OPTIMISATION OF THE EXTENDED KALMAN FILTER FOR SPEED ESTIMATION OF INDUCTION MOTOR DRIVES

Salinda Buyamin
B.Sc., M.Sc

A thesis submitted for the degree of Doctor of Philosophy

July, 2007

School of Electrical, Electronic and Computer Engineering

Newcastle University
United Kingdom

ACKNOWLEDGEMENTS

I would like to say thank you to everyone in the Power Electronics, Drives and Machines Group at the University of Newcastle upon Tyne for their help and support whether either directly or indirectly involved. Sincere thanks goes to Professor John W. Finch for his advice, support, understanding and useful discussion as well as competent supervision throughout the research.

Thanks also go to my friend and colleagues in the Upper Ground Floor Laboratory, Afida, Khairul Parman, Abdul Majeed, Shady, George, Ed, Matthew and Yousif for their help and friendly discussions. Not forgetting Dan Dan, Simon, and all in the lab for bringing special moments of joy while working in the lab. Thanks also to the Universiti Teknologi Malaysia UTM for the financial support.

A great appreciation to my parents, Buyamin and Misiah and my parents-in-law, Aliman and Kamilah for their continuous prayer, support, care and love. Thanks also for my brothers and sisters particularly my little sister, Saliana and her husband, Rashdin who gave the best of themselves to represent me in the family during my absence.

Another great appreciation and a special thanks to my dear husband Mohamad Nizam for his dedication, support, understanding, patience, care and love through out the studies. Not forgetting my son, Sofi Hazim for being a very lovely son.

Above all, I would like to express my great gratefulness to God for giving me the strength to do this work.

ABSTRACT

A speed sensorless drive requires the elimination of the sensor; therefore a speed estimator is required. Speed estimation using the Extended Kalman Filter (EKF) is investigated. The use of an EKF as an observer for a sensorless induction motor has been a longstanding subject of research. However, little attempt has been made to optimise the filter performance. First some speed estimation results are presented where the commonly used Trial and Error method is used for tuning the EKF.

The performance of the EKF is strictly dependant on the choice of the covariance matrices. Therefore to improve the performance of the EKF, a guided random search technique, Simulated Annealing is proposed. The work concentrates on finding the EKF parameters by the Simulated Annealing algorithm in both low and high performance drives, for constant V/f and vector control.

A Genetic Algorithm is also a guided random search technique and in this work the algorithm has been used for comparison purposes on optimising the EKF. The robustness of the EKF parameters tuned by Genetic Algorithm, Simulated Annealing and Trial and Error is compared. The results presented show that Simulated Annealing is more robust against machine parameter variations. Despite the large computation time Simulated Annealing does have the potential of being an alternative method for optimising the EKF. These novel results presented here show that Simulated Annealing is capable of tuning the EKF in the induction motor drives application.

NOMENCLATURE

List of Principal Symbols

i_{ds}^{*} i_{qs}^{*}	D-Q axis stator current in excitation reference frame
i^{s}_{ds}, i^{s}_{qs}	Components of stator current vector in stator reference frame
i_{ds}, i_{qs}	Components of stator current vector in excitation reference frame
i^{s}_{dr}, i^{s}_{qr}	Components of rotor current vector in stator reference frame
i^{r}_{dr}, i^{r}_{qr}	Components of rotor current in rotor reference frame
V^{s}_{ds}, V^{s}_{qs}	Component of stator voltage vector in stator reference frame
K_t	Torque constant
Ψ_f	Field flux linkage
Ψ_a	Armature flux linkage
Ψ_r	Rotor flux
I_a	Armature current
I_f	Field current
θ_e	Angle of synchronously rotating frame
i_a, i_b, i_c	Output currents of inverter, stator current
θ_r	Rotor Flux Angle
T_e	Electromagnetic Torque
T_L	Load Torque
J	Rotor Inertia
ω_r	Rotor angular velocity
i_{dse}	Flux producing current
i_{qse}	Torque producing current
T_r	Rotor Time Constant
R_s	Stator Resistance

R_r	Rotor Resistance
L_s	Stator Self Inductance
L_r	Rotor Self Inductance
L_m	Stator to Rotor Mutual Inductance
V_s	Stator Voltage
I_s	Stator Current
I_r	Rotor Current
I_m	Magnetizing Current
$A(t)$	State Matrix
$B(t)$	Input Matrix
$C(t)$	Output Matrix
$D(t)$	Feed Forward Matrix
$x(t)$	Continuous State Vector
$y(t)$	Continuous Output Vector
$u(t)$	Continuous Input Vector
x_k	Discrete State Vector
y_k	Discrete Output Vector
u_k	Discrete Input Vector
Q	Process Noise Covariance Matrix
R	Measurement Noise Covariance Matrix
G	Noise Gain Weight Matrix
p	Time differential operator
I	Identity Matrix
W_k	Process Noise
V_k	Measurement Noise
$\overline{\Psi}_r$	Significance of rotor flux vector

Abbreviations

AC	Alternating Current
DC	Direct Current

EKF	Extended Kalman Filter
KF	Kalman Filter
GA	Genetic Algorithm
IM	Induction Machine
PM	Permanent Magnet
MRAS	Model Reference Adaptive System
RMS	Root Mean Squared
SA	Simulated Annealing
SRM	Switched Reluctance Motor
SyncRM	Synchronous Reluctance Motor
VC	Vector Control
V/f	Voltage/frequency

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION TO THE CONTROL OF INDUCTION MOTOR DRIVES

1.1 Introduction

There has been much research and development in recent years for industrial and servo electric drives. This growth has been inspired by new material and construction techniques, as well as advances in power electronics and control. In both industrial and servo electric drives traditional DC brushed machines are being replaced by more compact and reliable brushless AC techniques. Therefore, some of the AC machines such as the induction motor (IM), switched reluctance motor (SRM), permanent magnet (PM) and synchronous reluctance motor (SyncRM) have become alternatives in electrical drives applications.

In particular, the IM is comparatively cheaper to manufacture than the DC motor. It also offers features such as ruggedness, high reliability, high efficiency, minimum maintenance and structure robustness due to its cast squirrel cage rotor construction. This advantage has made IM an alternative choice over DC and also the most commonly used among other AC machines.

Much of the research mentioned has been devoted to improving the drive systems of the IM, especially the control methodology. IM drives can be divided into low and high performance drives. The simple constant Volt/Hertz (V/f) drive ensures the stator flux is maintained at its constant value by adjusting the magnitude of the stator voltage in proportion to the frequency. It is operated on a feed forward system and is suitable for a low performance drive. High performance drives, usually based on field oriented control or vector control (VC), first developed more than 30 years ago, have shown that the IM under certain condition can behave as a DC motor.

One of the most significant developments in the control area is elimination of the speed sensor at the machine shaft. Speed sensorless control is required in applications where a speed sensor may not be installed due to the hostile environment, or to improve the reliability of the system. The sensorless approach has been extensively applied to electrical machines especially IMs. The nonlinear structure of IM causes limitations in the sensorless application and constitutes a challenging control problem but this has already been

overcome to some extent. Many different solutions have been proposed for sensorless IMs, such as improving the stator flux estimation, aiming for zero speed operation and eliminating the start-up hesitation.

As a consequence of using sensorless IM, the speed observer has always been a topic of interest amongst IM control researchers. The speed observer requires a plant model to represent the real model and measured quantities to evaluate the error between the actual and estimate. In the literature, many different techniques have been proposed to estimate the speed of an induction machine such as sliding mode, adaptive, MRAS based system, Neural Network and Extended Kalman Filter (EKF).

The Kalman Filter (KF) is a well established stochastic technique used in estimation problems. The stochastic method includes random disturbances and measurement errors of the system in solving the estimation problem. The KF is capable of estimating the non measured parts of a linear dynamic system. It is a recursive type of estimator and suitable for real time estimation although requiring a high digital signal processing rate.

For a sensorless closed loop system, speed information is essential. The application of the EKF as a speed observer has been a long standing issue. The EKF capable of estimating a system states, parameter and speed, simultaneously. One major problem in the application of EKF is the choice of the EKF covariance matrices. To determine the observer performance, it is essential to have an accurate knowledge of the EKF parameters. A tuning process based on the Trial and Error method is frequently used. This method is easy to use, but time consuming, less accurate and dependent on the individual skills of the designer. Therefore, to achieve optimal performance of the observer, a straight forward method is required for replacing the Trial and Error method.

The main intent of this work is to offer an alternative method in optimising the EKF for the estimation application of the squirrel cage IM. The main motivation is to reduce the complexity of the tuning process of the EKF parameters as well as reducing the mechanical connections, sensorless, by using speed observer. Investigation of the Simulated Annealing (SA) method has been carried out and the performance of the speed estimator has been compared to those obtained by using Trial and Error and the Genetic Algorithm (GA) method.

1.2 Scope of the Thesis

In this thesis, the work is presented as follows. Chapter 1 gives a general introduction of the main work. In chapter 2, sensorless IM drives control in low and high performance drives is discussed. A literature review on speed observers of sensorless IM, the observer problem and the development as well as the optimisation approach of the EKF is also presented to show what has been done by other researchers.

Chapter 3 covers the modelling of the IM. This chapter give details on modelling using space vector notation, mathematical equations and state variables. A continuous state space model is presented in the simulation and the dynamic performance of the models is also given.

In Chapter 4, the state estimator using a KF is introduced. The purpose of this chapter is to give some understanding of the KF algorithm. It first describes the continuous state space model in a stochastic form and as a discrete time model. Some information about the uncertainties of the IM model is also included. The estimation of non-measured states such as rotor flux and the modelling of the algorithm are presented.

For sensorless IM, the speed of the motor must be estimated and not measured. Therefore, the EKF algorithm is introduced and discussed in Chapter 5. This chapter describes the approach of estimating the rotor flux, which is measurable but normally very difficult, and the speed of the IM simultaneously. The results using the Trial and Error tuning method are presented.

In Chapter 6, a new approach is proposed for replacing the Trial and Error method. The proposal of an optimisation technique of EKF is using SA algorithm for a low performance drive. The work is illustrated through simulation implementation in the V/f control and the technique is compared to the Trial and Error method.

In Chapter 7, optimisation of the EKF used in VC drive is presented. The capability of the proposed method for a high performance drive is given and verified. This chapter is divided into three parts in which part one presents the vector controller itself, part two presents the results of the observer without in the system, and part three presents the observer directly used in the system.

Chapter 8 gives a brief review of GA theory and its application to optimisation of the EKF. At the beginning of the chapter, some background material on GA is provided. This is followed by the presentation of the generic GA. Then the major components of GA and the implementation on the IM drive are discussed. To assess the feasibility of the proposed

method, the simulation results of the proposed method, as presented in Chapter 6 and Chapter 7, are compared with the method in Chapter 8. The dynamic performance of each method is compared, examined and discussed.

Finally, Chapter 9 gives a general discussion of the topic, presents the conclusions of the work included in this thesis and the recommendation for further work.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

One of the most significant developments in the control of electrical machines is the use of sensorless techniques. Sensorless estimation of speed is required particularly in applications where a speed sensor may not be used due to the hostile environment or to increase the reliability of the system. Sensorless drives have been extensively applied to electrical machines especially IMs and have completely replaced the classical IM drives equipped with a speed sensor in some applications. The advantages of speed sensorless IM drives are reduced hardware complexity and lower cost, reduced size of the drive machine, elimination of the sensor cable, better noise immunity, increased reliability, and less maintenance requirements [1]. In this chapter, some of the work that has already been carried out by researchers is discussed.

The chapter is divided into two main sections. The first section covers the two main classes of sensorless IM drives: the low performance and high performance control drives. Low performance drives are usually based on scalar control, particularly constant V/f, while high-performance drives are most often based on the VC technique. The second section covers the observer problem as well as development approaches by other researchers. This section gives the review which yields the main work of this thesis.

2.2 Constant V/f Control

The operation of IMs based on the scalar control mode has been known for many decades. It represents the large majority of electric motor variable speed drive systems due to their simple, reliable, and inexpensive structure. Scalar control allows the steady state speed or torque of the motor to be controlled while maintaining the magnetic field constant at desired level by adjusting the magnitude and frequency of either the voltage or current supplied to the stator. The most common scalar control technique is referred to as constant V/f. This type of control is based on the magnitude of the stator voltage being adjusted in proportion to the frequency to maintain an approximately constant stator flux in the motor. With the changes of frequency, the ratio V/f remains constant and the stator flux remains constant too. The ratio of V/f is usually based on the rated values of these variables. At low

levels of supply frequency, when the stator voltage becomes very low, the voltage dropped across the stator resistance becomes a larger portion of the whole and this requires an increase in the voltage often known as voltage boost. Despite its advantages, constant V/f is not suitable for high dynamic performance or where the full torque control capabilities at low speed are required [2]. Its practical application at low frequency is challenging, due to the influence of the stator resistance and the necessary rotor slip to produce torque [3]. The V/f control is illustrated in Figure 2.1 [4].

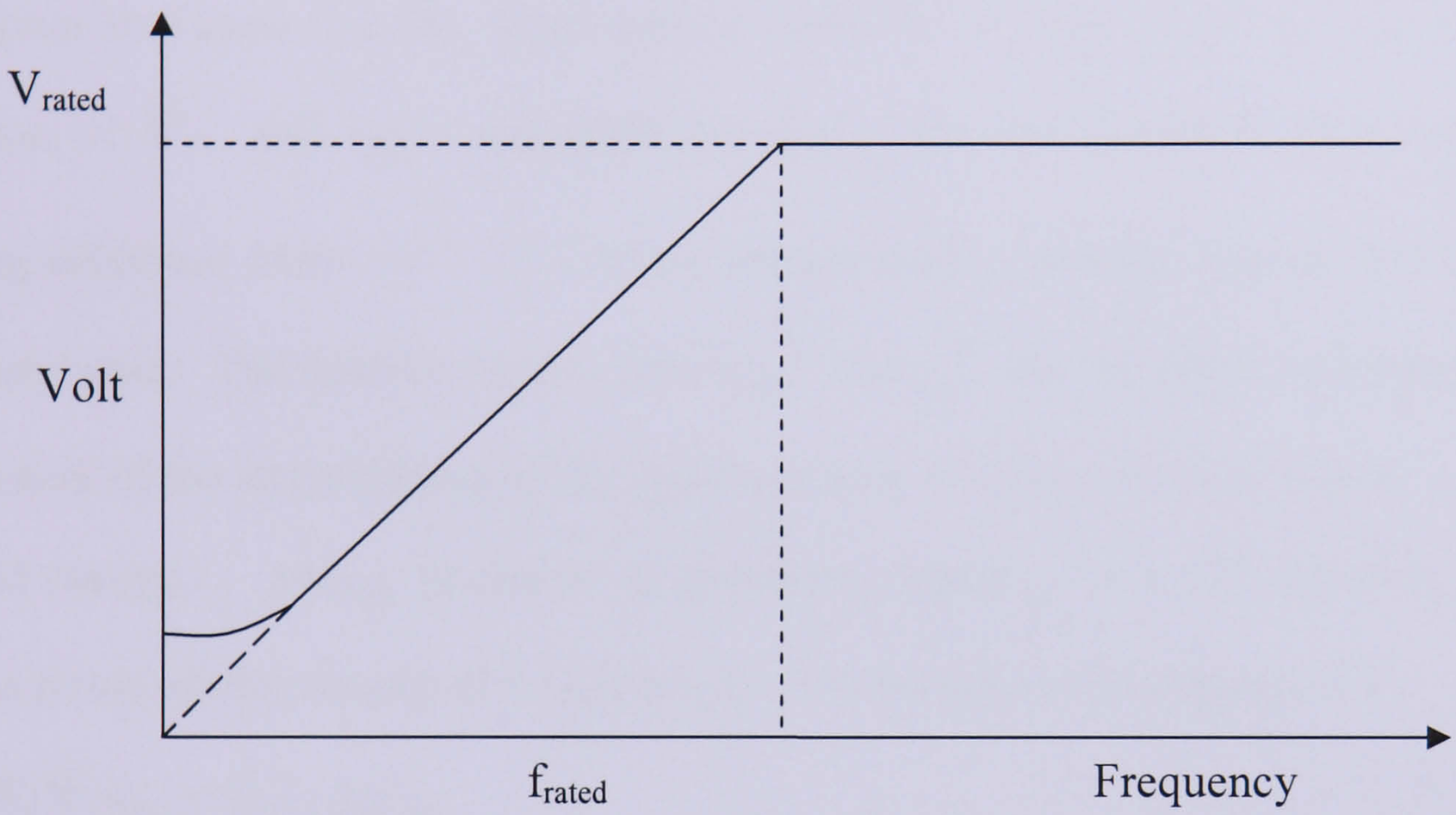


Figure 2.1: Voltage versus frequency curve under V/f control [4]

2.3 Vector Control

Field oriented control or VC has been strongly adopted in electrical drives. VC attempts to develop a torque proportional to the stator current so that the IM dynamics behave similarly to the DC machine and the control effort can be reduced. It offers better dynamic performance because accurate control of motor torque allows the construction of the high performance motor drive systems and also, the desired relation between the rotor flux and the stator current vectors can be assured by feeding a torque reference into the vector controller. This results in the motor torque being proportional to the torque reference while flux magnitude is kept at the desired level.

2.3.1 DC Drive Correlation

VC can be known as decoupling, orthogonal or transvector control. It is considered to offer ease of control due to the separation of torque and flux control. In a DC machine, the developed torque is given by:

$$T_e = K_t \Psi_f \Psi_a = K_t' I_a I_f \tag{2.0}$$

The I_f sets up magnetic conditions to give optimum flux level while I_a regulates the torque. Figure 2.2 (a) shows the field flux Ψ_f produced by the current I_f is perpendicular to the armature flux Ψ_a , which is produced by the armature current I_a. When torque is controlled by controlling the current I_a, the flux Ψ_f is not affected. Because of the decoupling scheme, whenever the current I_f is controlled, it only affects Ψ_f but not Ψ_a. As shown in Figure 2.2 (b), ideal control with VC is possible if i_{ds} is aligned in the direction of $\overline{\Psi}_r$ and i_{qs} is perpendicular to it. The IM control is in a synchronously rotating reference frame ($d^e - q^e$) where the sinusoidal variables appear as DC quantities in steady state. The control current inputs i_{ds}^* and i_{qs}^* are the direct and quadrature axes component of the stator current in the synchronously rotating reference frame. i_{ds} is similar to field current I_f and i_{qs} is similar to armature current I_a in a DC machine. The space vectors rotate synchronously at frequency ω_e. The torque can be expressed as:

$$T_e = K_t \overline{\Psi}_r i_{qs} = K_t L_m i_{ds} i_{qs} \tag{2.1}$$

When i_{ds}^* is controlled, it controls the flux only and does not affect the i_{qs} component of current and when i_{qs}^* is controlled, it affects the actual i_{qs} only but does not affect $\overline{\Psi}_r$.

