MODELLING OF SWITCHED RELUCTANCE MOTORS FOR TORQUE CONTROL

ZHENG QING

NATIONAL UNIVERSITY OF SINGAPORE

2003

MODELLING OF SWITCHED RELUCTANCE MOTORS FOR TORQUE CONTROL

ZHENG QING
(B. Eng. NCUT, CHINA)

A THESIS SUBMITTED
FOR THE DEGREE OF MASTER OF ENGINEERING
DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE
2003

Acknowledgments

I would like to express my most sincere gratitude to my main supervisor, Dr. Xu Jian-Xin, for his consistent guidance, patience and support throughout my M.Eng research. Dr. Xu's rigorous scientific approach and endless enthusiasm in research have influenced me greatly. This will definitely benefit me greatly in my future work. His erudite knowledge as well as deep insights in the fields of computational intelligence and control make this research a rewarding experience. Without his stimulating discussions and kindest help, this thesis and many other results would have been impossible.

I am greatly indebted to my co-supervisor, Dr. Sanjib Kumar Panda, who guided and encouraged me throughout my research work. Dr. Panda has proposed countless very helpful suggestions through my research work as well as during this thesis writing procedure. I have learned quite a lot from his comments, which are always inspiring and fruitful. I shall never forget his sacrifice for spending hours and hours with me for research discussions. I am extremely grateful and obliged to Dr. Panda for spending his personal time for the correction as well as revision of this thesis.

I deeply appreciate the National University of Singapore for giving me this

opportunity to pursue my M.Eng degree with research scholarship. I would also like to take this opportunity to thank Yan Rui, Sanjib Kumar Sahoo, Chen Jianping, Zhang Hengwei, Heng Chun Meng, Wu Chao, Yu Qi, Tang Huajin, Chng Chung Wei and all my friends in the Control and Simulation Lab for their interesting and helpful discussions and generous help.

My deepest gratitude is due to my family members. Without their love, patience, encouragement and sacrifice, I would not have accomplished this. Special thanks go to my husband, Wang Fangjing, for his warmest love and support during the long process of study. I wish to dedicate this thesis to all of them.

Contents

List of Figures

List of Tables

Nomenclature

t	time (sec)
q	number of phases in the motor
v	phase voltage
i	current
ψ	flux-linkage
R	phase winding resistance
T	motor total torque
T_L	load toque
T_j	motor torque of phase j
T_d	desired torque
θ	rotor angle (mechanical)
θ_e	rotor angle (electrical)
ω_m	speed of motor
N_r	number of rotor poles
N_s	number of stator poles
$\beta_r,\ \alpha_r$	rotor pole arc and pitch

β_s	stator pole arc
ϕ	step angle
θ_{on}	ON angle
θ_{off}	OFF angle
L_j	inductance of phase j
L_a	phase inductance at aligned position
L_u	phase inductance at unaligned position

Acronyms

SRM	Switched Reluctance Motor
SR	Switched Reluctance
SRD	Switched Reluctance Drive
VRM	Variable Reluctance Motor
PM	Permanent Magnet
VR	Variable Reluctance
DC	Direct Current
AC	Alternating Current
BLDC	Brushless DC
MMF	Magneto-motive Force
EMF	Electro-motive Force
VSC	Variable Structure Control
SMC	Sliding Mode Control
SM	Sliding Mode
PWM	Pulse-Width-Modulation
FFT	Fast Fourier Transform

TSF	Torque Sharing Function
LM	Levenberg-Marquardt
GA	Genetic Algorithm
CMAC	Cerebellar Model Articulation Controller
LMS	Least Mean Square
ANN	Artificial Neural Network
NN	Neural Network
MLPs	Multilayer Perceptrons
RBF	Radial Basis Function
DSP	Digital Signal Processor
RMS	Root-Mean-Square
VA	Volt-Ampere
BP	Backpropagation
PE	Propagation Error
GBDM	Backpropagation Gradient-Based Descent with Momentum
SSE	Sum Square Error

Summary

The switched reluctance motor (SRM) has salient poles both on stator and rotor. The stator consists of concentric windings while there are no windings or permanent magnets on the rotor. It has a lot of advantages due to its low cost, simple rugged construction, and relatively high torque-to-mass ratio. Contrary to the conventional motors (direct current and alternating current motors), the SRM is intended to operate in deep magnetic saturation to increase the output power density. Due to the saturation effect and variation of magnetic reluctance, the electromagnetic torque is a highly nonlinear function of rotor position and phase current. As a result, the conventional rectangular pulse excitation method of controlling this motor results in significantly high torque ripple. In the past couple of decades, several research works have been carried out addressing the problem of torque ripple reduction. However, complete elimination of torque ripple is still an existing problem.

This thesis aims at investigations on application of intelligent tools for SRM modelling and high performance torque control. An accurate nonlinear dynamic model is the basis for further investigations of the SRM. In this thesis, two classes of

modelling approaches, i.e., analytical model based approaches and blackbox based approaches are applied for the modelling of the SRM. For the analytical model based approaches, a Levenberg-Marquardt(LM) gradient expansion method and a genetic algorithm(GA) are used for flux-linkage modelling and torque modelling. For the model-free blackbox based approaches, artificial neural network(ANN) techniques with different algorithms are employed for flux-linkage modelling, torque modelling as well as inverse torque modelling. Simulation and experimental results verify the effectiveness of the derived models for achieving high accuracy and their respective advantages.

The purpose of obtaining an accurate SRM model is to minimize or eliminate the torque ripple for high performance torque control. In this thesis, we propose a new torque sharing function(TSF) and formulate it as an optimal design problem. Subsequently, we formulate the problem with distinct phases into a multiobjective optimal design problem and propose a new fitness function. GA is employed to determine the desired current waveforms of the SRM for torque ripple minimization through generating appropriate reference phase torques for a given desired torque. Simulation results show that the design parameters can be automatically selected by GA and much smoother current waveforms are generated when comparing with conventional TSF design using heuristic knowledge and therefore verify the effectiveness of the proposed TSF and fitness function.

Chapter 1

Introduction

1.1 Introduction

An electrical machine is the basic element of any electrical drive whose operation is based on electromagnetic forces. With recent advances in factory automation, variable-speed drives are now indispensable in modern industries because they can be used either to conserve energy or to meet exacting load requirements. The use of adjustable-speed drives is ever increasing and will maintain its momentum for several decades to come [1].

The traditional classification of motors is based on the type of supply and the construction of the magnetic system and electrical windings. Direct current (DC) motors, introduced more than a century ago, still constitute a major component of the variable speed drive industry because of their simple control requirements, i.e., the armature magneto-motive force (MMF) and the field MMF are decoupled and, as a consequence, it is an easy task to control torque by controlling the armature

current (since torque is directly proportional to armature current) and flux by controlling the field current [1]. However, the presence of carbon brushes and commutator segments of DC motor requiring regular maintenance is a potential disadvantage for a rugged and maintenance free drive system.

The alternating current (AC) motors, i.e, induction motors, synchronous motors etc., require relatively less maintenance and are more rugged than DC motors making them highly attractive in the electric drive industry. During the initial stage of development of AC motors, they were used for constant speed drives because of lack of variable frequency AC supply during that period. With the advent of high speed power electronic switching devices and sophisticated power electronics technology, AC motors could be used for variable speed drives. However, the performance is not very much satisfactory with the conventional control schemes because of the highly coupled multivariable structure of the AC motors. With the application of modern control algorithms to handle such complex systems and continuous improvement in power electronics knowhow, it is now possible to control AC motors efficiently. One of those modern control techniques is the field-oriented control or vector control [2, 3] which is highly popular and elegant. This control principle transforms the mathematical model of the AC motor to a simple structure like that of a DC motor, thereby, allowing the whole set of control tools for the DC motors to apply for the AC motors. Overall, the AC motors have now become an integral component of the industry.

It is also worth to mention about one important category of motors, i.e,

permanent magnet brushless synchronous motor or brushless DC (BLDC) motor. This motor has a linear torque-speed characteristics like the traditional DC motor without brushes and commutator(hence the name, BLDC). It is widely used in data storage industry (hard disk drive and CD-ROM drive) and robot manipulators. The BLDC motor has a smooth air gap and generally has low torque ripples. However, because of its high cost, it is not so widely used in drive industry. In addition, there is also another disadvantage with this motor, i.e., it has low starting torque [4]. Therefore, the BLDC motor is generally used for low power applications and, for high power applications, the induction motors are preferred. Hence, a motor which can be used for both low as well as high power applications is always beneficial.

In the past couple of decades, the switched reluctance motor (SRM) and switched reluctance drives (SRD) have been intensely developed. A reluctance machine is one in which torque is produced by the tendency of its movable part to move to a position where the inductance of the excited winding is maximized [5]. This definition covers both *switched* and *synchronous* reluctance machines. The switched reluctance motor has salient poles on both the rotor and the stator and operates like a variable-reluctance stepper motor except that the phase current is switched on and off when the rotor is at precise positions, which may vary with speed and torque. It is this switching which gives the switched reluctance motor its name. This type of motor cannot work without its electronic drive or controller. This motor can be used for both low and high power applications. Although the basic principle of operation of this motor was known almost 150 years ago, it has

emerged as an alternative to DC and AC motors only in the past few decades because of the availability of high power and high speed semiconductor switches [6, 7, 8] and availability of microcontrollers for complex control problems. The SRM is topologically and electromagnetically identical to the variable reluctance (VR) stepper motor although there are differences in engineering design and control methods. Since the principle of operation of SRM is based on the variation of inductance (or reluctance), it is also often called as variable reluctance motor (VRM). At low power levels, the SRM can be considered as an alternative to costly BLDC motor [9]. At high power levels, the SR motor is a viable substitute to induction motor and DC motors [10, 11]. The properties distinguishing SRM from other types of motors are as follows:

- It has salient poles both in the stator and rotor;
- The number of rotor poles is not equal to the number of stator poles;
- There are no magnets or windings on the rotor;
- The magnetic flux/current is not sinusoidal;
- The phase windings have a very low mutual inductance;
- The generated torque does not depend upon the sign/polarity of the stator current. Hence, unipolar power converters are generally used.

All the above features make traditional vector diagrams and equivalent schemes for other motors not very much suitable for SRM analysis. Because of its highly

nonlinear characteristics, it needs sophisticated control to obtain a high level of performance. Because of its very simple structure, its manufacturing cost is low and it has very low maintenance requirement. In addition, the requirement of a unipolar power converter makes the control system cost effective.

The rest of this chapter is organized as follows. The construction and operating principles of SRM are introduced in Section 1.2. Section 1.3 discusses the SRM's advantages, disadvantages, and its industrial applications. Section 1.4 gives a description of various power converters and control strategies used in practice. Then, in Section 1.5, a study of recently modelling techniques proposed major control algorithms has been included highlighting the major contributions. Finally, the research work carried out in this thesis is reported.

1.2 Basic Principles of SRM

The cross-section of a typical four-phase (8/6) SRM is shown in Fig. 1.1, where the coil of only one phase along with its power converter has been drawn for the sake of lucidity. The SR motor is salient both on stator and rotor, singly excited (stator only) and brushless. There are no magnets or windings on the rotor which is built from a stack of laminated steel. The stator houses concentric windings on its poles and diametrically opposite stator pole windings are connected in series to form one phase (as shown in Fig. 1.1).

Conditionally, electromechanical energy conversion may be considered as con-

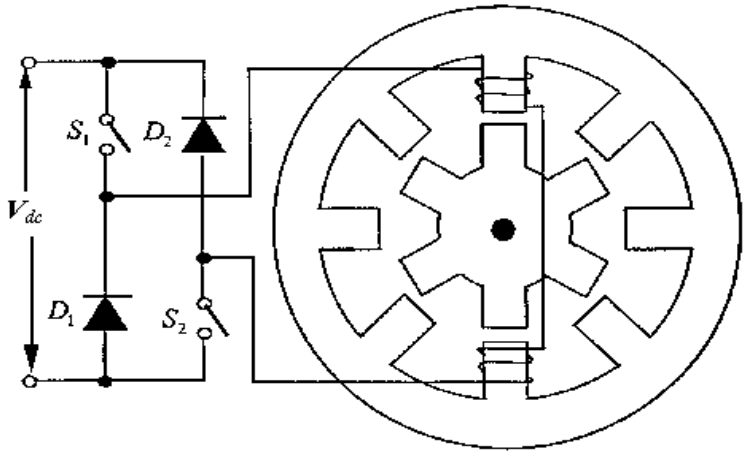

Figure 1.1: Cross-sectional view of a 4-phase SR motor with power converter showing only one phase winding.

secutive transformation of electrical energy into magnetic energy and of magnetic energy into mechanical energy. The transformation of electrical energy into magnetic energy is described by the differential equation

$$\frac{d\psi_j}{dt} = v_j - Ri_j \ , \ j = 1, \cdots, 4. \tag{1.1}$$

The coenergy (W_j) of the magnetic field is obtained as an integral at constant rotor position θ, where magnetic energy does not convert into mechanical energy:

$$W_j(\theta, i_j) = \int_0^{i_j} \psi_j(\theta, i_j) di_j. \tag{1.2}$$

Transformation of magnetic coenergy into mechanical energy without exchange between the winding and the electrical supply at constant current generates torque and is determined by differentiating the coenergy function W_j with respect to θ at constant current, i.e.,

$$T_j(\theta, i_j) = \left.\frac{\partial W_j}{\partial \theta}\right|_{i_j = constant} \ , \ j = 1, \cdots, 4. \tag{1.3}$$

The generation of torque can also be explained by the tendency of the magnetic circuit of the motor to adopt a configuration of minimum reluctance, i.e., rotor poles move into alignment with the energised stator poles so that the inductance of the excited phase is maximised. Hence, this motor is named so. Sustained rotating motion is achieved by sequentially energizing the various phases according to the rotor position as the rotor turns. If the energised phase is not driven into saturation, the flux-linkage can be expressed as:

$$\psi_j(\theta, i_j) = L_j(\theta) i_j(\theta) \tag{1.4}$$

which is the product of phase current and the phase inductance. Thus, by using the above equations, the expression for torque can be derived and is given by

$$T_j(\theta, i_j) = \frac{1}{2} i_j^2 \frac{dL_j}{d\theta}. \tag{1.5}$$

From equation (1.5), it is immediately understood that the generated torque is independent of the direction of current flow. Hence, unidirectional currents are generally used, thereby, greatly simplifying the design of the power converter. A better understanding of the relationship between the phase inductance profile and the torque profile can be obtained from Fig 1.2 where the top figure shows the inductance variations of one typical phase and the corresponding torque profile (according to equation (1.5)) at constant current is shown in the bottom figure. The descriptions of the various angular positions is given below [6].

- At θ_0, the leading edges of the rotor poles meet the edges of the stator poles and the inductance starts a linear increase with rotation, continuing until the

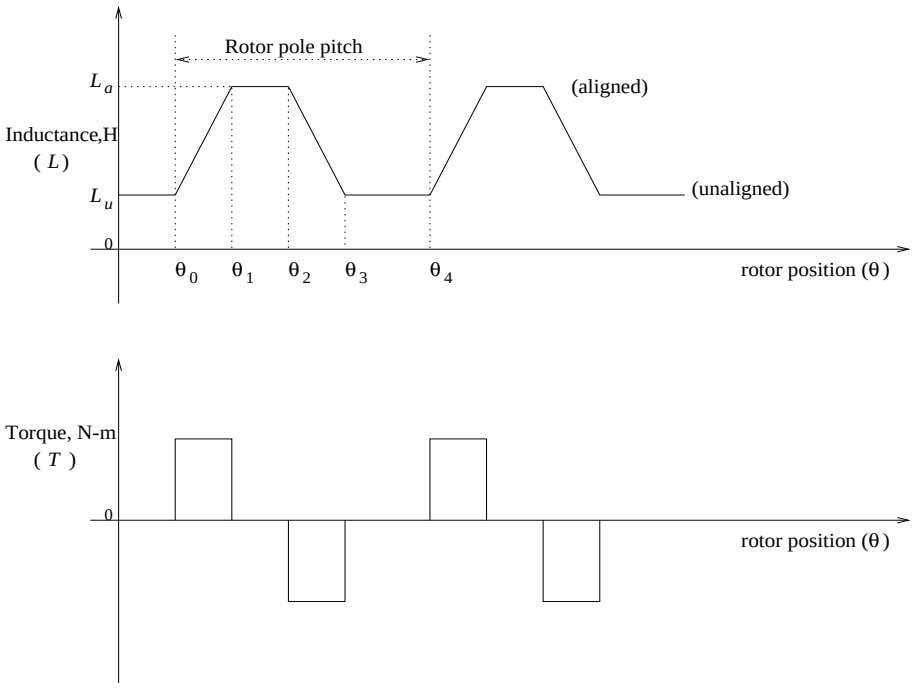

Figure 1.2: Top figure: Ideal variation of inductance of one stator phase as a function of rotor position. Bottom figure: Corresponding variation of torque with constant current as a function of rotor position.

poles are fully overlapped at θ_1, when the inductance reaches its maximum value L_a. Ideally, $\theta_1 - \theta_0 = \beta_s$ (stator pole arc).

- From θ_1 to θ_2, the inductance remains constant at L_a through the region of complete overlap. Ideally, $\theta_2 - \theta_0 = \beta_r$ (rotor pole arc).

- From θ_2 to θ_3, the inductance decreases linearly to the minimum value L_u. Hence, $\theta_3 - \theta_0 = \beta_s + \beta_r$.

- From θ_3 to θ_4, the stator and rotor poles are not overlapped and the inductance remains constant at L_u. So, the rotor pole pitch $\alpha_r = \theta_4 - \theta_0$.

From Fig. 1.2, the following conclusions can be easily derived.

(i) If $\dfrac{dL}{d\theta} > 0$, then $T > 0$. It is a case of motoring torque production.

(ii) If $\frac{dL}{d\theta} < 0$, then $T < 0$. It is a case of generating torque production.

It is clear that for the motor to be able to start in one direction (forward) from any initial rotor position, the pole arcs must be chosen such that at least one of the phase windings is in a region of increasing inductance (or decreasing inductance for rotation in the reverse direction). Further, in order to produce a unidirectional forward motoring torque, the stator phase windings have to be sequentially energised such that the current pulse for a phase must coincide with the angular interval where the $\frac{dL}{d\theta}$ for that phase is positive. Similarly, a braking torque (generating mode) or reverse rotation (motoring in the reverse direction) is obtained when the current pulses coincide with the angular intervals where the respective $\frac{dL}{d\theta}$ is negative. Therefore, a rotor position sensor is obviously required to start and stop the conduction various phases. When this motor is operated without feedback control and the phase currents are switched on and off in sequence, the rotor will advance in steps of angle (called as step angle) ϕ, given by

$$\phi = \frac{360^\circ}{qN_r} \tag{1.6}$$

where q is the number of phases in the SRM and N_r is the number of rotor poles. For a four-phase SR motor with six rotor poles, $\phi = 15^\circ$.

1.3 Features and Industrial Applications of the SRM

Simple construction is a prime feature of the SRM. SR motors eliminate rotor windings, permanent magnets (PMs), brushes, and commutators. With no rotor

windings, the rotor is basically a piece of steel (and laminations) shaped to form salient poles. The absence of brushes provides long life. The absence of permanent magnets and rotor winding reduces the cost. Besides, SR motors have some other attractive advantages that make it favorable for various industrial applications. In summary, the main advantages of the SRM include [12]:

- Low cost
- Long life
- High power-to-weight ratio
- High efficiency over a wide speed range
- High-speed and acceleration capability
- High fault tolerance

With all these advantages, as well as available and affordable power modules, DSPs, ASICs, the potential applications for SR motors are numerous. From fractional HP to very high powered motors where variable speed and torque are required, SR motors can address unique and varied requirements. For example, the favorable speed/torque relationships and high fault tolerance make them ideal for many electronic mopeds, utility vehicles, golf carts, electric cars, buses and trains [13]. In addition, there are many variable speed fan and blower applications where the energy efficiency of SRM offers cost savings over the long term [14]. Furthermore, the motor's very high speed capability and robust feature make the motor well suited in the aerospace field [15].

However, SR motors are not without their drawbacks. These disadvantages are listed below:

a) Need for position sensing

As mentioned above, for smooth SR motor rotation, current switching between phases must occur at precise angular points. Traditionally, the rotor position is measured with some mechanical angle transducers mounted on the rotor shaft. However, these sensors not only increase cost and size to the motor, but also are a major source of unreliability. Therefore, sensorless control is required to make the motor truly practical in many applications. Sensorless operation involves measuring motor characteristics at the drive and then using sophisticated calculations to derive the required position data.

b) High torque ripple

Another limitation of the SRM is torque ripple, especially when it is operated in single-pulse voltage control mode. For many applications where the machine is operating at fairly high speeds, this is not a problem since the mechanical time-constant is far larger than the fast rates of change of instantaneous torque produced by the motor. However, minimizing torque ripple becomes particularly important when SR motors are used in servo positioning. Increasing the number of commutation phases can reduce torque ripple but this will increase number of components for the commutation electronics, and consequently the cost. There are several other ways to control torque ripple, such as producing stator and rotor laminations in a twin-stack arrangement,

adding electronics to compensate or advanced control techniques [16, 17, 18], but these could add complexity, cost, and real-time computation requirement.

c) High acoustic noise and vibration

Torque and vibration become interrelated effects as the rotating pole of the SRM pieces "switch" in and out of flux contact with the stator poles. Forces developed during alignment of the salient poles are actually high enough to distort the stator structure. The motor then resonates due to harmonics of these forces to produce audible noise.

The vibration and noise can be reduced by adding components to the electronics, designing special magnetic circuits, and adjusting the mechanical design, but taking some or all of these steps could compromise the motor's benefits. Designers generally select the right combination of noise reduction and performance to suit the particular application.

Since the SRM has the above limitations, to make it more competitive in the variable speed drives market, various methods are in place to reduce these effects, with new methods under development. Both the motor and controller offer ways to make improvements, but treating them as a whole system brings the best results.

1.4 Operation and Control of the SRM

A typical SRM drive system comprises a DC power supply, a power converter, current/voltage sensors, the controller, SRM, position sensor and the load. The

diagram of the main components is shown in Fig. 1.3. The controller receives the external command signal such as reference speed or torque, and calculates the switching control signals for the switching devices in the power converter so that appropriate current is regulated in the stator circuits. The switching control signals are calculated based on a control strategy, the feedback signals (current, voltage and rotor position), and the power converter topology. Several common power converter topologies and control strategies are described in this section.

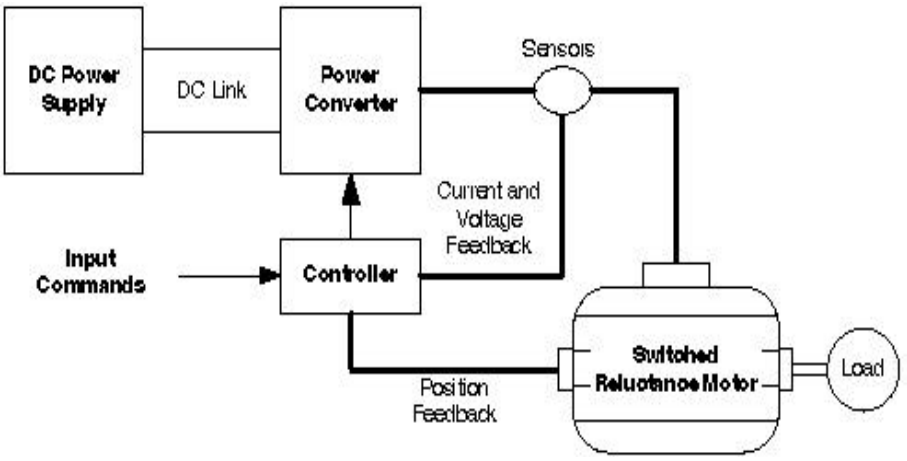

Figure 1.3: Diagram of a SRM drive system

1.4.1 Power Converter Topologies

The power converter supplies current pulses to the phase windings of the SR motor. As it was mentioned, the torque produced in the SRM is independent of the direction of current °ow in the motor phases. This makes many converter topologies possible. Various existing converters were reviewed and compared in [19]. Among the generally applicable converters, split DC and the asymmetric half bridge converters have attracted more attention because the °exibility of these three

converters is coupled with the lowest VA ratings, which in many cases provides a cost-effective solution [19]. The operation methods, advantages and limitations of these converters are described below.

A. Split DC Power Converter

A typical split DC power converter circuit connected to two phases is shown in Fig. 1.4. This converter requires only one power semiconductor switch per phase. It also requires a three-wire DC supply. For this purpose, the single DC-link capacitor is replaced by two capacitors in series. When the phase winding switch S_1 is closed, current flows through phase winding one. The freewheel diode is reverse biased, with a potential drop of V_S across it. The voltage across C_1 is applied to the phase. When the phase winding switch S_1 is opened, current freewheels through the phase winding diode D_1 and the voltage across C_2 is applied to the phase winding, decreasing the phase current. In this circuit, the energy flows to and from the capacitors must be balanced to maintain the common point voltage, and some loss of phase winding switching independence may result. Current ripple in the DC-link capacitors is large.

B. Asymmetric Bridge Power Converter

The asymmetric half bridge converter has two semiconductor switches and two freewheel diodes per phase. Fig. 1.5 shows its topology. With this type of converter, three modes of operation are possible:

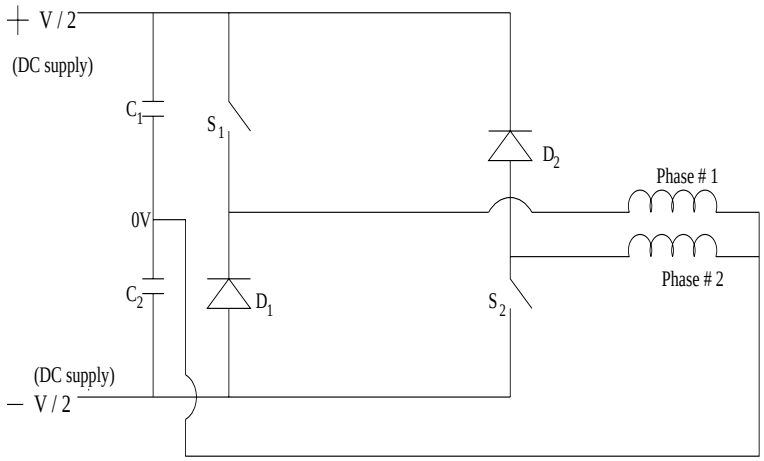

Figure 1.4: Split-rail power converter for two phases of the switched reluctance motor.

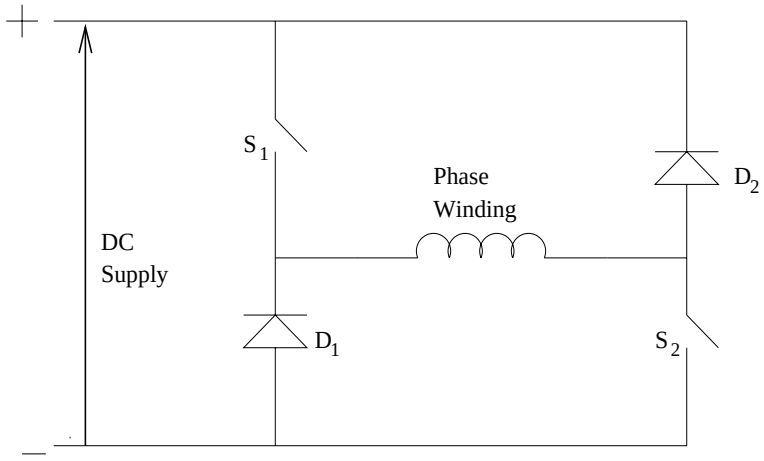

Figure 1.5: Bridge converter for one phase of the switched reluctance motor.

Mode 1 - Positive Voltage: The two power switches S_1 and S_2 are turned on. Thus, full DC link voltage is applied to the phase winding and phase current builds up.

Mode 2 - Zero Voltage: One of the switches is turned off. The phase current free-wheels through the other switch and one diode. The phase current decays slowly with a zero phase voltage.

Mode 3 - Negative Voltage: Both the switches S_1 and S_2 are turned

off. Therefore, the phase current will transfer to both of the freewheeling diodes and return energy to the DC supply. When both of the diodes are conducting, a negative voltage with amplitude equal to the DC power supply voltage level is applied to the phase winding.

This type of converter provides the most flexibility while maintaining phase independence. This advantage is very important especially when phase overlapping is used to improve the torque capability/control of the motor. Moreover, this converter circuit is more fault tolerant since, even if one of the phases has a fault, it is still possible to operate the SRM using the other phases with lower motor torque output. Therefore, in most applications, the advantages of this type of converter outweighs the disadvantages of extra cost due to more switching components per phase. Hence, the asymmetric half bridge converter will be used for the experimental SRM system developed for this research.

1.4.2 Control Strategies

The key to effectively control the SRM lies in the ability to control two parameters: *how much* of the current flows to the phase winding and *when* the current flows, with respect to the rotor position. The ideal waveform of the phase current for maximum production of motoring torque is a controlled current pulse flowing during the increasing inductance period only. However, this type of current waveform is difficult to produce in all practical circumstances primarily due to the inductance of the stator phase winding and the back EMF. This is explained as follows.

Using equation (1.4), i.e, neglecting magnetic saturation and mutual inductance, the following equation can be derived from equation 1.1.

$$\begin{aligned} v_j &= Ri_j + \frac{d\psi_j}{dt} \\ &= Ri_j + \frac{d(L_j i_j)}{dt} \\ &= Ri_j + L_j \frac{di_j}{dt} + i_j \frac{dL_j}{d\theta}\frac{d\theta}{dt} \\ &= Ri_j + L_j \frac{di_j}{dt} + i_j \frac{dL_j}{d\theta}\omega. \end{aligned} \tag{1.7}$$

The term $i_j \frac{dL_j}{d\theta}\omega$ is the motional EMF (back EMF) and, as expected, it increases with the motor speed ω. At low speed, the back EMF and the term Ri are small in value, therefore, the current can build up quickly when the inductance is low. Once the current reaches the desired value, the current will continue to rise unless the phase voltage is switched off. Therefore, the current magnitude should be regulated by switching off and on of the phase voltage by pulse-width modulation (PWM). This mode of control is termed as the "chopping mode control".

One of the chopping methods is the "voltage-PWM" [7]. With this method, the switching devices are operated with a constant PWM duty cycles during each phase's on-time, controlling phase current and the rate of current change. With this chopping method, the current waveform does not "hit" the overlapping pole-corners with a high current, thus produces quieter operation. However, this chopping method is not suitable for precise torque control because with this method the current is not well regulated and is more sensitive to DC link voltage ripple.

Another chopping method is "current-PWM" [7], in which the current is

regulated within a hysteresis band. This chopping method is alternatively called "current hysteresis control". In this method, the power devices are switched off or on by comparing actual and desired motor currents. The power-supply voltage is fixed, so the switching frequency varies with current error. This method has a better current regulation property compared with voltage chopping method. With a small hysteresis band, the current waveform has less ripples and has a more likely rectangular profile. However, it should be noted that smaller hysteresis band requires faster switching frequency.

In both voltage-PWM and current-PWM, hard chopping and soft chopping may be used. Hard chopping refers to switching the power devices synchronously while soft chopping refers to switching the high-side power device with PWM pulses while leaving the low-side device on during the dwell period. Compared with hard chopping, soft chopping is more desirable because it helps to reduce current ripple.

Fig. 1.6 shows the soft chopping waveforms of the SRM when operated under current hysteresis control. It can be seen that, at low speeds, the current waveform can roughly approximate to a rectangular profile.

