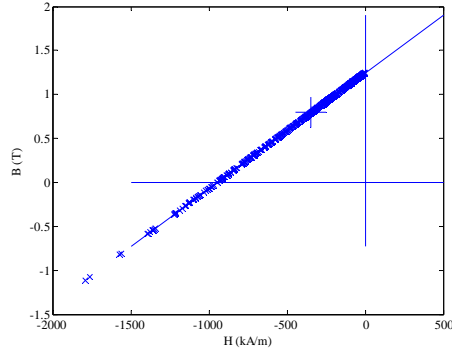
Fig. II.42 FEA field solutions of embedded bread loaf design when peak phase current of 9-phase winding was 1000A-t, with possible demagnetized meshes circled.



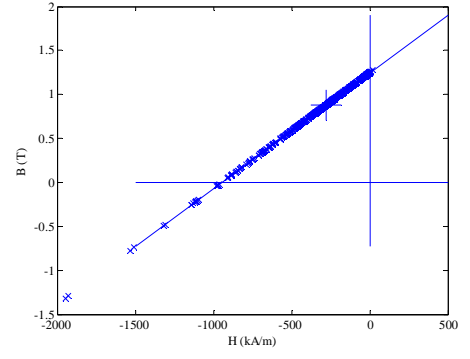
(a) Magnet\_1



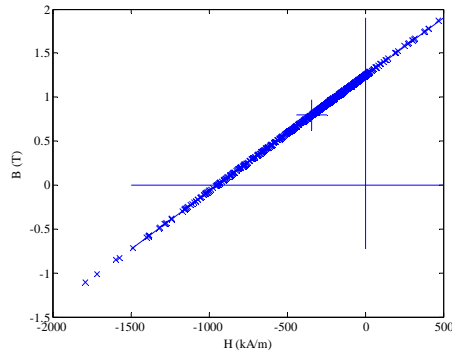
(b) Magnet\_5



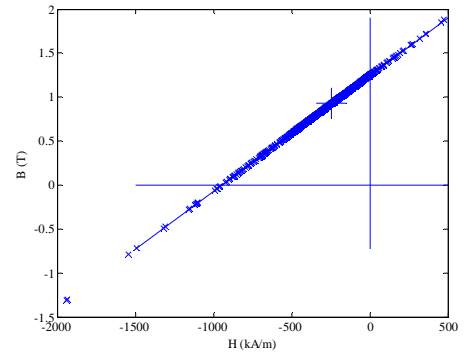
(c) Magnet\_2



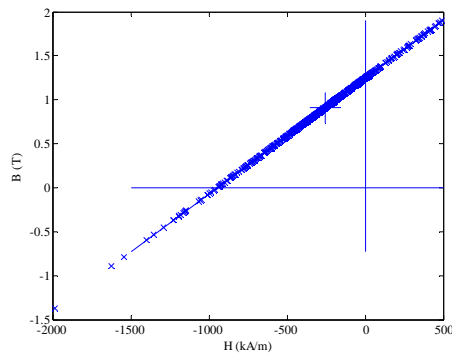
(d) Magnet\_6



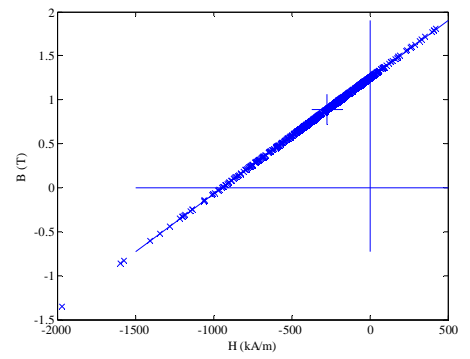
(e) Magnet\_3



(f) Magnet\_7



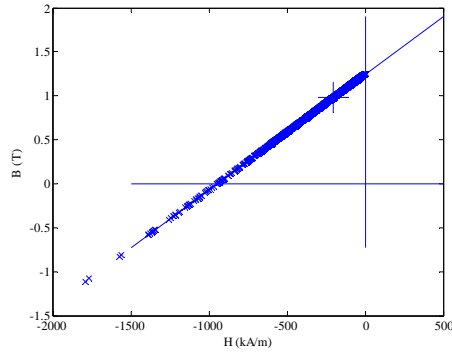
(h) Magnet\_4



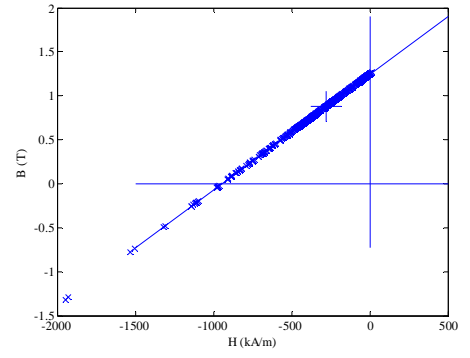
(g) Magnet\_8

Fig. II.43 Working points plots for embedded bread loaf designed 9-phase model with 1000A-t peak phase current at position A.

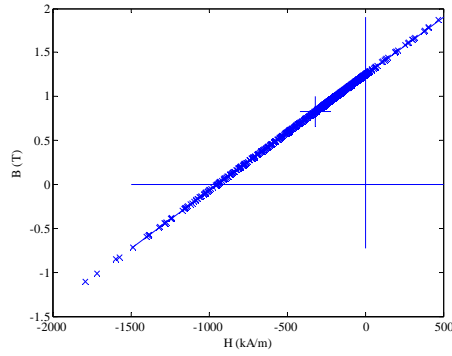




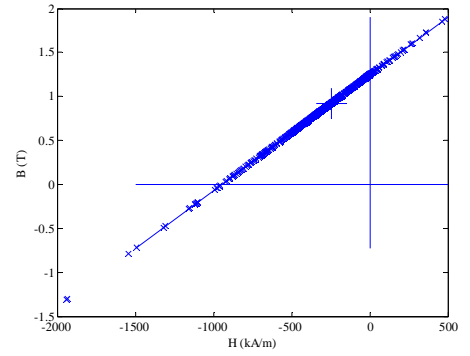
(a) Magnet\_1



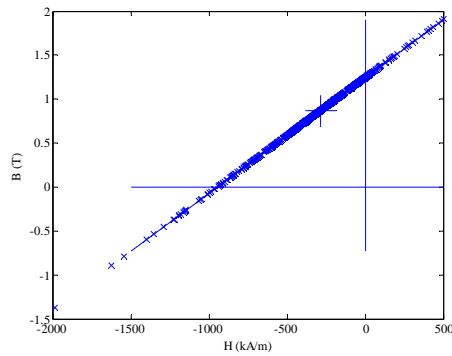
(b) Magnet\_5



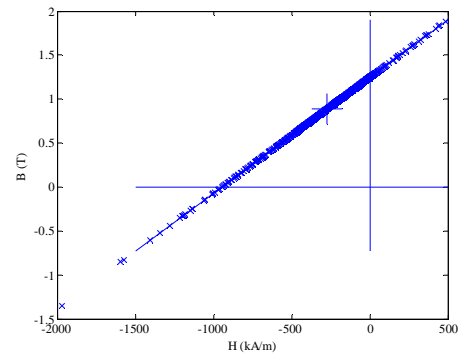
(c) Magnet\_2



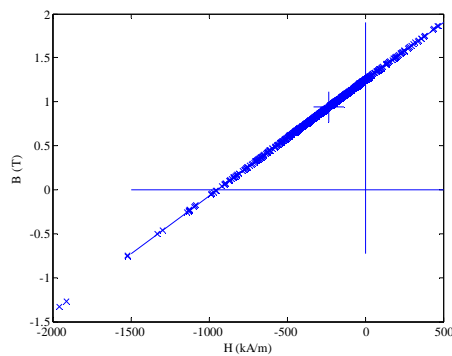
(d) Magnet\_6



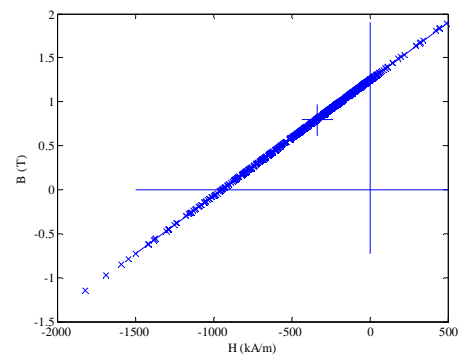
(e) Magnet\_3



(f) Magnet\_7

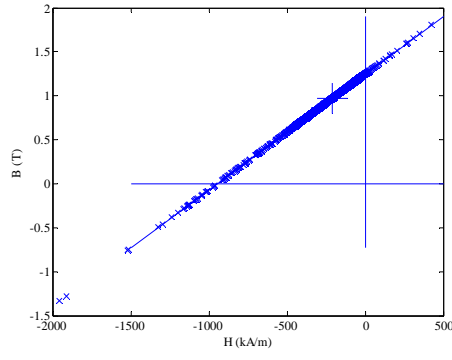


(h) Magnet\_4

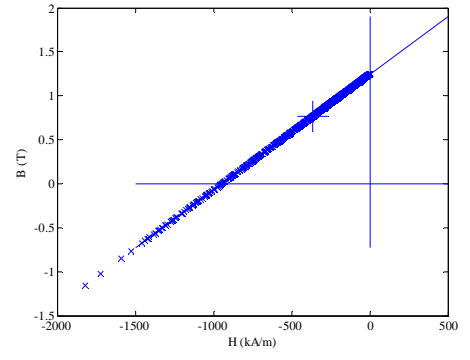


(g) Magnet\_8

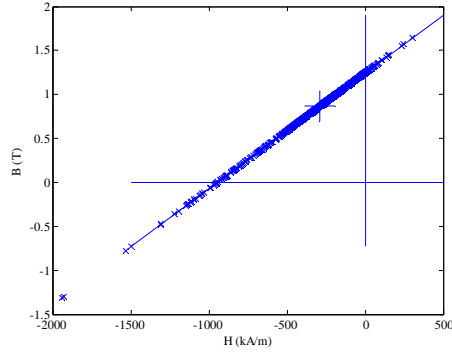
Fig. II.44 Working points plots for embedded bread loaf designed 9-phase model with 1000A-t peak phase current at position B.



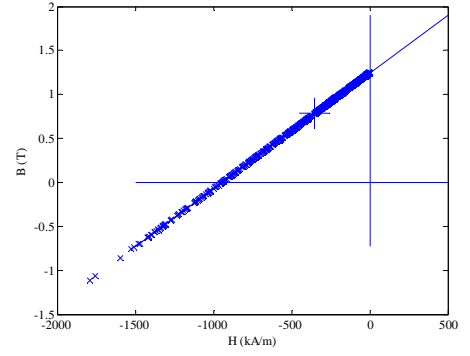
(a) Magnet\_1



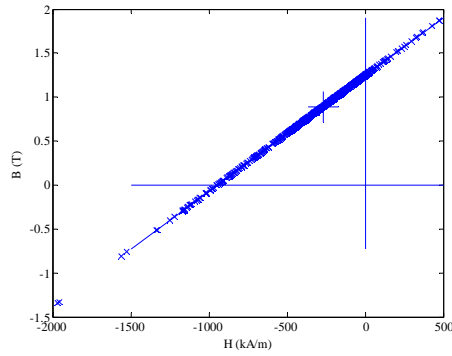
(b) Magnet\_5



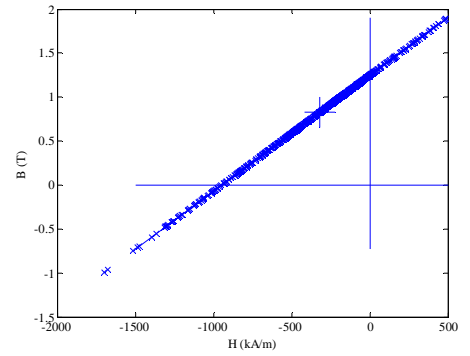
(c) Magnet\_2



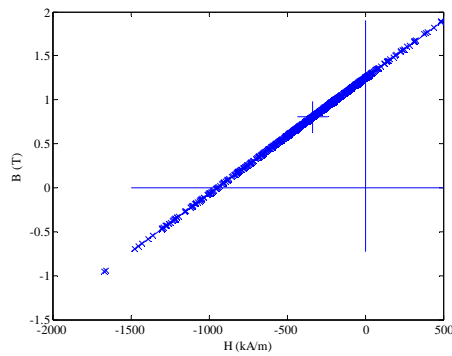
(d) Magnet\_6



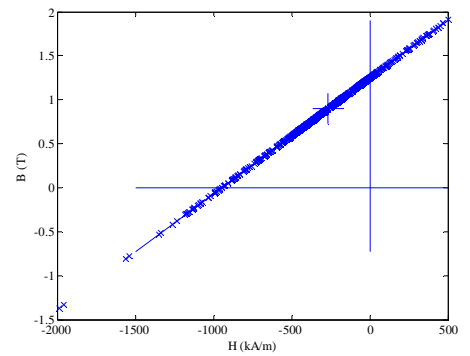
(e) Magnet\_3



(f) Magnet\_7

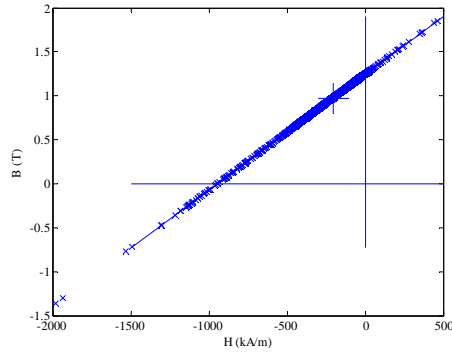


(h) Magnet\_4

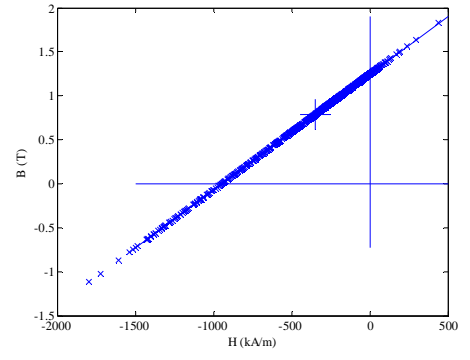


(g) Magnet\_8

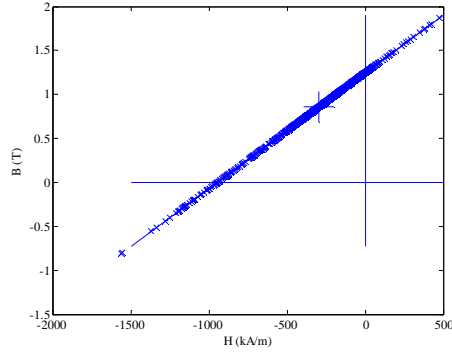
Fig. II.45 Working points plots for embedded bread loaf designed 9-phase model with 1000A-t peak phase current at position C.



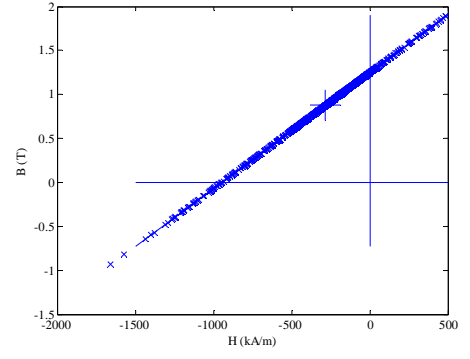
(a) Magnet\_1



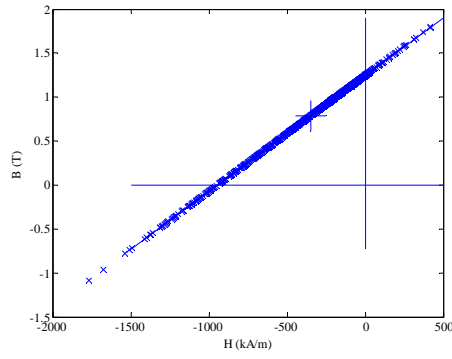
(b) Magnet\_5



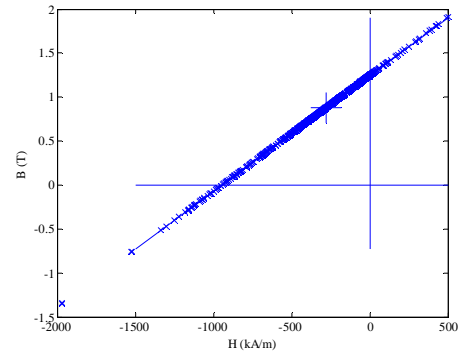
(c) Magnet\_2



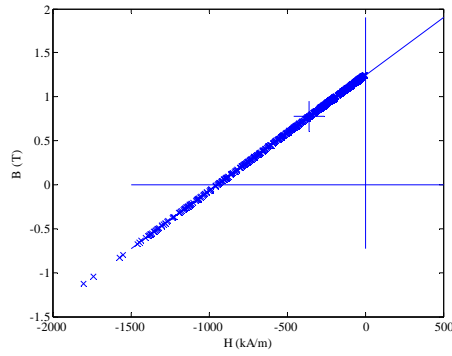
(d) Magnet\_6



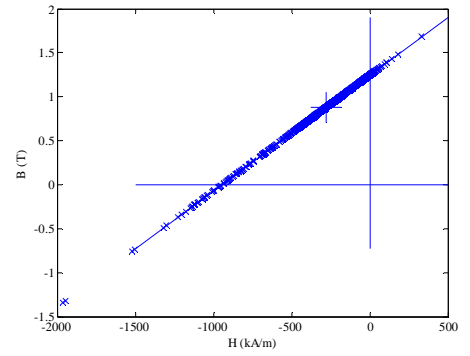
(e) Magnet\_3



(f) Magnet\_7

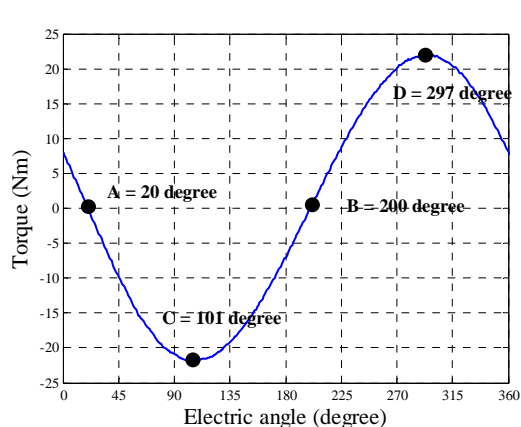


(h) Magnet\_4

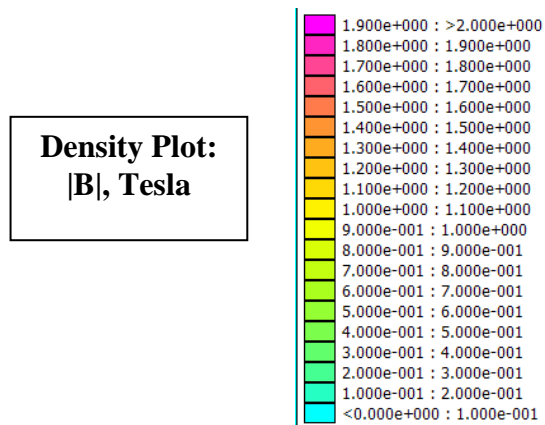


(g) Magnet\_8

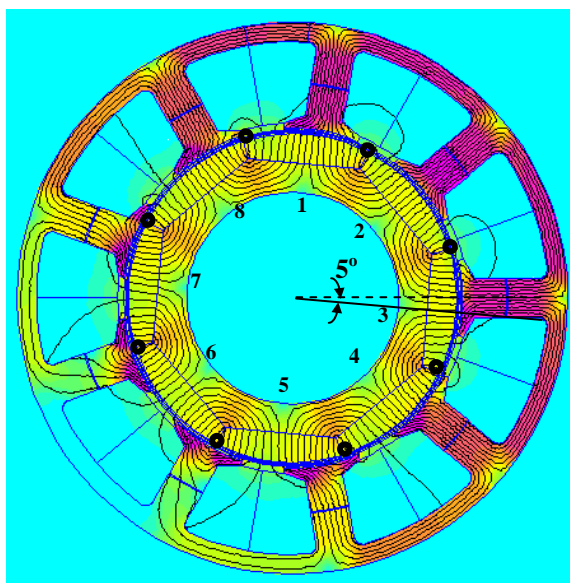
Fig. II.46 Working points plots for embedded bread loaf designed 9-phase model with 1000A-t peak phase current at position D.



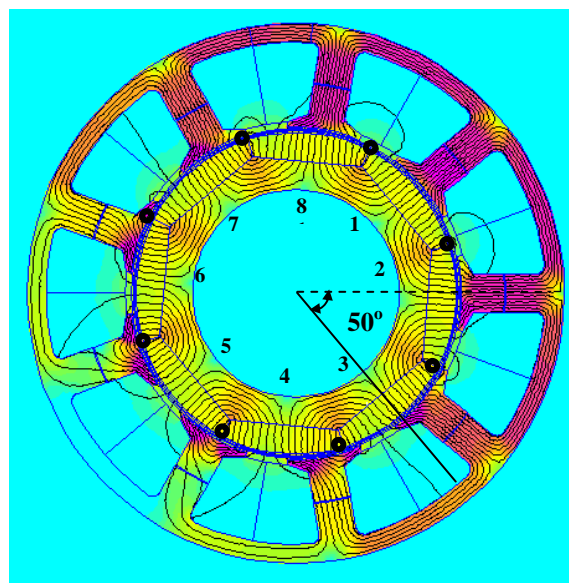
(a) Demagnetization checking position.



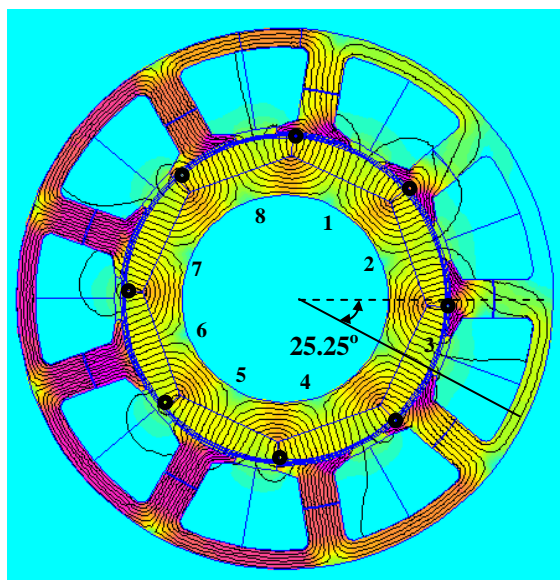
(b) Legend of field solution.



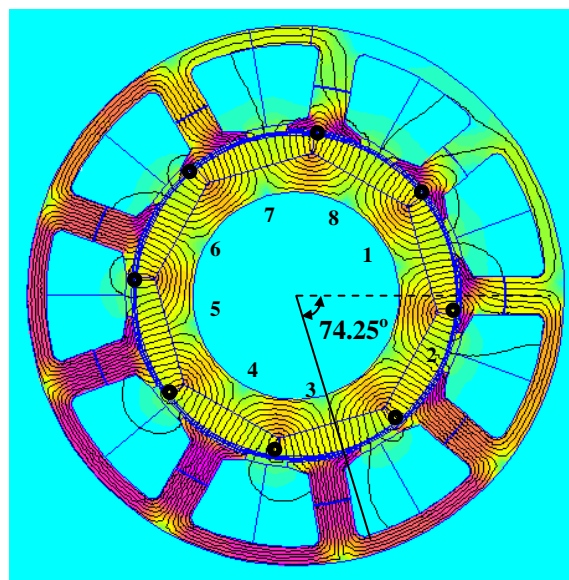
(c) FEA field solution for position A.



(d) FEA field solution for position B.

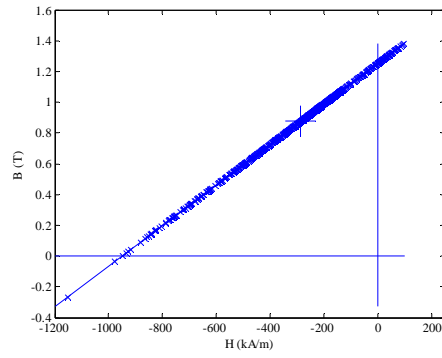


(e) FEA field solution for position C.

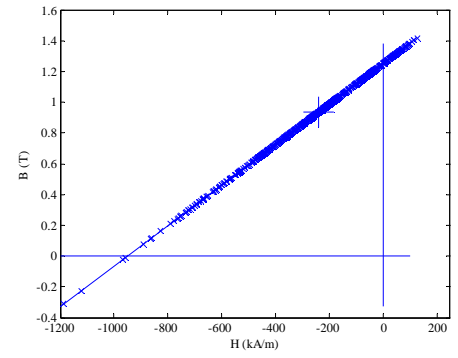


(f) FEA field solution for position D.

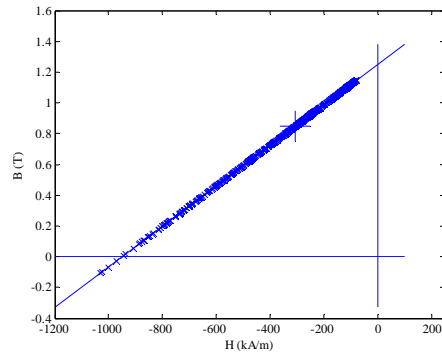
Fig. II.47 FEA field solutions of full pole-pitch and flat bottom design when peak phase current of 9-phase winding was 1000A-t, with possible demagnetized meshes circled.



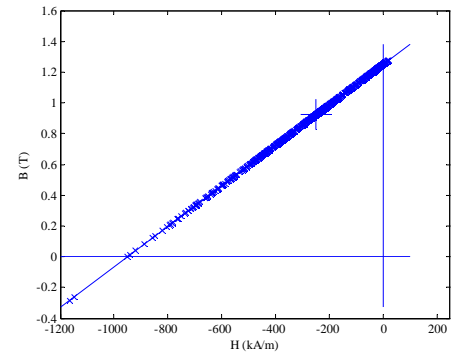
(a) Magnet\_1



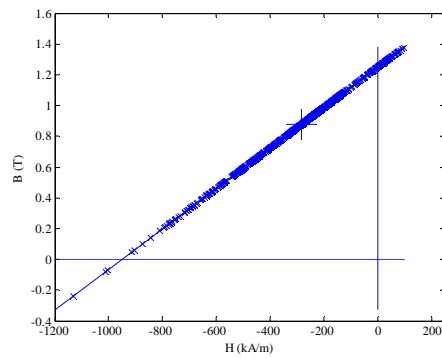
(b) Magnet\_5



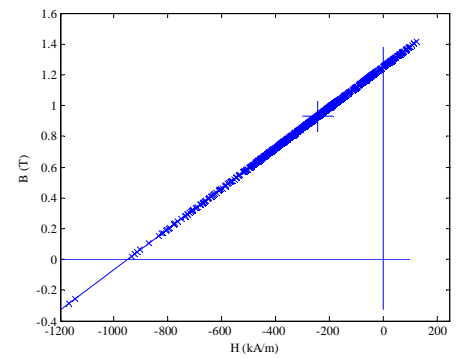
(c) Magnet\_2



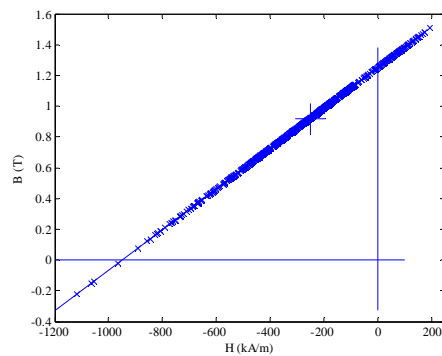
(d) Magnet\_6



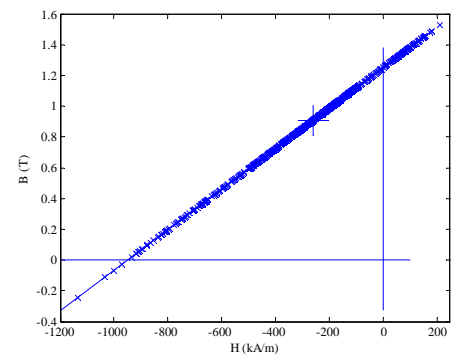
(e) Magnet\_3



(f) Magnet\_7

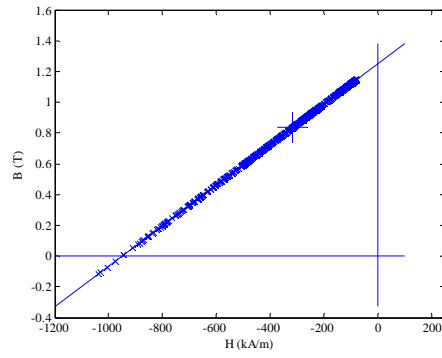


(h) Magnet\_4

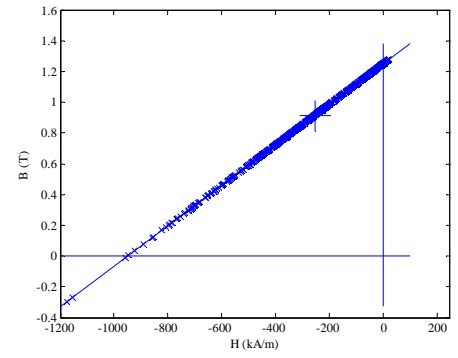


(g) Magnet\_8

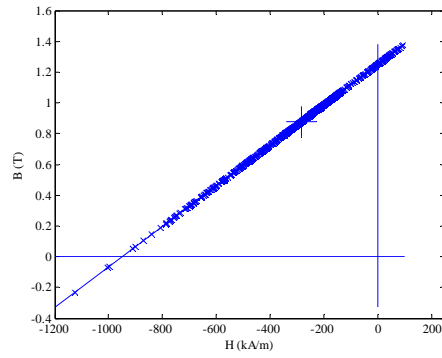
Fig. II.48 Working points plots for full pole-pitch and flat bottom designed 9-phase model with 1000A-t peak phase current at position A.



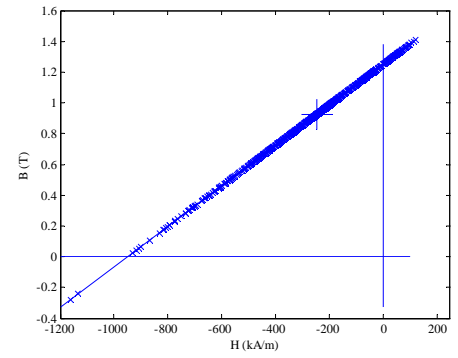
(a) Magnet\_1



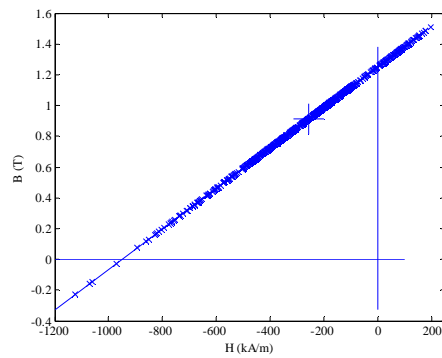
(b) Magnet\_5



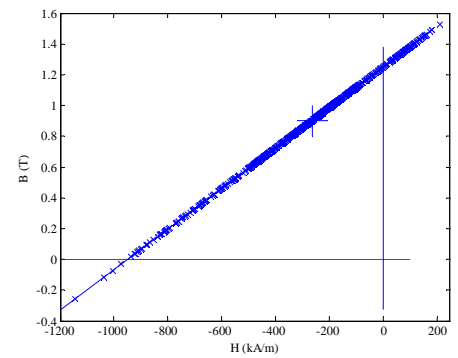
(c) Magnet\_2



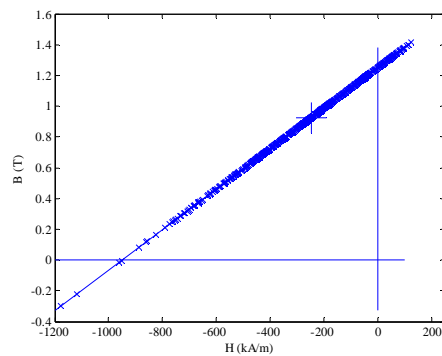
(d) Magnet\_6



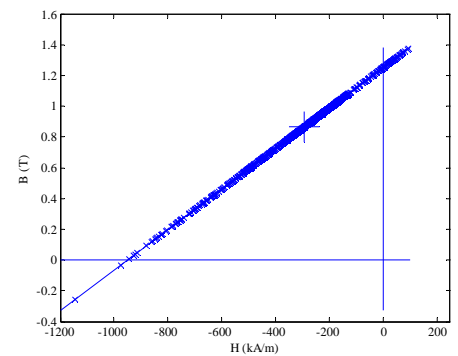
(e) Magnet\_3



(f) Magnet\_7

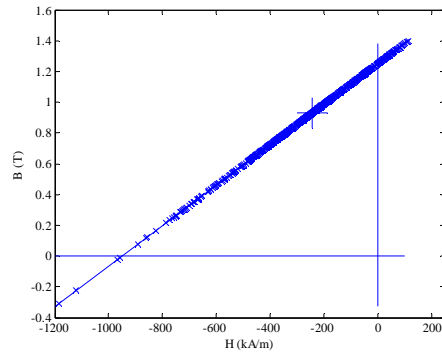


(h) Magnet\_4

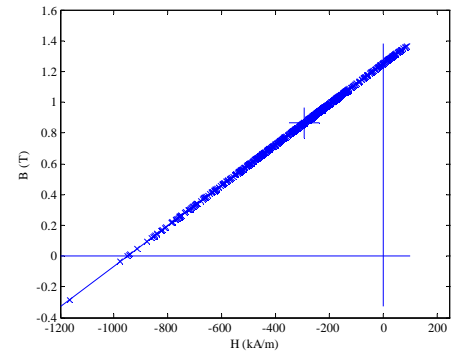


(g) Magnet\_8

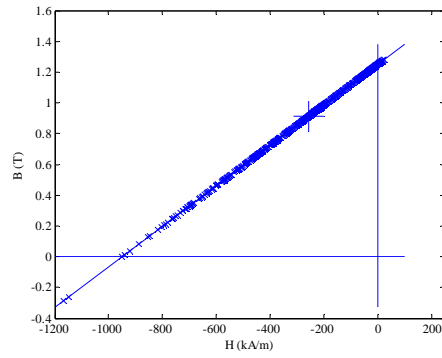
Fig. II.49 Working points plots for full pole-pitch and flat bottom designed 9-phase model with 1000A-t peak phase current at position B.



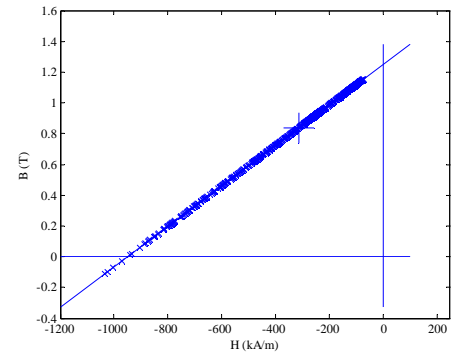
(a) Magnet\_1



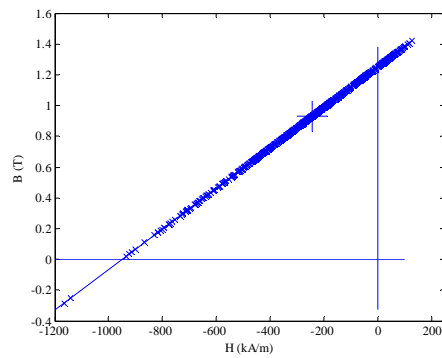
(b) Magnet\_5



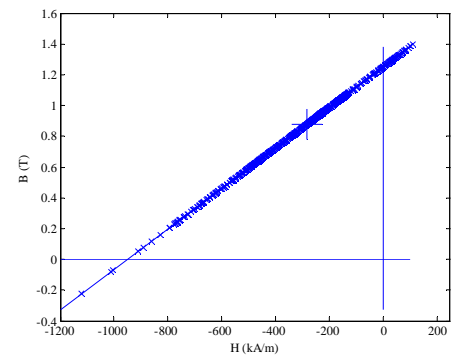
(c) Magnet\_2



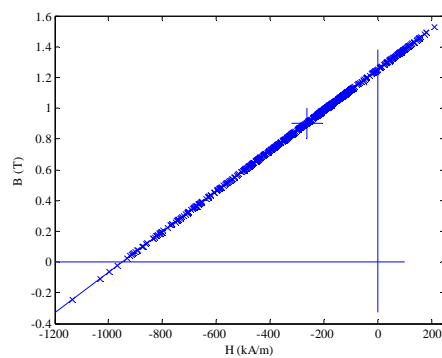
(d) Magnet\_6



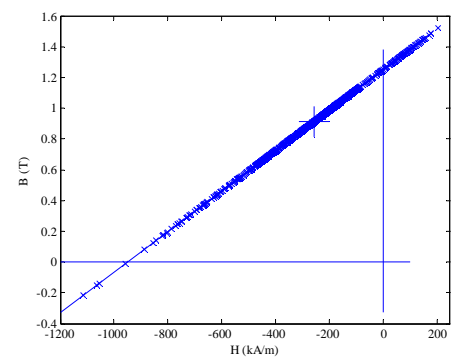
(e) Magnet\_3



(f) Magnet\_7

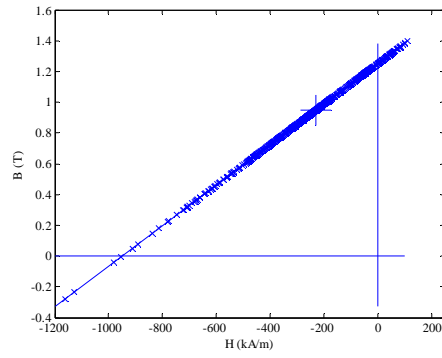


(h) Magnet\_4

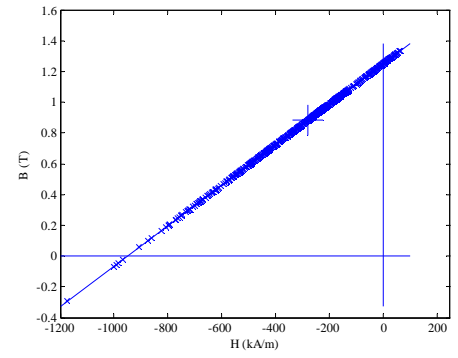


(g) Magnet\_8

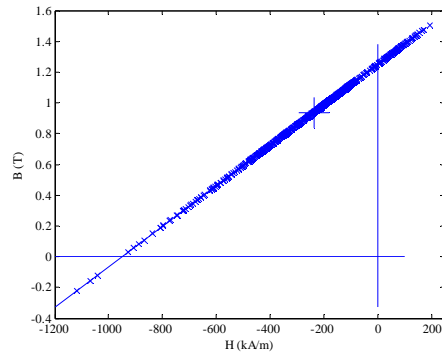
Fig. II.50 Working points plots for full pole-pitch and flat bottom designed 9-phase model with 1000A-t peak phase current at position C.



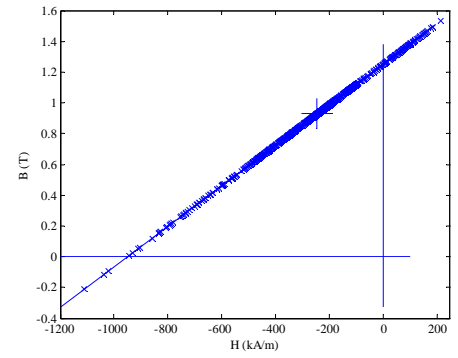
(a) Magnet\_1



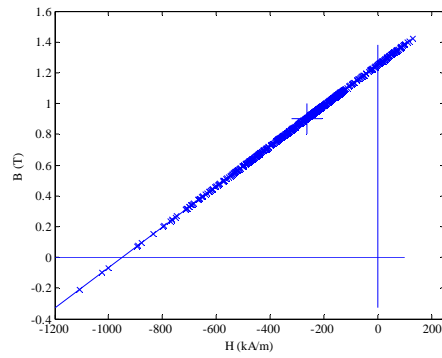
(b) Magnet\_5



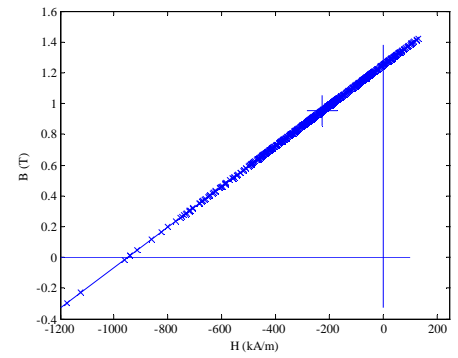
(c) Magnet\_2



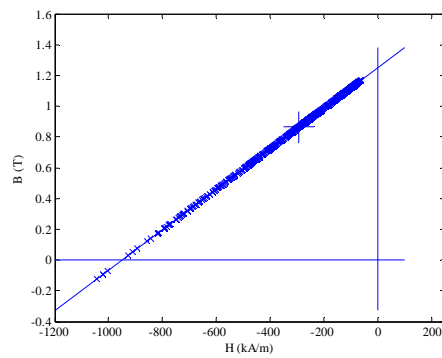
(d) Magnet\_6



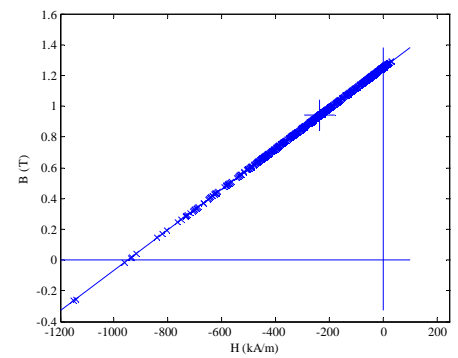
(e) Magnet\_3



(f) Magnet\_7



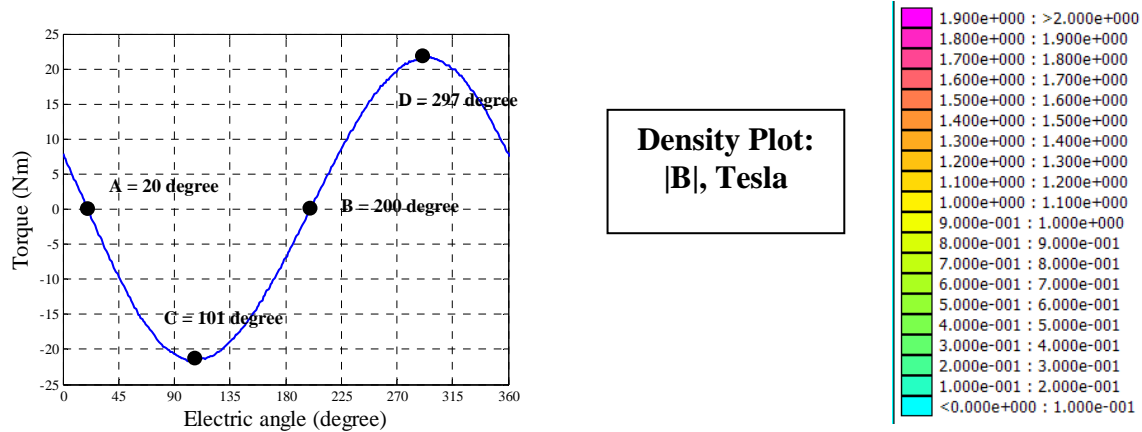
(h) Magnet\_4



(g) Magnet\_8

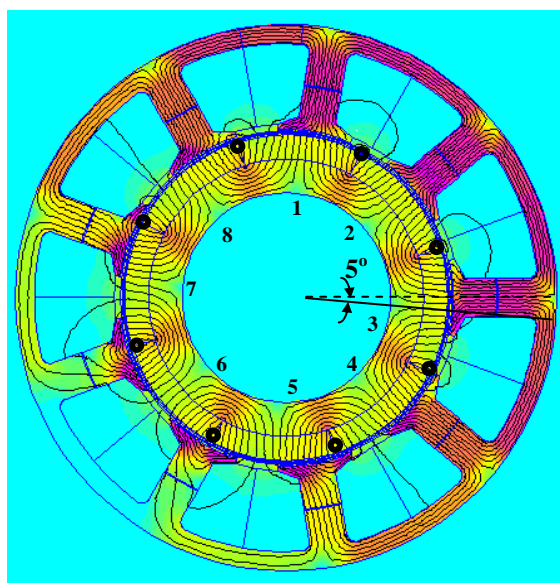
Fig. II.51 Working points plots for full pole-pitch and flat bottom designed 9-phase model with 1000A-t peak phase current at position D.



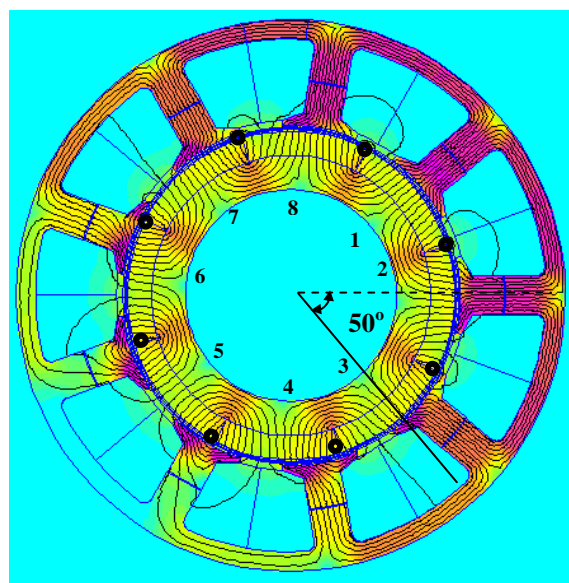


(a) Demagnetization checking position.