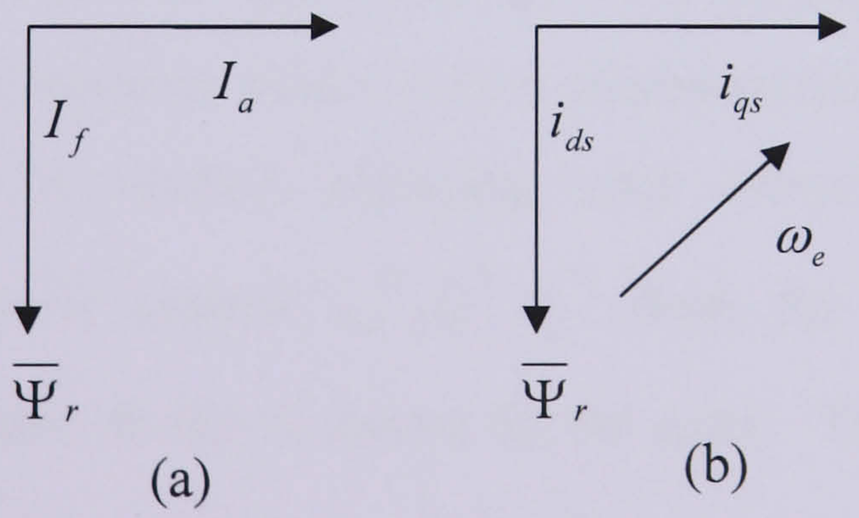
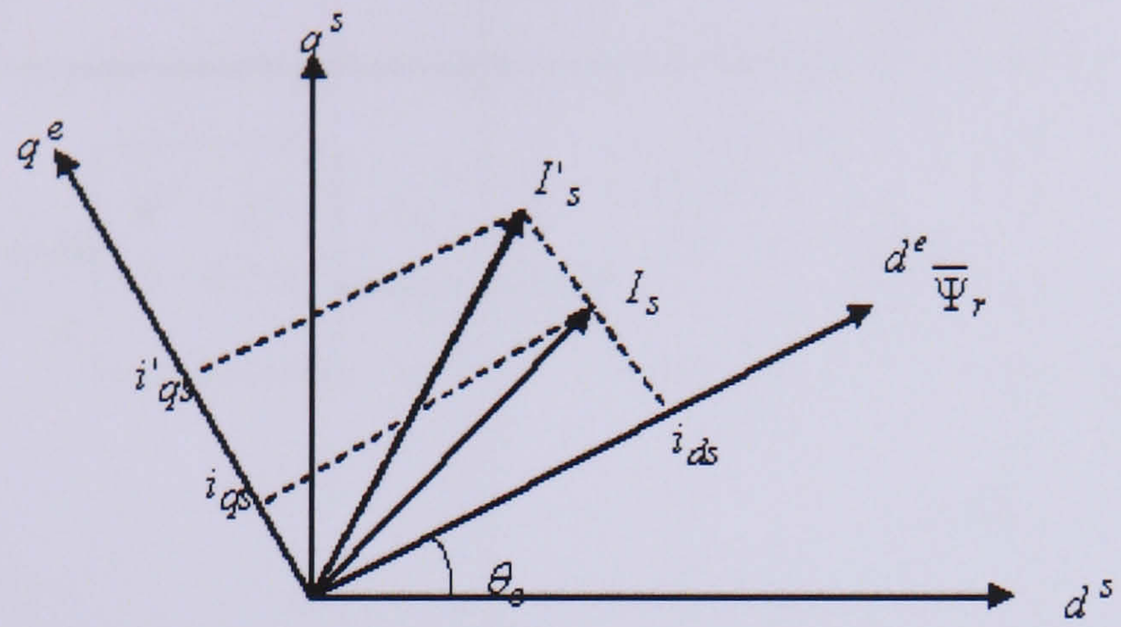


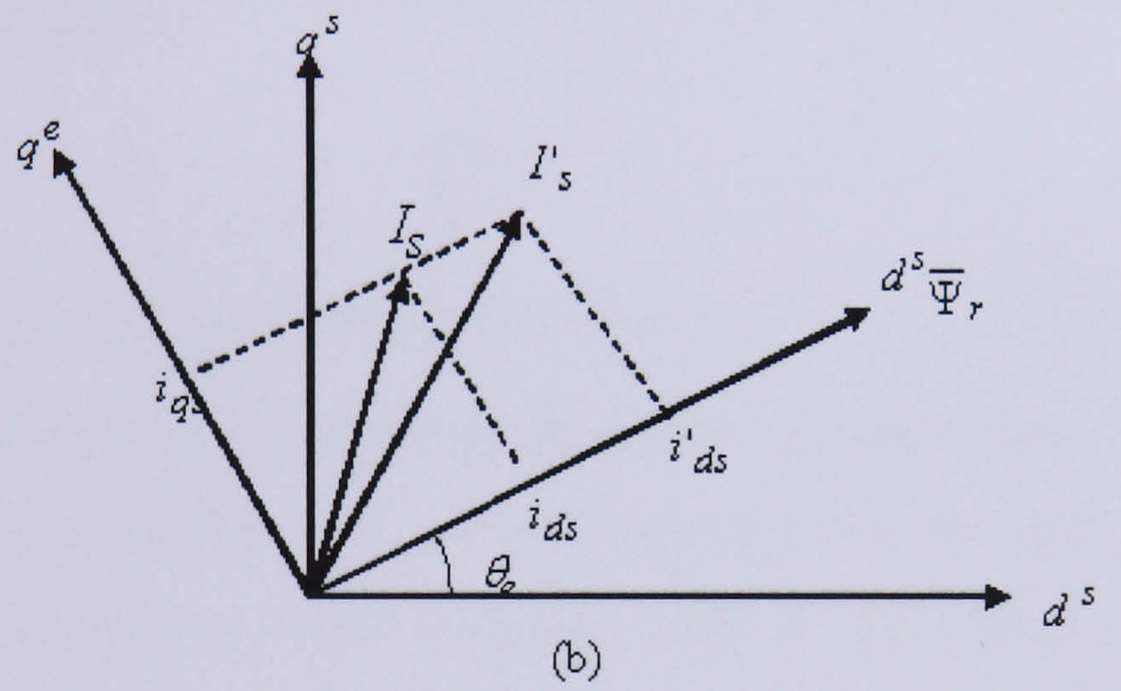
Figure 2.2: (a) Separately excited DC motor (b) VC induction motor

The torque producing current effect is illustrated in Figure 2.3 (a); the torque component is increased by increasing the i_{qs} component of stator current while holding the

flux constant. The flux producing current variation causes the flux level to increase by increasing the i_{ds} current component and this scenario is shown in Figure 2.3 (b).



(a)



(b)

Figure 2.3: (a) Torque Producing Current. (b) Flux Producing Current

2.3.2 Principles of Vector Control

The principle of VC is that the excitation reference frame axes are oriented so that the d^e axis is aligned with the rotor flux vector $\overline{\Psi}_r$. The concepts of VC can be explained using Figure 2.4 where the machine model is in a synchronously rotating reference frame. In the figure the inverter is omitted, assuming unity current gain where it generates current i_a, i_b, i_c by the control current i_a^*, i_b^*, i_c^* from the controller. The internal transformation in the machine model is shown on the right. The 3-phase current i_a, i_b, i_c are converted to $i^s{}_{ds}, i^s{}_{qs}$ by a 3-phase to 2 phase transformation. These currents are then converted to the synchronously rotating frame by unit vector components, $\cos\theta_e$ and $\sin\theta_e$ before applying them to the machine model as shown. The controller makes two stages of transformation, so that the control currents $i_{ds}{}^*, i_{qs}{}^*$ correspond to the

machine current i_{ds}, i_{qs}. In addition the unit vector assures correct alignment of i_{ds} current with $\overline{\Psi}_r$ and i_{qs} perpendicular to it.

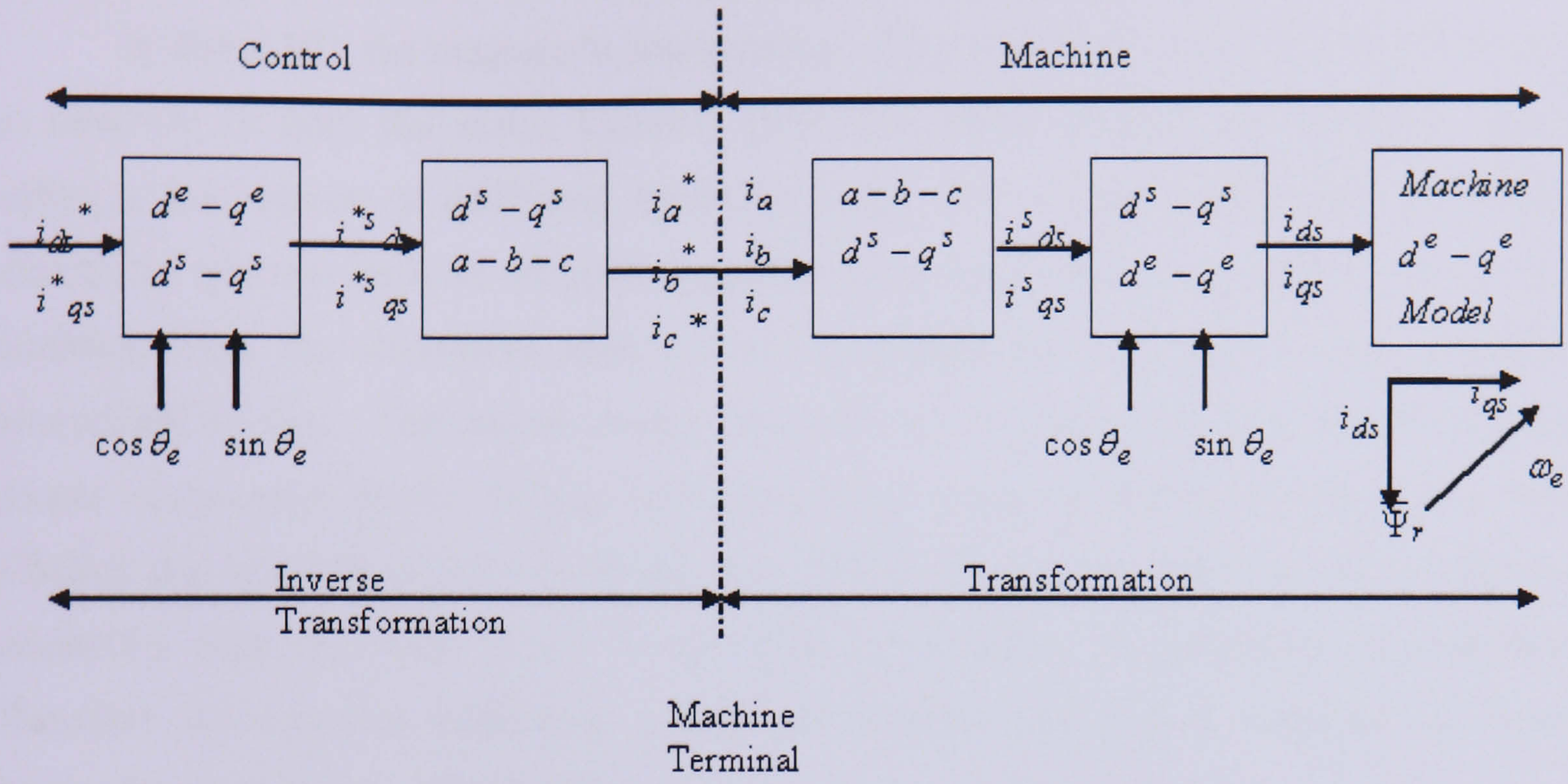


Figure 2.4: VC implementation principle with machine $d^e - q^e$ model [5]

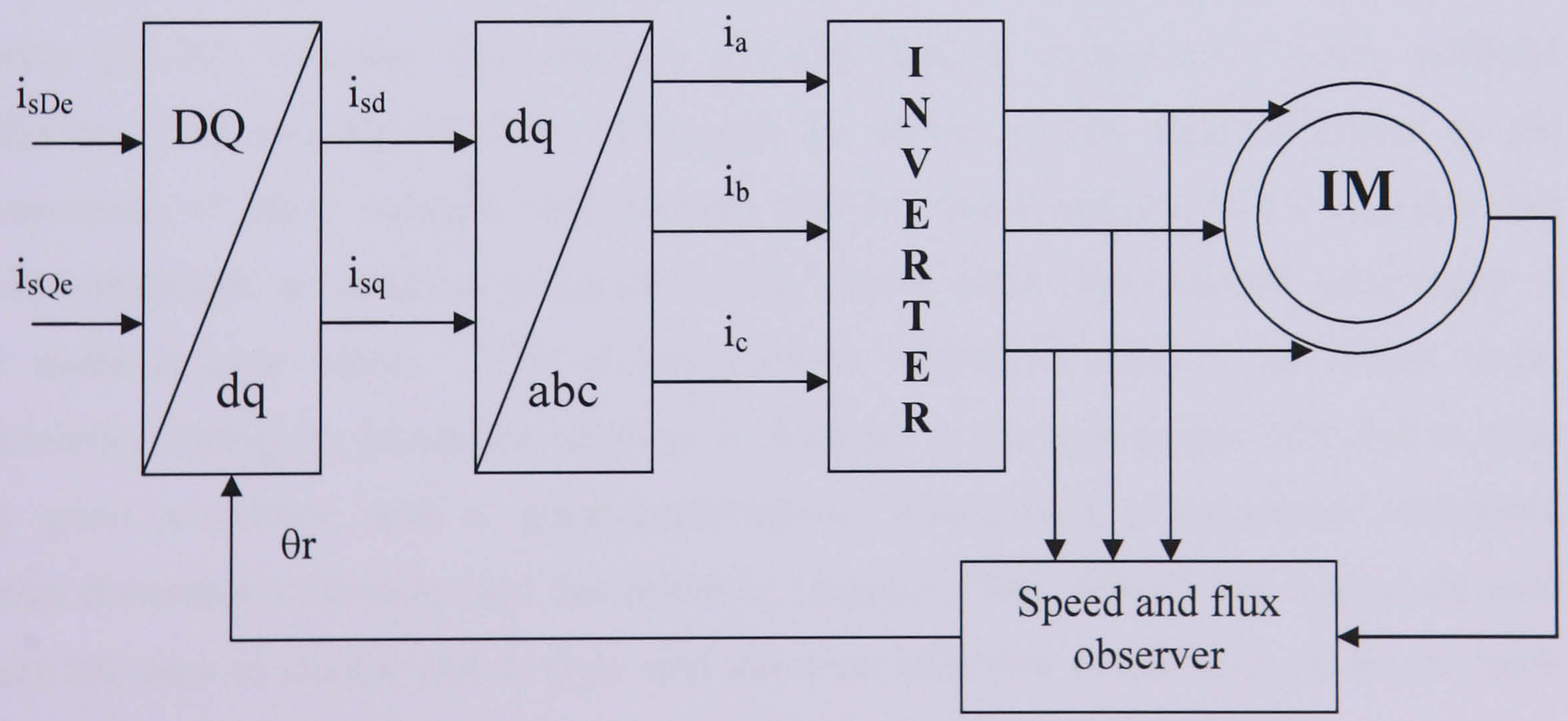
The basic concept and industrial applications of a sensorless vector controlled for an inverter fed IM is given in [6]. VC can be classified into the direct or feedback method (using direct measurements to obtain variables such as rotor flux) or the indirect or feed forward methods (including slip frequency controlled) with variants based on orientation of rotor flux or stator flux. The direct and indirect forms of VC are shown in Figure 2.5. These two methods are the two most commonly used techniques for high performance in IM, although indirect VC seems to be more popular in industrial applications [7]. As compared to direct VC, indirect VC does not require flux and torque estimators [5].

In indirect VC, the slip angular speed is used to obtain the position of the rotor flux vector and to orient it. Indirect VC has the drawback of being dependent on the motor temperature and the level of magnetic excitation of the motor [8] and requires knowledge of the rotor time constant, a parameter which can vary widely during the drive's operation [9]. Therefore much research has been done to overcome this problem including the application of broadband excitation technique for standstill identification of the IM rotor dynamic and static inductance with regard to dependencies on saturation [10]. Another method is by using an adaptation mechanism as proposed in [11] to tune the rotor time constant. In addition, a novel sliding surface with an integral component for the variable structure speed controller is designed. Using the proposed variable structure speed controller, it is said that

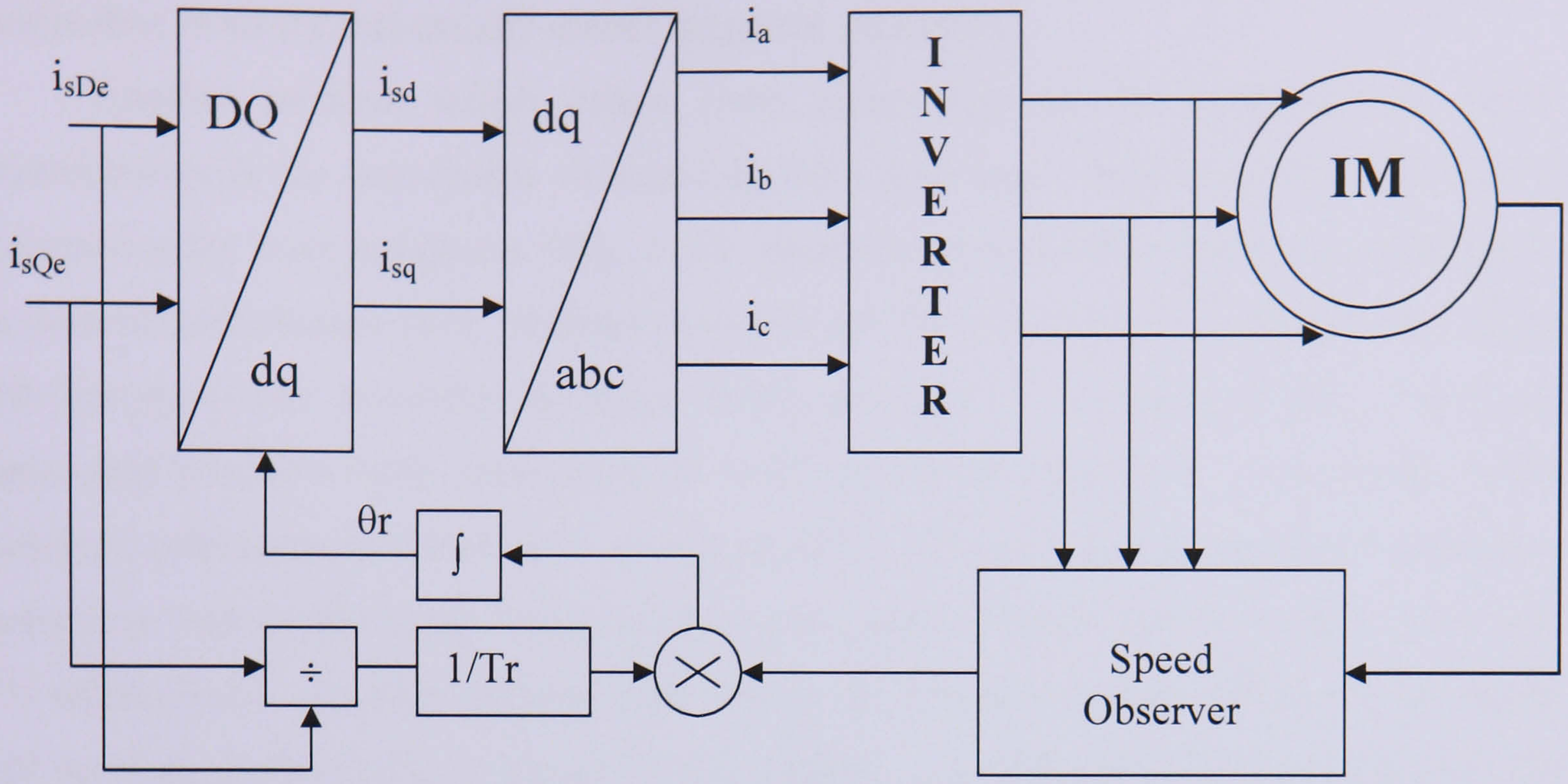
the controller is insensitive to parameter uncertainty and load disturbance. An adaptive mechanism based on an adaptive inverse model controller and stator current minimization controller is also proposed [12] to improve indirect VC in a heavy load condition.

In direct VC, the magnitude and position of the rotor flux vector is calculated using an observer or from the motor terminal quantities, stator voltages and currents without adding a flux sensor or additional hardware [13]. The voltage model used in the flux calculation process involves an open integration problem, which is generally solved by a feedback loop. An alternative approach but developed in [14] was a fuzzy controller determined by GA. The author claims that this new system provides a robust structure, simple design, and shorter settling time than other integration methods and is also more efficient and has a shorter development time. Methods also have been devised to improve sensitivity such as using GAs [15] to solve the problem of parameter identification. Therefore accurate flux estimation in high performance IM drive is important to ensure proper drive operation and stability.

The fundamental concepts of VC regarding the direct and indirect variants have been given. However, only direct VC will be modelled, implemented and considered later in this thesis.



(a)



(b)

Figure 2.5: (a) Basic direct VC system. (b) Basic indirect VC system

2.4 The Observer Problem and the Development Approach

Many different schemes have been proposed to estimate the speed of sensorless IMs. The speed estimation can be divided based on two groups: (a) Fundamental components of stator currents and voltages and (b) high frequency components which are injected into stator voltages or currents. Fundamental component methods include the sliding mode observer [16-20], adaptive flux observer [21-24], MRAS estimation [25-30], artificial intelligence [31] and KF [32,33]. Although the speed can be obtained based on the measurement of stator voltages and currents this has been claimed to be complex and heavily dependent on machine parameters [17]. Each author has claimed advantages of their methods over others. The sliding control scheme is said to be robust under uncertainties caused by parameter errors or by changes in the load torque [17] and to offer many good properties, such as good performance, insensitivity to parameter variations, external disturbance rejection and fast dynamic response [34]. MRAS are commonly used as they are easy to design and analyze and the most effective would seem to be the rotor flux and back EMF-based MRAC methods [35]. But these methods are inaccurate at low and zero speed [35]. Neural networks have the advantages of fast parallel computation and when they have been well trained, the approach can perform the required control in a very short time while being less sensitive to the stator resistor thermal effects [36]. Injection of high frequency components into stator voltages or currents has been implemented and tested

for estimating rotor resistance [37] and offers operation down to zero speed [2]. It relies on asymmetry; in an IM this usually means magnetic saturation.