At higher speed, the motor back EMF in the rising inductance region is significant relative to the DC supply voltage, and hence slows the rise and fall of current in the phase winding. Therefore in practical chopping mode control, especially when the motor is running at a high speed, the turn on angle θ_{on} is placed somewhat before the rising inductance region to ensure the current can

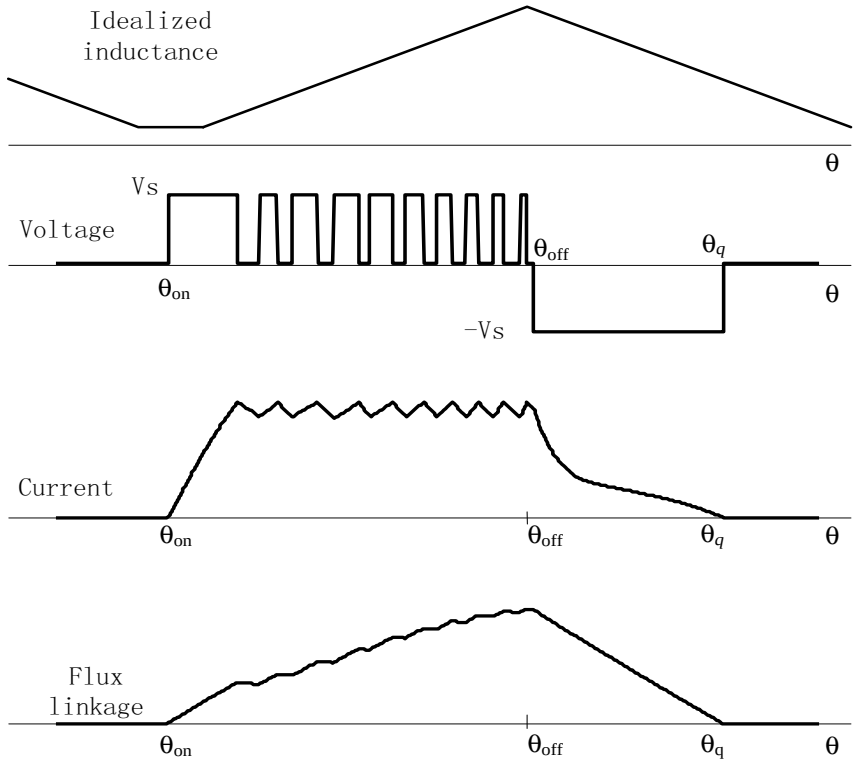

Figure 1.6: Current PWM waveforms - soft chopping.

rise quickly to the maximum level in the minimum inductance region before the rising inductance region. Similarly, the turn off angle θ_{off} is placed a little before the maximum inductance region so that the current has time to decay before the decreasing inductance region to avoid producing negative torque. However, due to the high speed, the rotor position reaches the turn off angle before the current reaches the hysteresis level. Therefore, the current will be switched on and off only once per cycle and hence the motor is operated in "single pulse mode". Typical waveforms at high speed are shown in Fig. 1.7. In this mode, PWM control is ineffective and the torque is controlled by regulating the duration of the current pulse.

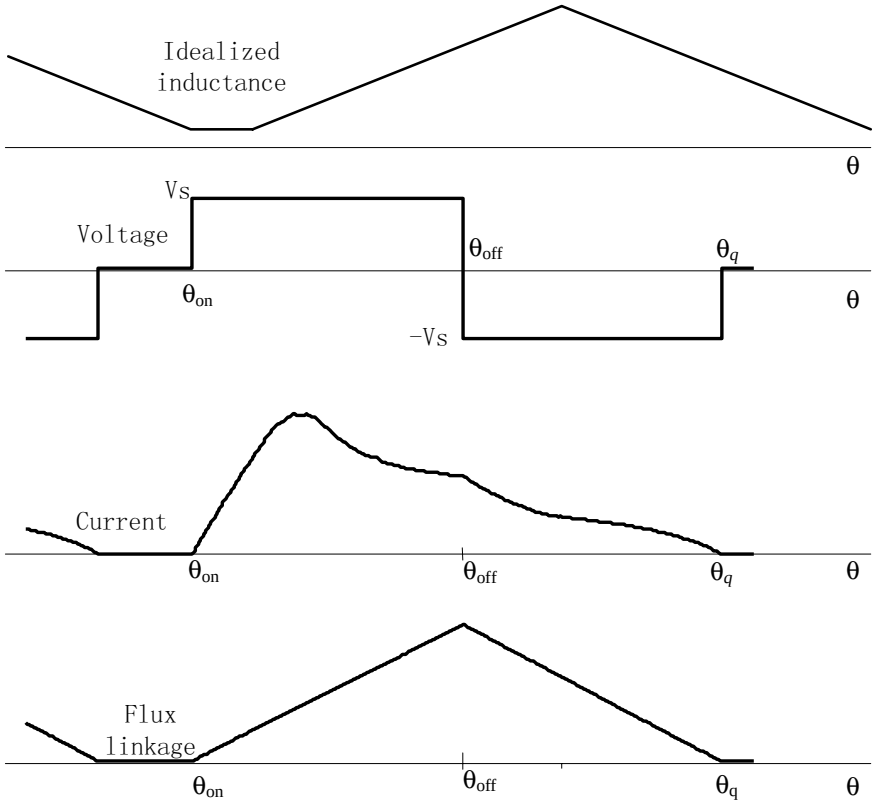

Figure 1.7: Single pulse waveforms.

In conclusion, at low speeds, the motor can be controlled by controlling the current hysteresis level as well as turn on and turn of angles while at high speeds, the motor can be controlled only by the turn on and turn off angles. Various control schemes have been proposed to achieve desired characteristics (such as constant speed, constant torque, constant power) by controlling either one or more of these parameters. These control methods include feedback linearization control [20, 21], sliding mode control [22], variable structure control [23], self-tuning control [17], fuzzy logic based control [24]. Detailed discussion of these control schemes is beyond the scope of this thesis. Interested readers may refer to related publications.

1.5 Literature Review of SRM Modelling and Control

Any application of modern control theory for the control design at any level generally rests upon analytical models of the flux-linkage and torque characteristics of the SR motor. Because of high nonlinearity, this is a difficult modelling problem. In the past, several different models for flux-linkage characteristics of SRM have been proposed which take magnetic saturation into account [23, 25, 26, 27, 28, 29, 30]. However, these methods suffer from either relatively low accuracy or high complexity. Look-up table is another popular method of description of the SRM system [31, 32]. The basic design objective is to build a two dimensional look-up table from an exhaustive magnetic measurement of the SRM. Look-up table allows fast computation but on the other hand needs a large memory to store the numerical table. Moreover, the numerical table look-up method normally requires accurate measured data for the table look-up operation, and thus sensitive to noise and error. Fahimi *et al* presented a method by which ANNs are employed to find the current to torque mapping between measured data and estimated data [34]. However, this paper does not show the experimental results for torque mapping. O'Donovan *et al* presented an approach for deriving torque model which allows ANNs to learn the non-linear flux-linkage characteristics while also incorporating a *prior* analytical knowledge of the torque production mechanism of the machine [37]. However, there are relatively high discrepancies for static torque mapping between measured data and estimated data and the paper does not show the experimental results for

torque mapping.

The highly nonlinear dynamic equations of SR motor makes it difficult to control with high performance. The classical linear control schemes can not provide the high dynamic performances required by variable speed regulation or position tracking. The earlier works on closed-loop control of SR motors, for instance [38, 39, 40], have proposed the so-called angle controller with a switching type commutation logic. Such controllers, however, do no permit adequate compensation for the nonlinearity of the motor characteristics. On the other hand, the increasing availability of inexpensive high-power fast switching devices and high-speed digital signal processors (DSP) has motivated the design of more sophisticated controllers based on nonlinear control techniques. For instance, the recent works in [25, 41] have considered magnetic saturation in the controller design. In [25], Ilić-Spong *et al.* have utilized the differential geometric control theories [42] to find a feedback controller that can linearize the dynamic equations of the SRM so that linear control techniques can be applied. Unfortunately, their controller is computationally quite complex and requires the accurate models of motor and load. To reduce the computational burden and the sensitivity to model uncertainties, Wallace and Taylor [41] have employed a kind of simplified linearization technique. Specifically, they have used a high-gain controller for direct control of stator currents and have made use of the inverse relationship of the torque generated by phase # 1 to stator current and rotor position to make the total torque linear to torque command. However, they have not considered the physical limitation in stator currents. In

[20], Kim and Ha have taken an approach where two optimal free functions have been chosen to design a controller to reduce the torque ripple. One function is meant for minimal rate of change in current commands. However, this function has been chosen under the constraints that speed and torque command are constant which is strictly not satisfied in closed-loop operation. When the second free function (meant for minimal power loss) is chosen in the design, comparatively larger current tracking error results. There are also more works reported using this feedback-linearisation approach for both speed [43] and position control [21]. Both of these works are based on the flux-linkage model proposed in [25] which models the Fourier series approximation of the function (that controls the curvature of the flux-linkage characteristics) by just the first harmonic term and that is not very much accurate. The control law proposed in [43] requires the acceleration signal (derivative of motor speed) which is not desirable for real-time implementation. Further, as expected, the control law derived in [21] is also quite involved.

Besides feedback linearization schemes, another potential control approach, i.e., variable structure control (VSC), has also been attempted for SRM control. One such work has been reported by Buja *et al.* [23]. However, here, the control law is based on a linear torque model and, thus, will certainly give poor performance in real-time. Another good work has been carried out by Rossi and Tonielli [22] where they have used both feedback linearization and sliding mode control (SMC). In this work, fluxes are taken as the state variables in the model. The feedforward torque compensator, which makes use of "torque-sharing" flux functions, generates

the desired flux vector. The inner flux controller design is based on sliding mode (SM). Recently, a hybrid approach combining fuzzy logic and feedback linearization has been proposed in [44] where, again, the drawback is the incorporation of the acceleration signal in the control law in addition to the extremely simplified modelling (as in [43]). There has also been some more good works on fuzzy logic based control [16, 24]. In [24], an adaptive fuzzy system has been used to learn the current profiles so that torque ripple is minimized. In [16], the authors have used a sliding mode based fuzzy logic controller (which takes position error as input) along with the fuzzy adaptive system [24] for learning of current profile. However, in both the cases, the performance obviously depends upon the linguistic *a priori* information and the parameters of the fuzzy system.

In another good work by Ilić-Spong *et al.* [45], a reference frame transformation has been carried out to express the dynamic equations of the SR motor in a rotating reference frame under the assumption of constant speed. Then, again, the concept of torque sharing functions (TSFs) has been introduced in an attempt to eliminate the torque ripples. A TSF for any phase indicates the fraction of the total motor torque to be shared by that phase. The limitations of this scheme for high dynamic performance are attributed to the assumption of linear magnetics of SRM in the reference frame transformation and the assumption of constant speed. Nevertheless, this study is a good step towards instantaneous torque control of SRM. Another constant torque control algorithm has been suggested in [46] where a steady state control strategy has been derived based on prediction and, later,

an optimal decision strategy has been incorporated to make the motor relatively impervious to perturbation. However, the overall performance of the algorithm depends upon the accuracy of the model which is derived from the measurement data.

Another good control scheme, self-tuning control, has been reported by Tandon *et al.* [17]. Here, the OFF angle for the phase current pulses are determined on-line by a self-tuning algorithm while the ON angle is read from a look-up table based on current and speed. The limitation with the basic scheme, here, is that the set speed and load torque should remain constant for the duration of the tuning process. If either of these variables changes, it is ideally necessary to start the tuning all over again. To overcome this difficulty, the authors have proposed a modified/hybrid approach. In another study [18], a fuzzy tuner based on Lyapunov theory has been designed to tune the dwell angle.

In the three-level cascade control architecture, the dynamic performance of the closed-loop, depends greatly on a good design of the feedforward compensator which, effectively, does the inverse mapping of the flux-linkage−current relationship of the SRM at any particular rotor position. The basic design aim is to build a two-dimensional look-up table from an exhaustive magnetic measurement of the motor. Several different approaches have been proposed in the past. In [31], a bi-cubic spline interpolation has been used for interpolation of experimental flux-linkage−current−position data. In the same work, this interpolation model has also been used to estimate torque and rotor position. An analytical method based

on linear magnetics of SRM has been proposed in [47] to find the optimal current profile using fast Fourier transform (FFT), where the stator currents are optimal in the sense that minimum average power is used to develop the specified average torque with minimum ripple. However, since the used torque model is linear, the performance is obviously degraded in a nonlinear model. In [48, 49], Lovatt and Stephenson have proposed a technique for generation of optimised current waveforms to give the lowest root-mean-square (RMS) current per phase for a given mean torque under the constraints of converter volt-ampere(VA) ratings. The same authors, in [50], have proposed another related design of the current waveforms which give rise to ripple-free torque. In both the works [48, 50], the authors have not fully addressed the limitations of finite bandwidth of the stator circuit in their design.

The brief survey of the recent developments in the SRM modelling and control clearly highlights the importance of this motor in the present technology. Because of highly nonlinear magnetic characteristics of this motor, it is really an extremely difficult task to achieve an accurate SRM model and a superlative degree of dynamic performance with specific regard to torque ripple elimination. In this research, we aim to get accurate flux-linkage model, torque model as well as inverse torque model of SRM for high performance torque control.

1.6 Motivation and Overview of This Thesis

The above rigorous study of previous works in SRM modelling and control has aroused immense inspiration, interest and motivation to search for powerful techniques for modelling of SRM for torque control since the torque control is at the core of all higher level control tasks. This thesis aims at investigations on application of intelligent tools for SRM modelling and torque ripple minimization. There are two general classes of modelling approaches, either based on analytical model, or based on balckbox such as artificial neural networks(ANNs), fuzzy neural networks(FNNs), etc. The analytical model is a kind of heuristic one, established in terms of the flux saturation phenomenon. The analytical model based approach has taken full advantage of the known domain knowledge and the known physical structure of the model. It renders the modelling problem into simple state dependent parameter search problem. Therefore it is the most common approach for SRM modelling. However, since the physical model is highly nonlinear, it is impossible to derive the inverse torque model according to the analytical forward torque model. The blackbox based approcah, on the other hand, is a universal one and solely data driven. Hence it can be applied for torque modelling as well as inverse torque modelling and significantly reduce the modelling difficulty. In this work, we use typical modelling methods in both classes to investigate their suitability or applicability to SRM. The main contribution of this thesis lies in the following two parts:

- An accurate nonlinear dynamic model is the basis for further investigations of the SRM. In this thesis, two classes of modelling approaches, i.e., analytical model based approaches and blackbox based approaches are applied for the modelling of the SRM. For the analytical model based approaches, a Levenberg-Marquardt(LM) gradient expansion method and a genetic algorithm(GA) are used for flux-linkage modelling and torque modelling. For the model-free blackbox based approaches, ANNs with different algorithms are employed for flux-linkage modelling, torque modelling as well as inverse torque modelling. Simulation and experimental results verify the effectiveness of the derived models for achieving high accuracy and their respective advantages.

- Determination of suitable current profiles for SRM is a challenging problem. In this thesis, we propose a new TSF and formulate it as an optimal design problem. Subsequently, we formulate the problem with distinct phases into a multiobjective optimal design problem and propose a new fitness function. GA is employed to determine the desired current waveforms of the SRM for torque ripple minimization through generating appropriate reference phase torques for a given desired torque. Simulation results show that the design parameters can be automatically selected by GA and much smoother current waveforms are generated when comparing with conventional TSF design using heuristic knowledge and therefore verify the effectiveness of the proposed TSF and fitness function.

This thesis is organized as follows.

In Chapter 2, analytical model based approaches and blackbox based approach for the flux-linkage modelling of SRM are described. For analytical model based approaches, LM gradient expansion method [51] and GA [52] are employed. For blackbox based approach, ANNs are applied for flux-linkage modelling. Simulation results verify the effectiveness of ANNs for achieving high accuracy in flux-linkage modelling.

Chapter 3 begins with applying the flux-linkage model into the standard electric machine theory for torque prediction. However, it is not accurate because of the possibly unavoidable measurement error. Then both analytical model based approaches and blackbox based approaches for torque modelling and inverse torque modelling of the SRM are described. For analytical model based approaches, we let GA provide coarse search, and LM gradient-expansion method refine the search to achieve an accurate analytical torque model. For blackbox based approaches, ANNs with different algorithms are applied for torque modelling and inverse torque modelling. Simulation and experimental results verify the effectiveness of the derived torque models for achieving high accuracy and their respective advantages.

In Chapter 4 the TSF approach is proposed to address the problem of minimizing or eliminating the torque ripple for high performance torque control. First we propose a new TSF and formulate it as an optimal design problem. Subsequently, we formulate the problem with distinct phases into a multiobjective

optimal design problem and propose a new fitness function. GA is employed to determine the desired current waveforms of the SRM for torque ripple minimization through generating appropriate reference phase torques for a given desired torque. Simulation results verify the effectiveness of the proposed TSF and fitness function.

Chapter 5 concludes this thesis highlighting the major contributions of this research. It also includes recommendations on possible future research directions that may be pursued based on the insights gained from this research.

Finally, the most common symbols used in the descriptions of SRM are listed in the Nomenclature in the beginning pages of this thesis. The other symbols used specifically in the design of any particular system are defined before using them. There are some more conventions adopted in this thesis. They are listed below.

Since the electromagnetic events over an electrical cycle (rotor pole pitch) repeat themselves throughout the entire operation of SRM, in this study, some designs of the control systems based primarily on this periodic nature have been investigated over the electrical cycle. For the sake of convenience in handling electrical cycles, the designs of the control systems use electrical angle θ_e ($\theta_e = N_r\theta$) rather than mechanical angle (θ). However, there are some cases, i.e., the flux-linkage characteristics modelling and the torque characteristics modelling of the SRM, the mechanical angle has been used to offer a better picture. Hence, for the sake of clarity and to avoid any sort of confusion, in this thesis, the notion of the angle (electrical/mechanical) used has always been mentioned first followed by

the design procedure. Wherever electrical angle has been used, they are indicated clearly labeled by θ_e. Wherever mechanical angle is used, they are simply mentioned without the adjective "mechanical" and are labeled by θ.

Chapter 2

Flux-linkage Modelling of the SRM

2.1 Introduction

Direct-drive motors are generally employed in servo applications where high performance is expected. The performance of SRM is influenced by its flux-linkage characteristics and control strategy. For works aiming at minimizing torque ripple for the SRM, an accurate nonlinear dynamic model of the motor is the basis for further investigations. Due to the fact that SR motors usually operate in saturation, in order to obtain maximum of its torque-to-mass ratio, the linear model of SRM described in Chapter 1, which is often used in the early control schemes due to its simplicity, is inaccurate. All the pertinent characteristics of the machine model, i.e., flux-linkage, inductance and phase torque are highly nonlinear functions of both rotor position and phase current. Magnetic saturation introduces significant nonlinearity and as a result, it is a rather challenging task to derive an accurate mathematical model for flux-linkage characteristics and torque characteristics. In

this chapter, we aim to acquire an accurate flux-linkage model of SRM through various efforts.

In this chapter, analytical model based approaches and blackbox based approach for the flux-linkage modelling of SRM are described. For analytical model based approaches, a LM gradient expansion method [51] and GA [52] are employed. For blackbox based approach, ANNs are applied.

This chapter is organized as follows. Section 2.2 describes the flux-linkage measurement for the SRM. Section 2.3 shows the procedure of analytical model based approaches for SRM flux-linkage modelling using LM gradient expansion method and GA. Blackbox based approach, in which ANNs are employed, is applied for SRM flux-linkage modelling in Section 2.4. Finally Section 2.5 concludes this chapter.

In this Chapter, a 1 hp, 8/6 pole SRM has been used. All the four phases are assumed to be identical and symmetrically located with respect to each other.

2.2 Flux-linkage Measurement for the SRM

For our experimental SRM, the flux-linkage data are obtained by the procedure outlined in [53], i.e., measurement of rising current at constant voltage. The measurement procedure is based on the voltage equation (2.1) of the SRM,

$$\frac{d\psi_j}{dt} = v_j - Ri_j \ , \ j = 1, \cdots, 4. \tag{2.1}$$

From this equation, the following equation can be derived where $\psi(0)$ is the flux-linkage at $t = 0$.

$$\psi(t) = \psi(0) + \int_0^t (v(t) - i(t)R)dt. \tag{2.2}$$

Thus, the flux-linkage at any instant can be computed from the terminal measurements of the phase winding voltage and current. The integration to obtain flux-linkage is performed by software numerical integration, i.e., Simpson's Rule. In the experimental arrangement, a graduated disc is mounted on the shaft of the motor. The disc is also provided with a strong mechanical locking arrangement. Firstly, the rotor is mechanically blocked (locked) at any particular rotor position (with respect to the aligned position of any particular phase, i.e., phase # 1). Then, a constant (step) voltage is applied to the corresponding phase winding (with the other phases open-circuited) and both instantaneous voltage and current waveforms are stored/recorded in a digital storage oscilloscope. The recording is terminated when the current attains its steady-state value. Since there are six electrical cycles per revolution, this measurement process is repeated for various rotor positions over the complete rotor pole pitch. A rotor position of 0° corresponds to complete alignment of phase #1 stator poles with the corresponding rotor poles(phase chosen for measurement) and 30° corresponds to compete unalignment of this phase. After completion of the total experiment, the voltage and current data are used in numerical integration (with $\psi(0) = 0$) to obtain flux-linkage characteristic curves at the various rotor positions. The flux-linkage−current relationship for each phase of the motor is the same except for the angular dependence, which takes into account the physical interpolar spacing. In our case, the flux-linkage−current relationships

for different phases are obtained by applying appropriate phase shifts. A set of numerically obtained (using measured voltage and current data) curves for phase #1 is shown in Fig. 2.1.

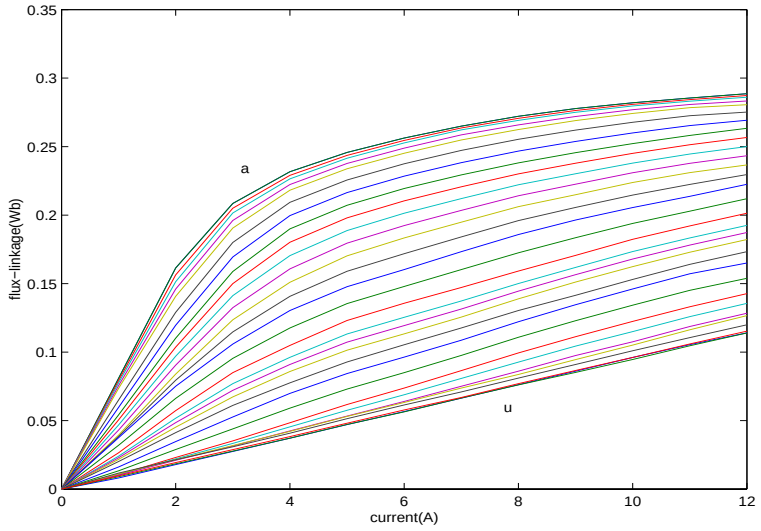

Figure 2.1: Measured flux-linkage versus current for various rotor positions for phase #1, { a ≡ aligned (0°); u ≡ unaligned (30°)}.

2.3 SRM Flux-linkage Modelling Using Analytical Model Based Approaches

2.3.1 Flux-linkage Model

Since the torque developed by an SRM is independent of the sign of the stator currents (and, accordingly, unipolar power converters are used to energize the phase windings), the case of $i \geq 0$ is considered for the flux-linkage modelling without loss of generality. Moreover, since all the phases of SRM are almost identical to each other with appropriate phase shifts between them, the modelling of flux-linkage characteristics is performed for one of the four phases (assigned to be phase #

1). The characteristics of any SRM flow from the flux-linkage−current relationship as a function of rotor position. This characteristic represents the coupling between the electrical and mechanical variables of the motor. Magnetic saturation is highly important for the high-performance operation of the SRM drive and is difficult to model analytically. Because the piecewise linearization of magnetic characteristics has limitations, it is necessary to obtain an analytic expression for flux-linkage−current−position data. The basic aim of this mathematical expression is to provide all of the flux-linkage−current information for every rotor position in one summary equation that matches closely with the experimental (or numerical) data and can have physical interpretation.

In this study, the basic flux-linkage model proposed in [26] has been used and is given in the following equation

$$\psi(i,\theta) = a_1(\theta)(1 - e^{a_2(\theta)i}) + a_3(\theta)i \ , \ i \geq 0. \tag{2.3}$$

The various coefficients a_1, a_2, and a_3 model different aspects of the characteristics. Here, a_1 is a measure of the flux-linkage as the motor enters into saturation, a_2 represents the severity of saturation (as it controls the curvature of the profile), and a_3 is the incremental inductance at high current levels (since the exponential step decays to a constant leaving a linear relationship between the flux-linkage and current). From their physical meanings, there are constraints for $a_1 > 0$, $a_2 < 0$, and $a_3 > 0$. Note the basic form of this type of analytical model has been suggested by Ilić-Spong *et al.* [25], however they have not allowed coefficient a_1 to be a function of rotor position, and in that model, there is no term involving

a_3. In fact, a first hand examination of the nature of flux-linkage profile suggests that treating a_1 to be a constant and absence of the term involving a_3 would result in an inaccurate model. These coefficients of a_1, a_2, and a_3 proposed in [26] are individually represented by a Fourier cosine series, which is very complicated. In this study, we propose an expression for a_1, a_2, and a_3 as follows

$$a_i(\theta) = a_{i1}\theta^5 + a_{i2}\theta^4 + a_{i3}\theta^3 + a_{i4}\theta^2 + a_{i5}\theta + a_{i6} \tag{2.4}$$

The numerical curve fitting is applied to closely match equation (2.3) with the data of Fig. 2.1 for the experimental SRM. For the curve fitting, the maximum value of current is taken as 12A. This limit is to be chosen by the designer based upon the experimentally obtained data for current.

2.3.2 LM Gradient Expansion Method Based Computation

A. LM Gradient Expansion Method

The algorithm of LM gradient expansion method used for the determination of coefficients in the flux-linkage model is briefly described here. It is clearly a nonlinear curve fitting problem. It is an established fact that the method of least squares [51] and multiple regression are restricted to fitting functions that are linear in parameters. In the past, there are several algorithms and search methods suggested for the nonlinear fitting problem. The LM gradient expansion method is one of the most useful ones amongst them. Its efficiency stems from the fact that it makes use of a hybrid approach, i.e., function expansion and gradient search. The algorithm

of least squares is described below.

Let the model to be fitted to the data be

$$y = f(x_1, x_2, \cdots, x_m; a_1, a_2, \cdots, a_k) \tag{2.5}$$

where $a_1, a_2, \cdots, a_k$ are the parameters to be determined, $x_1, x_2, \cdots, x_m$ are the independent variables, and y is the dependent variable. Pairs of observations $\{(y_z, X_z), z = 1, \cdots, n\}$ are obtained from experiment where $X_z = [x_{1z}, x_{2z}, \cdots, x_{mz}]$. The objective is to minimize the loss function

$$V(a_1, a_2, \cdots, a_k) = \frac{1}{n}\sum_{z=1}^{n}(y_z - \hat{y_z})^2 \tag{2.6}$$

with respect to the unknown parameters where $\hat{y_z}$ is the predicted value of y_z given by equation (2.5) at the data point. For the sake of convenience, all the unknown parameters of the model are collected in a vector $\mathbf{a}$ and all the independent (measured) variables in another vector $\mathbf{x}$. The first approach adopted here is to expand the fitting function y in a Taylor series about a point $(\mathbf{x}, \mathbf{a^0})$ to first order in the parameter increments $\delta a_l = a_l - a_l^0$:

$$y(\mathbf{x}, \mathbf{a}) \approx f(\mathbf{x}, \mathbf{a^0}) + \sum_{l=1}^{k}\left[\frac{\partial f(\mathbf{x}, \mathbf{a^0})}{\partial a_l}\delta a_l\right] \tag{2.7}$$

where $f(\mathbf{x}, \mathbf{a^0})$ is the value of the fitting function when the parameters have the starting point values $\mathbf{a^0}$ and the derivatives are evaluated at the starting point. Now, the result is a linear function in the parameter increments δa_l to which the standard method of linear least squares can be applied and the resulting solution expressed in matrix form is:

$$\boldsymbol{\alpha}\ \boldsymbol{\delta a} = \boldsymbol{\beta} \tag{2.8}$$

where ("†" denotes the transpose)

$$\alpha_{k\times k} = P^{\dagger}P \tag{2.9}$$

$$P_{n\times k} = \left[\frac{\partial f_z}{\partial a_l}\right] \ , \ z = 1, \cdots, n \ ; l = 1, \cdots, k, \tag{2.10}$$

$$\beta_{k\times 1} = \left[\sum_{z=1}^{n}(y_z - f_z)\frac{\partial f_z}{\partial a_l}\right] \ , \ l = 1, \cdots, k. \tag{2.11}$$

Thus the solutions of equation (2.8) are the parameter increments δa_l. In essence, the linear least squares method is applied to fit the parameter increments to the difference between the actual data and the starting values of the fitting function. This method works well in the majority of the circumstances. However, the convergence characteristics depends upon the curvature of the hypersurface defined by the loss-function. The central point is that, although, the analytical method converges quite rapidly to the point of minimum loss-function value from points nearby, it may not be relied on approaching to the minimum with an accuracy from a point outside the region where the hypersurface is approximately parabolic. In particular, the analytical solution is clearly unreliable whenever the curvature of the hypersurface becomes negative [51] . This problem can be overcome by gradient search which is ideally suited for approaching the minimum from points far away, but does not converge rapidly near the minimum. Therefore, there is an obvious necessity of an algorithm that behaves like a gradient search in the first portion of the search and behaves more like an analytical solution as the search converges. This is the LM gradient-expansion algorithm which is described in detail below.

The LM gradient-expansion algorithm, which combines the best features of

the gradient search with the above described method of linearizing the fitting function, is obtained by increasing the diagonal terms of the matrix $\boldsymbol{\alpha}$ by a factor of $(1+\lambda)$ that controls the interpolation of the algorithm between the two extremes [51, 54]. Thus equation (2.8) becomes

$$\boldsymbol{\alpha'}\,\boldsymbol{\delta a} = \boldsymbol{\beta} \tag{2.12}$$

where

$$\alpha'_{lt} = \begin{cases} \alpha_{lt}(1+\lambda) & \text{for } l = t \\ \alpha_{lt} & \text{for } l \neq t \end{cases} \tag{2.13}$$

If λ is very small, solutions of equation (2.12) are similar to the solutions of equation (2.8). If λ is very large, the diagonal terms of the matrix $\boldsymbol{\alpha'}$ dominate and the matrix equations degenerate into separate equations which yield the vector increment $\delta\mathbf{a}$ in a direction opposite to the gradient direction of the loss-function. The solutions of equation (2.12) can be found by matrix inversion.

Generally, the initial value of λ may be chosen small to take advantage of the analytical solution, but should be large enough so that the loss-function decreases. The procedure of the algorithm is:

1) Compute $V(\mathbf{a})$.

2) Start initially with a very small value of λ (say, 0.001).

3) Compute $\delta\mathbf{a}$ and $V(\mathbf{a}+\delta\mathbf{a})$ with this choice of λ.

4) If $V(\mathbf{a}+\delta\mathbf{a}) > V(\mathbf{a})$, increase λ by a factor (say, 10 or 20 or even higher) and repeat step 3. The factor should be decided by a judgment of performance.

5) If $V(\mathbf{a}+\delta\mathbf{a}) < V(\mathbf{a})$, decrease λ by a factor (say, same as that of step 4). Now, take $\mathbf{a^0} = \mathbf{a}+\delta\mathbf{a}$ as the new starting point and return to step 3 substituting $\mathbf{a^0}$ for $\mathbf{a}$.

In general, for each iteration, it might be necessary to recompute the parameter increments and the matrices several times to optimize λ. As the solution approaches the minimum, the value of λ will decrease. In practice, a lower limit for λ may be set.

B. Prediction of Flux-linkage by LM Gradient Expansion Method

For flux-linkage modelling, we choose the loss function as follows,

$$V = \frac{1}{n}\sum_{i=1}^{n}(\psi_{mi} - \psi_{ei})^2 \tag{2.14}$$

where n is the number of sampling; ψ_{mi} is the computed flux-linkage from the terminal measurements of the phase winding voltage and current for phase #1; ψ_{ei} is the estimated flux-linkage from (2.3). The objective is to minimize the loss function, i.e, obtaining a suitable analytical model of ψ_e to minimize the loss function V. Here ψ_e is expressed by equation (2.3). In the equation (2.3), the task is to determine the coefficients a_1, a_2, and a_3, whose expressions are given by equation (2.4). Thus the task turns to determine the coefficients of equation (2.4).

In this study, for each rotor position from 0 degree to 30 degree, LM gradient expansion method is employed to get the coefficients a_1, a_2, and a_3 in equation (2.3). After obtaining a_1, a_2, and a_3 for all rotor positions from 0 degree to 30 degree, we use polynomial regression to derive the coefficients of equation (2.4).

The curve fitting quality is sensitive to initial estimates of a_1, a_2, and a_3. A poor initial estimate will result in poor fitting. In this work, many trials have been done and finally the calculated a_1, a_2, and a_3 for all rotor positions obtaining from LM gradient expansion method are listed in Table 2.1.