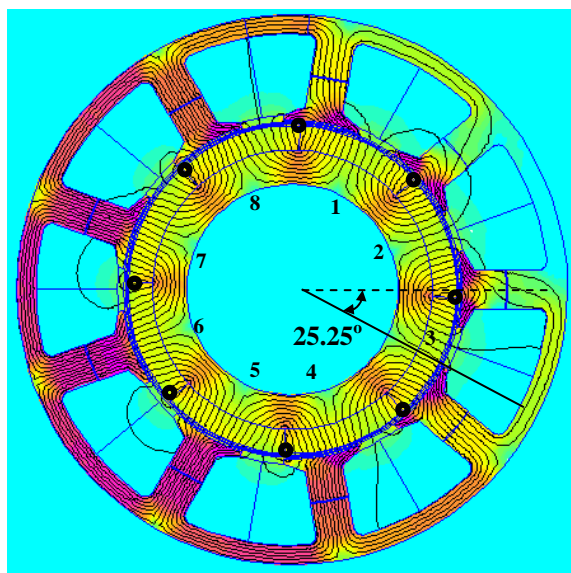
(b) Legend of field solution.



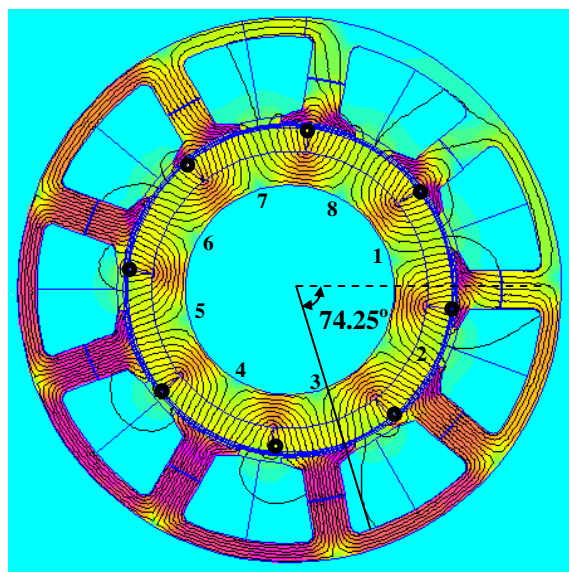
(c) FEA field solution for position A.



(d) FEA field solution for position B.

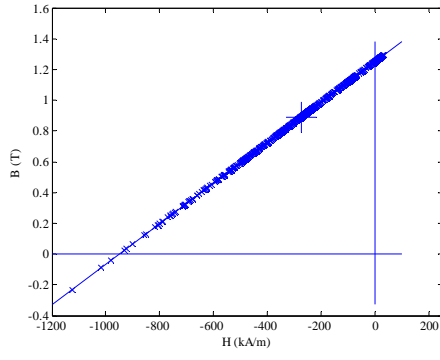


(e) FEA field solution for position C.

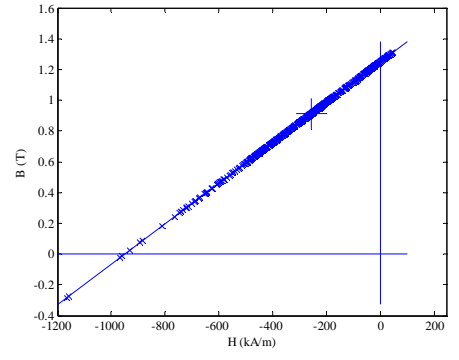


(f) FEA field solution for position D.

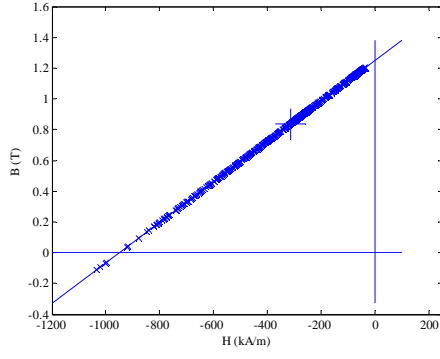
Fig. II.52 FEA field solutions of full pole-pitch arc design when peak phase current of 9-phase winding was 1000A-t, with possible demagnetized meshes circled.



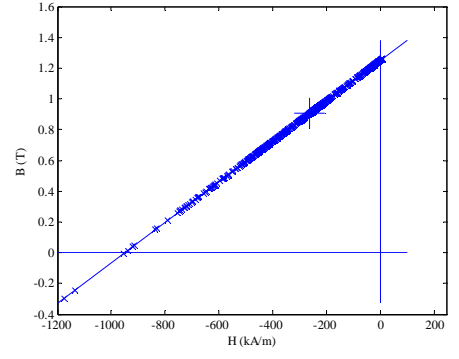
(a) Magnet\_1



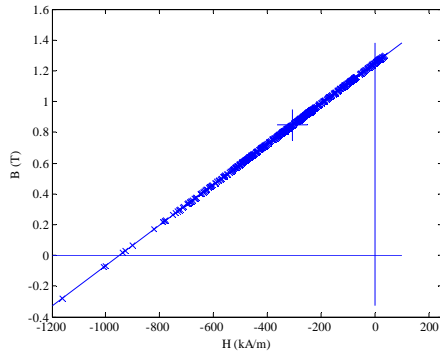
(b) Magnet\_5



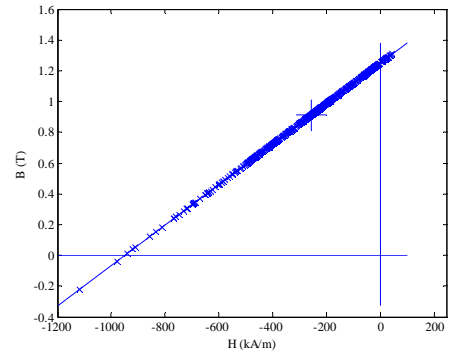
(c) Magnet\_2



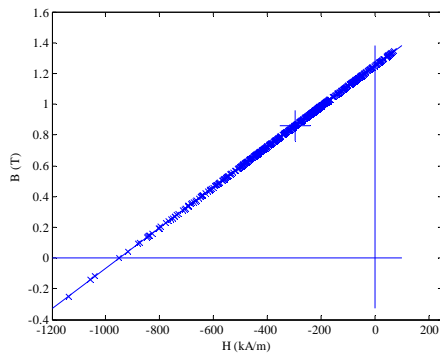
(d) Magnet\_6



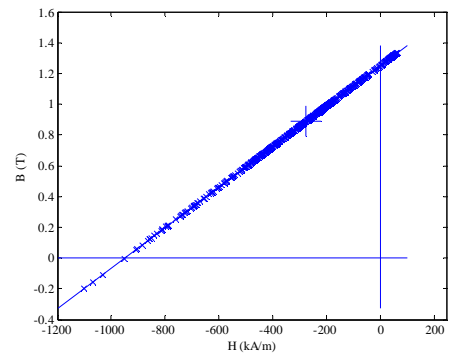
(e) Magnet\_3



(f) Magnet\_7

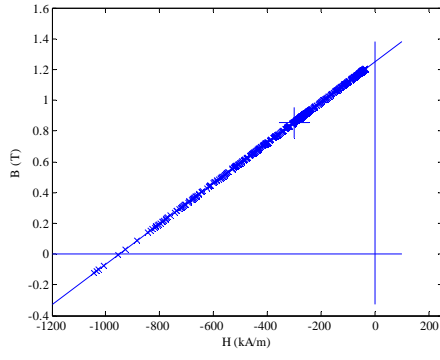


(h) Magnet\_4

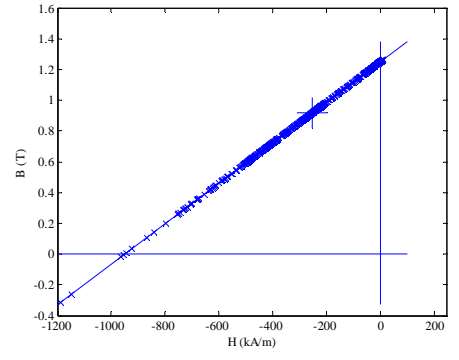


(g) Magnet\_8

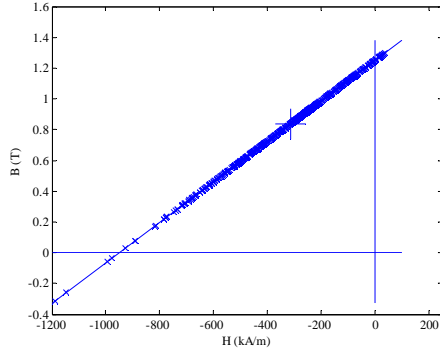
Fig. II.53 Working points plots for full pole-pitch arc designed 9-phase model with 1000A-t peak phase current at position A.



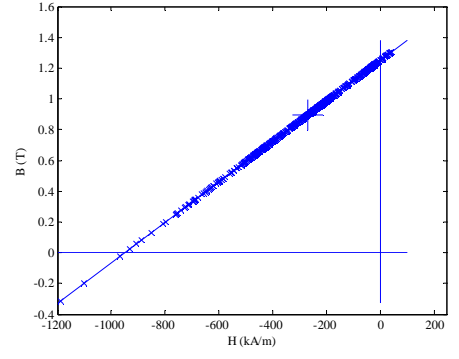
(a) Magnet\_1



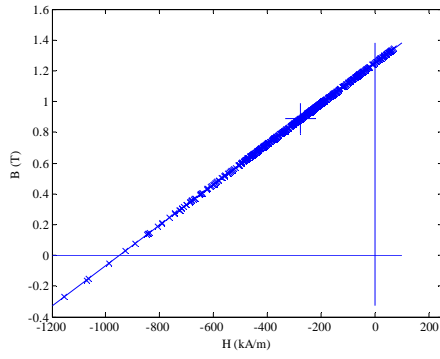
(b) Magnet\_5



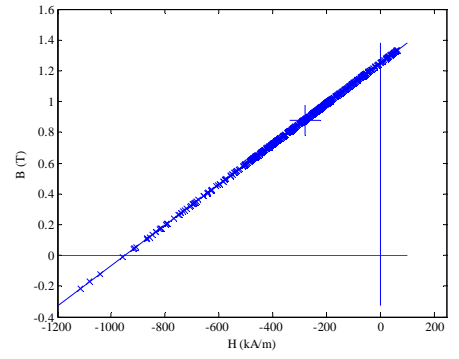
(c) Magnet\_2



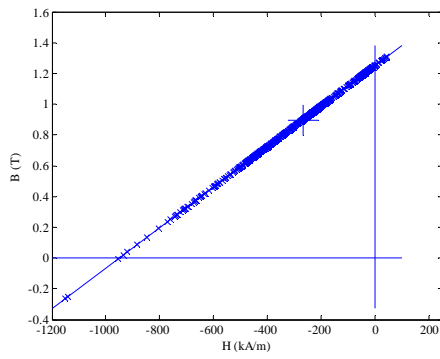
(d) Magnet\_6



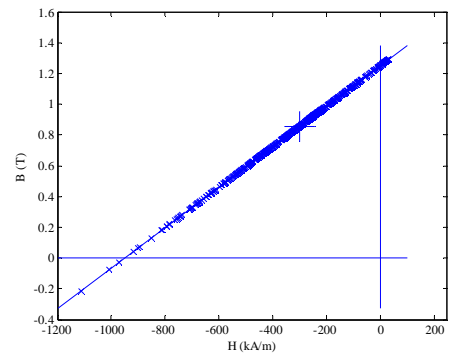
(e) Magnet\_3



(f) Magnet\_7

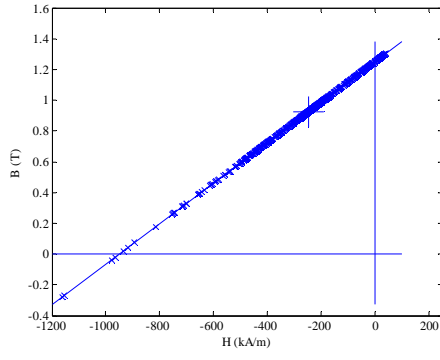


(h) Magnet\_4

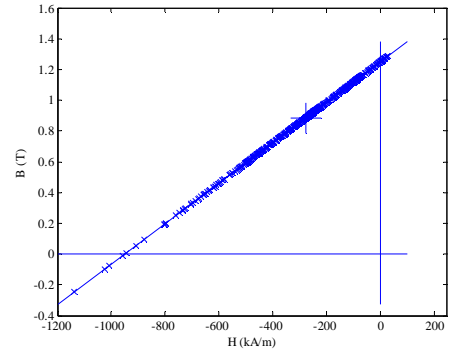


(g) Magnet\_8

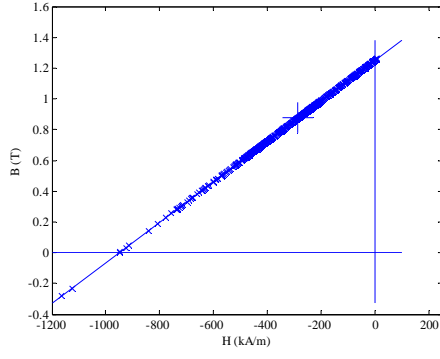
Fig. II.54 Working points plots for full pole-pitch arc designed 9-phase model with 1000A-t peak phase current at position B.



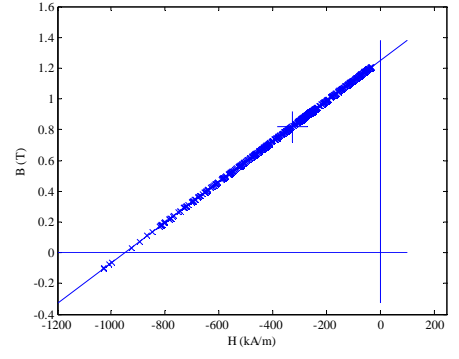
(a) Magnet\_1



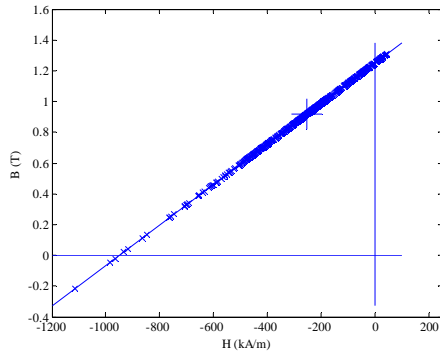
(b) Magnet\_5



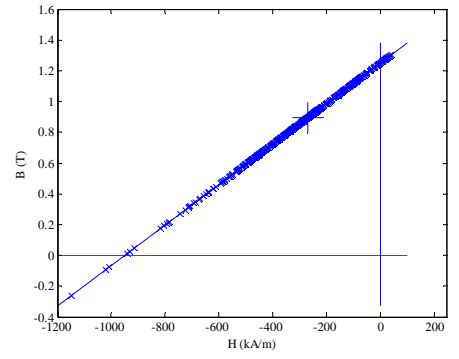
(c) Magnet\_2



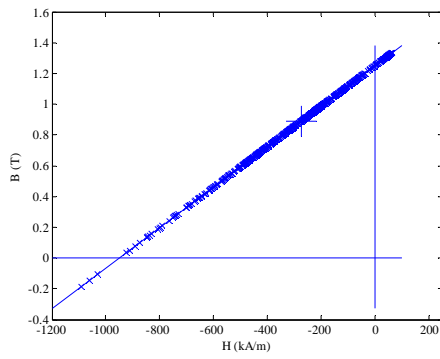
(d) Magnet\_6



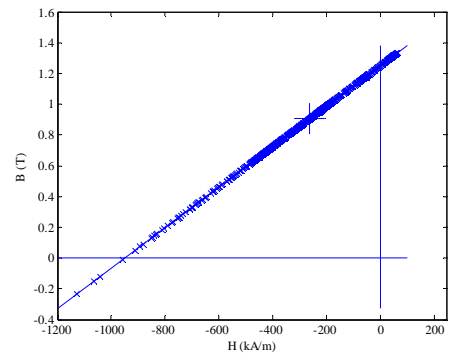
(e) Magnet\_3



(f) Magnet\_7

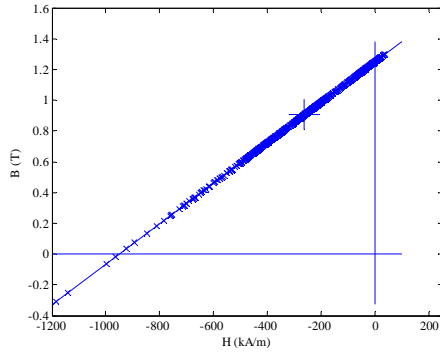


(h) Magnet\_4

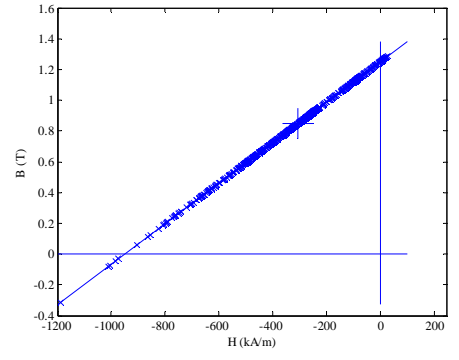


(g) Magnet\_8

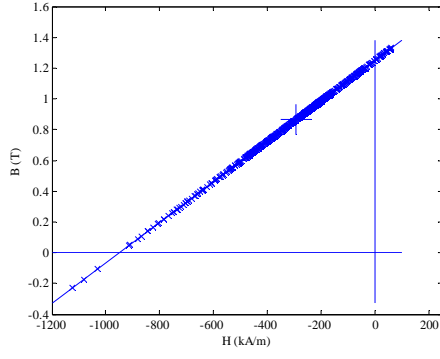
Fig. II.55 Working points plots for full pole-pitch arc designed 9-phase model with 1000A-t peak phase current at position C.



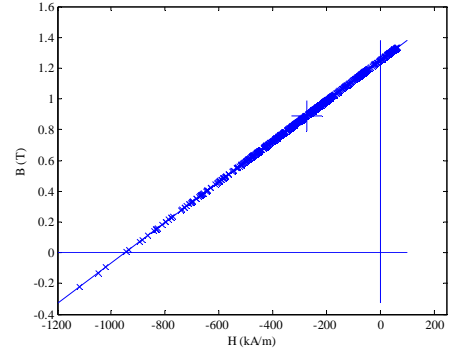
(a) Magnet\_1



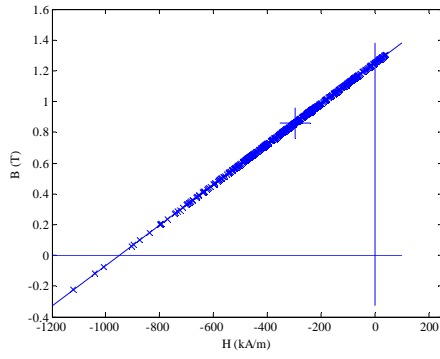
(b) Magnet\_5



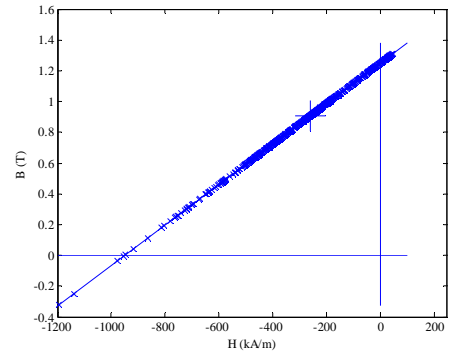
(c) Magnet\_2



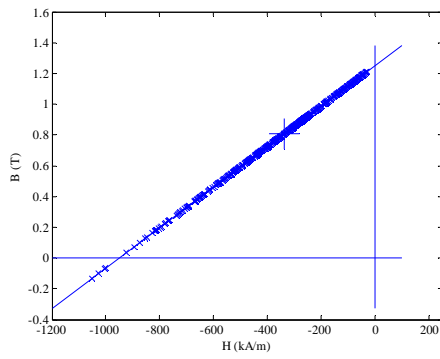
(d) Magnet\_6



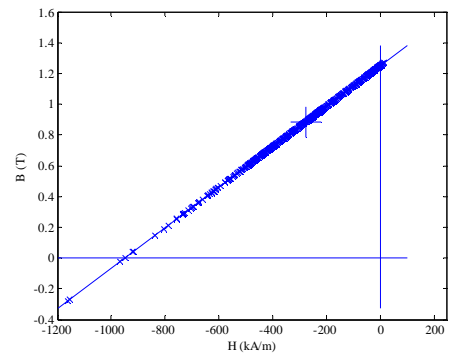
(e) Magnet\_3



(f) Magnet\_7

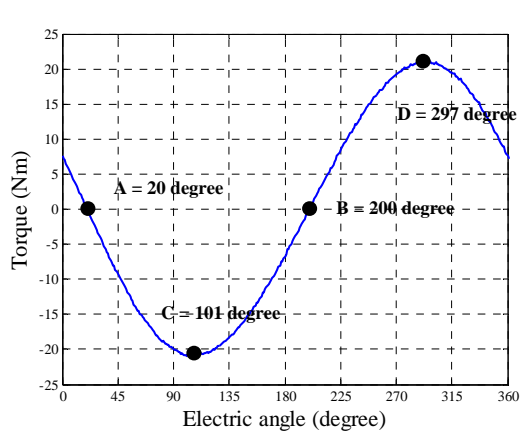


(h) Magnet\_4

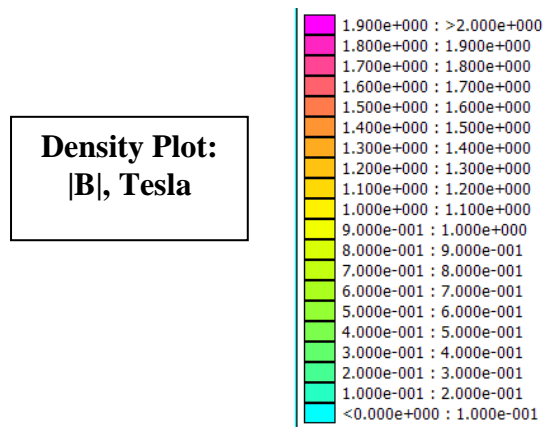


(g) Magnet\_8

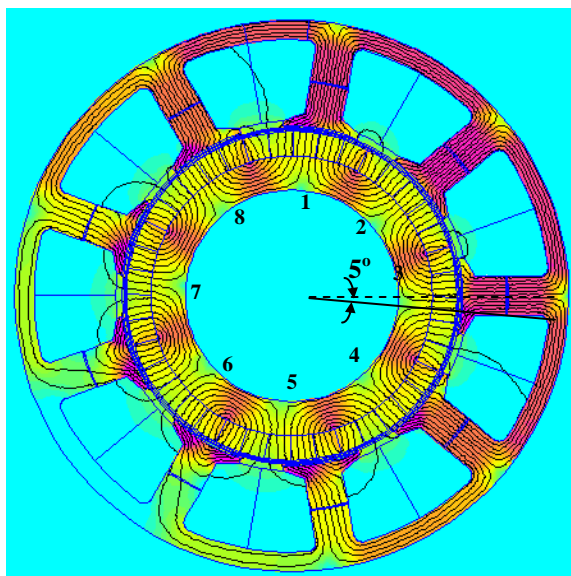
Fig. II.56 Working points plots for full pole-pitch arc designed 9-phase model with 1000A-t peak phase current at position D.



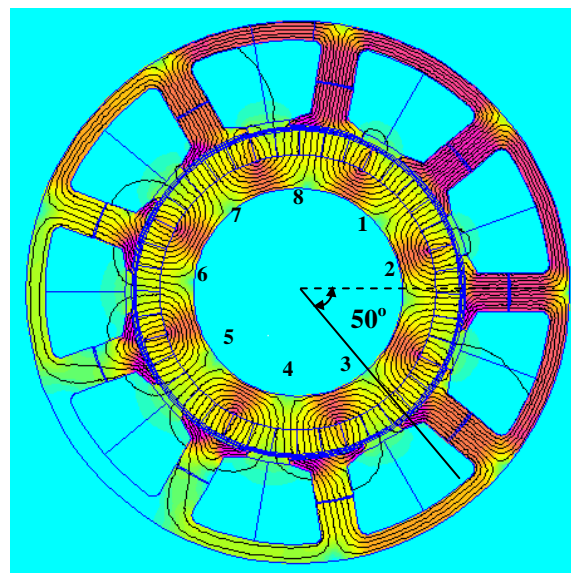
(a) Demagnetization checking position.



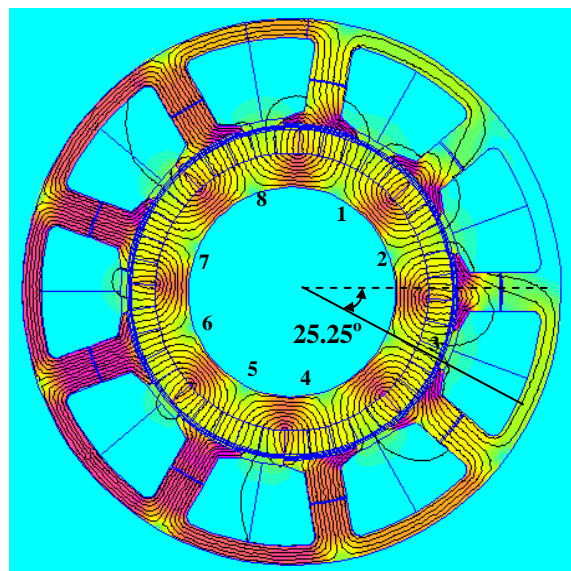
(b) Legend of field solution.



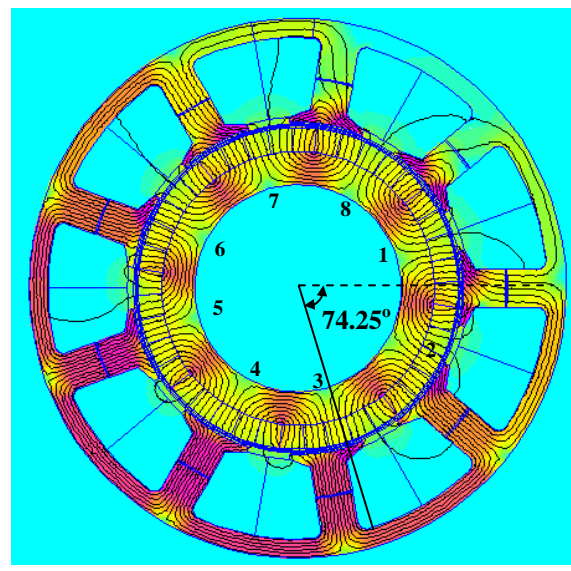
(c) FEA field solution for position A.



(d) FEA field solution for position B.

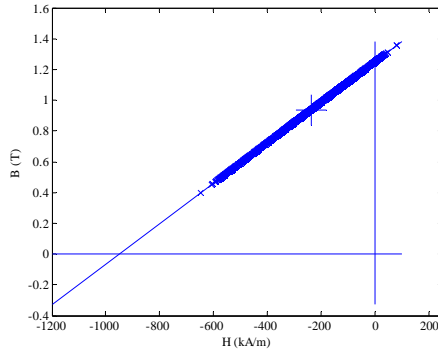


(e) FEA field solution for position C.

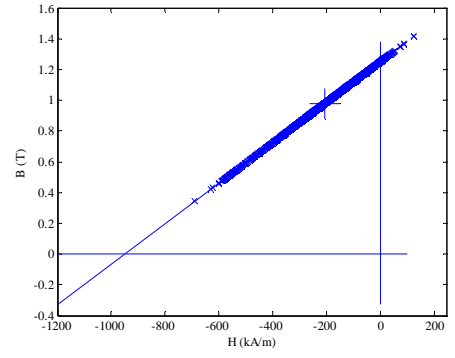


(f) FEA field solution for position D.

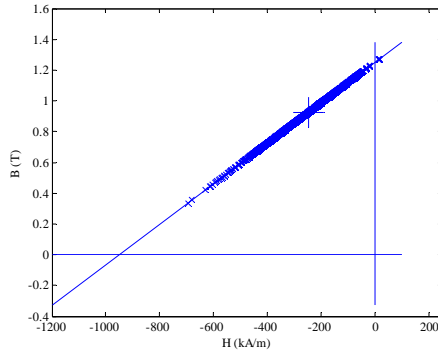
Fig. II.57 FEA field solutions of full pole-pitch pseudo-radial magnetization design when peak phase current of 9-phase winding was 1000A-t, with possible demagnetized meshes circled.



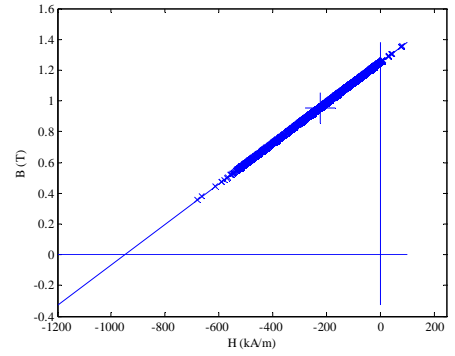
(a) Magnet\_1



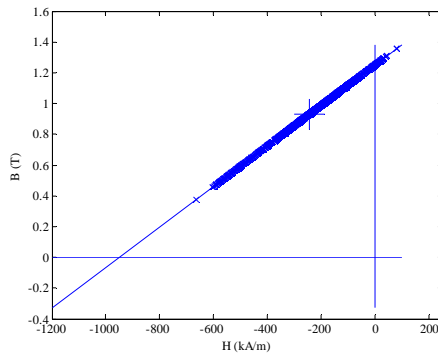
(b) Magnet\_5



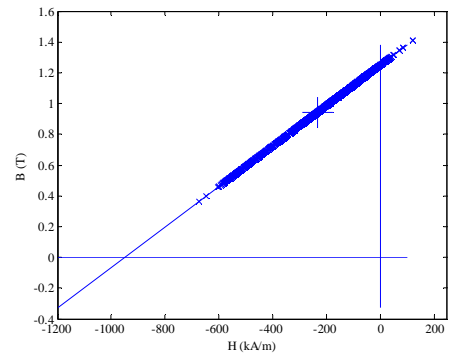
(c) Magnet\_2



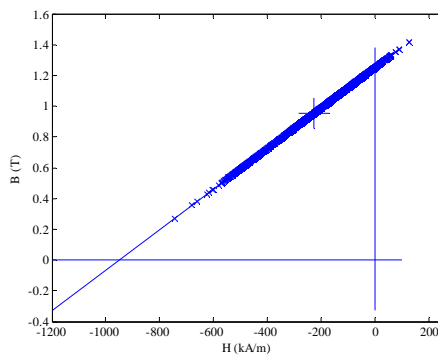
(d) Magnet\_6



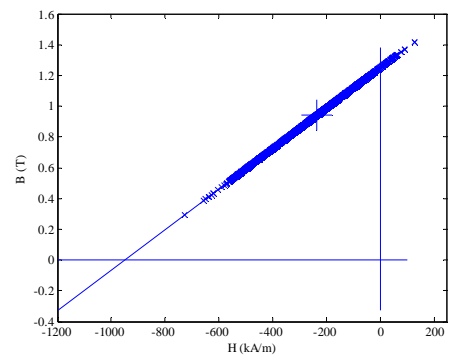
(e) Magnet\_3



(f) Magnet\_7

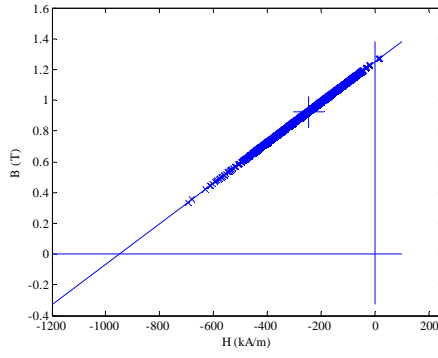


(h) Magnet\_4

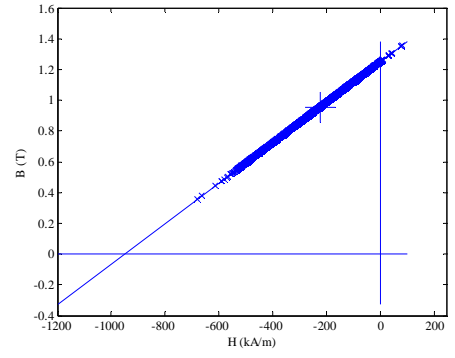


(g) Magnet\_8

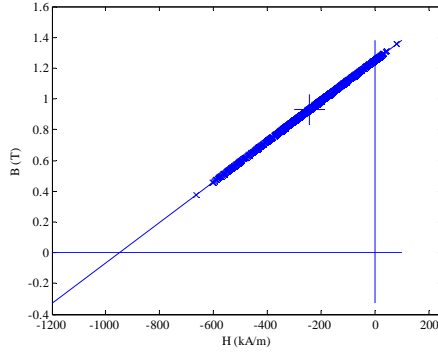
Fig. II.58 Working points plots for full pole-pitch pseudo-radial magnetization designed 9-phase model with 1000A-t peak phase current at position A.



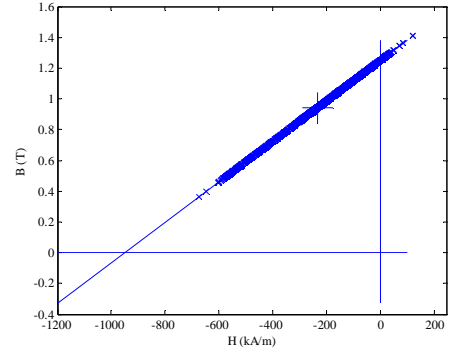
(a) Magnet\_1



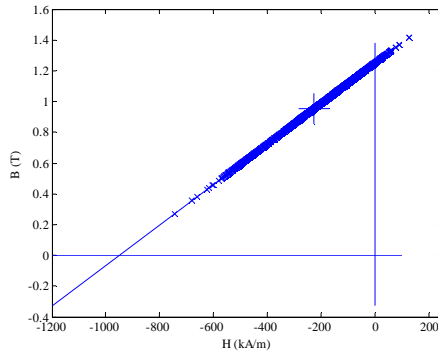
(b) Magnet\_5



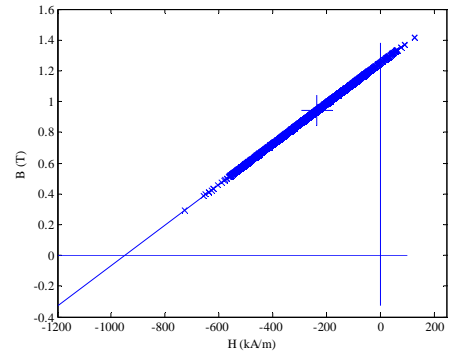
(c) Magnet\_2



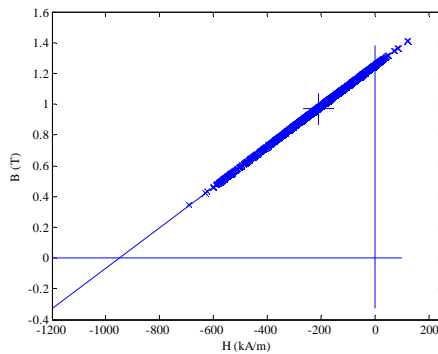
(d) Magnet\_6



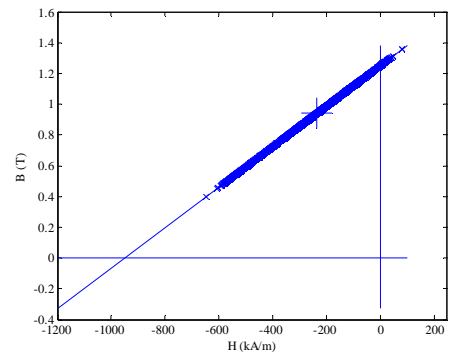
(e) Magnet\_3



(f) Magnet\_7



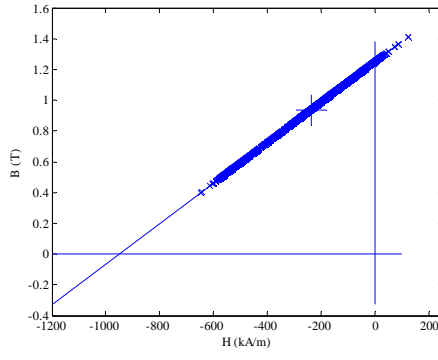
(h) Magnet\_4



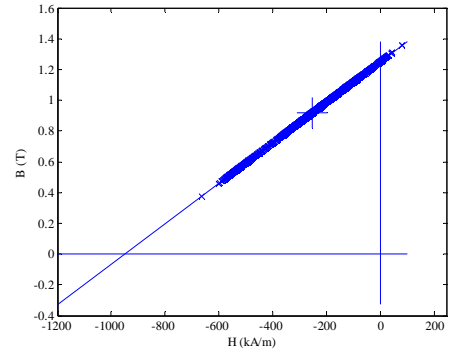
(g) Magnet\_8

Fig. II.59 Working points plots for full pole-pitch pseudo-radial magnetization designed 9-phase model with 1000A-t peak phase current at position B.

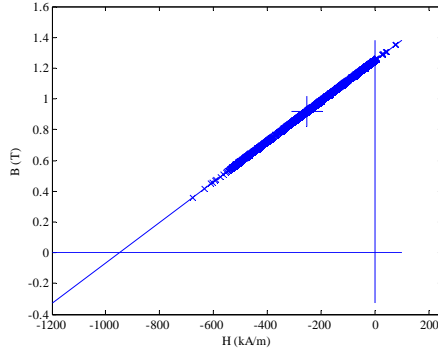




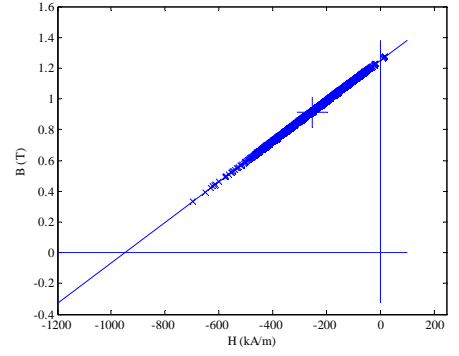
(a) Magnet\_1



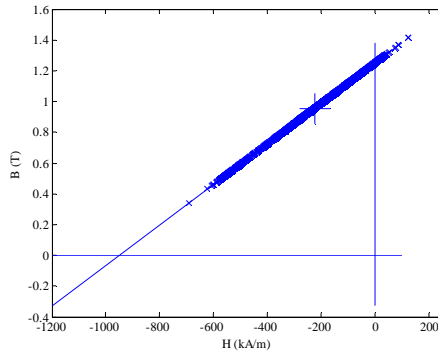
(b) Magnet\_5



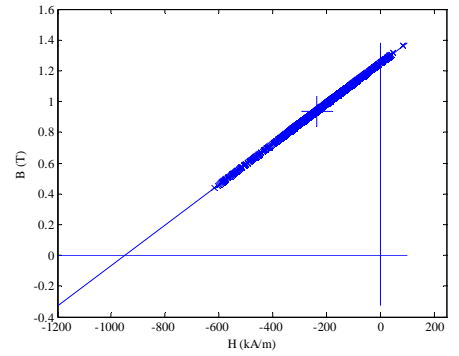
(c) Magnet\_2



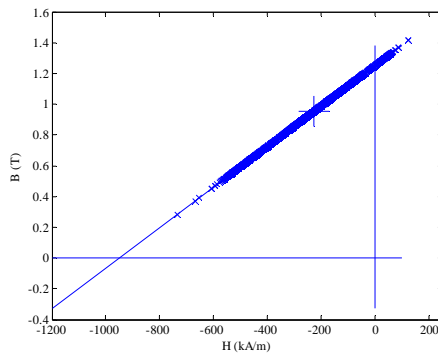
(d) Magnet\_6



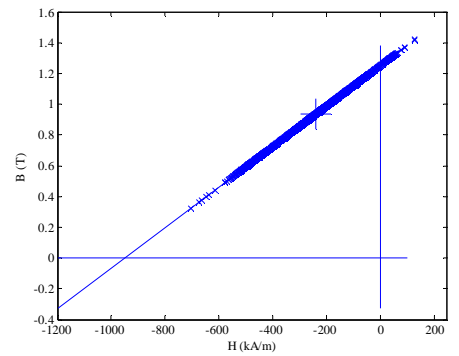
(e) Magnet\_3



(f) Magnet\_7

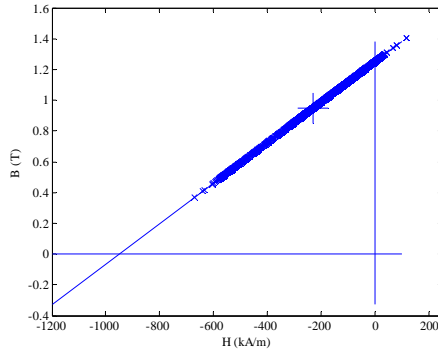


(h) Magnet\_4

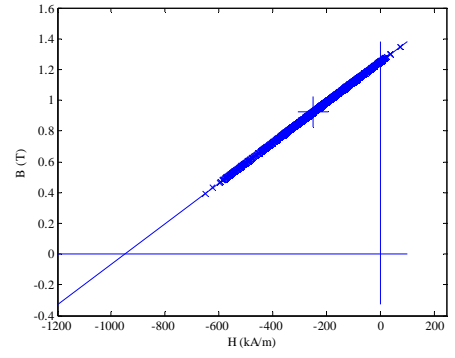


(g) Magnet\_8

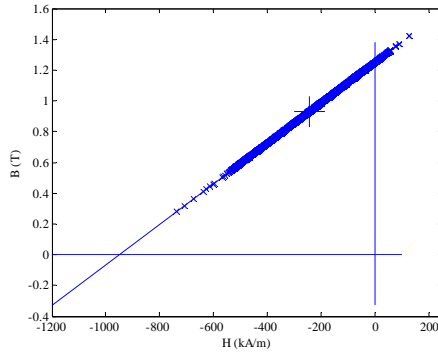
Fig. II.60 Working points plots for full pole-pitch pseudo-radial magnetization designed 9-phase model with 1000A-t peak phase current at position C.



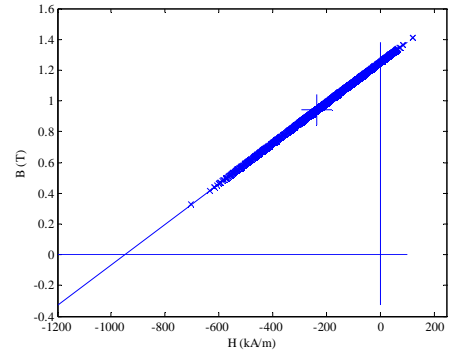
(a) Magnet\_1



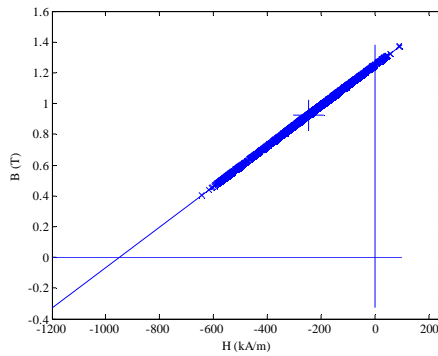
(b) Magnet\_5



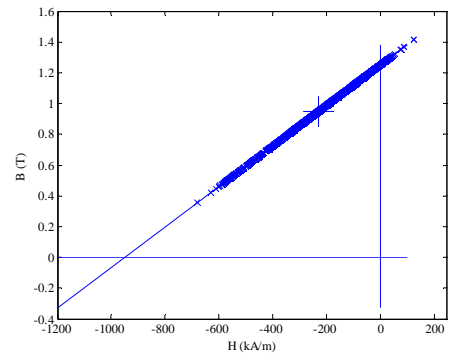
(c) Magnet\_2



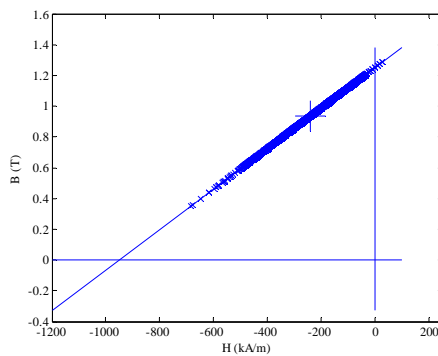
(d) Magnet\_6



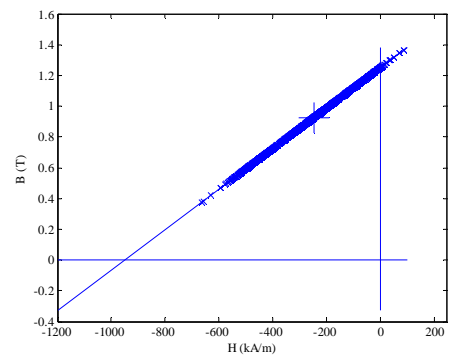
(e) Magnet\_3



(f) Magnet\_7



(h) Magnet\_4



(g) Magnet\_8

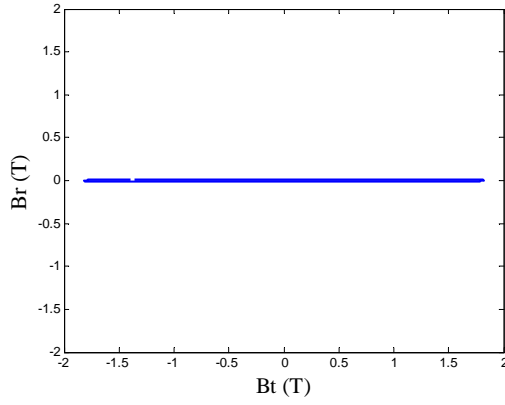
Fig. II.61 Working points plots for full pole-pitch pseudo-radial magnetization designed 9-phase model with 1000A-t peak phase current at position D.