Another problem which arises when controlling the IM without rotor speed measurements is the importance of knowing the temperature, and hence the difficulty in determining the rotor resistance [38]. Such parameter uncertainties impose a challenge to the control performance [17]. Related problems are the influence of parameter deviations, with low speed and standstill operations being the most critical aspects [19]. Therefore, many improvements have been done to overcome these problems. The online stator resistance estimation has been done addressed in [39,40] and the zero speed and frequency problem as well as the regenerating mode at low speed operation are covered by [41] and [21] respectively. Another improvement relates to operation at high efficiency by taking core loss into account [42], to improving the stability of a multiple motor drive system [23] and setting aside limitation to low cost implementation [43].

There are several different methods proposed for the observer such as sliding mode [16,17,44], model reference adaptive [26,45] and adaptive observer [21,46]. Since the 1960's [47], the KF has become a very valuable tool for estimation and prediction problems. A comparative study of nonlinear state estimators applied to AC drives has been carried out in [48]. The authors describe the difference between two observers, MRAS and KF. Estimation performance based on parameter deviation effect, computational burden, low speed operation, coefficients tuning sensitivity and the pure integrator issue are covered. The KF is shown to have an advantage over the MRAS in all of the performance criteria except for the computational burden in which the MRAS algorithm required 60% less program execution time than the KF. In [49], a sliding mode observer is compared with the KF for full state estimation of the IM. The sliding mode observer is claimed to be robust to parameter uncertainty and external noise, be simpler to implement, its dynamic performance can be altered, and no knowledge of the noise statistics is required. The EKF, however, is shown to produce the best estimation results, although this was dependent on appropriate selection of the process and measurement noise covariance matrices, and the application involves significant numerical complexity due to the system order increasing from 4 to 5.

Several applications using the EKF for nonlinear drives systems particularly estimation have been reported from about the early 1990's and there is still motivation for such developments [50, 51]. In spite of its computational complexity, caused by the computational demands of the Ricatti Differential Equation (RDE) [52], the algorithm is

still favourable for use as an observer in the sensorless control of IMs. The KF and in particular the EKF have been used extensively with the IM for state and parameter estimation [38,53]. The EKF applies the KF to nonlinear systems by simply linearising all the nonlinear models so that the traditional linear KF equations can be applied. The reason why the KF is found in a wide of applications is because the uncertainties of the motor model can be treated as noise, and, within a reasonable range of parameter variations, the KF can therefore be assumed accurate [54]. The possibility of performing the simultaneous estimation of the states and parameters [55] also gives a great advantage to the observer. Another joint estimation which includes load torque, speed and parameter was proposed [38] and claimed to be estimated simultaneously.

To overcome the high KF computational demand, several works have been reported. The rotor flux and speed estimation has been estimated using a linear KF claimed to reduce the number of operations needed [56]. Instead of using full order, a reduced order EKF was applied [33,57,58] for reduction in computational requirements.

In recent years, the optimisation of the EKF has attracted great interest. The EKF performance directly depends on the noise covariance matrices, and so these matrices need to be properly selected. The noise covariance matrices for both the KF and EKF are often tuned via a Trial and Error method [55,58]. This method is easy to use, but time consuming. In addition, the selected value matrices may not be accurate and optimal.

In optimising the filter performance, the selection of the process and the measurement noise covariances, **Q** and **R** respectively, and the weight matrices **G** for the EKF were claimed to be efficiently chosen using a real-number code GA in the closed loop constant V/f control of an IM and a field orientation controller [59]. Tuning the matrices **Q** and R of the EKF in a speed sensorless field oriented controller using simplified GA was proposed [60]. The choice of **Q** and **R** for the EKF, based on the complete normalisation of the EKF algorithm representation for a Permanent Magnet Synchronous Motor was presented and claimed to work well for any other drive, but must be tuned with the same sampling time [61]. Optimising using adaptive tuning of the KF for **R** covariance matrix was proposed via fuzzy logic [62]. The process noise **Q** for the EKF was said to be effectively and systematically calculated by using a Taylor series expansion of the nonlinear equations around the nominal parameter values and by Monte Carlo simulations [63].

Optimisation methods such as GA and SA have been used extensively in engineering fields, including electrical engineering. The methods choose their path in a

directed search with heuristic rules inspired by evolutionary biology through the design space. They can be simple to implement, are usually stable in convergence and able to find the global optimum with a fairly good probability [64]. Both GA and SA are viable approaches to finding optimal, or near optimal solutions for large scale problems. An attractive feature of SA is that it is very easy to program and the algorithm typically has few parameters that require tuning. A serious drawback is that the standard SA can be very slow, often requiring much more cost function evaluations to converge, compared with a carefully designed and tuned GA [65]. GA and SA have been proven to solve the same tuning problem in [66]. For some problems, SA appears to work better than GA [67,68].

SA in particular has been widely used in large combinatorial optimization problems such as parameter optimisation [69-71], controller optimisation [72, 73] and filter tuning [74] in various areas. The automated tuning process of the EKF by the application of GA has been proposed before [59]. This leads to the idea of using SA as an alternative approach for the tuning of the EKF. The drawback of applying SA, as already mentioned, is that it requires long computation time [75]. Combining SA with the EKF, which both require computation time, may make the system much slower. However, tuning times may be reduced by using this technique as compared to a Trial and Error method.

2.5 Conclusion

This chapter gives a review of two common types of control for drives using sensorless speed estimation for IMs, constant V/f and VC. First, the fundamental principles of these controllers have been introduced. Then, the approaches used for sensorless VC of the IM, and their associated issues have been described. The advantages as well as disadvantages of one of the observers, EKF, also have been carefully reviewed. From the investigations already carried out by researchers around the world on the EKF there is still much to be done in terms of optimisation of its performance.

CHAPTER 3: INDUCTION MOTOR MODELLING

3.1 Introduction

The IM is the most common motor in existence today. It has been around for the last century and has been used in many applications. IM features such as low cost, structure robustness, ruggedness, high reliability, high efficiency and minimum maintenance have ensured that it is an alternative in servo applications and high performance drives. A dynamic model of the motor must be known to understand and design the estimator and the drive. This chapter introduces a dynamic IM model based on the idea of space vector notation and state variables.

The chapter is divided into two parts. The first is the modelling of the motor from the space vector theory as a set of mathematical equations in state variable form and is not limited to the simulation of the continuous model. A comparison of the motor parameters is presented. The second part addresses the performance of the model, covering not only its steady state conditions but its transient behaviour as well.

3.2 Induction Motor Equivalence Circuit

In many applications, the IM usually operates in a steady state condition with constant motor speed and fixed, balanced, sinusoidal supply voltage. This can be analysed using an equivalent circuit. The equivalent circuit shown in Figure 3.1 allows the calculation of motors' basic quantities such as stator current, magnetizing current, developed torque and rotor flux linkage without considering the core loss. A method for calculations using the equivalent circuit has been done using Microsoft Excel and an example of how to use the equivalent circuit calculation at constant speed for an IM is given in Appendix A.

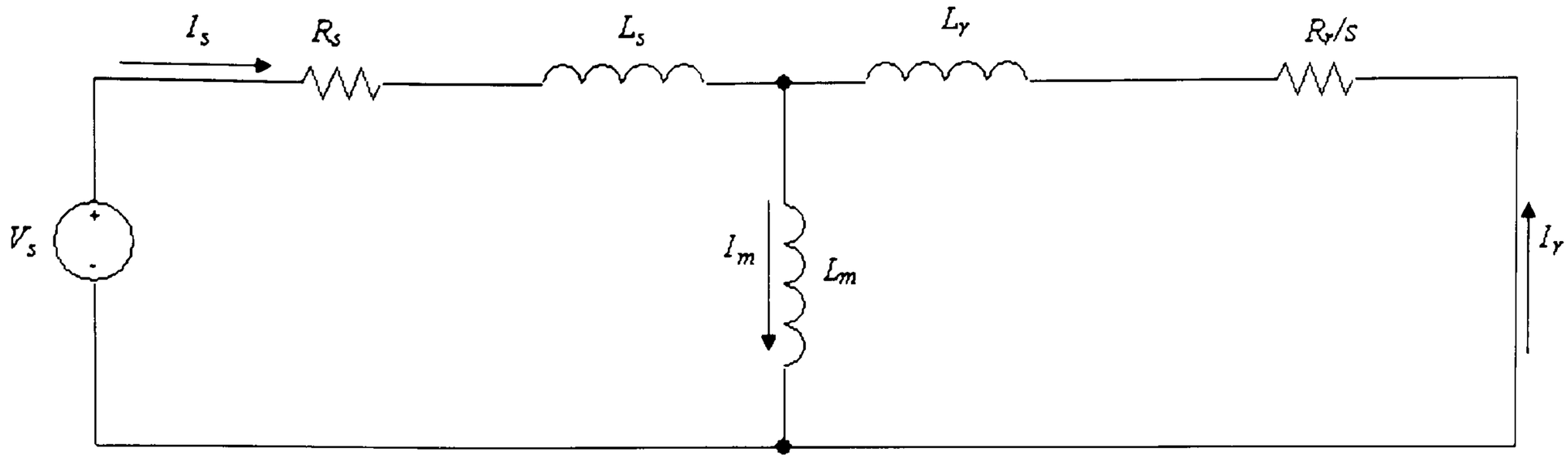


Figure 3.1: Steady State per Phase Equivalent Circuit

However for higher performance drives, it is useful to have a more general model which is not limited to a constant speed and fixed supply voltage. Such a model can be obtained by using space vector theory or two phase theory.R_r/s

3.3 Dynamic Modelling using Space Vector Notation

Park's transform, also called the dq transformation, allows a three phase machine to be represented as a two phase dq dynamic model. The dq IM model has been used for a long time, especially for transient studies, to facilitate motor analysis and reduce computation time [76]. The physical transformation of the three windings of the IM for the stator vector to just two windings is shown in Figure 3.2 [4]. The three phase winding consists of three coils each distributed symmetrically. Each coil ideally produces a sinusoidal distributed magnetomotive force (mmf). It is represented by a vector oriented with the direction of peak mmf and having a magnitude proportional to the peak mmf. The resultant stator mmf vector constitutes a vector sum of the mmfs and this can be represented by a single $\bar{I}_s$ vector where

$$\bar{I}_s = \frac{2}{3}(i_a + i_b + i_c) \tag{3.1}$$

$\bar{I}_s$ could also be produced by a two phase stator as shown in Figure 3.3. In two axes coordinates or using complex numbers, Eqn (3.1) can be written as $\bar{I}_s = i_s e^{j\alpha_s} = i_{sD} + j i_{sQ}$. In a balanced, three phase system, the phase coil is assumed to be in the vertical axes of the stator while the other phase coils are 120^0 apart from each other. Eqn (3.1) can be written as:

$$\bar{I}_s = \frac{2}{3}(i_a + a i_b + a^2 i_c) \tag{3.2}$$

where $a = e^{j2\pi/3}$ and $a^2 = e^{j4\pi/3}$

Employing Euler's identity:

$$e^{j\omega t} = \cos(\omega t) + j\sin(\omega t) \tag{3.3}$$

Therefore;

$$a = e^{j2\pi/3} = -\frac{1}{2} + j\frac{\sqrt{3}}{2}$$

$$a^2 = e^{j4\pi/3} = -\frac{1}{2} - j\frac{\sqrt{3}}{2} \tag{3.4}$$

and the expressions for the two axes currents in terms of the three phase currents are as below:

$$i_{sD} = \frac{2}{3}(i_a - \frac{1}{2}i_b - \frac{1}{2}i_c)$$

$$i_{sQ} = \frac{2}{3}(\frac{\sqrt{3}}{2}i_b - \frac{\sqrt{3}}{2}i_c) \tag{3.5}$$

It is also possible to express the three phase currents in terms of two axes currents:

$$i_a = i_{sD}$$

$$i_b = -\frac{1}{2}i_{sD} + \frac{\sqrt{3}}{2}i_{sQ}$$

$$i_c = -\frac{1}{2}i_{sD} - \frac{\sqrt{3}}{2}i_{sQ} \tag{3.6}$$

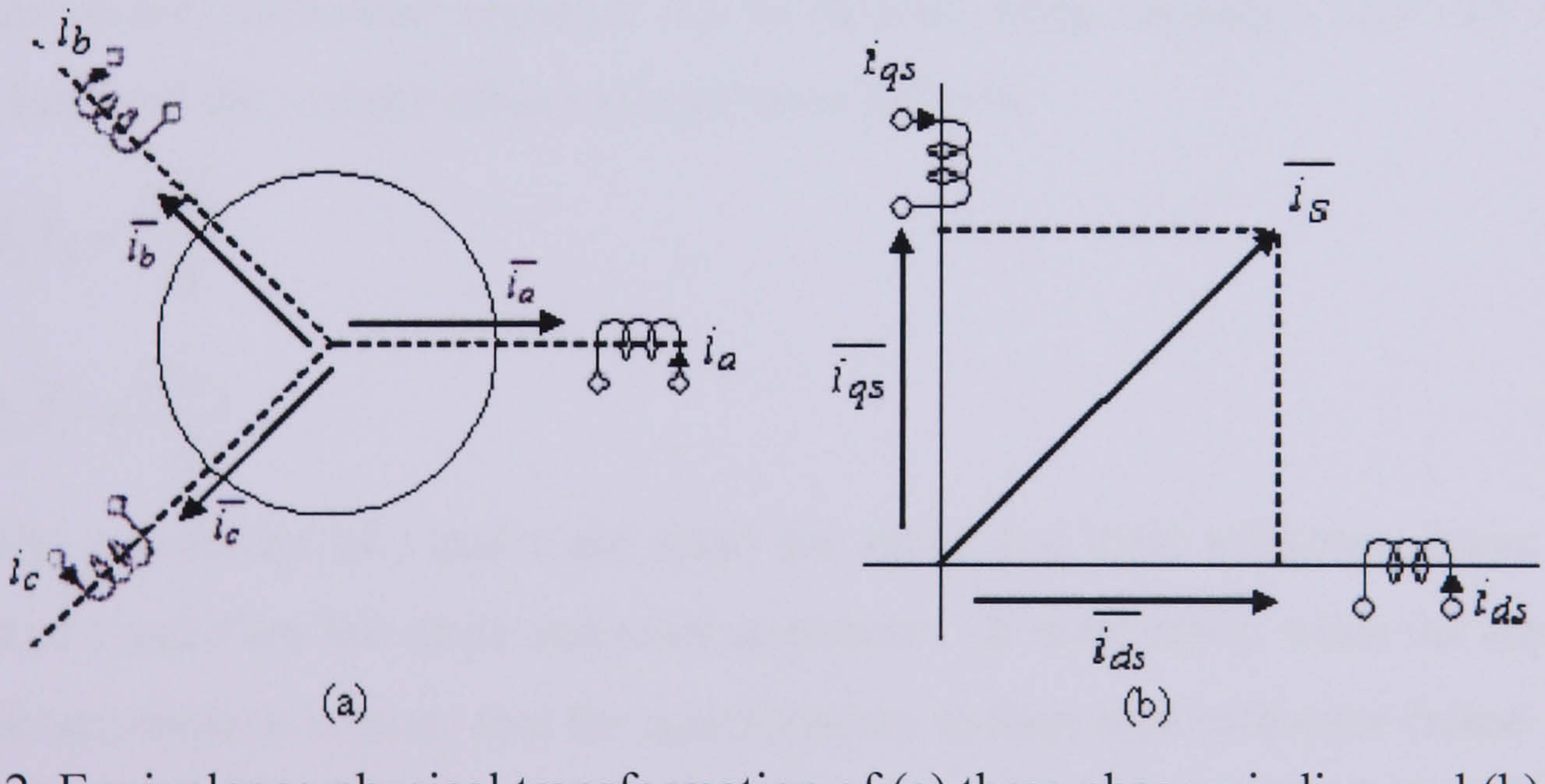


Figure 3.2: Equivalence physical transformation of (a) three phase winding and (b) two phase axes

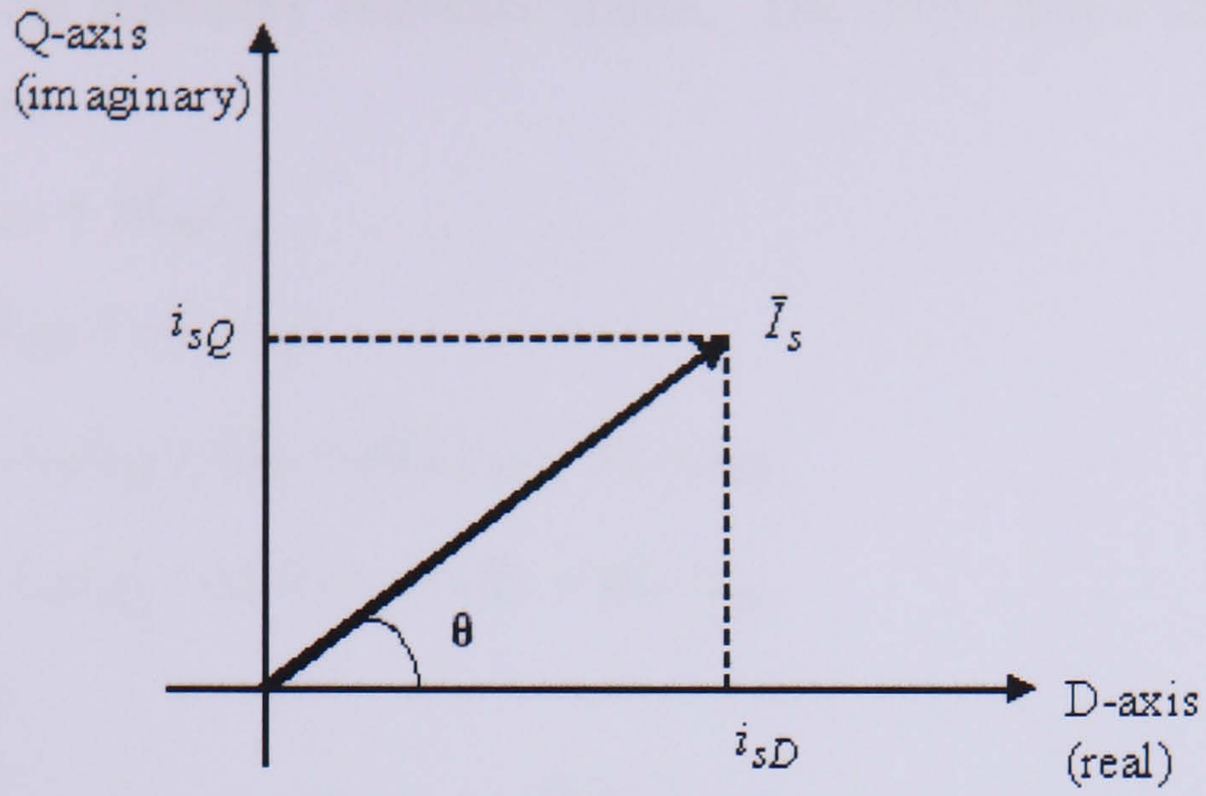


Figure 3.3: Two-axes coordinate of stator current

Similarly, using the same transformation as for the stator current, the space vector of stator voltage and flux can be written as Eqn (3.7) and (3.8) below and it is also possible to express the two phase voltage in terms of three phase vectors and vice versa.

$$\bar{u}_s = \frac{2}{3}(u_a + au_b + a^2 u_c) \tag{3.7}$$

$$\bar{\Psi}_s = \frac{2}{3}(\Psi_a + a\Psi_b + a^2 \Psi_c) \tag{3.8}$$

The modelling of a dynamic IM can be described by a set of differential equations. The space vector differential equation can be derived using Faraday's Law and Kirchoff's Voltage Law and the voltage equation is given as follows:

$$\bar{u}^s{}_s = R_s \bar{I}_s + \frac{d\bar{\Psi}_s}{dt} \tag{3.9}$$

$$\bar{u}^r{}_r = R_r \bar{I}_r + \frac{d\bar{\psi}_r}{dt} \tag{3.10}$$

The superscript of *s* and *r* are stand for stator and rotor reference frame while the subscript of *s* and *r* are the stator and rotor quantities. In most cases, when the superscript is absent the assumption is made that the quantities are in their own reference frame.