Fig. 2.2 shows the calculated coefficients a_1, a_2, and a_3 values derived by LM gradient expansion method and their corresponding polynomial regression curves. The coefficients in the equation (2.4) for flux-linkage model are obtained as follows after polynomial regression

$$\begin{aligned} a_1(\theta) &= 77.7138\theta^5 - 98.3075\theta^4 + 47.8297\theta^3 - 11.6246\theta^2 + 0.7997\theta + 0.2645, \\ a_2(\theta) &= -137.1467\theta^5 + 149.4460\theta^4 - 54.1350\theta^3 + 7.5329\theta^2 + 0.5101\theta - 0.4304, \\ a_3(\theta) &= -2.3275\theta^5 + 2.6533\theta^4 - 1.2707\theta^3 + 0.3556\theta^2 - 0.0306 \times 10^{-4}\theta + 0.0016. \end{aligned} \tag{2.15}$$

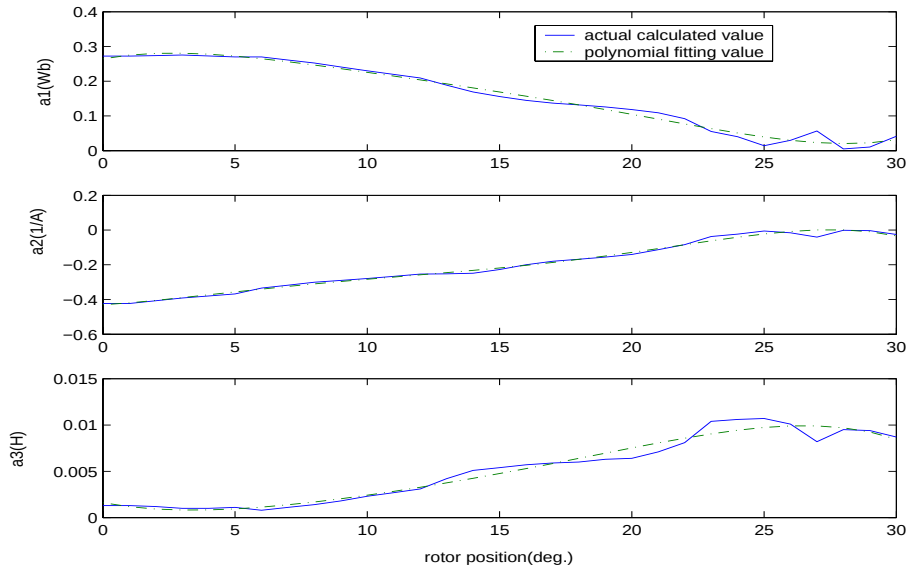

Figure 2.2: The calculated coefficients a_1, a_2, and a_3 derived by LM gradient expansion method and their corresponding polynomial regression curves.

Table 2.1: The calculated a_1, a_2, and a_3 for all rotor positions derived by LM gradient expansion method

Rotor Position(degree)	a_1	a_2	a_3
0	0.2725	-0.4232	0.0013
1	0.2725	-0.4232	0.0013
2	0.2738	-0.4074	0.0012
3	0.2753	-0.3919	0.0010
4	0.2726	-0.3801	0.0010
5	0.2701	-0.3682	0.0011
6	0.2700	-0.3347	0.0008
7	0.2610	-0.3179	0.0011
8	0.2522	-0.3006	0.0014
9	0.2409	-0.2905	0.0018
10	0.2300	-0.2794	0.0023
11	0.2195	-0.2671	0.0027
12	0.2093	-0.2536	0.0031
13	0.1886	-0.2523	0.0042
14	0.1690	-0.2491	0.0051
15	0.1558	-0.2273	0.0054
16	0.1450	-0.1998	0.0057
17	0.1370	-0.1801	0.0059
18	0.1317	-0.1682	0.0060
19	0.1260	-0.1559	0.0063
20	0.1185	-0.1404	0.0064
21	0.1090	-0.1135	0.0071
22	0.0920	-0.0842	0.0081
23	0.0552	-0.0375	0.0104
24	0.0401	-0.0237	0.0106
25	0.0143	-0.0055	0.0107
26	0.0299	-0.0161	0.0101
27	0.0566	-0.0408	0.0082
28	0.0051	-0.0013	0.0095
29	0.0104	-0.0036	0.0094
30	0.0414	-0.0255	0.0087

The loss function value of (2.14) is 1.6963×10^{-5}. According to equation (2.15), the coefficients of the flux-linkage model given by equation (2.3) can be derived and the corresponding flux-linkage model is subsequently used to predict the flux-linkage for any rotor position. The final values of the coefficients a_1, a_2, and a_3 in equation (2.3) for different rotor positions are listed in Table. 2.2. The quality of the curve fitting is illustrated in Fig. 2.3 and Fig. 2.4 for different rotor positions where the measured data are compared with those obtained from the analytical flux-linkage model (2.3) based on LM gradient expansion method. Finally, it is worth to mention again that, the fitting coefficients of the model for different phases are obtained by applying appropriate phase shifts.

C. The Advantages and Disadvantages of LM Gradient Expansion Method

The analytical Marquardt method can be viewed as computationally fast. The initial values of the coefficients can be expanded anywhere as needed. However, its disadvantages are obvious. First, there is no guarantee of a global minimum. It uses the gradient method in search of the lowest loss function. Many trials using different initial estimates are needed to achieve the lowest value of the loss function. Second, there can be many different solutions for a_1, a_2, and a_3 due to the many local minima. Third, it is extremely sensitive to initial estimates. Poor initial estimates can result in poor results. Moreover, it is found from Table. 2.2 that a_2 can not satisfy the constraint: $a_2 < 0$ for all rotor positions, although many sets of initial estimates have been tried. What is more, when the flux-linkage model is applied into the standard electric machine theory for torque modelling, the torque

prediction is not accurate which will be shown in the next chapter. Therefore GA is adopted and the computation is described in the following section.

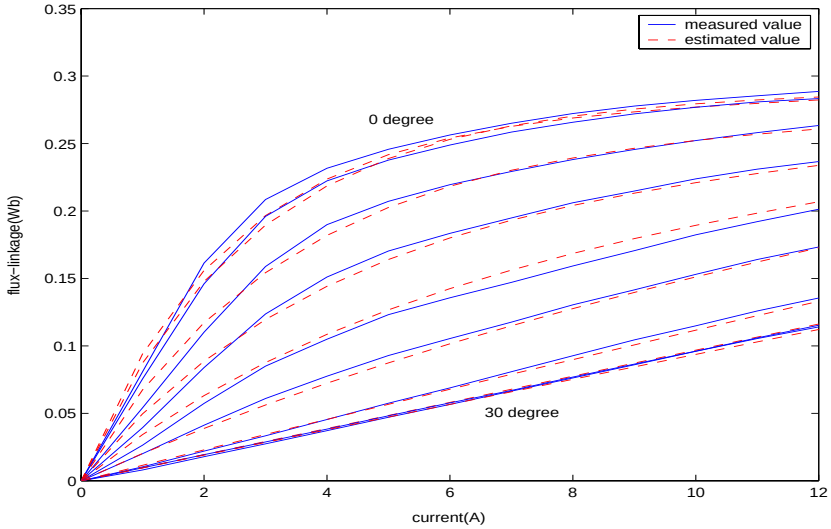

Figure 2.3: Comparison between measured and estimated flux-linkage−current data for the rotor positions of 0°(top figure),4°,8°,12°,16°,20°,24°,and 30° (bottom figure) for phase # 1 based on LM gradient expansion method.(solid line-measured value; dashed line-estimated value)

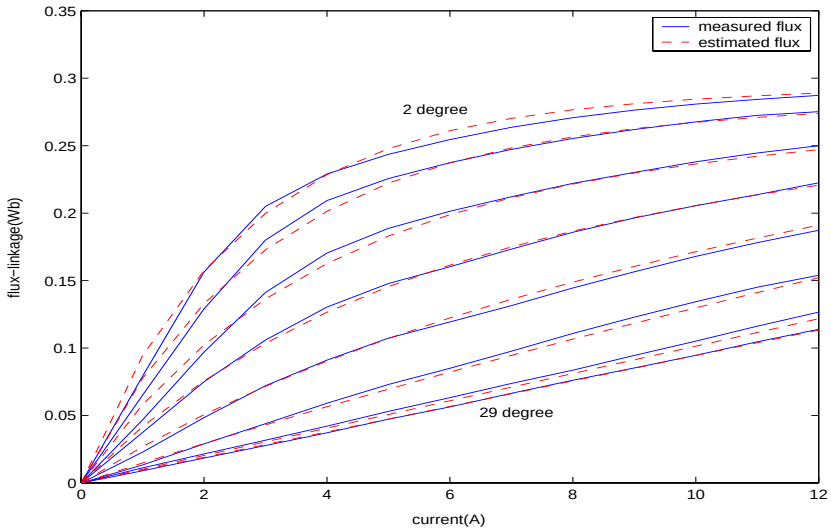

Figure 2.4: Comparison between measured and estimated flux-linkage−current data for the rotor positions of 2°(top figure), 6°,10°,14°,18°,22°,26°, and 29° (bottom figure) for phase # 1 based on LM gradient expansion method.(solid line-measured value; dashed line-estimated value)

Table 2.2: The final values of the coefficients a_1, a_2, and a_3 in equation (2.3) for different rotor positions

Rotor Position(degree)	a_1	a_2	a_3
0	0.2645	-0.4304	0.0016
1	0.2751	-0.4195	0.0012
2	0.2801	-0.4055	0.0009
3	0.2806	-0.3897	0.0008
4	0.2777	-0.3732	0.0008
5	0.2722	-0.3565	0.0009
6	0.2648	-0.3403	0.0008
7	0.2561	-0.3248	0.0011
8	0.2465	-0.3102	0.0014
9	0.2362	-0.2964	0.0017
10	0.2256	-0.2833	0.0020
11	0.2147	-0.2706	0.0024
12	0.2036	-0.2582	0.0028
13	0.1923	-0.2456	0.0033
14	0.1808	-0.2325	0.0038
15	0.1691	-0.2186	0.0042
16	0.1570	-0.2035	0.0048
17	0.1445	-0.1871	0.0053
18	0.1316	-0.1692	0.0058
19	0.1183	-0.1497	0.0064
20	0.1047	-0.1290	0.0070
21	0.0909	-0.1071	0.0075
22	0.0771	-0.0847	0.0081
23	0.0635	-0.0625	0.0086
24	0.0508	-0.0415	0.0090
25	0.0393	-0.0229	0.0094
26	0.0298	-0.0083	0.0097
27	0.0231	0.0004	0.0099
28	0.0204	0.0009	0.0097
29	0.0227	-0.0095	0.0092
30	0.0315	-0.0338	0.0085

2.3.3 GA Based Computation

The same as the above section, our objective is to find the coefficients of the flux-linkage model given by equation (2.3) so that the estimated flux-linkage from this analytical model can match the measured data as perfectly as possible. It turns out that we are dealing with an optimization problem, which can be mathematically expressed as the following

$$J = \begin{cases} \min\limits_{\mathbf{c}\in C} \{\frac{1}{n}\sum\limits_{i=1}^{n}(\psi_{mi} - \psi_{ei})^2\} \\ \\ such\ that\ a1 > 0,\ a2 < 0,\ a3 > 0 \end{cases} \tag{2.16}$$

where

$$\begin{aligned} \mathbf{c} &= \{a_{i1},\ a_{i2}, ...,\ a_{i6}\}_{i=1,2,3} \\ C &= [b_1 - 0.1|b_1|,\ b_1 + 0.1|b_1|] \cap ... \cap [b_{18} - 0.1|b_{18}|,\ b_{18} + 0.1|b_{18}|] \end{aligned}$$

where n is the number of sampling; ψ_{mi} is the computed flux-linkage from the terminal measurements of the phase winding voltage and current for phase #1; ψ_{ei} is the estimated flux-linkage from (2.3). C is a domain with estimated intervals in which the lower bounds are 10% smaller than $\{b_q\}_{q=1,2,...,18}$ and the upper bounds are 10% larger than $\{b_q\}_{q=1,2,...,18}$ for the first generation; and

$$\begin{aligned} \{b_q\}_{q=1,2,...,18} = \quad & \{-51.4214, 70.7537, -26.7524, 0.6118, 0.2098, 0.2472, \\ & -95.6801, 120.2026, -49.3132, 7.1177, 0.4047,\ -0.4132 \\ & 1.0167, -1.7023, 0.7900, -0.0506, -0.0067, 0.0009\ \} \end{aligned} \tag{2.17}$$

which are the given values according to the experiences. Thus we have the following lower bound *vlb* and upper bound *vub* for the first generation:

$$\begin{aligned} vlb = \quad & \{-56.5635\ 63.6783\ -29.4276\ 0.5506\ 0.1889\ 0.2468 \\ & -105.248\ 108.1823\ -54.2445\ 6.4059\ 0.3642\ -0.4545 \\ & 0.9150\ -1.8725\ 0.7110\ -0.0557\ -0.0074\ 0.00081\} \\ vub = \quad & \{-46.2793\ 77.8291\ -24.0772\ 0.6729\ 0.2308\ 0.3016 \\ & -86.1121\ 132.2229\ -44.3819\ 7.8295\ 0.4452\ -0.3719 \\ & 1.1184\ -1.5321\ 0.8690\ -0.0455\ -0.0060\ 0.00099\} \end{aligned} \tag{2.18}$$

The cost function J is highly nonlinear in the parametric space C. Theoretically speaking, there exist infinitely many solutions associated with the parameter space C which is essentially an uncountable set. It is impossible to exhaust the entire solution space to find the global minimum. We have to make trade-off between the best solution and computation power. GA allows us to search the whole solution space and come up with the global optimum.

A. Genetic Algorithm

GA [52] is one kind of direct search algorithms, which borrows the idea and the developing mechanism from genetic evolution and natural selection. It begins by randomly creating its population. Each individual of the population represents a search point in the space of potential solutions of the given optimization problem. Candidate solutions are combined by a crossover operator to produce offspring,

which expands the current population of solutions. Thus the individuals in the population are evaluated via the fitness function. Meanwhile a mutation operator is performed at a certain probability level to increase variation in the search space. By favoring the mating of the more fit individuals, the more promising areas of the search space are explored. The process of evaluation, selection, crossover and mutation is repeated until a predetermined number of generations are reached or a satisfied solution has been found. Fig. 2.5 shows the procedure of standard simple GA operation. The following parts describe each of the components of GA.

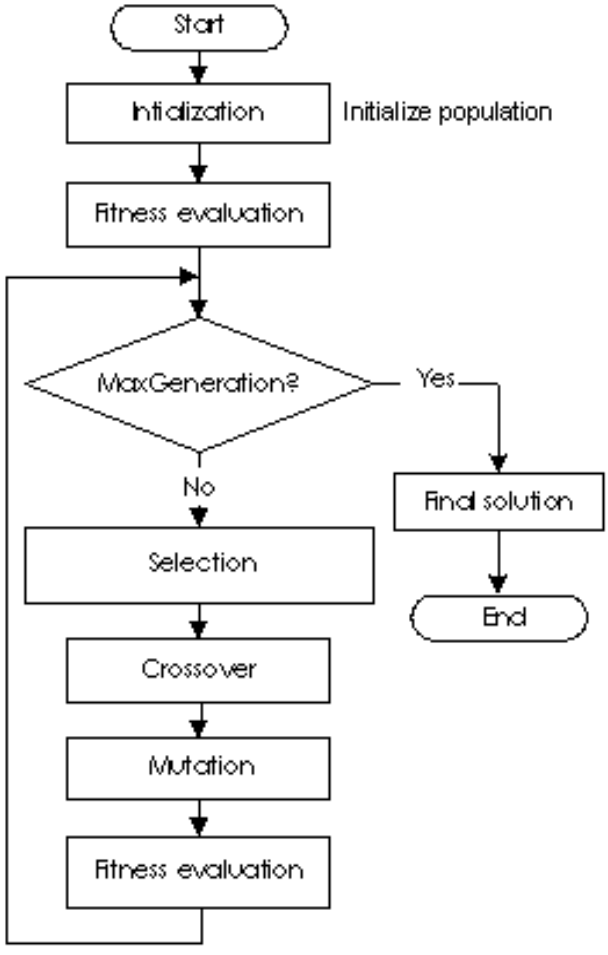

Figure 2.5: The procedures of standard simple GA operation.

1) Initiation

The initial population of candidate solution is generated randomly. Before the GA-based algorithm can be executed, a suitable encoding is necessary, which means that each possible solution in the search space can be encoded into a unique individual of binary numbers. In this study, the candidate solutions of the 18 parameters are represented by the binary values. Each individual of the candidate solutions has the appearance shown as below:

$$\begin{array}{|c|c|c|c|} a_{i1} & a_{i2} & \ldots & a_{i6} \\ xx\ldots x & xx\ldots x & \ldots & xx\ldots x \end{array} \tag{2.19}$$

where the output value of each parameter is represented by a 10-bit binary number. In the flux-linkage modelling, the selection of GA population size and generation number is a trade-off between precision and computation time. The population size is chosen to be 200 and the generation size is chosen to be 3000. The results with different population number and generation number will be shown in Section 2.3.3.C for comparison.

2) Evaluation

A fitness function must be devised for each problem to be solved as it provides the criterion for evaluating the achievement of the problem-specific objective. This procedure calculates the fitness values for all members of the population. Because our objective is to minimize the cost of the mean square error between the measured flux-linkage and the estimated flux-linkage data, we define the fitness function F as the following:

$$F(\vec{X}_i) = J^{-1} \tag{2.20}$$

where $\vec{X_i}$ represents the ith individual in the population. The larger the fitness value is, the better the individual fits. Therefore the optimization task is to search the individual, which maximizes the fitness function $F(\vec{X_i})$.

3) Reproduction

The selection, crossover and mutation operation are the reproductive phase of GA. During this phase, individuals are selected according to their fitness value, and recombined to make up the new population of individuals. The selection uses the roulette wheel method. Those individuals, which have higher fitness value, have the higher opportunity to reproduce. Each new individual will inherit some parts from either of the former individuals.

The purpose of selection is to provide more reproductive chances, on the whole, to those individuals better fit. The reproduction of individual is based on its percentage probability that is calculated through dividing its fitness value by the total population fitness:

$$P(\vec{X_i}) = F(\vec{X_i}) / \sum_{j=1}^{NN} F(\vec{X_j}) \tag{2.21}$$

where $P(X_i)$ and $F(X_j)$ represent the selection probability and fitness function value of the ith individual in the population, and NN is the population size. From this probabilistic selection operator, we can notice that the individuals with a higher fitness value have a higher probability of contributing one or more offspring in the following generation.

The purpose of crossover consists in the combination of useful string segments from different individuals to form new and better performing offspring. In this

research, one point of crossover technique is used with the probability of 0.7 and a crossover point is generated randomly. Two parent strings are cut from the crossover point to produce two "head" segments and two "tail" segments. The tail segments are swapped over to produce two new full strings. This operator decides randomly whether each bit in the parent strings is to be exchanged or not, therefore it causes a strong mixing effect which is helpful to overcome the local optima.

Another method to avoid the local optimum is the mutation operation. Mutation randomly alters each gene in the string with a small probability. A typical mutation is shown as below:

mutation point							↓							
original offspring:	0	1	0	0	1	1	**1**	1	1	0	1	1	0	(2.22)
mutated offspring:	0	1	0	0	1	1	**0**	1	1	0	1	1	0.	

By coding, each point in the search space can be represented by a unique individual of binary numbers. Each gene on the individual shares the probability of being 0 or 1 by mutation. Therefore mutation helps insure that no point in the search space has a zero probability of being explored. Small mutation probability is chosen. This is because the process will lose its characteristics of inheritance and will become a pure random optimization process, thus the algorithm will be ineffective when the probability is too large.

B. Modified Genetic Algorithm

Using GA to search for the solutions of the eighteen parameters, it is found that the population loses its diversity after a while, and all individuals converge to

one(they all look alike), or the convergence speed is very slow. In recent years, a lot of methods have been proposed to improve the performance of evolutionary algorithm based application [55, 56, 57, 58]. In this paper, To cope with these problems, the diversity (i.e. cost difference between the best and worst individual in the population) is measured after each generation. If the diversity is below a certain level, the best individual is retained and the rest population are generated randomly, and the process continues from there. In order to speed up the convergence speed of GA, the mutation process is done with an adaptive approach. The adaptive mutation rate control scheme we propose here is a variant of the constant gain method. If the fitness cost does not vary for a certain period of time, then the mutation rate is increased by a factor $\gamma \times generation$ when $5 < generation < 800$. Here we let the factor γ be 0.0005. This will greatly increase the convergence speed of GA.

Conventional evolutionary algorithms operate in a fixed search space with limiting parameters' range, which is often predefined via a-priori knowledge or trial-and-error in order to 'guess' for a suitable region comprising the global optimal solution. This method only works if the space assumed comprises the global optimal solutions. Moreover, this approach requires a long chromosome with substantially large population and generation size, which may result in extensive computational effort in order to find the global optimum. In our case, It is highly difficult to determine an appropriate boundary of parameter space that includes the optimum. Therefore an adaptive search space method described in [55] is adopted

in this research. In [55], Khor *et al.* proposed an inductive-deductive learning approach for single-objective and multi-objective evolutionary optimization. Fig. 2.6 illustrates the flowchart of the proposed deductive search space updating rule [55]. From the information of current distribution of evolved candidate solutions (given by *maxpar*, *minpar* and *prange*), the current search space of interest (in terms of *curlowbd* and *curuppbd*) is updated based on the deductive search space updating rule (as depicted in Fig. 2.6) and the parameter *pstep* to the new search space of interest (in terms of *newlowbd* and *newuppbd*). Considering n-dimensional parameter search space and $\forall i = 1, 2, ..., n$, the notations applied in the rule can be defined formally as follows:

(i) Search space of interest:

- $curlowbd(i)$ = current lower boundary of parameter search space in $i-th$ dimension.
- $curuppbd(i)$ = current upper boundary of parameter search space in $i-th$ dimension.
- $newlowbd(i)$ = new lower boundary of parameter search space in $i-th$ dimension.
- $newuppbd(i)$ = new upper boundary of parameter search space in $i-th$ dimension.

(ii) Distribution of candidate solutions: Let the $i-th$ parameter of $j-th$ evolved individual vector in the individual list of N_c candidates denotes $X_j(i)$,

- $maxpar(i)$=$max(X_j(i), \forall j = 1,\ 2,\ ...,\ N_c)$,
- $minpar(i)$=$\min(X_j(i), \forall j = 1,\ 2,\ ...,\ N_c)$,
- $prange(i)$=$maxpar(i) - minpar(i)$.

(iii) Parameter for the updating rule:

- $pstep(i)$ = additive step size of parameter search space in $i-th$ dimension.

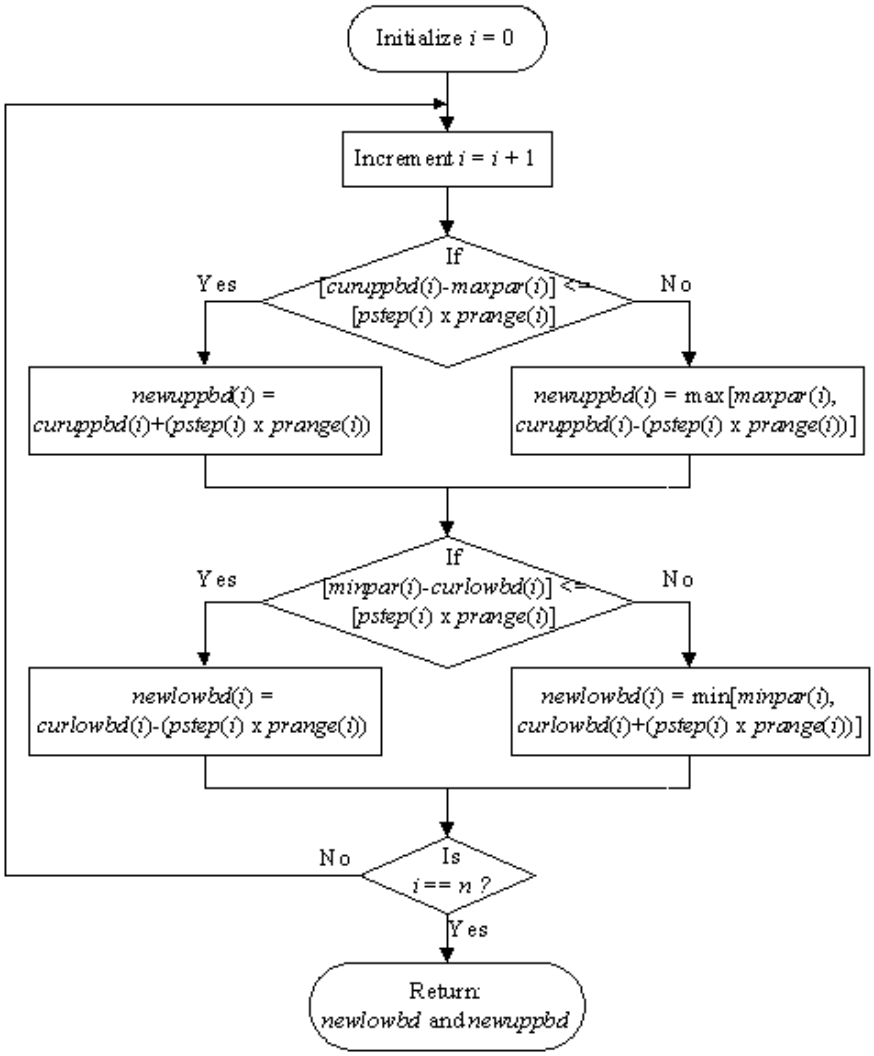

Figure 2.6: Inductive-deductive search space updating rule

Using this approach of dynamic search space for optimization is capable of directing the evolution towards more promising search regions even if these regions are outside the initial predefined space. For problems where the global optimum is included in the initial search space, it is capable of shrinking the search space dynamically for better resolution in genetic representation to facilitate the evolutionary search towards more accurate optimal solutions and facilitate the evolutionary search faster.

C. Prediction of Flux-linkage by GA

With the population size of 200 and the generation size of 3000, Table. 2.3 shows the simulation results over 30 runs, where J_{min} is the minimum value of J, J_{mean} is the average value of J, StdDev is the standard deviation of J, and the eighteen parameters $\{a_{i1},\ a_{i2},\ ...,\ a_{i6}\}_{i=1,\ 2,\ 3}$, are for the best solutions over 30 runs. It shows that two parameters are below the original lower bound and six parameters are above the original upper bound shown in (2.18). Note that with the population size of 100 and the generation size of 3000, the fitness value acquired is 1.33×10^{-4}; with the population size of 300 and the generation size of 3000, the fitness value obtained is 6.476×10^{-5}. From the simulation, the fitness value converges with the population size of 200 and the generation size of 3000 for flux-linkage modelling.

The quality of the curve fitting is illustrated in Fig. 2.7 and Fig. 2.8 for different rotor positions where the measured data are compared with those obtained from the analytical flux-linkage model based on GA.

Table 2.3: The simulation results over 30 runs for flux-linkage modelling

	$i=1$	$i=2$	$i=3$	J_{min}	J_{mean}	$StdDev$
a_{i1}	-44.8523	-82.3929	0.9536			
a_{i2}	62.5194	112.86	-1.62			
a_{i3}	-23.9304	-52.48	0.7432	6.476×10^{-5}	6.669×10^{-5}	6.037×10^{-6}
a_{i4}	0.6971	8.3204	-0.0511			
a_{i5}	0.1695	0.3934	-0.0053			
a_{i6}	0.2706	-0.4102	9.1292×10^{-4}			

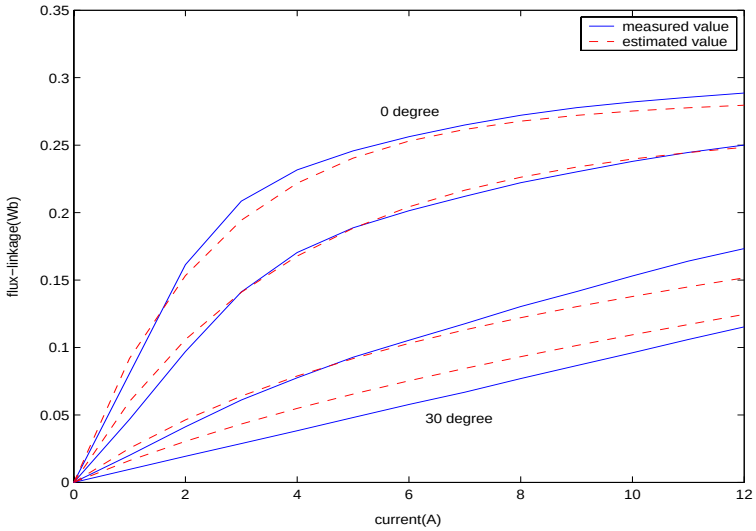

Figure 2.7: Comparison between measured and estimated flux-linkage−current data for the rotor positions of 0°(top figure),10°, 20°, and 30° (bottom figure) for phase # 1 based on GA. (solid line-measured value; dashed line-estimated value)

D. The Advantages and Disadvantages of GA

Comparing Fig. 2.3 and Fig. 2.4 with Fig. 2.7 and Fig. 2.8 respectively, it can be observed that the quality of curve fitting by LM method is much more satisfactory than that by GA, especially at the unaligned positions. It comes from two reasons: first, the results based on GA are obtained with the constraints $a_1 > 0$, $a_2 < 0$, and $a_3 > 0$; second, GA only provides coarse search randomly

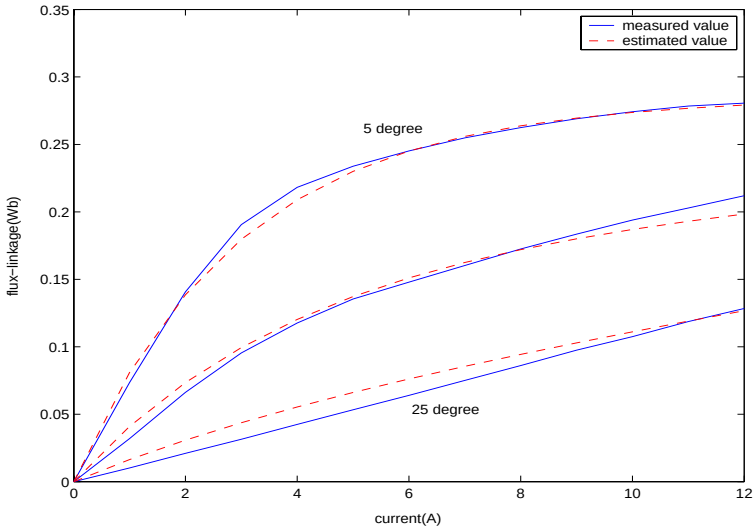

Figure 2.8: Comparison between measured and estimated flux-linkage−current data for the rotor positions of 5°(top figure), 15°, and 25° (bottom figure) for phase # 1 based on GA. (solid line-measured value; dashed line-estimated value)

for global optimum and it does not use the gradient information over the model which LM gradient-expansion method uses for finely searching. Hence GA is not efficient for finely results. From the above results, we can find that it is easy to satisfy any constraints of the parameters by employing GA. However, it is very time-consuming since quite a number of generations and populations are needed for highly nonlinear optimization problem and GA is not efficient for finding the final best solution. Therefore ANNs are applied to identify the flux-linkage model of the SRM.

2.4 SRM Flux-linkage Modelling Using Blackbox Based Approach

Ability and adaptability to learn, generalization, less information requirement, fast real-time operation and ease of implementation have made ANNs popular in re-

cent years. ANNs have been applied in a variety of areas, such as signal processing, pattern recognition and automatic control. ANN technology has the potential to accommodate an improved method of determining nonlinear model which is complementary to conventional techniques. It offers a nonlinear identification scheme which is capable of matching the nonlinear and complex structures [59]. Since ANNs do not require any prior information regarding the SRM system apart from the input and output signals, it is one kind of blackbox based approach and it is quite simple, robust and cost effective. Fig. 2.9 shows the supervised learning paradigm that is the basic way by which ANNs learn. The error signal is used to adjust the parameters to better values.

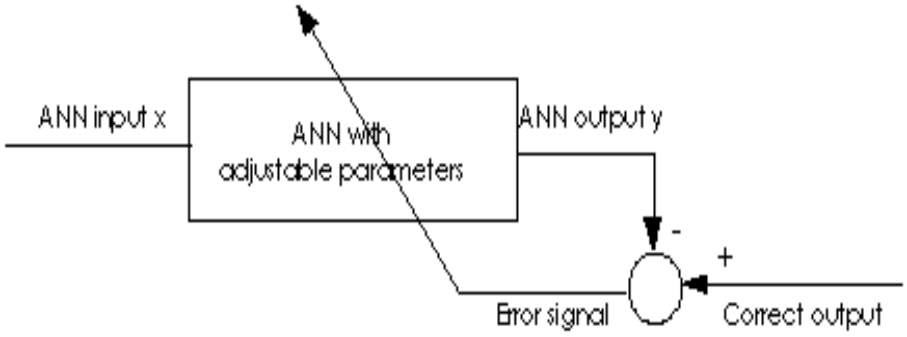

Figure 2.9: ANN learning algorithm

In this study, ANNs are employed for flux-linkage modelling to obtain the accurate mapping between measured data and estimated data. Several different neural network paradigms have been developed over the past few decades. Of the major different paradigms, BP networks constitute more that 90% of the current neural network applications. The standard BP algorithm proposed by Rumelhart and McClelland has been demonstrated on various problems [60].

2.4.1 Backpropagation Algorithm

The reasons for using this algorithm are that its structure is well understood and its recent successful applications encourage the applicants. This algorithm consists of a number of propagation errors(PEs), a transfer function for each PE in the layers, number of connections between layers and an algorithm or learning rule which is the generalized rule. The rule is simple and gives a prescription for changing the weights in feedforward NNs to map the input-output pairs.