## Appendix III

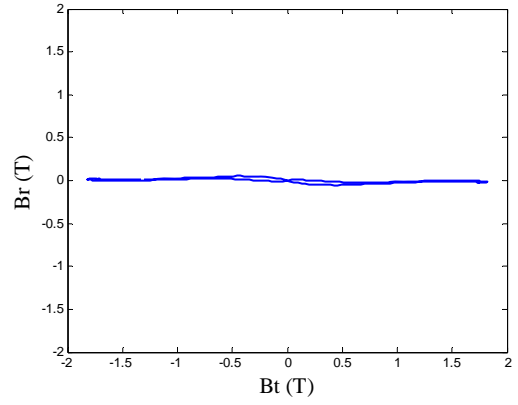
### Flux Results

#### III.1 Plots for rotational component checking and defining rotational loss parameters.

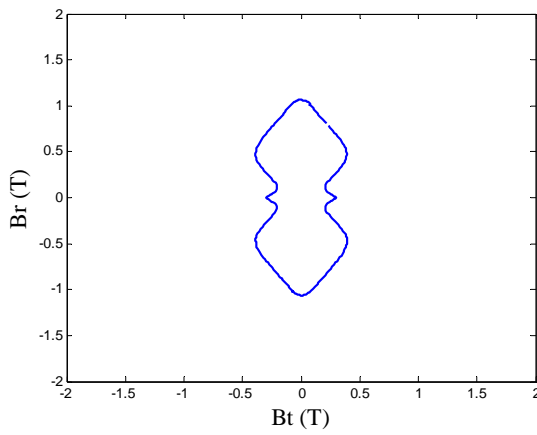
Flux rotation check. Open circuit.



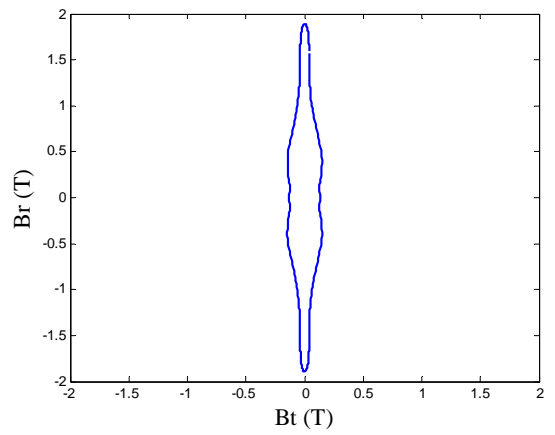
(a) Testing block 1



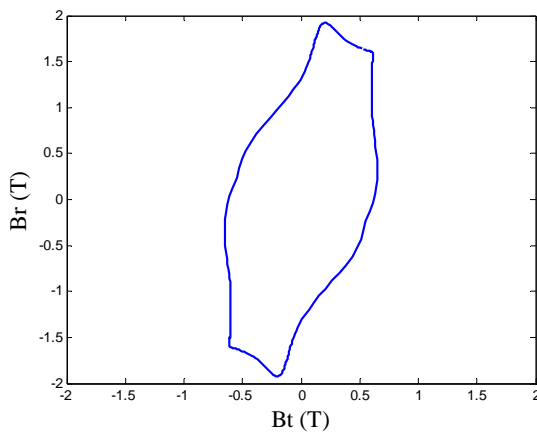
(b) Testing block 2



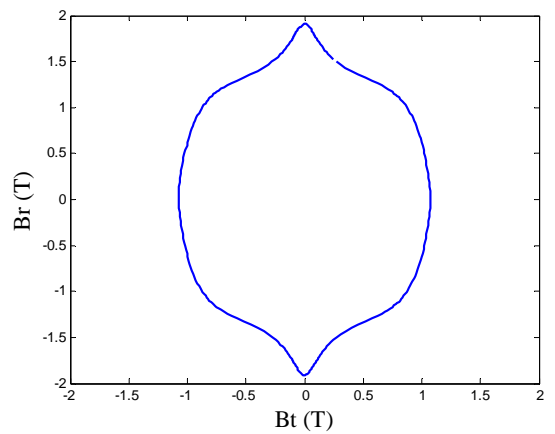
(c) Testing block 3



(d) Testing block 4

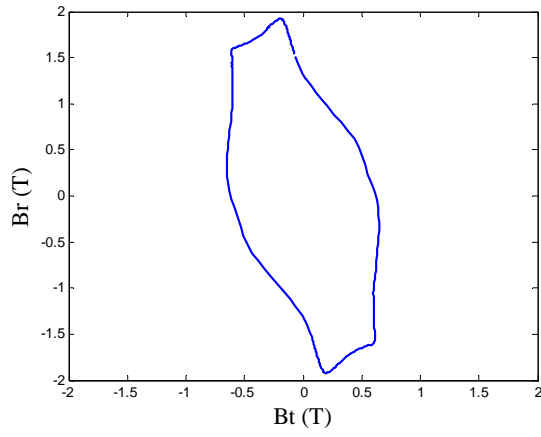


(e) Testing block 5

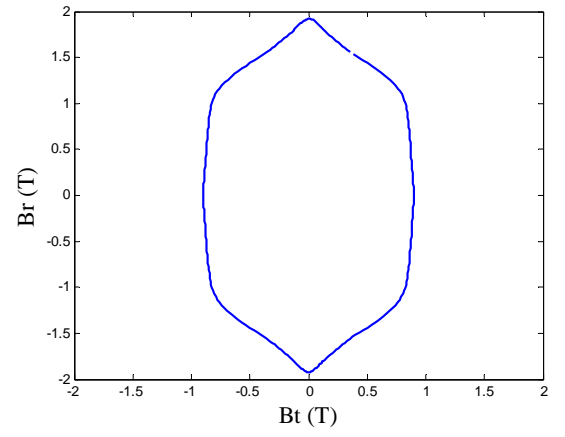


(f) Testing block 6

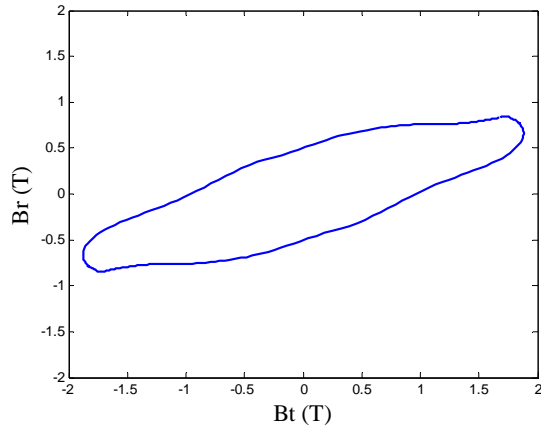
Fig. III.1 Plots of  $Br$  versus  $Bt$ .



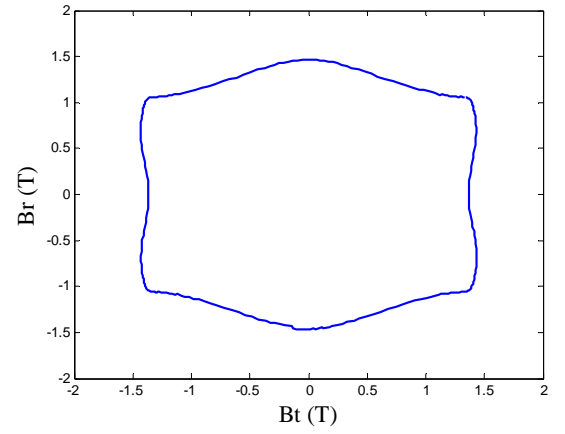
(a) Testing block 7



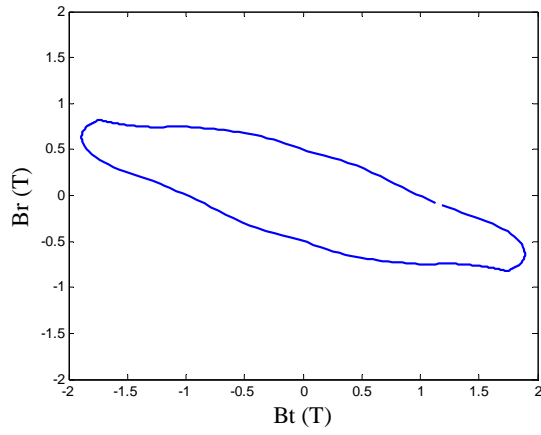
(b) Testing block 8



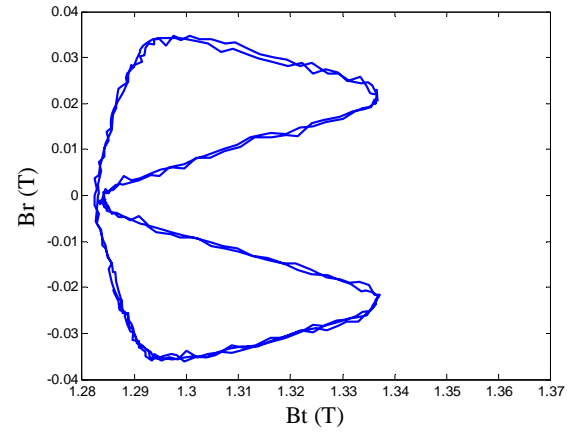
(c) Testing block 9



(d) Testing block 10

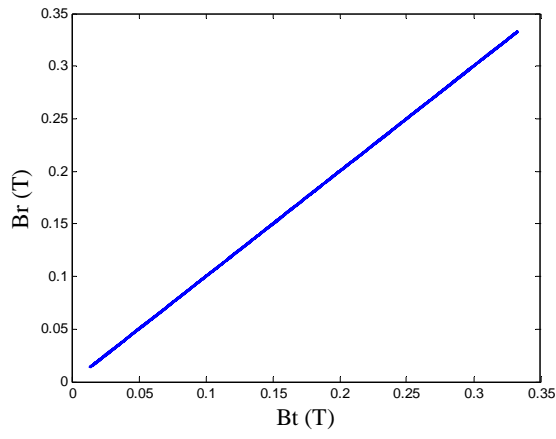


(e) Testing block 11

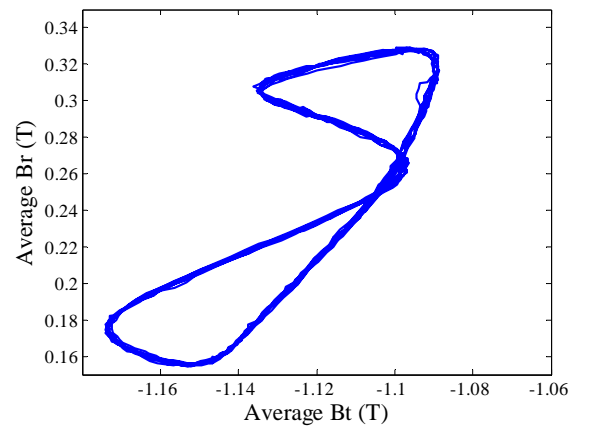


(f) Testing block 12

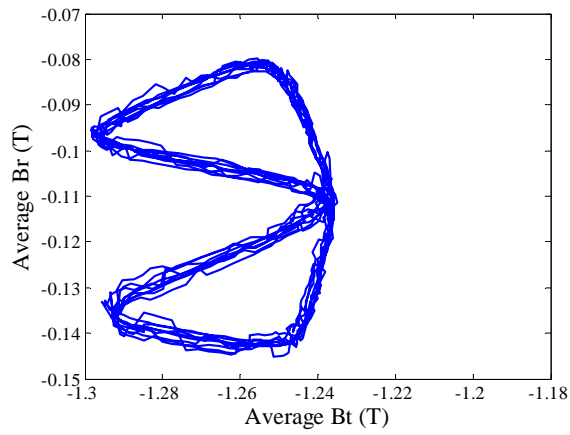
Fig. III.2 Plots of Br versus Bt.



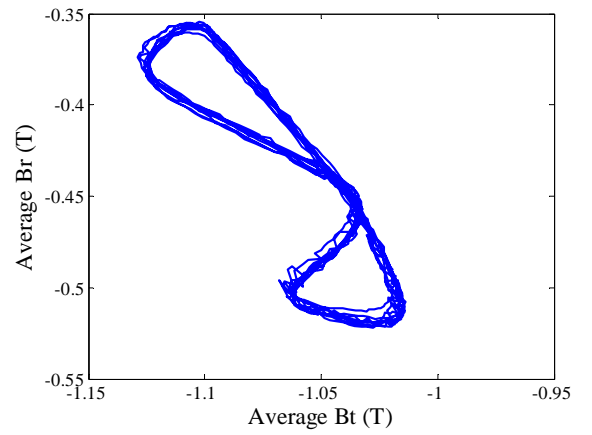
(a) Testing block 13



(b) Testing block 14



(c) Testing block 15



(d) Testing block 16

Fig. III.3 Plots of Br versus Bt.

### III.2 Flux density results in Open circuit

Testing blocks on steel.

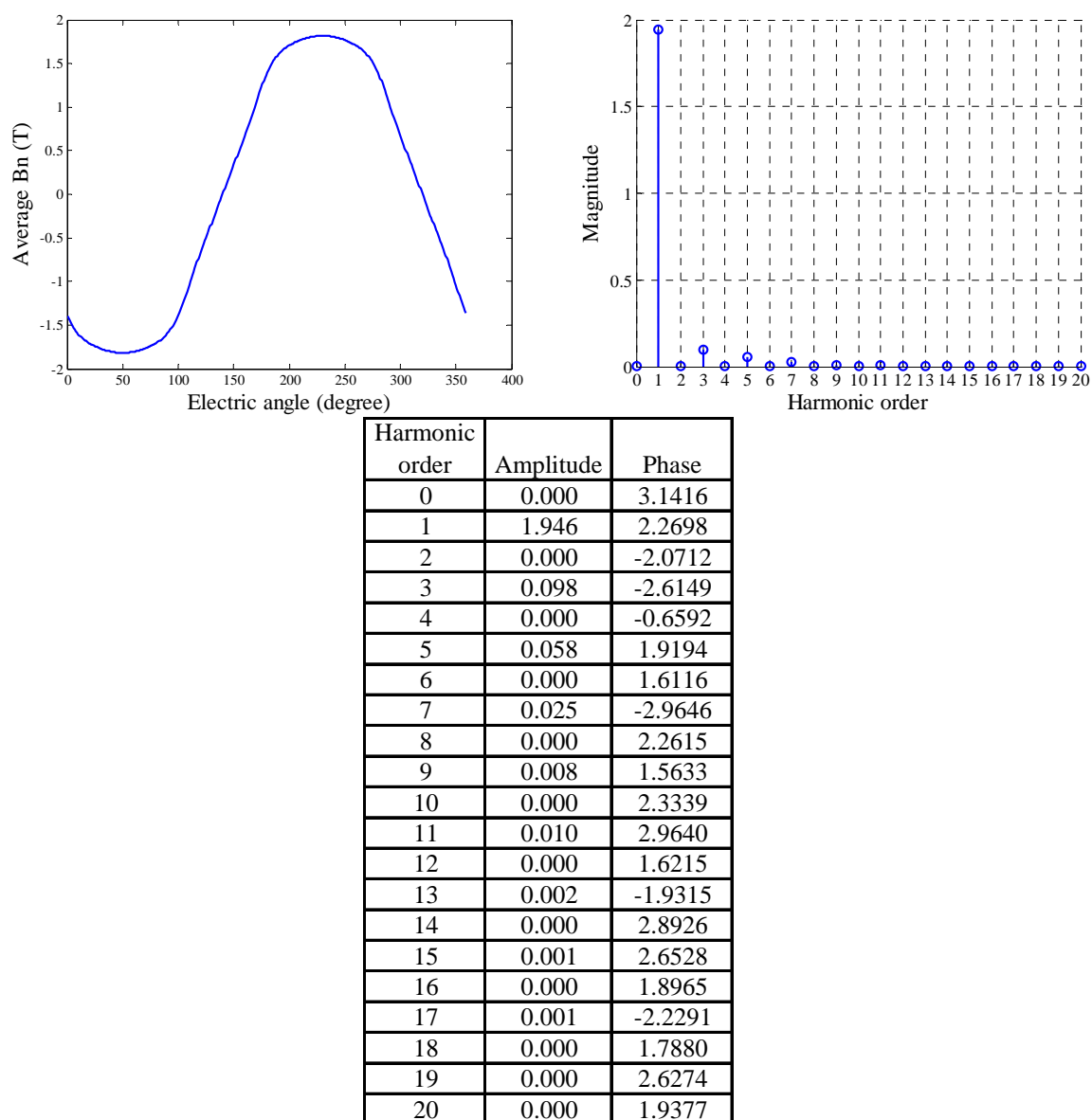


Fig. III.4 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 1. Phase in radian.

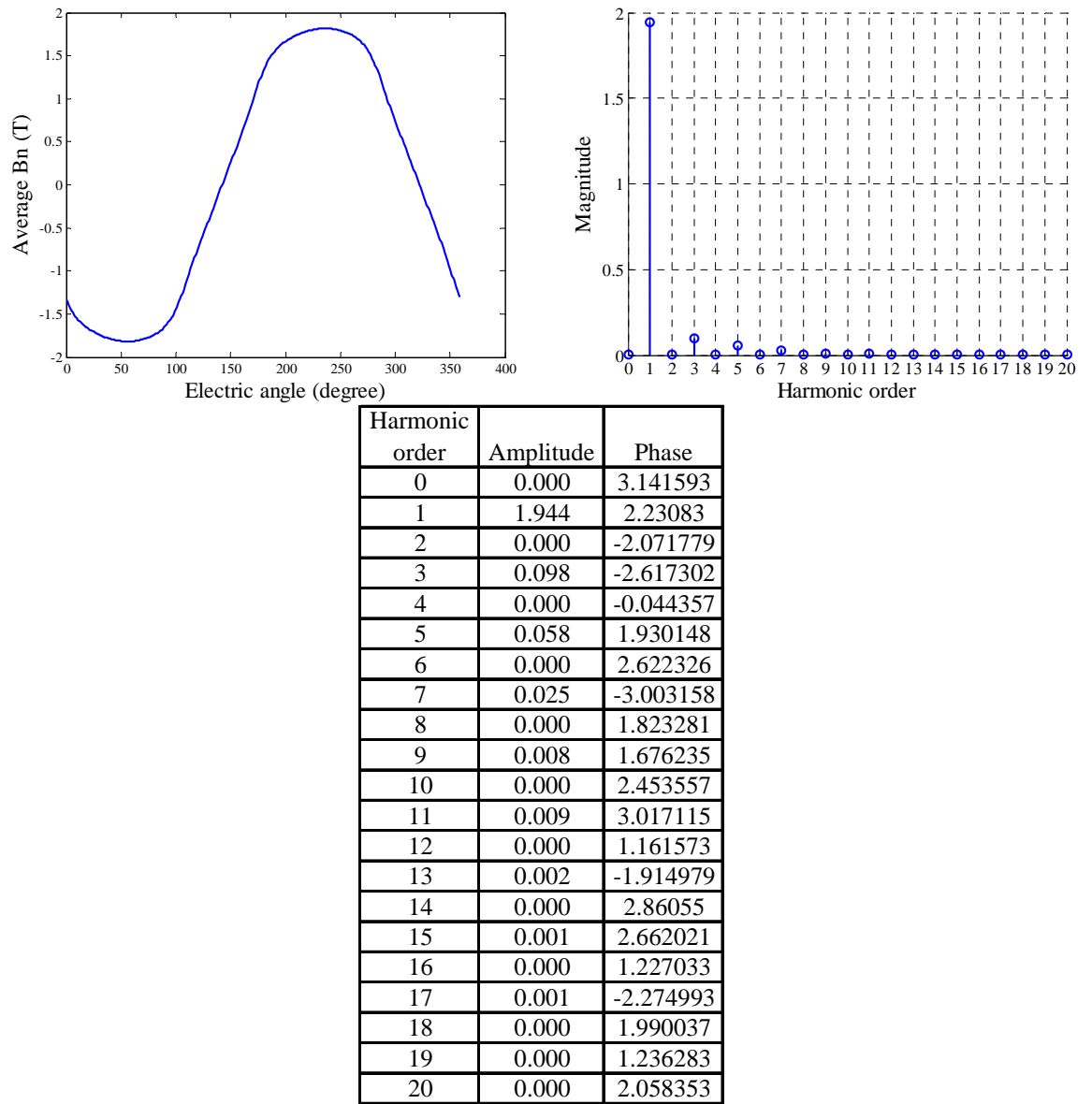
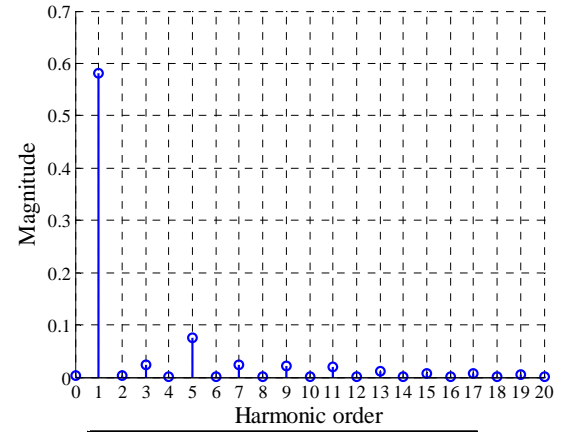
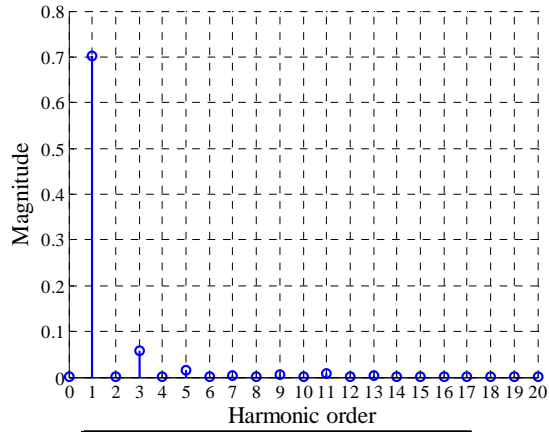
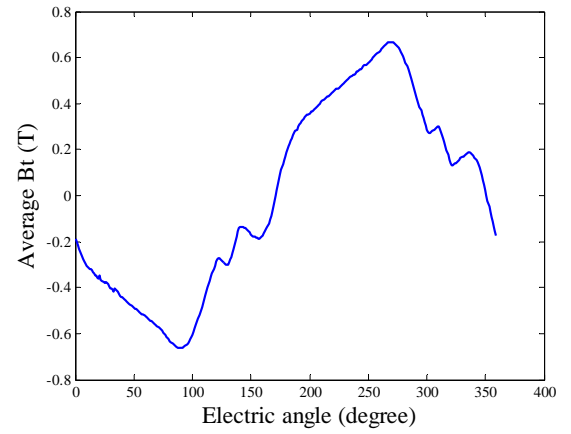
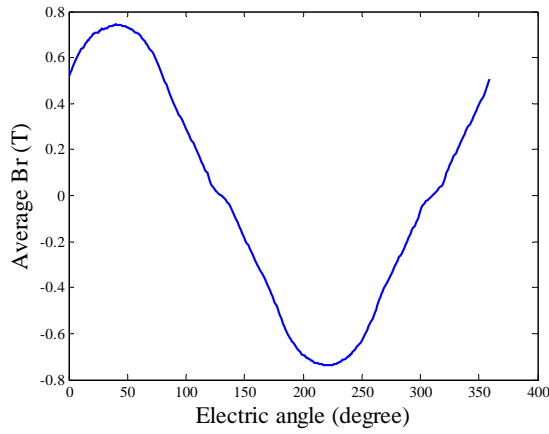


Fig. III.5 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 2. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.002	0
1	0.701	-0.698841
2	0.001	-2.152306
3	0.058	-2.108805
4	0.000	2.164053
5	0.016	-0.346783
6	0.000	0.934497
7	0.004	-1.706068
8	0.000	-0.69389
9	0.006	-3.050843
10	0.000	2.761691
11	0.008	-1.446474
12	0.000	0.427037
13	0.003	0.469334
14	0.000	1.605584
15	0.001	2.046872
16	0.000	0.362027
17	0.002	-2.398335
18	0.000	-0.695969
19	0.002	-0.771931
20	0.000	-1.026036

Harmonic order	Amplitude	Phase
0	0.003	0
1	0.582	1.832254
2	0.003	-2.000553
3	0.024	-0.754074
4	0.001	2.520824
5	0.076	1.636703
6	0.000	2.789392
7	0.023	2.903681
8	0.001	2.265098
9	0.021	2.019362
10	0.001	0.943517
11	0.020	-2.92711
12	0.000	0.103806
13	0.012	-1.60784
14	0.000	-2.79001
15	0.008	0.077426
16	0.000	1.792758
17	0.007	2.038645
18	0.000	1.334588
19	0.006	-2.522832
20	0.000	0.314522

Fig. III.6 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 3. Phase in radian.



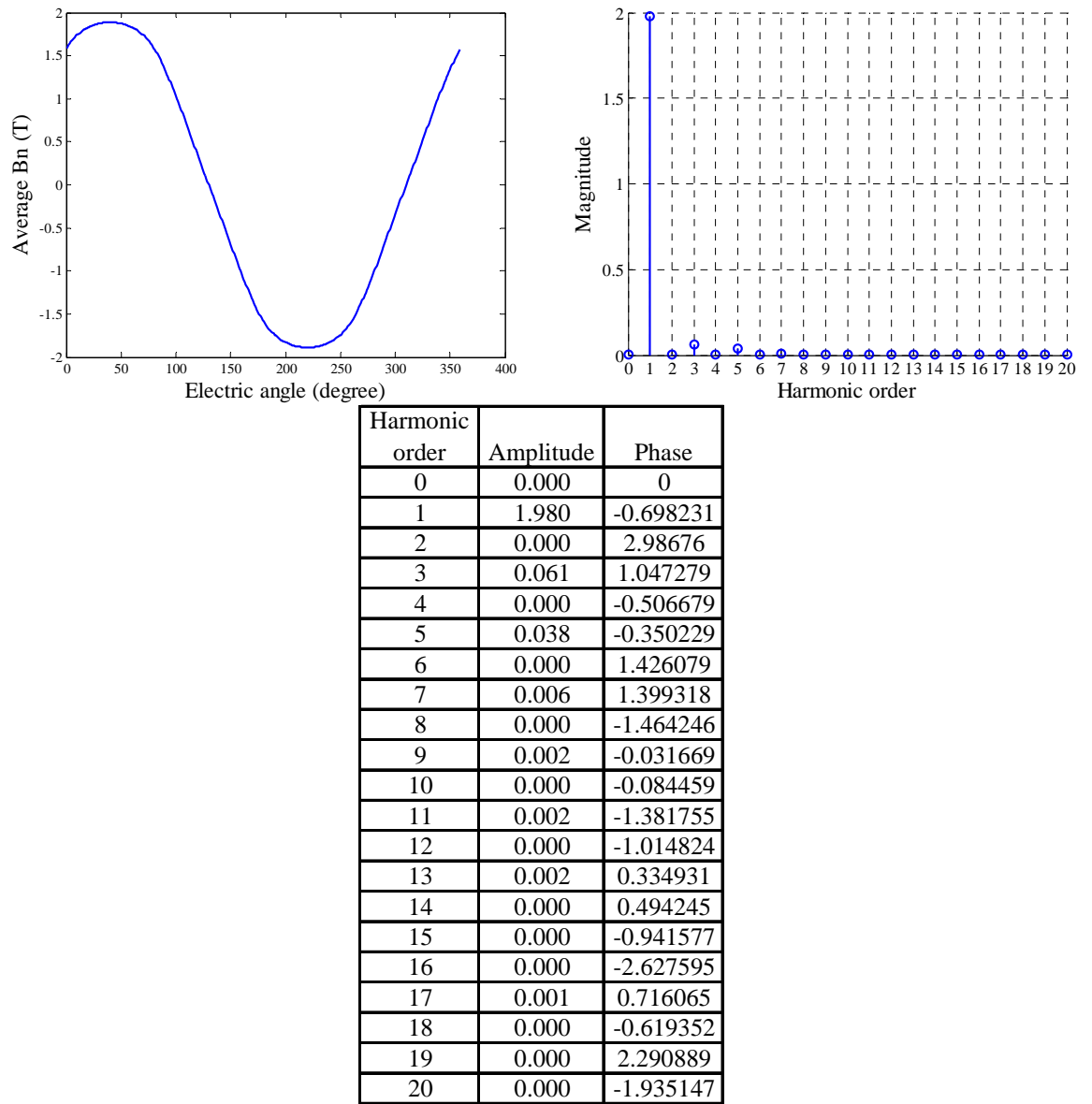


Fig. III.7 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 4. Phase in radian.

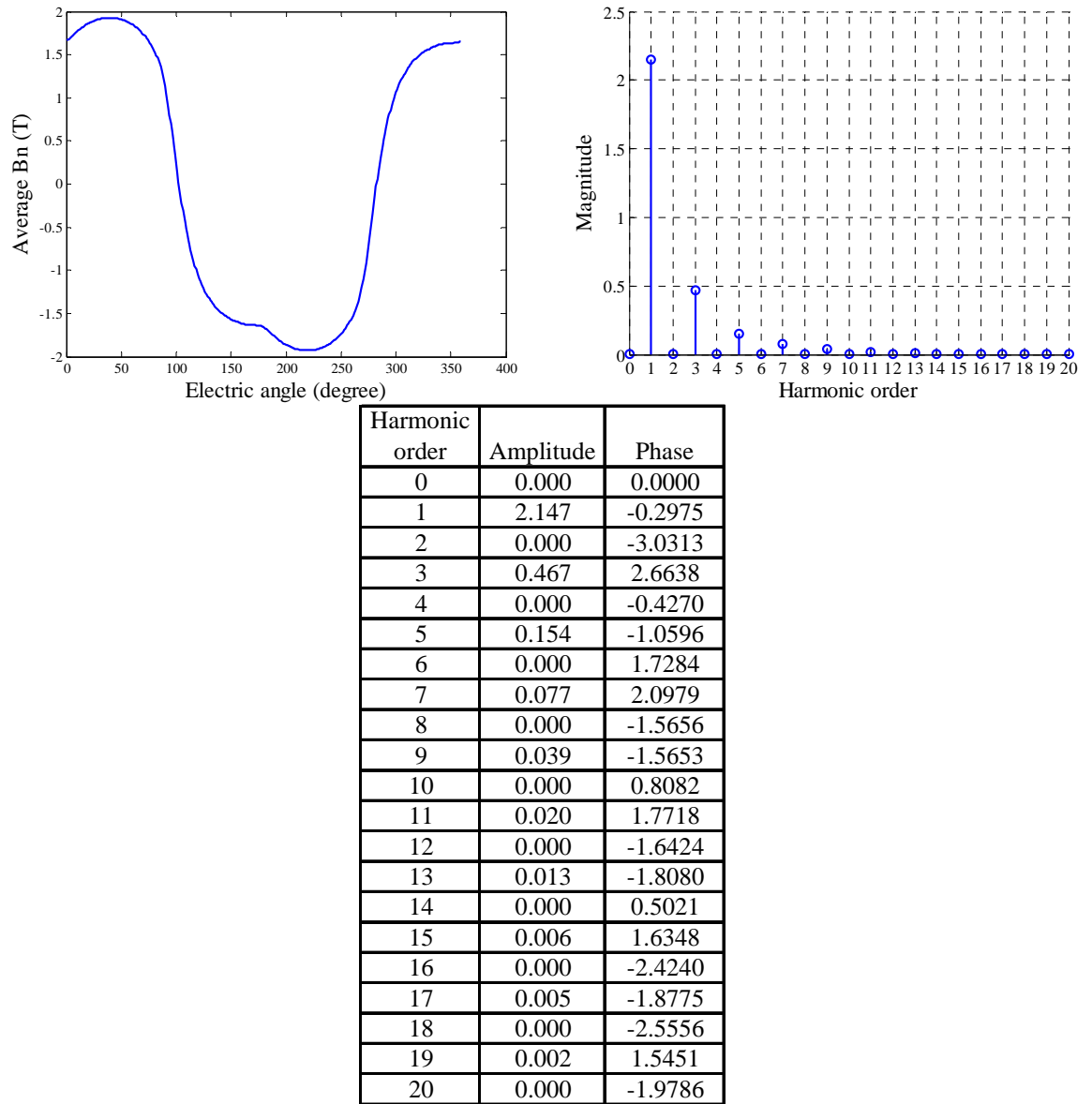


Fig. III.8 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 5. Phase in radian.

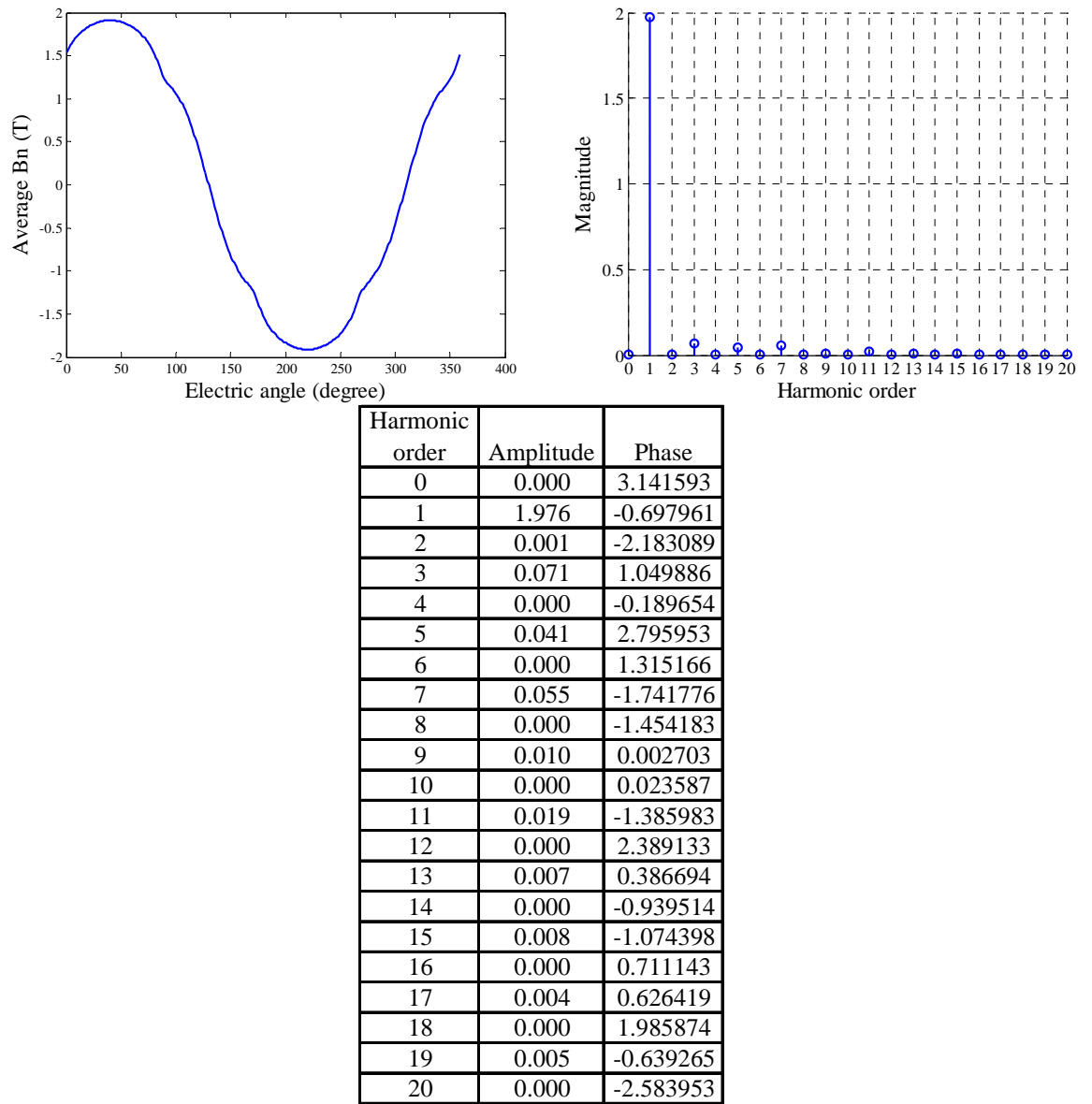


Fig. III.9 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 6. Phase in radian.

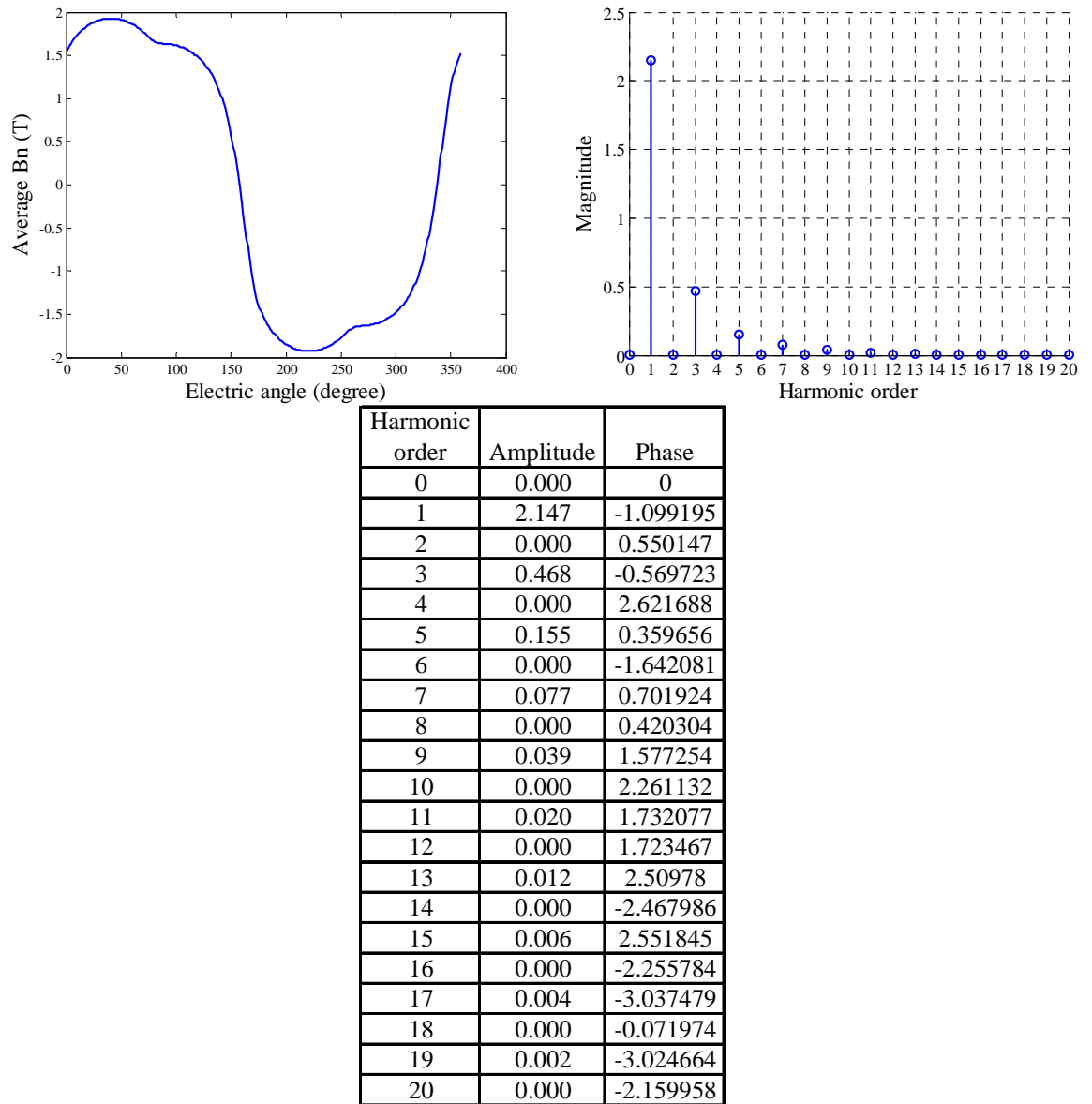


Fig. III.10 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 7. Phase in radian.