3.4 Mathematical modelling

Various machine models can be used for the purpose of estimation of the rotor speed of an IM. The equations may be expressed in the rotor reference frame or in the stationary reference frame. The selection of the reference frame has an important effect on the execution time, which is a crucial factor especially because of the computational intensity of the EKF. To speed up the execution time of the EKF algorithm [77] has suggested using a

machine model in the stationary reference frame. The stator frame equations in the twin axes are given below:

$$u_{sD} = (R_s + pL_s)i_{sD} + pL_m i_{rd}$$

$$u_{sQ} = (R_s + pL_s)ji_{sQ} + pL_m i_{rq}$$

$$u_{rd} = pL_m i_{sD} + \omega_r L_m i_{sQ} + (R_r + pL_r)i_{rd} + \omega_r L_r i_{rq}$$

$$u_{rq} = -\omega_r L_m i_{sD} + L_m i_{sQ} - \omega_r L_r i_{rd} + (R_r + pL_r)i_{rq} \qquad (3.11)$$

And in matrix form:

$$\begin{bmatrix} \bar{u}_s \\ \bar{u}'_r \end{bmatrix} = \begin{bmatrix} R_s + pL_s & pL_m \\ (p - j\omega_r)L_m & R_r + (p - j\omega_r)L_r \end{bmatrix} \begin{bmatrix} \bar{I}_s \\ \bar{I}'_r \end{bmatrix} \qquad (3.12)$$

The modification of the matrix form as in Eqn (3.12) with short circuited rotor voltage can be written as Eqn (3.13). The detail of the mathematical modelling is shown in Appendix B.

$$\begin{bmatrix} u_{sD} \\ u_{sQ} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_s + p(L_s - \frac{L_m{}^2}{L_r}) & 0 & p\frac{L_m}{L_r} & 0 \\ 0 & R_s + p(L_s - \frac{L_m{}^2}{L_r}) & 0 & p\frac{L_m}{L_r} \\ \frac{R_r L_m}{L_r} & 0 & -\frac{R_r}{L_r} & -\omega_r \\ 0 & \frac{R_r L_m}{L_r} & \omega_r & -\frac{R_r}{L_r} \end{bmatrix} \begin{bmatrix} i_{sD} \\ i_{sQ} \\ \psi_{rd} \\ \psi_{rq} \end{bmatrix} \qquad (3.13)$$

The electromagnetic torque, T_e is calculated using the equation:

$$T_e = \frac{3}{2}\frac{p}{2}\frac{L_m}{Lr}(i_{sQ}\Psi_{dr} - i_{sD}\Psi_{qr}) \qquad (3.14)$$

3.5 State Variable Form

The state space model is a convenient method for solving the estimation and control problem. Using the state space model, any system described by with higher order differential equations can be reduced to a set of first order differential equations. In addition, the internal behaviour of the system can be easily determined together with the desired input and output. Moreover, it is usually an efficient form for computer simulation. The model for a linear dynamic system in continuous time can be expressed in general form as:

$$\dot{\mathbf{X}}(t) = \mathbf{A}\mathbf{X}(t) + \mathbf{B}\mathbf{U}(t) \tag{3.15}$$

$$\mathbf{Y}(t) = \mathbf{C}\mathbf{X}(t) + \mathbf{D}\mathbf{U}(t) \tag{3.16}$$

If the state vector A and input matrix B do not vary with time then the system is called a linear time invariant (LTI) system, otherwise the model is called linear time varying (LTV) system and Eqn (3.15) can be written as:

$$\dot{\mathbf{X}}(t) = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{B}(t)\mathbf{U}(t) \tag{3.17}$$

$$\mathbf{Y}(t) = \mathbf{C}(t)\mathbf{X}(t) + \mathbf{D}(t)\mathbf{U}(t) \tag{3.18}$$

Generally, it is easier to solve a LTI than a LTV system. Eqn (3.15) and (3.17) are usually known as the state equations and can be referred to as the process model that models the transformation for the system's state while Eqn (3.16) and (3.18) are the measurement model (output) that describes the relationship between the process state and the measurement respectively. Figure 3.4 shows the state space model of a system.

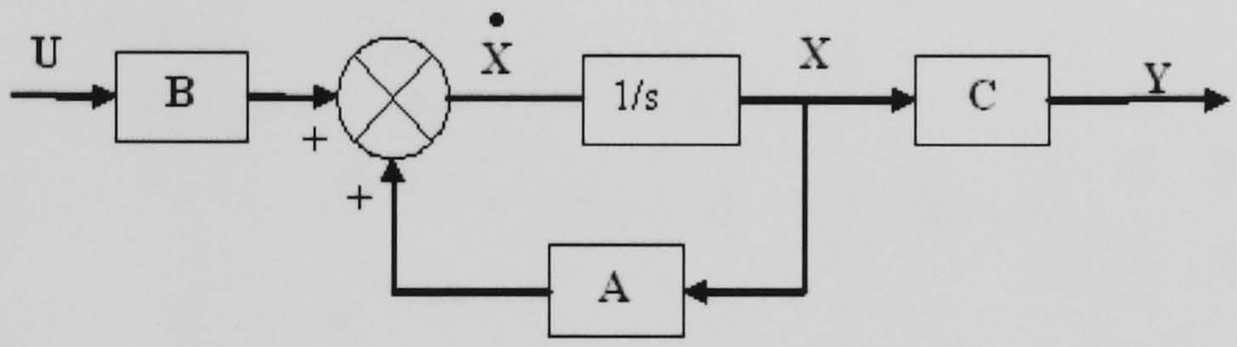


Figure 3.4: State space model

where :

A- n x n state matrix;

B-n x q input matrix;

C-p x n output matrix;

D-p x q feed forward matrix (usually zero);

X - n x 1 state vector;

U - q x 1 input vector;

Y- p x 1 output vector;

In practice, only the stator currents can be measured and compared. The rotor flux is normally not available for measurement. Using the measurement of the stator voltages and currents, the state space form can be used to estimate the rotor flux and with some modification of the equations, the rotor flux can be estimated together with the rotor speed.

Using $T_r = \frac{L_r}{R_r}, K_1 = L_s - \frac{{L_m}^2}{L_r}$ and $K_2 = R_s + \frac{{L_m}^2}{L_r T_r}$, the model (3.13) can be rewritten in state space form as:

$$\frac{d}{dt}\begin{bmatrix} i_{sD} \\ i_{sQ} \\ \psi_{rd} \\ \psi_{rq} \end{bmatrix} = \begin{bmatrix} -\frac{K_2}{K_1} & 0 & \frac{L_m}{K_1 L_r T_r} & \frac{\omega_r L_m}{K_1 L_r} \\ 0 & -\frac{K_2}{K_1} & -\frac{\omega_r L_m}{K_1 L_r} & \frac{L_m}{K_1 L_r T_r} \\ \frac{L_m}{T_r} & 0 & -\frac{1}{T_r} & -\omega_r \\ 0 & \frac{L_m}{T_r} & \omega_r & -\frac{1}{T_r} \end{bmatrix} \begin{bmatrix} i_{sD} \\ i_{sQ} \\ \psi_{rd} \\ \psi_{rq} \end{bmatrix} + \begin{bmatrix} \frac{1}{K_1} & 0 \\ 0 & \frac{1}{K_1} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_{sD} \\ u_{sQ} \end{bmatrix} \tag{3.19}$$

Recalling the state space vector/matrix notation:

$$\dot{\mathbf{X}}(t) = \mathbf{A}\mathbf{X}(t) + \mathbf{B}\mathbf{U}(t) \tag{3.15}$$

$$\mathbf{Y}(t) = \mathbf{C}\mathbf{X}(t) \tag{3.16}$$

Eqn (3.19) can be expressed as:

$$\mathbf{A} = \begin{bmatrix} -\frac{K_2}{K_1} & 0 & \frac{L_m}{K_1 L_r T_r} & \frac{\omega_r L_m}{K_1 L_r} \\ 0 & -\frac{K_2}{K_1} & -\frac{\omega_r L_m}{K_1 L_r} & \frac{L_m}{K_1 L_r T_r} \\ \frac{L_m}{T_r} & 0 & -\frac{1}{T_r} & -\omega_r \\ 0 & \frac{L_m}{T_r} & \omega_r & -\frac{1}{T_r} \end{bmatrix} \mathbf{B} = \begin{bmatrix} \frac{1}{K_1} & 0 \\ 0 & \frac{1}{K_1} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} i_{sD} & i_{sQ} & \psi_{rd} & \psi_{rq} \end{bmatrix}^T$$

$$\mathbf{U} = \begin{bmatrix} u_{sD} \\ u_{sQ} \end{bmatrix} \mathbf{Y} = \begin{bmatrix} i_{sD} \\ i_{sQ} \end{bmatrix} \tag{3.20}$$

The dynamic coefficient matrix represented by the matrix **A** in (3.20) in this case is not constant. The model is time varying due to the presence of the rotor speed which is varying in the equation. Since the system is linear, the overall system can be said to be a linear time varying system (LTV).

3.6 Dynamic Induction Motor Simulation

To make the simulation of the IM as close as possible to the real motor, a continuous time representation is used instead of a discrete time system. The simulations and all the work done in this thesis were performed on an IM with the parameters listed in Table 3.1. The squirrel cage IM is represented by its stator and rotor flux voltage equations discussed in section 3.4. Therefore using the parameter values listed in Table 3.1, the matrices of IM

parameters have been created using Simulink and the mask editor. The matrices are separated into scalar numbers.

By using Eqn (3.20), the variables $a_{11}, a_{12}, .., a_{44}$ in Figure 3.5 represent the values of row and column in the matrix **A** and $b_{11}, b_{12}, .., b_{22}$ represent the matrix **B.** The speed is multiplied with the selected scalar which is based on the equation. To create the whole matrix, values are multiplied, for instance a_{11} is multiplied by [1 0 0 0;0 0 0 0;0 0 0 0;0 0 0 0], and a_{12} by [0 1 0 0;0 0 0 0;0 0 0 0;0 0 0 0] and so on. These matrices are then added together to produce the matrix **A.** For **B**, the creation of the matrix is straight forward since there is no calculation of speed involved. The drawback of this method compared to using S-functions is the need to implement time-varying matrix operations. Although quite complicated, this method is actually preferable for those who are not keen to incorporate programming code as in an S-function. The matrices **A** and **B** are used to produce the state space model as shown in Figure 3.6. The torque calculator block is used to calculate the electromagnetic torque, T_e, using Eqn (3.14). The IM model should be capable of operating in the transient and steady state. To verify the IM model, its response is compared in the next section with the IM model from the University's Drive Library and an equivalent circuit calculation. The University Drive's Library is a consists of a set of machine and power electronic modelling blocks in Matlab / Simulink developed for Industrial and Group use. The performance of the Library has been verified by practical work.

Rated Power	7.5 kW
Rated Voltage	400 V
Number of Phase	3
Number of Poles	4
Rated Frequency	50 Hz
Rated Line Current	16 A
Rated Speed	1466 rpm (153.61 rad/s)
Synchronous Speed	1500 rpm (157.08 rad/s)
Moment of Inertia (J)	0.05 kgm^2
Rotor Type	Squirrel Cage
Stator Resistance (R_s)	0.6 Ω
Rotor Resistance (R_r)	0.4 Ω
Stator Inductance (L_s)	0.123 H
Rotor Inductance (L_r)	0.1274 H
Mutual Inductance (L_m)	0.12 H

Table 3.1: IM Ratings and Parameters

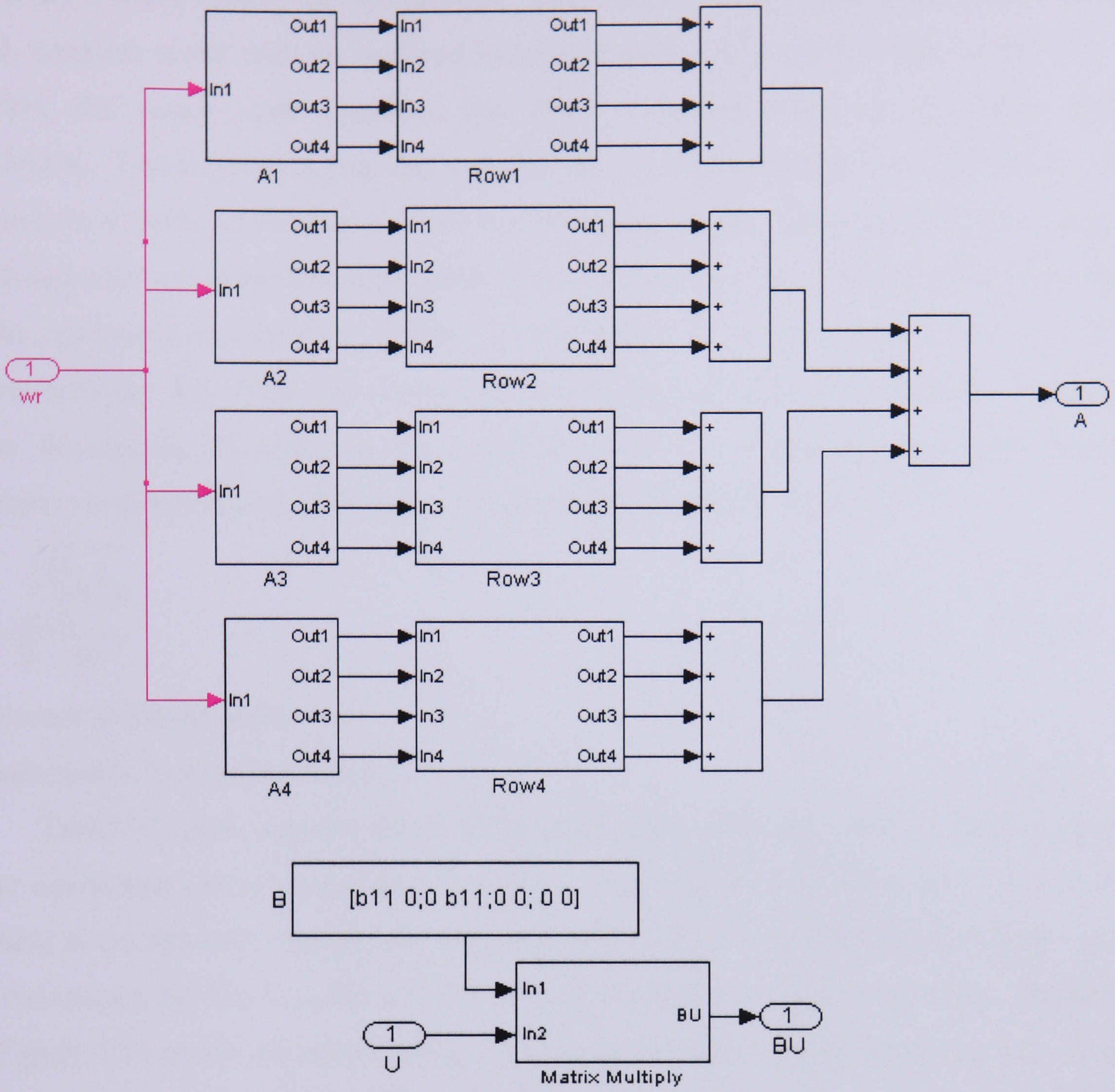


Figure 3.5: Creation of matrices A and matrices B

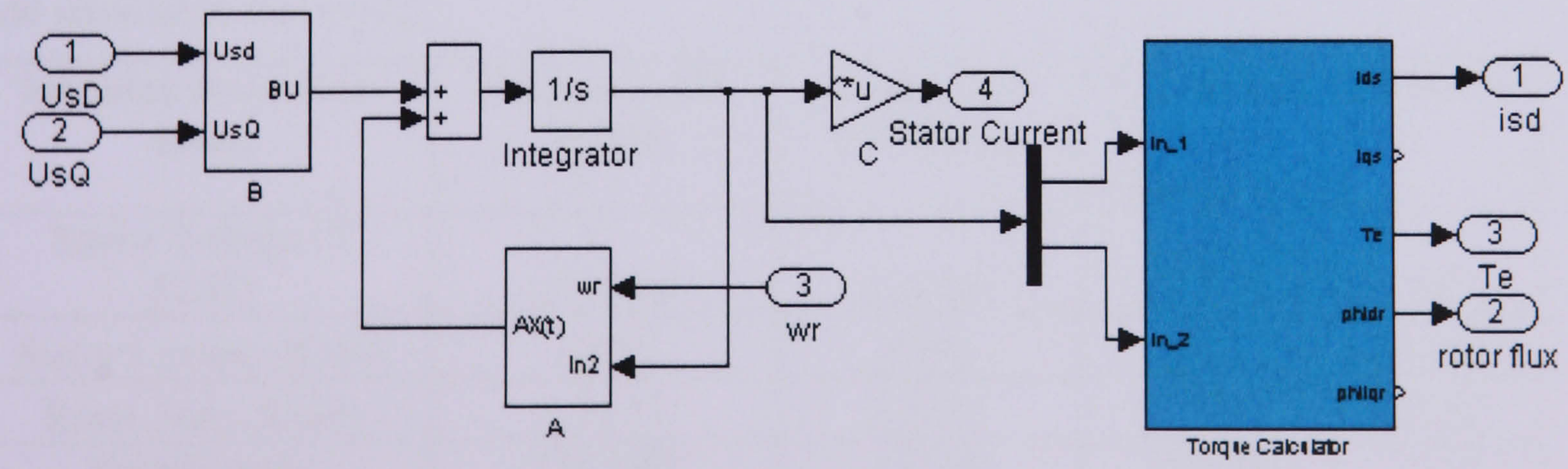


Figure 3.6: Continuous state space model of IM

3.6.1 Continuous Time IM Model Comparison

The IM comparison utilises input voltages which are produced as if directly from the supply. The continuous time IM model has been compared with the IM block from the

University's Drive Library for verification. The comparison has been made using constant speed, transient speed with no load and transient speed with a constant load of 30 Nm. In addition, the steady state operation has been compared using an equivalent circuit calculation. The simulation runs for 0.5s for the no load case and 1.0 s with load. The comparison is made on the d-axes stator current and rotor flux including torque for constant speed and rotor speed for transient speed. For an IM running at constant speed, the results for the references are shown in Figure 3.7 and the result from the model of Eqn 3.20 is shown in Figure 3.8. Note that to compare the steady state simulation with the equivalent circuit calculation, the max step size required is 0.0001s. The numerical value from the simulation is taken from the root mean squared (RMS) using the equation:

$$y_j = \sqrt{\frac{\sum_{i=1}^{M} u^2{}_{ij}}{M}} \tag{3.21}$$

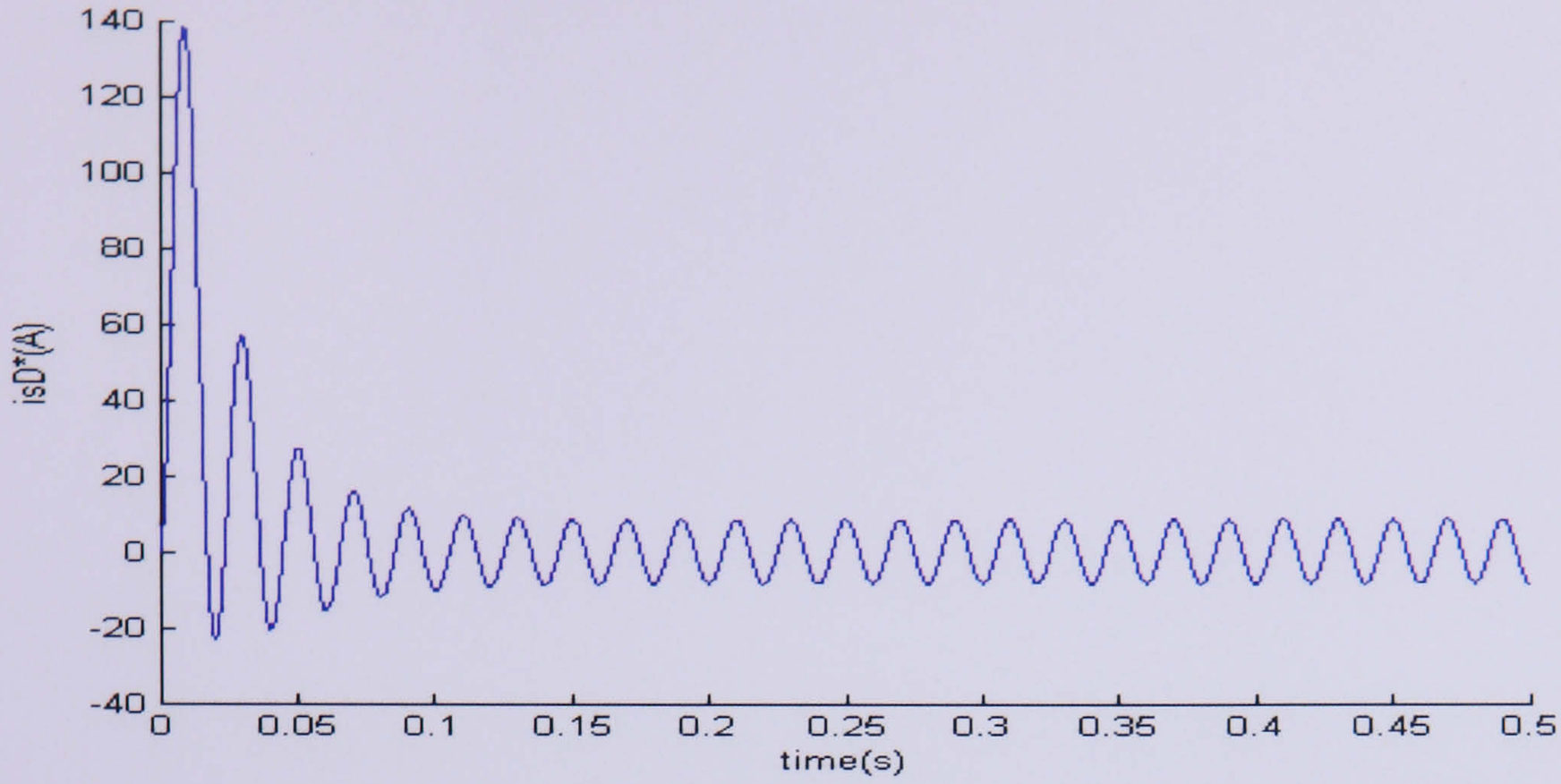
Equivalent to Matlab code:

y = sqrt(sum(u.*conj(u))/size(u,1)) (3.22)