A. BP Gradient-Based Descent with Momentum Algorithm

In the present work, the backpropagation gradient-based descent with momentum(GBDM) algorithm is used to update a multi-layered feedforward NN weights and biases[61]. Fig. 2.10 shows the topology of feedforward NNs. A multi-layered feedforward NN consists of an input layer, one or more hidden layers and an output layer. The input layer connects to the first hidden layer through an array of adjustable weights. The outputs of this hidden layer are weighted and act as inputs to the second hidden layer and so on to the output layer.

Fig. 2.11 illustrates the computation at each node within ANNs. The weighted input s_i to any neuron i in the network is written as

$$s_i = \sum_j w_{ij} x_j + b_i. \tag{2.23}$$

where w_{ij} is a weight connecting unit j of the previous layer to unit i of the present layer and x_j is the output of unit j. The bias weight b_i is the weight multiplying

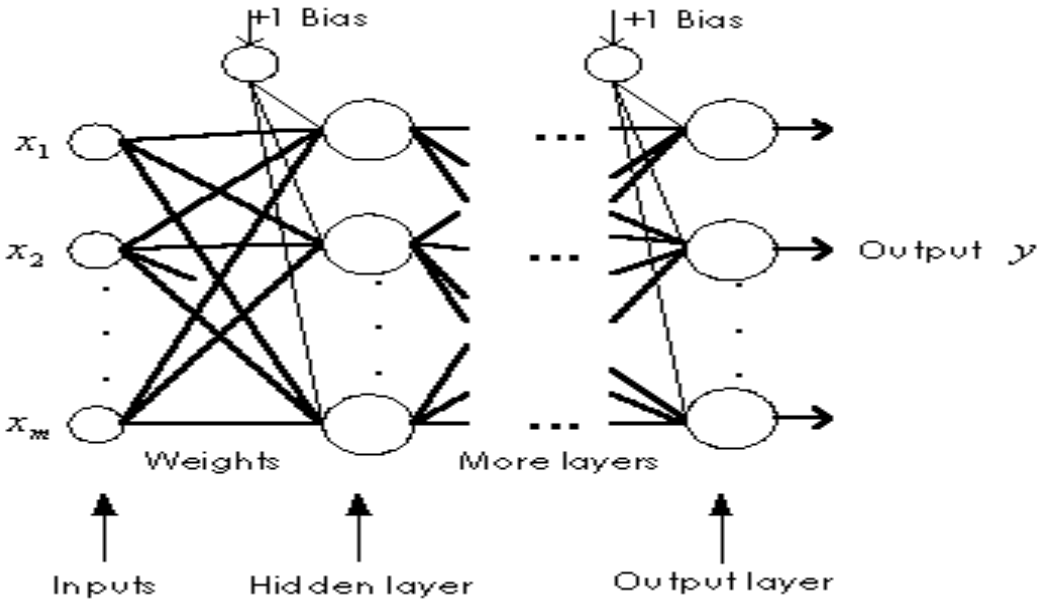

Figure 2.10: Topology of feedforward neural networks

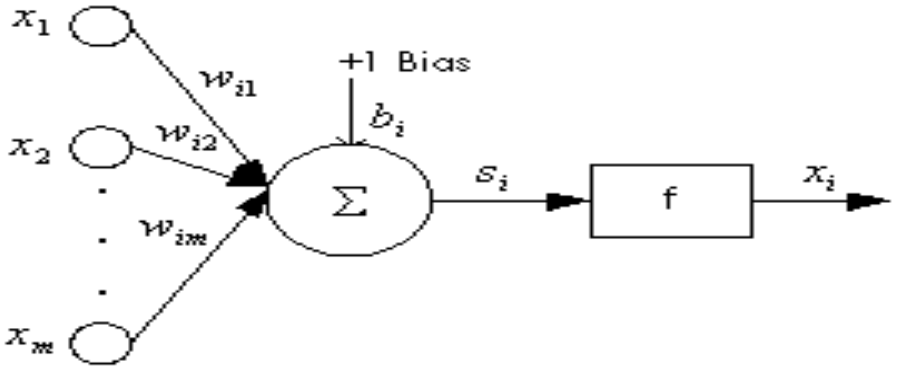

Figure 2.11: Computation at each node within artificial neural networks

the bias input (unity) and acts to shift the weighted input from the origin. The output of neuron i is

$$x_i = f(s_i). \tag{2.24}$$

where f is the neuron activation function. The following parts describe each of the components of BP algorithm.

1) Activation Function

The activation function employed by the neurons belongs to one of the several types: threshold, sigmoidal, and linear activation function. The threshold and sigmoidal activation functions play a very important role. Sigmoidal activation functions include hyperbolic tangent sigmoid transfer function − *tansig* and log sigmoid transfer function − *logsig*. The mathematical expression of *tansig* is

$$f(x) = tansig(x) = 2/(1 + exp(-2 * x)) - 1 \tag{2.25}$$

The mathematical expression of *logsig* is

$$f(x) = logsig(x) = 1/(1 + exp(-x)) \tag{2.26}$$

The mathematical expression of the linear activation function − *purelin* is

$$f(x) = x \tag{2.27}$$

It is important to note that, if we use only linear activation function for all neurons in a feedforward NN, then the overall relationship between input and output is only linear irrespective of how many neurons there are and how complicated they are connected. Thereby such a net with only linear neurons is equivalent to a single layer perceptron and hence of little use.

2) Weights Initialization

Generally, the weights between the layers are initialized with small random values. This ensures that the networks are trained easier and the functions are formulated easier. The specific technique which is used to initialize a given network will depend on how the network initialization parameters are

set. The default initialization function for the feedforward NNs in Matlab ANN toolbox is *initlay*, which allows each layer to use its own initialization function. For feedforward NNs there are two different layer initialization methods which are normally used: *initwb* and *initnw*. The *initwb* function causes the initialization to revert to the individual initialization parameters for each weight matrix and bias. The weight initialization is usually set to *rands*, which sets weights to random values between -1 and 1. It is normally used when the layer transfer function is linear. The function *initnw* is normally used for layers of feedforward NNs where the transfer function is sigmoidal. It generates initial weight and bias values for a layer so that the active regions of the layer's neurons will be distributed roughly evenly over the input space. It has several advantages over purely random weights and biases:

- Few neurons are wasted since the active regions of all the neurons are in the input space
- Training the works faster since each area of the input space has active neuron regions.

In this study, a default weights initialization method in Matlab NN toolbox is used.

3) Weights Updating Law

The square measure of error between desired network output $d(j)$ and actual

network output $y(j)$, is used to modify the network weights through learning

$$E(\mathbf{w}) = \sum_{j=1}^{N} F(j) \tag{2.28}$$

where

$$F(j) = [d(j) - y(j)]^2 \tag{2.29}$$

and the summation is taken over the entire set of training vectors. The weight modification rule is given by

$$\mathbf{w} = \bar{\mathbf{w}} + \eta d + \mu \boldsymbol{\Delta} \bar{\mathbf{w}} \tag{2.30}$$

Note $\mathbf{w}$ and $\bar{\mathbf{w}}$ denote the vector of the new weights and the old weights in the feedforward NNs respectively. d is the searching direction. It is usually referred to as the direction finding step. It is given by

$$d = \nabla E(\bar{\mathbf{w}}) \tag{2.31}$$

The quantity η is the learning rate. It should be chosen so that

$$E(\mathbf{w}) < E(\bar{\mathbf{w}}) \tag{2.32}$$

– *Learning Rate*

Although the learning rate can be chosen to any positive value, choosing it properly will dictate how fast the learning algorithm converge to a right solution. If η is chosen very large and applied to the $m{-}th$ example, then learning is very excellent as far as that example is concerned, but at the cost of spoiling the learning that has taken place earlier with respect to other examples. Thus a large value of η is not necessarily

good. If an extremely small value is chosen for η, that also leads to slow learning. Some intermediate value is the best. The learning rate can be set separately for each weight and bias. In this study, the learning rate adopted is 0.1[61].

– *Momentum Constant*

μ is the momentum constant and $\Delta\bar{w}$ is the previous weight change. The use of momentum can be greatly beneficial to the learning process. Momentum allows a network to respond not only to the local gradient, but also to recent trends in the error surface. Acting as a low pass filter, momentum allows the network to ignore small features in the error surface. Without a momentum a network may get stuck in a shallow local minimum. With momentum a network can slide through such a minimum. The momentum constant can be any number between 0 and 1. When the momentum constant is 0 a weight change is based solely on the gradient. When momentum constant is 1 the new weight change is set to equal the last weight change and the gradient is simply ignored. In this study, the momentum constant adopted is 0.95[61].

Summarizing the above procedure, the GBDM algorithm procedure is as follows:

1) *Initialization.* Choose a random $\mathbf{w}(\mathbf{0})$ to begin with. Set $k = 0$. Go to step 2.

2) *k-th Iteration.* Set $\bar{\mathbf{w}} = \mathbf{w}(\mathbf{k})$. Compute $E(\bar{\mathbf{w}})$ by forward pass and $\nabla E(\bar{\mathbf{w}})$ by using backpropagation. If the gradient is extremely small in size, or the training has reached the limit on the total number of epochs, or the mean square error has reached acceptable limits, then take $\bar{\mathbf{w}}$ to be the solution and stop. Else, go to step 3.

3) Choose the direction d based on the gradient $\nabla E(\bar{\mathbf{w}})$.

4) Choose η and μ so that the change: $\mathbf{w} = \bar{\mathbf{w}} + \eta d + \mu \boldsymbol{\Delta} \bar{\mathbf{w}}$ achieves equation (2.32).

5) Now set $\mathbf{w}(\mathbf{k}+\mathbf{1}) = \mathbf{w}$, $k := k + 1$, and go back to step 2.

This is the well-known Gradient-Based Descent (GBD) with momentum algorithm. Gradient-Based Descent algorithm is also called as Steepest Descent algorithm. Clearly, this algorithm is very slow to converge. Several numerical optimization techniques, such as Newton's method, Quasi-Newton Methods, Levenberg-Marquardt method, and Conjugate-Gradient methods can be applied to GBDM algorithm to optimize it so that it can achieve a faster convergence. All the methods fit into the format of the GBDM algorithm whose steps are outlined in the above. The methods mainly differ in the way they choose the search direction d in step 3 of the algorithm. In this study, LM method is employed to optimize the GBD algorithm and train the feedforwad NNs. Thus we explain the LM method which is employed to optimize the GBD algorithm in the following.

B. The optimized GBDM algorithm by Levenberg-Marquart method

In order to give an explanation to the Levenberg-Marquardt method, which is applied to GBDM algorithm to achieve a faster convergence, first Newton's method need be introduced.

- **Newton's method.** This method is based on a second order Taylor series analysis to determine d. As in the GBDM algorithm, let $\bar{\mathbf{w}}$ denote the current point. Expanding E in second order Taylor series around $\bar{\mathbf{w}}$, we get

$$E(\mathbf{w}) \approx \tilde{E}(\bar{\mathbf{w}}) = E(\bar{\mathbf{w}}) + \nabla E^T(\bar{\mathbf{w}})(\mathbf{w} - \bar{\mathbf{w}}) + \frac{1}{2}(\mathbf{w} - \bar{\mathbf{w}})^T H(\mathbf{w} - \bar{\mathbf{w}}) \quad (2.33)$$

where H is the Hessian matrix of E at $\bar{\mathbf{w}}$, i.e., it is a symmetric, square matrix whose $(i,j) - th$ component is the second partial derivative of E with respect to $\mathbf{w_i}$ and $\mathbf{w_j}$:

$$H_{ij} = \frac{\partial^2 E}{\partial \mathbf{w_i} \partial \mathbf{w_j}}|_{\mathbf{w}=\bar{\mathbf{w}}} \quad (2.34)$$

Newton's method minimizes $\tilde{E}$ instead of E. This is done by setting the gradient of $\tilde{E}$ to zero. This gives

$$\mathbf{w} - \bar{\mathbf{w}} = -(H)^{-1} \nabla E \bar{\mathbf{w}} \quad (2.35)$$

Newton's method uses

$$d = -(H)^{-1} \nabla E(\bar{w}) \quad (2.36)$$

in the GBD algorithm. One major advantage of Newton's method is rapidity with which it converges to the solution when the algorithm nears it. However, the main drawback of Newton's method is the need for computation of second order derivatives. Unlike the computation of ∇E by BP, the computation of second order derivatives is very expensive for BP networks. Although some

researchers have considered efficient ways of computing them, they are impractical to compute even for medium sized networks. The need for inverting H is another disadvantage.

- **Levenberg-Marquart method.** LM method imitates Newton's method in a clever fashion. It is specially suited for the popular least squares error functions (2.28) and (2.29). As we mentioned above, LM method fits into the format of the GBDM algorithm and it mainly differs in the way it chooses the search direction d in step 3 of the algorithm. For choosing the search direction d, Now

$$\nabla E = \sum_{j=1}^{N} \nabla F(j) \tag{2.37}$$

and

$$H = \sum_{j=1}^{N} P(j) \tag{2.38}$$

where $P(j)$ is the Hessian matrix of $F(j)$. From the equations (2.28) and (2.29), we can get

$$\nabla F(j) = -2(d(j) - y(j))\nabla y(j) \tag{2.39}$$

Let us write down the expression for the matrix $P(j)$ by differentiating once more:

$$\nabla P(j) = -2(d(j) - y(j))R(j) + 2\nabla y^T(j)\nabla y(j) \tag{2.40}$$

where $R(j)$ is a square matrix consisting of the second order partial derivatives of $y(j)$ with respect to the weights. It is the term which causes the evaluation of $P(j)$ to be expensive. Fortunately, $R(j)$ multiplies with the term $(d(j) - y(j))$ which is the difference between the actual and desired feeforward NN

outputs. Since the difference is expected to be small (expecially when we come close to the minimum point) we can omit the first term and approximate $P(j)$ by only the second term of equation (2.40). This leads us to the LM direction:

$$d = -(L)^{-1}\nabla E(\bar{w}) \tag{2.41}$$

where

$$L = \sum_{j=1}^{N} 2\nabla y^T(j)\nabla y(j) \tag{2.42}$$

Given the fact that $\nabla y(j)$ is a by-product of the BP computations, the determination of L is very straightforward. The optimized GBDM algorithm by LM method can achieve a much faster convergence.

The advantage of using a neural net is that its learning properties can be fully exploited to provide an analytical function representing the SRM flux-linkage characteristics. Thus, the modelling effort is significantly reduced. The SRM flux-linkage modelling results based on ANNs are described as follows.

2.4.2 Modelling of Flux-linkage Characteristics by ANNs

The BP network used in modelling SRM flux-linkage characteristics is shown in Fig. 2.12. After training, it is found that three layers network achieves the mapping task in high accuracy, including one input layer, one hidden layer and one output layer. The most suitable network configuration is $2 \times 10 \times 1$, i.e., there are two inputs in the input layer, 10 neurons in the hidden layer, and one neuron in the output layer. Fig. 2.13 shows the plot of the mean square error vs training epochs

with this BP net for flux-linkage modelling. A hyperbolic tangent sigmoid transfer function - *tansig* is chosen to be the activation function of the hidden layer. The output layer is a linear summation element. The number of training epochs is 300 and the mean square error is 2.5245×10^{-6}. It is reduced by 85.12% than the minimum loss function value of LM gradient expansion method. The actual motor flux-linkage and the neural net approximation are illustrated in Fig. 2.14 and Fig. 2.15 for different rotor positions. The worst case is when $\theta = 14°$. It contribute to 5.3% of the total error. The curve fitting is sufficiently accurate.

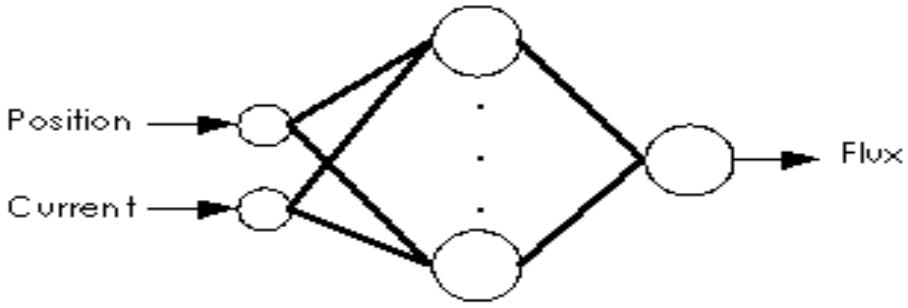

Figure 2.12: Flux-linkage approximation neural net

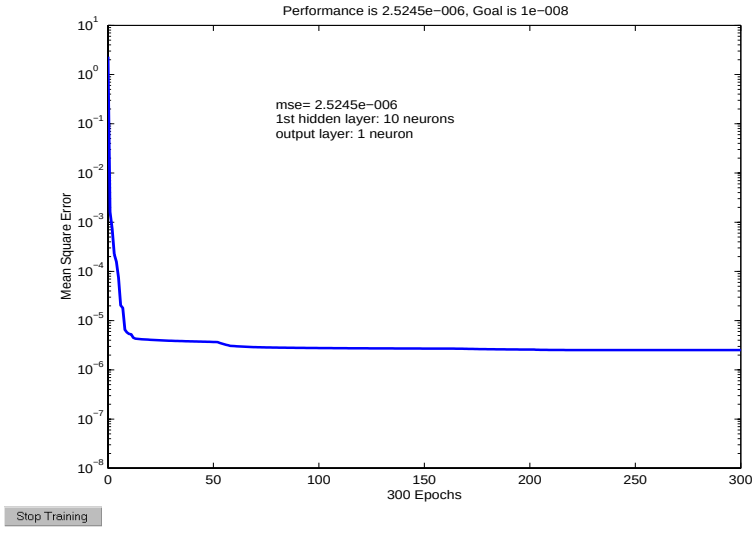

Figure 2.13: The mean square error vs training epochs with one hidden layer BP net for flux-linkage modelling

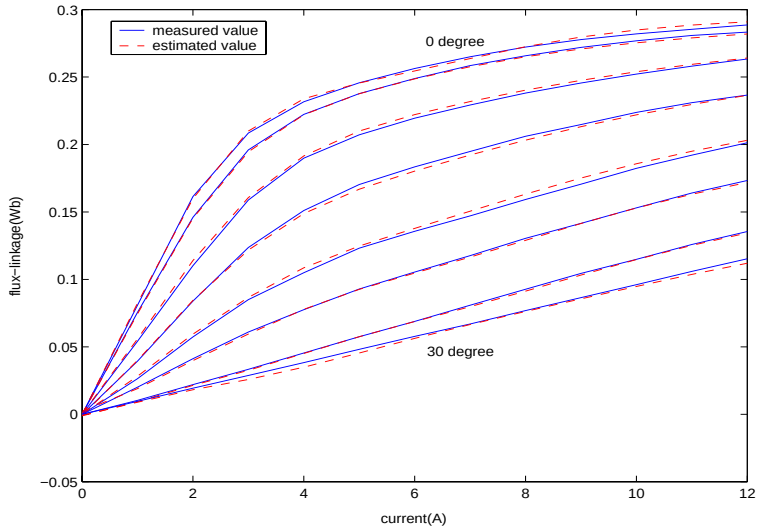

Figure 2.14: Comparison between measured and estimated flux-linkage−current data for the rotor positions of 0°(top figure),4°,8°,12°,16°,20°,24°,and 30°(bottom figure) for phase # 1 based on ANNs.(solid line-measured value; dashed line-estimated value)

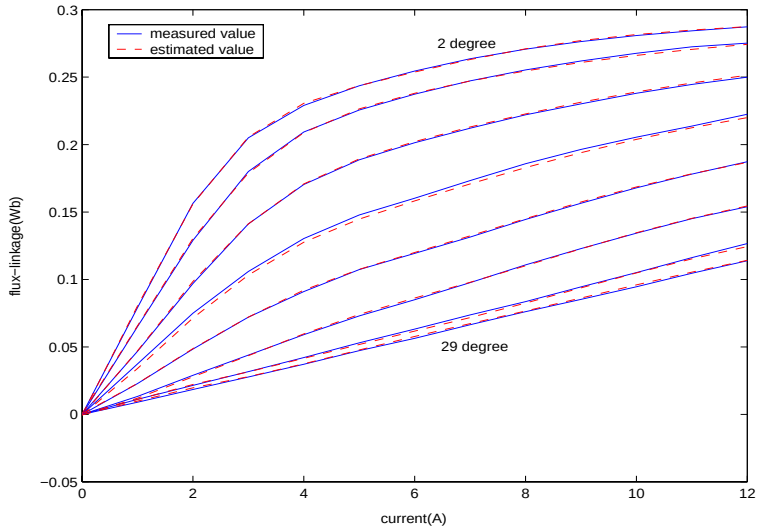

Figure 2.15: Comparison between measured and estimated flux-linkage−current data for the rotor positions of 2°(top figure), 6°,10°,14°,18°,22°,26°, and 29° (bottom figure) for phase # 1 based on ANNs.(solid line-measured value; dashed line-estimated value)

2.5 Conclusion

This chapter has been concerned with the flux-linkage modelling of the SRM. For the physical SRM, the flux-linkage has been obtained from the experimental volt-

age and current data using the method of rising current at constant voltage and numerical integration. In this chapter, various efforts have been made for the flux-linkage modelling. Both analytical model based approaches and blackbox based approach are applied for flux-linkage modelling of the SRM. For the analytical model based approaches, LM gradient expansion method and GA have been employed. The measured flux-linkage data have been fitted with a suitable analytical model whose coefficients are found by LM gradient-expansion method and GA. These coefficients are represented by fifth order polynomials. For the model-free blackbox based approach, ANN techniques are employed. Simulation results verify the effectiveness of ANNs for achieving high accuracy in flux-linkage modelling. The ANN flux-linkage model is accurate enough for torque modelling, which will be presented in the next chapter.

Chapter 3

Torque Modelling of the SRM

3.1 Introduction

Measurement of torque developed by an electric machine is an essential part in many industrial applications, since some of important objectives in SRM control are minimization of torque ripple at low speed [62], maximization of torque per ampere [63], reduction of vibration and audible noise. However, commercially available torque transducers are expensive, increase drive dimensions considerably, and do not provide high bandwidth output. Elimination of external torque sensor in SRM drive is an attractive issue in many applications. Thus predicting electromagnetic torque generated by a SRM drive using proper identification tools is very important. An electric machine processes its input current based on its physical characteristics and converts it to electromagnetic torque and hence can be considered as a current to torque converter. In this chapter we aim to acquire accurate torque model of the SRM to obtain the perfect mapping between the measured torque data and the estimated torque data.

The most common approach for torque prediction is to derive the torque model through the accurate flux-linkage model and the standard electric machine theory. However, the torque prediction using this method is not accurate because of the possible unavoidable measured error. Therefore various intelligent methods are applied for torque modelling of the SRM, including the analytical model based approaches and blackbox based approaches. For the analytical model based approaches, we let GA provide coarse search, and LM gradient-expansion method refine the search to achieve an accurate analytical torque model. Because the process for identifying the torque model by analytical model based approaches is time-consuming and it is impossible to derive the inverse torque model according to the analytical forward torque model, the model-free blackbox based approaches are adopted and ANNs with different algorithms are applied for torque modelling and inverse torque modelling. For comparison, RBF networks are also employed for torque modelling in this study.

This chapter is organized as follows. Section 3.2 introduces SRM dynamics and torque prediction by applying the flux-linkage model into the standard electric machine theory. Section 3.3 shows the procedure of analytical model based approaches for SRM torque modelling by combining GA and LM gradient expansion method. ANN based approach with BP algorithm is employed for torque modelling and inverse torque modelling of SRM in Section 3.4, in which the comparative study of RBF with BP networks is also presented. The experimental results are given in Section 3.5. Finally Section 3.6 concludes this chapter.

In this Chapter, a 1 hp, 8/6 pole SRM has been used. All the four phases are assumed to be identical and symmetrically located with respect to each other.

3.2 SRM Dynamics and Torque Prediction by Applying Flux-linkage Model

The mathematical model of the SRM is a set of controlled differential equations which are obtained by using standard electric machine theory [64]. All the equations are described with respect to a generic phase $j(j = 1, \cdots, 4)$. The differential equations that describe the SRM are:

$$\frac{d\theta}{dt} = \omega_m. \tag{3.1}$$

$$\frac{d\omega_m}{dt} = \frac{T - T_L}{J}. \tag{3.2}$$

$$\frac{d\psi_j}{dt} = v_j - Ri_j \ , \ j = 1, \cdots, 4. \tag{3.3}$$

The load torque may be a constant or a function of motor speed ω_m. In general ψ_j is a nonlinear function of θ and i_j due to magnetic saturation. Furthermore, ψ_j is periodic in θ with period $\frac{2\pi}{N_r}$ and shifted by an angle equal to $(j-1)\frac{2\pi}{qN_r}$ from the reference phase (phase # 1).

The mathematical framework for deriving the expression for torque has also been reported in Chapter 1. However, for the sake of completeness, they are again described here. The torque produced by phase j is determined by differentiating the coenergy function $W_j(\theta, i_j)$ with respect to θ, i.e.,

$$T_j(\theta, i_j) = \frac{\partial W_j}{\partial \theta} \ , \ j = 1, \cdots, 4 \tag{3.4}$$

where

$$W_j(\theta, i_j) = \int_0^{i_j} \psi_j(\theta, i_j)\, di_j. \tag{3.5}$$

Finally, the total torque produced is the sum of the phase torques, i.e.,

$$T = \sum_{j=1}^{4} T_j(\theta, i_j). \tag{3.6}$$

Fig. 3.1 shows the measured torque versus rotor position for different currents for phase #1.

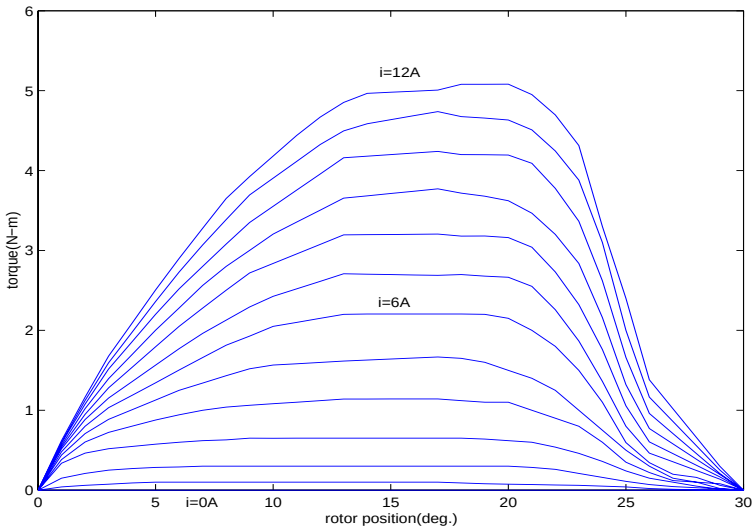

Figure 3.1: Measured torque versus rotor position for different currents from $0A$(bottom figure) to $12A$ (top figure) for phase #1.

As we know, the most common approach to modelling the torque characteristics of the SRM is to derive the torque model through the accurate flux-linkage model and the standard electric machine theory. A forward torque estimation model can be derived by replacing the flux-linkage model into (3.4), (3.5), (3.6).

The mathematical expression of the analytical flux-linkage model obtained in Chapter 2 is shown as below:

$$\psi(i, \theta) = a_1(\theta)(1 - e^{a_2(\theta)i}) + a_3(\theta)i\ ,\ i \geq 0. \tag{3.7}$$

After replacing (3.7) into (3.4) and (3.5), the mathematical expression for analytical torque model for one phase is shown as follows [65]:

$$T_j = -\left\{ i_j + \frac{1}{a_{2j}}(1 - e^{a_{2j} i_j}) \right\} \frac{da_{1j}}{d\theta} + \left\{ \frac{a_{1j}}{a_{2j}^2}(1 - e^{a_{2j} i_j}) + \frac{a_{1j} i_j}{a_{2j}} e^{a_{2j} i_j} \right\} \frac{da_{2j}}{d\theta} - \frac{1}{2} i_j^2 \frac{da_{3j}}{d\theta} \tag{3.8}$$

In Chapter 2, the coefficients of the analytical flux-linkage model have been derived by LM gradient expansion method. They are rewritten here for convenience:

$$\begin{aligned} a_1(\theta) &= 77.7138\theta^5 - 98.3075\theta^4 + 47.8297\theta^3 - 11.6246\theta^2 + 0.7997\theta + 0.2645, \\ a_2(\theta) &= -137.1467\theta^5 + 149.4460\theta^4 - 54.1350\theta^3 + 7.5329\theta^2 + 0.5101\theta - 0.4304, \\ a_3(\theta) &= -2.3275\theta^5 + 2.6533\theta^4 - 1.2707\theta^3 + 0.3556\theta^2 - 0.0306 \times 10^{-4}\theta + 0.0016. \end{aligned} \tag{3.9}$$

Substituting the coefficients of equations (3.9) into equation (3.8), the static torque characteristics for phase #1 computed from the analytical model can be obtained. The quality of the curve fitting is illustrated in Fig. 3.2 and Fig. 3.3 for different phase currents. The mean square error between the estimated torque and the actual torque is 0.6289. It can be observed that the matching between the measured data and estimated data by this way is very bad.

Replacing the ANN flux-linkage model obtained in Chapter 2 into (3.4) and (3.5), the torque characteristics for one phase can be derived and is illustrated in Fig. 3.4. It can be observed that the torque estimate using this method is not accurate.

From the above discussion, it is clear that the torque characteristics cannot

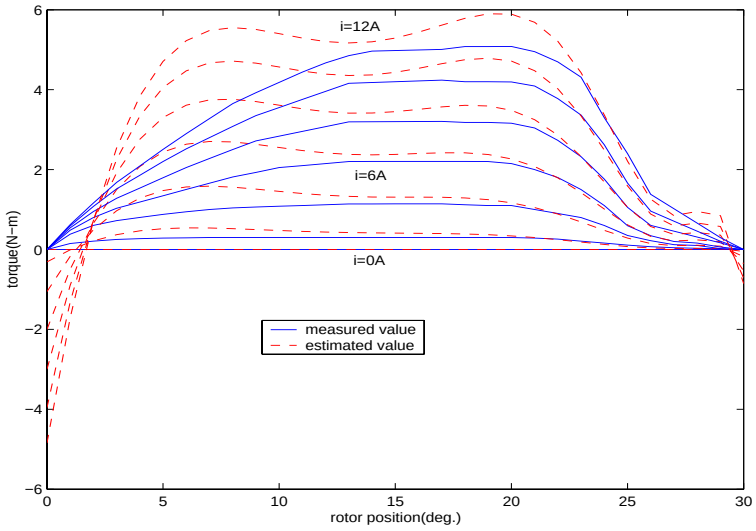

Figure 3.2: Comparison between measured and estimated torque−rotor position data for the currents of $0A$(bottom figure),$2A$, $4A$, $6A$, $8A$, $10A$, and $12A$ (top figure) for phase # 1 based on the analytical flux-linkage model. (solid line-measured value; dashed line-estimated value)

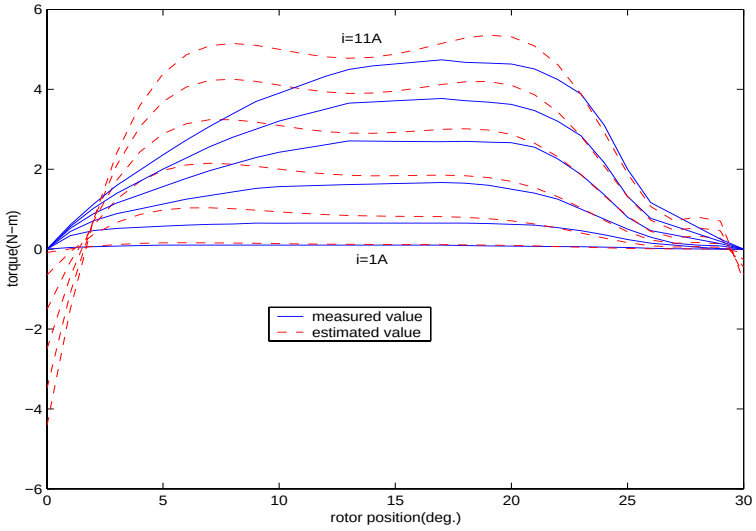

Figure 3.3: Comparison between measured and estimated torque−rotor position data for the currents of $1A$(bottom figure),$3A$, $5A$, $7A$, $9A$, and $11A$ (top figure) for phase # 1 based on the analytical flux-linkage model. (solid line-measured value; dashed line-estimated value)

be accurately reflected by replacing the flux-linkage model obtained in Chapter 2 into (3.4) and (3.5). This could be due to both high-nonlinearity and the following sources of error in the modelling process

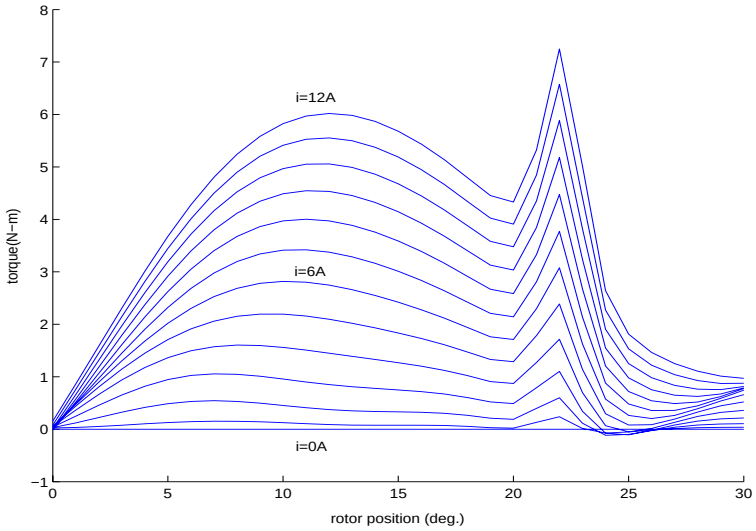

Figure 3.4: The torque characteristics derived by applying equations (3.4), (3.5) and (3.6) into the flux-linkage ANN model.