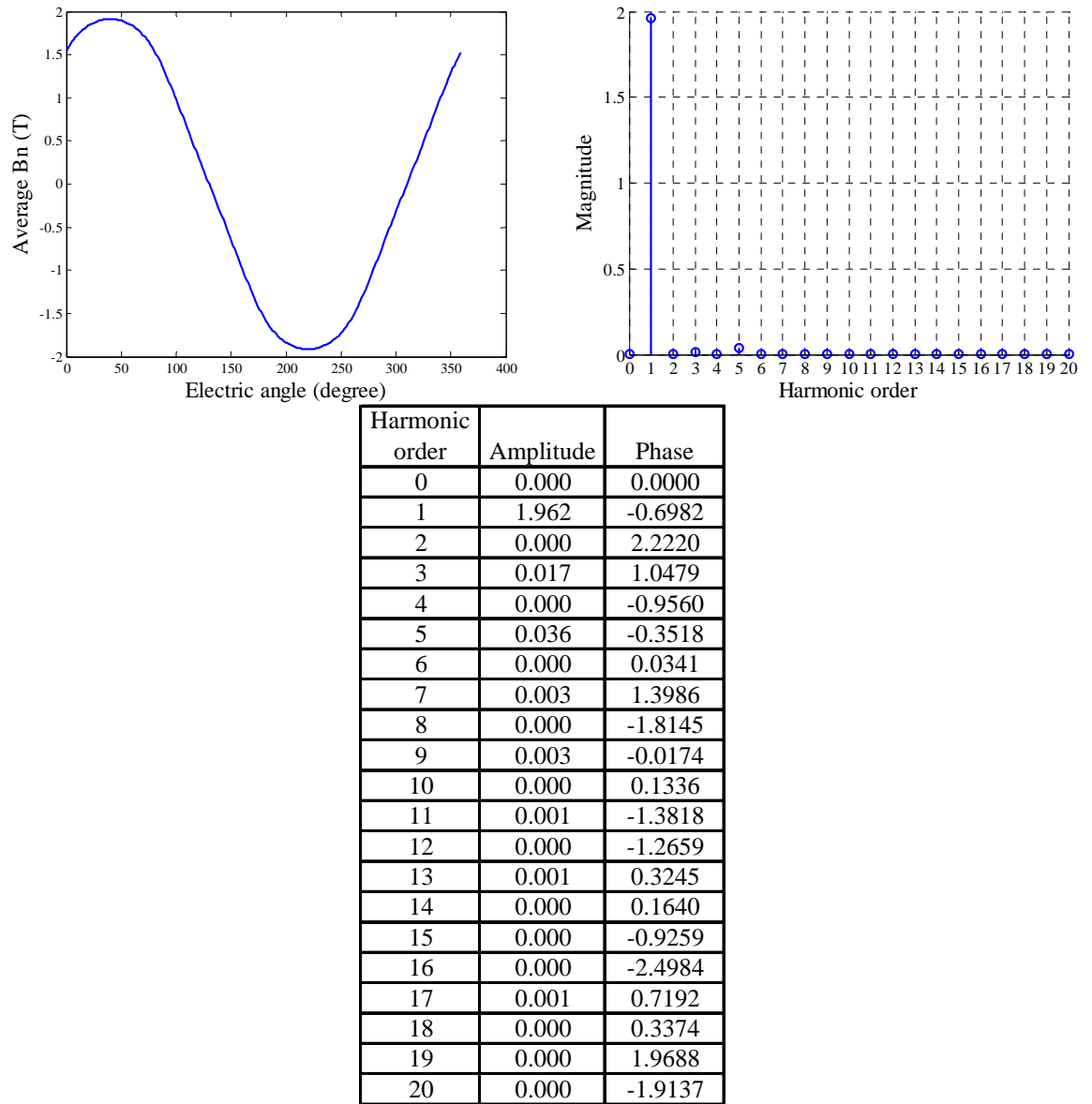
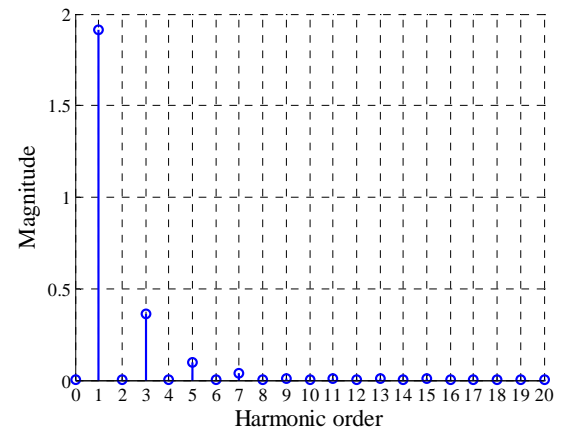
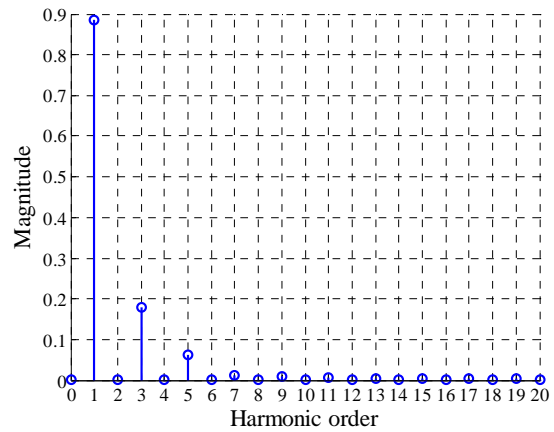
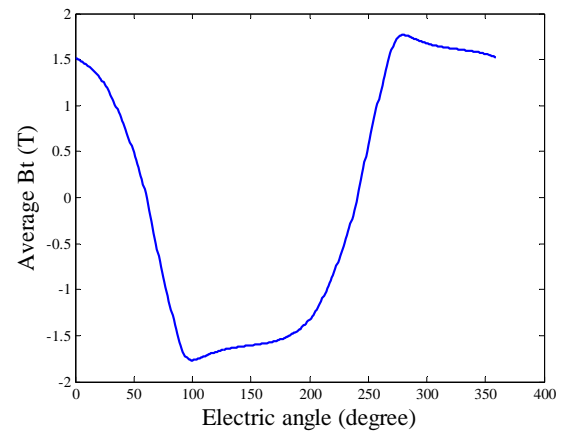
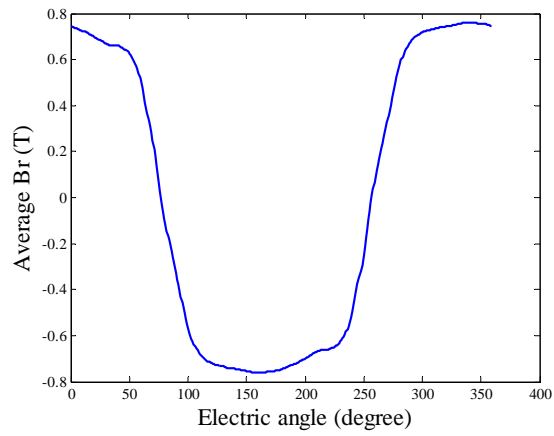


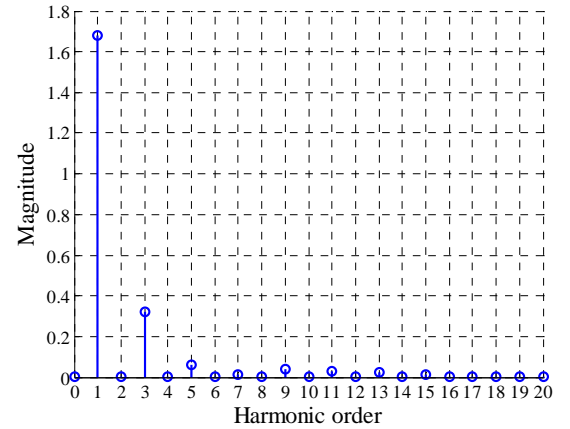
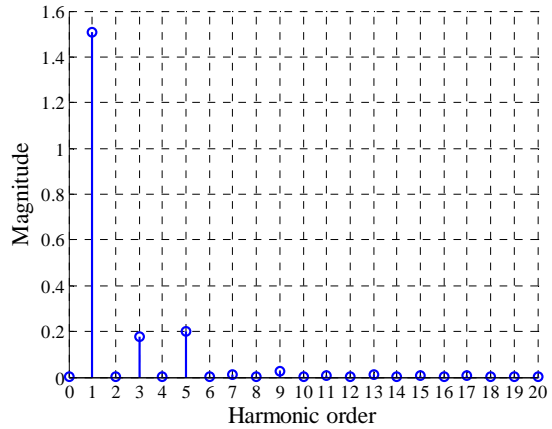
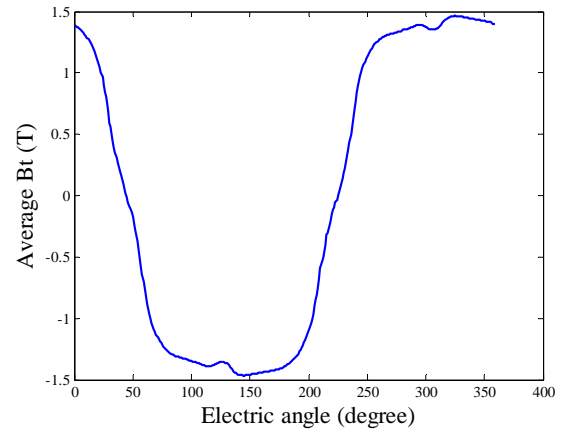
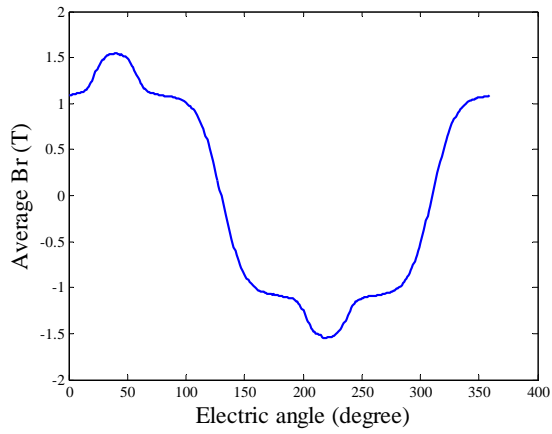
Fig. III.11 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 8. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	0.885	0.226826
2	0.000	-1.021881
3	0.178	-2.59067
4	0.000	1.223439
5	0.063	1.014887
6	0.000	-2.839316
7	0.013	-1.017636
8	0.000	0.841885
9	0.009	-2.322522
10	0.000	-1.951219
11	0.006	1.504421
12	0.000	1.953119
13	0.004	-1.732152
14	0.000	2.693541
15	0.004	1.317719
16	0.000	-0.09889
17	0.003	-1.581508
18	0.000	2.917653
19	0.003	1.563025
20	0.000	-0.001212

Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	1.914	0.60293
2	0.000	2.577311
3	0.364	-1.736797
4	0.000	2.009965
5	0.099	1.543417
6	0.000	0.399978
7	0.039	-1.592107
8	0.000	3.015973
9	0.011	1.631619
10	0.000	1.066106
11	0.007	-3.059838
12	0.000	2.860444
13	0.009	0.27685
14	0.000	1.863562
15	0.007	-2.661151
16	0.000	-1.892264
17	0.005	0.781374
18	0.000	1.193702
19	0.003	-1.973388
20	0.000	2.135275

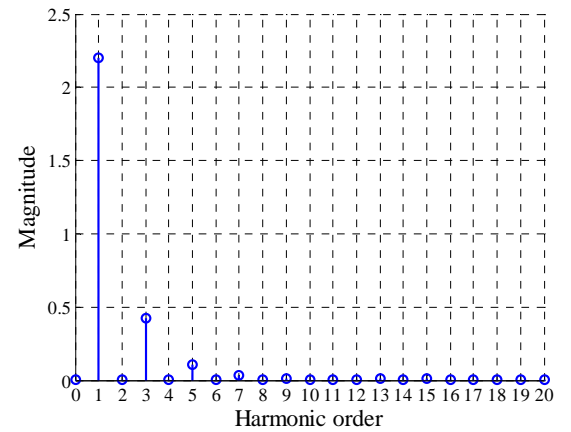
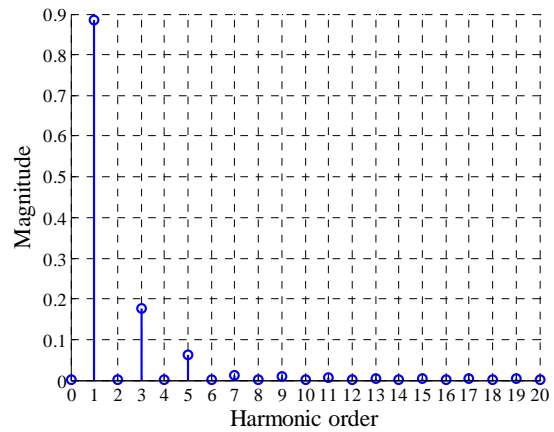
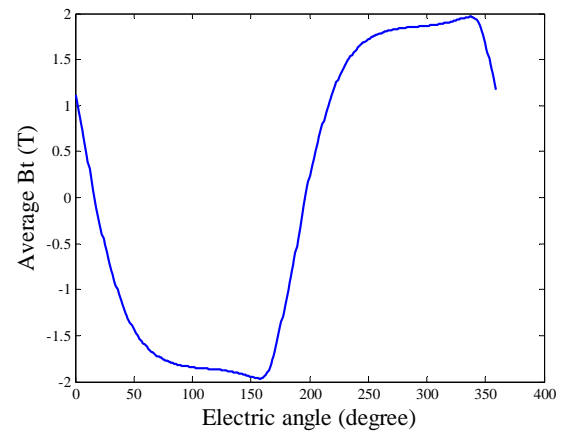
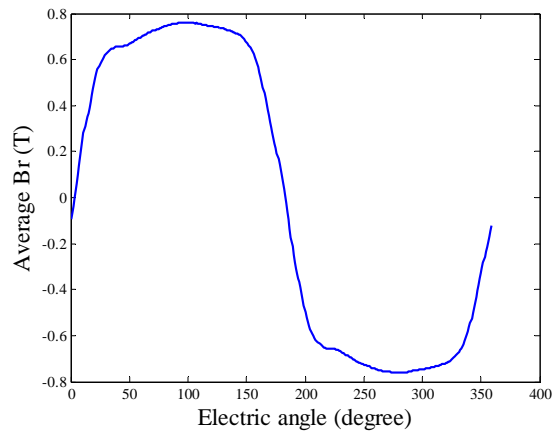
Fig. III.12 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 9. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.001	0
1	1.509	-0.698676
2	0.001	-1.441879
3	0.177	1.04231
4	0.000	-2.832449
5	0.201	2.796868
6	0.000	3.04957
7	0.012	1.489189
8	0.000	-1.034035
9	0.024	0.004789
10	0.000	1.995511
11	0.005	1.640034
12	0.000	0.112316
13	0.011	0.348758
14	0.000	-2.538879
15	0.009	-1.071625
16	0.000	-0.919049
17	0.007	-2.381657
18	0.001	-3.079391
19	0.001	-3.08249
20	0.000	1.849147

Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	1.678	0.786558
2	0.000	-0.801437
3	0.323	-0.645277
4	0.000	0.187682
5	0.060	-2.449607
6	0.000	-0.819061
7	0.013	-0.613104
8	0.000	2.762554
9	0.042	-1.812606
10	0.000	0.433659
11	0.028	2.436638
12	0.001	-1.027126
13	0.025	1.417252
14	0.000	-2.992653
15	0.014	-0.733123
16	0.001	1.804286
17	0.005	-1.587523
18	0.000	-0.072747
19	0.003	1.334043
20	0.000	-1.049391

Fig. III.13 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 10. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	0.885	-1.623539
2	0.000	-0.683642
3	0.177	-1.59584
4	0.000	-0.583601
5	0.063	-1.713911
6	0.000	0.454812
7	0.013	-2.448573
8	0.000	0.63846
9	0.009	2.344912
10	0.000	2.807906
11	0.006	1.974627
12	0.000	-1.076703
13	0.004	2.47489
14	0.000	-0.426541
15	0.004	2.884996
16	0.000	-1.193304
17	0.003	3.006788
18	0.000	-0.512533
19	0.003	-2.949541
20	0.000	-1.078152

Harmonic order	Amplitude	Phase
0	0.001	0
1	2.203	1.21429
2	0.000	-3.000694
3	0.425	0.806469
4	0.000	1.450389
5	0.110	0.937142
6	0.000	2.895172
7	0.034	1.445651
8	0.000	2.591313
9	0.012	2.052676
10	0.000	2.69969
11	0.006	-2.694974
12	0.000	1.187375
13	0.009	-2.01091
14	0.000	1.956077
15	0.009	-1.895565
16	0.000	-2.525265
17	0.006	-2.013543
18	0.000	0.027474
19	0.003	-1.867205
20	0.000	-0.190761

Fig. III.14 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 11. Phase in radian.



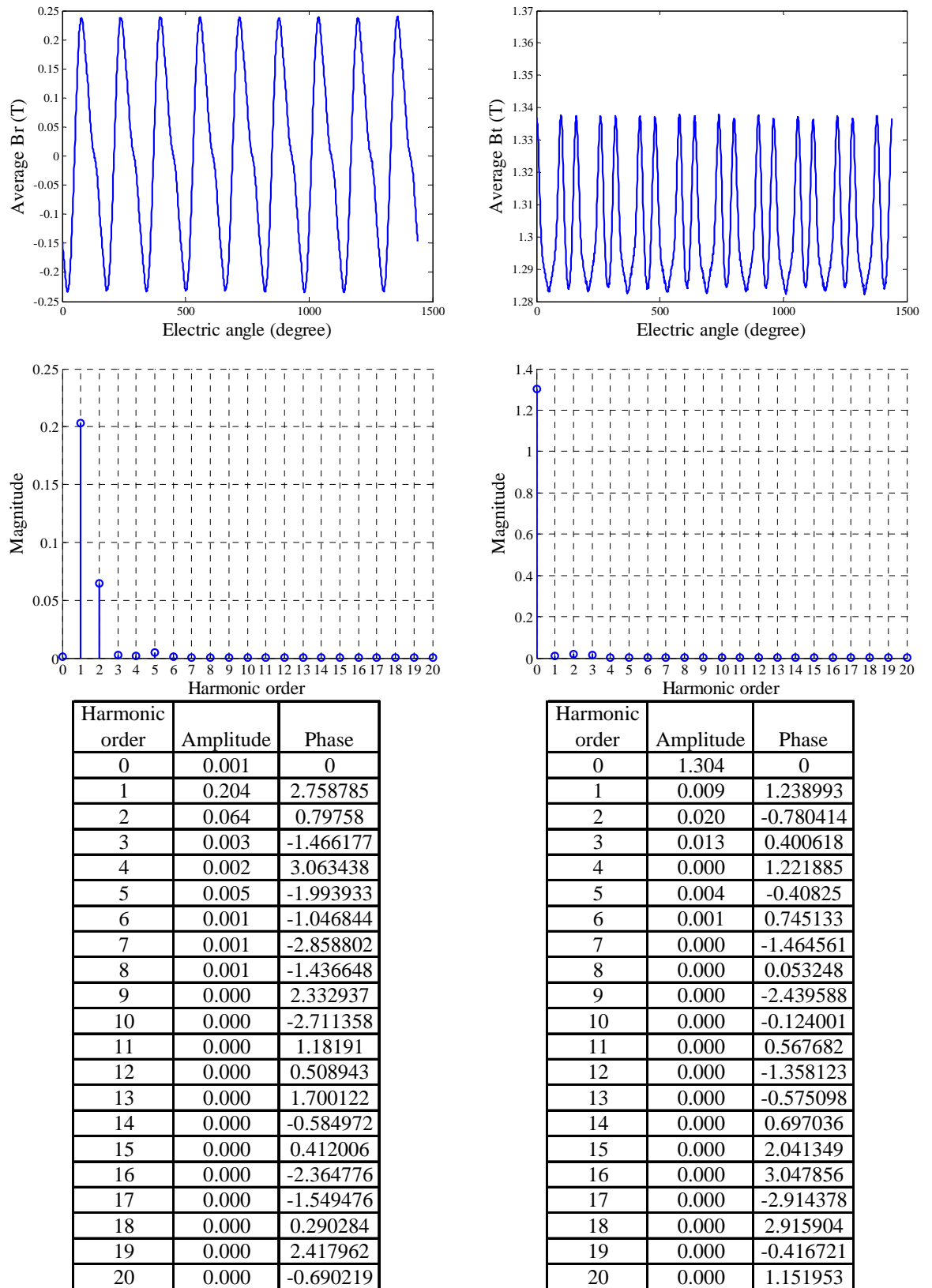


Fig. III.15 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 12. Phase in radian.

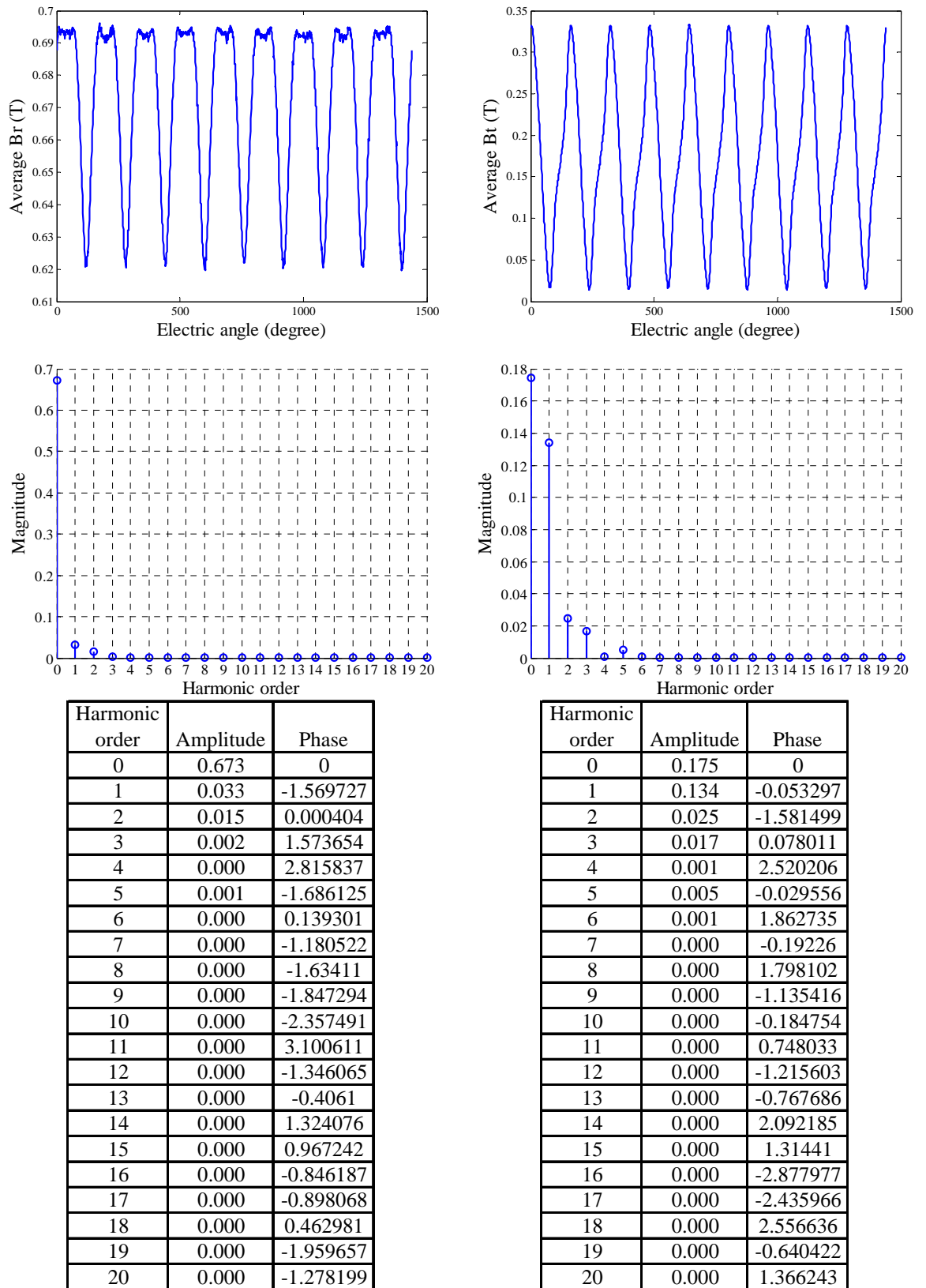


Fig. III.16 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 13. Phase in radian.

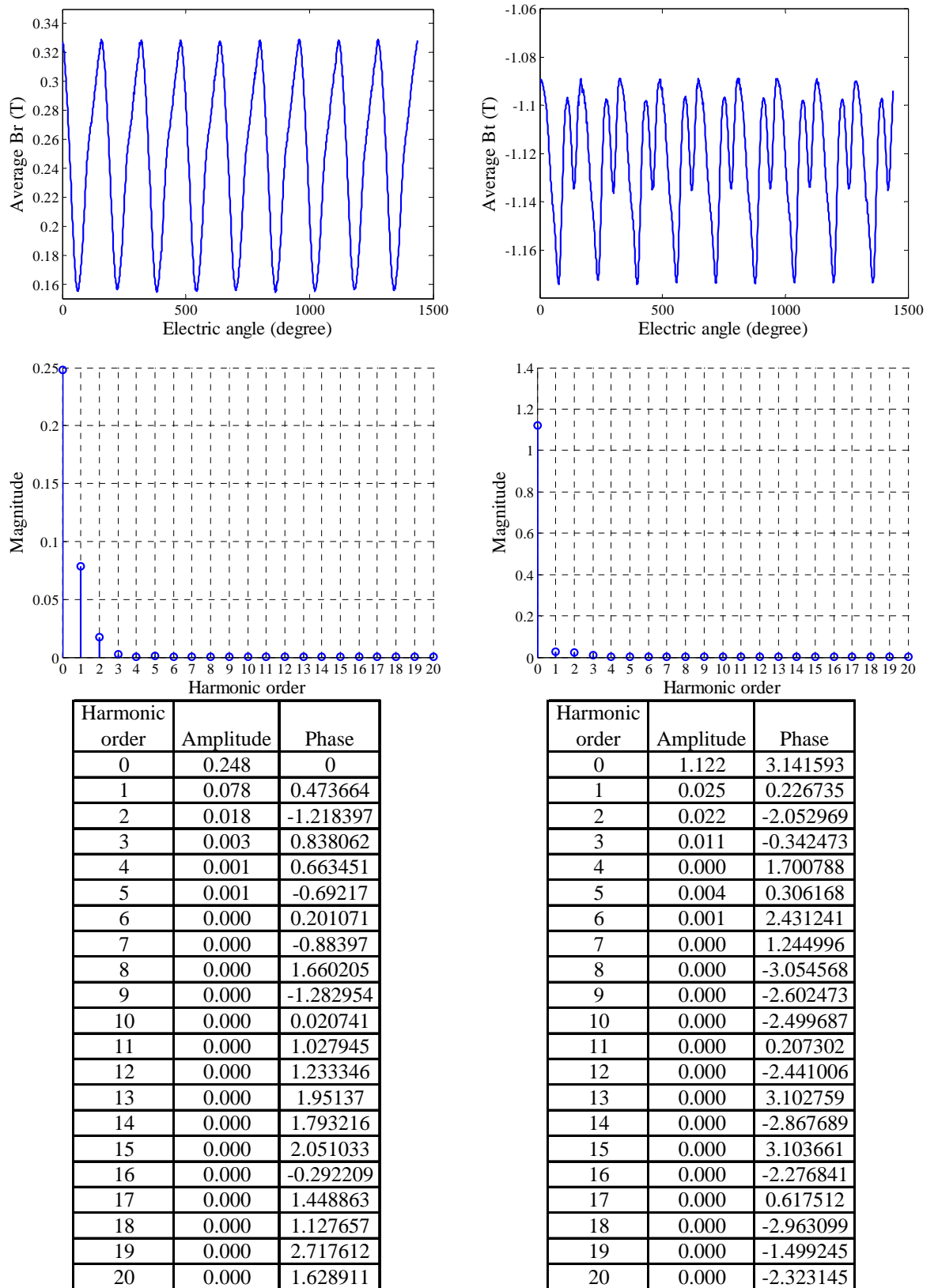


Fig. III.17 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 14. Phase in radian.

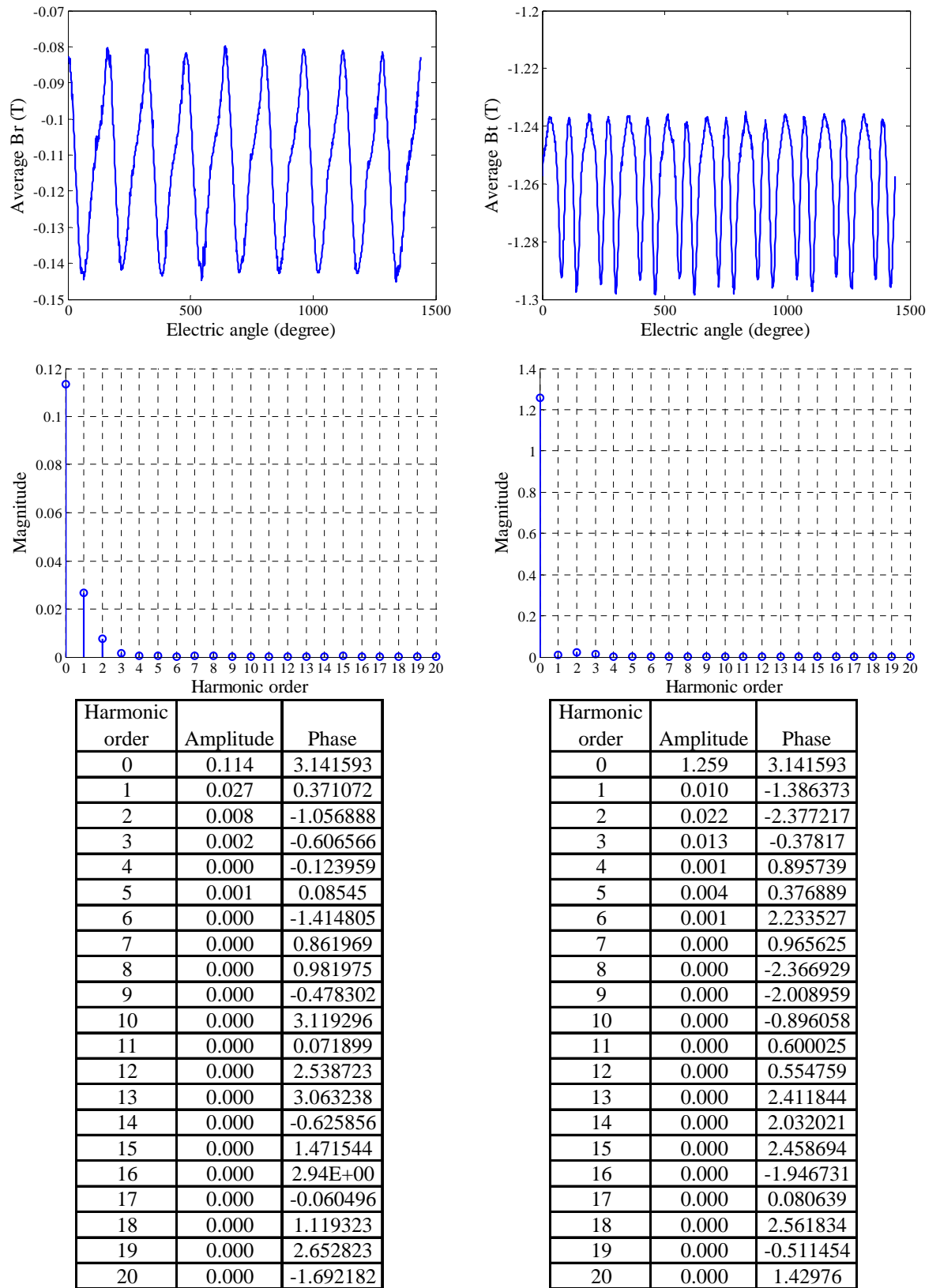


Fig. III.18 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 15. Phase in radian.

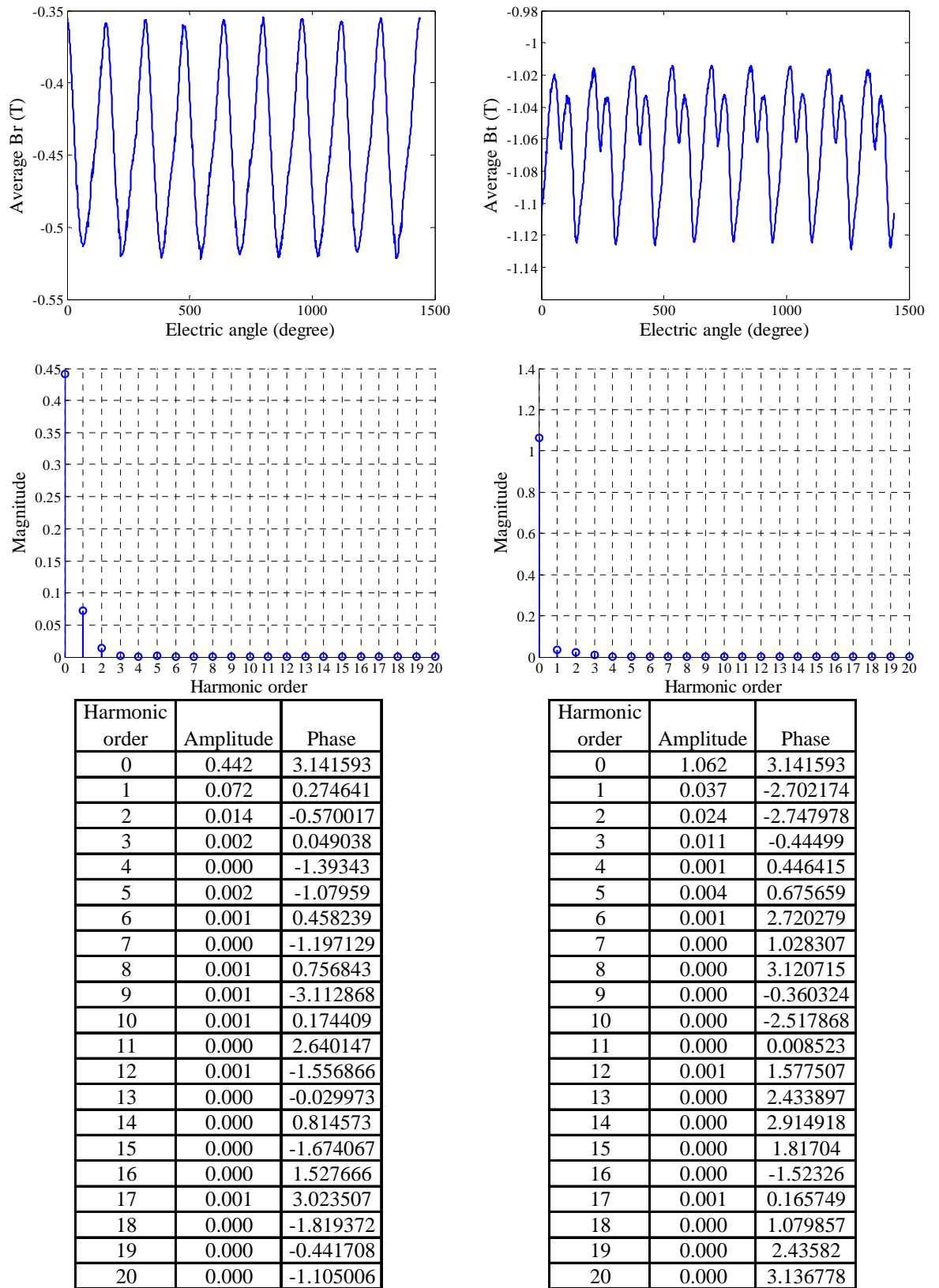
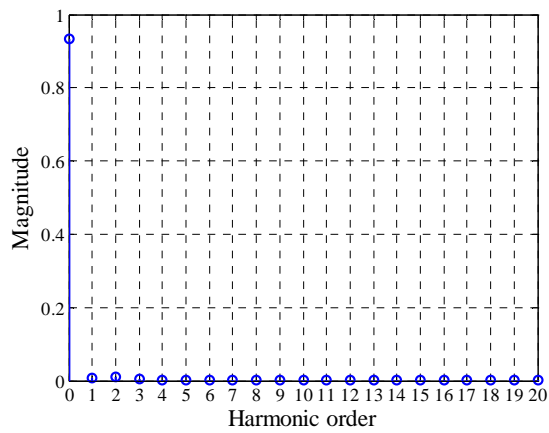
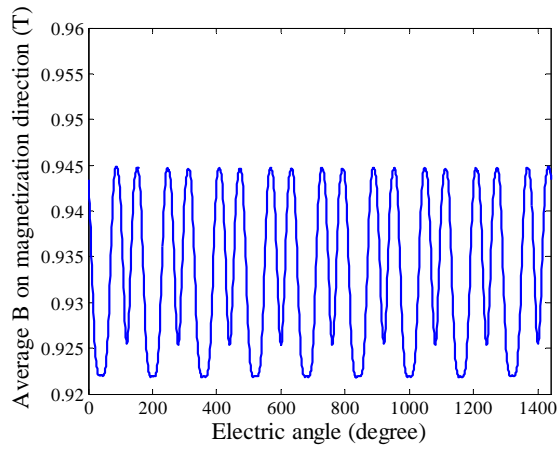


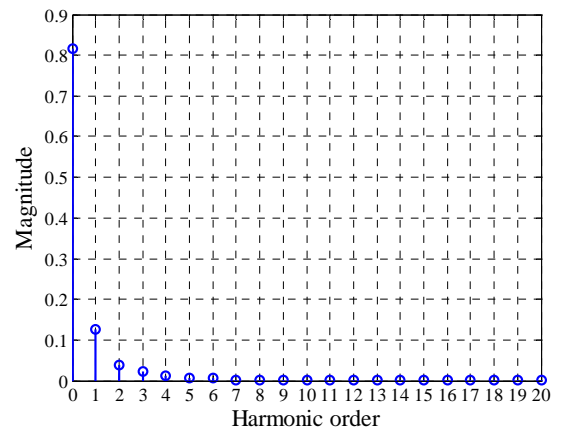
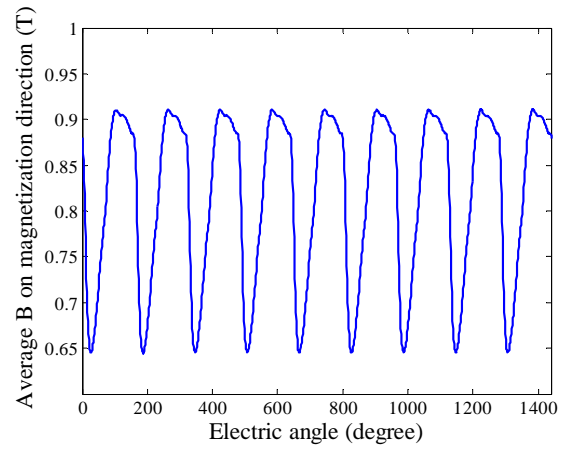
Fig. III.19 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 16. Phase in radian.

Test block on magnet:



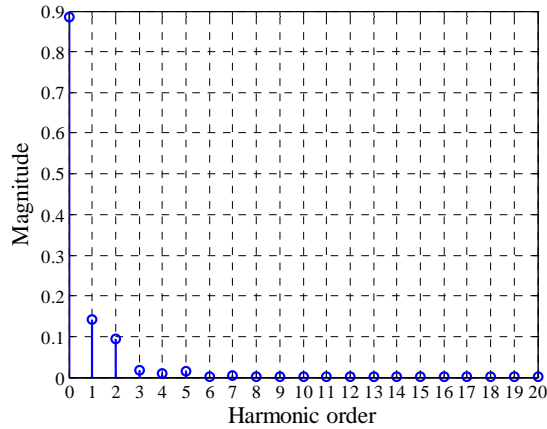
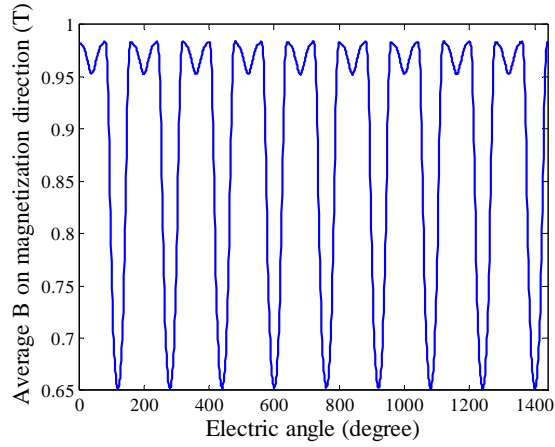
Harmonic order	Magnitude	Phase
0	0.933	0
1	0.006	1.574369
2	0.009	-0.002563
3	0.004	1.572635
4	0.000	-0.175664
5	0.001	-1.571432
6	0.000	-0.032379
7	0.000	-1.575347
8	0.000	0.110526
9	0.000	-1.32786
10	0.000	-0.109057
11	0.000	1.49137
12	0.000	-1.738589
13	0.000	-0.557889
14	0.000	0.352601
15	0.000	0.100505
16	0.000	0.015363
17	0.000	-0.911363
18	0.000	-2.052349
19	0.000	1.278857
20	0.000	1.156221

Fig. III.20 Average B in magnetization direction of testing block 1 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



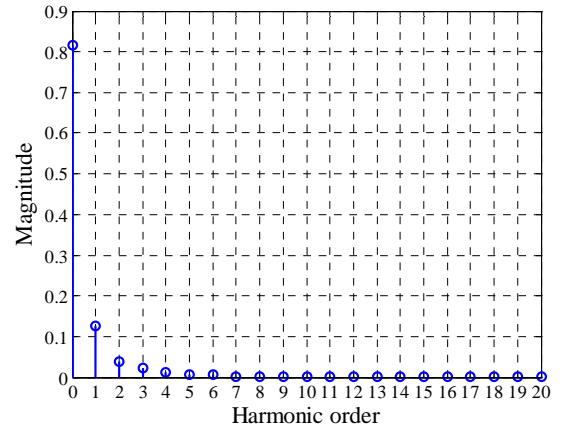
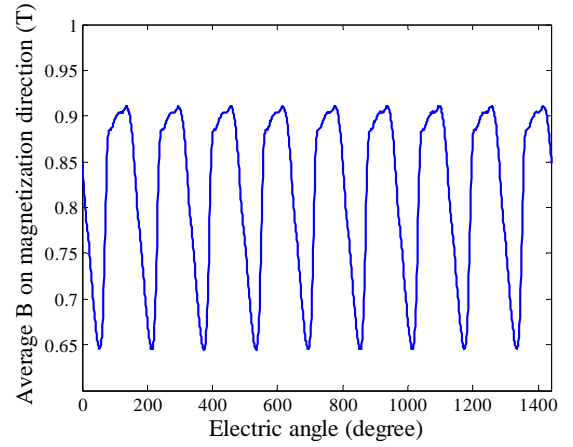
Harmonic order	Magnitude	Phase
0	0.816	0
1	0.125	1.674785
2	0.038	0.497721
3	0.022	0.362393
4	0.012	-0.477969
5	0.008	-0.38938
6	0.006	-0.84327
7	0.002	-1.275058
8	0.002	-1.138639
9	0.001	-1.700354
10	0.000	-1.730943
11	0.000	-2.031282
12	0.000	-2.927828
13	0.000	-2.489422
14	0.000	2.429922
15	0.000	-0.733044
16	0.000	-2.501951
17	0.000	0.837734
18	0.000	-2.352821
19	0.000	1.605352
20	0.000	-1.109246

Fig. III.21 Average B in magnetization direction of testing block 2 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



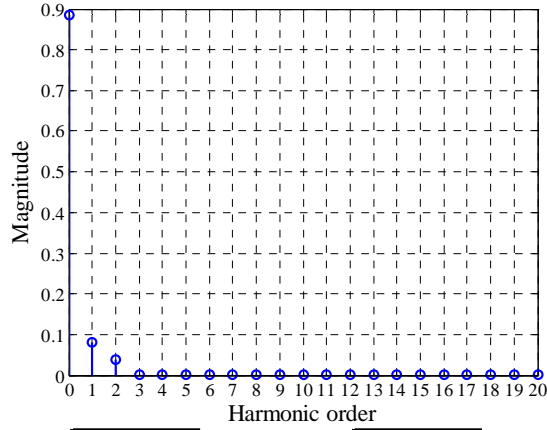
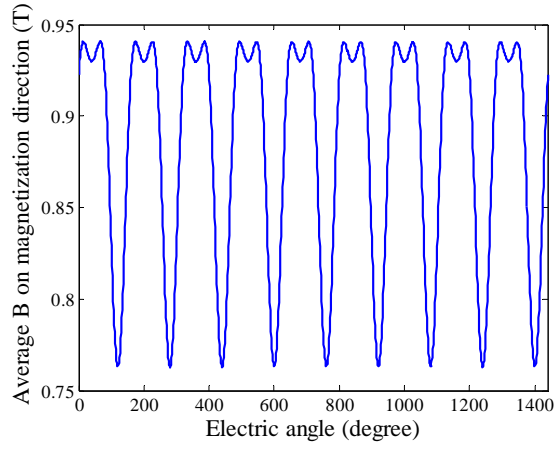
Harmonic order	Magnitude	Phase
0	0.886	0
1	0.143	-1.571101
2	0.094	-0.000489
3	0.018	1.571341
4	0.010	-0.003477
5	0.014	1.571183
6	0.003	-3.129086
7	0.003	1.568359
8	0.003	-3.13218
9	0.000	1.517013
10	0.001	3.134791
11	0.000	-1.634686
12	0.000	-2.814749
13	0.000	-1.368126
14	0.000	0.560008
15	0.000	-1.777971
16	0.000	0.034472
17	0.000	-2.981361
18	0.000	-1.109302
19	0.000	1.325897
20	0.000	1.754522

Fig. III.22 Average B in magnetization direction of testing block 3 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



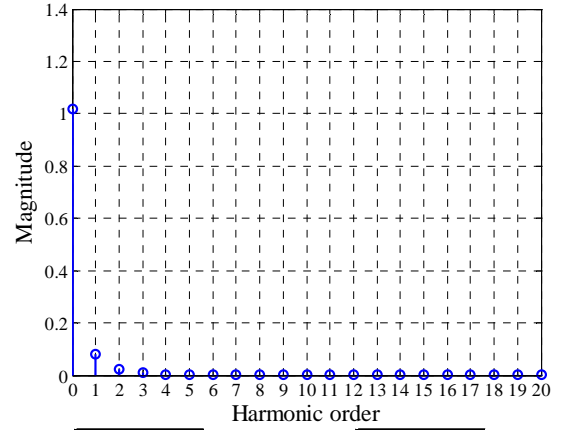
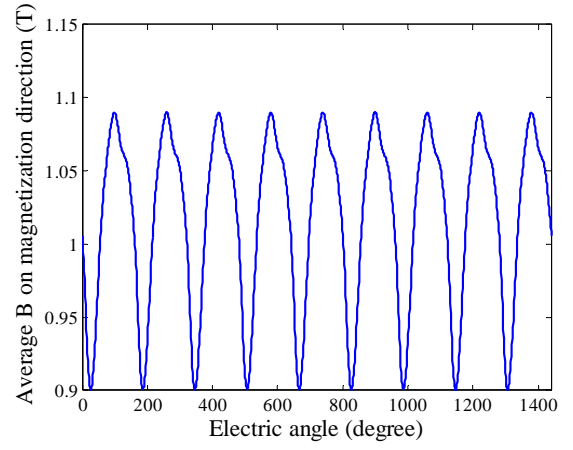
Harmonic order	Magnitude	Phase
0	0.816	0
1	0.125	1.467239
2	0.038	-0.496946
3	0.023	2.779701
4	0.012	0.470052
5	0.008	-2.753519
6	0.006	0.829884
7	0.002	-1.872708
8	0.002	1.121227
9	0.001	-1.446502
10	0.000	1.881865
11	0.000	-1.086288
12	0.000	-3.1065
13	0.000	-1.028417
14	0.000	-2.61775
15	0.000	1.914269
16	0.000	1.610026
17	0.000	2.654032
18	0.000	-0.97813
19	0.000	1.230943
20	0.000	2.468961

Fig. III.23 Average B in magnetization direction of testing block 4 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



Harmonic order	Magnitude	Phase
0	0.884	0
1	0.081	-1.572072
2	0.038	-0.001938
3	0.003	1.573537
4	0.000	-0.018628
5	0.000	-1.698633
6	0.000	0.005601
7	0.000	-1.582546
8	0.000	0.344079
9	0.000	-2.715323
10	0.000	0.256712
11	0.000	1.993495
12	0.000	0.579394
13	0.000	-3.027468
14	0.000	-2.030271
15	0.000	2.069703
16	0.000	-2.134858
17	0.000	-2.637539
18	0.000	-1.379823
19	0.000	0.959574
20	0.000	-2.476978

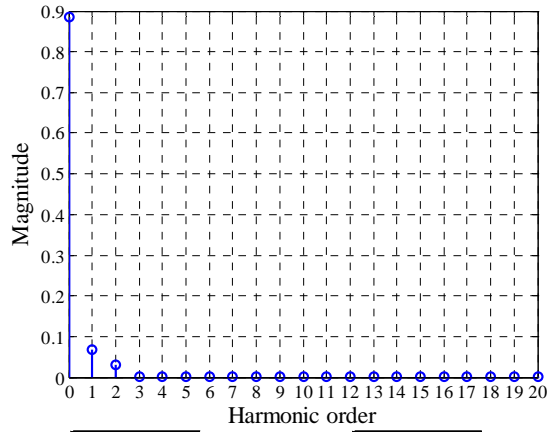
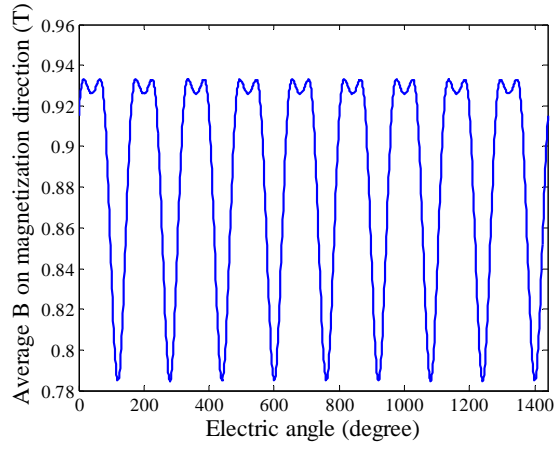
Fig. III.24 Average B in magnetization direction of testing block 5 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



Harmonic order	Magnitude	Phase
0	1.018	0
1	0.081	2.103703
2	0.025	0.764678
3	0.010	0.284604
4	0.002	-1.385645
5	0.002	-1.00588
6	0.001	-1.266962
7	0.000	-2.298393
8	0.000	-1.386409
9	0.000	-2.482999
10	0.000	-2.048387
11	0.000	2.220633
12	0.000	2.081205
13	0.000	1.914984
14	0.000	1.195045
15	0.000	-2.700598
16	0.000	0.806569
17	0.000	-1.232292
18	0.000	-2.680226
19	0.000	1.137421
20	0.000	1.714486

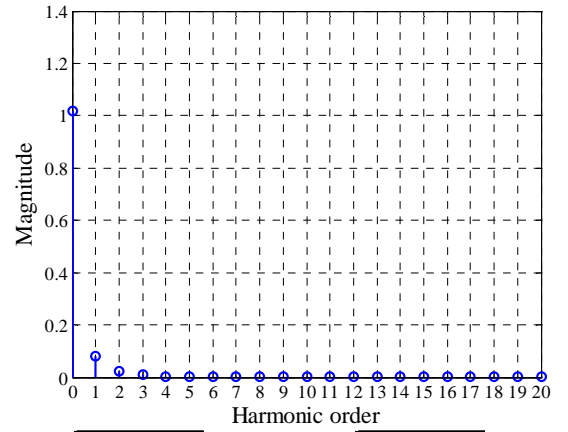
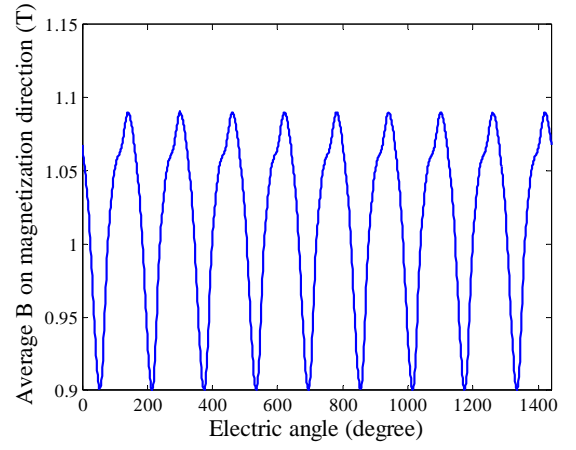
Fig. III.25 Average B in magnetization direction of testing block 6 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.