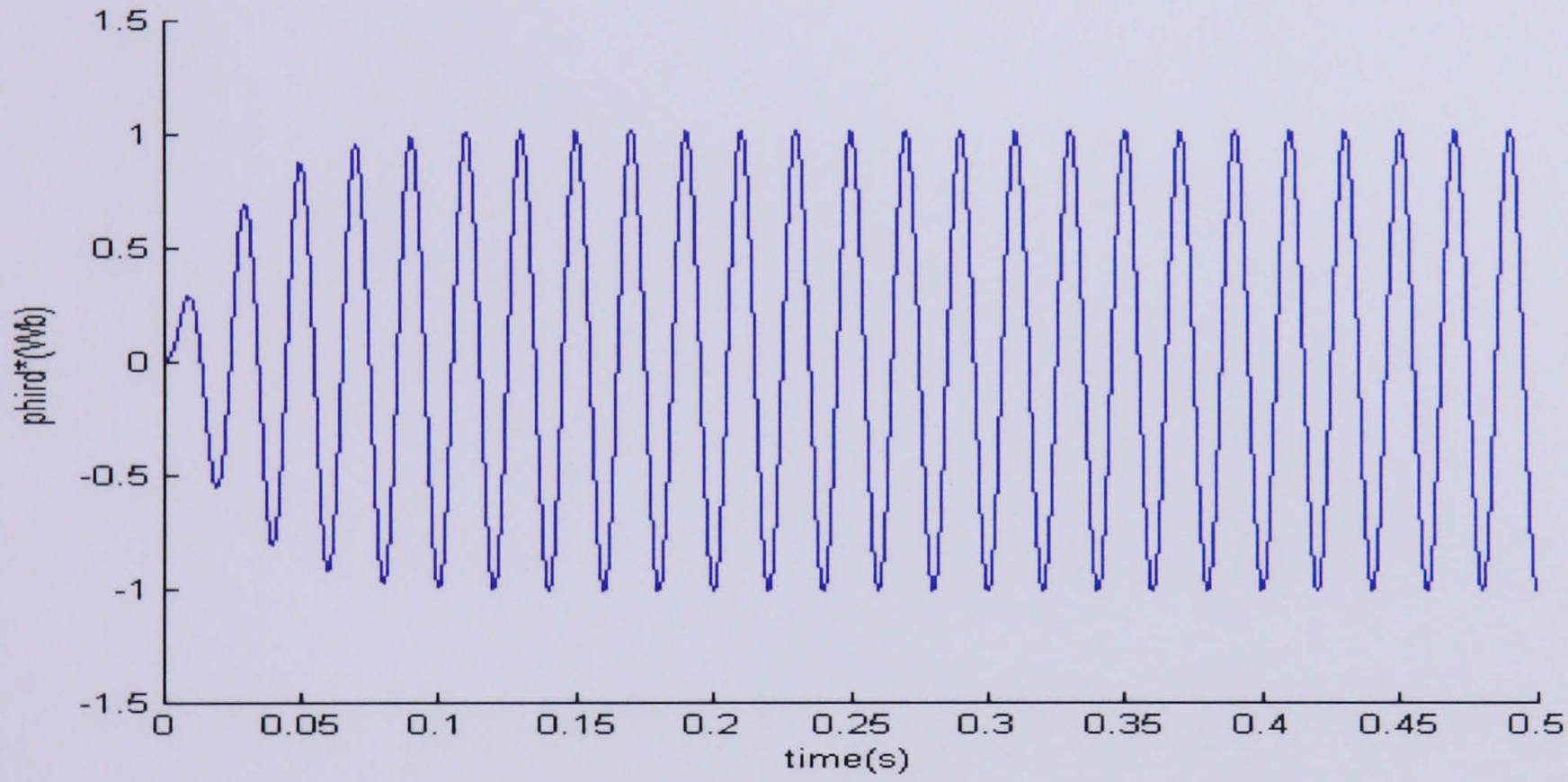
Table 3.2 shows that the errors of the simulation to 4 decimal places when compared to the equivalent circuit calculation, are zero. This suggests that the model used here is identical to the reference model, and is strong verification of the Simulink implementation. The continuous IM block works well for transients either with load or no load. Figure 3.9 and Figure 3.10 shows the results using transient speed without load while Figure 3.11 and Figure 3.12 shows the results with constant load of 30Nm. Based on these promising results, the IM block was adopted as the IM reference and the estimation process compared and referred to this model.

Machine at constant speed	**Reference IM Model**	**Continuous IM Model**	**Equivalent Circuit Calculation**	**Error %**
Stator Voltage (V RMS)	230.940	230.940	230.940	0.000
Stator Current (RMS)	5.976	5.976	5.976	0.000
Rotor Flux (RMS)	0.7171	0.7171	0.7171	0.000
Speed (rad/s)	157.080	157.08	157.080	0.000

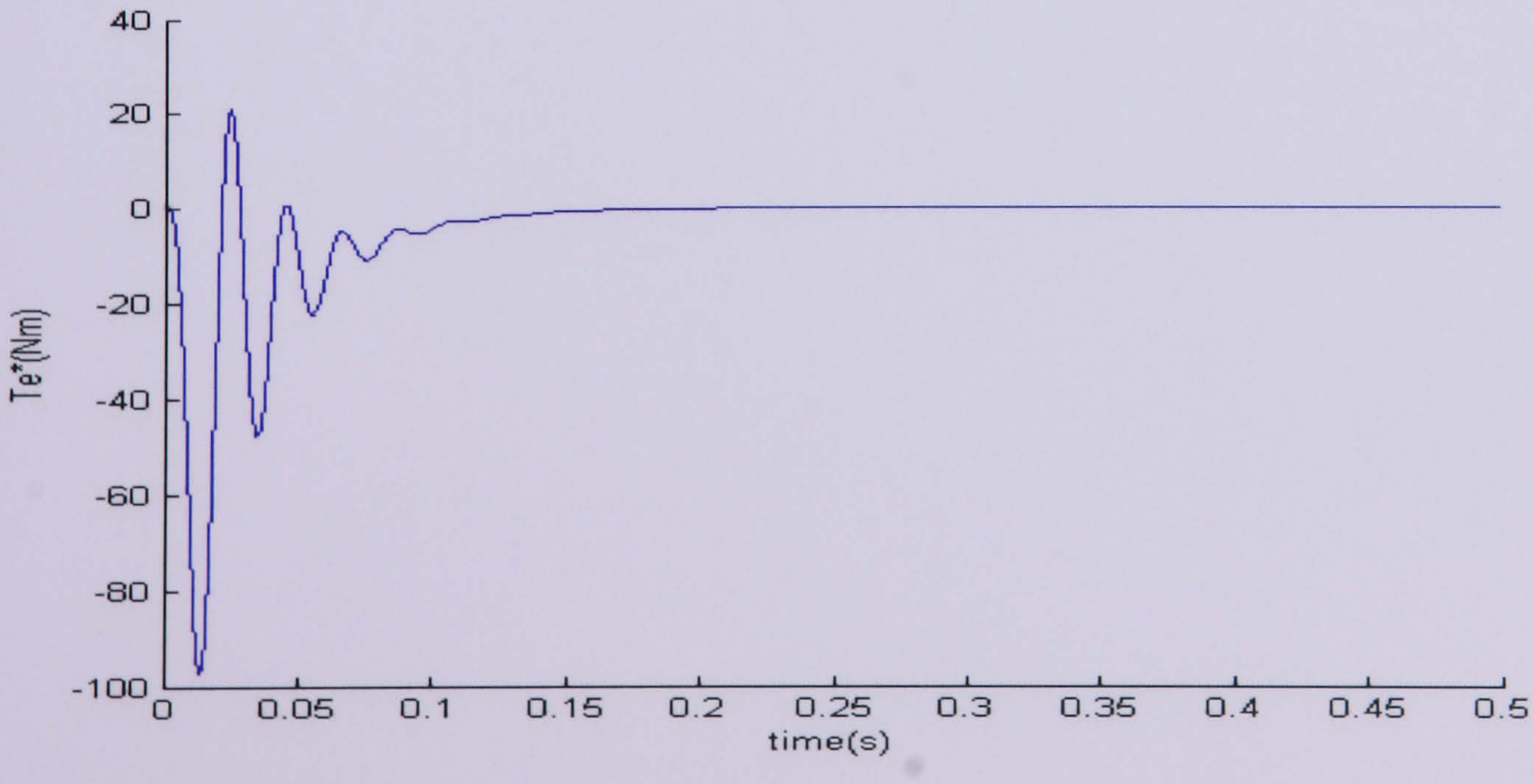
Table 3.2: Comparison result of simulation equivalent circuit calculation.



(a)

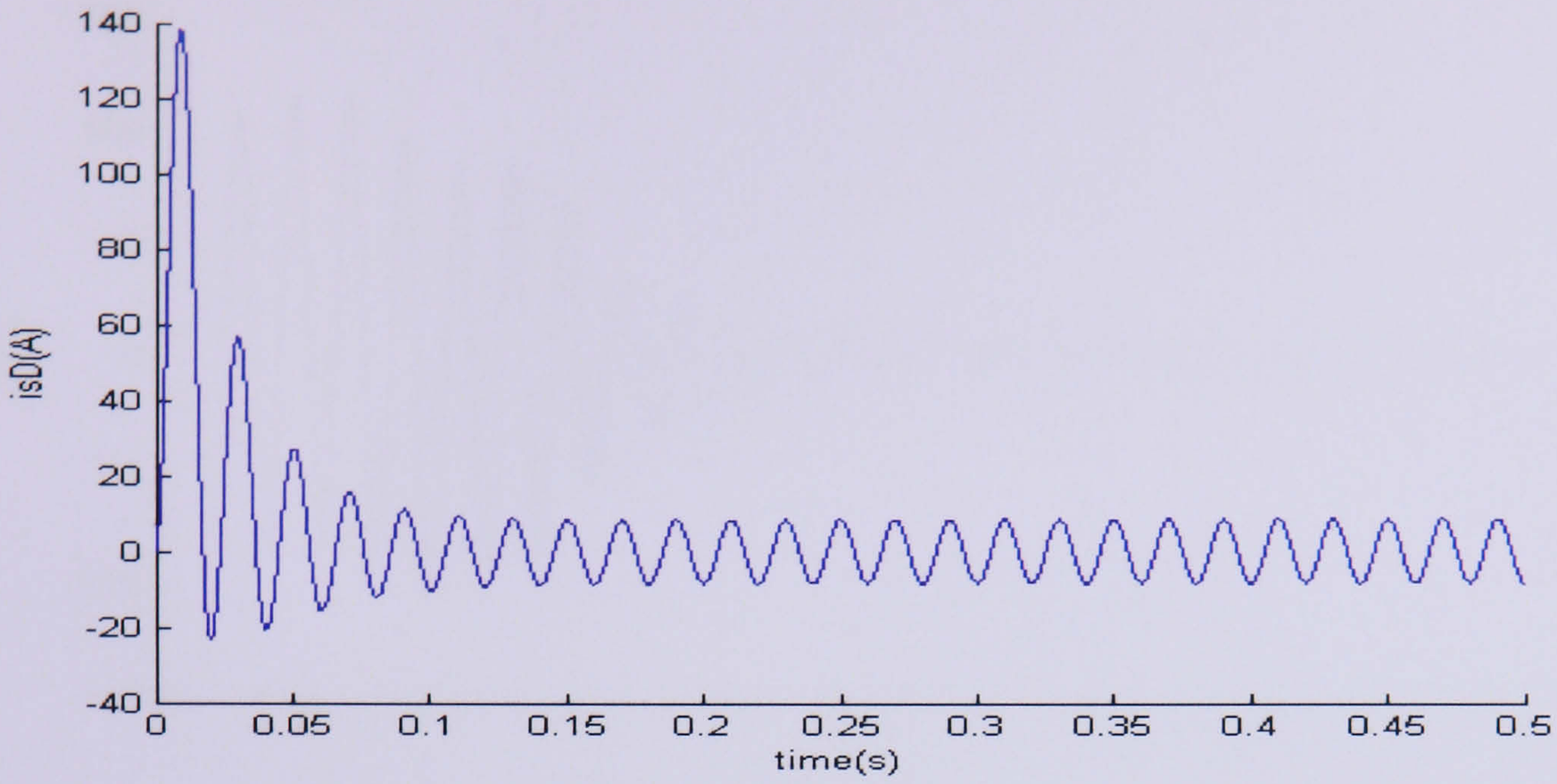


(b)

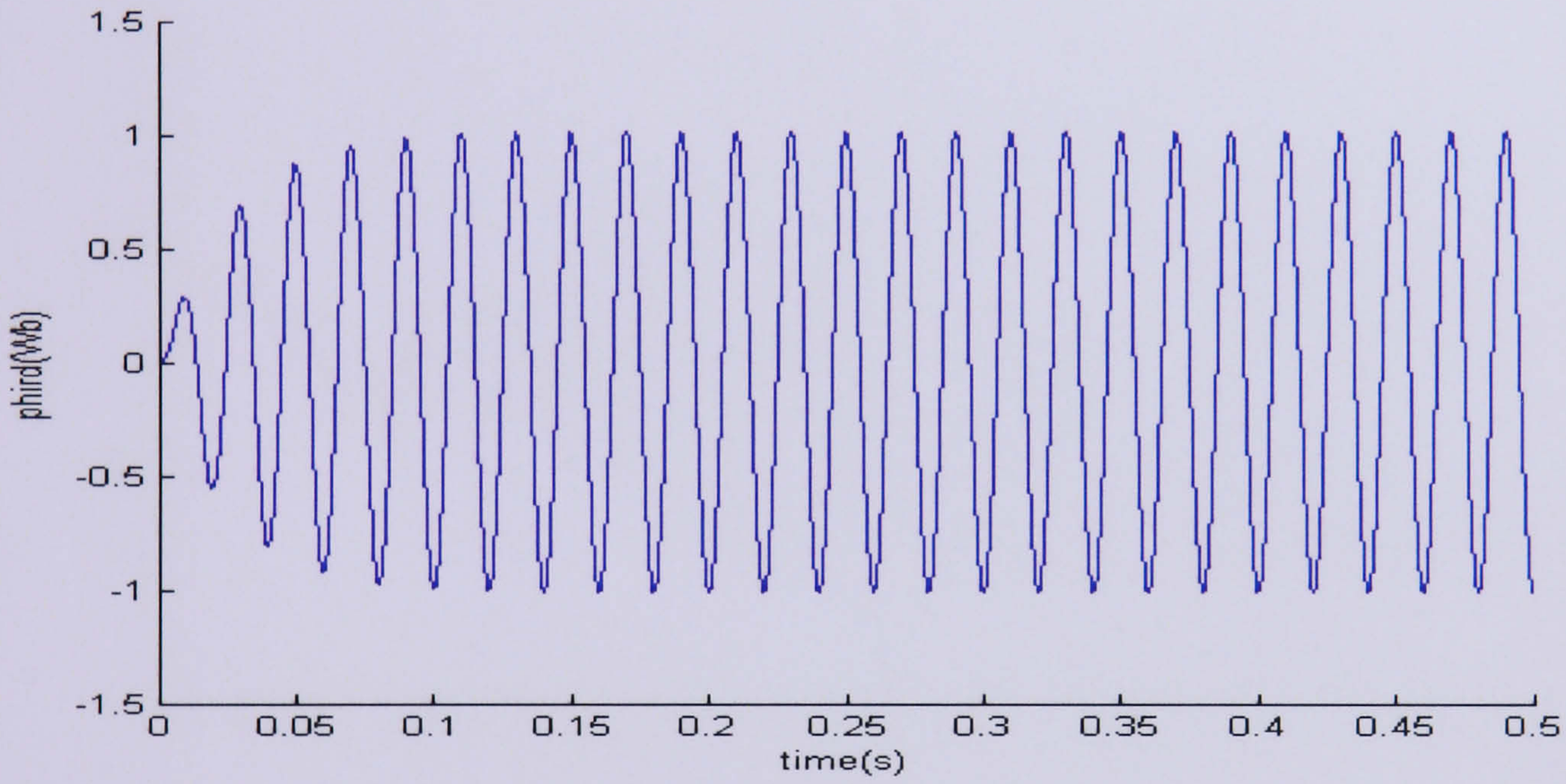


(c)

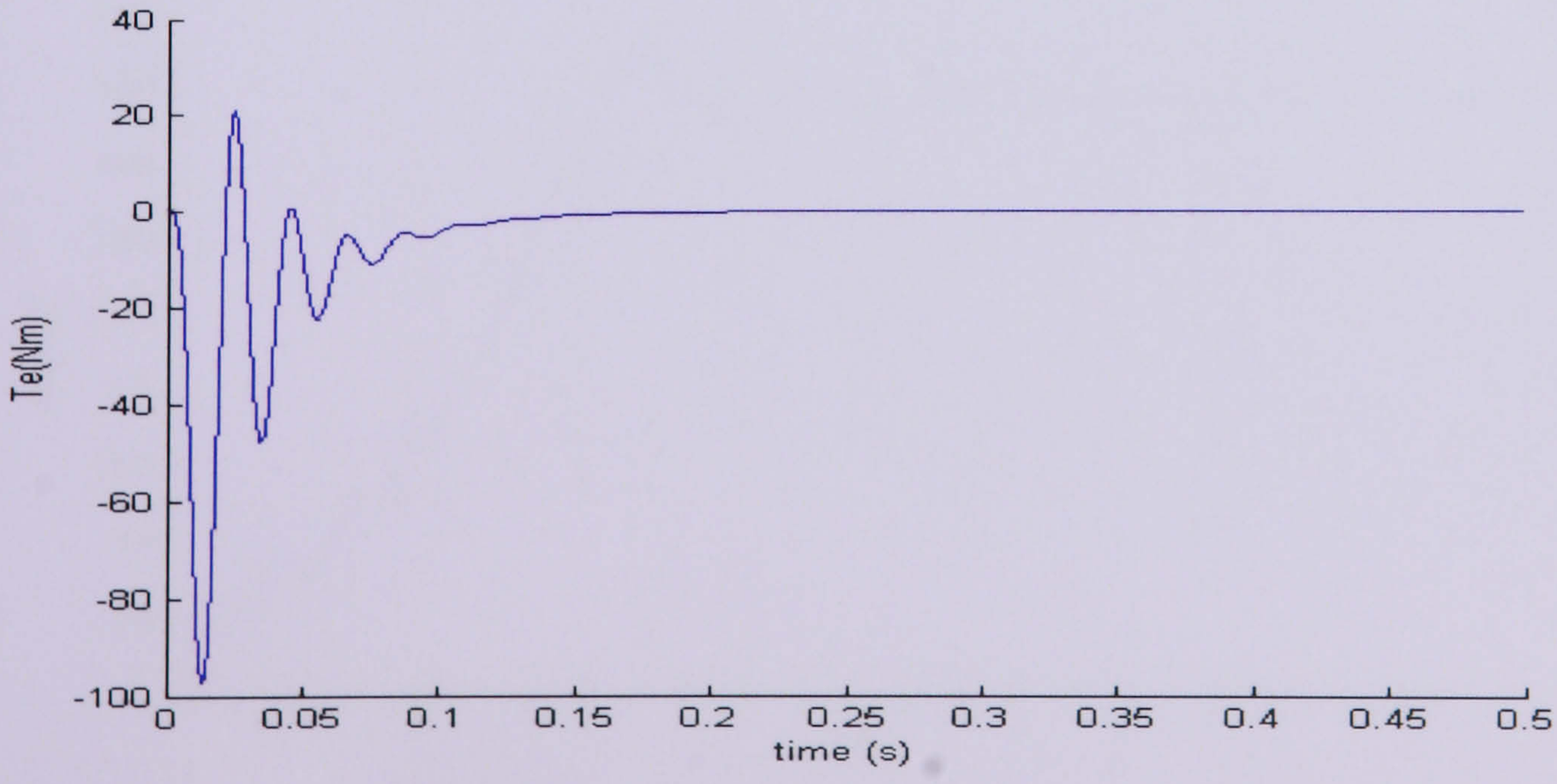
Figure 3.7: (a) Stator current, i_{sD} (b) rotor flux ψ_{dr} (c) electromagnetic torque T_e of reference motor model at constant speed