- measurement noise in voltage and current data. This results in inaccurate measured flux-linkage data and inaccurate estimated flux-linkage model is derived, and then inaccurate estimated torque model is obtained.

- error due to numerical integration. This results in inaccurate measured flux-linkage data and inaccurate estimated flux-linkage model is derived, and then inaccurate estimated torque model is obtained.

- error incurred during the curve fitting process to determine the coefficients of the model. This results in inaccurate parameters of the estimated flux-linkage model, therefore inaccurate estimated torque model is acquired.

- Moreover, there is always some error at higher currents because of the use of rising current method for the flux measurement as reported in [53]. This fact can be observed, by a close examination, from the flux-linkage characteristics at higher currents.

Mapping between measured torque data and estimated torque data is a very complicated function of machine geometry, winding configuration and magnetic characteristics of the material used for laminations. A 3-D plot of torque shown in Fig. 3.5 depicts the profile of measured torque versus rotor position and phase current for phase #1. The simplest form of commutation, in which constant currents are applied to the different phases in turn, leads to an uneven torque− rotor position characteristic. This is caused by the nonlinear torque production characteristics. For high dynamic performance tasks, the nonlinearities of the torque characteristics should be taken into account in its mathematical modelling. For our experimental SRM, various efforts undertaken for modelling the SRM torque characteristics are discussed in the following sections.

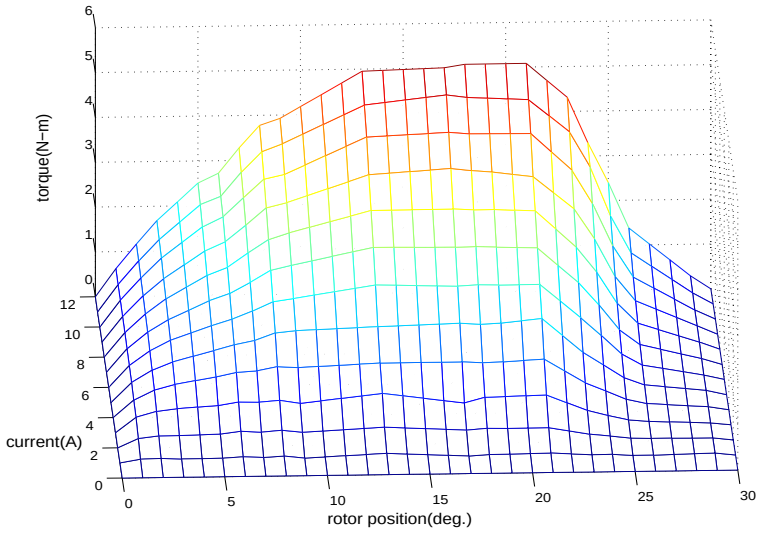

Figure 3.5: The profile of measured torque versus rotor position and phase current for phase #1.

3.3 SRM Torque Modelling Using Analytical Model Based Approaches

Since the torque model obtained from applying the flux-linkage model and the standard electric machine theory fails to provide accurate torque prediction. we have to modelling the torque characteristics of the SRM by directly employing various intelligent algorithms. Because the task of GA is to find a neighborhood around global minimum with precise results, and the task of LM gradient-expansion method is to find the best solution in the neighborhood, we let GA provide coarse search, and LM gradient-expansion method refine the search taking the results of the analytical torque model obtained from GA as the initial values, to achieve an accurate analytical torque model.

3.3.1 GA Based Computation

From Fig. 2.7 and Fig. 2.8 in Chapter 2, it can be found that the estimated flux-linkage data from the analytical flux-linkage model based on GA cannot match the measured flux-linkage data quite well. Thus GA is applied directly for modelling the torque characteristics, whose analytical expression is given by equation (3.8). In equation (3.8), for this study, we propose an expression for a_1, a_2, and a_3 as follows

$$a_i(\theta) = a_{i1}\theta^5 + a_{i2}\theta^4 + a_{i3}\theta^3 + a_{i4}\theta^2 + a_{i5}\theta + a_{i6} \tag{3.10}$$

It turns out that we are dealing with an optimization problem, which can be

mathematically expressed as the following

$$J = \min_{\mathbf{c}\in C} \{\frac{1}{n}\sum_{i=1}^{n}(T_{mi} - T_{ei})^2\} \tag{3.11}$$

where

$$\begin{aligned} \mathbf{c} &= \{a_{i1},\ a_{i2}, ...,\ a_{i6}\}_{i=1,2,3} \\ C &= [h_1 - 0.2|h_1|,\ h_1 + 0.2|h_1|] \cap ... \cap [h_{18} - 0.2|h_{18}|,\ h_{18} + 0.2|h_{18}|] \end{aligned}$$

n is the number of sampling; T_{mi} is the measured torque for phase #1; T_{ei} is the estimated torque from (3.8); C is a domain with estimated intervals in which the lower bounds are 20% smaller than $\{h_q\}_{q=1,2,...,18}$ and the upper bounds are 20% larger than $\{h_q\}_{q=1,2,...,18}$ for the first generation; and

$$\begin{aligned} \{h_q\}_{q=1,2,...,18} = \quad & \{-51.4214, 70.7537, -26.7524, 0.6118, 0.2098, 0.2472, \\ & -95.6801, 120.2026, -49.3132, 7.1177, 0.4047, -0.4132, \\ & 1.0167, -1.7023, 0.7900, -0.0506, -0.0067, 0.0009\ \} \end{aligned} \tag{3.12}$$

which are the given values according to the experiences. Thus we have the following lower bound vlb and upper bound vub for the first generation:

$$\begin{aligned} vlb = \quad & \{-61.7057\ 56.6030\ -32.1029\ 0.4894\ 0.1678\ 0.2194 \\ & -114.8161\ 96.1621\ -59.1758\ 5.6942\ 0.3238\ -0.4958 \\ & 0.8134\ -2.0428\ 0.632\ -0.0607\ -0.008\ 0.0007\ \} \\ vub = \quad & \{-41.1371\ 84.9044\ -21.4019\ 0.7342\ 0.2518\ 0.3290 \end{aligned}$$

$$-76.5441\ 144.2431\ -39.4506\ 8.5412\ 0.4856\ -0.3306$$

$$1.2200\ -1.3618\ 0.948\ -0.0405\ -0.005\ 0.0011\ \} \tag{3.13}$$

The selection of GA population size and generation number is a trade-off between precision and computation time. The population size is chosen to be 300 and the generation size is chosen to be 3000. The results with different population number and generation number will be shown in the following for comparison.

Because our objective is to minimize the cost of the mean square error between the measured torque data and the estimated torque data, we define the fitness function F as the following:

$$F(\vec{X_i}) = J^{-1} \tag{3.14}$$

where $\vec{X_i}$ represents the ith individual in the population. The optimization task is to search the individual, which maximizes the fitness function $F(\vec{X_i})$.

With the population size of 300 and the generation size of 3000, using the same algorithm and the same modification of GA as those in Chapter 2 to search for the solutions of the eighteen parameters in their respective searching spaces. Fig. 3.6 shows the fitness value against the number of generations. Obviously the fitness value converges when it goes to the 3000th generation. Table. 3.1 shows the simulation results over 30 runs, where J_{min} is the minimum value of J, J_{mean} is the average value of J, StdDev is the standard deviation of J, and the eighteen parameters $\{a_{i1},\ a_{i2},\ ...,\ a_{i6}\}_{i=1,\ 2,\ 3}$, are for the best solutions over 30 runs. Note that with the population size of 100 and the generation size of 3000, the fitness

Table 3.1: The simulation results over 30 runs for torque modelling

	$i=1$	$i=2$	$i=3$	J_{min}	J_{mean}	$StdDev$
a_{i1}	-42.0027	-103.6943	0.9013			
a_{i2}	60.1662	130.3262	-1.8593			
a_{i3}	-23.6667	-56.5336	0.9380	0.05158	0.05221	8.2559×10^{-4}
a_{i4}	0.6482	7.4815	-0.0596			
a_{i5}	0.1707	0.3622	-0.0066			
a_{i6}	0.3006	-0.4113	8.4213×10^{-4}			

value acquired is 0.1639; with the population size of 200 and the generation size of 3000, the fitness value obtained is 0.0941; with the population size of 400 and the generation size of 3000, the fitness value obtained is 0.05158. Therefore the population size of 300 and the generation size of 3000 are reasonable for torque modelling. The quality of the curve fitting based on GA is illustrated in Fig. 3.7 and Fig. 3.8 for different currents.

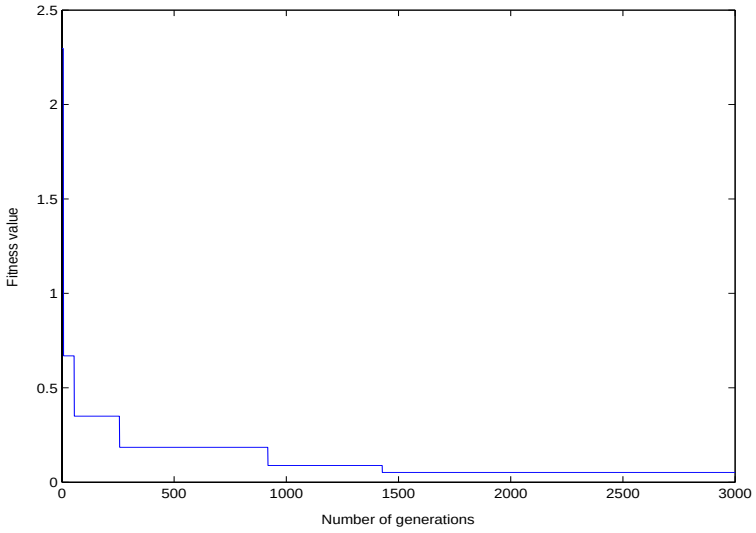

Figure 3.6: The fitness value vs the number of generations using the modified GA ($J = 0.05158$)

It can be observed that the quality of torque curve fitting based on GA is

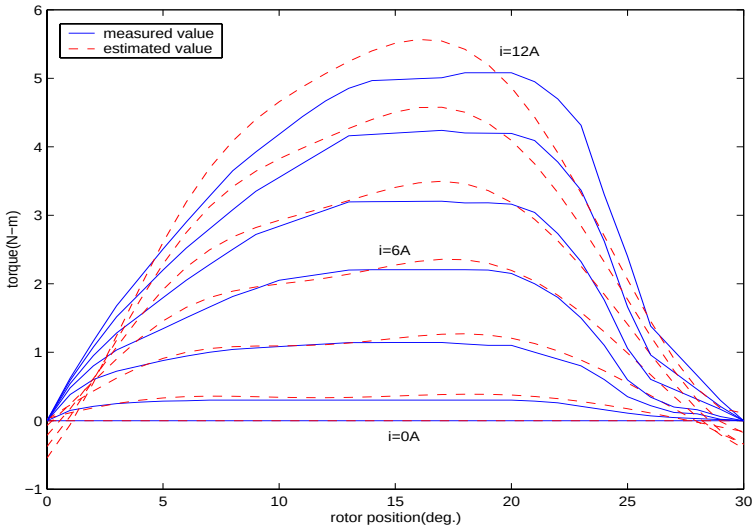

Figure 3.7: Comparison between measured and estimated torque−rotor position data for the currents of $0A$(bottom figure),$2A$, $4A$, $6A$, $8A$, $10A$, and $12A$ (top figure) for phase # 1 based on GA, (solid line-measured value; dashed line-estimated value)

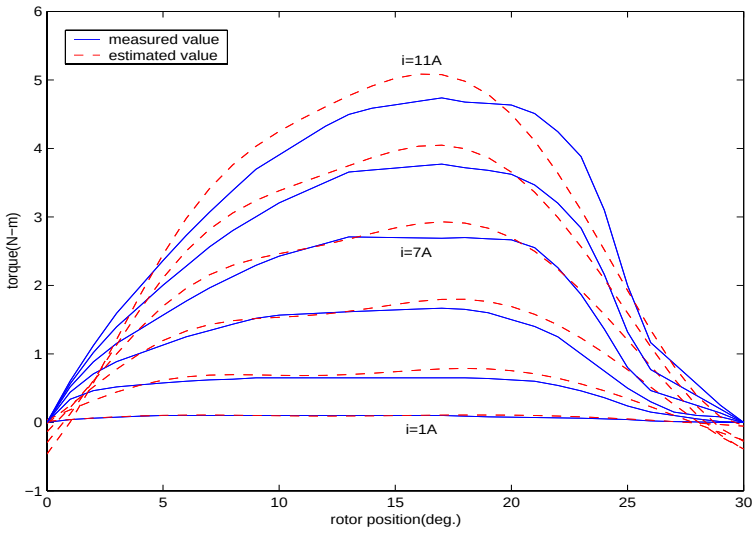

Figure 3.8: Comparison between measured and estimated torque−rotor position data for the currents of $1A$(bottom figure),$3A$, $5A$, $7A$, $9A$, and $11A$ (top figure) for phase # 1 based on GA. (solid line-measured value; dashed line-estimated value)

not very accurate although it has been improved greatly. This could be due to the following reasons: GA only provides coarse search randomly for global optimum and it does not use the gradient information over the model for finely searching which LM gradient-expansion method uses. Hence GA is not efficient for finely

results and LM gradient-expansion method is applied to refine the search in the following section.

3.3.2 LM Gradient Expansion Method Based Computation

We know that it is a very difficult task to decide the initial values of LM gradient expansion method. Many trails are needed since we have no clues for the solution. Here we take the equation (3.11) as the loss function and take the coefficients of the analytical torque model derived by GA as the initial values, to refine the search and obtain the coefficients of an accurate analytical torque model. The coefficients in the equation (3.10) for torque model are obtained as follows:

$$\begin{aligned} a_1(\theta) &= 118.4462\theta^5 - 116.1946\theta^4 + 29.0446\theta^3 - 0.9585\theta^2 + 0.2784\theta + 0.2654, \\ a_2(\theta) &= -97.8875\theta^5 + 124.1010\theta^4 - 55.6521\theta^3 + 9.6390\theta^2 + 0.4294\theta + 0.3558, \\ a_3(\theta) &= -12.9340\theta^5 + 11.8540\theta^4 - 2.6892\theta^3 + 0.0959\theta^2 - 0.0241 \times 10^{-4}\theta + 0.0016. \end{aligned} \tag{3.15}$$

The profiles for the three coefficients, i.e., a_1, a_2, and a_3, over a rotor pole pitch are shown in Fig. 3.9 for using GA method and LM gradient-expansion method. The quality of the curve fitting based on LM gradient expansion method is illustrated in Fig. 3.10 and Fig. 3.11 for different currents.

It is obvious that the quality of torque curve fitting after applying LM gradient expansion method to refine the search is much better than that based on GA only and relatively satisfactory for torque prediction. The minimum loss function

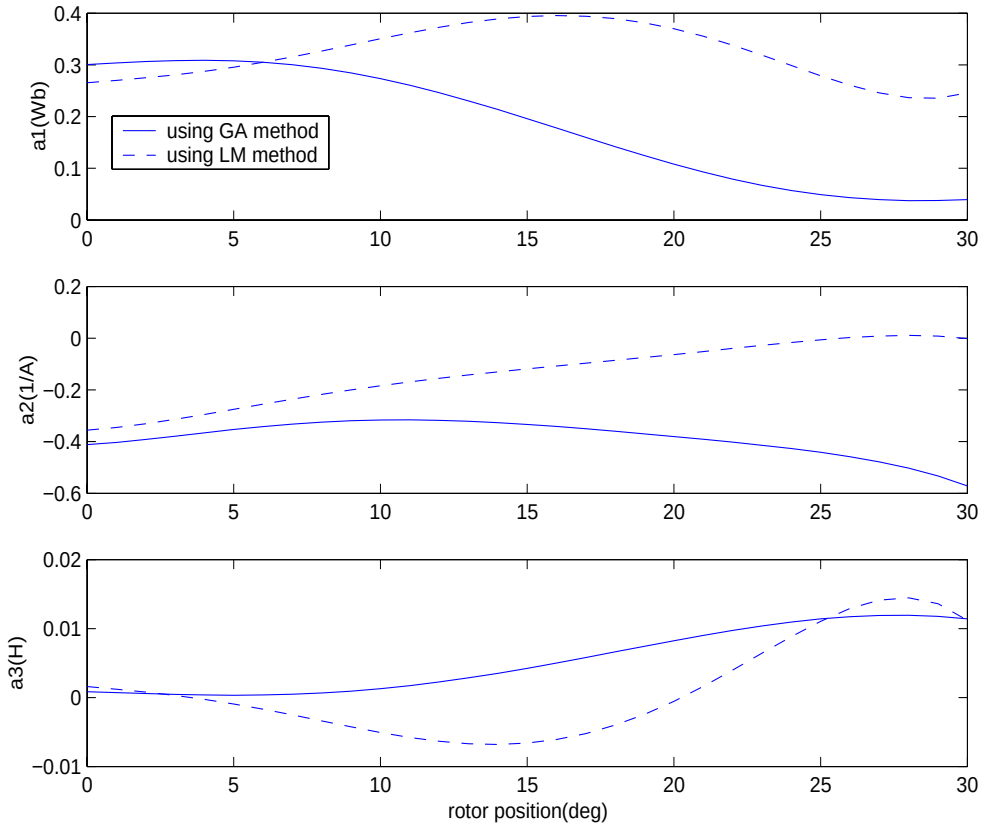

Figure 3.9: The coefficients a_1, a_2, and a_3 as functions of rotor position using GA method and LM gradient-expansion method.

value after using LM gradient expansion method is 3.9927×10^{-3}, which is reduced by 92.26% than minimum value of J obtained by GA. However, this process is very time-consuming and complex since quite a number of generations and populations in GA are needed for highly nonlinear optimization problem. Moreover, the analytical model based approach has taken full advantage of the known domain knowledge and the known physical structure of the model. Since the physical torque model is highly nonlinear, it is impossible to derive the inverse torque model according to the physical forward torque model. The blackbox based approach, such as ANN model, on the other hand, is solely data driven. Hence it can be applied for torque modelling as well as inverse torque modelling and significantly reduce the modelling

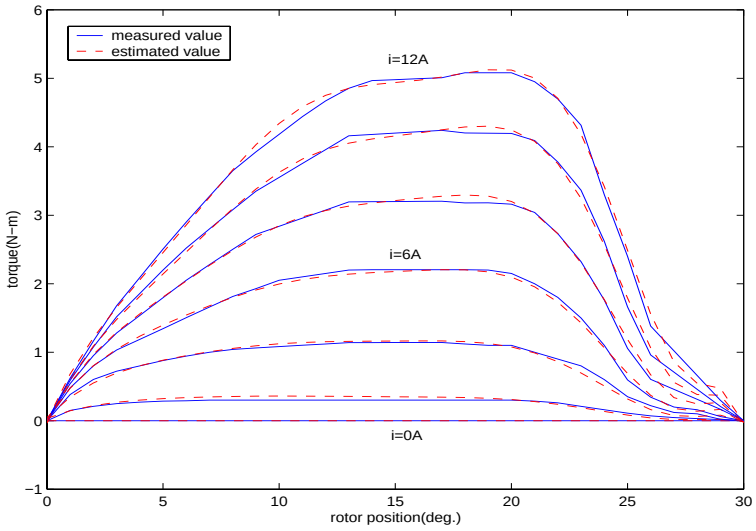

Figure 3.10: Comparison between measured and estimated torque−rotor position data for the currents of $0A$(bottom figure),$2A$, $4A$, $6A$, $8A$, $10A$, and $12A$ (top figure) for phase # 1 based on LM gradient expansion method, (solid line-measured value; dashed line-estimated value)

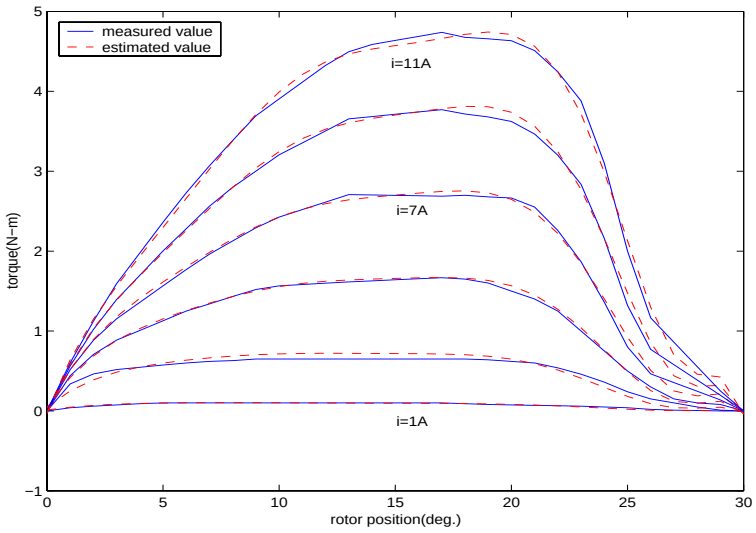

Figure 3.11: Comparison between measured and estimated torque−rotor position data for the currents of $1A$(bottom figure),$3A$, $5A$, $7A$, $9A$, and $11A$ (top figure) for phase # 1 based on LM gradient expansion method. (solid line-measured value; dashed line-estimated value)

difficulty. Now let us explore the advantages of the model-free blackbox based approach for SRM torque modelling and inverse torque modelling in the following section.

3.4 SRM Torque Modelling Using Blackbox Based Approaches

In this study, ANNs with different algorithms are directly employed for torque modelling as well as inverse torque modelling to obtain the accurate mapping between electrical and mechanical parameters of the electric machine. The training result for torque modelling based on RBF networks is shown to compare with the result for torque modelling based on BP algorithm. The SRM modelling results based on ANNs are shown as follows.

3.4.1 Modelling of SRM Torque Characteristics

The torque approximation neural net is shown in Fig. 3.12. After training, it is found that three layers network achieves the mapping task in high accuracy, including one input layer, one hidden layer and one output layer. The most suitable network configuration is $2 \times 8 \times 1$, i.e., there are two inputs in the input layer, 8 neurons in the hidden layer, and one neuron in the output layer. Fig. 3.13 shows the plot of the mean square error vs training epochs with this BP net for torque modelling. A hyperbolic tangent sigmoid transfer function - *tansig* is chosen to be the activation function of the hidden layer. The output layer is a linear summation element. The number of training epochs is 500 and the mean square error between the estimated torque and actual torque is 2.18996×10^{-3}. It is reduced by 45.15% than the minimum loss function value obtained from LM gradient expansion method. The training process takes much less time than LM gradient

expansion method. The actual motor torque and the neural net approximation are illustrated in Fig. 3.14 for different phase currents. It is obvious that the quality of curve fitting is highly satisfactory. The worst case for curve fitting is $i = 12A$, it contributes to 12.76% of the total error. For a better appreciation of the prediction of torque by the neural net model, the torque characteristics for every 0.5° interval positions from 0° to 30° for different currents are shown in Fig. 3.15. The torque characteristics for every $0.5A$ interval currents from $0A$ to $12A$ for different currents are shown in Fig. 3.16. In our neural net training, the torque data of 1° interval positions from 0° to 30° and $1A$ interval currents from $0A$ to $12A$ are adopted. Fig. 3.15 and Fig. 3.16 demonstrate that the ANN torque model has perfect interpolation results.

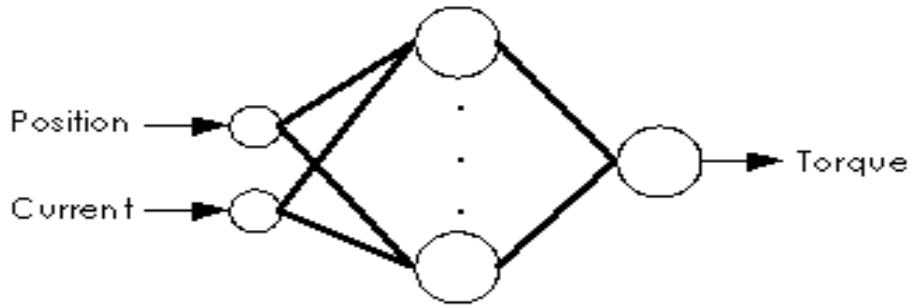

Figure 3.12: Torque approximation neural net

3.4.2 Modelling of the Inverse Torque Characteristics

Torque ripple minimization has been the biggest hurdle in promoting SRM as a better alternative to the AC and DC motors, prevalent in the industry. Researchers have proposed either a direct torque control scheme [66] or an indirect cascaded torque control scheme as in [65, 67] etc. In cascaded control schemes, whose archi-

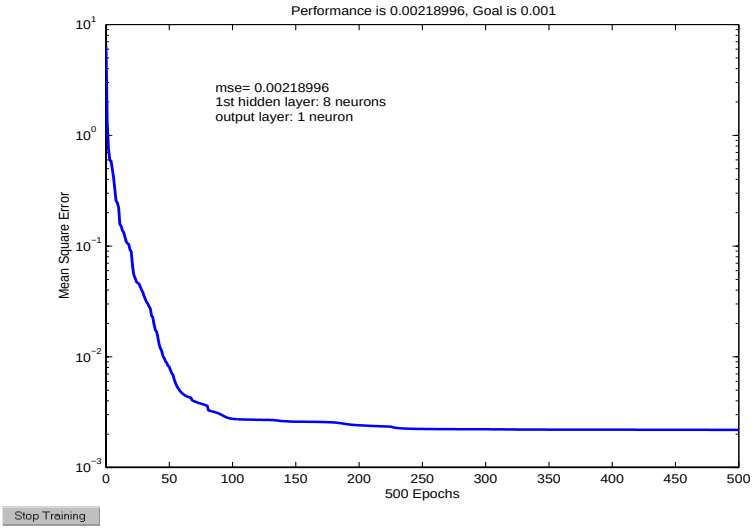

Figure 3.13: The mean square error vs training epochs with one hidden layer BP net for torque modelling

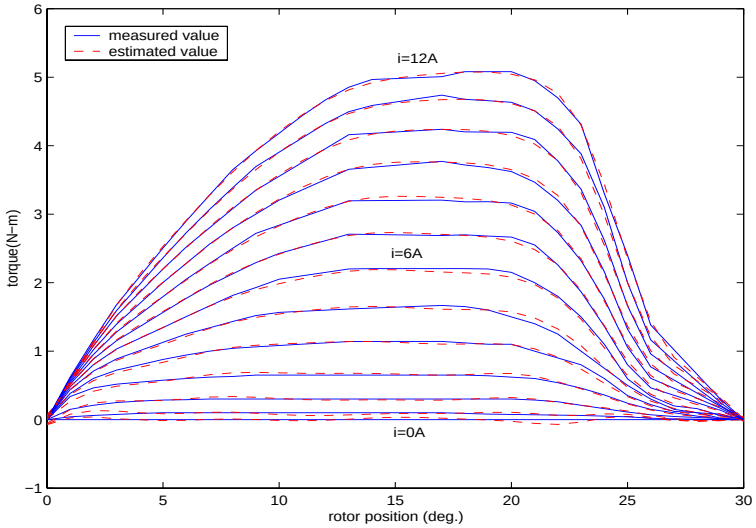

Figure 3.14: Comparison between measured and estimated torque−rotor position data for currents from $0A$(bottom figure) to $12A$ (top figure) for phase # 1 based on ANNs.(solid line-measured value; dashed line-estimated value)

tecture is shown in Fig. 3.17, one of the main tasks in this scheme is: torque to current conversion to generate accurate phase current profiles.

It is impossible so far to find an accurate analytical model for $i(T, \theta)$, i.e., current as a function of torque and rotor position. The forward torque estimation

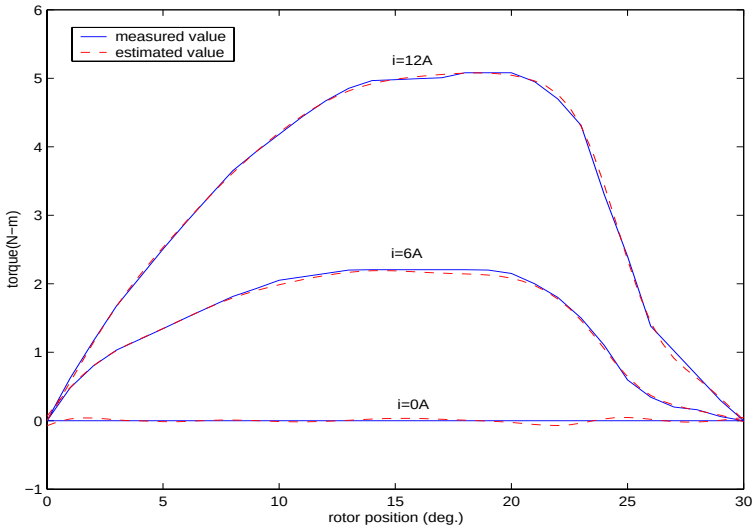

Figure 3.15: Torque comparison for every 0.5° interval positions from 0° to 30° for currents of $0A$(bottom figure), $6A$(middle figure), and $12A$(top figure).(solid line-measured value; dashed line-estimated value)

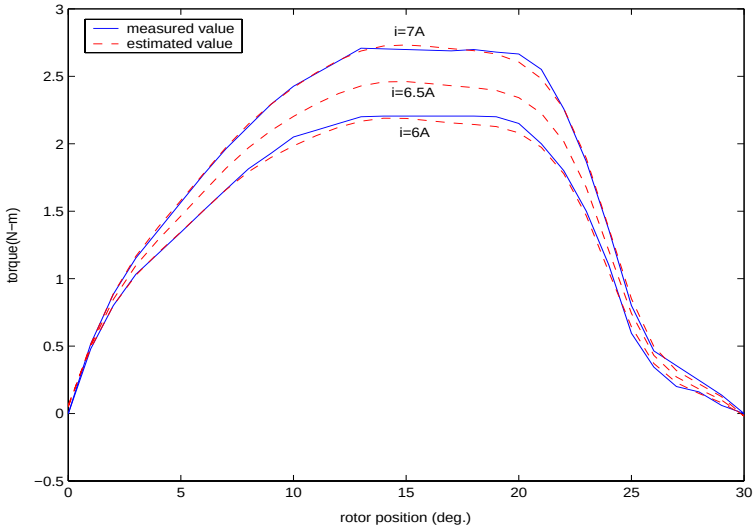

Figure 3.16: Torque comparison for every $0.5A$ interval currents from $0A$ to $12A$ for currents of $6A$(bottom figure), $6.5A$(middle figure), and $7A$(top figure).(solid line-measured value; dashed line-estimated value)

network $\hat{T}(i, \theta)$ can be inverted iteratively at every sample instant to produce the desired currents based on a specified desired torque. To implement such a scheme in real-time implementations would be time consuming and is not feasible due to sampling time restrictions. Consequently, for real-time implementations an inverse

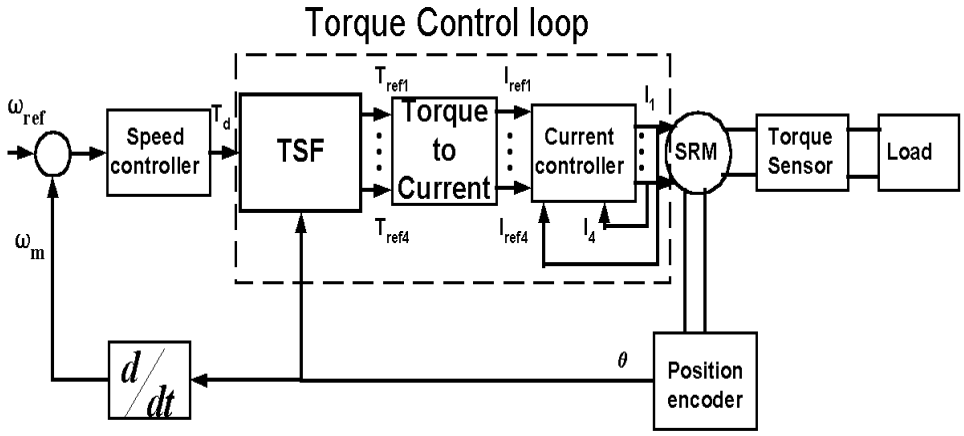

Figure 3.17: Closed loop speed control of SRM

torque neural net model is trained using the BP algorithm. The trained inverse torque model operates at each sample instant to output the desired currents based on inputs of torque reference and rotor positions. The data to train this network is obtained from the inversion of the measured torque data. The inverse torque neural net is trained to represent the inverse behavior of a single phase as dictated by the data obtained from the inversion of the measured torque data. The current approximation neural net is shown in Fig. 3.18.