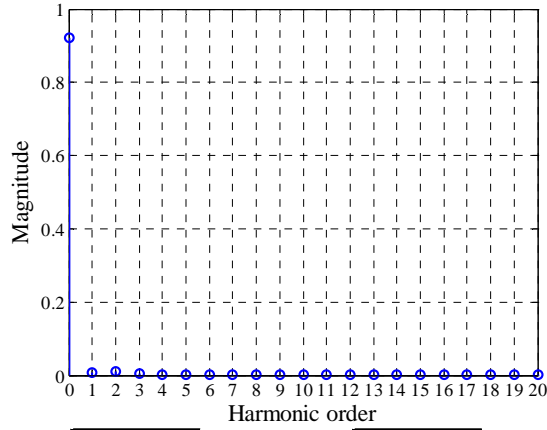
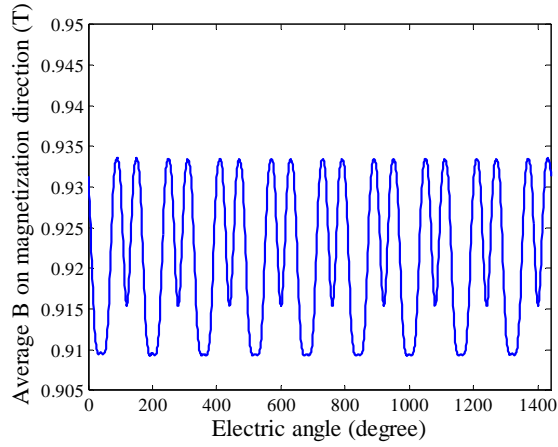
Harmonic order	Magnitude	Phase
0	0.885	0
1	0.069	-1.572044
2	0.030	-0.001789
3	0.002	1.582787
4	0.000	-0.710489
5	0.001	-1.610124
6	0.000	-0.049188
7	0.000	-1.624447
8	0.000	0.218183
9	0.000	-2.864482
10	0.000	0.191638
11	0.000	1.660372
12	0.000	2.378298
13	0.000	-3.04903
14	0.000	-1.937719
15	0.000	-1.855659
16	0.000	-1.493902
17	0.000	-2.026689
18	0.000	-2.013534
19	0.000	-0.125863
20	0.000	0.599014

Fig. III.26 Average B in magnetization direction of testing block 7 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



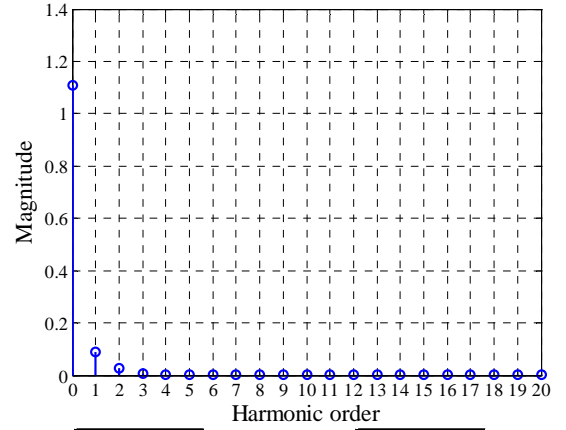
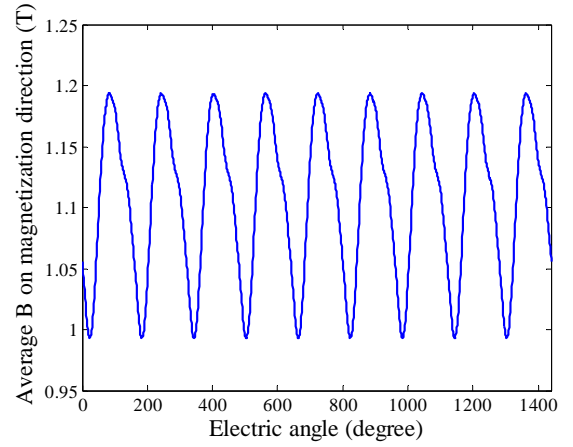
Harmonic order	Magnitude	Phase
0	1.018	0
1	0.081	1.035658
2	0.025	-0.765977
3	0.010	2.855552
4	0.002	1.359486
5	0.002	-2.126528
6	0.001	1.269785
7	0.000	-1.11885
8	0.000	1.422032
9	0.000	-0.616364
10	0.000	1.710298
11	0.000	1.015995
12	0.000	-1.211247
13	0.000	2.315584
14	0.000	-0.572127
15	0.000	-2.612445
16	0.000	0.88504
17	0.000	-2.114476
18	0.000	-1.374949
19	0.000	-1.39152
20	0.000	1.306255

Fig. III.27 Average B in magnetization direction of testing block 8 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



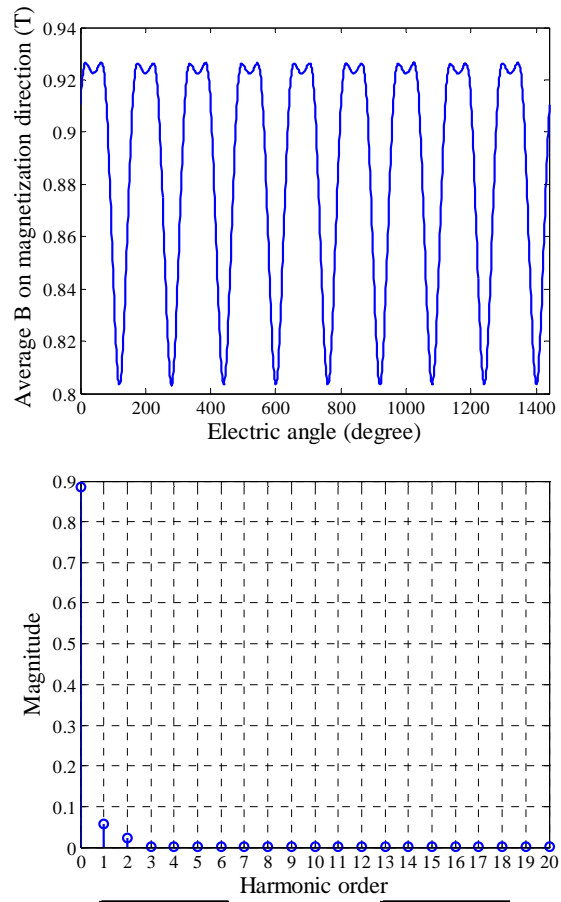
Harmonic order	Magnitude	Phase
0	0.922	0
1	0.008	1.573384
2	0.009	-0.00287
3	0.004	1.572084
4	0.000	-0.067718
5	0.001	-1.571082
6	0.000	-0.030163
7	0.000	-1.574322
8	0.000	0.062585
9	0.000	-1.508575
10	0.000	-0.03765
11	0.000	1.514339
12	0.000	-1.61472
13	0.000	-0.896994
14	0.000	0.393928
15	0.000	-0.471026
16	0.000	0.094132
17	0.000	-0.850176
18	0.000	-1.998546
19	0.000	1.284924
20	0.000	1.387245

Fig. III.28 Average B in magnetization direction of testing block 9 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



Harmonic order	Magnitude	Phase
0	1.107	0
1	0.089	2.514998
2	0.026	0.846269
3	0.004	0.347409
4	0.001	-2.40095
5	0.002	-1.65196
6	0.001	-0.90562
7	0.000	-2.485092
8	0.000	-1.402526
9	0.000	2.80657
10	0.000	-1.214484
11	0.000	1.650063
12	0.000	2.747719
13	0.000	-0.5682
14	0.000	0.034832
15	0.000	0.818379
16	0.000	-1.681084
17	0.000	-0.421183
18	0.000	-2.256379
19	0.000	1.443451
20	0.000	0.039503

Fig. III.29 Average B in magnetization direction of testing block 10 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



Harmonic order	Magnitude	Phase
0	0.887	0
1	0.058	-1.573141
2	0.023	-0.001713
3	0.001	1.584834
4	0.000	-2.935719
5	0.001	-1.554264
6	0.000	-0.037806
7	0.000	-1.577687
8	0.000	0.00179
9	0.000	-1.428159
10	0.000	-0.125766
11	0.000	1.27602
12	0.000	-1.300767
13	0.000	-0.838708
14	0.000	-0.186674
15	0.000	1.467289
16	0.000	1.368452
17	0.000	-2.770113
18	0.000	-1.768603
19	0.000	2.065662
20	0.000	-2.099895

Fig. III.30 Average B in magnetization direction of testing block 11 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.

### III.3 Flux density results of 3-phase model with 1000 A-t peak phase current excitation

Test block on steel.

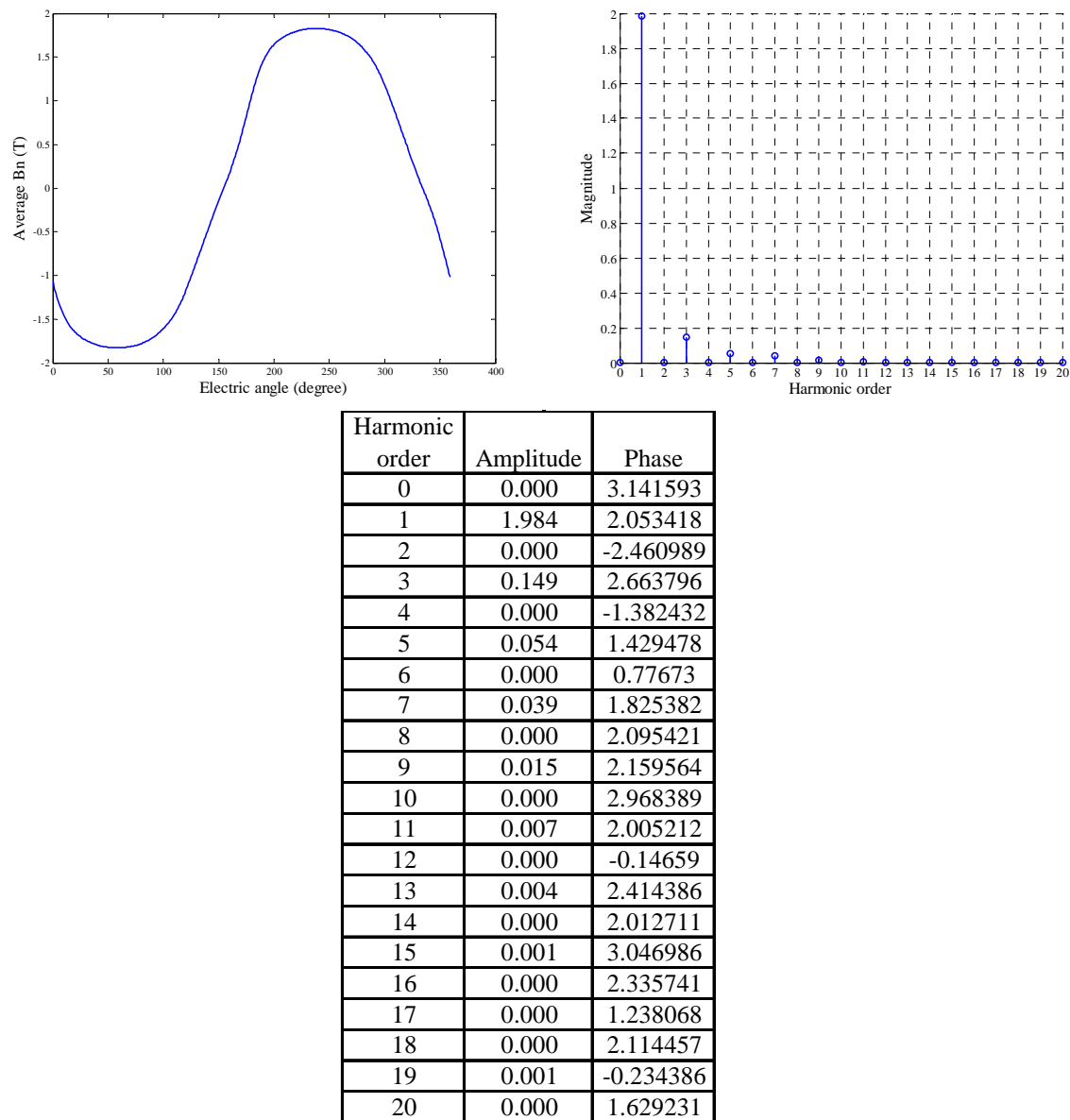


Fig. III.31 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 1. Phase in radian.

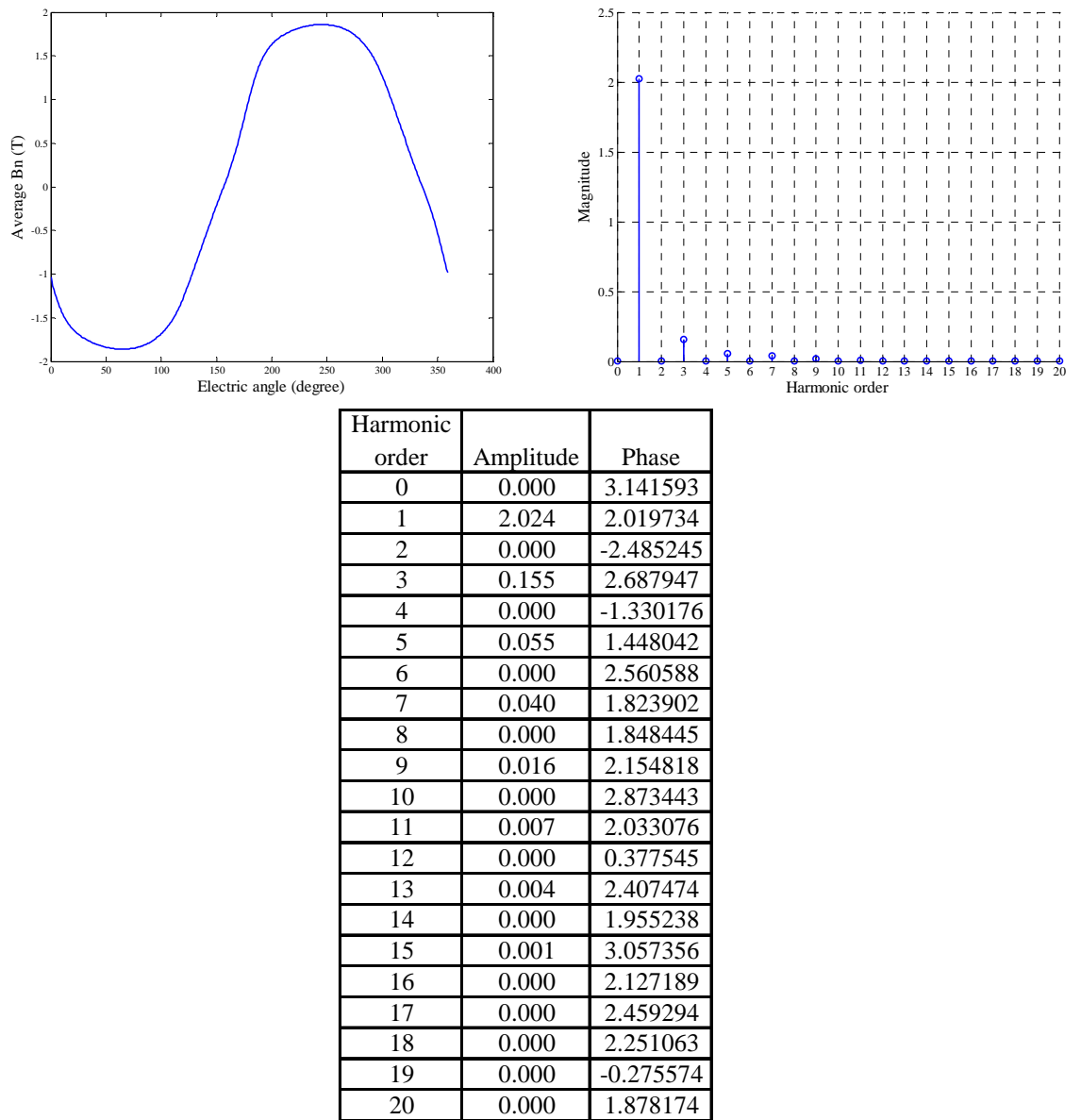
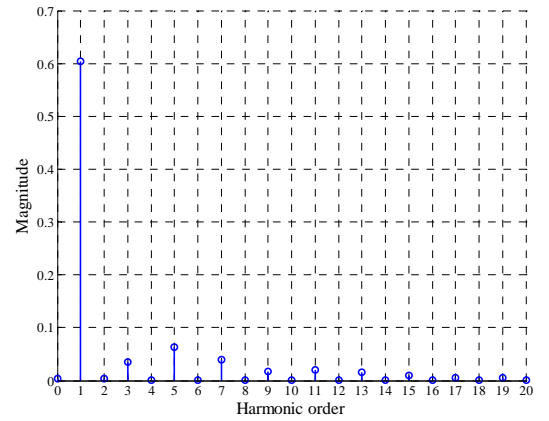
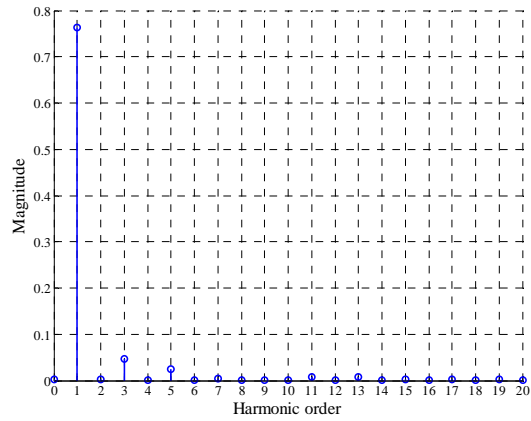
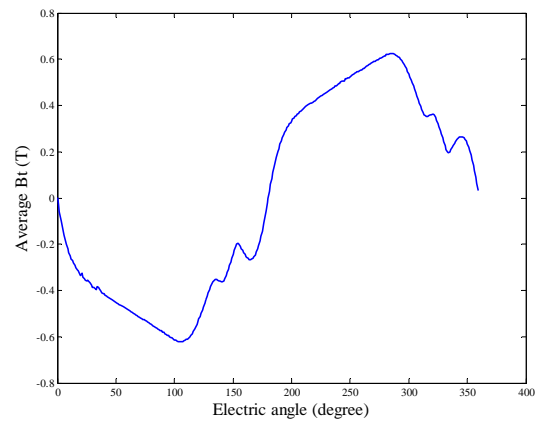
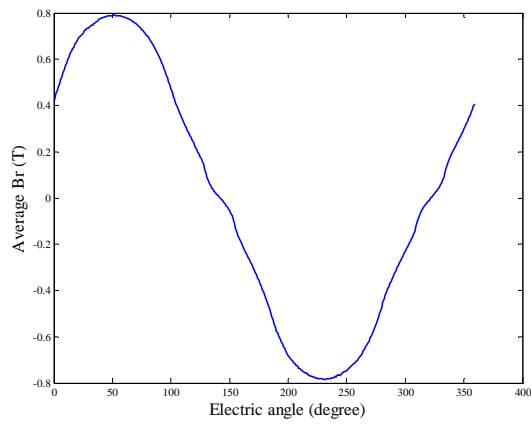


Fig. III.32 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 2. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.003	0
1	0.764	-0.900905
2	0.002	-2.365147
3	0.047	-2.652516
4	0.001	1.734428
5	0.024	-1.374921
6	0.000	0.314447
7	0.005	3.134894
8	0.000	-1.690462
9	0.002	-2.257898
10	0.000	1.882659
11	0.008	2.626818
12	0.000	-1.27762
13	0.007	-2.412654
14	0.000	1.32684
15	0.003	-0.947364
16	0.000	0.623316
17	0.002	0.248743
18	0.000	-0.956962
19	0.002	1.407928
20	0.000	1.872308

Harmonic order	Amplitude	Phase
0	0.004	0
1	0.603	1.611373
2	0.003	-2.302682
3	0.035	1.979119
4	0.001	1.706155
5	0.063	0.831469
6	0.001	-3.075667
7	0.040	1.48558
8	0.001	1.825914
9	0.017	1.24894
10	0.000	1.349983
11	0.020	1.408377
12	0.000	0.168484
13	0.016	2.211208
14	0.000	-2.439539
15	0.010	-3.062827
16	0.000	1.415801
17	0.006	-1.68098
18	0.000	1.305941
19	0.005	-0.277254
20	0.000	0.207552

Fig. III.33 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 3. Phase in radian.

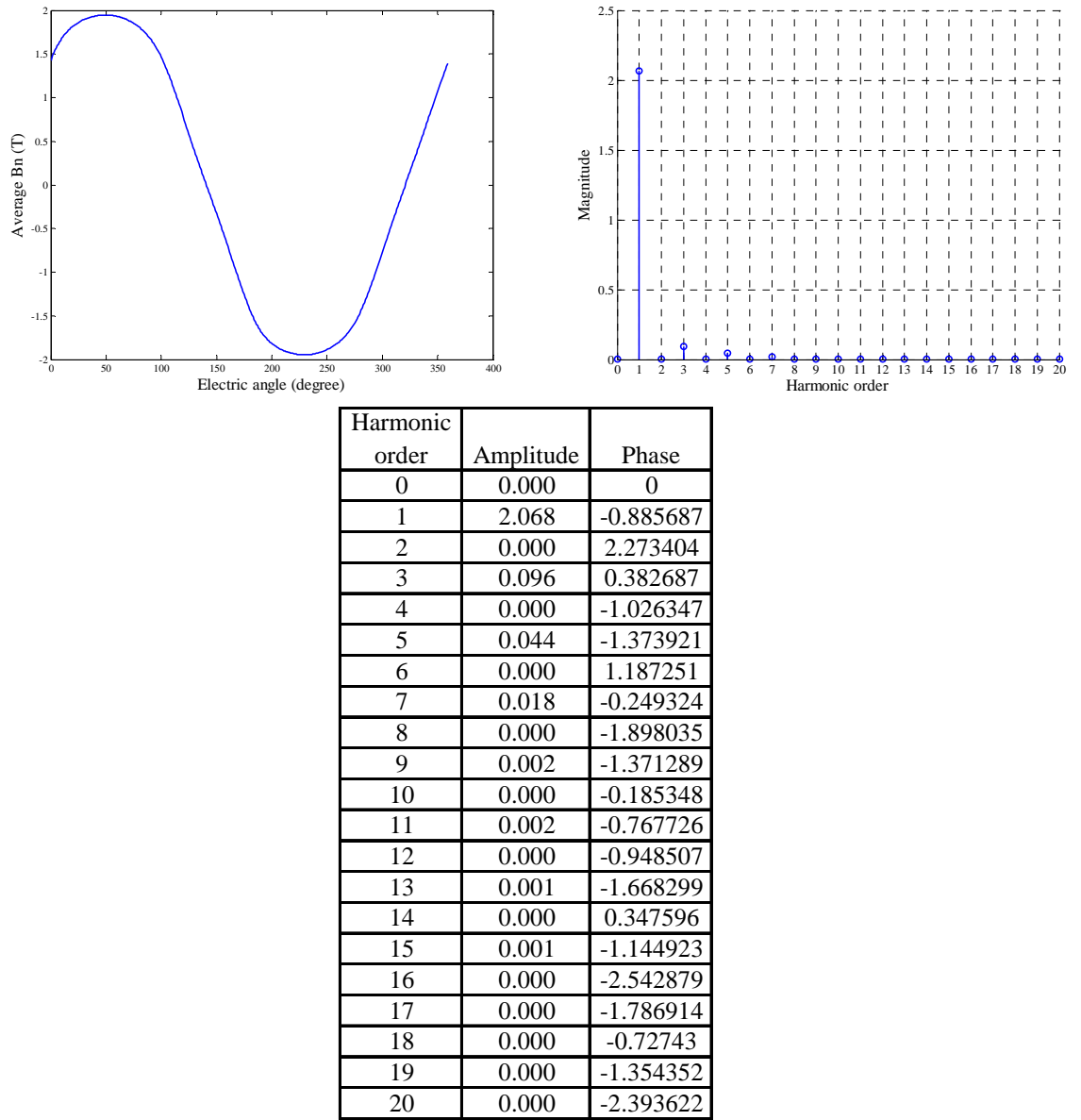


Fig. III.34 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 4. Phase in radian.

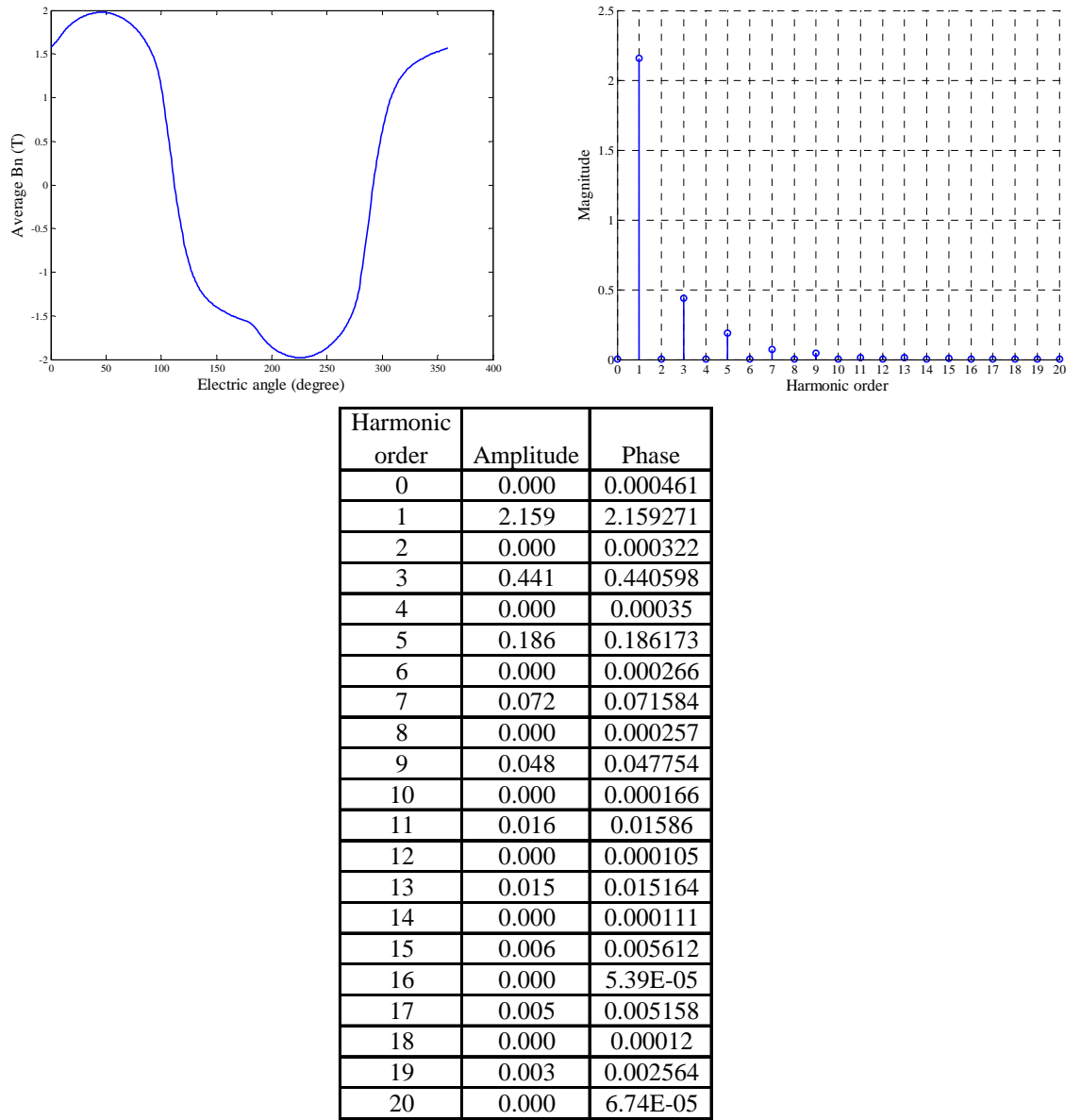
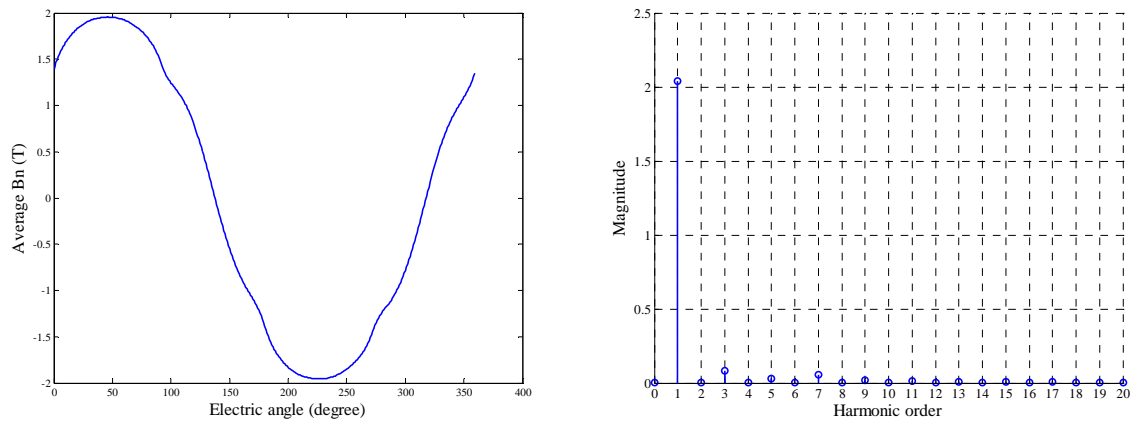


Fig. III.35 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 5. Phase in radian.





Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	2.038	-0.834618
2	0.001	-2.284442
3	0.080	0.430248
4	0.001	-0.24317
5	0.029	2.46121
6	0.000	1.955442
7	0.054	-2.370791
8	0.000	-0.44295
9	0.020	-0.938522
10	0.000	1.150839
11	0.012	-2.479492
12	0.000	2.878865
13	0.009	-0.919978
14	0.000	-1.258217
15	0.005	-2.565955
16	0.000	0.426715
17	0.006	-0.904813
18	0.000	2.143848
19	0.002	-3.048108
20	0.000	-2.434924

Fig. III.36 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 6. Phase in radian.

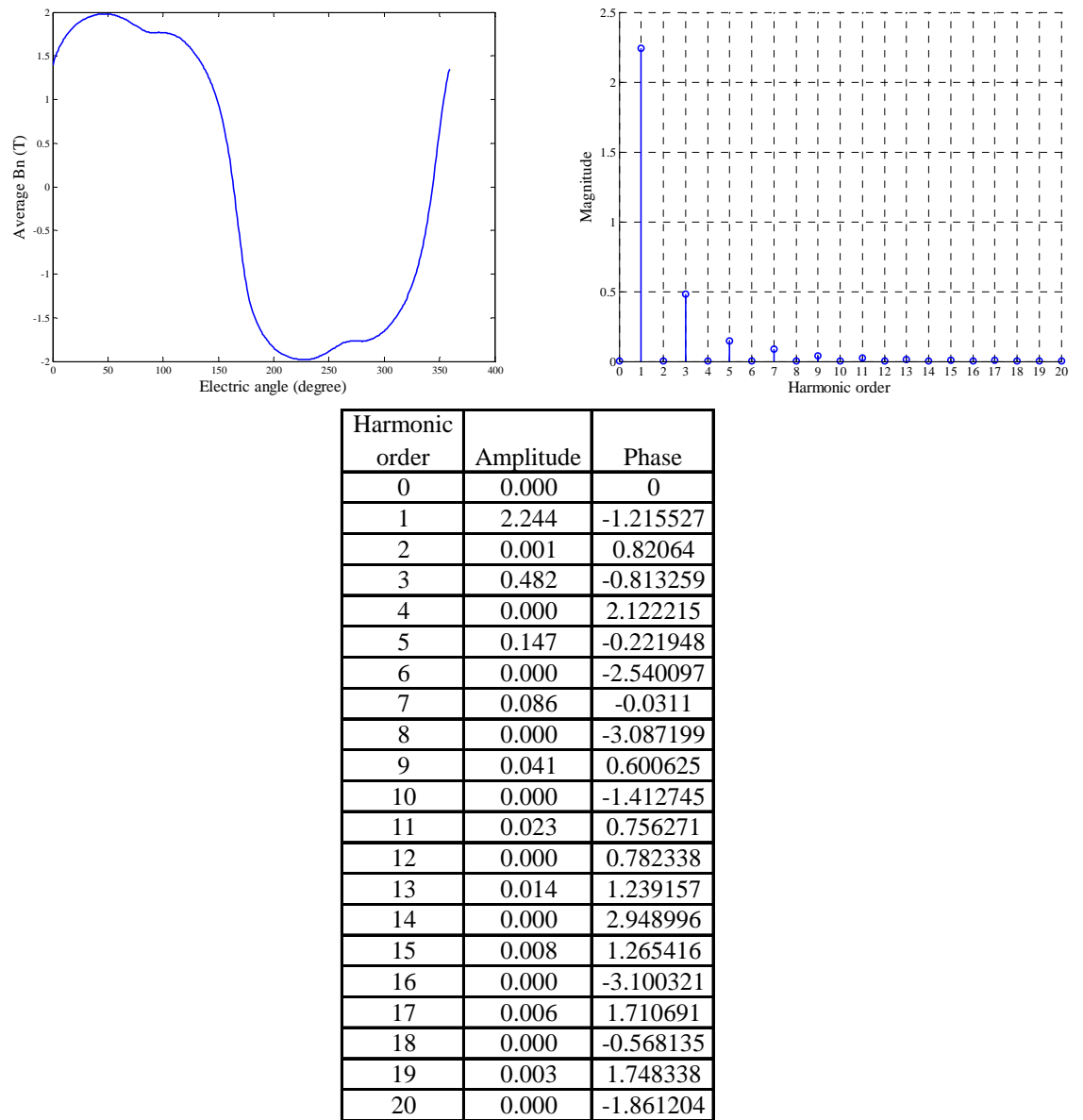


Fig. III.37 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 7. Phase in radian.

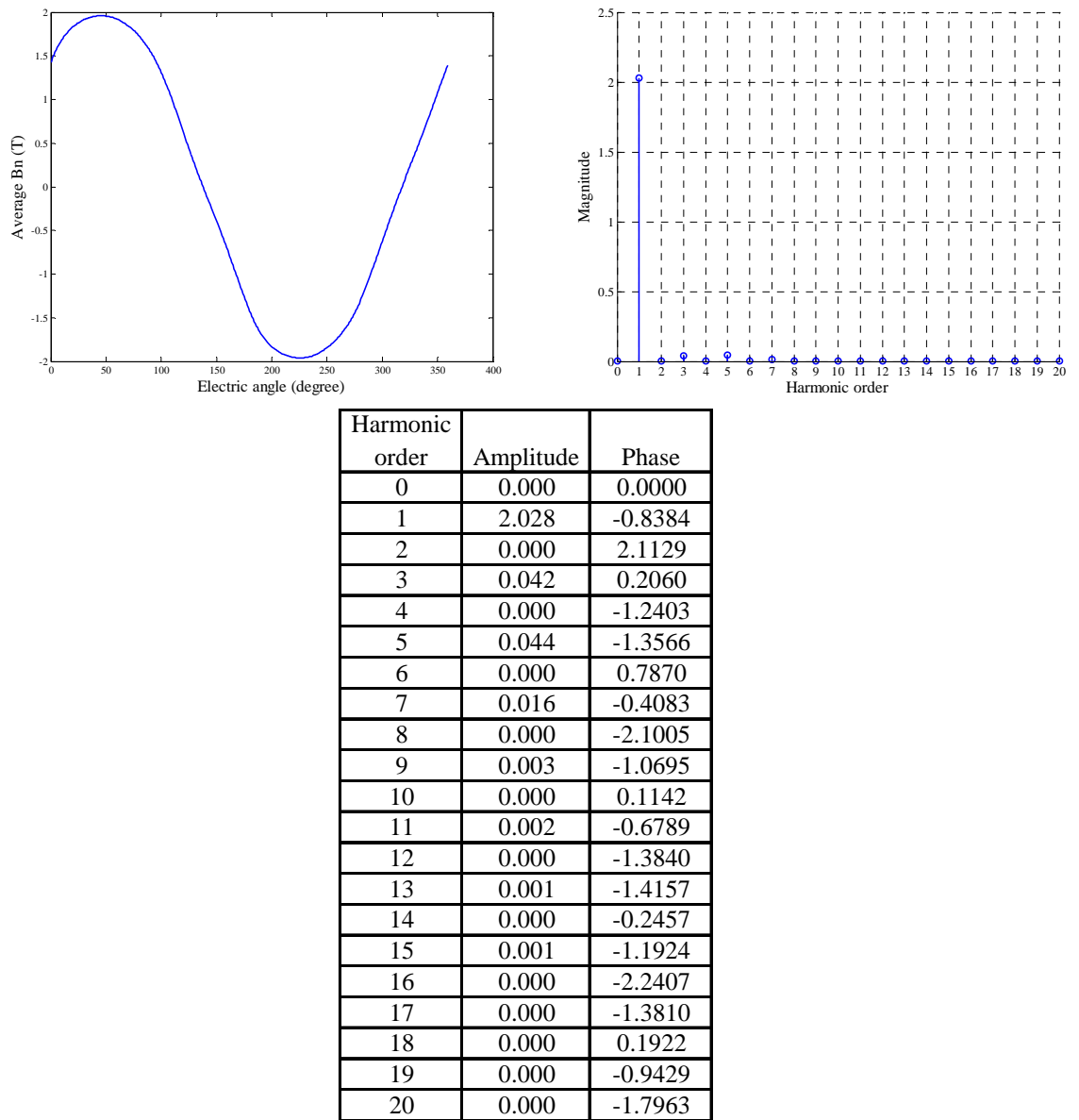
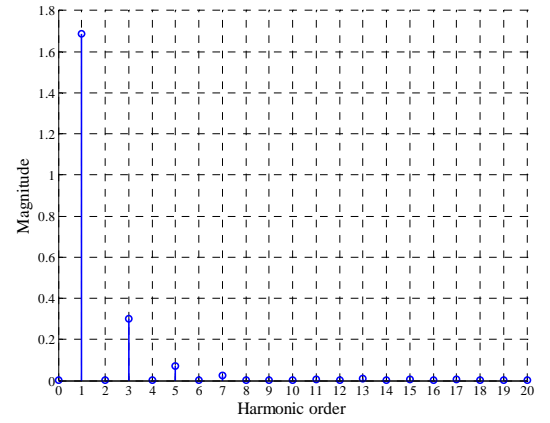
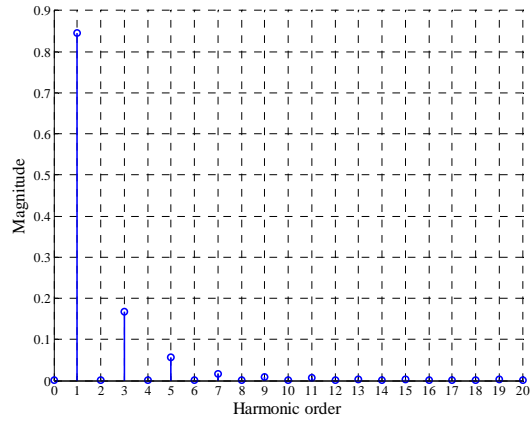
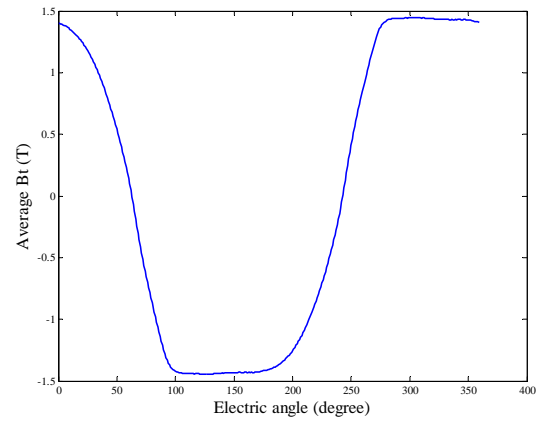
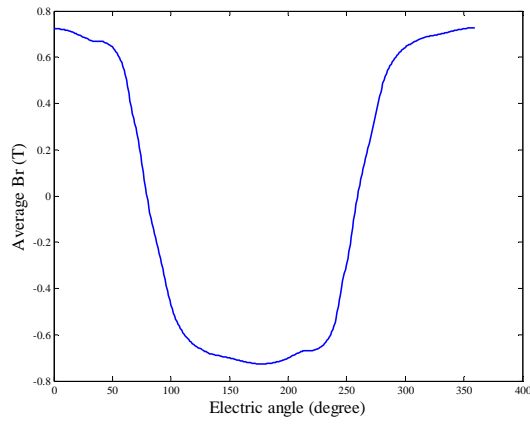


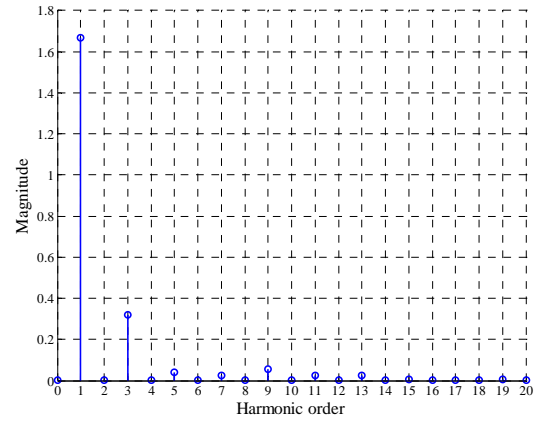
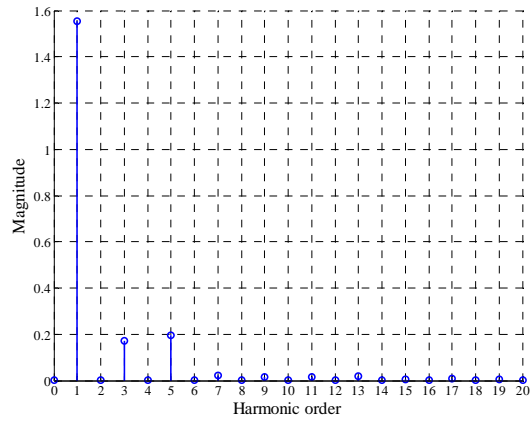
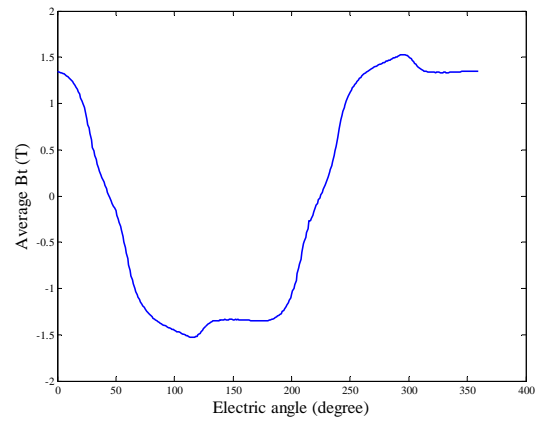
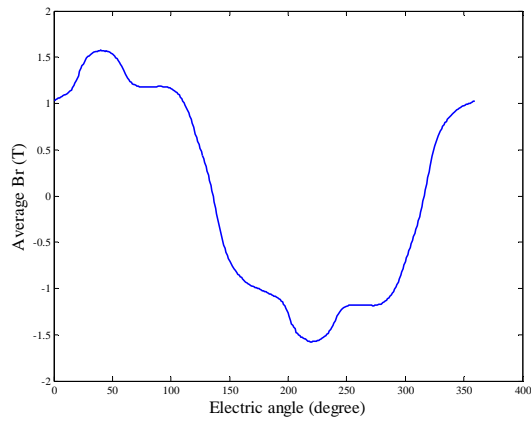
Fig. III.38 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 8. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	0.845	0.152408
2	0.000	-0.881607
3	0.167	-2.622511
4	0.000	1.41632
5	0.056	0.956099
6	0.000	-3.006082
7	0.016	-0.987877
8	0.000	0.741337
9	0.008	-2.51726
10	0.000	-2.012423
11	0.006	1.451047
12	0.000	2.098726
13	0.003	-1.705244
14	0.000	1.681058
15	0.002	1.270928
16	0.000	0.064908
17	0.001	-2.281101
18	0.000	2.671421
19	0.003	0.785528
20	0.000	-0.350161

Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	1.689	0.529179
2	0.001	2.529221
3	0.300	-1.74534
4	0.000	2.10202
5	0.070	1.651086
6	0.000	0.64462
7	0.024	-1.209898
8	0.000	-3.110684
9	0.003	2.053656
10	0.000	0.860879
11	0.006	2.610165
12	0.000	2.755698
13	0.008	-0.204487
14	0.000	1.798077
15	0.007	-2.918645
16	0.000	-2.021232
17	0.004	0.593618
18	0.000	1.119755
19	0.004	-2.035349
20	0.000	2.20854

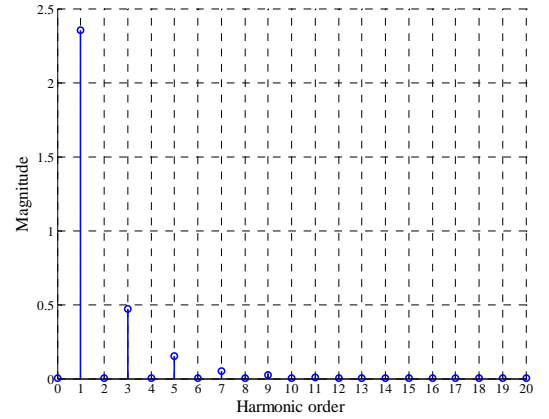
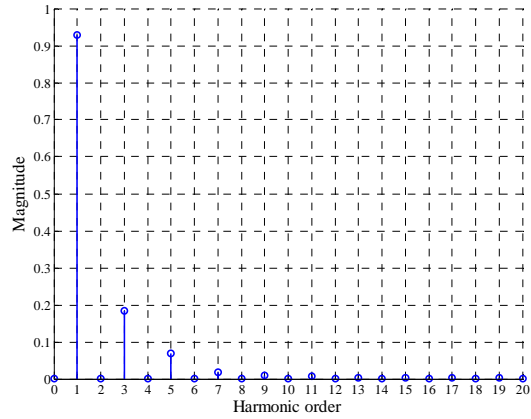
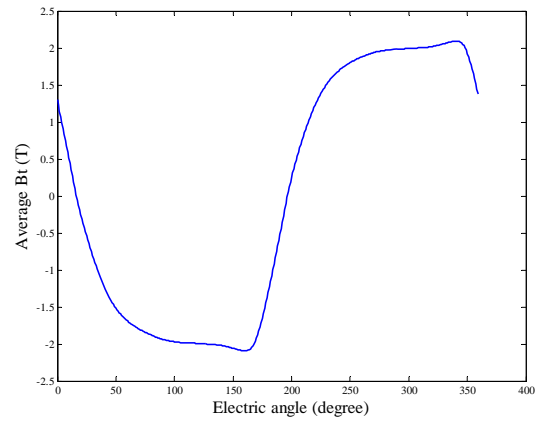
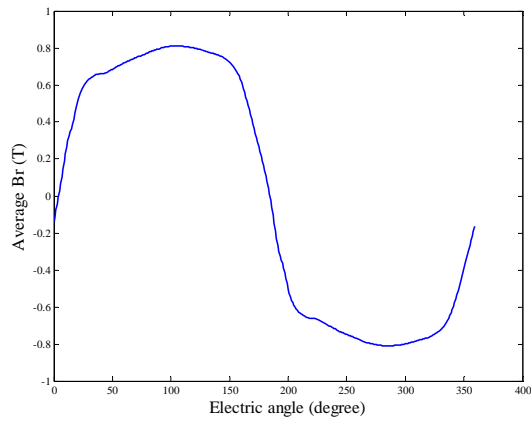
Fig. III.39 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 9. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.001	0
1	1.553	-0.815603
2	0.001	-1.563779
3	0.173	0.735257
4	0.000	-2.570722
5	0.197	2.680039
6	0.000	2.957903
7	0.023	2.201602
8	0.000	2.291041
9	0.015	-0.961473
10	0.000	2.632893
11	0.014	0.339215
12	0.000	-0.578198
13	0.020	0.672316
14	0.000	1.05416
15	0.007	-2.139263
16	0.000	-1.0744
17	0.009	-2.088494
18	0.001	-2.995498
19	0.004	-0.648309
20	0.000	1.999749

Harmonic order	Amplitude	Phase
0	0.001	3.141593
1	1.668	0.814271
2	0.000	-0.04783
3	0.318	-0.884863
4	0.000	-0.038946
5	0.039	-1.294661
6	0.000	-1.036025
7	0.026	-0.116938
8	0.000	2.493224
9	0.055	-2.133572
10	0.000	0.277533
11	0.024	2.626682
12	0.001	-1.078808
13	0.025	1.262041
14	0.000	-2.9555
15	0.007	-1.056365
16	0.001	1.825657
17	0.002	-0.75489
18	0.000	-0.139037
19	0.005	0.283164
20	0.000	-0.954736

Fig. III.40 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 10. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	0.930	-1.654402
2	0.000	-0.414096
3	0.183	-1.603815
4	0.000	-0.735038
5	0.069	-1.804331
6	0.000	0.553534
7	0.018	-2.313694
8	0.000	0.464534
9	0.010	2.594287
10	0.000	2.79741
11	0.008	2.068678
12	0.000	-1.223195
13	0.004	2.293046
14	0.000	0.830463
15	0.003	2.857601
16	0.000	-1.286909
17	0.003	2.872218
18	0.000	-0.341325
19	0.003	2.991351
20	0.000	-1.088121

Harmonic order	Amplitude	Phase
0	0.000	0
1	2.352	1.200028
2	0.000	-2.880705
3	0.473	0.784556
4	0.000	1.86328
5	0.151	0.798403
6	0.000	2.944606
7	0.053	1.141884
8	0.000	2.656034
9	0.025	1.41982
10	0.000	2.727356
11	0.008	1.787313
12	0.000	1.634793
13	0.003	3.13793
14	0.000	1.902782
15	0.005	-2.537149
16	0.000	-2.498562
17	0.005	-2.567443
18	0.000	0.482938
19	0.004	-2.488594
20	0.000	0.037071

Fig. III.41 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 11. Phase in radian.