(a)

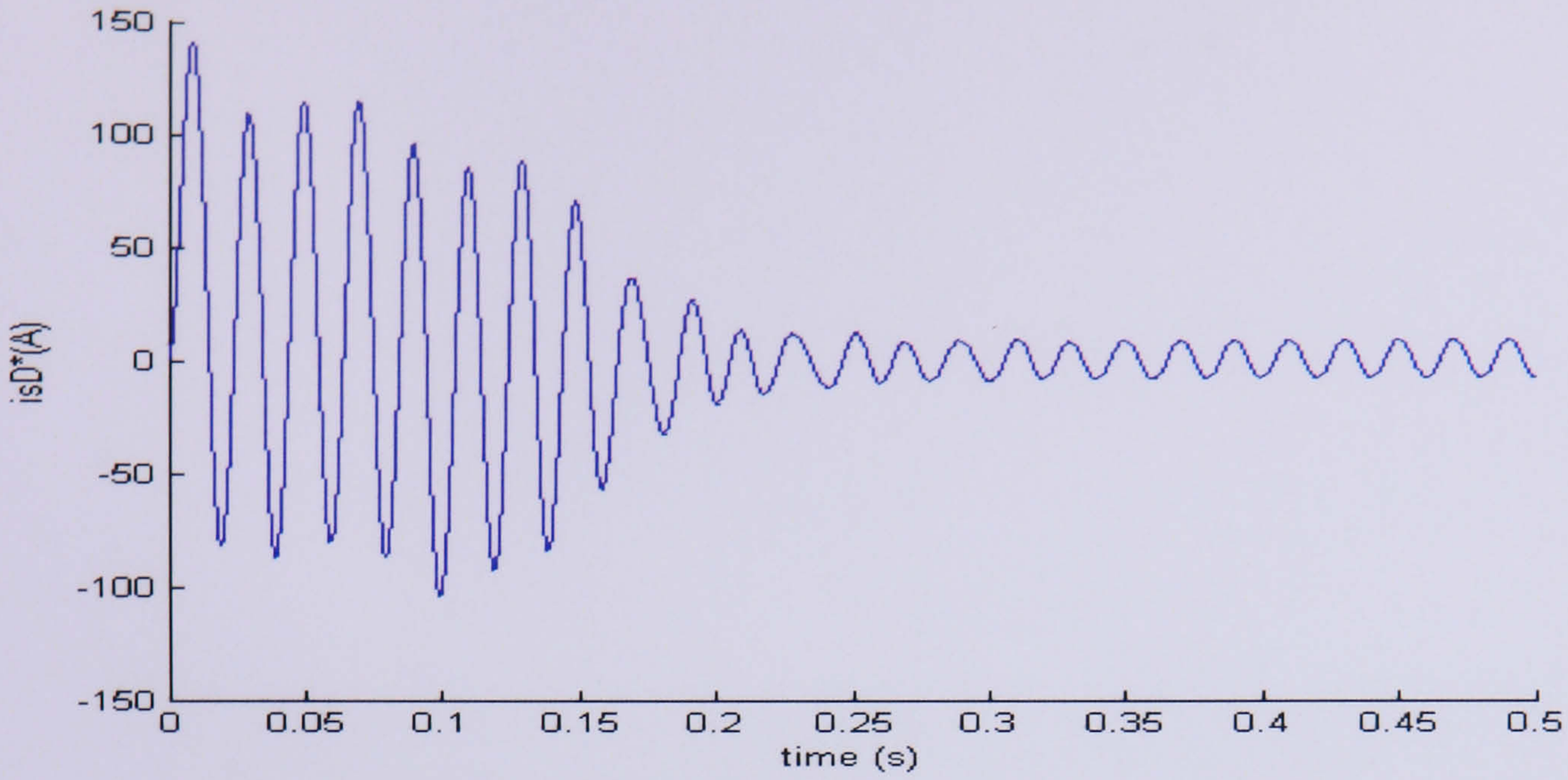


(b)

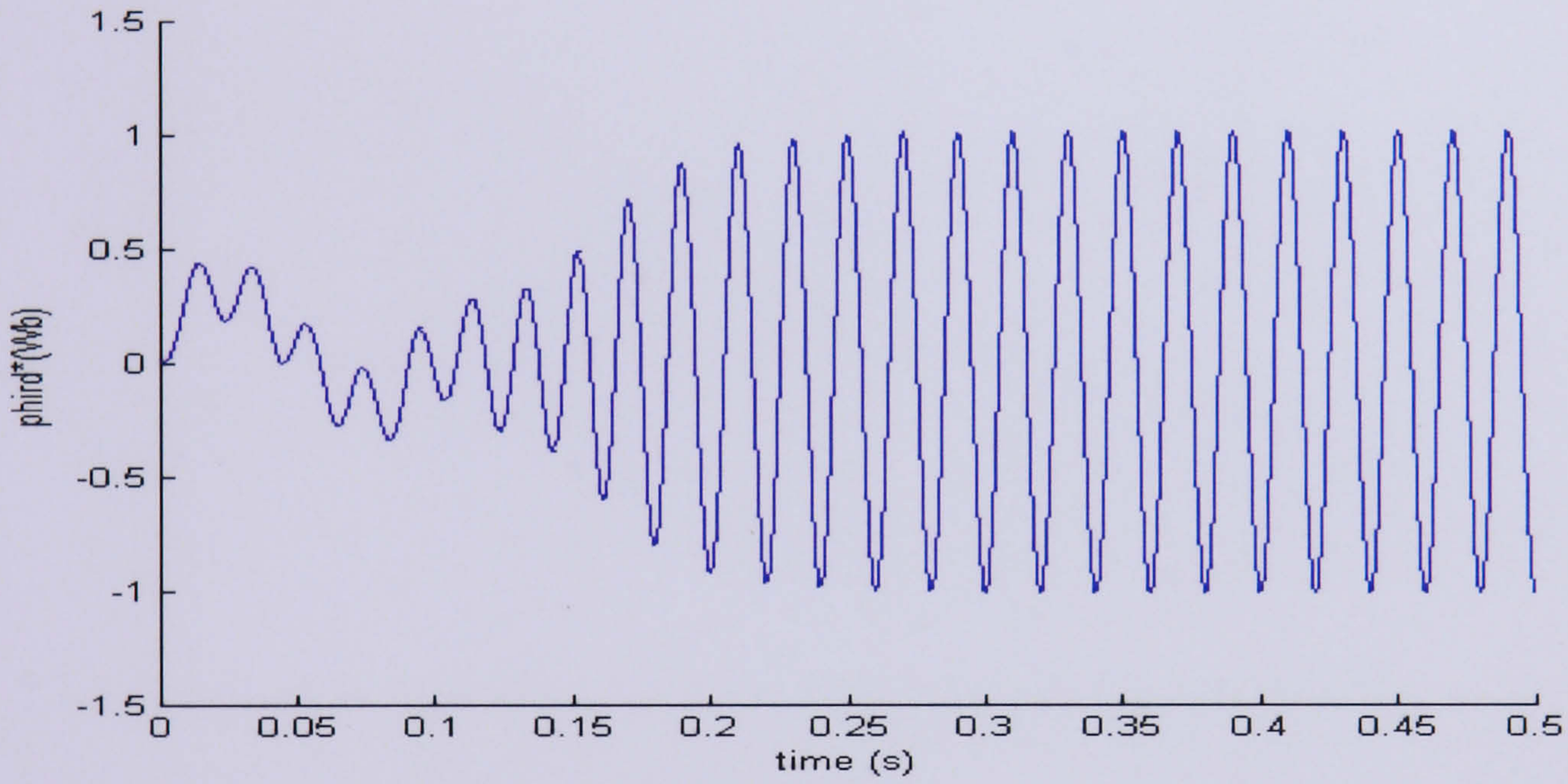


(c)

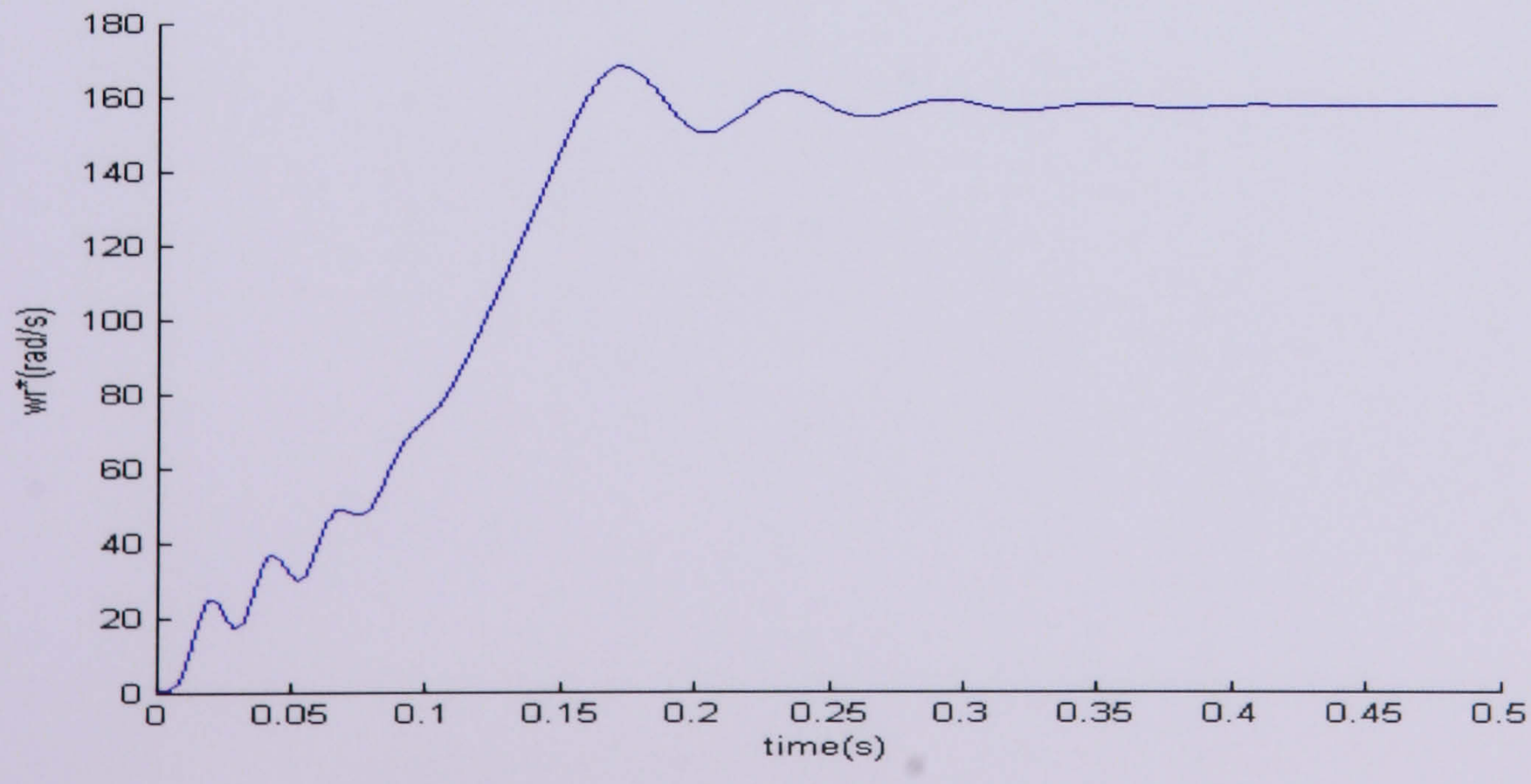
Figure 3.8: (a) Stator current i_{sD} (b) rotor flux ψ_{dr} (c) electromagnetic torque T_e of continuous-time motor model at constant speed



(a)

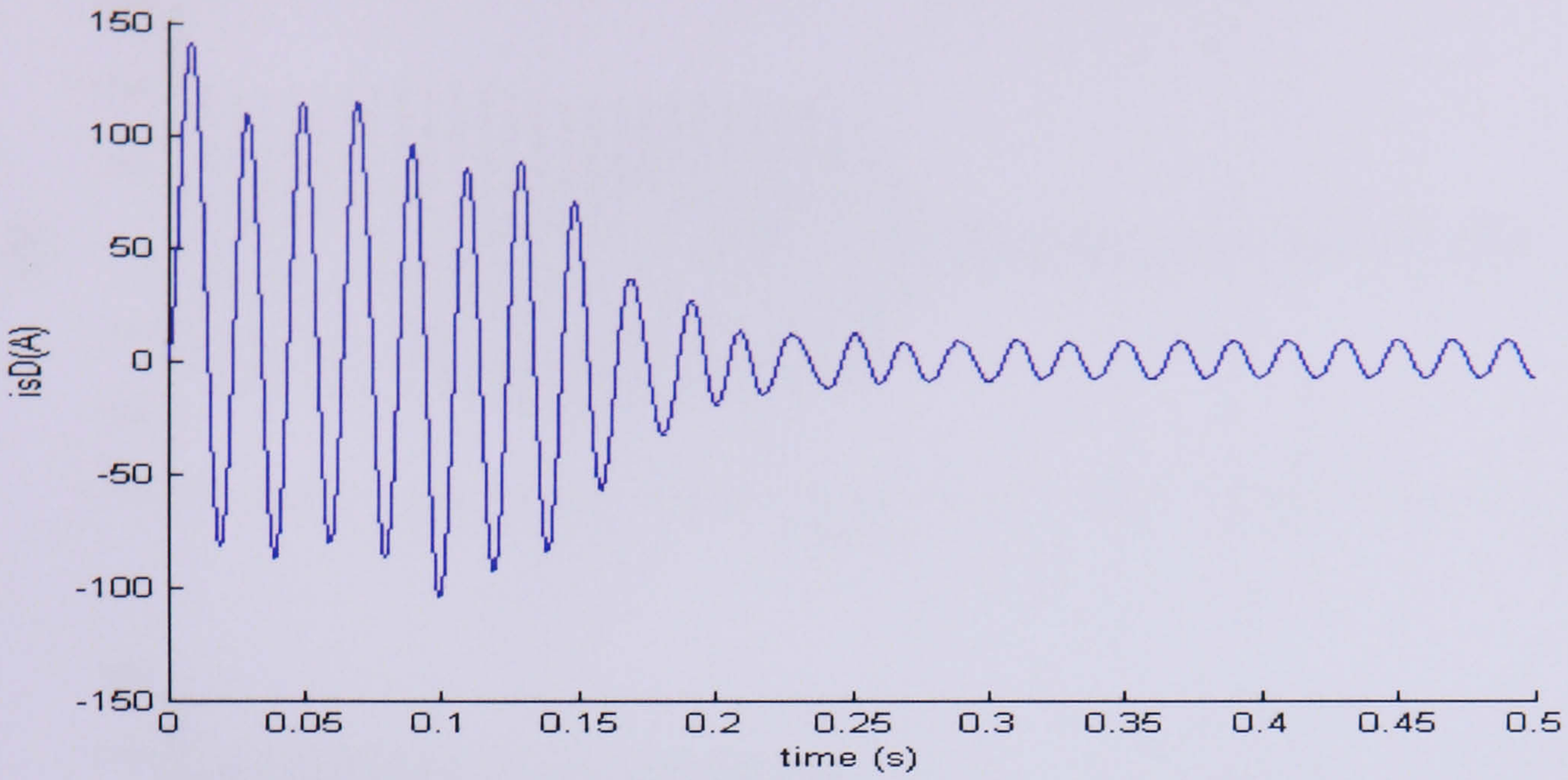


(b)

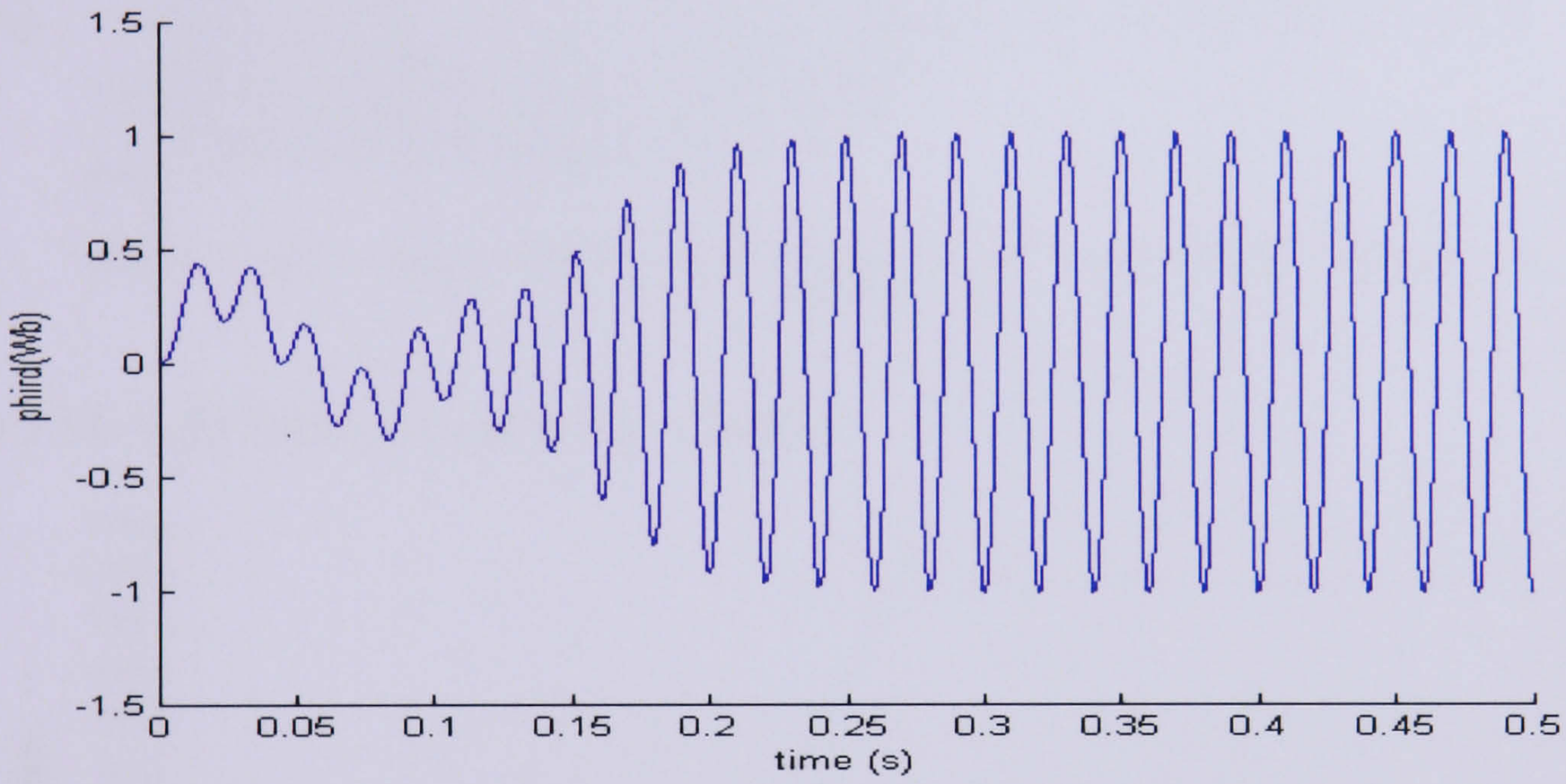


(c)