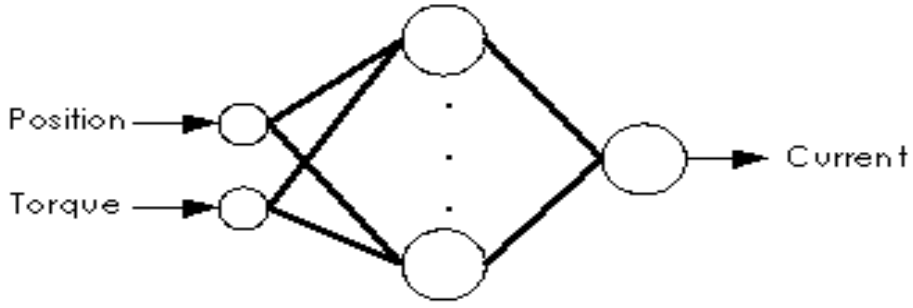

Figure 3.18: The current approximation neural net

After training, it is found that three layers network achieves the mapping task in high accuracy, including one input layer, one hidden layer and one output layer. The most suitable network configuration is $2 \times 50 \times 1$, i.e., there are two

inputs in the input layer, 50 neurons in the hidden layer, and one neuron in the output layer. Fig. 3.19 shows the plot of the mean square error vs training epochs with this BP net for inverse torque modelling. A log sigmoid transfer function - *logsig* is chosen to be the activation function of the hidden layer. The output layer is a linear summation element. The number of training epochs is 1500 and the mean square error is 0.001757. The quality of the curve fitting based on ANNs is illustrated in Fig. 3.20 and Fig. 3.21 for different rotor positions. In Fig. 3.20 and Fig. 3.21 the worst case for curve fitting is $\theta = 21°$. It contributes to 12.76% of the total error. It results from the characteristics of the torque which is shown in Fig. 3.1. It can be observed from Fig. 3.1 that the torque becomes higher with the rotor position rises below 21° and it becomes lower with the rotor position rises above 21° and therefore the torque−current variation reaches the maximum at 21°. This makes torque modelling very difficult at 21°. It is clear that the quality of curve fitting is highly satisfactory.

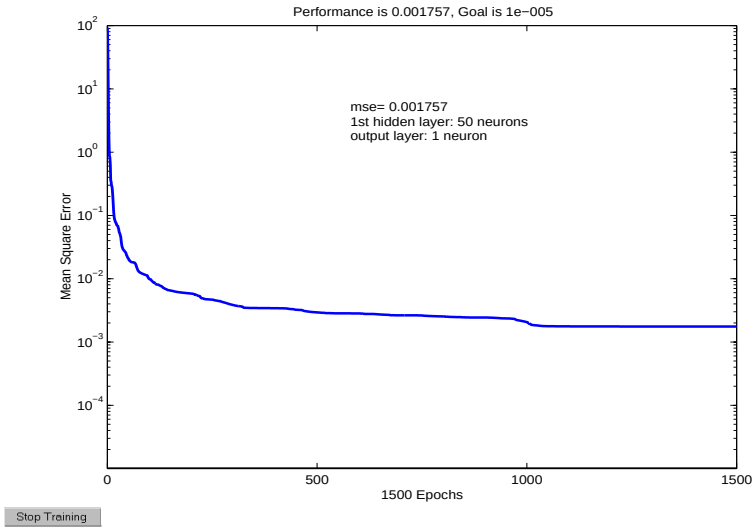

Figure 3.19: The mean square error vs training epochs with one hidden layer BP net for inverse torque modelling

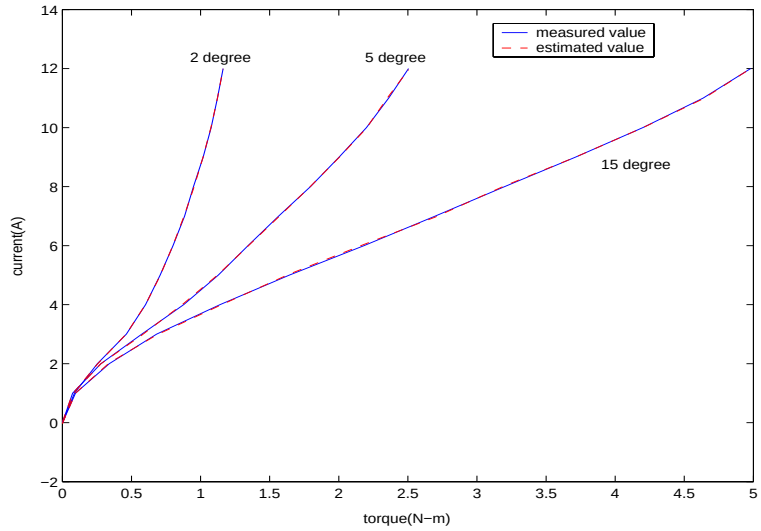

Figure 3.20: Comparison between measured and estimated current − torque data for various rotor positions of 2°, 5°, and 15° for phase # 1 based on ANNs.(solid line-measured value; dashed line-estimated value)

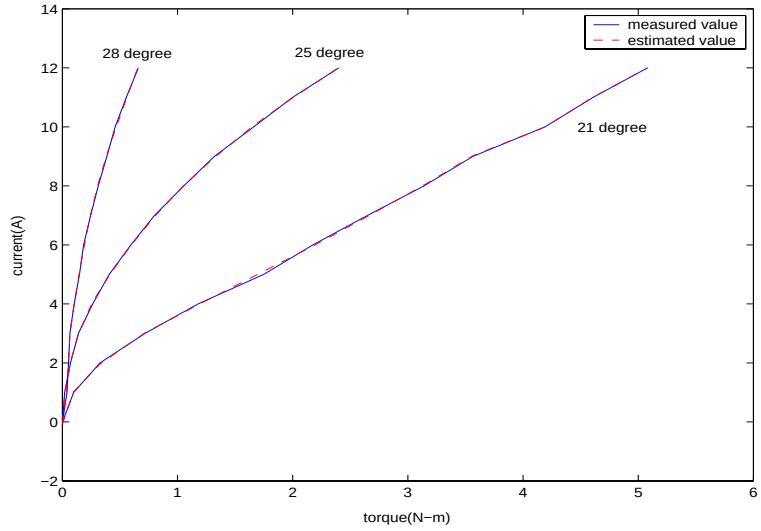

Figure 3.21: Comparison between measured and estimated current − torque data for various rotor positions of 21°, 25°, and 28° for phase # 1 based on ANNs.(solid line-measured value; dashed line-estimated value)

3.4.3 Comparative study of RBF with BP networks

For comparison, RBF networks are employed for torque modelling in this study.

The idea of RBF networks derives from the theory of function approximation. We

have already seen how BP networks with hidden layers can learn to approximate functions. RBF networks learn in a slightly different approach. Their main features are:

1) They are two-layer feedforward networks.

2) Universal approximator.

3) The hidden(kernel) nodes implement a set of *Radial Basis Functions*(e.g Gaussian functions).

4) The output nodes implement *linear* summation functions as in a BP network.

5) The network training is divided into two stages: first the weights from the input to hidden layer are determined, and then the weights from the hidden to output layer.

6) The training/learning is very fast.

7) The networks are very good at interpolation.

Many theoretical and empirical studies indicate that many properties of the interpolating function are relatively insensitive to the precise form of the basis functions. Some of the commonly used basis functions are: *Gaussian Functions, Multi-Quadric Functions, Generalized Multi-Quadric Functions, Inverse Multi-Quadric Functions, etc.* The basic mathematical expressions can be found in [68].

RBF networks and BP networks differ from each other in a number of important aspects:

1) A RBF network in its natural form has a single hidden layer, whereas BP networks can have any number of hidden layers.

2) RBF networks are usually fully connected, whereas it is common for BP networks to be only partially connected.

3) In BP networks the computation nodes (processing units) in different layers share a common neuronal model, though not necessary the same activation function. In RBF networks the hidden nodes (basis functions) operate very differently, and have a very different purpose, to the output nodes.

4) In RBF networks, the argument of each hidden unit activation function is the *distance* between the input and the "weights" (RBF centers), whereas in BP networks it is the *inner product* of the input and the weights.

5) BP networks are generally trained with a single global supervised algorithm, whereas RBF networks are usually trained one layer at a time with the first layer unsupervised.

6) BP networks construct *global* approximations to non-linear input-output mappings with *distributed* hidden representations, whereas RBF networks tend to use *localised* non-linearities (Gaussians) at the hidden layer to construct *local* approximations.

Although, for approximating non-linear input-output mappings, the RBF networks can be trained much faster, BP networks may require a smaller number of parameters. The training process with RBF networks is shown in Fig. 3.22. We let the sum square error (SSE) goal be 0.403 and the learning rate be 0.01, i.e., the MSE goal is 1×10^{-3} since we totally have 403 sets of data. It shows that the network stops training at the $275 - th$ epoch because of the low gradient variation. The training result shows that SSE becomes 1.18066 with 293 neurons in the hidden layer, i.e., the MSE becomes 2.93×10^{-3}. It is much more complicated than BP net that we have got and this result is not suitable for our experiment for its high complexity.

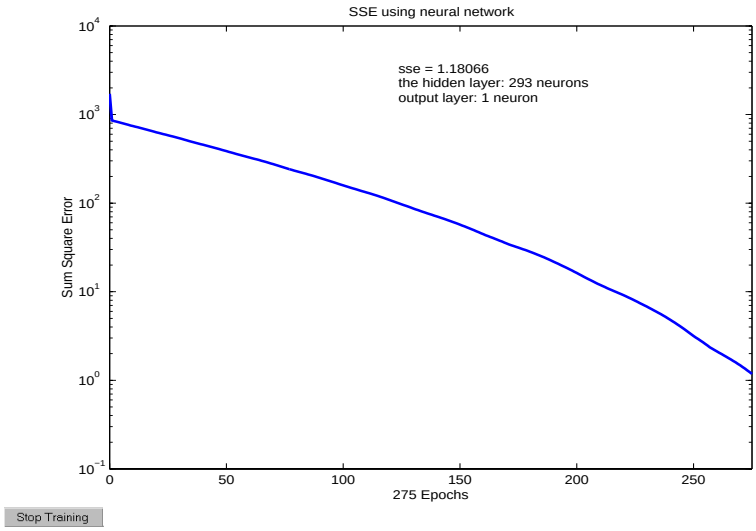

Figure 3.22: The sum square error vs training epochs with RBF networks for torque modelling

3.5 Experimental Verification

As we know, simulation results only show the static mapping. To explore the effectiveness of the model based torque estimator and ANN based torque estimator,

practical experimental work has been carried out for a 1 hp, 8/6 pole SRM to observe the mapping for dynamic conditions when the static mapping is accurate. We have implemented the experiment for both model based torque estimator and the ANN based torque estimator in the platform. Fig. 3.23 and Fig. 3.24 show the experimental results for comparison between the model based torque estimator output and the torque transducer output at the torque of $0.6Nm$ and at the torque of $1.2Nm$ respectively. Fig. 3.25 and Fig. 3.26 show the experimental results for comparison between the ANN based torque estimator output and the torque transducer output at the torque of $0.6Nm$ and at the torque of $1.2Nm$ respectively. From the experimental results, it is clear that the model based torque estimator output matches the torque transducer output perfectly and the ANN based torque estimator output matches the torque transducer output well for the average torque. However, the instantaneous ANN based torque estimator output does not match the measured torque quite well. The reasons are as below: The physical model is a kind of heuristic one, established in terms of the flux saturation phenomenon. The physical model based approach has taken full advantage of the known domain knowledge and the known physical structure of the model. It renders the modelling problem into simple state dependent parameter search problem. However, ANN based torque model is solely data driven and it can map the data exactly even when the high-frequency noises present in the measurement data. Therefore the model based torque estimator is more accurate than the ANN based torque estimator. Hence, in the experiment, the model based torque estimator output matches better the torque transducer output than the ANN based torque estimator output.

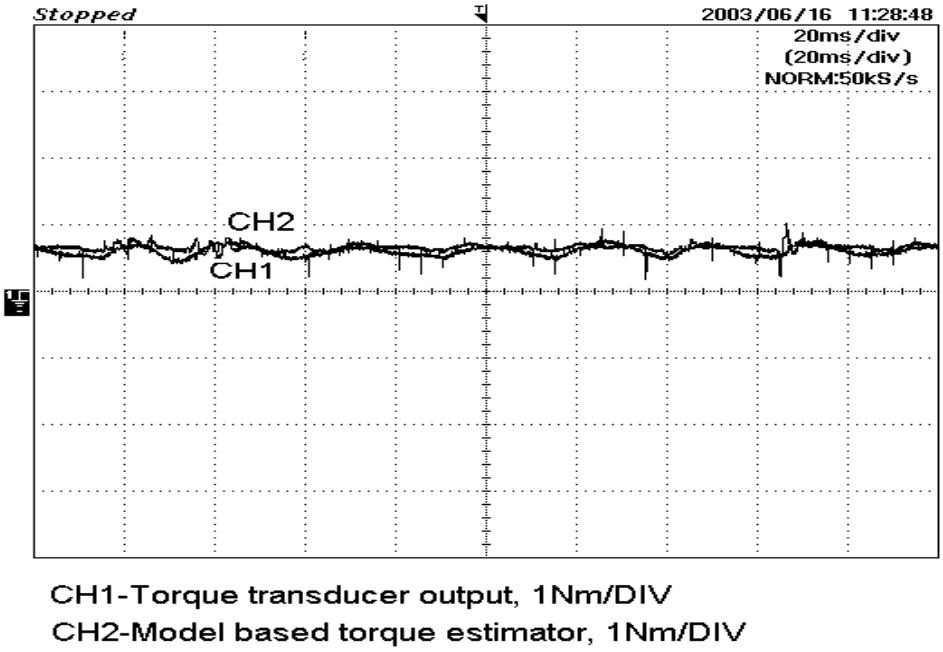

Figure 3.23: Experimental result for comparison between the model based torque estimator output and the torque transducer output at the torque of $0.6Nm$

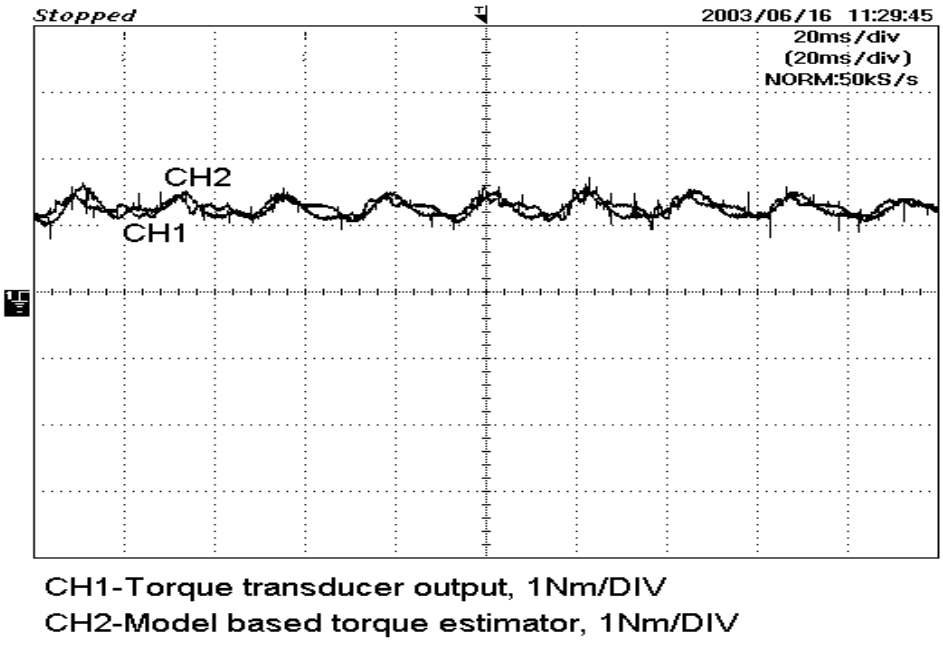

Figure 3.24: Experimental result for comparison between the model based torque estimator output and the torque transducer output at the torque of $1.2Nm$

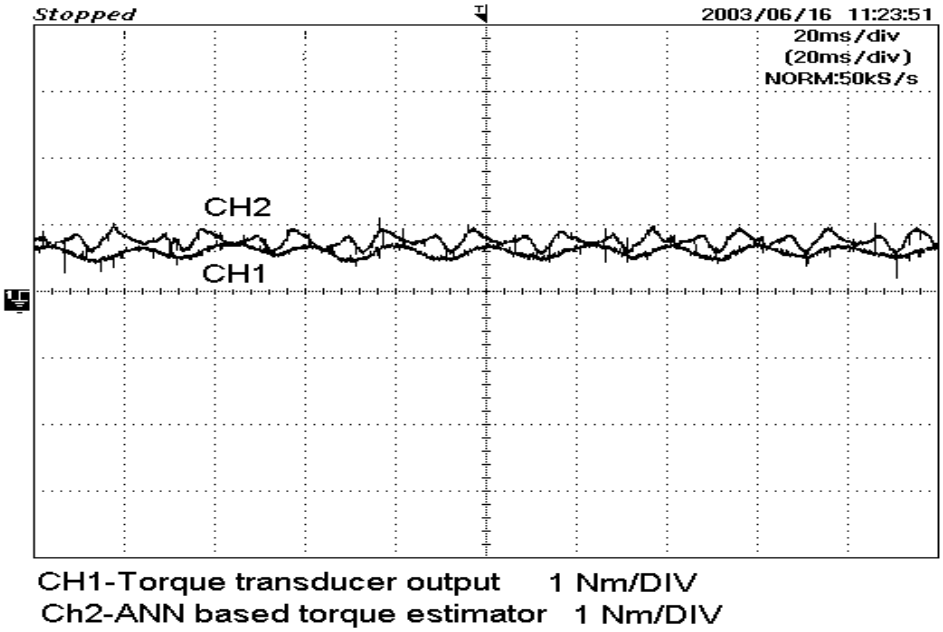

Figure 3.25: Experimental result for comparison between the ANN based torque estimator output and the torque transducer output at the torque of $0.6Nm$

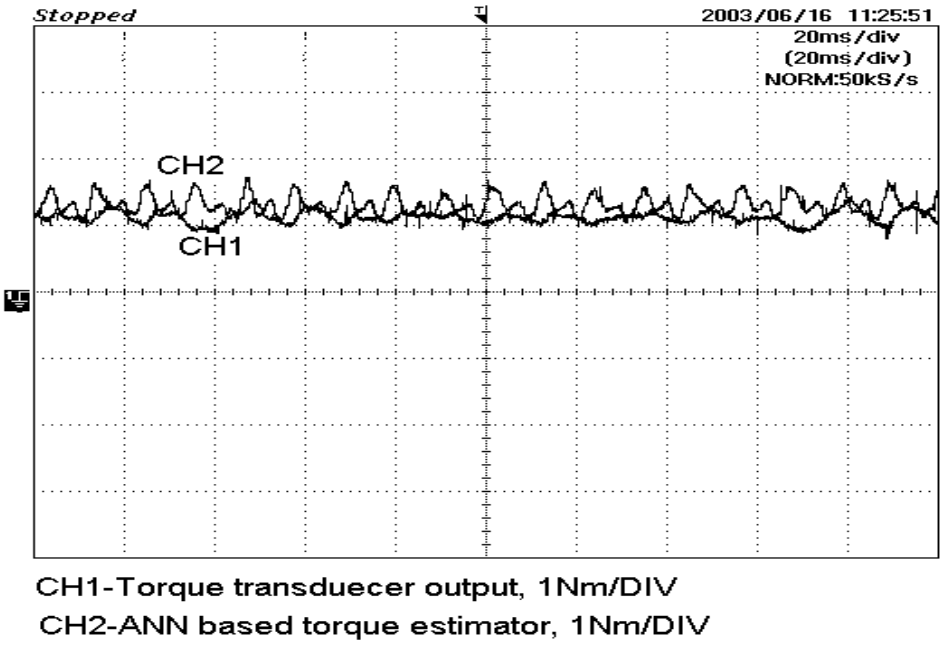

Figure 3.26: Experimental result for comparison between the ANN based torque estimator output and the torque transducer output at the torque of $1.2Nm$

3.6 Conclusion

This chapter has been concerned with the highly nonlinear torque modelling and inverse torque modelling of the SRM. It begins with applying the flux-linkage model into the standard electric machine theory for torque prediction. However, it is not accurate because of the possible unavoidable measured error. Then both analytical model based approaches and blackbox based approaches for torque modelling and inverse torque modelling of the SRM are adopted. For analytical model based approaches, we let GA provide coarse search, and LM gradient-expansion method refine the search taking the results of the analytical torque model obtained from GA as the initial values, to achieve an accurate analytical torque model. The quality of curve fitting for torque is relatively satisfactory after combining GA and LM gradient-expansion method. Considering the long process for identifying the torque model and the impossibility of modelling the inverse torque characteristics by analytical model based approaches, the model-free blackbox based approaches are employed and ANNs with different algorithms are applied for torque modelling and inverse torque modelling. Simulation results show that ANNs can achieve the mapping task in high accuracy and fast with BP algorithm. For comparison, RBF networks are used for torque modelling. The training result shows that RBF networks are not suitable for this study. Last, experimental results verify that the model based torque estimator output matches the torque transducer output perfectly and the ANN based torque estimator output matches the torque transducer output well for the average torque. Simulation and experimental results verify the

effectiveness of the derived torque models for achieving high accuracy and their respective advantages.

Chapter 4

GA Based Optimization of Current Waveforms for Torque Ripple Minimization in the SRM

4.1 Introduction

Due to the saturation effect and the variation of magnetic reluctance, all pertinent characteristics of the machine model are highly nonlinear functions of both rotor position and phase current. The outcome of all these nonlinearities is that torque ripple is incurred when the motor is excited by the conventional rectangular pulse excitation scheme.

A cascaded control structure shown in Fig. 3.17 is often adopted for its obvious advantages of easy design and inherent over current protection etc. This approach consists of a suitable TSF, accurate conversion of each phase torque reference to phase current reference and an effective current tracking by the current controller. The smoother the current waveforms, the better will be the SRM per-

formance. Taking the current as input to the SRM, the corresponding output is the torque that can be expressed as a nonlinear function of the rotor position and current. It is well known that single-phase excitation requires higher peak current demand from each phase and leads to significant torque ripples. However, this problem can be reduced to some extent by using two-phase excitations. The problem can be formulated as to find the appropriate torque references of two adjacent operating phases, which constitute a constant desired output torque, and meanwhile minimize the input current variations.

The TSF approach is proposed to address this problem in this research. Two TSFs are used to specify the reference torque of two adjacent operating phases. By adjusting the TSF, one can obtain different current waveforms. Unfortunately, it is still a trial and error manner in the existing TSF design methods. In this study, based on the TSF, we introduce a new fitness function aiming at reducing the current variations. In order to have a much higher probability of efficiently finding the optimal setting of design parameters, GA [52] is used to provide the multivariable optimization in the searching process and achieve the global minimum. First only a single phase is considered, i.e. single objective optimization. This piece of work is published in [69]. Then the aim to reduce the current variations of any two adjacent phases is formulated into a multiobjective optimization problem. This piece of work is published in [70].

This chapter is organized as follows. the SRM torque model, which is derived from a differnt flux-linkage model from Chapter 2 and Chapter 3, is given in

Section 4.2. Section 4.3 shows the single phase optimization. Section 4.4 discusses multiobjective optimization. Finally Section 4.5 concludes this chapter.

In this Chapter, we use a different SRM from Chapter 2 and Chapter 3 for simulation. A four-phase, 7.5kW SRM is used in the study of this chapter.

4.2 SRM Torque Model

The switched reluctance motor has a salient pole stator with concentrated windings and also a salient pole rotor with no magnets or coils. The basic principles of inductance variation and torque production have been described in detail in Chapter 1. In our investigation, a four-phase SRM is used. All the four phases have the same profile. Throughout this chapter, reference to a generic phase j is indicated by a subscript j in the variables. Because of the double saliency of the motor and the intentional operation of the motor under magnetic saturation, the inductance of the phase $j(j = 0, 1, 2, 3)$, L_j , is a function of both rotor position θ_e and the current i_j. An accurate expression of the torque can be obtained by taking a flux-linkage model that incorporates the effect of magnetic saturation. Because the work in this chapter has been done earlier than the work in Chapter 2 and Chapter 3, a simpler flux-linkage model which has been proposed in [25] is used in this study. The flux-linkage model in this chapter is denoted by $\phi_j(\theta_e, i_j)$ to tell it from the flux-linkage model in Chapter 2 and Chapter 3. This nonlinear

flux-linkage model is expressed by

$$\phi_j(\theta_e, i_j) = \phi_s\{1 - exp(-i_j f_j(\theta_e))\} \quad i_j \geq 0 \tag{4.1}$$

where ϕ_s is the saturated flux-linkage, θ_e is the electrical angle of the rotor position, and i_j is the current. $f_j(\theta_e)$ is expressed as a strictly positive Fourier series expansion. For the degree of accuracy for the present investigation, we include only the first two terms, as suggested in [25], of the Fourier series. Here, we have evaluated $f_j(\theta_e)$ by a sinusoidal approximation [43] of the induction produce, i.e.

$$f_j(\theta_e) = \frac{L_a + L_u}{2} + \frac{L_a - L_u}{2}\sin(\theta_e - j\frac{\pi}{2}) \tag{4.2}$$

where L_a and L_u are the phase inductances at aligned and unaligned positions respectively. Comparing the flux-linkage models (2.3) in Chapter 2 and (4.1) in this chapter, it is easy to find:

(i) ϕ_s in (4.1) plays a role as $a_1(\theta)$ in (2.3), but we have taken it as a constant in the study of this chapter;

(ii) $f_j(\theta_e)$ in (4.1) plays a role as $a_2(\theta)$ in (2.3), but we have evaluated $f_j(\theta_e)$ by a sinusoidal approximation of the induction produce in the study of this chapter;

(iii) $a_3(\theta)$ in (2.3) is ignored in the study of this chapter.

With the flux-linkage model (4.1), the torque produced by phase j can be obtained as [71]

$$T_j = f(\theta_e, i_j) = \frac{N_r\phi_s}{f_j(\theta_e)^2}\frac{df_j(\theta_e)}{d\theta_e}\{1 - [1 + i_j f_j(\theta_e)] \times \exp(-i_j f_j(\theta_e))\} \tag{4.3}$$

In a two-phase-on scheme, two adjacent operating phases will jointly produce a desired torque T_d, which is a constant in this work

$$T_d = T_{d,j} + T_{d,j+1} \tag{4.4}$$

Note that each phase current and torque are periodic over $[(j-1)\frac{\pi}{2}, (j+1)\frac{\pi}{2}]$.

4.3 Single Phase Optimization

4.3.1 Torque Sharing Function

TSFs are used to distribute the motor torques produced by each phase so that total torque is equal to the desired torque. We can find that the TSF directly influence the current waveform. To improve the current waveform, we have to change and improve the TSF. The objective behind the assignment of the desired torque to the individual phases is to co-ordinate the production of torque by the individual phases so that the total torque at any instant is equal to the desired torque. A TSF is defined for a phase only in the region where it can produce positive torque (assuming motoring operation). For the rest of the electrical cycle, the TSF for that phase is zero. There are a number of ways one can define these TSFs. An arbitrary choice of the TSF will not be suitable for our purpose. Our approach will be obviously for a smoother function. The simplest trajectory that can be used for this purpose is the cubic segment (cubic spline). The cubic segment, here, is a cubic polynomial of θ_e . This function is characterized by four arbitrary parameters. With the two-phase-on scheme, each TSF will have a rising segment followed by a

constant segment and a falling segment.

Since all four phases are identical except for a phase shift of $\frac{\pi}{2}$, only one phase TSF is presented below. The torque trajectory for the rising segment is

$$T_{d,j} = c_1 + c_2\theta_e + c_3{\theta_e}^2 + c_4{\theta_e}^3 \tag{4.5}$$

It starts from a zero value and ends at the full value T_d in an interval $\theta_e \in [(j-1)\frac{\pi}{2},\ (j-1)\frac{\pi}{2}+l]$ where $l \in [0,\ \frac{\pi}{2}]$. During the constant segment we have

$$T_{d,j} = T_d \quad \theta_e \in [(j-1)\frac{\pi}{2}+l,\ (j+1)\frac{\pi}{2}-l] \tag{4.6}$$

In the falling segment the desired torque trajectory starts from the full value T_d and ends with zero value. It is complementary to the rising segment. i.e., obtained by subtracting the rising segment from the constant torque T_d. This is because all four phases must generate identical torque, consequently the same current waveforms, and the falling segment is exactly overlapped with the rising segment of the next phase. The falling segment thus has the same length as rising segment. The interval of the falling segment is $\theta_e \in [(j+1)\frac{\pi}{2}-l,\ (j+1)\frac{\pi}{2}]$.

The four parameters of the cubic segment are uniquely determined by four conditions: either zero or full values for the initial and end torque, and zero derivatives $\frac{dT_{d,j}}{d\theta_e} = 0$ at both the points. Fig. 4.1 shows a set of TSF for a desired torque of $5N-m$. Fig. 4.2 shows the current profiles for two adjacent operating phases for a desired torque of $5N-m$ when TSF shown in Fig. 4.1 is employed.

From equation (4.3), we can see there are two singular points at $(j-1)\frac{\pi}{2}$ and $(j+1)\frac{\pi}{2}$ for each phase, because $\frac{df_j}{d\theta_e} = k\cos(\theta_e - j\frac{\pi}{2})$, where $k = \frac{L_a - L_u}{2}$. The

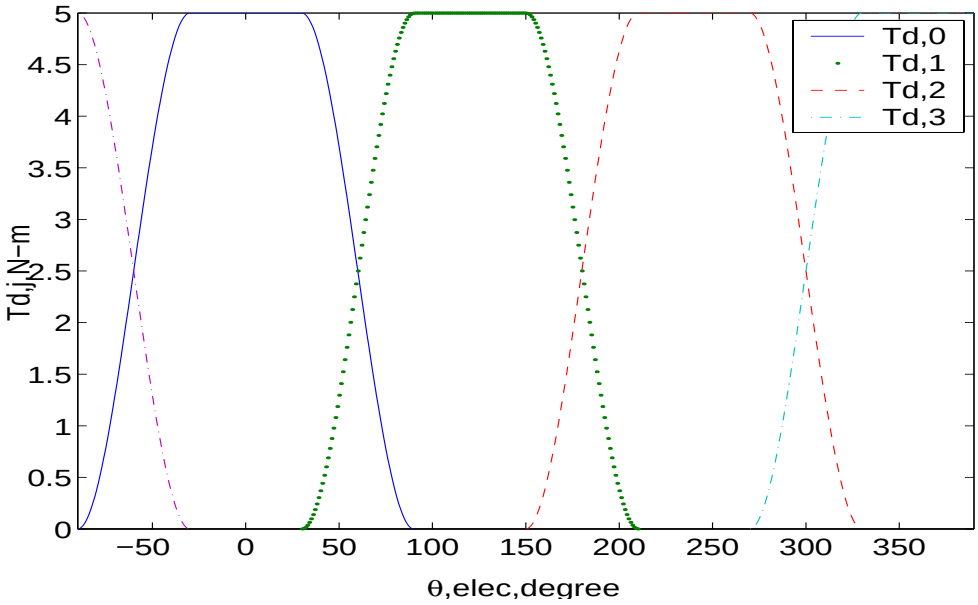

Figure 4.1: Examples of torque sharing function with cubic segments.(solid line-$T_{d,0}$; dotted line-$T_{d,1}$; dashed line-$T_{d,2}$; dash-dotted line-$T_{d,3}$.)