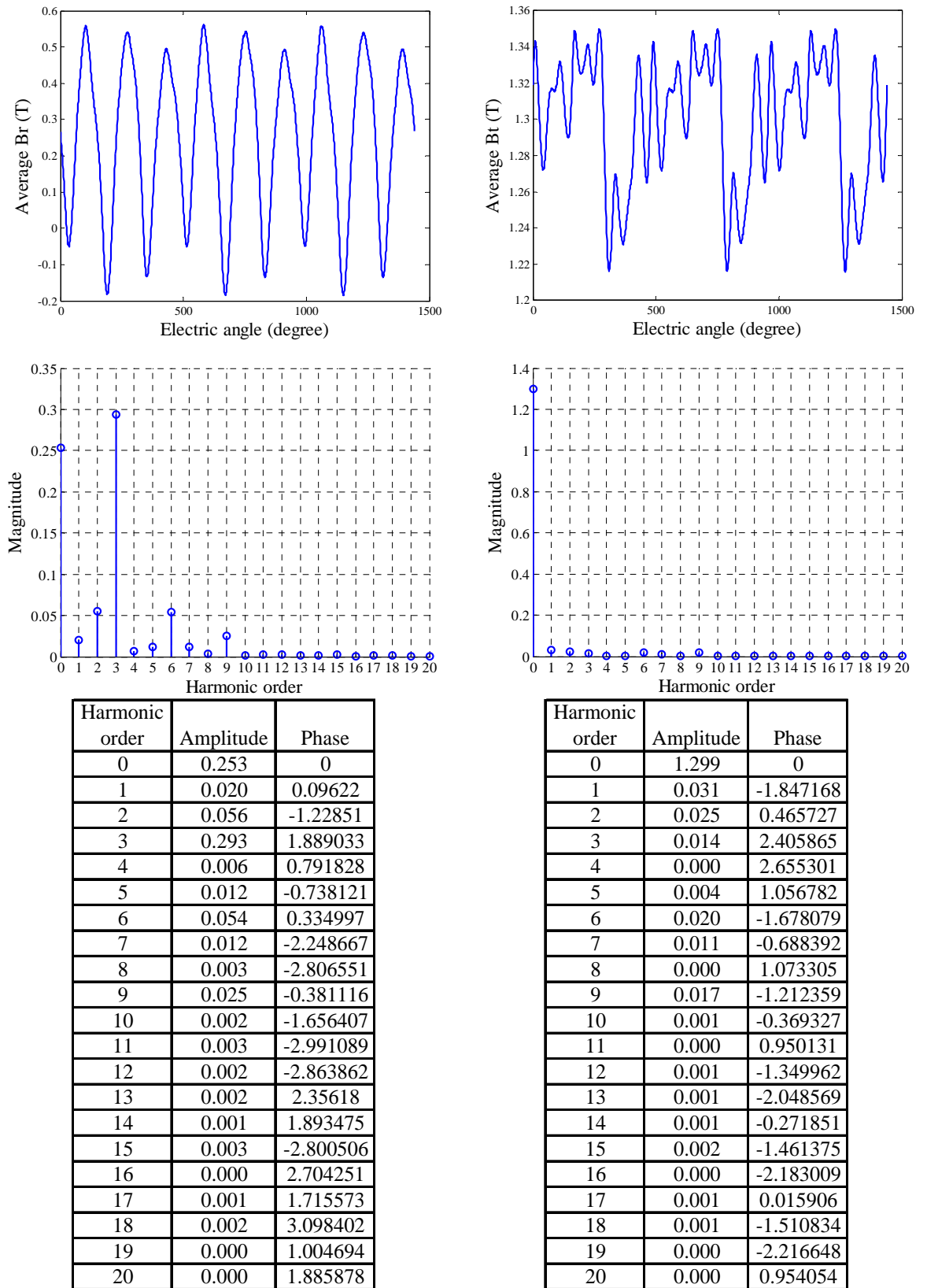


Fig. III.42 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 12. Phase in radian.

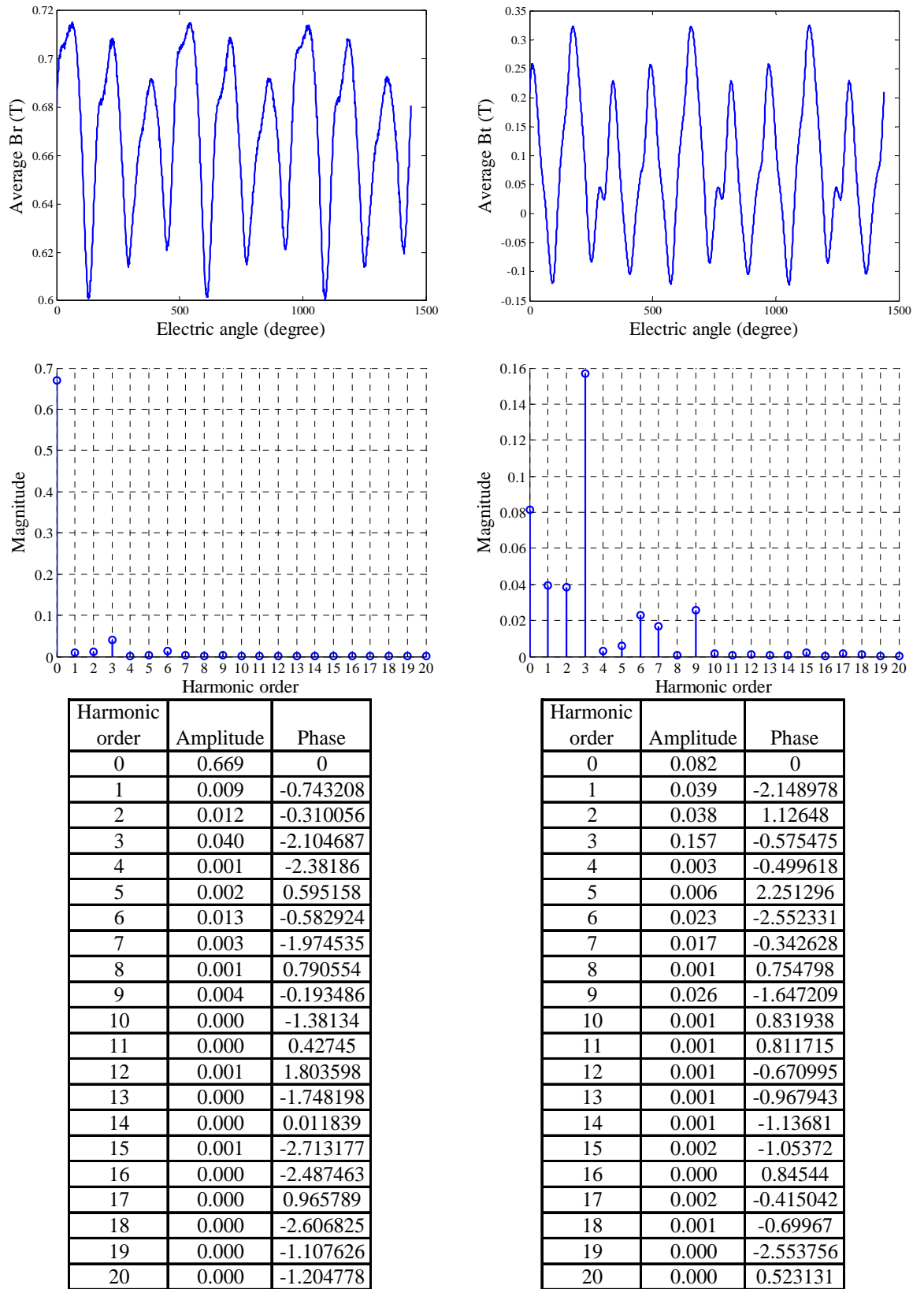


Fig. III.43 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 13. Phase in radian.



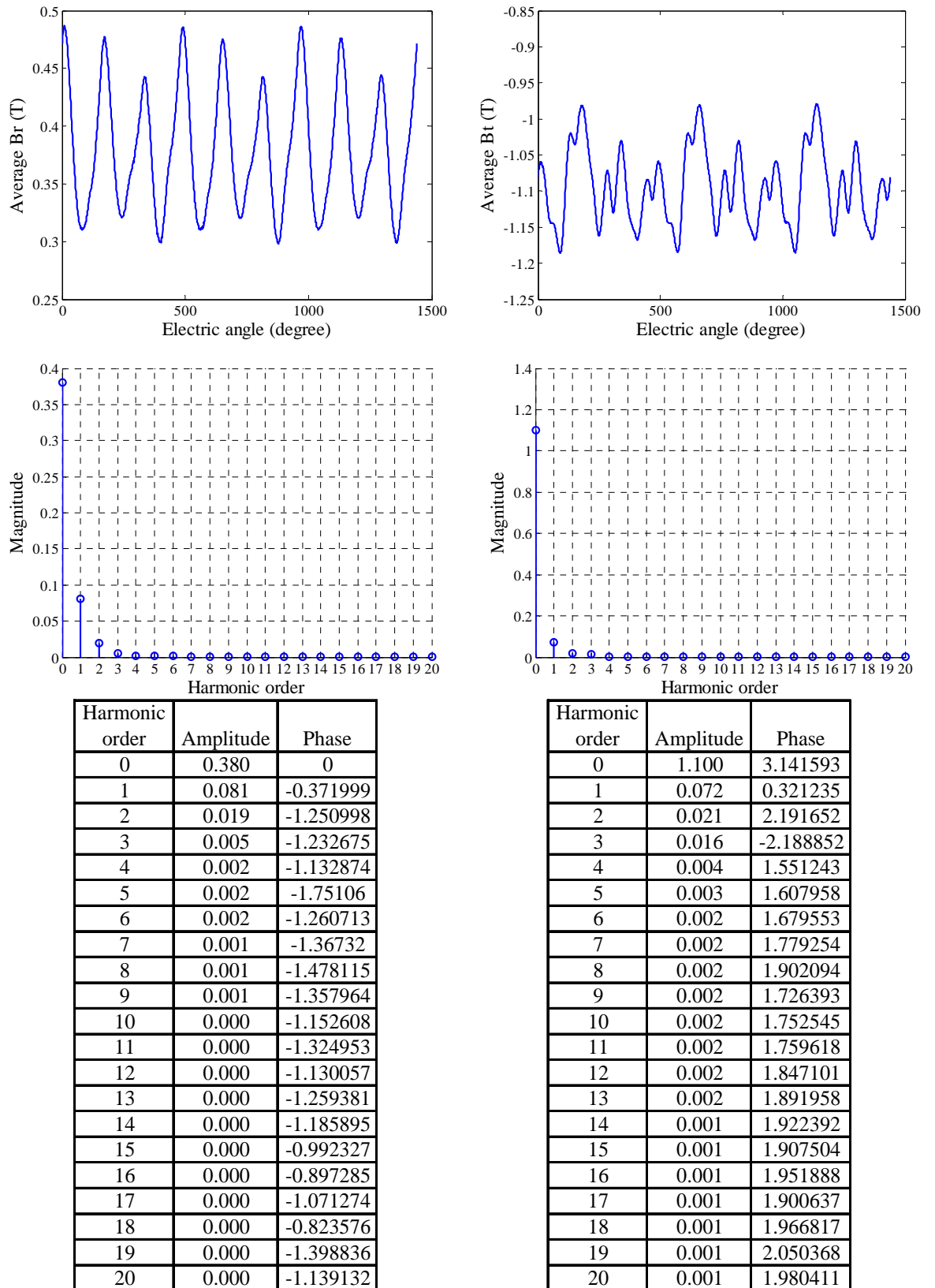


Fig. III.44 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 14. Phase in radian.

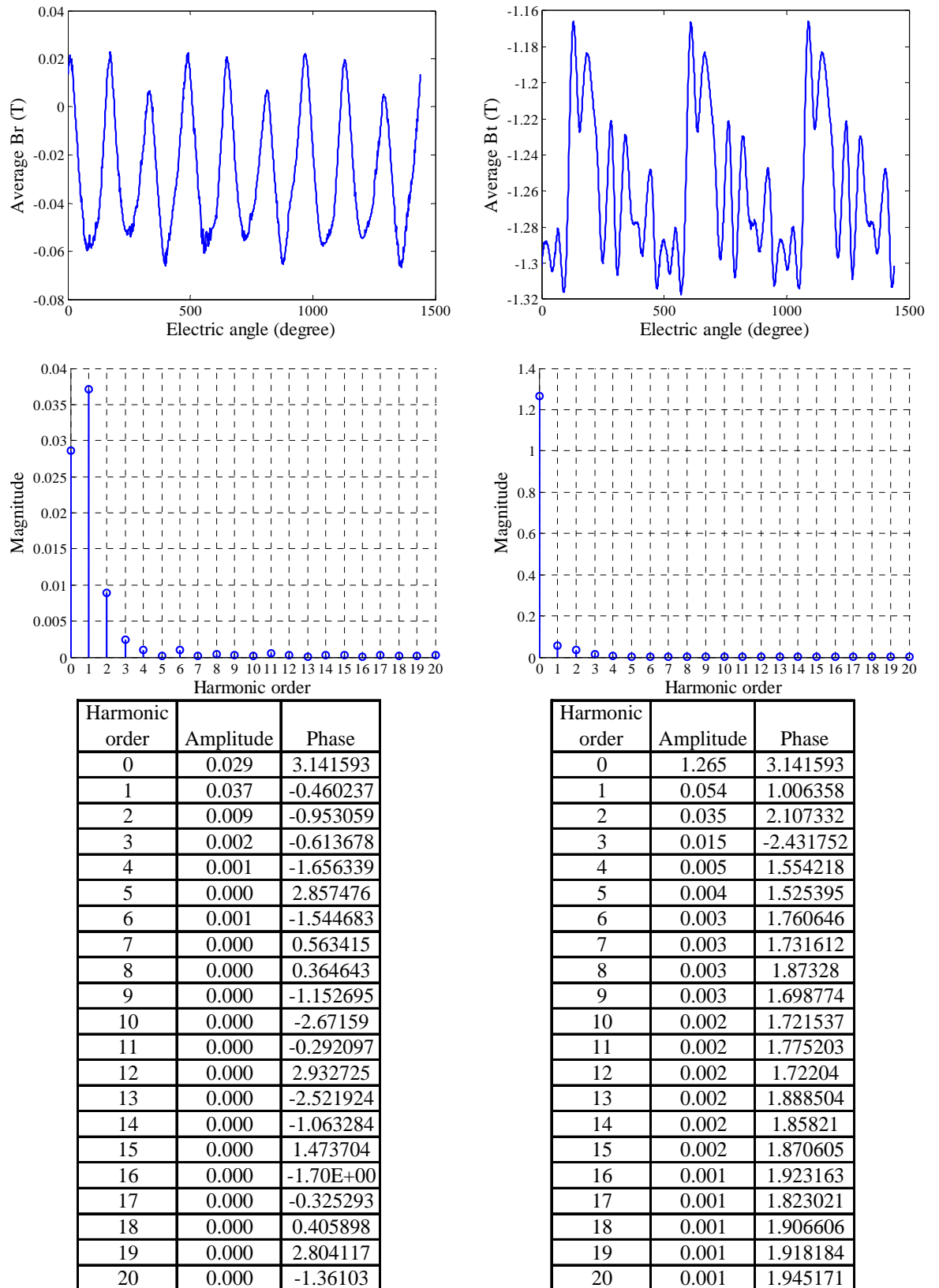


Fig. III.45 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 15. Phase in radian.

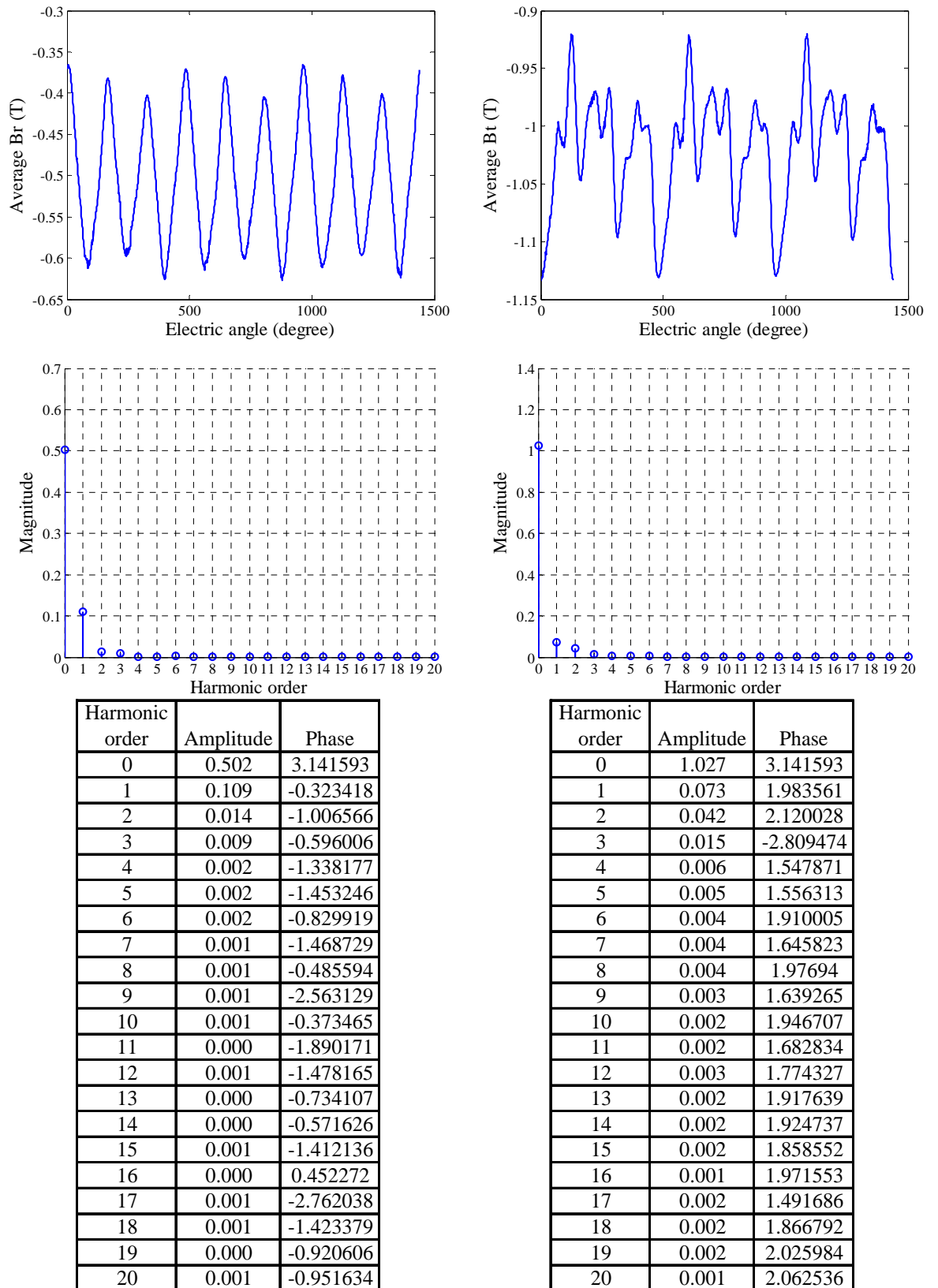
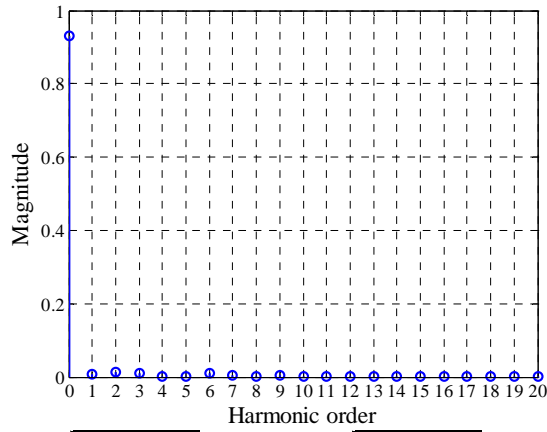
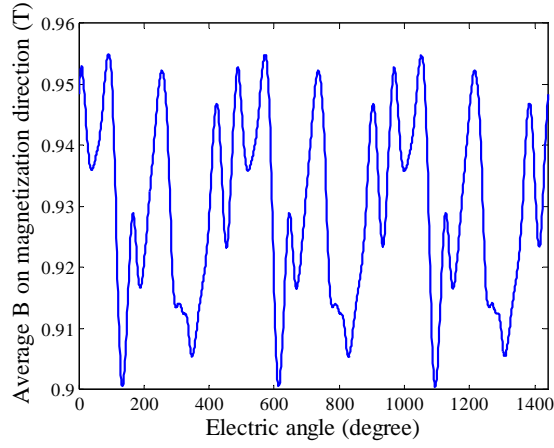


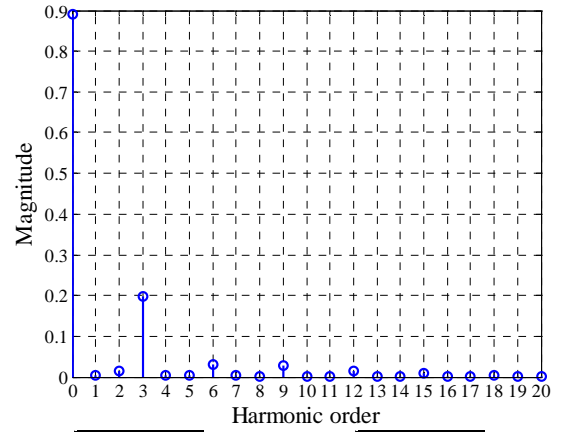
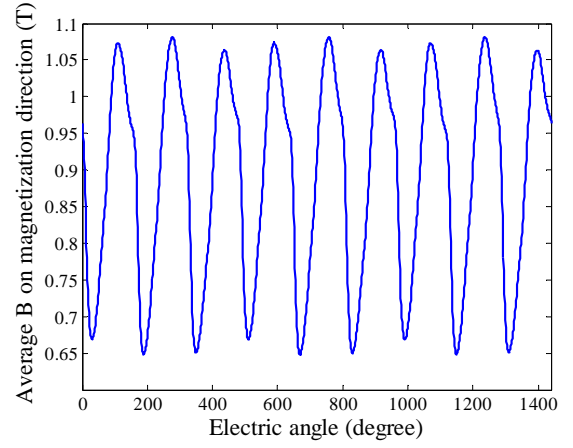
Fig. III.46 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 16. Phase in radian.

Test block on magnet:



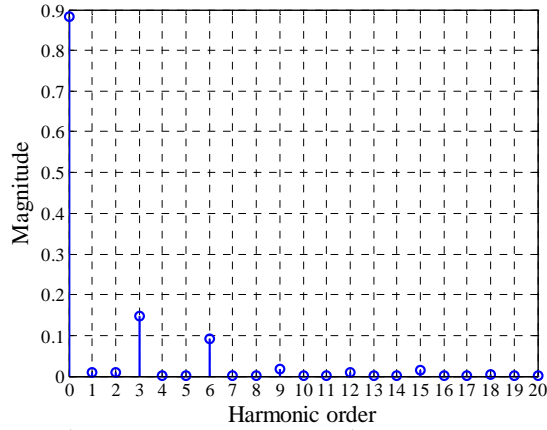
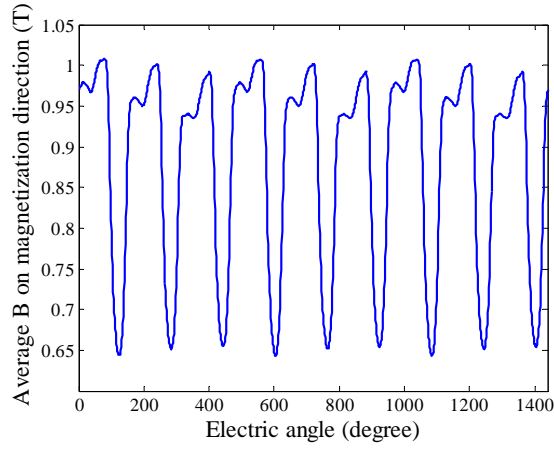
Harmonic order	Magnitude	Phase
0	0.930	0
1	0.008	-0.680971
2	0.012	-0.3131
3	0.011	2.872249
4	0.000	-2.270454
5	0.002	0.688426
6	0.010	-0.936447
7	0.003	-1.867869
8	0.000	0.951619
9	0.005	0.038974
10	0.000	-1.116419
11	0.000	1.236536
12	0.000	0.337309
13	0.000	-2.448456
14	0.000	0.553287
15	0.000	-2.668966
16	0.000	-2.133536
17	0.000	1.197898
18	0.000	-2.345561
19	0.000	-1.841047
20	0.000	-2.16579

Fig. III.47 Average B in magnetization direction of testing block 1 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



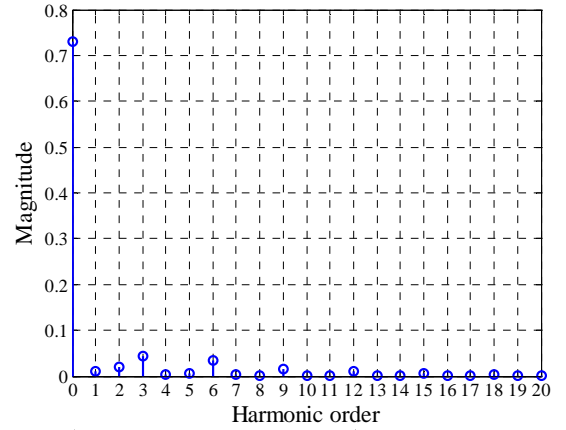
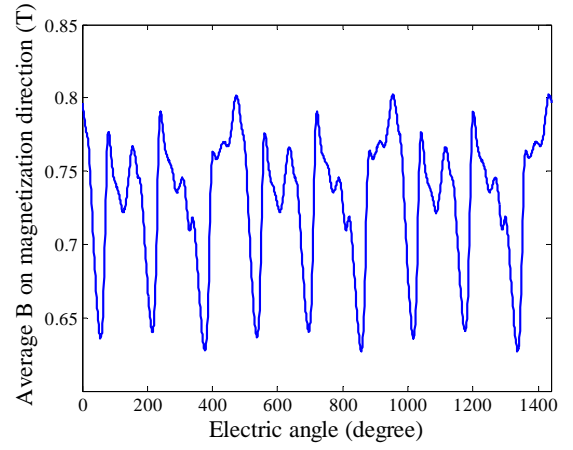
Harmonic order	Magnitude	Phase
0	0.890	0
1	0.004	-1.093524
2	0.015	-1.147535
3	0.198	1.626704
4	0.004	1.657261
5	0.003	0.090312
6	0.029	0.306953
7	0.003	-1.842833
8	0.000	2.528718
9	0.028	0.027095
10	0.001	-1.395616
11	0.001	-2.701079
12	0.013	-0.495223
13	0.000	2.131269
14	0.000	1.773804
15	0.008	-0.476201
16	0.000	-2.324596
17	0.000	2.094327
18	0.005	-1.128339
19	0.000	1.154254
20	0.000	2.189999

Fig. III.48 Average B in magnetization direction of testing block 2 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



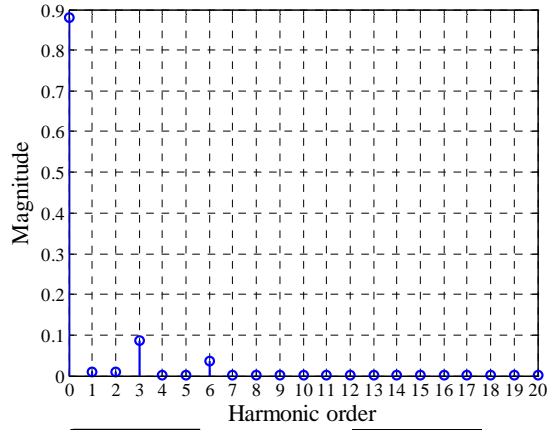
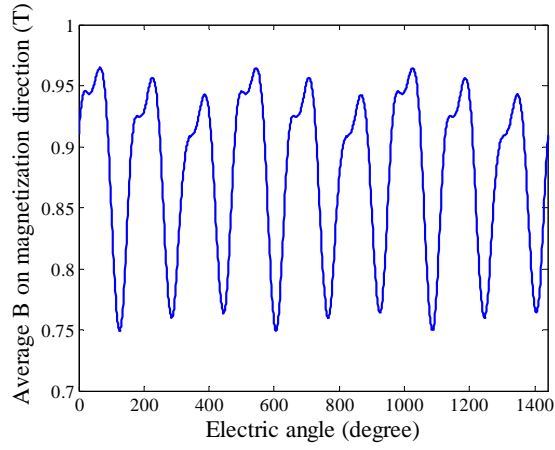
Harmonic order	Magnitude	Phase
0	0.883	0
1	0.009	-0.835705
2	0.010	-0.256725
3	0.147	-1.752168
4	0.002	-2.247002
5	0.002	0.436602
6	0.091	-0.064032
7	0.002	-1.873602
8	0.001	0.858429
9	0.016	1.414352
10	0.000	-1.310554
11	0.000	0.339948
12	0.010	-0.103425
13	0.000	-3.004595
14	0.000	1.545997
15	0.014	1.519217
16	0.000	0.343401
17	0.000	1.19775
18	0.003	-2.979438
19	0.000	-2.195475
20	0.000	-2.865022

Fig. III.49 Average B in magnetization direction of testing block 3 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



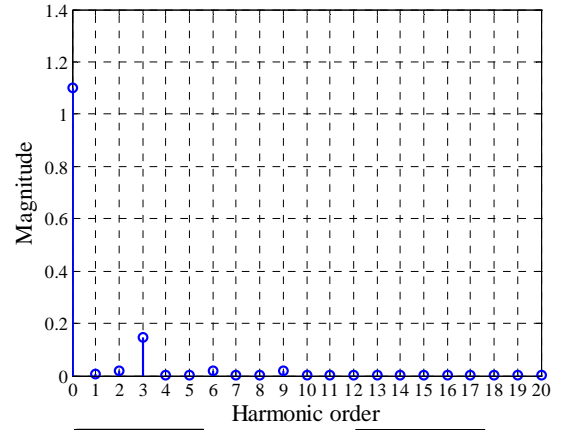
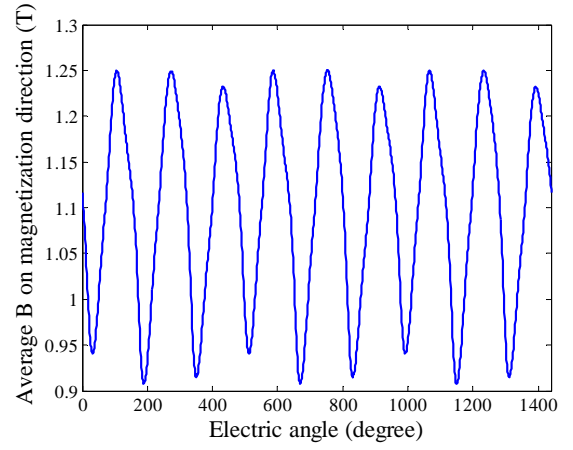
Harmonic order	Magnitude	Phase
0	0.730	0
1	0.010	-0.30191
2	0.019	0.279052
3	0.044	1.224713
4	0.003	-0.241909
5	0.006	1.300032
6	0.035	-0.688916
7	0.005	-1.670162
8	0.002	0.008729
9	0.016	2.87286
10	0.001	-1.16924
11	0.000	0.976817
12	0.010	0.585884
13	0.001	-2.054125
14	0.000	-0.272084
15	0.007	-3.103814
16	0.000	3.080461
17	0.000	0.656479
18	0.004	0.864163
19	0.000	-2.240908
20	0.000	-2.90022

Fig. III.50 Average B in magnetization direction of testing block 4 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



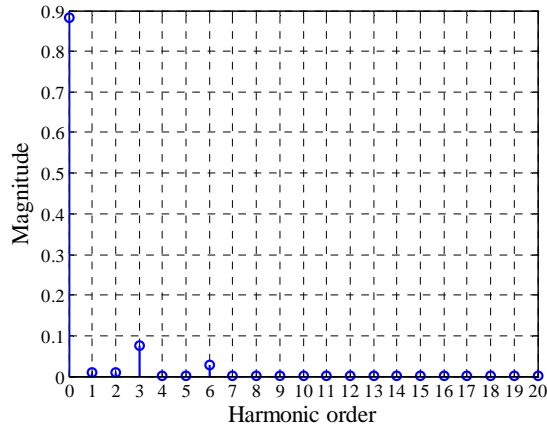
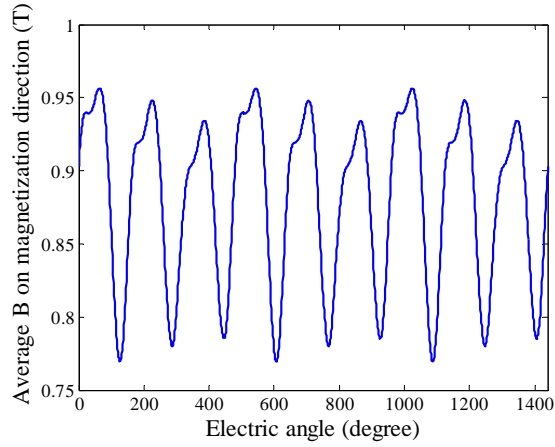
Harmonic order	Magnitude	Phase
0	0.881	0
1	0.009	-0.802106
2	0.010	-0.265833
3	0.087	-1.857315
4	0.002	-2.252186
5	0.002	0.577649
6	0.035	-0.175836
7	0.002	-1.908492
8	0.001	0.855824
9	0.003	0.259275
10	0.000	-1.484943
11	0.000	0.321643
12	0.000	0.173006
13	0.000	-2.492699
14	0.000	0.36991
15	0.001	2.003041
16	0.000	-2.298361
17	0.000	1.058172
18	0.000	-2.500806
19	0.000	-2.18045
20	0.000	-1.364781

Fig. III.51 Average B in magnetization direction of testing block 5 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



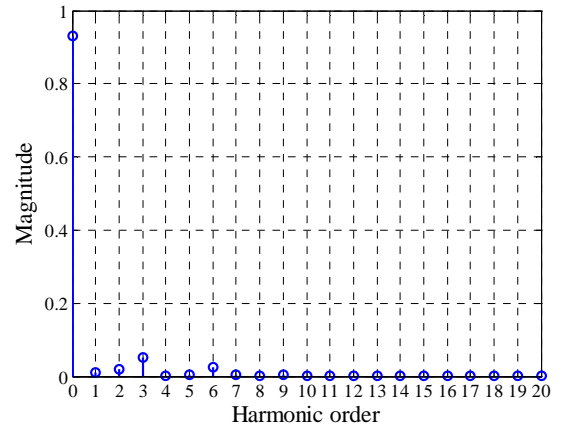
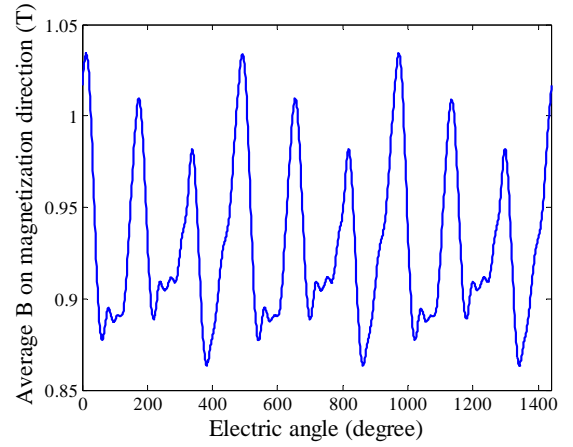
Harmonic order	Magnitude	Phase
0	1.100	0
1	0.005	-0.536569
2	0.017	-1.155237
3	0.147	1.761124
4	0.003	1.430536
5	0.003	-0.32694
6	0.017	0.380097
7	0.004	-2.052217
8	0.001	-2.948737
9	0.018	-0.094974
10	0.001	-1.428084
11	0.001	-2.822873
12	0.003	-1.284438
13	0.000	2.6757
14	0.000	1.849876
15	0.002	-1.107521
16	0.000	-2.968999
17	0.000	1.818545
18	0.001	-2.285931
19	0.000	0.798933
20	0.000	1.971091

Fig. III.52 Average B in magnetization direction of testing block 6 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



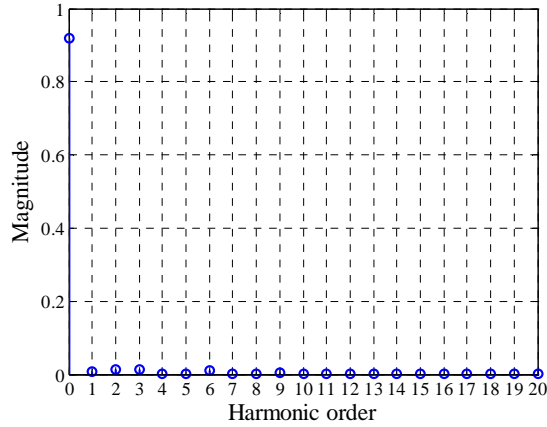
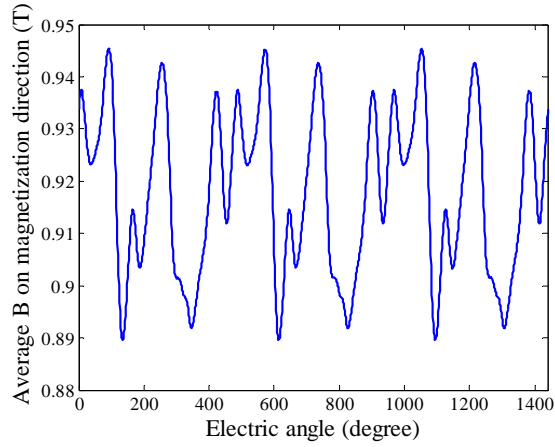
Harmonic order	Magnitude	Phase
0	0.882	0
1	0.009	-0.792632
2	0.010	-0.266374
3	0.075	-1.894445
4	0.002	-2.263224
5	0.002	0.601203
6	0.027	-0.235013
7	0.002	-1.902772
8	0.000	0.852901
9	0.003	-0.139126
10	0.000	-1.431727
11	0.000	0.364013
12	0.000	1.705246
13	0.000	-2.439305
14	0.000	0.347677
15	0.000	2.69479
16	0.000	-2.270623
17	0.000	1.00787
18	0.000	-2.414608
19	0.000	-2.172592
20	0.000	-1.481563

Fig. III.53 Average B in magnetization direction of testing block 7 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



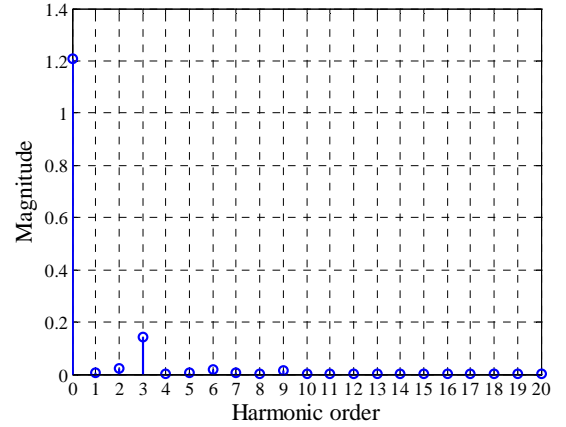
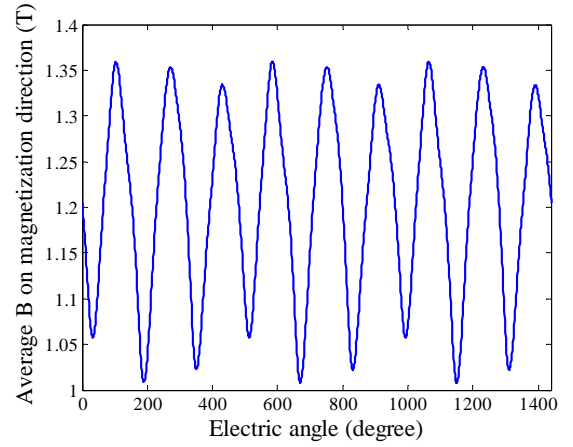
Harmonic order	Magnitude	Phase
0	0.930	0
1	0.009	-0.426982
2	0.020	0.293541
3	0.052	-0.222271
4	0.003	0.05812
5	0.005	1.411168
6	0.024	-1.155095
7	0.004	-1.569891
8	0.001	0.276551
9	0.006	-2.526981
10	0.001	-0.680236
11	0.001	0.969019
12	0.002	1.625542
13	0.001	-1.707932
14	0.000	0.17079
15	0.002	-2.891234
16	0.000	-1.765704
17	0.000	0.783842
18	0.000	2.402492
19	0.000	-1.945207
20	0.000	-1.940475

Fig. III.54 Average B in magnetization direction of testing block 8 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



Harmonic order	Magnitude	Phase
0	0.918	0
1	0.008	-0.691536
2	0.012	-0.305607
3	0.013	2.800956
4	0.000	-2.28671
5	0.002	0.702204
6	0.009	-0.968448
7	0.003	-1.868715
8	0.000	0.934392
9	0.004	0.147927
10	0.000	-1.125866
11	0.000	1.174735
12	0.000	0.259478
13	0.000	-2.43517
14	0.000	0.526499
15	0.000	-2.713157
16	0.000	-2.133668
17	0.000	1.194665
18	0.000	-2.332389
19	0.000	-1.887026
20	0.000	-2.046156

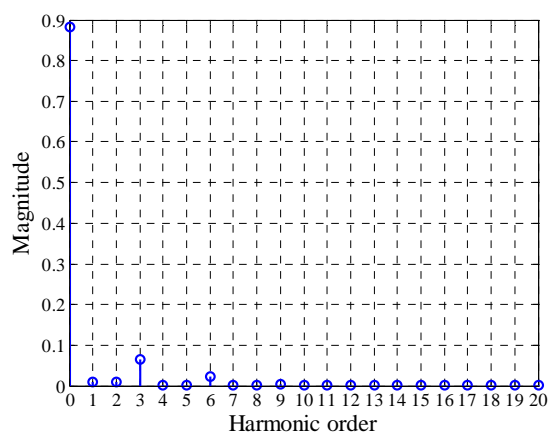
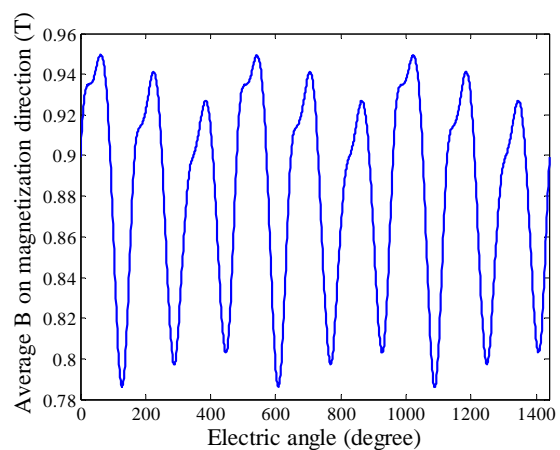
Fig. III.55 Average B in magnetization direction of testing block 9 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



Harmonic order	Magnitude	Phase
0	1.207	0
1	0.007	-0.22482
2	0.022	-1.179959
3	0.145	1.892194
4	0.003	1.183479
5	0.004	-0.547503
6	0.019	0.379712
7	0.005	-2.146093
8	0.001	-2.936036
9	0.014	-0.173382
10	0.001	-1.524801
11	0.001	-2.849045
12	0.001	-2.050172
13	0.001	2.540898
14	0.000	1.82671
15	0.001	-2.171818
16	0.000	-3.011845
17	0.001	1.729221
18	0.001	-2.876602
19	0.000	0.584121
20	0.000	1.931696

Fig. III.56 Average B in magnetization direction of testing block 10 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.