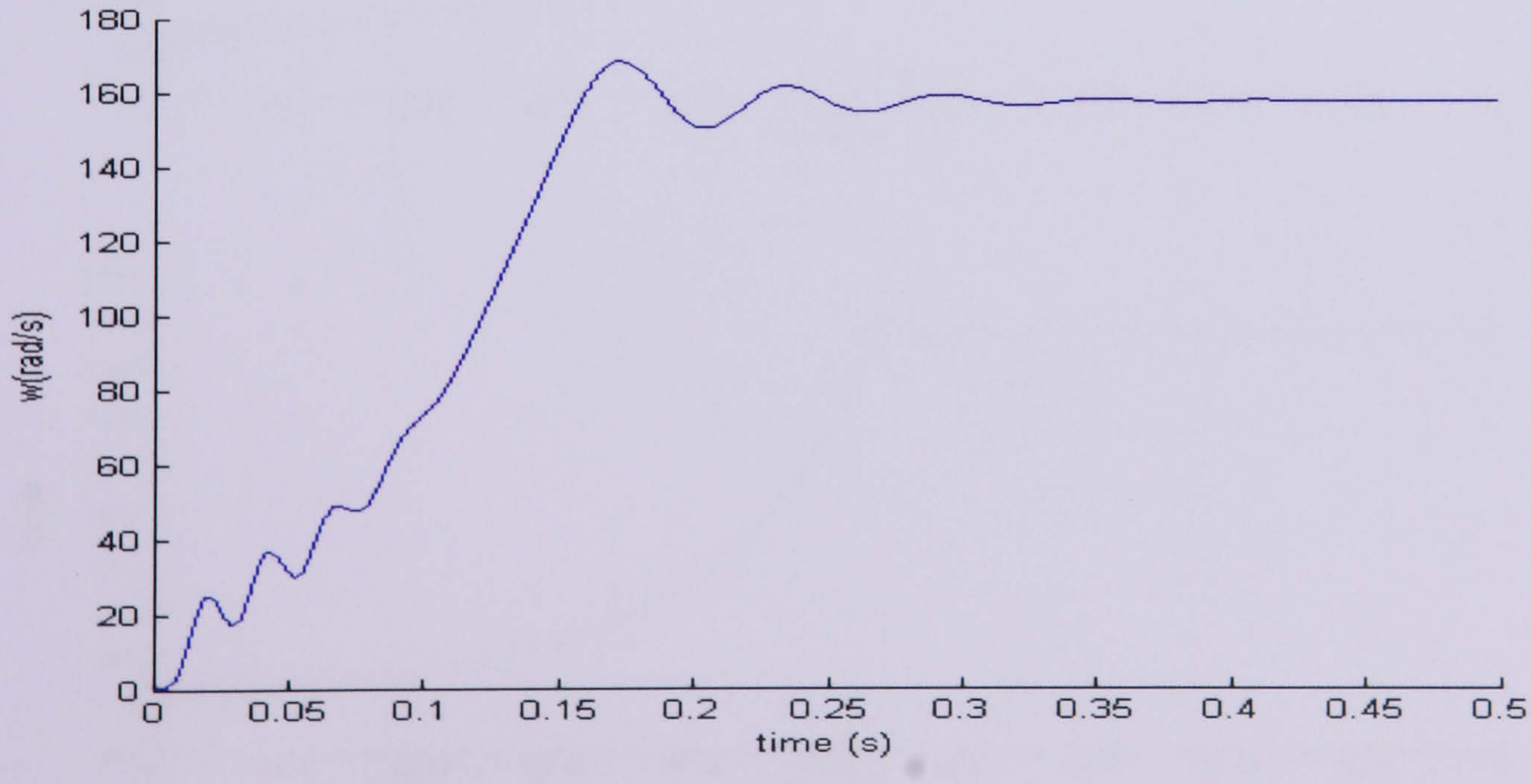
Figure 3.9: (a) i_{sD} (b) ψ_{dr} and (c) ω_r of reference motor model with no load condition



(a)

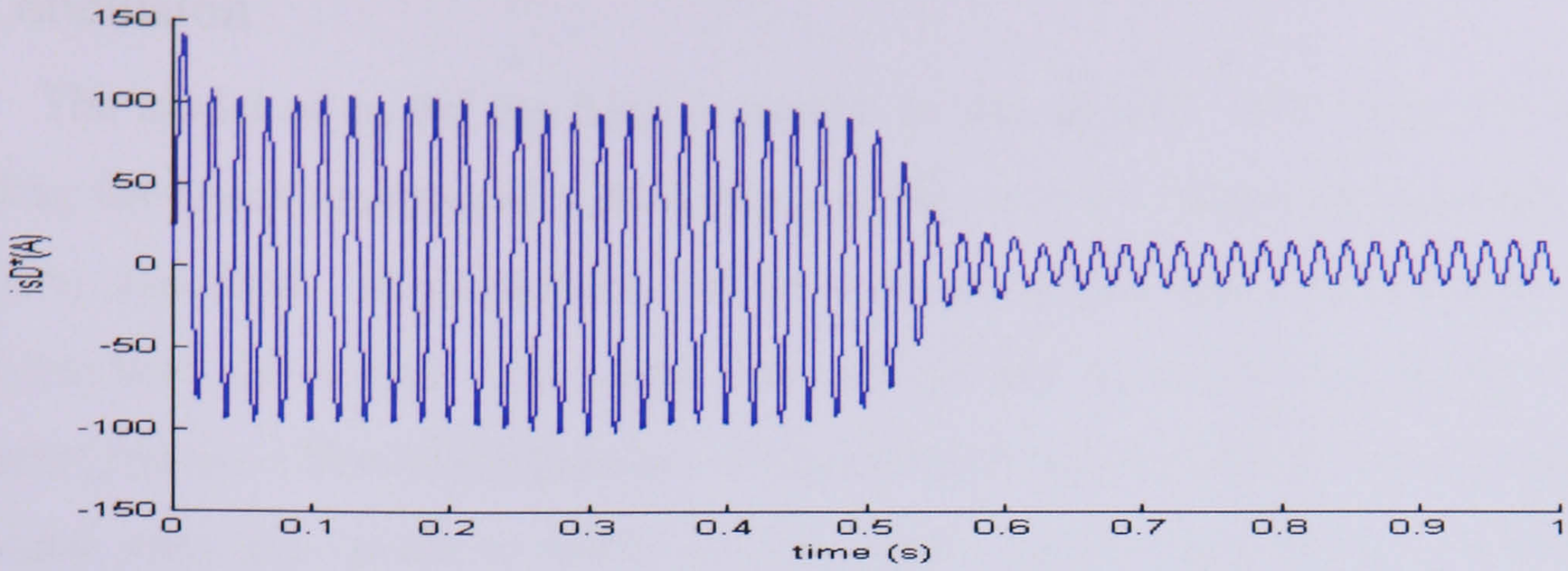


(b)

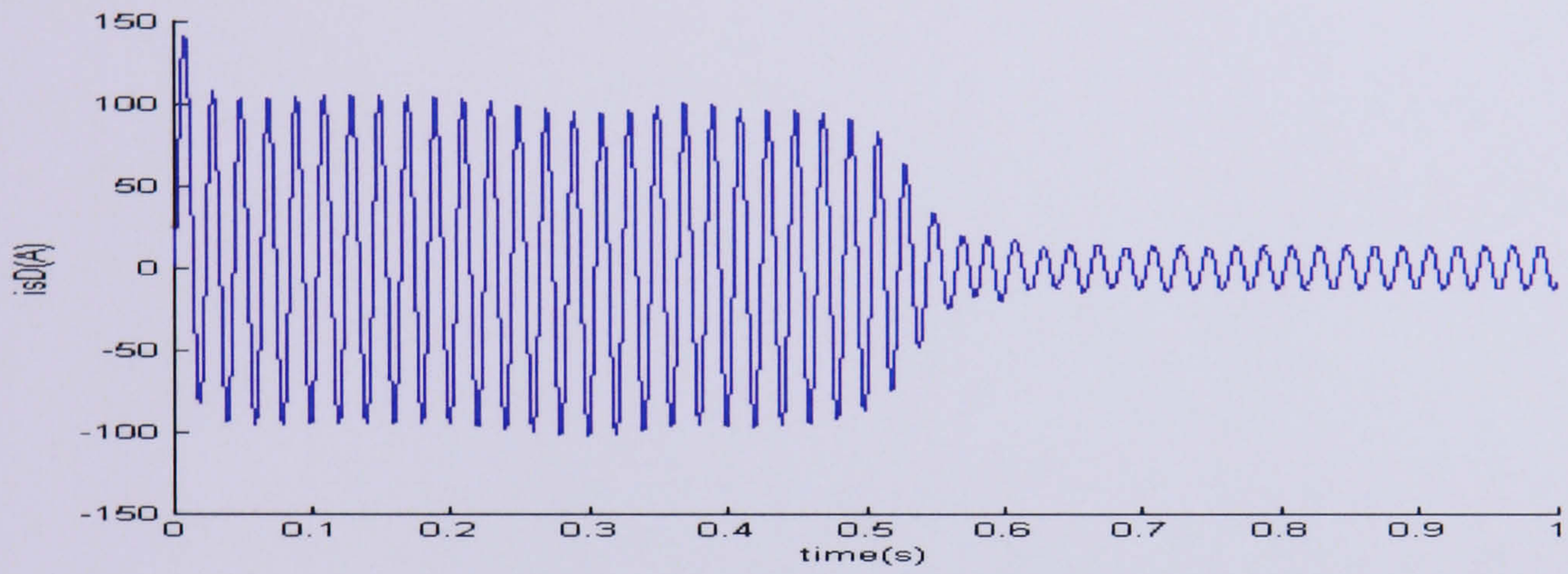


(c)

Figure 3.10: (a) i_{sD} (b) ψ_{dr} and (c) ω_r for continuous-time motor model with no load condition

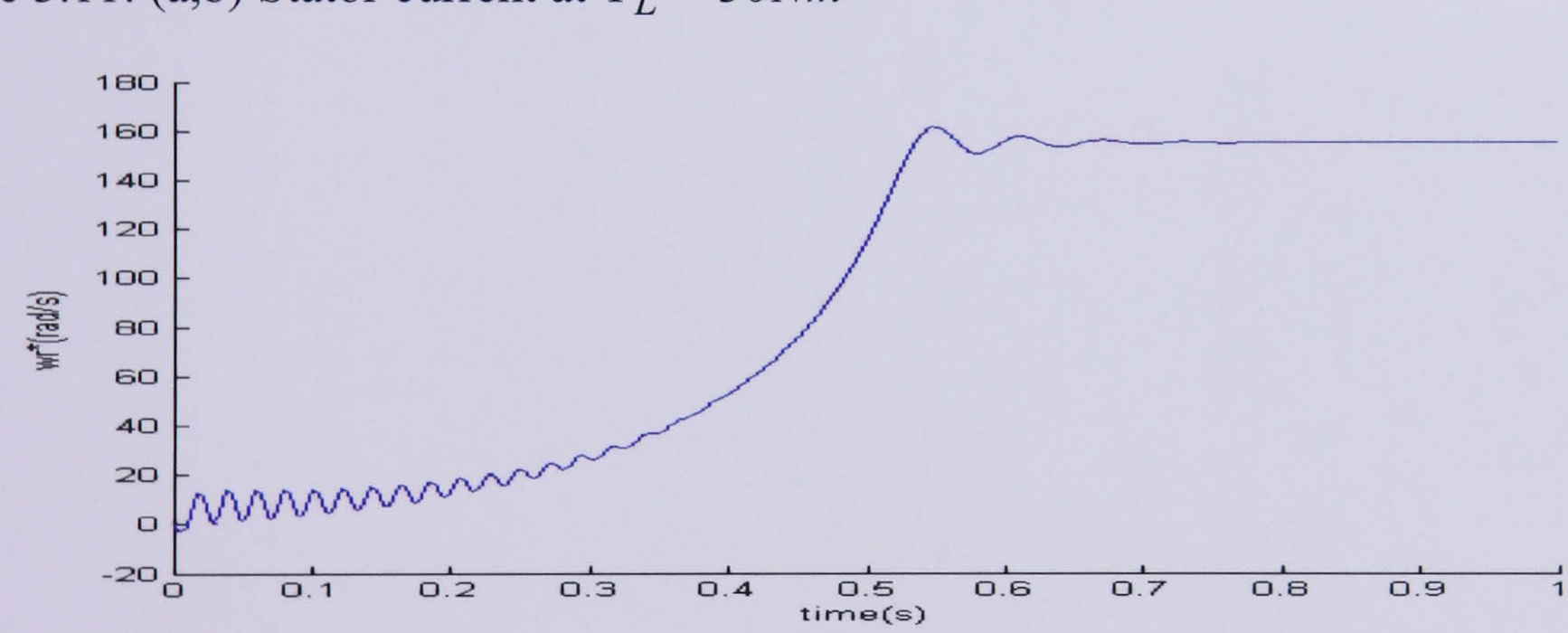


(a)

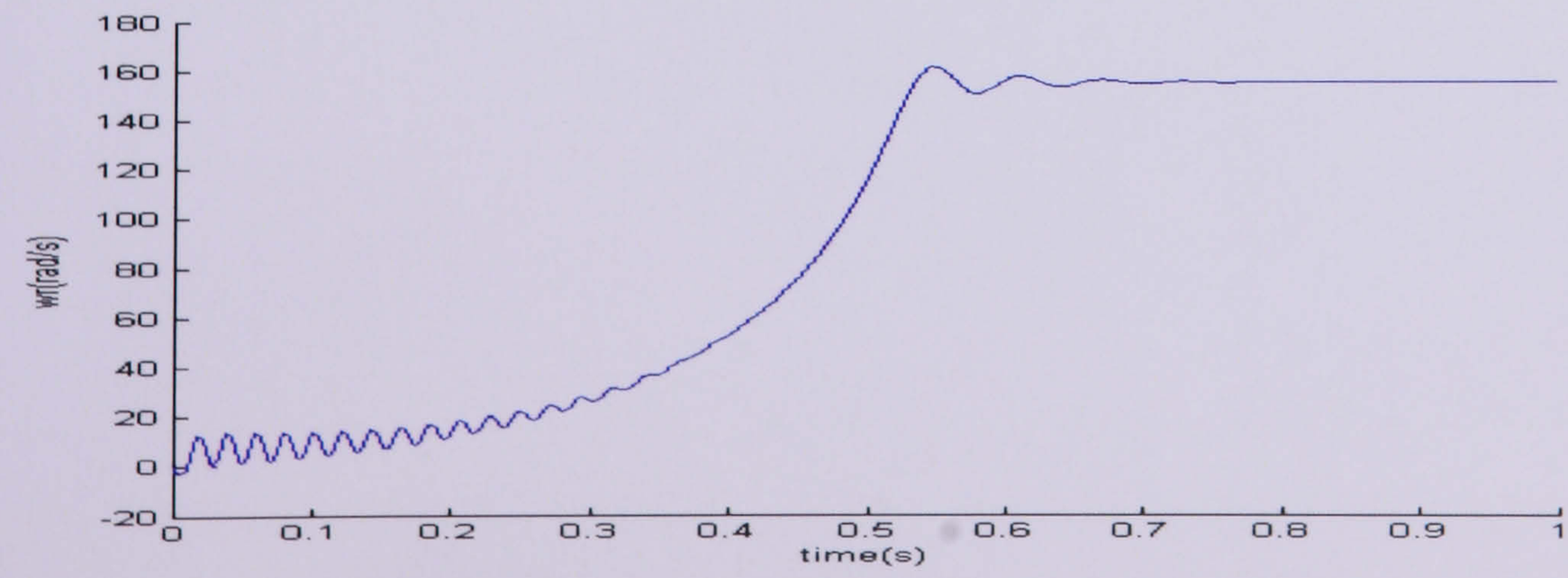


(b)

Figure 3.11: (a,b) Stator current at $T_L = 30Nm$



(a)



(b)

Figure 3.12: (a, b) rotor speed, ω_r at $T_L = 30Nm$

3.7 Conclusion

The model of an IM has been presented in this chapter. The space vector notation including the transformation of a three phase model to a two phase dq dynamic model has also been discussed. In preparation for the estimation problem, later on, the state space model has been developed. The fourth order differential equation of an IM is reduced to a first order matrix differential equation. Computer simulation results of the continuous time IM model were also given to verify its behaviour against other models with real motor parameters.

CHAPTER 4: KALMAN FILTERING

4.1 Introduction

Sensorless speed estimation has been a topic of interest among IM control researchers. The elimination of the sensor requires a control scheme where no mechanical parameters such as speed and torque are measured, only estimated. In general, there are two types of estimator, open loop and closed loop. The difference is that the closed loop estimator involves calculating an estimation error and then a correction term. This closed loop estimator is well known as an observer. The observer itself can be classified according to the type of the system to be observed which is either deterministic or stochastic. In some cases a deterministic system may not be totally sufficient. Therefore a stochastic observer is proposed.

One of the well known stochastic observers is the KF. The KF is defined as an optimal recursive data processing algorithm which works by incorporating all the information provided without need to store all previous data and reprocess it every time a new measurement is taken [78]. It is also just a computer program which incorporates mostly in discrete time measurement samples rather than continuous time inputs. The implementation of a KF may be slow but it is capable of good accuracy. In addition, a KF offers optimal filtering for Gaussian distributed noise in the system and during measurement as long as the noise covariance is known.

In this chapter, the different types of estimator are briefly discussed. The general structure of the KF is reviewed and derived. An initial study of a KF for estimating the stator and rotor flux of an IM drive has been carried out. Using the motor parameters in Table 3.1, the load torque as well as the rotor speed motor can be estimated. The results of the simulation are shown.

4.2 Conventional Observer Design

In the case of only the system input and outputs being available, while the states of the system cannot be measured, a closed loop estimator or "observer" is used. Using the input and the outputs of a system the observer is implemented to produce estimates of the system states. Based on the state space model described previously, it is reasonable to

simply reproduce the same system since all the states must be produced. This is shown in Figure 4.1 below. The method is known as an open loop estimator and has drawbacks. Firstly, if in an idealised situation the matrices **A, B, C** were known for both system and estimator the error would of course be zero. However since some differences between system and estimator will exist in practice, the error could not be used to correct the estimation.

To compensate for these problems, the closed loop estimator is used. The observer design proceeds by defining the error between the states and their estimates. In Figure 4.2 the gain K is added to correct the error between the actual and the estimator models.