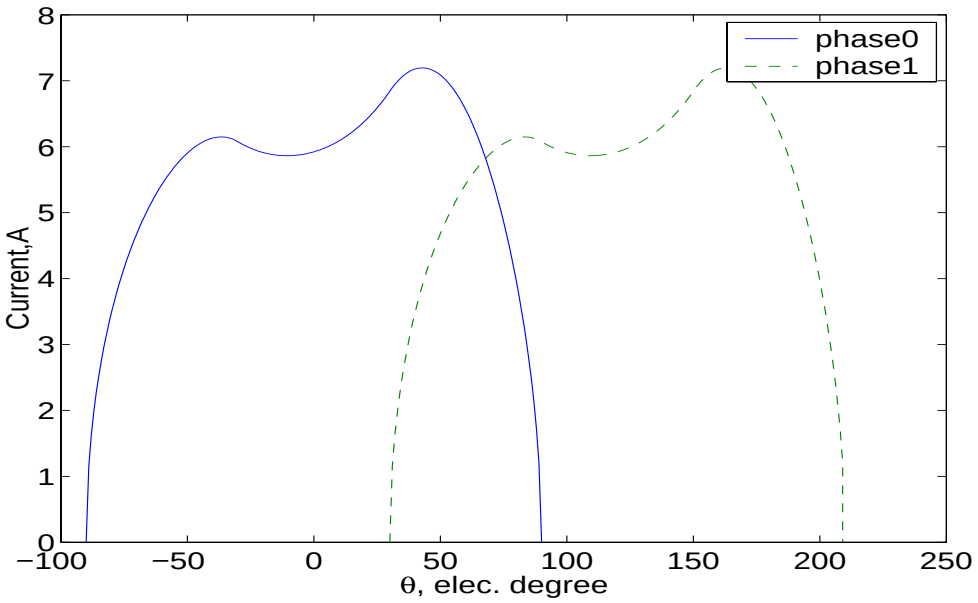

Figure 4.2: The current profiles for two consecutive phases for desired torque of $5N - m$. (solid line-phase0; dashed line-phase1)

numerical calculation of current $i_{d,j}$ is impossible at these two points. To overcome this difficulty, two null segments are introduced in each phase. Fig. 4.3 shows an example of TSF for a desired torque of 5N-m in one phase when the null segments

are introduced. We define the length of one null segment to be ϵ. From this figure, we can observe that, during a null segment in the phase j, phase $j-1$ or phase $j+1$ should have correspondingly a full torque segment to meet the total torque requirement. The length of the full torque segment is 2ϵ. From Fig. 4.3, it is easy to derive the relationship between l and ϵ : $l + 2\epsilon = \frac{\pi}{2}, l$ is the length of the rising segment. Therefore each phase should provide torque over a period $\theta_e \in [(j-1)\frac{\pi}{2}+\epsilon,\ (j+1)\frac{\pi}{2}-\epsilon]$. In Fig. 4.3, $\epsilon = \frac{\pi}{12}$. Fig. 4.4 shows the corresponding current profiles when TSF shown in Fig. 4.3 is employed.

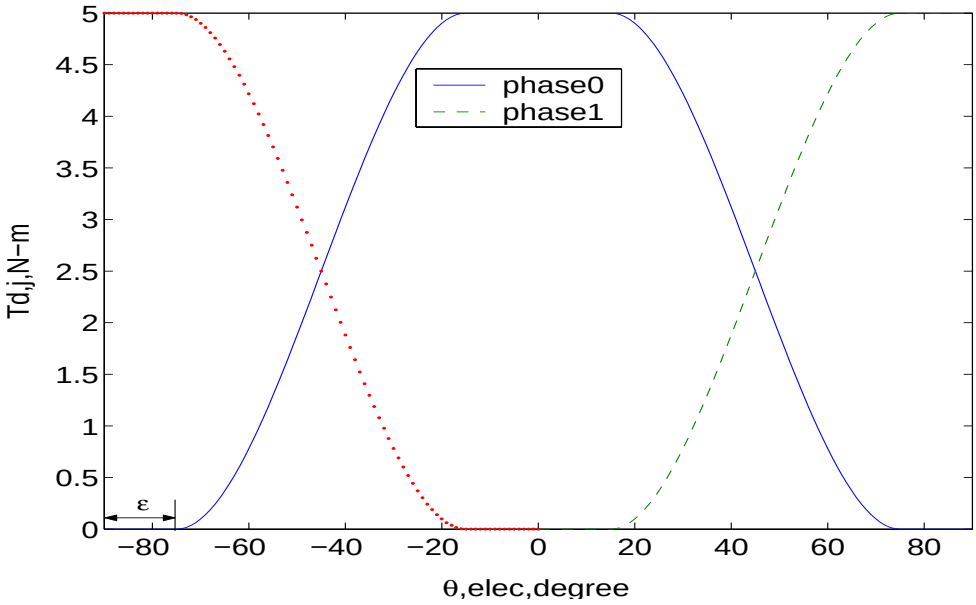

Figure 4.3: An example of TSF for a desired torque of $5N-m$. (solid line-phase0; dashed line- phase1)

4.3.2 New Torque Sharing Function Design

From Fig. 4.1 and Fig. 4.2, we can make the following observations.

1) With TSF, the currents are smooth in general, but with rather different cur-

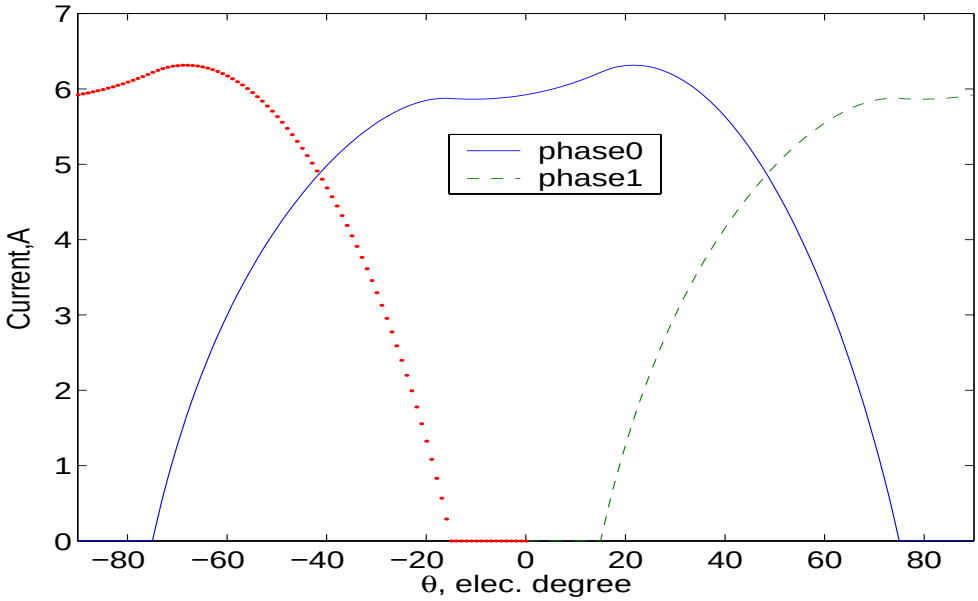

Figure 4.4: The current profiles for a desired torque of $5N-m$. (solid line-phase0; dashed line- phase1)

rent variation rate. $\left|\frac{\partial i_j}{\partial \theta_e}\right|$.

2) Though a symmetric torque profile is provided, the corresponding current i_j waveform is not symmetric.

3) Near the starting point of the rising segment i_j is much stiffer, and near the center area i_j is very flat.

Based on the first observation, it is natural to choose $\left|\frac{\partial i_j}{\partial \theta_e}\right|$ as the cost function to be minimized. It is easy to derive

$$\left|\frac{\partial i_j}{\partial \theta_e}\right| = \frac{\left|\frac{dT_{d,j}(\theta_e)}{d\theta_e} - \frac{\partial T_j(\theta_e, i_j)}{\partial \theta_e}\right|}{\left|\frac{\partial T_j}{\partial i_j}\right|} \tag{4.7}$$

From equation (4.7) it is clear that the current variation is directly proportional to the torque variation $\frac{dT_{d,j}}{d\theta_e}$, but may not be monotonic in the entire operating period due to other two terms in equation (4.7). From observations 2 and 3 and equation (4.7), it is clear that the reference torque variation should be lowered at the beginning of the rising segment, such that a less current variation is desired. On the contrary, a higher torque variation profile can be considered at the later half of the rising segment, as the current response tends to be flat. To serve this purpose, a new torque compensation curve shown in Fig. 4.5 is added to the original TSF shown in Fig. 4.1.

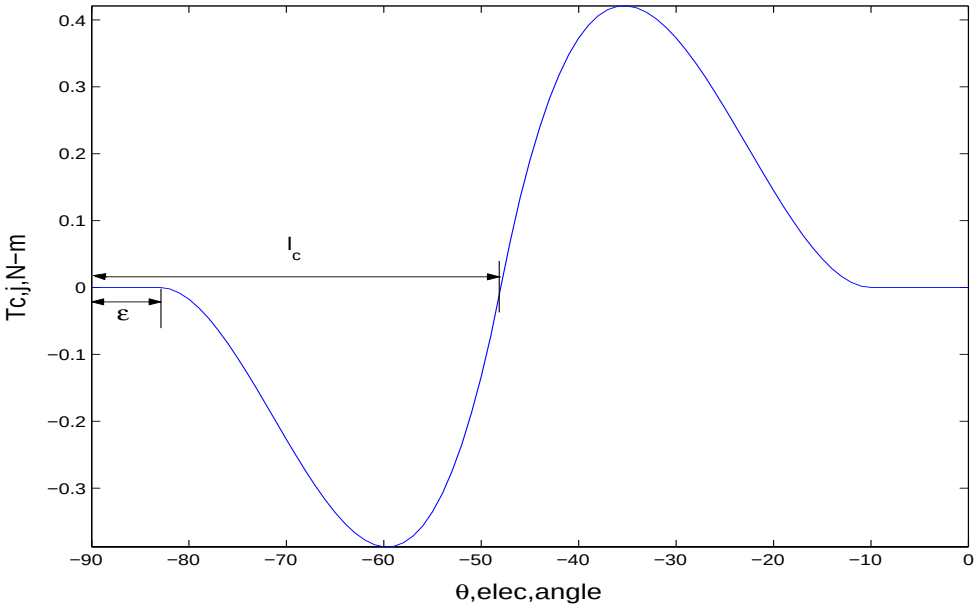

Figure 4.5: Example of the compensating torque $T_{c,j}$

$T_{c,j}$ is to reshape the original $T_{d,j}$ for the rising segment only, thus it spans $[(j-1)\frac{\pi}{2}+\epsilon,\ j\frac{\pi}{2}-\epsilon]$. It actually consists of two cubic splines over two intervals $[(j-1)\frac{\pi}{2}+\epsilon,\ (j-1)\frac{\pi}{2}+l_c]$ and $[(j-1)\frac{\pi}{2}+l_c,\ j\frac{\pi}{2}-\epsilon]$ respectively, where l_c is the distance between the point of $(j-1)\frac{\pi}{2}$ and the joint point of two compensating

splines. The objective of the first cubic spline is to reduce the torque rising rate in the first portion of the rising segment and the second cubic spline is to increase torque rising rate in the second portion of the rising segment. Note that these two cubic splines together have eight design parameters to be determined from boundary conditions, which are:

$$T_{c,j}(\theta_e) = 0 \tag{4.8}$$

$$\frac{dT_{c,j}}{d\theta_e} = 0 \tag{4.9}$$

at $\theta_e = (j-1)\frac{\pi}{2} + \epsilon$ for the first spline;

$$T_{c,j}(\theta_e) = 0 \tag{4.10}$$

$$\frac{dT_{c,j}}{d\theta_e} = 0 \tag{4.11}$$

at $\theta_e = j\frac{\pi}{2} - \epsilon$ for the second spline;

$$T_{c,j}(\theta_e) = 0 \tag{4.12}$$

$$\frac{dT_{c,j}}{d\theta_e} = \delta \tag{4.13}$$

at $\theta_e = (j-1)\frac{\pi}{2} + l_c$ for both splines. Here δ is an arbitrary positive constant.

So far three design parameters are introduced and free for us to choose. They are:

1) The null segment length ϵ .

2) The distance between the point of $(j-1)\frac{\pi}{2}$ and the joint point of two compensating splines l_c.

3) The derivative δ at the joint point of two compensating splines.

By altering these three parameters, the shape of reference torque $T_{d,j}$ varies; accordingly we can obtain various current waveforms. It turns out that we are dealing with an optimization problem, which can be mathematically expressed as the following

$$J = \min_{\mathbf{c}\in C} \max_{\theta_e \in \Theta} \left| \frac{\partial i_j}{\partial \theta_e} \right| \tag{4.14}$$

where

$$\begin{aligned} \mathbf{c} &= \{\epsilon, l_c, \delta\}, \\ C &= [0,\ \frac{\pi}{2}) \cap [\epsilon,\ \frac{\pi}{2} - \epsilon] \cap [0,\ \infty), \\ \Theta &= \left[(j-1)\frac{\pi}{2} + \epsilon,\ (j+1)\frac{\pi}{2} - \epsilon\right] \end{aligned}$$

The cost function J is highly nonlinear in the parametric space C. GA allows us to search the whole solution space and come up with the global optimum.

4.3.3 GA Based Computation

In this chapter, standard simple GA is applied. For the three parameters: $\epsilon,\ l_c,\ \delta$, we selected their lower and upper bound VLB and VUB as follows respectively from a practical point of view: $\epsilon = [\frac{\pi}{180},\ \frac{\pi}{6}], l_c = [\frac{7\pi}{36},\ \frac{pi}{4}], \delta = [0, 5]$. The candidate solutions of the three parameters are represented by the binary values. Each string

of the candidate solutions has the appearance shown as below:

ϵ	l_c	δ
$xx...x$	$xx...x$	$xx...x$

where the output value of each parameter is represented by a 10-bit binary number. A population size of 100 strings is chosen and the generation size of 100 is chosen.

Because our objective is to minimize the cost of the maximum derivative of the current with respect to the motor position, we define the fitness function F as the following:

$$F(\vec{X_i}) = \left|\frac{\partial i_j}{\partial \theta_e}\right|^{-1} \tag{4.15}$$

where $\vec{X_i}$ represents the ith string in the population. The larger the fitness value is, the better the string fits. Therefore the optimization task is to search the string, which maximizes the fitness function $F(\vec{X_i})$.

The reproduction of string is based on its percentage probability that is calculated through dividing its fitness value by the total population fitness.

$$P(\vec{X_i}) = F(\vec{X_i}) / \sum_{j=1}^{NN} F(\vec{X_J}) \tag{4.16}$$

where $P(X_i)$ and $F(X_j)$ represent the crossover probability and fitness function value of ith string in the population, and NN is the population size. In this study, one point of crossover technique is used with the probability of 0.7, and mutation randomly alters each gene in the string with a small probability of 0.002.

4.3.4 Simulation Results

The parameters of equation (4.1) and equation (4.2) are taken from [23]: number of rotor poles(Nr)=6, number of phases=4, $L_a = 110mH$, $L_u = 10mH$, and $\phi_s = 1.2Wb$. To save space, we only plot one phase profile including the desired torque profiles and the current profiles over $\theta_e \in [-\frac{\pi}{2}+\epsilon,\ \frac{\pi}{2}-\epsilon]$. Fig. 4.6 shows one phase optimal TSF for a desired torque of $5N-m$.

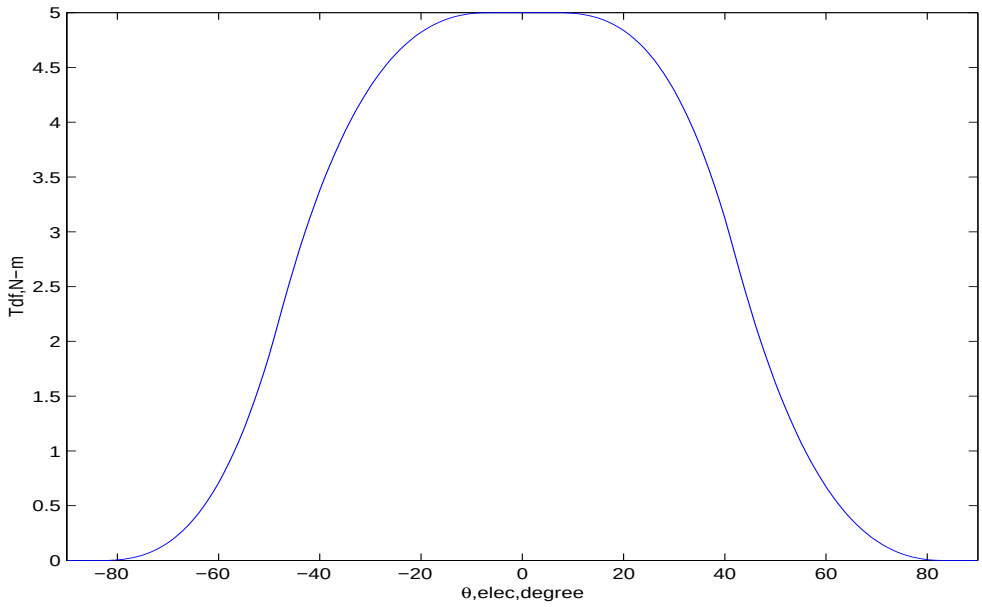

Figure 4.6: The desired torque sharing functions with the global optimum for one phase of $5N-m$.

Using GA to search for the null segment length ϵ, the distance between the point of $(j-1)\frac{\pi}{2}$ and the joint point of two compensating splines l_c and the derivative δ at the joint point of two compensating splines in their own solution ranges, the global optimum cost over 30 runs is $J = 8.3810$, when $\epsilon = 0.1121rad$, $l_c = 0.7241rad$, and $\delta = 4.2857rad/sec$. If we let $\epsilon = 0rad$, $\delta = 0rad/sec$, i.e. we apply the conventional TSF, then we get $J = 23.0474$. Fig. 4.7 shows the current

profile with the global optimum for one phase for desired torque of $5N - m$. The improvement of the current waveform is immediately obvious compared with Fig. 4.2, as the rising segment becomes gentler and the current shape in the falling segment is getting similar to that of the rising segment.

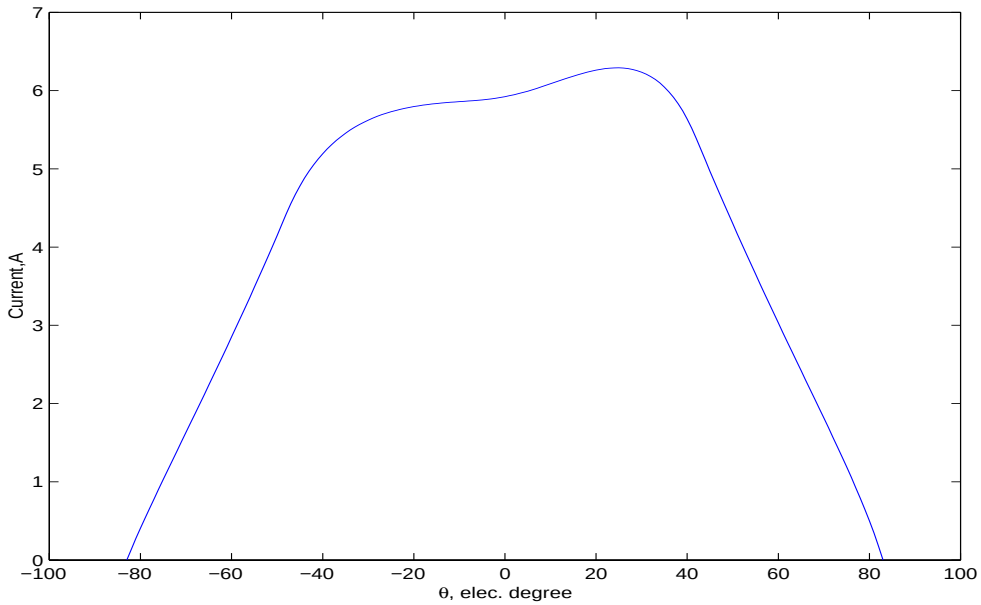

Figure 4.7: The current profile with the global optimum for one phase for desired torque of $5N - m$.

4.4 MultiObjective Optimization

Using the TSFs, two adjacent operating phases will jointly produce a desired torque T_d. In the previous section, the minimization of current variations of only one phase has been considered. In the following study, the minimization of current variations of the two adjacent phases is considered concurrently.

4.4.1 Torque Sharing Function for Multiobjective Optimization

As we discussed in the above, the simplest trajectory that can be used for this purpose is the cubic segment (cubic spline) [69]. With the two-phase-on scheme, each TSF will have a beginning null torque segment, a rising torque segment followed by a full torque segment, a falling segment and an end null segment, which is shown in Fig. 4.3. We define, for the first half period of the torque operation cycle (span from -90 to 0 degrees) of the phase j, the length of the beginning null torque segment to be ϵ_1, the length of the rising torque segment to be l , the length of the full torque segment to be ϵ_2.

Here, the torque trajectory for the rising torque segment is:

$$T_{d,j} = c_1 + c_2\theta_e + c_3{\theta_e}^2 + c_4{\theta_e}^3 \quad \theta_e \in [(j-1)\frac{\pi}{2} + \epsilon_1,\ j\frac{\pi}{2} - \epsilon_2] \tag{4.17}$$

During the full torque segment we have:

$$T_{d,j} = T_d \quad \theta_e \in [j\frac{\pi}{2} - \epsilon_2,\ j\frac{\pi}{2}] \tag{4.18}$$

As for the second half period of phase j, the TSF is determined by the first half period of the next phase analogous to (4.17) and (4.18). This is because the desired torque trajectory of the second half period of phase j is complementary to the first half period of the phase $j+1$, i.e. obtained by subtracting the first half period of phase $j+1$ from the full torque T_d.

From Fig. 4.3, we can observe that, during a null torque segment in the phase $j-1$ and phase $j+1$, phase j should have correspondingly a full torque segment to

meet the total torque requirement. The length of the full torque segment is $\epsilon_1 + \epsilon_2$. It is easy to derive the relationship between l, ϵ_1 and ϵ_2: $l + \epsilon_1 + \epsilon_2 = \frac{\pi}{2}$.

Based on the observation in Section 3, we still choose $\left|\frac{\partial i_j}{\partial \theta_e}\right|$, which is expressed in equation (4.7), as the cost function to be minimized. And a new torque compensation curve shown in Fig. 4.5 is added to the original TSF shown in Fig. 4.3.

$T_{c,j}$ is to reshape the original $T_{d,j}$ for the rising segment only, thus it spans $[(j-1)\frac{\pi}{2} + \epsilon_1,\ j\frac{\pi}{2} - \epsilon_2]$. It actually consists of two cubic splines over two intervals $[(j-1)\frac{\pi}{2}+\epsilon_1,\ (j-1)\frac{\pi}{2}+l_c]$ and $[(j-1)\frac{\pi}{2}+l_c,\ j\frac{\pi}{2}-\epsilon_2]$ respectively. l_c is the distance between the point of $(j-1)\frac{\pi}{2}$ and the joint point of two compensating splines. These two cubic splines together have eight design parameters to be determined from boundary conditions as follows.

$$T_{c,j}(\theta_e) = 0 \tag{4.19}$$

$$\frac{dT_{c,j}}{d\theta_e} = 0 \tag{4.20}$$

at $\theta_e = (j-1)\frac{\pi}{2} + \epsilon_1$ for the first spline;

$$T_{c,j}(\theta_e) = 0 \tag{4.21}$$

$$\frac{dT_{c,j}}{d\theta_e} = 0 \tag{4.22}$$

at $\theta_e = j\frac{\pi}{2} - \epsilon_2$ for the second spline;

$$T_{c,j}(\theta_e) = 0 \tag{4.23}$$

$$\frac{dT_{c,j}}{d\theta_e} = \delta \tag{4.24}$$

at $\theta_e = (j-1)\frac{\pi}{2} + l_c$ for both splines. Here δ is an arbitrary positive constant.

To reduce computation load only four design parameters are introduced and free for us to choose over the period $[(j-1)\frac{\pi}{2}, j\frac{\pi}{2}]$. They are the null torque segment length ϵ_1, the full torque segment length ϵ_2, the distance between the point of $(j-1)\frac{\pi}{2}$ and the joint point of two compensating splines l_c and the derivative δ at the joint point of two compensating splines. By altering these four parameters, the shape of reference torque $T_{d,j}$ varies; accordingly we can obtain various current waveforms. It turns out that we are dealing with an optimization problem, which can be mathematically expressed as below when a single phase is concerned:

$$J_j = \min_{\mathbf{c}\in C} \max_{\theta_e \in \Theta} \left| \frac{\partial i_j}{\partial \theta_e} \right| \tag{4.25}$$

where

$$\begin{aligned} \mathbf{c} &= \{\epsilon_1, \epsilon_2, l_c, \delta\}, \\ C &= [0,\ \frac{\pi}{2}) \cap [0,\ \frac{\pi}{2}) \cap [\epsilon_1,\ \frac{\pi}{2} - \epsilon_2] \cap [0,\ \infty), \\ \Theta &= \left[(j-1)\frac{\pi}{2} + \epsilon_1,\ j\frac{\pi}{2} - \epsilon_2\right] \end{aligned}$$

4.4.2 New Fitness Function

Using the TSFs, two adjacent operating phases will jointly produce a desired torque T_d, i.e:

$$T_d = T_{d,j} + T_{d,j+1} \tag{4.26}$$

Thus the current waveforms of phase j and phase $j+1$ will jointly determine the torque at any instant. Thus the minimization of current variations of the two

adjacent phases should be considered concurrently. Intuitively the objective to be minimized should simply be:

$$J = \frac{1}{2}(J_j + J_{j+1}) \tag{4.27}$$

It does work if the system parameters of two adjacent phases are perfectly identical. If however they are non-identical, the minimization of J may not give reasonable result. For instance, we may have in one case $J_0 = 6$ and $J_1 = 12$ yielding $J = 9$, and in another case $J_0 = 10$ and $J_1 = 10$ yielding $J = 10$. From practical point of view, the later is better as it is more balanced in both phases and the worst case value is less, though its overall J is larger. It is thus necessary to take into consideration of the difference between two adjacent phases. Then the problem turns out to be a multiobjective optimization one. The fitness function can be expressed as the following:

$$J = \frac{1}{2}(J_j + J_{j+1} + \beta\,|J_j - J_{j+1}|) \tag{4.28}$$

It is easy to check that, when $\beta = 1$

$$\begin{array}{ll} J = J_j & \text{when} \quad J_j \geq J_{j+1} \\ J = J_{j+1} & \text{when} \quad J_{j+1} \geq J_j \end{array}$$

The four parameters and their solution space are the same as in equation (4.25). With the new cost function (4.28), we need find out the best solution to determine the suitable torque references so that current waveforms can be greatly smoothed. Fig. 4.8 shows the Pareto curve in the objective domain. An optimal solution for the objective function is to achieve the balance in both phases. Fig. 4.9 shows the relationship between $\max_{\theta_e \in \Theta} \left|\frac{\partial i_j}{\partial \theta_e}\right|$ and l_c, δ with fixed ϵ_1, ϵ_2. Because the cost

function J is highly nonlinear in the parametric space C, GA is applied again to find out the best solution for determining the suitable torque references.

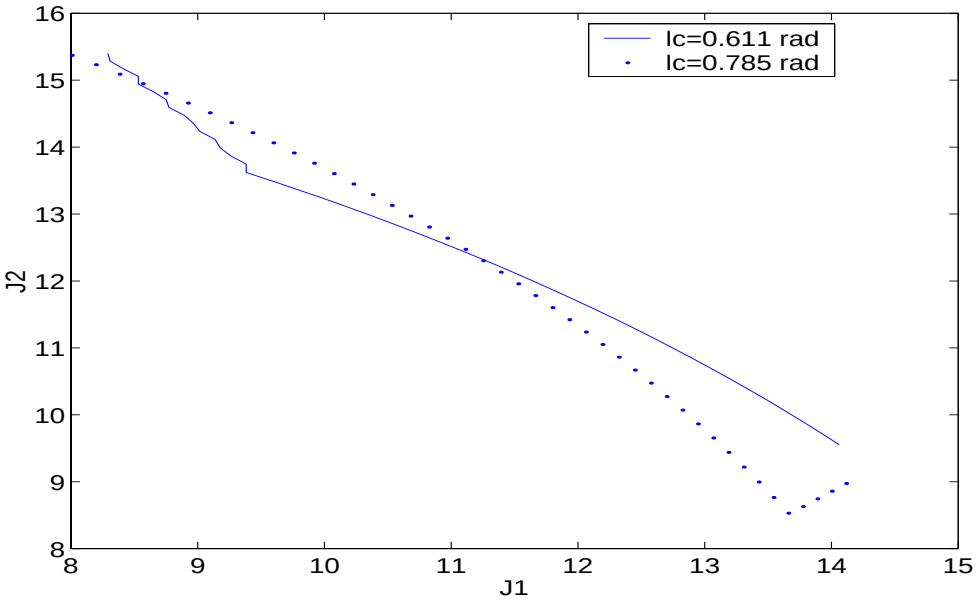

Figure 4.8: The Pareto curve in the objective domain. ($l_c = 0.611rad$: solid line; $l_c = 0.785rad$: dotted line)

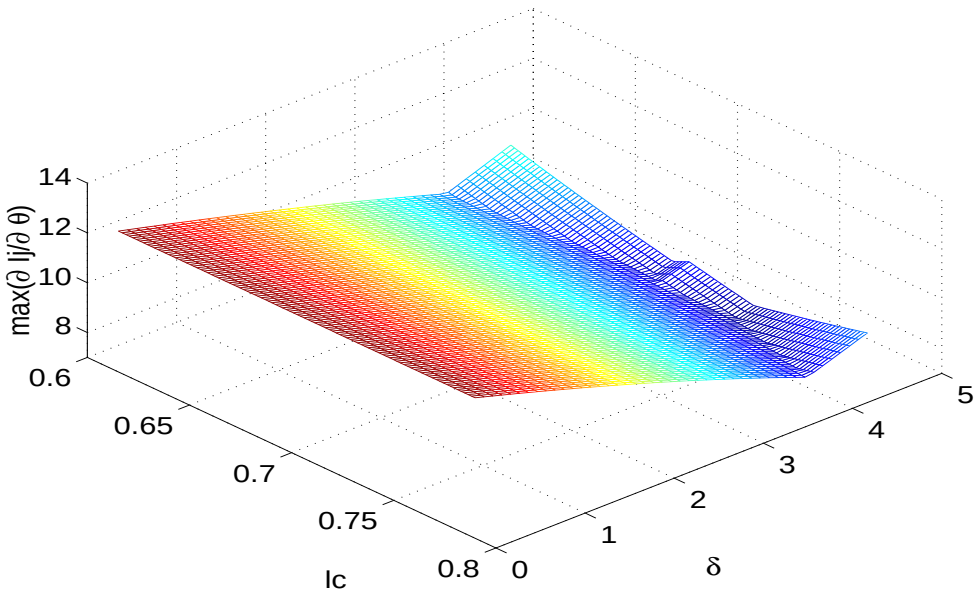

Figure 4.9: The relationship between $\max_{\theta_e \in \Theta} \left| \frac{\partial i_j}{\partial \theta_e} \right|$ and l_c, δ with fixed ϵ_1, ϵ_2.

4.4.3 Case Studies

For the four parameters: $\epsilon_1,\ \epsilon_2,\ l_c,\ \delta$, we selected their lower and upper bound VLB and VUB as follows respectively from a practical point of view: $\epsilon_1 = [\frac{\pi}{180},\ \frac{\pi}{6}], \epsilon_2 = [\frac{\pi}{180},\ \frac{\pi}{6}], l_c = [\frac{7\pi}{36},\ \frac{pi}{4}], \delta = [0, 5]$. The candidate solutions of the four parameters are represented by the binary values. Each individual of the candidate solutions has the appearance shown as below:

ϵ_1	ϵ_2	l_c	δ
$xx...x$	$xx...x$	$xx...x$	$xx...x$

where the output value of each parameter is represented by a 10-bit binary number. A population size and the number of generations are chosen to be 100. We define the fitness function F as the following:

$$F(\vec{X_i}) = J^{-1} \tag{4.29}$$

where $\vec{X_i}$ represents the ith individual in the population. The larger the fitness value is, the better the individual fits. The mutation probability in this study is 0.03.

The parameters of equation (4.1) and equation (4.2) are the same as in single phase optimization. It is assumed that N_r and ϕ_s are fixed for both phases. Phase 1 has the default values of phase inductances L_a and L_u, whereas for phase 0 both phase inductances L_a and L_u are deviate from default values by a scale α. The following five cases were compared. The desired torque is $5N - m$ for all cases.

- Case(i): $\alpha = 1$ with optimization;

Table 4.1: The simulation results over 30 runs for multiobjective optimization

Case	α	J_{min}	J_{mean}	StdDev	ϵ_1	ϵ_2	l_c	δ
Unit					rad	rad	rad	Nm/deg
i	1	8.3569	8.9669	0.1881	0.0941	0.1371	0.7324	4.5238
ii	1.1	8.3829	9.0366	0.2208	0.1011	0.1758	0.6909	4.2063
iii	1.5	8.0524	8.6363	0.2260	0.1288	0.0928	0.7628	4.3651
iv	1	16.7182	16.7182	0	0	0		0
v	1	14.7626	14.7626	0	0.2617	0.2617		0

- Case(ii): $\alpha = 1.1$ with optimization;
- Case(iii): $\alpha = 1.5$ with optimization;
- Case (iv) and Case (v) show the cost of J without optimization, only heuristic parameters for comparison.