Harmonic order	Magnitude	Phase
0	0.883	0
1	0.009	-0.783902
2	0.011	-0.270571
3	0.065	-1.937788
4	0.001	-2.280473
5	0.002	0.614468
6	0.021	-0.313594
7	0.002	-1.899291
8	0.000	0.838374
9	0.003	-0.309403
10	0.000	-1.330048
11	0.000	0.418118
12	0.000	2.711032
13	0.000	-2.446
14	0.000	0.292519
15	0.000	-3.036223
16	0.000	-2.079544
17	0.000	1.027121
18	0.000	-2.407147
19	0.000	-2.191105
20	0.000	-1.216633

Fig. III.57 Average B in magnetization direction of testing block 11 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.

### III.4 Flux density results of 9-phase with 1000 A-t peak phase current excitation

Test block on steel:

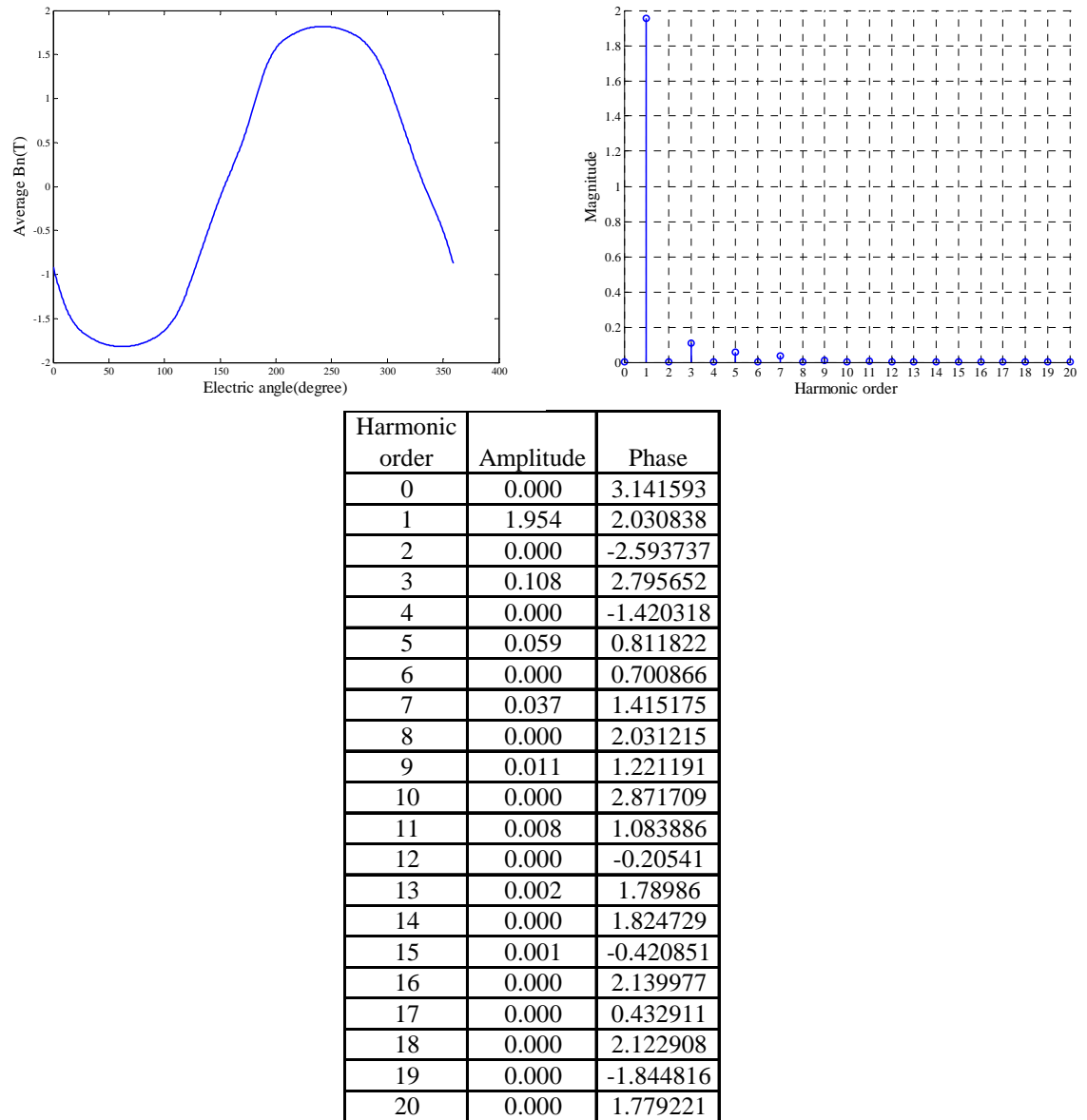


Fig. III.58 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 1. Phase in radian.

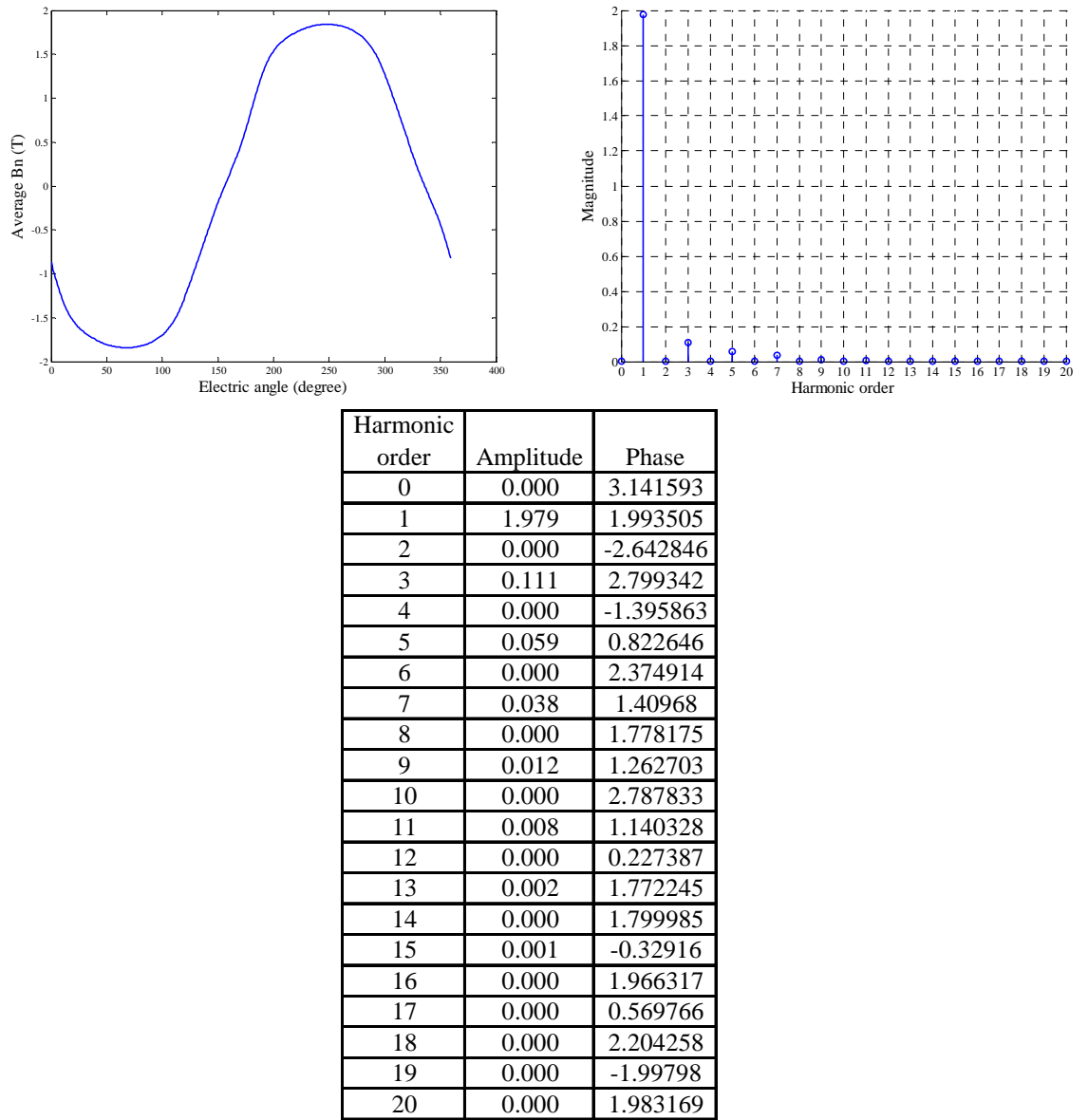
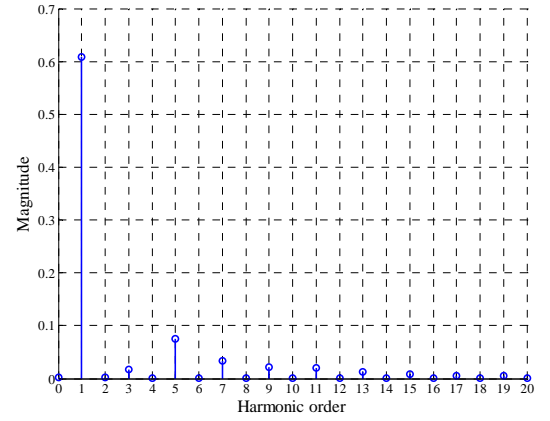
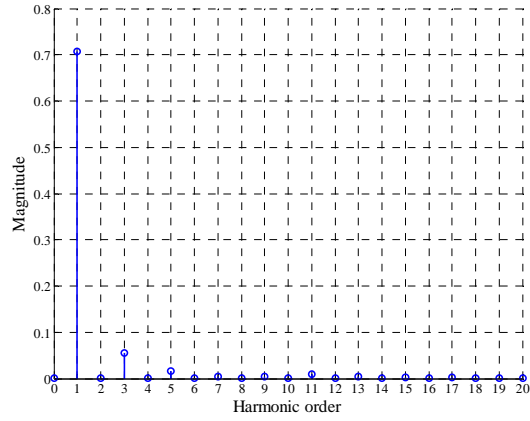
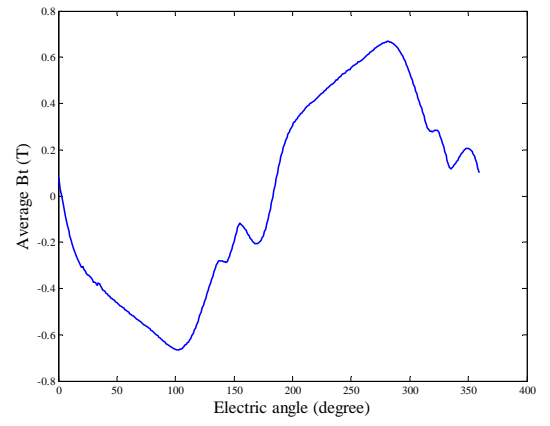
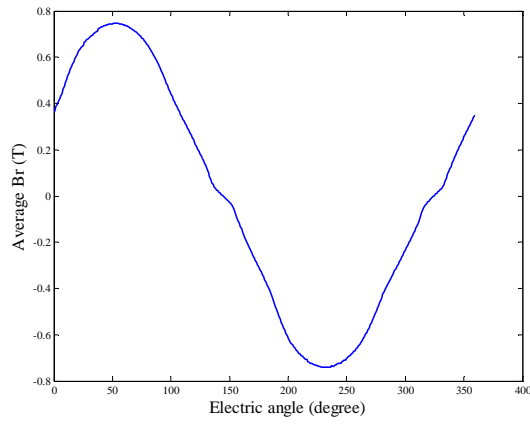


Fig. III.59 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 2. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.001	0
1	0.707	-0.935646
2	0.002	-2.421007
3	0.055	-2.64265
4	0.001	1.73828
5	0.016	-1.599401
6	0.000	0.503656
7	0.004	-2.692094
8	0.000	-1.273759
9	0.004	1.056168
10	0.000	1.725735
11	0.009	2.29188
12	0.000	-1.385951
13	0.004	-3.139491
14	0.000	1.448142
15	0.002	-2.103648
16	0.000	0.625489
17	0.002	-0.269507
18	0.000	-0.764644
19	0.002	0.933776
20	0.000	-3.086387

Harmonic order	Amplitude	Phase
0	0.002	0
1	0.609	1.625416
2	0.003	-2.317335
3	0.017	-1.321263
4	0.001	1.750584
5	0.075	0.490448
6	0.001	-3.079444
7	0.034	1.112314
8	0.001	1.766349
9	0.021	0.539381
10	0.000	1.031894
11	0.020	0.920934
12	0.000	0.115555
13	0.013	1.558136
14	0.000	-2.215908
15	0.008	2.636108
16	0.000	1.75937
17	0.006	-2.194492
18	0.000	1.347015
19	0.006	-0.95632
20	0.000	0.202073

Fig. III.60 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 3. Phase in radian.

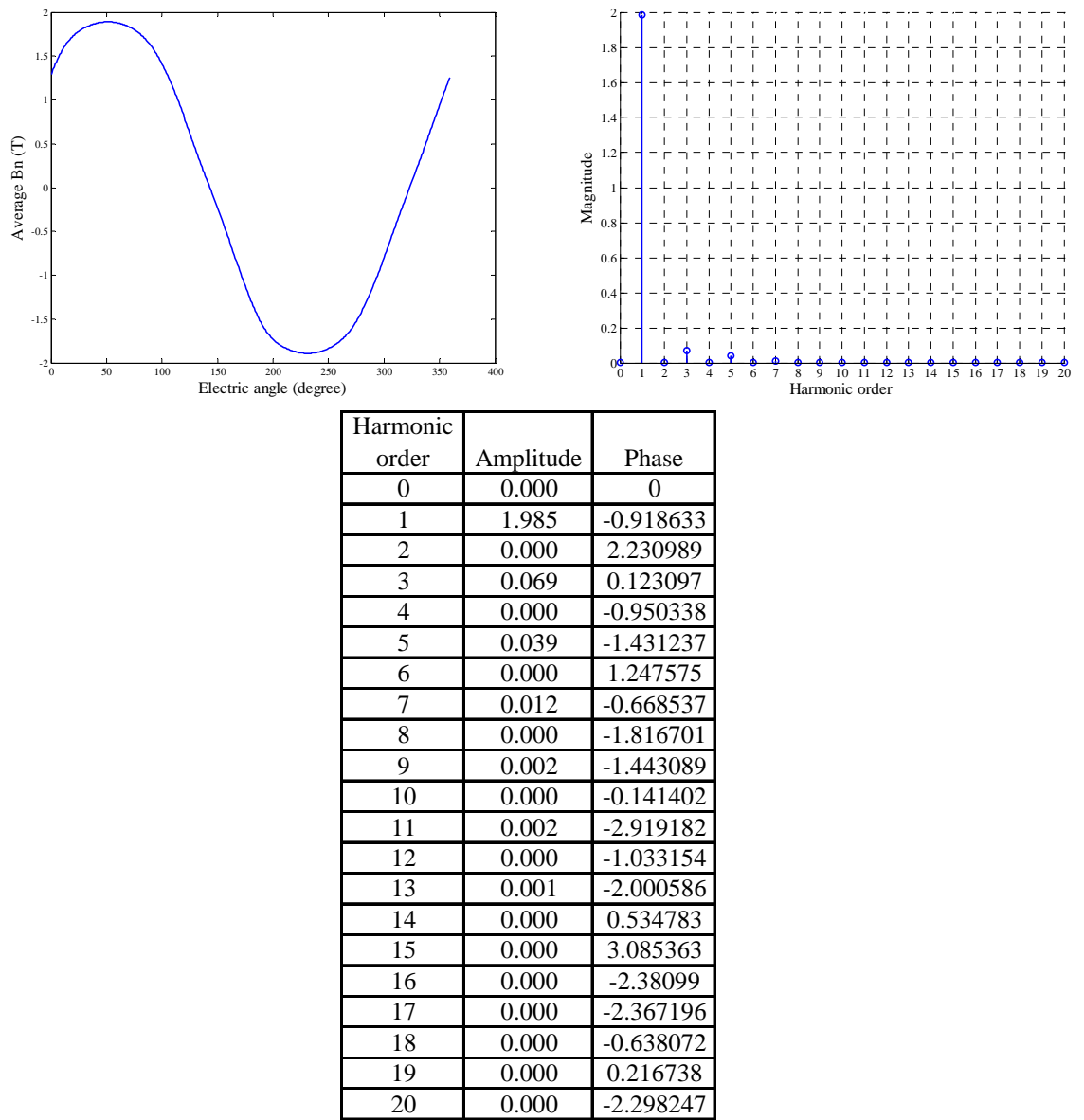
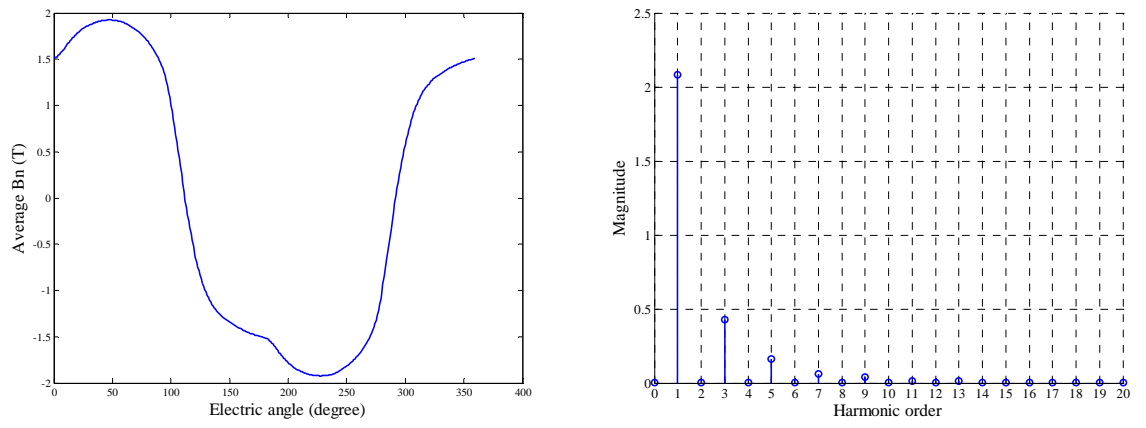


Fig. III.61 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 4. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.000	0
1	2.081	-0.482599
2	0.000	2.944354
3	0.428	2.304169
4	0.000	-0.739149
5	0.163	-1.843736
6	0.000	1.454015
7	0.064	0.995231
8	0.000	-2.142764
9	0.038	-2.973143
10	0.000	0.039284
11	0.011	0.309315
12	0.000	-3.118299
13	0.011	2.722596
14	0.000	-0.98176
15	0.003	-0.250953
16	0.000	1.468724
17	0.004	2.12962
18	0.000	-2.668676
19	0.000	0.034906
20	0.000	-1.394405

Fig. III.62 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 5. Phase in radian.

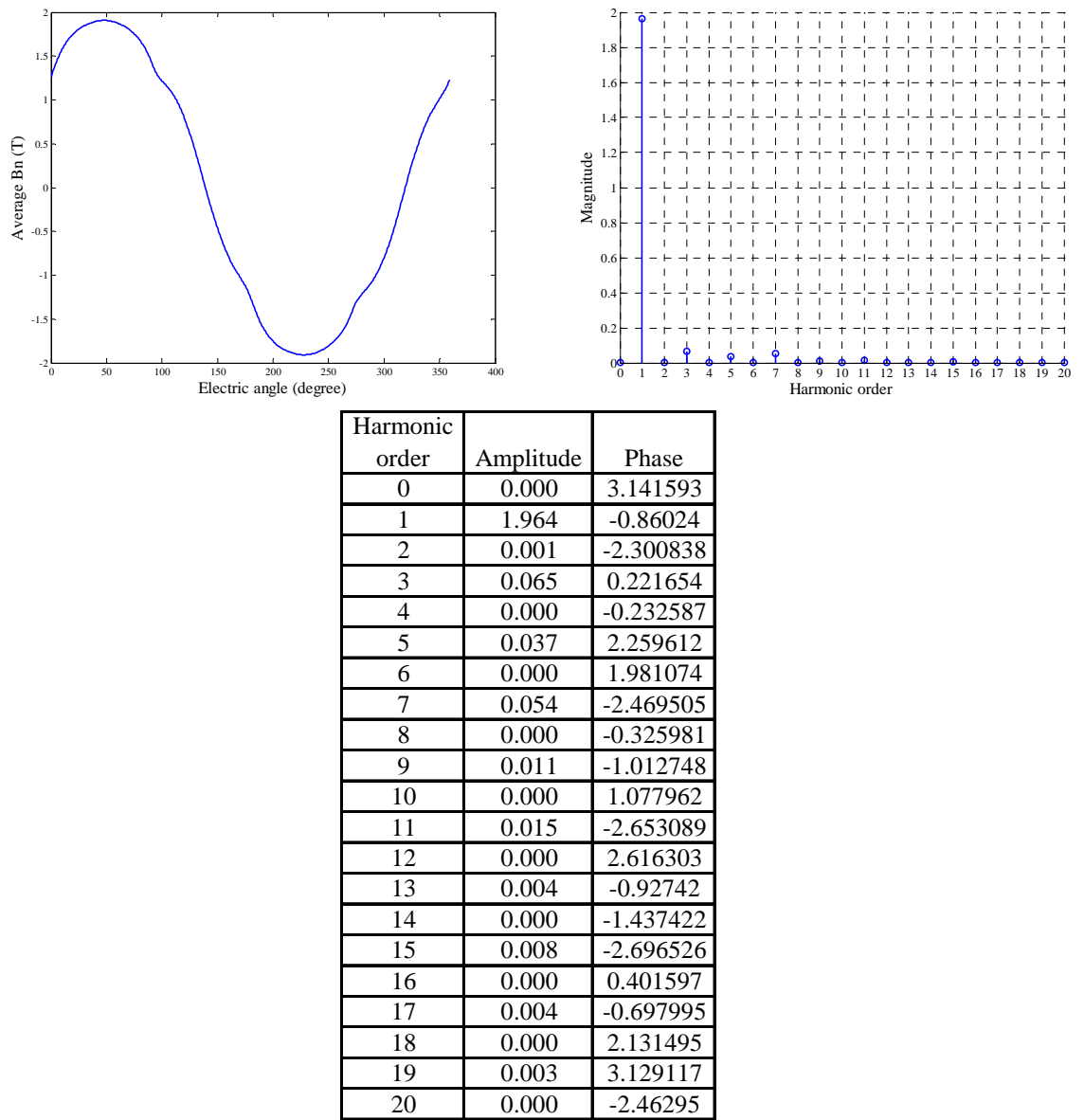


Fig. III.63 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 6. Phase in radian.

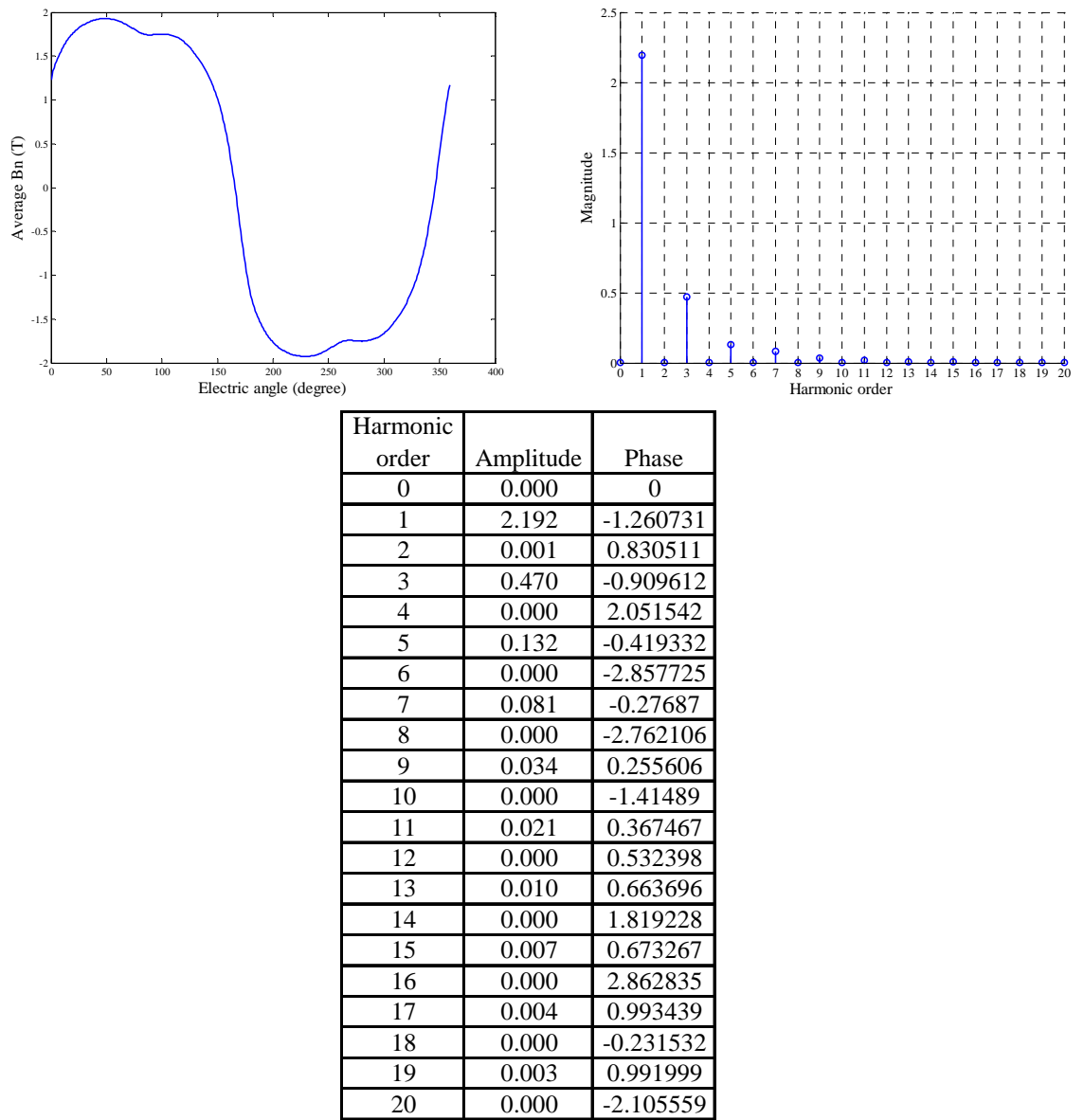


Fig. III.64 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 7. Phase in radian.



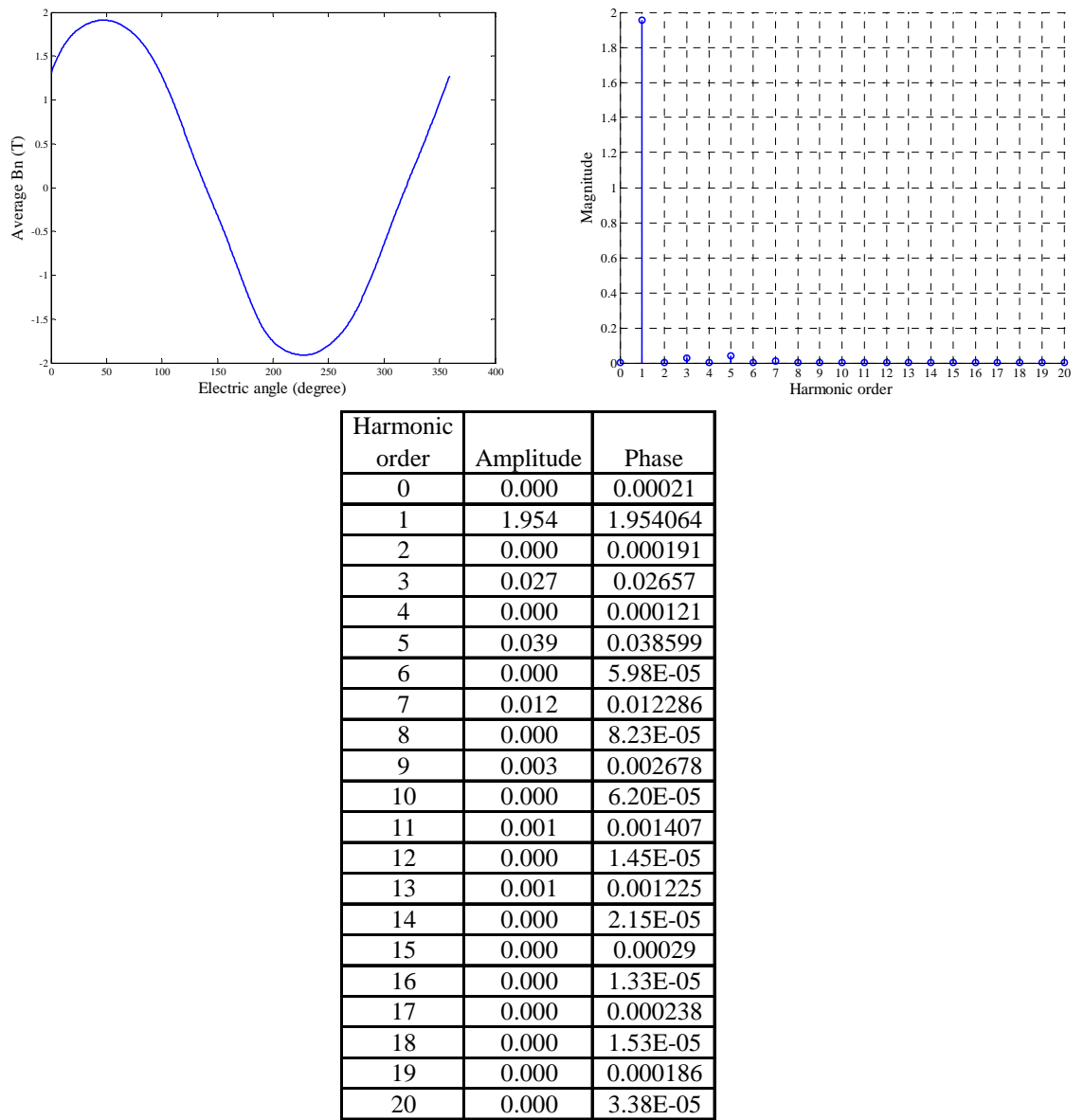
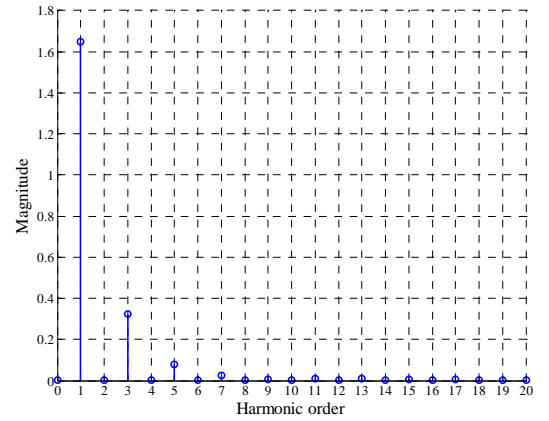
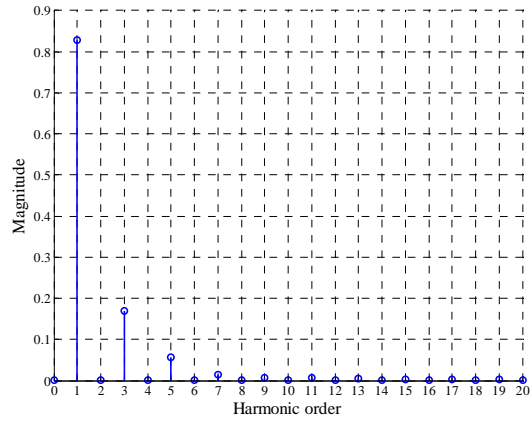
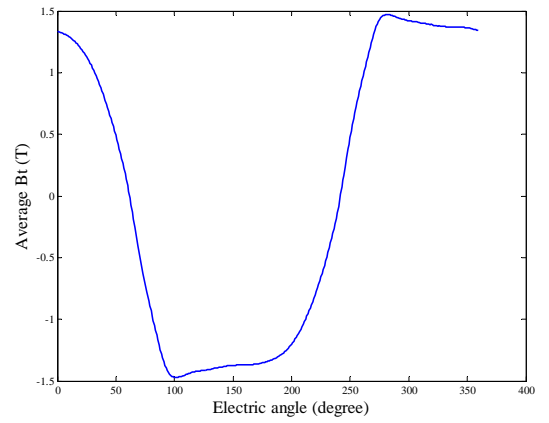
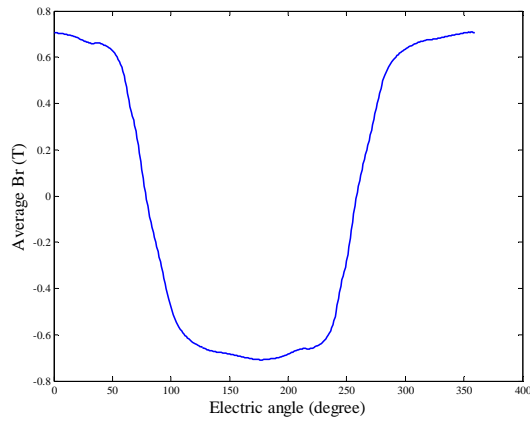


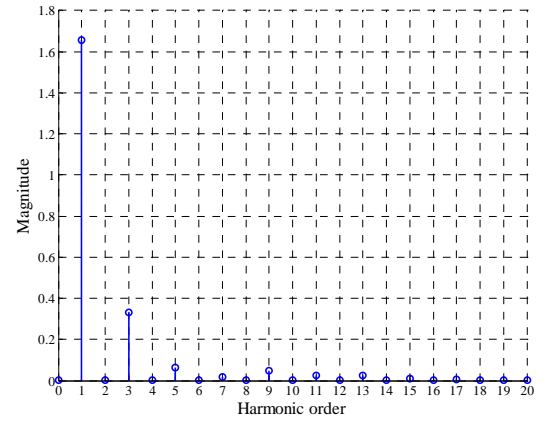
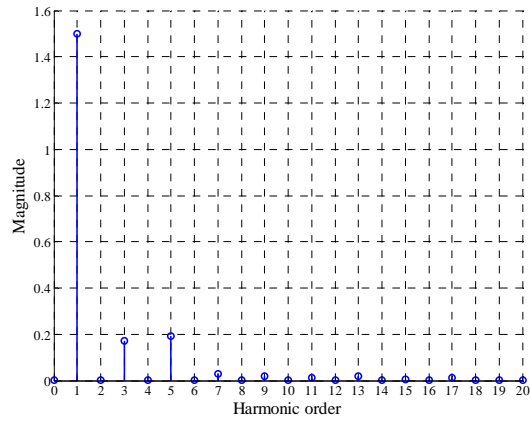
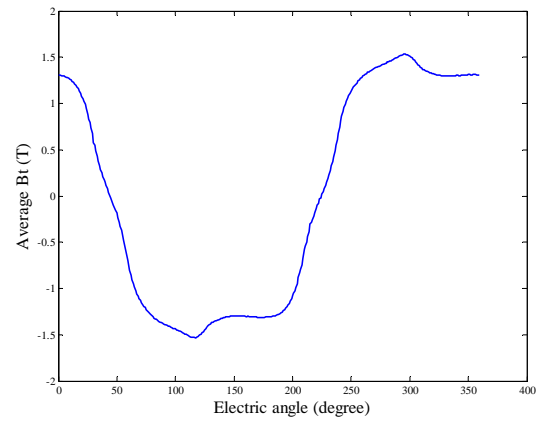
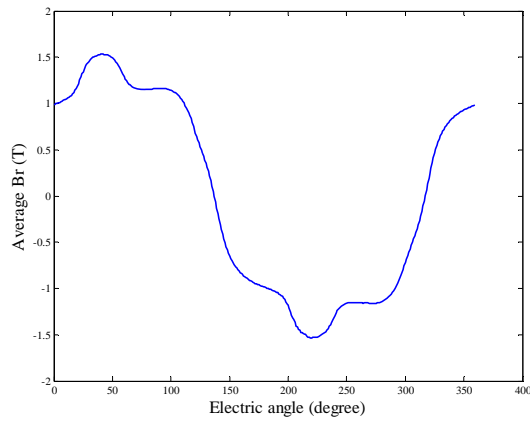
Fig. III.65 Average Bn (top left), magnitude spectrum (top left) and magnitude and phase table (bottom) of left side of steel testing block 8. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	0.828	0.16622
2	0.000	-0.832102
3	0.169	-2.579978
4	0.000	1.207861
5	0.056	1.005435
6	0.000	-2.90313
7	0.015	-0.968671
8	0.000	0.788689
9	0.007	-2.392397
10	0.000	-1.949053
11	0.006	1.495246
12	0.000	2.008726
13	0.004	-1.827826
14	0.000	1.420711
15	0.003	1.315129
16	0.000	-0.179858
17	0.002	-2.10394
18	0.000	2.709616
19	0.003	1.103155
20	0.000	-0.148024

Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	1.650	0.565484
2	0.001	2.57131
3	0.323	-1.721382
4	0.000	2.12733
5	0.078	1.672984
6	0.000	0.670039
7	0.027	-1.340087
8	0.000	-3.102532
9	0.005	1.660464
10	0.000	0.841483
11	0.008	2.708567
12	0.000	2.811507
13	0.008	-0.164176
14	0.000	1.834655
15	0.007	-2.914734
16	0.000	-1.869502
17	0.004	0.648282
18	0.000	1.137216
19	0.003	-2.092589
20	0.000	2.296227

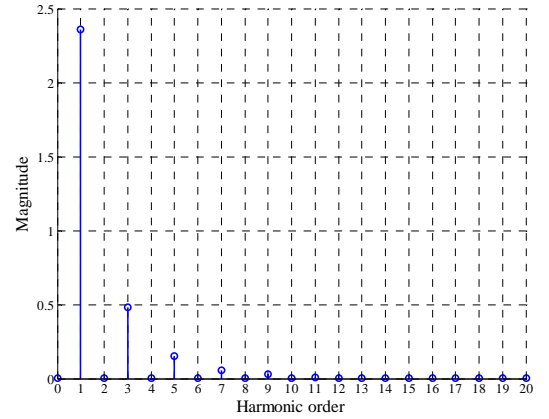
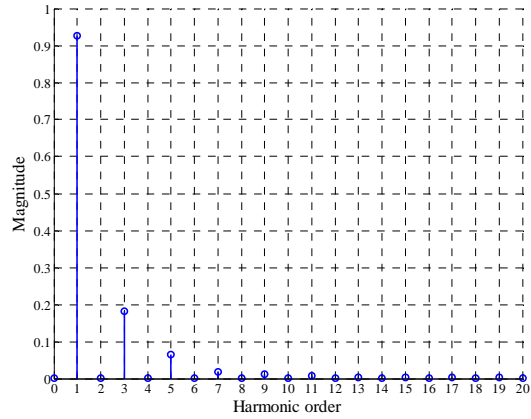
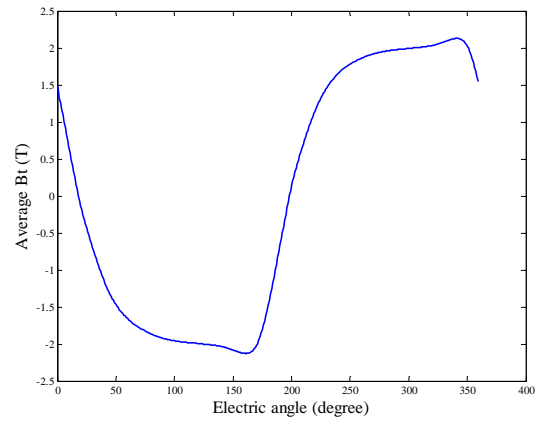
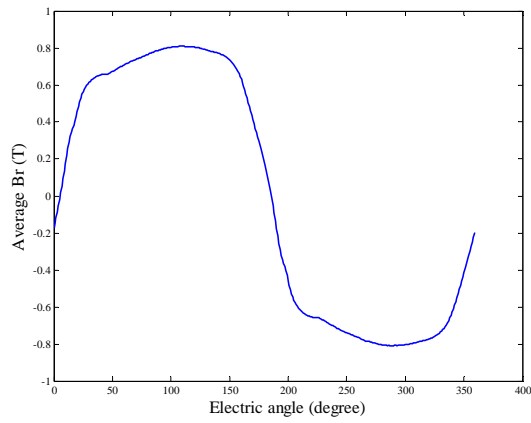
Fig. III.66 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 9. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.000	0
1	1.501	-0.837366
2	0.001	-1.624433
3	0.170	0.701022
4	0.000	-2.695928
5	0.191	2.633268
6	0.000	2.731311
7	0.029	1.997431
8	0.000	0.255325
9	0.018	-0.829556
10	0.000	2.716673
11	0.013	-0.206502
12	0.000	-0.593967
13	0.019	0.603588
14	0.000	0.582373
15	0.006	-2.270008
16	0.000	-0.980055
17	0.011	-2.30423
18	0.001	-3.028995
19	0.002	-0.363341
20	0.000	2.016511

Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	1.655	0.820848
2	0.000	-0.321913
3	0.330	-0.917515
4	0.000	0.785742
5	0.062	-1.611439
6	0.000	-0.956087
7	0.018	0.329175
8	0.000	2.611195
9	0.046	-2.222048
10	0.000	0.68679
11	0.023	2.665121
12	0.000	-1.004501
13	0.026	1.245512
14	0.000	-2.969913
15	0.010	-0.820777
16	0.001	1.795222
17	0.005	-1.631965
18	0.000	-0.161752
19	0.003	1.085274
20	0.000	-1.019008

Fig. III.67 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 10. Phase in radian.



Harmonic order	Amplitude	Phase
0	0.000	3.141593
1	0.926	-1.680898
2	0.000	-0.416499
3	0.182	-1.648074
4	0.000	-0.742305
5	0.066	-1.896546
6	0.000	0.569404
7	0.018	-2.546944
8	0.000	1.757333
9	0.011	2.453782
10	0.000	2.670625
11	0.008	2.113735
12	0.000	-1.18328
13	0.004	2.545733
14	0.000	1.262566
15	0.004	2.748646
16	0.000	-1.255986
17	0.003	2.588855
18	0.000	-0.033295
19	0.002	2.667598
20	0.000	-0.935711

Harmonic order	Amplitude	Phase
0	0.000	0
1	2.361	1.160554
2	0.000	-2.857779
3	0.484	0.697226
4	0.000	2.268058
5	0.153	0.676772
6	0.000	2.99051
7	0.058	0.919501
8	0.000	2.623339
9	0.028	1.019278
10	0.000	2.65564
11	0.009	0.898552
12	0.000	1.461298
13	0.001	0.619955
14	0.000	1.860384
15	0.003	-2.534312
16	0.000	-2.716977
17	0.003	-3.040521
18	0.000	0.625747
19	0.002	-3.049756
20	0.000	0.583403

Fig. III.68 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 11. Phase in radian.

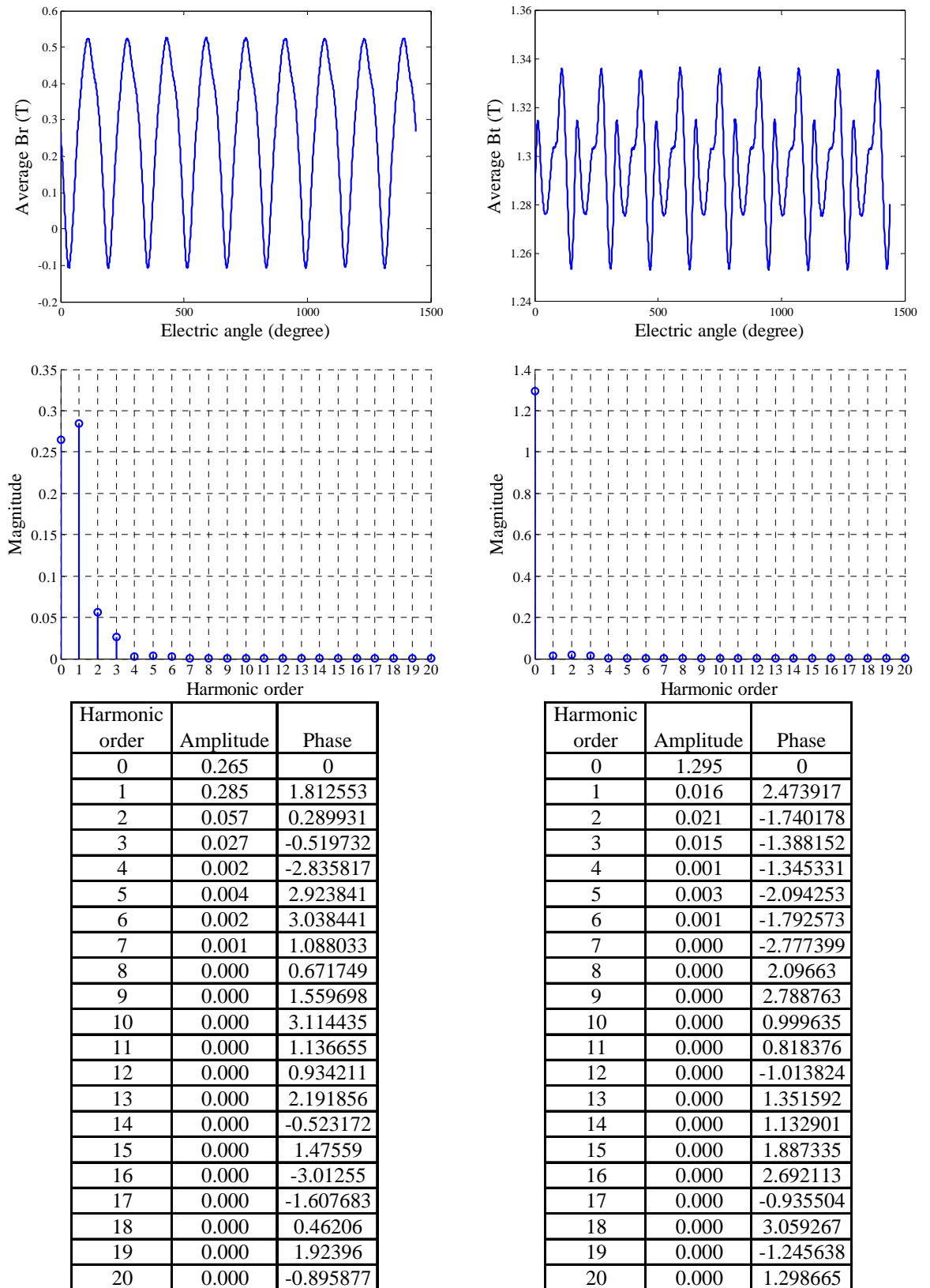


Fig. III.69 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 12. Phase in radian.

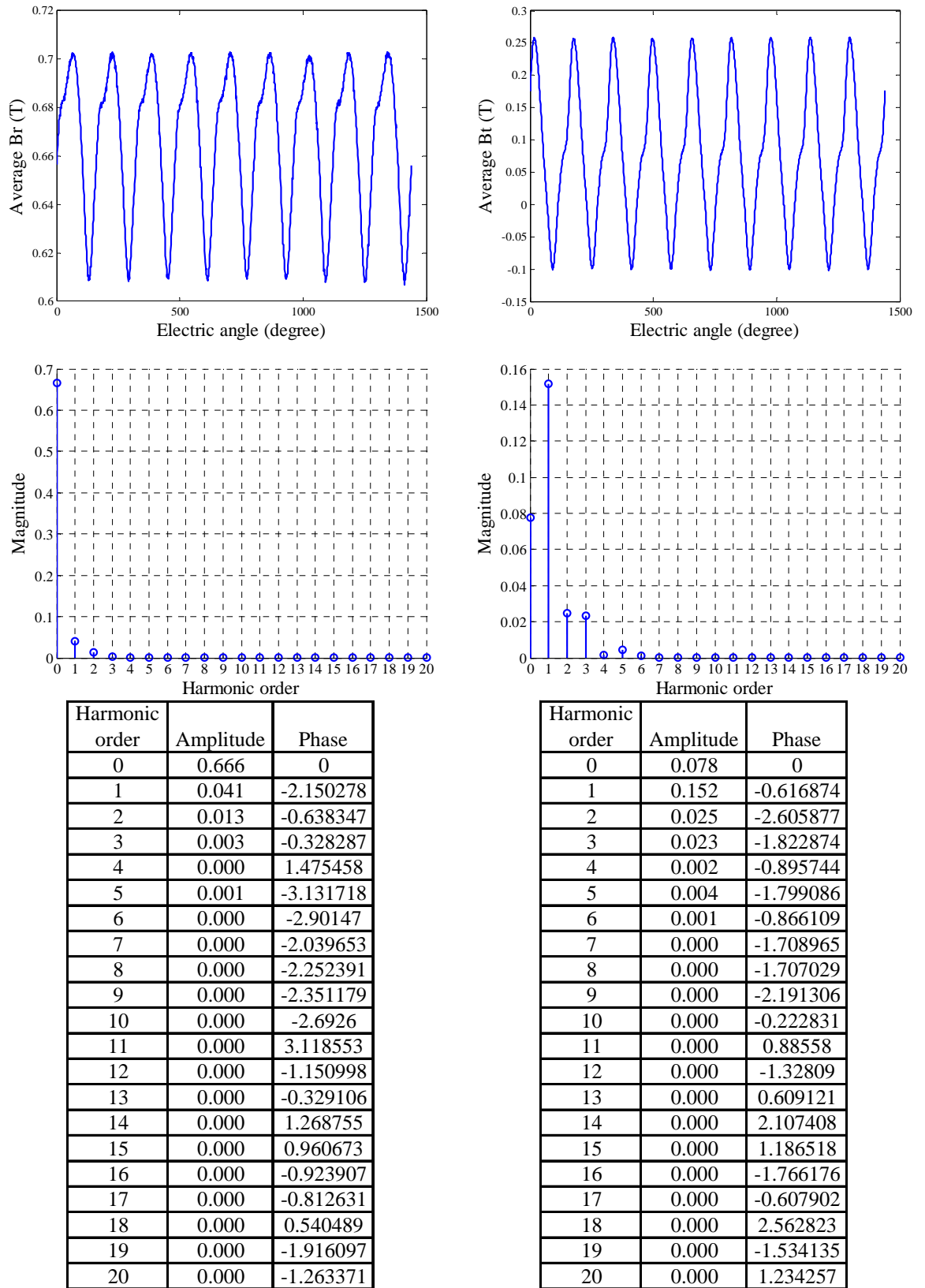


Fig. III.70 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 13. Phase in radian.

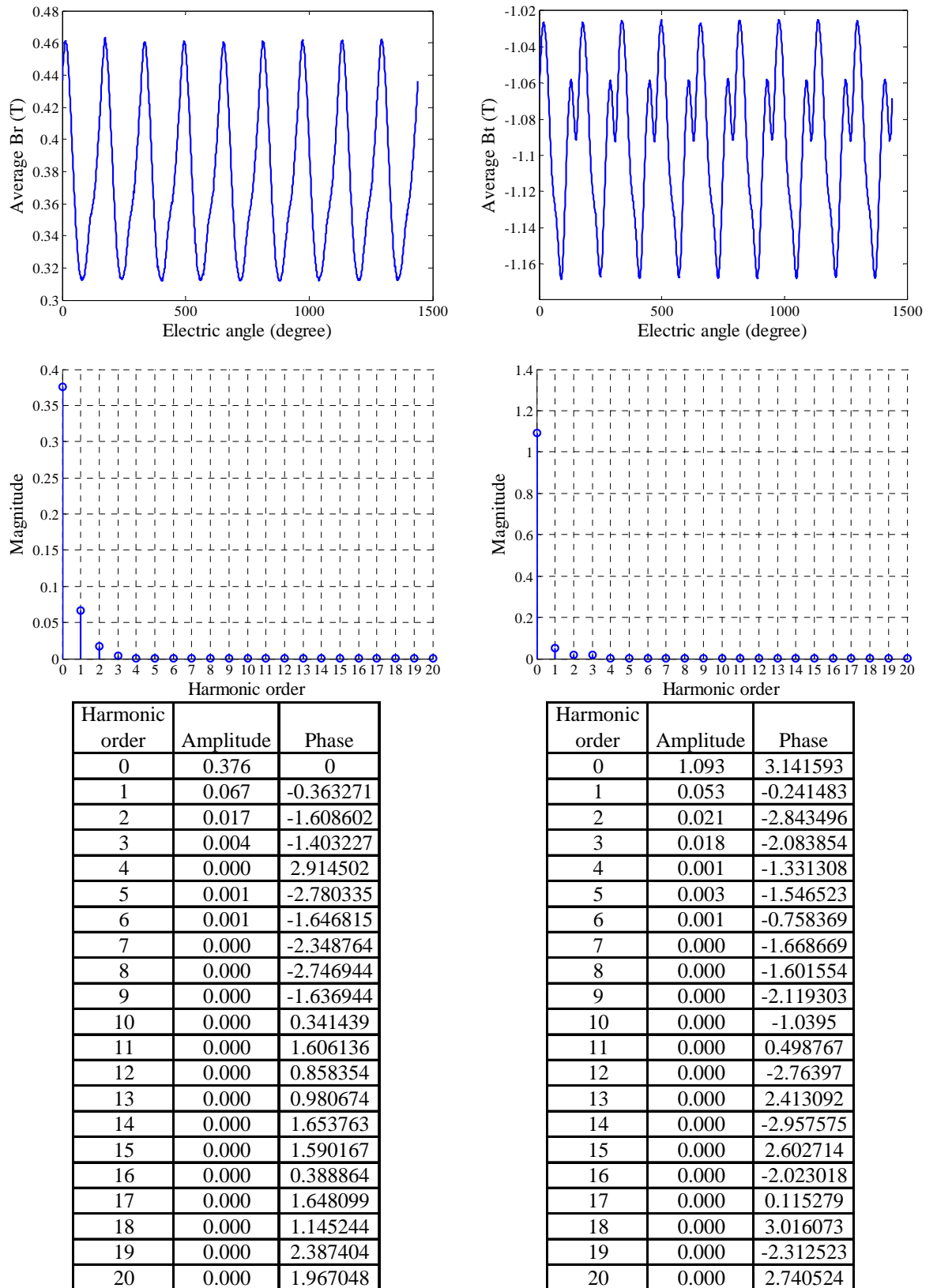


Fig. III.71 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 14. Phase in radian.

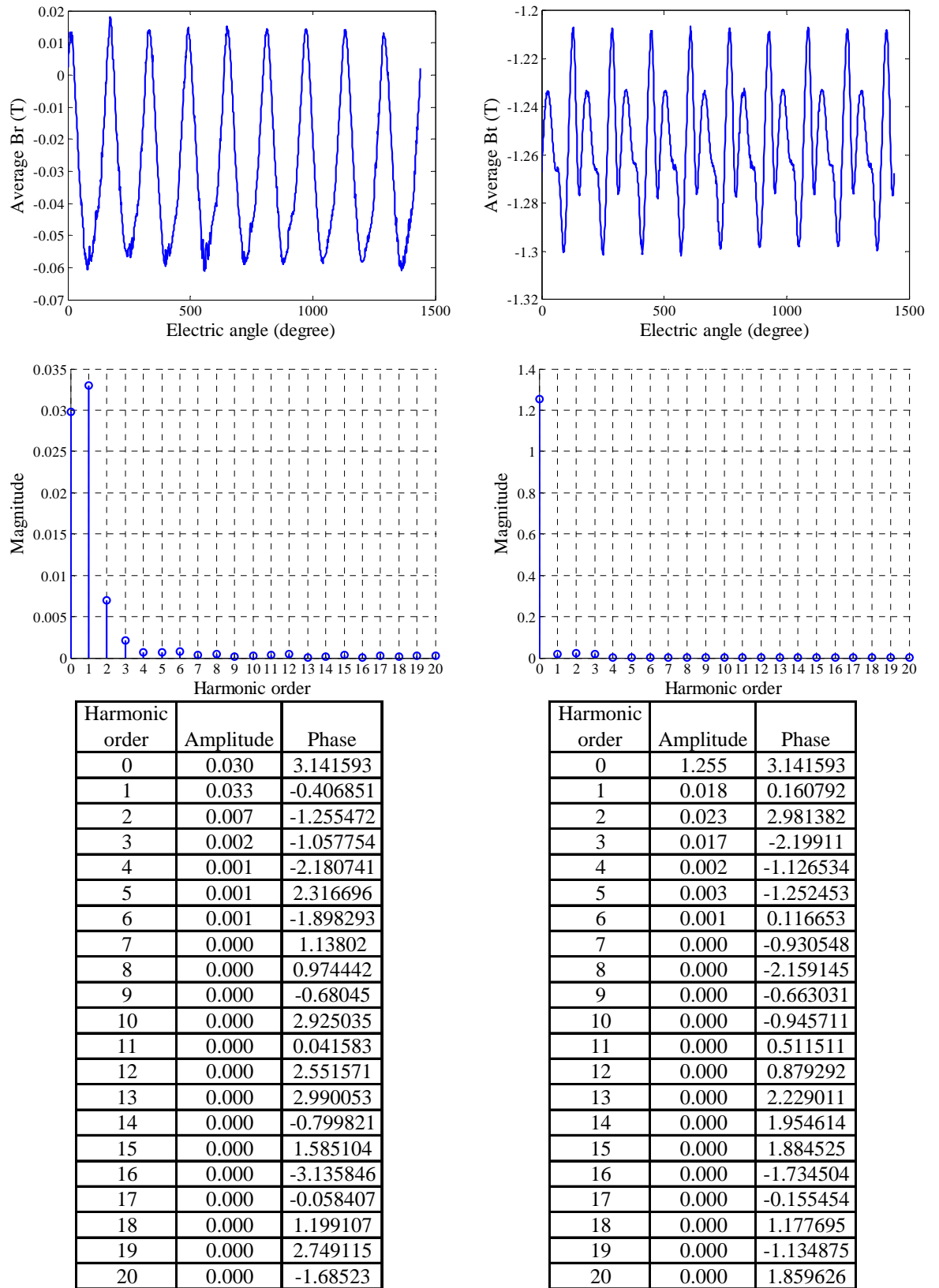


Fig. III.72 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 15. Phase in radian.



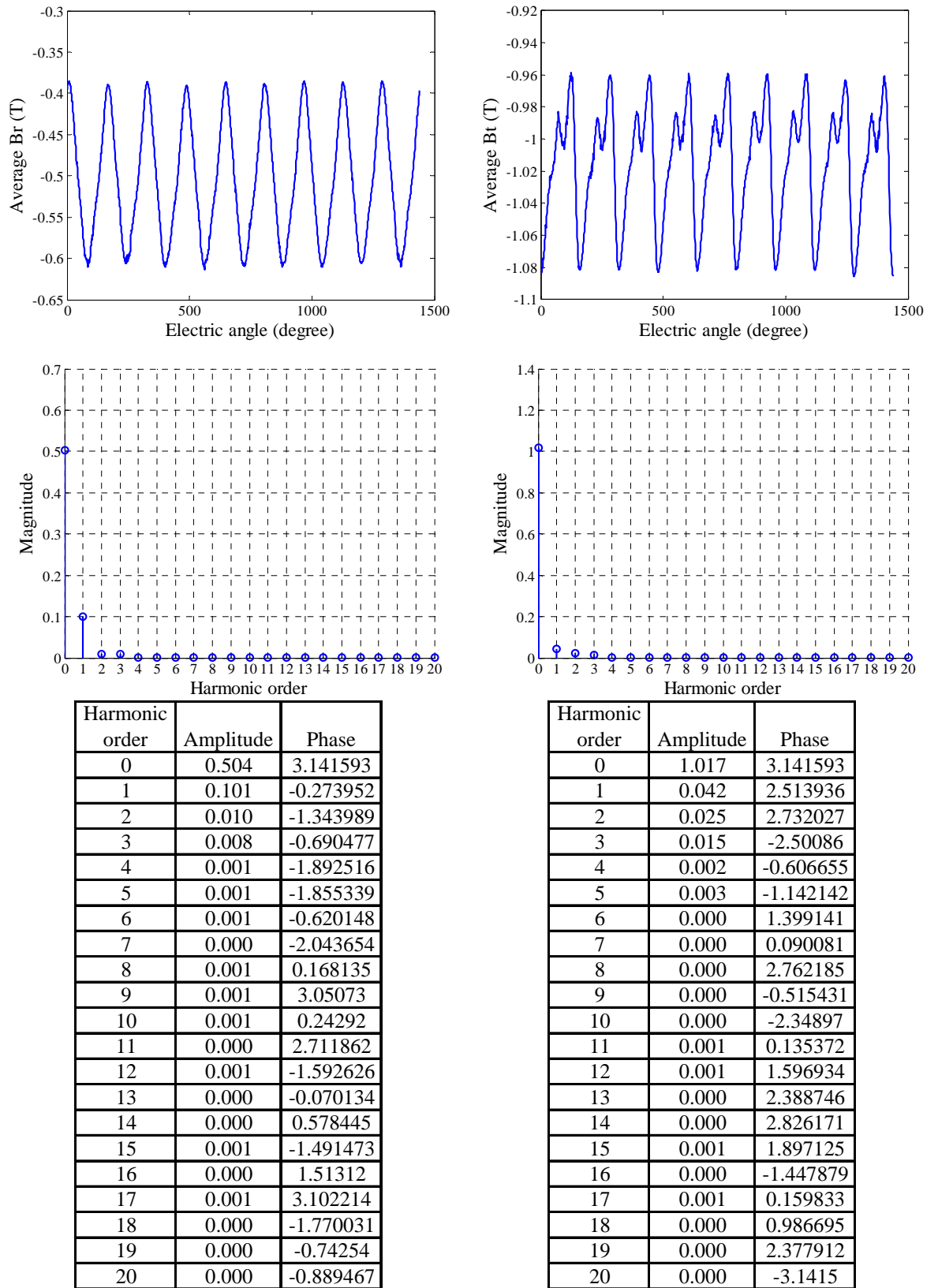
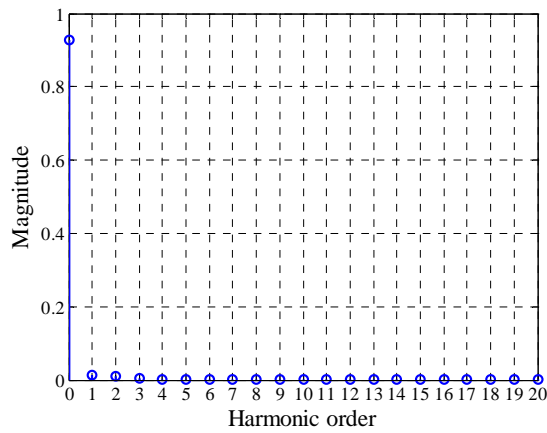
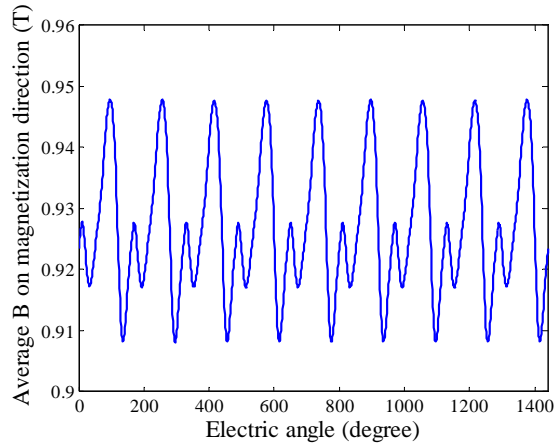


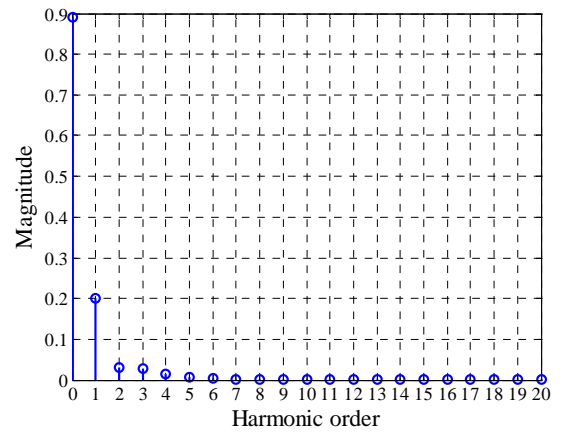
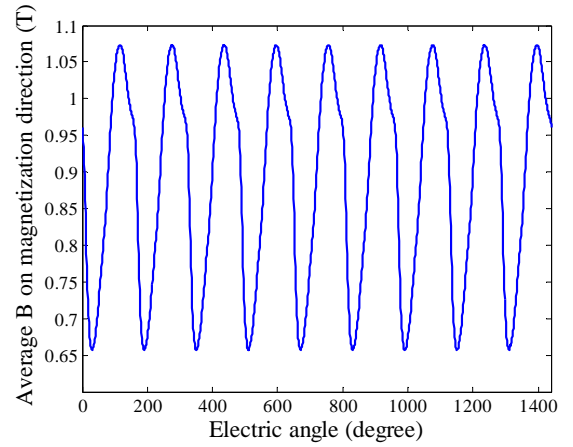
Fig. III.73 Average B in radius direction (top left), Average B in direction of tangential of radius (top right), magnitude spectrums (middle) and magnitude and phase tables (bottom) of left side of steel testing block 16. Phase in radian.

Test block on magnet.



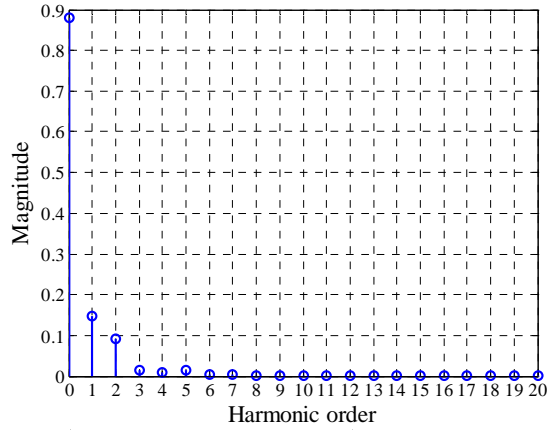
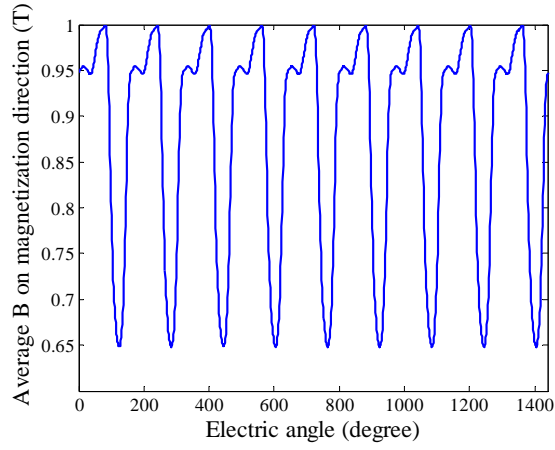
Harmonic order	Magnitude	Phase
0	0.927	0
1	0.012	2.922406
2	0.010	-1.000059
3	0.004	-0.129183
4	0.000	0.261442
5	0.001	3.00429
6	0.000	-2.643101
7	0.000	3.037093
8	0.000	2.440507
9	0.000	2.853182
10	0.000	2.2559
11	0.000	1.847255
12	0.000	-0.774395
13	0.000	-0.017212
14	0.000	0.403951
15	0.000	-0.048787
16	0.000	1.017692
17	0.000	-0.817736
18	0.000	-0.961748
19	0.000	-0.573128
20	0.000	-0.91366

Fig. III.74 Average B in magnetization direction of testing block 1 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



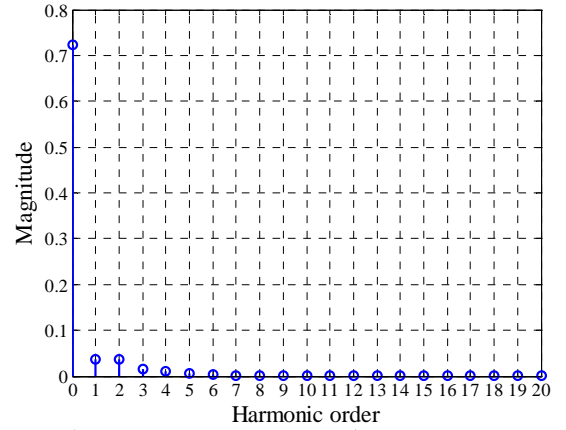
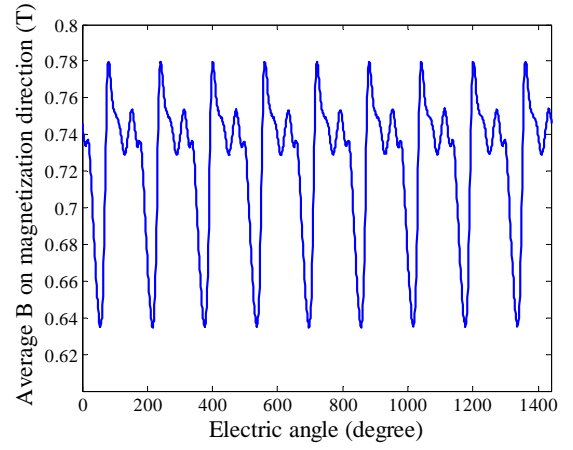
Harmonic order	Magnitude	Phase
0	0.891	0
1	0.200	1.597354
2	0.029	0.295109
3	0.028	-0.00159
4	0.014	-0.510355
5	0.007	-0.465536
6	0.005	-1.142926
7	0.002	-1.511502
8	0.002	-1.49684
9	0.001	-2.073758
10	0.000	-2.611895
11	0.000	-2.499173
12	0.000	2.98712
13	0.000	3.064954
14	0.000	2.205638
15	0.000	0.697653
16	0.000	-2.969038
17	0.000	0.25241
18	0.000	-0.76455
19	0.000	-0.867391
20	0.000	-1.2173

Fig. III.75 Average B in magnetization direction of testing block 2 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



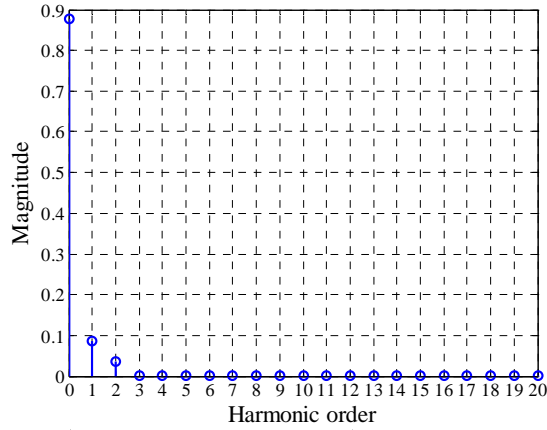
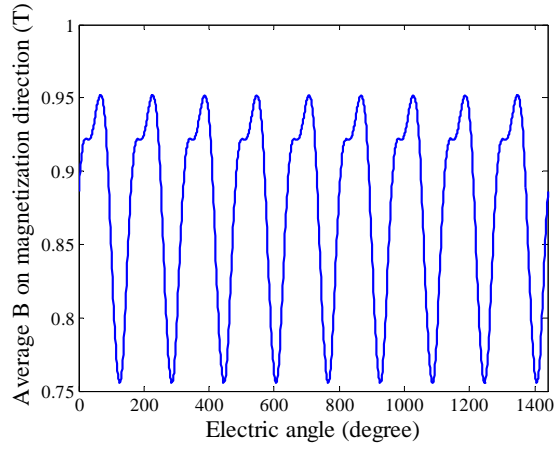
Harmonic order	Magnitude	Phase
0	0.879	0
1	0.146	-1.76859
2	0.090	-0.070182
3	0.016	1.430911
4	0.010	-0.112784
5	0.014	1.533966
6	0.003	-2.970856
7	0.003	1.484025
8	0.003	3.12788
9	0.000	0.987593
10	0.001	3.014921
11	0.000	-1.589013
12	0.000	-3.04182
13	0.000	-1.245037
14	0.000	0.360563
15	0.000	-1.792991
16	0.000	0.16878
17	0.000	-2.258115
18	0.000	-0.972313
19	0.000	-0.351488
20	0.000	1.848415

Fig. III.76 Average B in magnetization direction of testing block 3 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



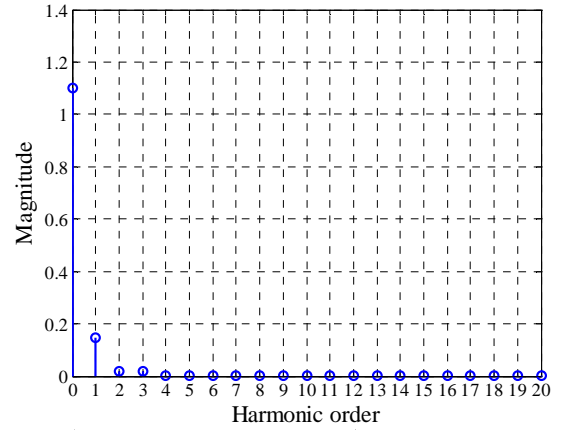
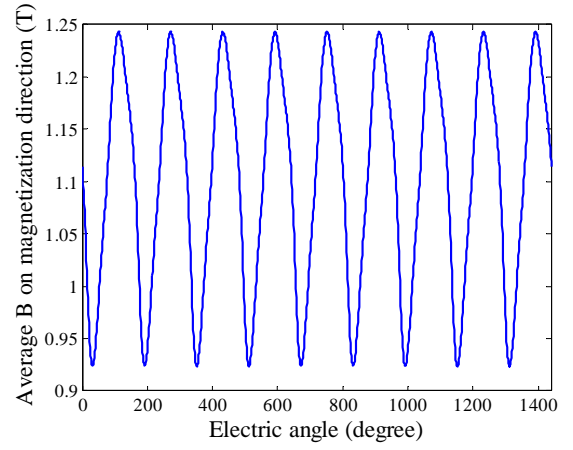
Harmonic order	Magnitude	Phase
0	0.724	0
1	0.038	1.394087
2	0.036	-0.729091
3	0.016	2.918882
4	0.011	0.598374
5	0.007	3.123715
6	0.005	0.883552
7	0.001	-2.115905
8	0.002	0.982267
9	0.001	-1.437325
10	0.000	2.273339
11	0.000	-1.186713
12	0.000	-2.997922
13	0.000	-0.915771
14	0.000	-2.691389
15	0.000	1.43575
16	0.000	1.806692
17	0.000	2.408982
18	0.000	-0.727696
19	0.000	-0.268723
20	0.000	2.265833

Fig. III.77 Average B in magnetization direction of testing block 4 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



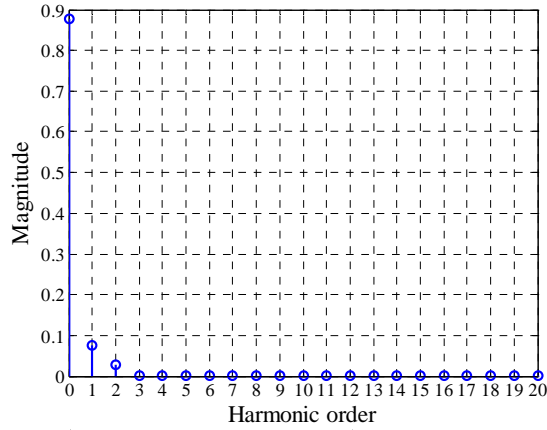
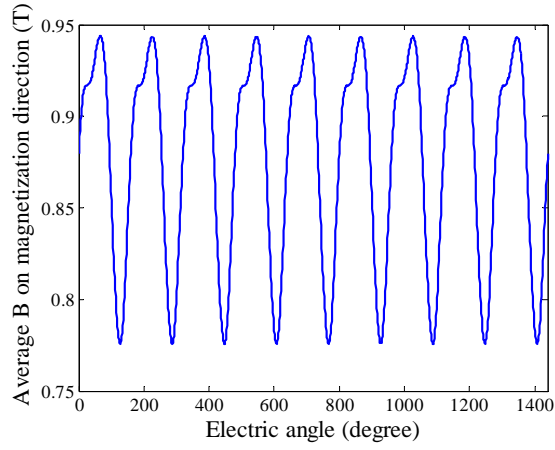
Harmonic order	Magnitude	Phase
0	0.877	0
1	0.087	-1.884622
2	0.035	-0.194106
3	0.002	0.08181
4	0.000	0.275142
5	0.001	2.16388
6	0.000	-2.855803
7	0.000	2.680075
8	0.000	2.331837
9	0.000	3.044413
10	0.000	2.045878
11	0.000	2.265131
12	0.000	-0.241416
13	0.000	-2.188786
14	0.000	-1.574647
15	0.000	0.859834
16	0.000	-2.475722
17	0.000	-1.766758
18	0.000	-0.937424
19	0.000	-0.18162
20	0.000	-2.442247

Fig. III.78 Average B in magnetization direction of testing block 5 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



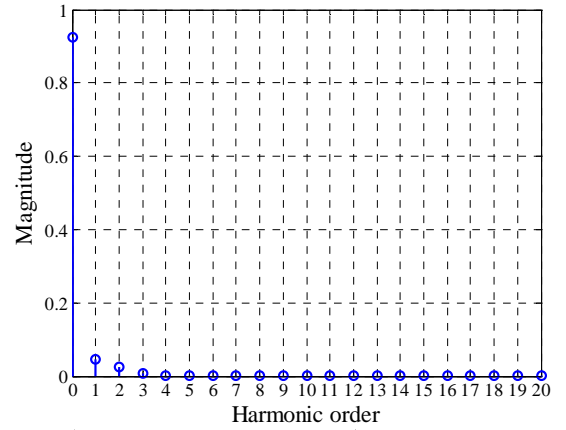
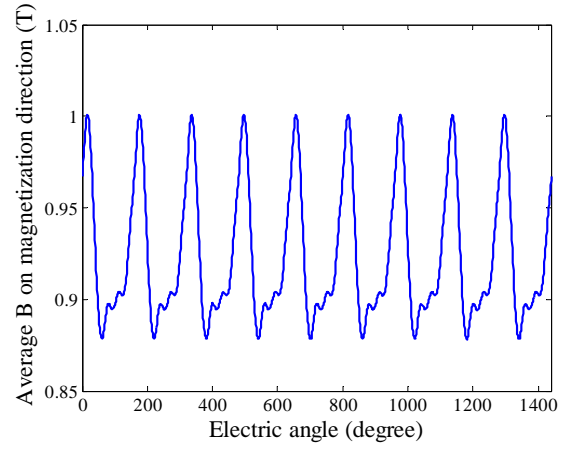
Harmonic order	Magnitude	Phase
0	1.102	0
1	0.147	1.713359
2	0.018	0.340494
3	0.017	-0.158437
4	0.003	-1.292398
5	0.001	-1.333064
6	0.001	-2.383482
7	0.000	2.828212
8	0.000	2.686784
9	0.000	2.579915
10	0.000	2.016203
11	0.000	2.252267
12	0.000	1.550306
13	0.000	1.205698
14	0.000	1.397529
15	0.000	1.853922
16	0.000	1.389696
17	0.000	-0.840243
18	0.000	-2.463105
19	0.000	0.136144
20	0.000	1.71607

Fig. III.79 Average B in magnetization direction of testing block 6 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



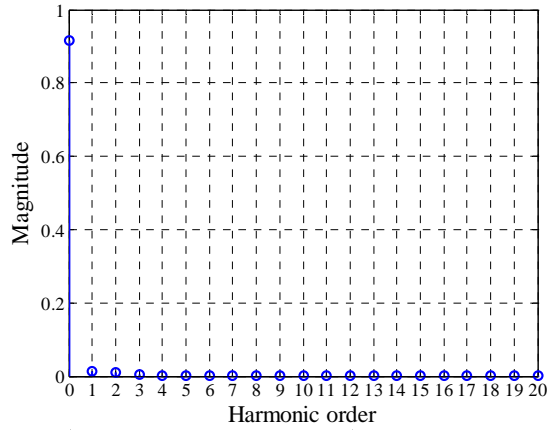
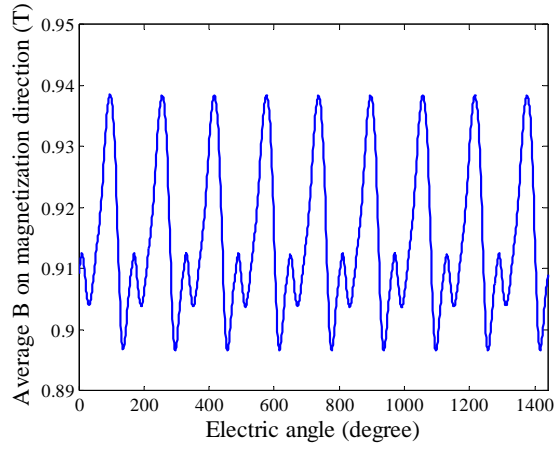
Harmonic order	Magnitude	Phase
0	0.878	0
1	0.075	-1.924737
2	0.027	-0.257987
3	0.002	-0.373251
4	0.000	1.899765
5	0.001	2.523263
6	0.000	-2.759058
7	0.000	2.900289
8	0.000	2.310359
9	0.000	3.006172
10	0.000	2.23377
11	0.000	1.876336
12	0.000	2.187162
13	0.000	-2.293695
14	0.000	-1.664896
15	0.000	-0.529954
16	0.000	-2.004232
17	0.000	-1.105625
18	0.000	-1.18307
19	0.000	-1.23085
20	0.000	0.468245

Fig. III.80 Average B in magnetization direction of testing block 7 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



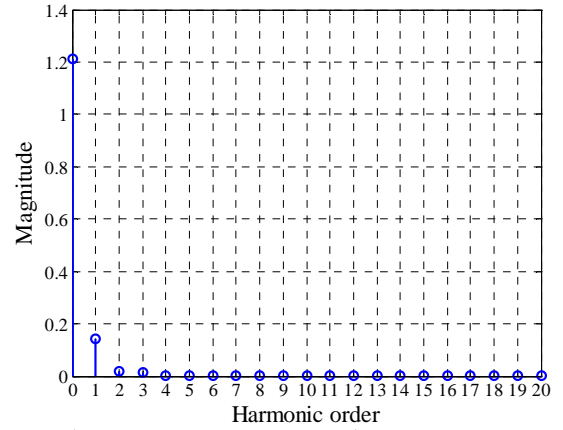
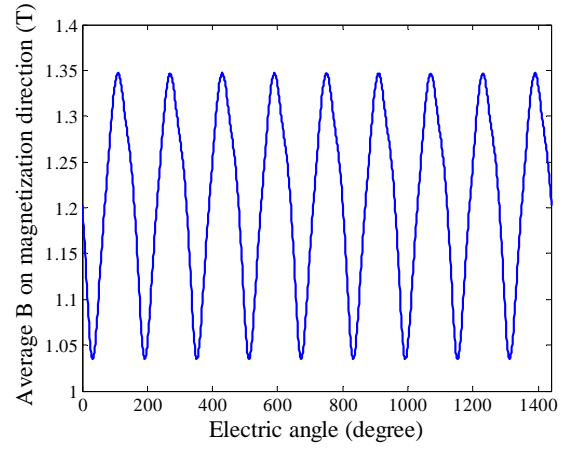
Harmonic order	Magnitude	Phase
0	0.924	0
1	0.046	-0.358774
2	0.025	-1.204538
3	0.007	-2.523829
4	0.002	1.612185
5	0.003	-3.065387
6	0.001	2.40967
7	0.000	3.073177
8	0.000	1.687206
9	0.000	-1.710652
10	0.000	2.837393
11	0.000	1.303313
12	0.000	-1.520448
13	0.000	0.868432
14	0.000	-0.638843
15	0.000	-1.133043
16	0.000	0.525111
17	0.000	-0.787846
18	0.000	-0.701847
19	0.000	-1.332067
20	0.000	1.428244

Fig. III.81 Average B in magnetization direction of testing block 8 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



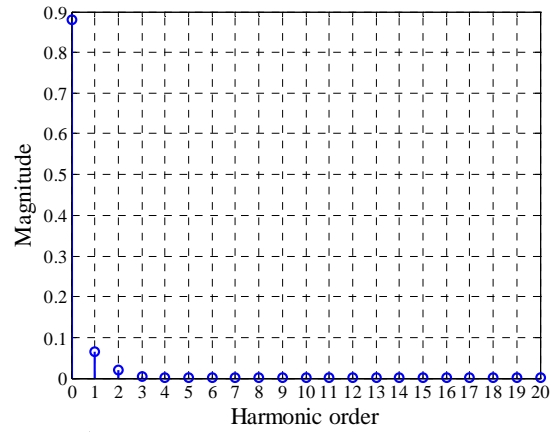
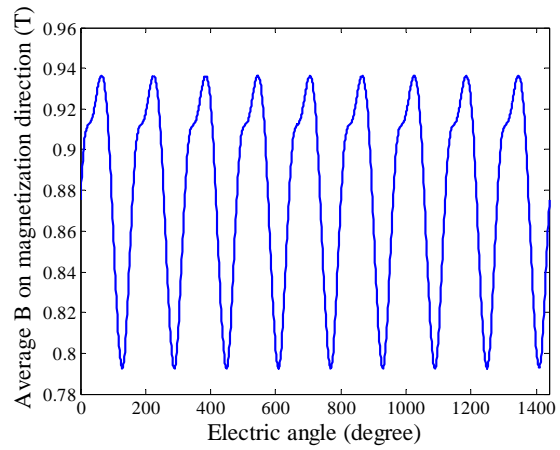
Harmonic order	Magnitude	Phase
0	0.915	0
1	0.014	2.84318
2	0.010	-1.032874
3	0.004	-0.004447
4	0.000	0.234055
5	0.001	2.971949
6	0.000	-2.640656
7	0.000	3.067438
8	0.000	2.473192
9	0.000	2.895928
10	0.000	2.254286
11	0.000	1.890348
12	0.000	-0.41043
13	0.000	-0.072679
14	0.000	0.425143
15	0.000	-0.152713
16	0.000	0.920138
17	0.000	-0.77844
18	0.000	-0.938173
19	0.000	-0.581493
20	0.000	1.659477

Fig. III.82 Average B in magnetization direction of testing block 9 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



Harmonic order	Magnitude	Phase
0	1.210	0
1	0.142	1.833035
2	0.019	0.327715
3	0.014	-0.276723
4	0.001	-1.994242
5	0.001	-3.024699
6	0.001	-2.998035
7	0.000	1.923069
8	0.000	2.424802
9	0.000	2.126313
10	0.000	1.719468
11	0.000	1.906514
12	0.000	1.697549
13	0.000	-0.373473
14	0.000	0.51099
15	0.000	0.847438
16	0.000	2.146606
17	0.000	-0.311694
18	0.000	-2.870913
19	0.000	0.036555
20	0.000	-0.551457

Fig. III.83 Average B in magnetization direction of testing block 10 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.



Harmonic order	Magnitude	Phase
0	0.880	0
1	0.065	-1.972283
2	0.021	-0.344545
3	0.003	-0.539365
4	0.000	2.670953
5	0.001	2.748151
6	0.000	-2.765707
7	0.000	3.049601
8	0.000	2.466957
9	0.000	3.078205
10	0.000	2.29096
11	0.000	1.78348
12	0.000	-0.363917
13	0.000	-0.440846
14	0.000	-0.559391
15	0.000	-0.620884
16	0.000	1.476003
17	0.000	-1.88198
18	0.000	-0.768419
19	0.000	-1.220935
20	0.000	-2.715792

Fig. III.84 Average B in magnetization direction of testing block 11 versus electric angle (top) and its magnitude spectrum (middle) and table. Phase in radian.