Using GA to search for the null torque segment length ϵ_1, the full torque segment length ϵ_2, the distance l_c and the derivative δ, Table. 4.1 shows the simulation results over 30 runs, where J_{min} is the minimum value of J, J_{mean} is the average value of J, StdDev is the standard deviation of J, and the four parameters ϵ_1, ϵ_2, l_c,and δ are for the best solutions over 30 runs. Here $\beta = 1$ is considered

Fig. 4.10 shows the optimal TSF for Case (i). Fig. 4.11, Fig. 4.12 and Fig. 4.13 show the current profiles with the global optimum for Case (i), Case (ii) and Case (iii) respectively. All the four figures use the parameters for the best solutions. To save space, we only plot profiles for two half period including the desired torque profiles and the current profiles over $\theta_e \in [0, \frac{\pi}{2}]$. The improvement

of the current waveforms is immediately obvious compared with Fig. 4.4, as the rising torque segment becomes much more smoother and the current shape in the falling torque segment is getting similar to that of the rising torque segment.

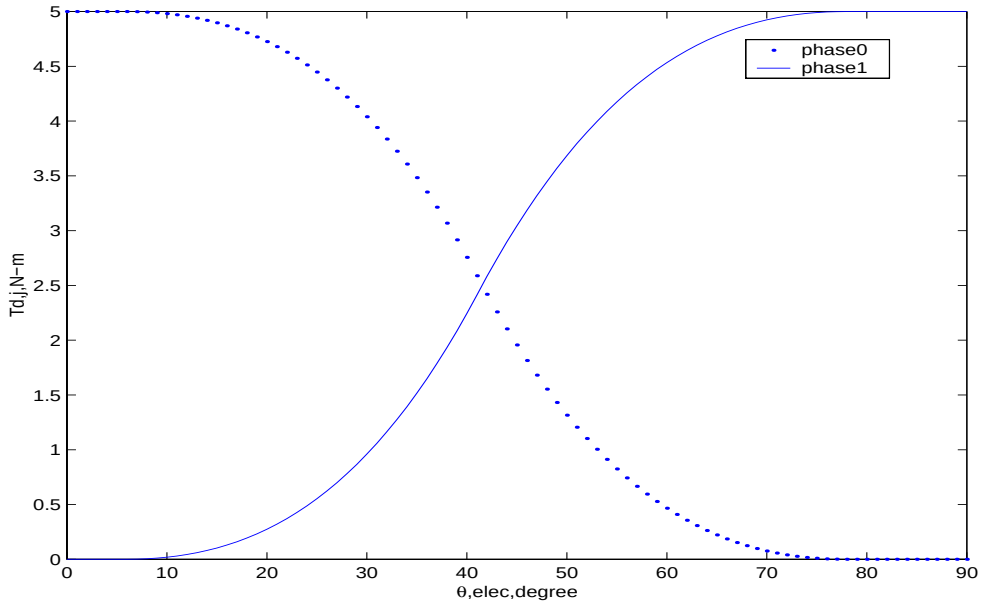

Figure 4.10: The desired torque sharing functions with the global optimum for the half period of two phases in Case (i) (dotted line-phase0; solid line-phase1).

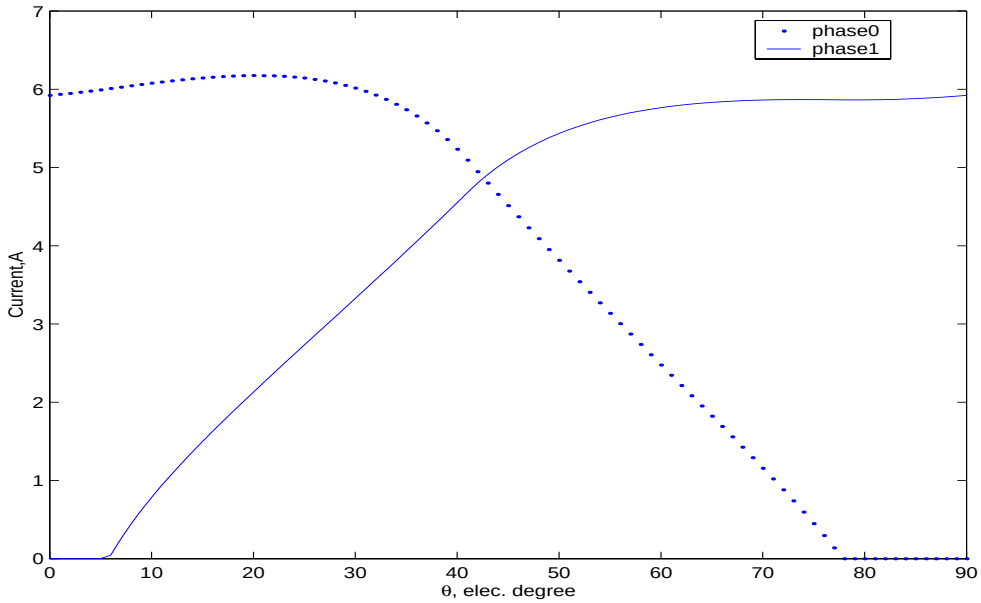

Figure 4.11: The current profile with the global optimum for Case (i) (dotted line-phase0; solid line-phase1).

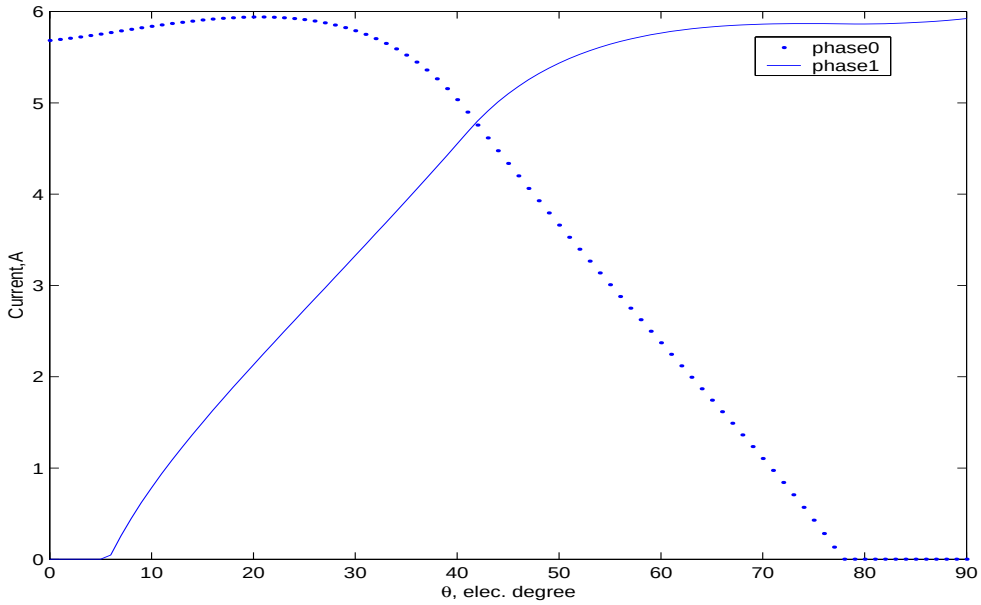

Figure 4.12: The current profile with the global optimum for Case (ii) (dotted line-phase0; solid line-phase1).

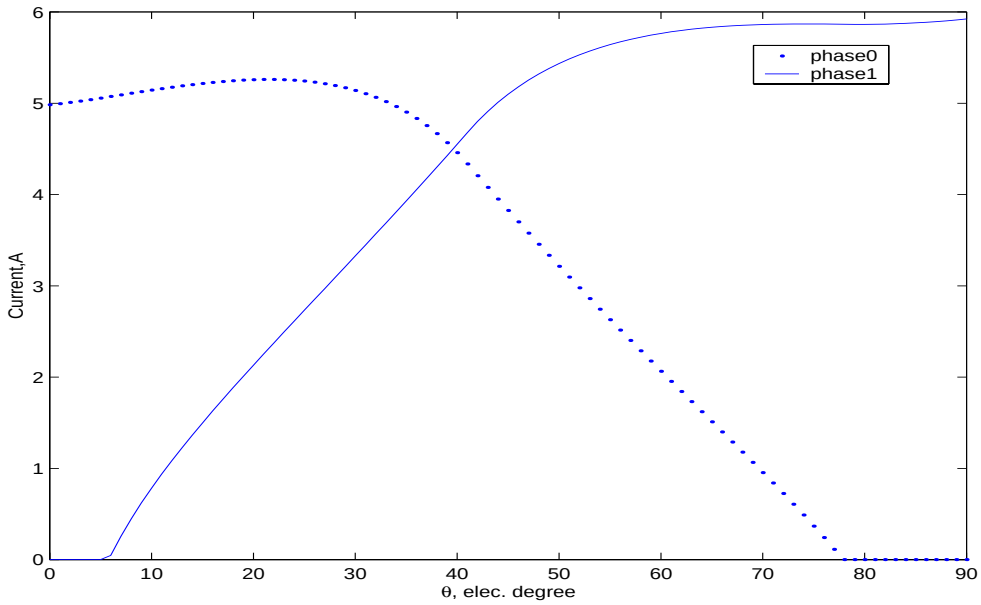

Figure 4.13: The current profile with the global optimum for Case (iii) (dotted line- phase0; solid line-phase1).

Finally let us investigate the effect of the parameter β, which reflects the weighting on the balance of two adjacent phase currents. From

$$J = \frac{1}{2}(J_0 + J_1 + \beta\,|J_0 - J_1|) \tag{4.30}$$

and the discussion in subsection 4.4.2, we know that the larger the β, the more balance will be the maximum current variations of adjacent phases. While zero weighting may lead to unbalanced currents, the over weighted may also generates larger current profiles due to the highly nonlinearities in the parametric space. Consider Case(i), using GA to search for the null torque segment length ϵ_1, the full torque segment length ϵ_2, the distance l_c and the derivative δ when β is different. The results over 50 runs are shown in Table. 4.2. Here J_{0max} and J_{1max} are the values of $\max_{\theta_e \in \Theta} \left| \frac{\partial i_j}{\partial \theta_e} \right|$. We can observe that, J_{0max} and J_{1max} will get closer when β is kept increasing. The best result is achieved at $\beta = 1.0$. Consider Case(iii), the results over 50 runs are shown in Table. 4.3. We can observe that, the best result is achieved at $\beta = 0.8$ instead of $\beta = 1$. This clearly indicates the highly nonlinear nature of the optimization problem. If the computation power allows, β can also be incorporated into GA as the 5th design parameter.

4.5 Conclusion

Determination of suitable current profiles for switched reluctance motors is a challenging problem. In this study, we propose a new TSF and formulated it as an optimal design problem. First GA is applied to provide the global minimum for a single phase. Then we formulated the problem with distinct phases into a multiobjective optimal design problem and proposed a new fitness function. GA is applied again to determine the desired current waveforms for SRM. Simulation results show that the design parameters can be automatically selected by GA and much

Table 4.2: The simulation results over 50 runs for Case(i)

β	J	J_0	J_1	J_{0max}	J_{1max}	ϵ_1	ϵ_2	l_c	δ
0	6.3541	0.8093	11.8988	10.9823	11.8988	0.1011	0.2225	0.6410	2.3810
0.1	6.4092	5.9362	6.7962	11.0359	8.5600	0.1149	0.0983	0.6715	3.2540
0.2	6.8504	11.2156	0.3026	11.2156	9.9966	0.1634	0.1509	0.7352	2.8571
0.3	7.0817	5.9601	7.6857	10.1778	9.2833	0.1426	0.1758	0.6936	3.4921
0.4	7.3905	7.1654	7.4869	9.4769	8.2105	0.1149	0.1509	0.6770	3.8889
0.5	7.7696	6.7136	8.1215	9.9597	9.2774	0.1288	0.2118	0.6715	3.8095
0.6	7.8379	0.9365	9.5633	9.5098	9.5633	0.0941	0.1343	0.7795	3.5714
0.7	8.0159	0.9414	9.2643	9.0829	9.2643	0.0872	0.1952	0.6770	3.8889
0.8	8.7551	9.7099	0.1617	9.7099	8.8758	0.0941	0.1426	0.7269	3.8095
0.9	8.9963	0.9358	9.4205	8.4285	9.4205	0.0872	0.2284	0.6604	4.0476
1.0	8.3568	6.0892	8.3568	8.0664	8.3568	0.0941	0.1371	0.7324	4.5238

Table 4.3: The simulation results over 50 runs for Case(iii)

β	J	J_0	J_1	J_{0max}	J_{1max}	ϵ_1	ϵ_2	l_c	δ
0	6.3081	0.6919	11.9242	9.6586	11.9242	0.0940	0.1731	0.7047	2.1429
0.1	6.1087	5.5649	6.5537	9.9732	10.0617	0.1011	0.1288	0.6964	2.8571
0.2	6.5392	10.6445	0.3813	10.6445	9.9786	0.0941	0.0872	0.7684	2.8571
0.3	6.7482	6.4829	6.8910	9.1250	8.5807	0.1149	0.1011	0.7241	3.4921
0.4	7.6883	10.8522	0.3059	10.8522	8.0087	0.1426	0.1038	0.6964	3.5714
0.5	7.7933	7.2174	7.9852	7.7225	8.2864	0.1011	0.1758	0.6909	4.2063
0.6	7.7979	0.8387	9.5376	9.1370	9.5376	0.1011	0.1537	0.7241	3.4921
0.7	8.2211	8.5424	6.4003	8.5424	8.0431	0.0872	0.1426	0.7296	4.8413
0.8	8.1023	7.6353	8.1541	7.9012	8.2543	0.1357	0.1259	0.7047	4.2857
0.9	8.2043	5.8106	8.1408	7.9531	8.2733	0.1011	0.1619	0.6521	3.8889
1.0	8.6799	8.6799	0.0121	8.6799	8.6127	0.1218	0.1897	0.6687	3.9683

smoother current waveforms are generated when comparing with conventional TSF design using heuristic knowledge.

Chapter 5

Conclusion

5.1 Findings and Conclusions

This thesis deals with an investigative study on application of intelligent tools for SRM modelling and high performance torque control. An accurate nonlinear dynamic model is the basis for further investigations of the SRM, such as the design of a high-performance ripple-free dynamic torque controller. However, both the modelling task and the control task are inherently difficult because of highly nonlinear torque-current-position characteristics of the motor.

Accurate measurement of the flux-linkage characteristics of SRM is required for modelling the SRM and developing high-performance torque control schemes. Therefore, in Chapter 2, the study is started with the measurement of flux-linkage characteristics of the experimental SRM. For the physical SRM, the flux-linkage has been obtained from the experimental voltage and current data using the method of rising current at constant voltage and numerical integration. In this chapter,various

efforts have been made for the flux-linkage modelling. Both analytical model based approaches and blackbox based approach are applied for flux-linkage modelling of the SRM. For the analytical model based approaches, LM gradient expansion method and GA have been employed. The measured flux-linkage data have been fitted with a suitable analytical model whose coefficients are found by LM gradient-expansion method and GA. These coefficients are represented by fifth order polynomials. For the model-free blackbox based approach, ANN techniques are employed. Simulation results verify the effectiveness of ANNs for achieving high accuracy in flux-linkage modelling.

Chapter 3 has been concerned with the highly nonlinear torque modelling and inverse torque modelling of the SRM. First the standard electric machine theory is applied for torque prediction through combining with the flux-linkage model. However, it is not accurate because of the possibly unavoidable measurement error. Then both analytical model based approaches and blackbox based approaches for torque modelling and inverse torque modelling of the SRM are adopted. For analytical model based approaches, we let GA provide coarse search, and LM gradient-expansion method refine the search to achieve an accurate analytical torque model. The quality of curve fitting for torque is relatively satisfactory after combining GA and LM gradient-expansion method. Considering the long process for identifying the torque model and the impossibility of modelling the inverse torque characteristics by analytical model based approaches, the blackbox based approaches are employed and ANNs with different algorithms are applied for torque modelling

and inverse torque modelling. Simulation results show that ANNs can achieve the mapping task in high accuracy and fast with BP algorithm. For comparison, RBF networks are used for torque modelling. The training result shows that RBF networks are not suitable for this study. Last, experimental results verify that the model based torque estimator output matches the torque transducer output perfectly and the ANN based torque estimator output matches the torque transducer output well for the average torque. Simulation and experimental results verify the effectiveness of the derived torque models for achieving high accuracy and their respective advantages.

In Chapter 4, the TSF approach has been proposed to address the problem of minimizing or eliminating the torque ripple for high performance torque control. First we propose a new TSF and formulate it as an optimal design problem. Subsequently, we formulate the problem with distinct phases into a multiobjective optimal design problem and propose a new fitness function. GA is employed to determine the desired current waveforms of the SRM for torque ripple minimization through generating appropriate reference phase torques for a given desired torque. Simulation results show that the design parameters can be automatically selected by GA and much smoother current waveforms are generated when comparing with conventional TSF design using heuristic knowledge and therefore verify the effectiveness of the proposed TSF and fitness function.

5.2 Remarks on Future Research

Further investigations to improve the performance of the proposed algorithms can be done with the following possible directions:

- In Chapter 3, Experimental results verify that the model based torque estimator output matches the torque transducer output perfectly and the ANN based torque estimator output matches the torque transducer output well for the average torque. However, the instantaneous ANN based torque estimator output does not match the measured torque quite well. Hence on-line ANN based training model for dynamic conditions could be considered as the torque estimator.

- ANN based off-line training may not fit real dynamics. Using ANNs to generate initial values for iterative learning control in real dynamics is an alternative way to improve torque control performance.

Bibliography

[1] P. C. Sen, "Electric motor drives and control - Past, Present and Future," *IEEE Transactions on Industrial Electronics*, Vol. IE-37, No.6, December 1990, pp. 562-575.

[2] P. Vas, "Vector Control of AC Machines", *Oxford University Press*, New York, 1990.

[3] W. Leonard, "Control of Electric Drives," *Springer-Verlag*, Berlin, 1985.

[4] G. K. Dubey, "Power Semiconductor Controlled Drives," *Prentice Hall International*, Englewood Cliffs, 1989.

[5] T.J.E. Miller, Ed., *Electronic Control of Switched Reluctance Motors.* ser. Newnes Power Engineering Series. Oxford, U.K.: Newnes,2001.

[6] P. J. Lawrenson, J. M. Stephenson, P. T. Blenkinsop, J. Corda and N. N. Fulton, "Variable-speed switched reluctance motors," *IEE Proceedings Part-B*, Vol. 127, No. 4, July 1980, pp. 253-265.

[7] T. J. E. Miller, "Switched Reluctance Motors and their Control," *Clarendon Press, Oxford*, 1993.

[8] J. M. Stephenson, "Switched Reluctance Motors," *Section 2 of Tutorial on Switched Reluctance Drives, IEEE IAS Annual Meeting*, Seatle, WA, 1990.

[9] T. J. E. Miller and T. M. Jahns, "A current-controlled switched reluctance drive for FHP applications," *Proceedings of the Conference on Applied Motion Control*, June 1986, pp. 109-117.

[10] J. M. Stephenson, P. J. Lawrenson and N. N. Fulton, "High Power Switched Reluctance Drives," *Proceedings of the International Conference on Electrical machines*, Munich, West Germany, 1986, pp. 1052-1055.

[11] N. N. Fulton and P. J. Lawrenson, "Switched reluctance drives for electric vehicles: A comparative assessment," *Proceedings of the Intelligent Motion*, June 1993, pp. 562-579.

[12] J.A. Kline, "Opportunities for switched reluctance motor-drives ", *Industry Technical Conference Record of Pulp and Paper*, June 1999, pp. 42 -47.

[13] J.G.W., West, "DC, induction, reluctance and PM motors for electric vehicles", *Power Engineering Journal*, Vol. 3, April 1994, pp.77-88.

[14] F.J. Barto,"Forward to the Past" with SR Technology, Technical Report, CONTROL ENGINEERING, www.controleng.com, 1999.

[15] C.A. Ferreira, S.R. Jones, B.T. Drager, and W.S. Heglund, "Design and implementation of a five-hp, switched reluctance, fuel-lube, pump motor drive for a gas turbine engine.", *IEEE Transactions on Power Electronics*, Vol. 10, No. 1, Jan 1995, pp. 55-61.

[16] D. S. Reay, M. M. Moud and B. W. Williams, "On the appropriate use of fuzzy systems: fuzzy sliding mode position control of a switched reluctance motor," *International Journal of Intelligent Control and Systems*, Vol. 1, No. 3, 1996, pp. 417-426.

[17] P. Tandon, A. V. Rajarathnam and M. Ehsani, "Self-tuning control of a switched reluctance motor drive with shaft position sensor," *Proceedings of the IEEE IAS Annual Meeting*, Vol. 1, October 1996, pp. 101-108.

[18] U. Y. Huh and T. G. Lee, "Fuzzy logic based switching angle controller for SR motor speed control," *Proceedings of the IEEE International Symposium on Industrial electronics*, Vol. 2, July 1995, pp. 809-814.

[19] M. Barnes and C. Pollock,"Power Electronic Converters for Switched Reluctance Drives", *IEEE Transactions on power electronics*, Vol. 13, No. 6, November 1998, pp.1100-1111.

[20] C. H. Kim and I. J. Ha, "A new approach to feedback-linearizing control of variable reluctance motors for direct drive applications," *IEEE Transactions on Control Systems Technology*, Vol. 4, No. 4, July 1996, pp. 348-362.

[21] L. B. Amor, L. Dessaint, O. Akhrif and G. Oliver, "Adaptive feedback linearization for position control of a switched reluctance motor: analysis and simulation," *International Journal of Adaptive Control and Signal Processing*, Vol. 7, 1993, pp. 117-136.

[22] C. Rossi and A. Tonielli, "Feedback linearizing and sliding mode control of a variable reluctance motor," *International Journal of Control*, Vol. 60, No. 4, 1994, pp. 543-568.

[23] G. S. Buja, R. Menis amd M. I. Valla, "Variable structure control of an SRM drive," *IEEE Transactions on Industrial Electronics*, Vol. IE-40, No. 1, February 1993, pp. 56-63.

[24] D. S. Reay, M. M. Moud, T. C. Green and B. W. Williams, "Switched reluctance motor control via fuzzy adaptive systems," *IEEE Control Systems*, Vol. 15, No. 3, June 1995, pp. 8-15.

[25] M. Ilić-Spong, R. Marino, S. M. Peresada and D. G. Taylor, "Feedback linearizing control of switched reluctance motors," *IEEE Transactions on Automatic Control*, Vol. AC-32, No. 5, May 1987, pp. 371-379.

[26] D. A. Torrey and J. H. Lang, "Modeling a nonlinear variable reluctance motor drive," *IEE Proceedings Part-B*, Vol. 137, No. 5, September 1990, pp. 314-326.

[27] D. G. Taylor, "An experimental study on composite control of switched reluctance motors," *IEEE Control Systems*, Vol. 11, No. 2, February 1991, pp. 31-36.

[28] F. Filicori, C. G. L. Bianco and A. Tonielli, "Modeling and control strategies for a variable reluctance direct-drive motor," *IEEE Transactions on Industrial Electronics*, Vol. 40, No. 1, February 1993, pp. 105-115.

[29] D. G. Manzer, M. Varghese and J. S. Thorp, "Variable reluctance motor characterization," *IEEE Transactions on Industrial Electronics*, Vol. 36, No. 1, February 1989, pp. 56-63.

[30] T. J. E. Miller and M. McGilp, "Nonlinear theory of the switched reluctance motor for rapid computer-aided design," *IEE Proceedings Part-B*, Vol. 137, No. 6, November 1990, pp. 337-347.

[31] J. C. Moreira, "Torque ripple minimization in switched reluctance motors via bicubic spline interpolation," *IEEE Power Electronics Specialists Conference Record*, Toledo, Spain, June 1992, pp. 851-856.

[32] Stephenson J. M. and Corda. J., " Computation of torque and current in doubly salient reluctance motors from nonlinear magnetization data", *IEE Proceedings*, Vol. 126, No. 5, May 1979, pp. 393-396.

[33] C. Elmas, S. Sagiroglu, I. Colak, G. Bal,"Modeling of a nonlinear switched reluctance drive based on artificial neural network," *Proceedings of the fifth International Conference on Power Electronics and Variable-Speed Drives*, No. 399, 26-28 Oct 1994, pp. 7-12.

[34] B. Fahimi, G. Suresh, M. Ehsani, "Torque estimation in switched reluctance motor drive using artificial neural networks", *Proceedings of 23rd International Conference on Industrial Electronics, Control and Instrumentation*, Vol. 1, 9-14 Nov 1997, pp. 21 -26.

[35] V. Trifa, E. Gaura, L. Moldovan, "Neuro-control approach of switched reluctance motor drives", *Proceedings of Electrotechnical Conference on MELECON*, Mediterranean, Vol: 3, 13-16 May 1996, PP. 1461 -1464.

[36] Changjing Shang, D. Reay, B. Williams, "Learning rate functions in CMAC neural network based control for torque ripple reduction of switched reluctance motors", *Proceedings of IEEE International Conference on Neural Networks*, Vol. 4 , 3-6 Jun 1996, pp. 2078 -2083.

[37] J.G. O'Donovan, P.J. Roche, R.C. Kavanagh, M.G. Egan, J.M.D. Murphy, "Neural network based torque ripple minimization in a switched reluctance motor", *Proceedings of the 20th International Conference on Industrial Electronics, Control and Instrumentation*, Vol. 2, 5-9 Sep 1994, pp. 1226 -1231.

[38] B. K. Bose, T. J. E. Miller, P. M. Szczesny and W. H. Bicknell, "Microcomputer control of switched reluctance motor," *IEEE Transactions on Industry Applications*, Vol. IA-22, No. 4, 1986, pp. 708-715.

[39] A. R. Oza, R. Krishnan and S. Adkar, "A microprocessor control scheme for switched reluctance motor drives," *Proceedings IECON*, 1987, pp. 448-453.

[40] J. Ish-Shalom and D. G. Manzer, "Commutation and control of step motors," *Proceedings of the 14th Annual Symposium on Incremental Motion, Control Systems and Devices*, 1985, pp. 283-292.

[41] R. S. Wallace and D. G. Taylor, "A balanced commutator for switched reluctance motors to reduce torque ripple," *IEEE Transactions on Power Electronics*, Vol. 7, No. 4, October 1992, pp. 617-626.

[42] A. Isidori, "Nonlinear Control Systems," *Springer-Verlag*, Berlin, 1989.

[43] S.K Panda, and P.K Dash, "Application of nonlinear control of switched reluctance motors: a feedback linearisation approach," *IEE Proc. Electr. Power Appl.*, Vol. 143, No. 5, 1996, pp.371-379.

[44] S. K. Panda, C. W. Low and P. K. Dash, "Hybrid feedback linearization-fuzzy controller for variable reluctance motors," *International Journal of Electronics*, Vol. 80, No. 2, 1996, pp. 155-167.

[45] M. Ilić-Spong, T. J. E. Miller, S. R. Macminn and J. S. Thorp, "Instantaneous torque control of electric motor drives," *IEEE Transactions on Power Electronics*, Vol. PE-2, No. 1, January 1987, pp. 55-61.

[46] M. Y. Chow and R. J. Thomas, "Constant torque control of a variable reluctance machine," *International Journal of Energy Systems*, Vol. 11, No. 1, 1991, pp. 47-51.

[47] J.Y. Hung, "Torque ripple minimization for variable reluctance motors," *Mechatronics*, Vol. 4, No. 8, 1994, pp.785-798.

[48] H.C. Lovatt, and J.M Stephenson, "Computer-optimized current waveforms for switched reluctance motors," *IEE Proc. Electr. Power Appl.*, Vol. 141, No. 2, 1994, pp. 45-51.

[49] J.M. Stephenson, "Computer-optimized smooth-torque current waveforms for switched reluctance motors," *IEE Proc. Electr. Power Appl.*, Vol. 144, No. 5, 1997, pp. 310-316.

[50] H. C. Lovatt and J. M. Stephenson, "Computer-optimised smooth-torque current waveforms for switched reluctance motors," *IEE Proceedings Part-B*, Vol. 144, No. 5, 1997, pp. 310-316.

[51] P. R. Bevington and D. K. Robinson, "Data reduction and error analysis for the physical sciences," *McGraw Hill*, Singapore, 1992.

[52] D.E. Goldberg, *Genetic Algorithm in Search, Optimization, and Machine Learning*, Addison-wesley, 1989.

[53] R. Krishnan and P. Materu, "Measurement and instrumentation of a switched reluctance motor," *Proceedings of the IEEE IAS Annual Meeting*, Vol. 1, 1989, pp. 116-121.

[54] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, Vol. 11, No. 2, June 1963, pp. 431-441.

[55] Khor, E. F., Tan, K. C. and Lee, T. H., "Learning the search range for evolutionary optimization in dynamic environments",*Knowledge and Information Systems: An International Journal*, vol. 4, 2002, pp. 228-255.

[56] Dipti Srinivasan, Tian Hou Seow, Jian-Xin Xu, "Automated time table generation using multiple context reasoning for university modules", *Congress on Evolutionary Computation 2002*, Hawaiian,USA, 2002, pp. 1751-1756.

[57] Bahram Kimiaghalam, Abdollah Homaifar, Marwan Bikdash, Bijan Sayyarrodsari, "Genetic algorithm based gain scheduling", *Congress on Evolutionary Computation 2002*, Hawaiian,USA, 2002, pp. 540-545.

[58] Dirk Thierens, "Adaptive mutation rate control schemes in genetic algorithms", *Congress on Evolutionary Computation 2002*, Hawaiian,USA, 2002, pp. 980-985.

[59] K.S. Narendra, "Neural networks for control theory and practice", *Proceedings of the IEEE*, Vol. 84, No. 10, Oct 1996, pp. 1385-1406.

[60] D.E. Rumelhart and J.L. McClelland, *Parallel Distributed Processing: Explorations in the Micro-structure of Cognition*, Cambridge MA: MIT Press, Vol. 1, 1986.

[61] H. Demuth and M. Beale, *Neural network toolbox user's guide for use with Matlab*, The MathWorks, Inc, July 1998.

[62] I. Husain, M. Ehsani, "Torque ripple minimization in switched reluctance motor drives by PWM current control", *IEEE APEC Records*, vol.1, 13-17 Feb 1994, pp. 72 -77.

[63] P. Tandon, A. Velayutham Rajarathnam, M. Ehsani, "Self-tuning control of a switched-reluctance motor drive with shaft position sensor" *IEEE Trans-*

actions on Industry Applications,Vol. 33, No. 4, July/August 1997, pp. 1002 -1010.

[64] P. C. Krause, "Analysis of Electric Machinery," *McGraw Hill*, New York, 1986.

[65] N. C. Sahoo, " A Study on Application of Modern Control techniques for Torque Control of Switched Reluctance Motors", Ph.D Thesis, National University of Singapore, 2001.

[66] R. B. Inderka, and R. W. De Doncker, "DITC-direct instantaneous torque control of switched reluctance drives", *IEEE IAS Annual Meeting*,Vol. 3, 2002, pp. 1605-1609.

[67] I. Husain, M. Ehsani, "Torque ripple minimization in switched reluctance motor drives by PWM current control", *IEEE Transactions on Power Electronics*, Vol. 11, No. 1 , Jan 1996, pp. 83 -88.

[68] Simon Haykin, "Neural Networks," *Prentice Hall*, New Jersey, 1999.

[69] J.X. Xu, S.K. Panda, and Q. Zheng, "Genetic Algorithm based optimization of current waveforms for switched reluctance motors," *Proceedings of International Conference on Computational Intelligence, Robotics and Autonomous Systems*, Novemeber 2001, Singapore, pp. 260-265.

[70] J.X. Xu, S.K. Panda, and Q. Zheng, "Multiobjective Optimization of Current Waveforms for Switched Reluctance Motors by Genetic Algorithm," *Proceedings of the 2002 Congress on Evolutionary Computation*, Vol. 2 May 2002, USA, pp. 1860 -1865.

[71] N.C. Sahoo, J.X. Xu, and S.K. Panda, "Determination of current waveforms for torque ripple minimisation in switched reluctance motors using iterative learning: an investigation," *IEE Proc-Electr. Power Appl.*, Vol. 146, No. 4, 1999, pp. 369-377.

Appendix A

Author's Publications

1. J.X. Xu, S.K. Panda, and Q. Zheng, "Genetic Algorithm based optimization of current waveforms for switched reluctance motors," *Proceedings of International Conference on Computational Intelligence, Robotics and Autonomous Systems*, November 2001, Singapore, pp. 260-265.

2. J.X. Xu, S.K. Panda, and Q. Zheng, "Multiobjective Optimization of Current Waveforms for Switched Reluctance Motors by Genetic Algorithm," *Proceedings of the 2002 Congress on Evolutionary Computation*, Vol. 2, May 2002, USA, pp. 1860 -1865.

3. S.K. Sahoo, Q. Zheng, S.K. Panda, J.X. Xu, "Model-based Torque Estimator for Switched Reluctance Motors", *International Conference on Power Electronics and Drive Systems*, November 2003, Singapore, pp. 959-963.

4. J.X., Xu, S.K., Panda, and Q., Zheng, "Multiobjective Optimization of Current Waveforms for Switched Reluctance Motors by Genetic Algorithm", *the International Journal of Modelling and Simulation*, accepted for publication.

5. J.X. Xu, Q. Zheng, S.K. Panda, S.K. Sahoo, "A Study on Torque Modelling of Switched Reluctance Motors", submitted.