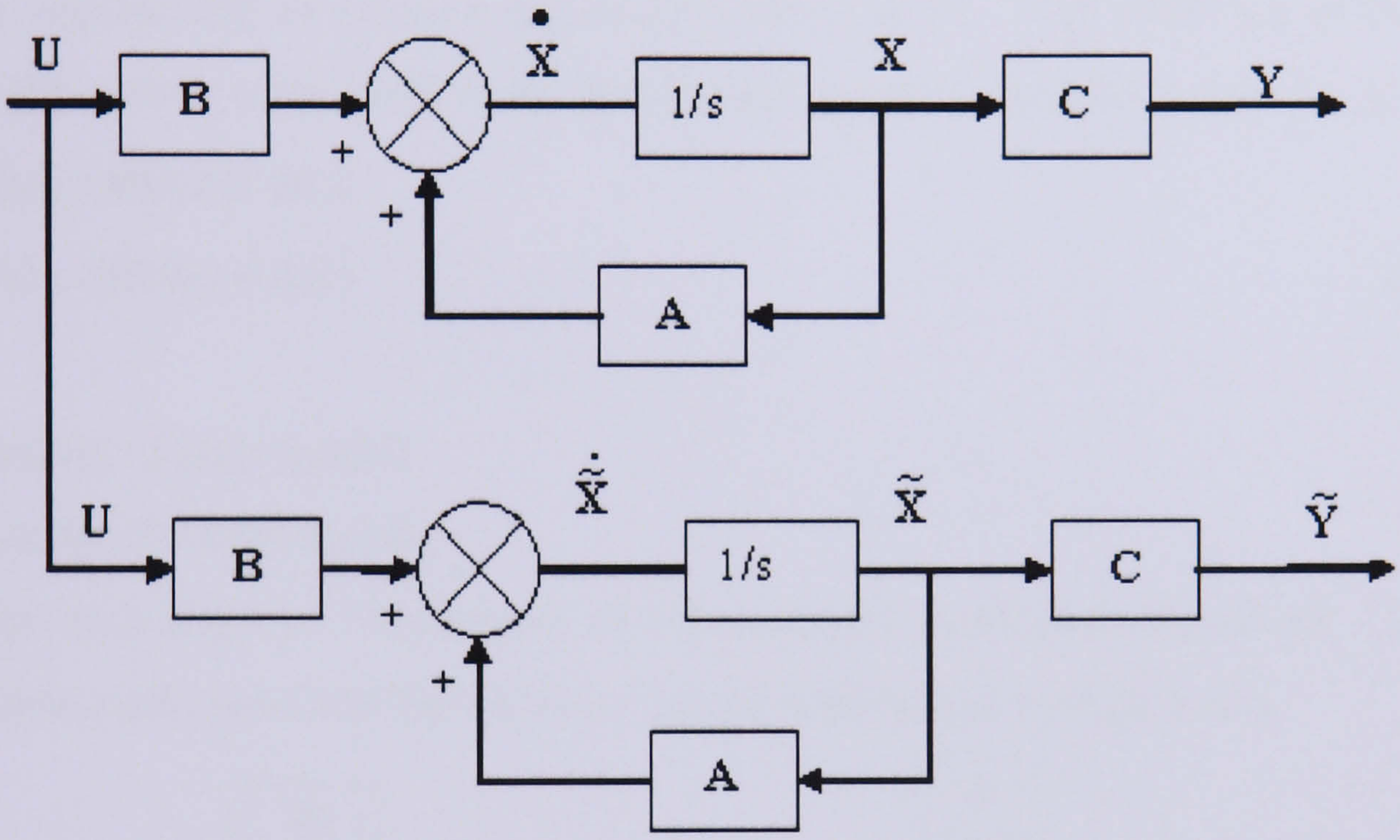


Figure 4.1: Open loop estimator

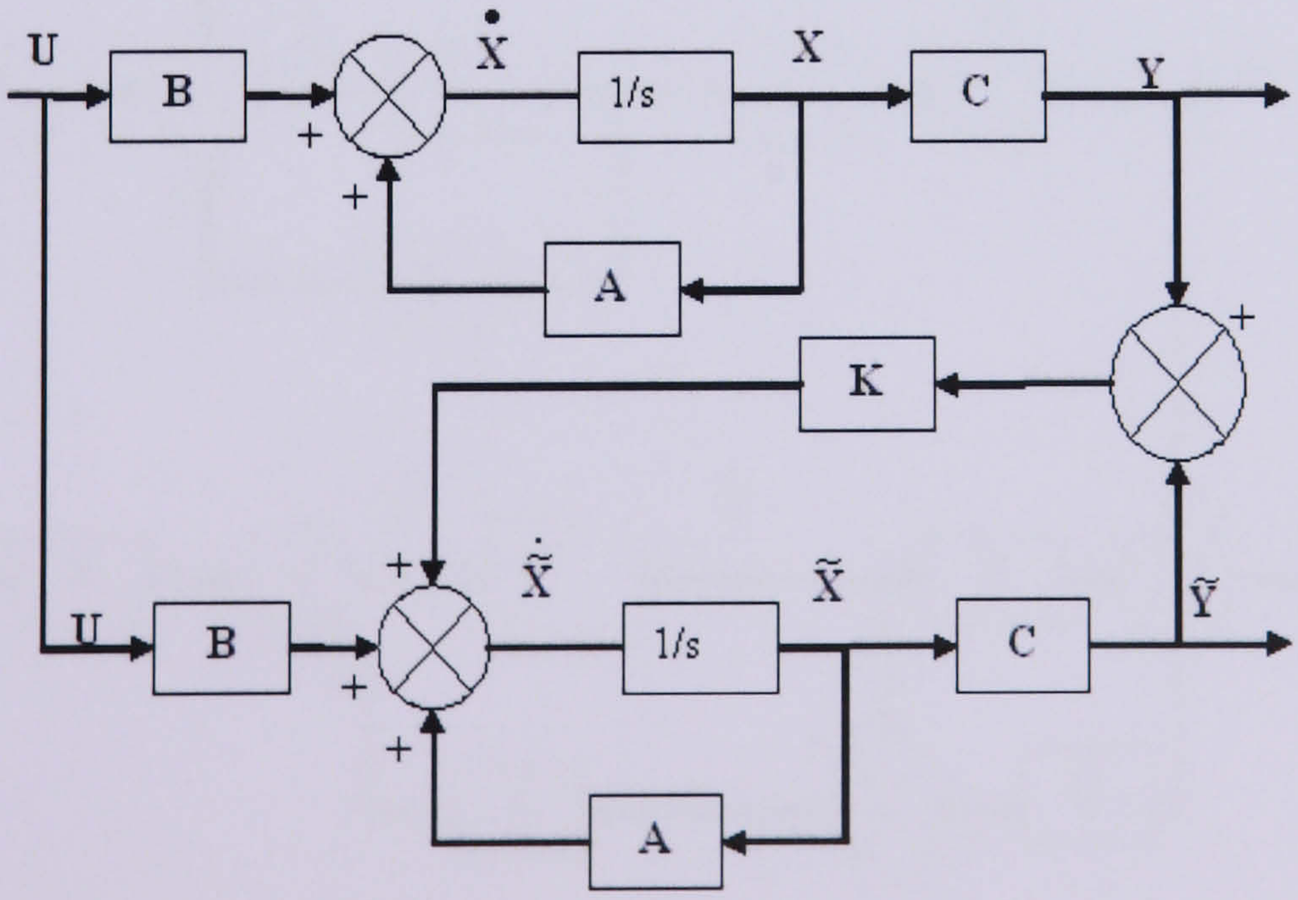


Figure 4.2: Closed loop estimator or observer

4.3 Kalman Filter Derivations

The state space model discussed in the previous section is an example of a deterministic system. In a deterministic system, the system is assumed to be perfect and the measurement made using a perfect sensor. However, in reality, there are uncertainties in the control system. First the control system, developed using a mathematical model, usually is imperfect. The mathematical model uses only a finite number of states and has approximations such as unmodelled dynamics. Secondly, the system may be forced by disturbances as well as the control inputs. Thirdly, the sensors used may have limitations so do not provide accurate data about the system [78]. These disturbances and limitation of the sensors are represented as process and measurement noise. The influence of these noise sources on the system is included in the steady state equation and can be written as:

$$\dot{\mathbf{X}}(t) = \mathbf{A}\mathbf{X}(t) + \mathbf{B}\mathbf{U}(t) + \mathbf{W}(t)$$

$$\mathbf{Y}(t) = \mathbf{C}\mathbf{X}(t) + \mathbf{D}\mathbf{U}(t) + \mathbf{V}(t) \qquad (4.1)$$

where

W- noise matrix of state model;

C -noise matrix of output model;

These new Eqn (4.1) are known as the stochastic state space equations. The model is in stochastic state space and the observer can be represented by Figure 4.3.

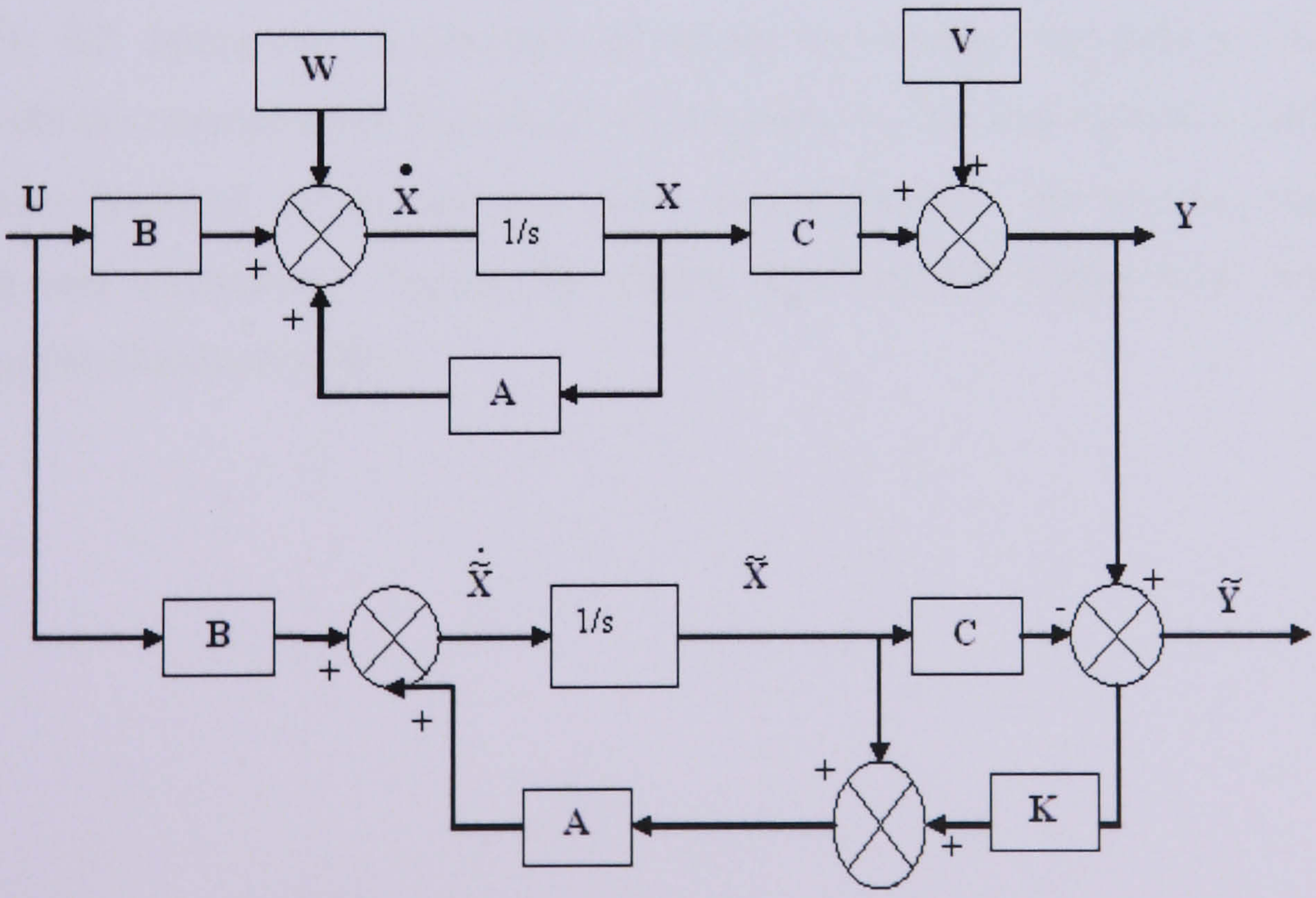


Figure 4.3: Stochastic state space model

The KF is often used for stochastic estimation. It is based on mathematical equations which implement a predictor and corrector type estimator. It is an optimal

estimator in the sense of minimising the estimated error covariance. Based on the original formulation of the KF [47] and with the aid of [79,80], the discrete time recursive equations of KF have been derived and are given in Appendix C. However, some slight changes in notation have been made for ease of understanding. A stochastic continuous system as in Eqn (3.1) and (3.2) can be described in discrete time as:

$$\mathbf{X}_{k+1} = \mathbf{\Phi}_k \mathbf{X}_k + \mathbf{\Gamma}_k \mathbf{U}_k + \mathbf{W}_k$$

$$\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{V}_k \tag{4.2}$$

In the following derivation several assumptions are needed. The noisy signals $\mathbf{W}_k$ and $\mathbf{V}_k$ are regard as zero mean, white noise and being totally uncorrelated with each other.

$$E[\mathbf{W}_k] = 0 \tag{4.3}$$

$$E[\mathbf{V}_k] = 0 \tag{4.4}$$

$$E[\mathbf{W}_k \mathbf{W}_i^T] = \begin{cases} \mathbf{Q}_k & i = k \\ 0 & i \neq k \end{cases} \tag{4.5}$$

$$E[\mathbf{V}_k \mathbf{V}_i^T] = \begin{cases} \mathbf{R}_k & i = k \\ 0 & i \neq k \end{cases} \tag{4.6}$$

$$E[\mathbf{W}_k \mathbf{V}_i^T] = 0 \text{ For all k and i} \tag{4.7}$$

The KF addresses the problem of trying to estimate the state of the controlled process with the measurement Eqn (4.2). It estimates the process state at a certain time and then obtains feedback in the form of noisy measurement. The process has two steps, prediction and correction. Figure 4.4 shows how the KF works with correction and prediction simultaneously [81].

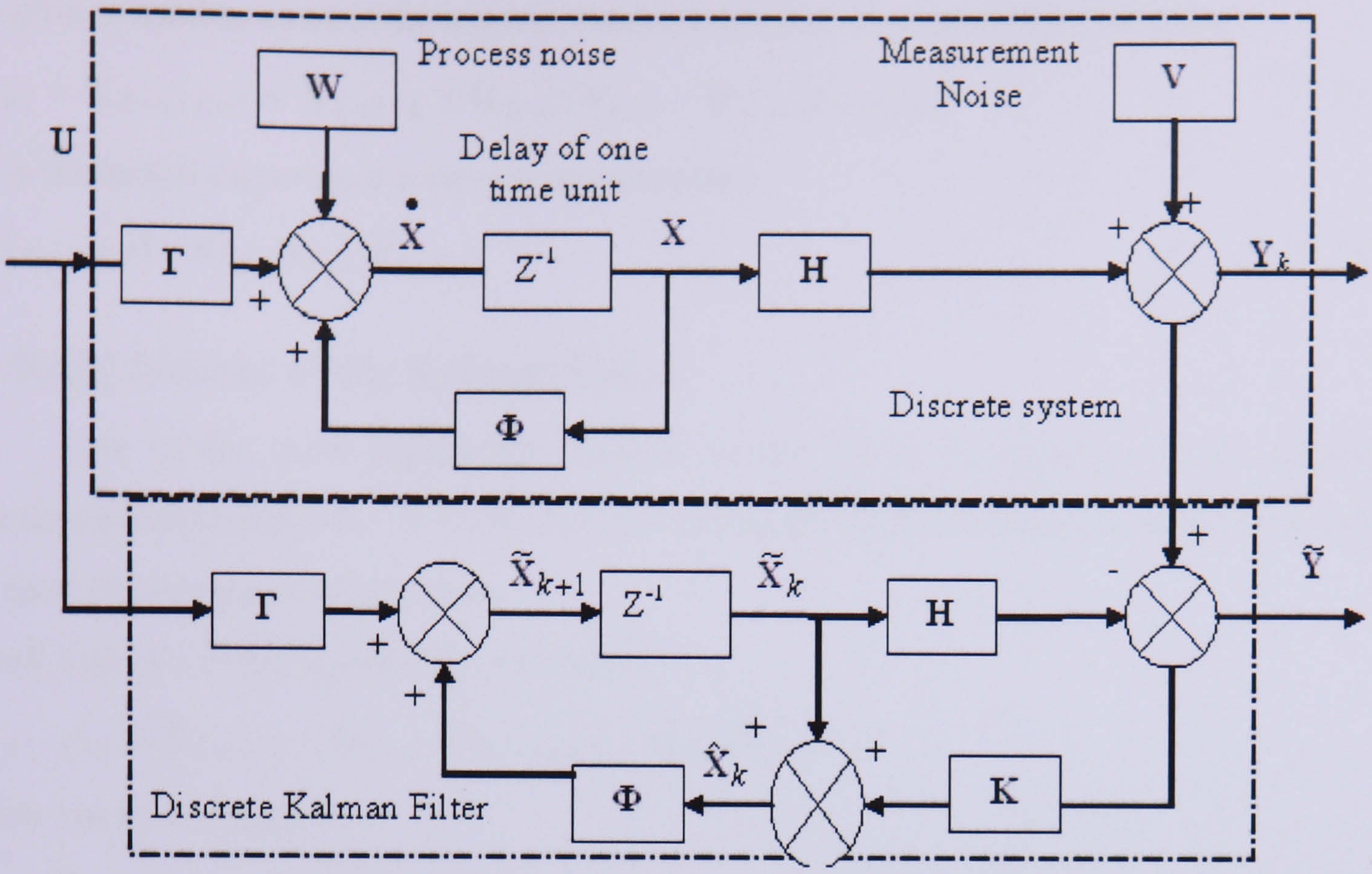


Figure 4.4: Block diagram of the discrete time system and KF

The Prediction Stage:

The prediction is responsible for projecting forward the current state based on past corrected measurements of the outputs. The a priori state estimate is based on the previous corrected estimate of the state and the current value of the input.

$$\tilde{\mathbf{X}}_{k+1} = \hat{\mathbf{X}}_{k+1/k} = \mathbf{\Phi}_k \hat{\mathbf{X}}_k + \mathbf{\Gamma}_k \mathbf{U}_k = \mathbf{\Phi}_k \hat{\mathbf{X}}_{k/k} + \mathbf{\Gamma}_k \mathbf{U}_k \tag{4.8}$$

Using the priori state estimate, the a priori covariance can be calculated

$$\overline{\mathbf{P}}_{k+1/k} = \mathbf{\Phi}_k \overline{\mathbf{P}}_{k/k} \mathbf{\Phi}_k{}^T + \mathbf{Q}_k \tag{4.9}$$

Note that these two equations use previous values of the a posteriori state estimate and posteriori covariance. Therefore the first iteration of a KF requires estimates of these two variables (which are often just guesses). The exact estimate is not that important as the values will be corrected over time. However a bad initial estimate just needs more iteration to converge.

The Correction Stage:

The correction is responsible for updating and refines the result given from prediction step. In other words, the correction is a feedback to incorporate a new measurement into the priori estimate so that this value can be used to improve the estimate. To correct the a priori estimate, the KF gain, K is needed.

$$\mathbf{K}_{k+1} = \mathbf{H}_{k+1}{}^T \overline{\mathbf{P}}_{k+1/k} (\mathbf{H}_{k+1} \overline{\mathbf{P}}_{k+1/k} \mathbf{H}_{k+1}{}^T + \mathbf{R}_{k+1})^{-1} \tag{4.10}$$

The gain is used to correct the a priori estimate to give the a posterior estimates.

$$\hat{\mathbf{X}}_{k+1} = \bar{\mathbf{X}}_{k+1/k+1} = \bar{\mathbf{X}}_{k+1/k} + \mathbf{K}_{k+1}(\mathbf{Y}_{k+1} - \mathbf{H}_{k+1}\bar{\mathbf{X}}_{k+1/k}) \tag{4.11}$$

The a posteriori covariance now can be calculated

$$\overline{\mathbf{P}}_{k+1/k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\overline{\mathbf{P}}_{k+1/k} \tag{4.12}$$

4.4 Brief features of the Kalman Filter

One of the most significant features of the KF is its recursive form property in processing measurements. It works by processing the measurements as they occur without the need for storage of all the data.

Recall Eqn (4.11) the a posteriori estimator:

$$\bar{\mathbf{X}}_{k+1/k+1} = \bar{\mathbf{X}}_{k+1/k} + \mathbf{K}_{k+1}(\mathbf{Y}_{k+1/k+1} - \mathbf{H}_{k+1}\bar{\mathbf{X}}_{k+1/k})$$

which can be written as:

$$\hat{\mathbf{X}}_k = \tilde{\mathbf{X}}_k + \mathbf{K}_k(\mathbf{Y}_k - \mathbf{H}_k\tilde{\mathbf{X}}_k)$$

The difference $(\mathbf{Y}_k - \mathbf{H}_k\tilde{\mathbf{X}}_k)$ is called the residual or innovation. In this case, the residual reflects the discrepancy between the predicted measurement and the actual measurement. A small residual means that the observer is good and vice versa. A zero residual means that the system and observer are in complete agreement which is usually unfeasible.

Recall Eqn (4.10),

$$\mathbf{K}_{k+1} = \mathbf{H}_{k+1}{}^{T}\overline{\mathbf{P}}_{k+1/k}(\mathbf{H}_{k+1}\overline{\mathbf{P}}_{k+1/k}\mathbf{H}_{k+1}{}^{T} + \mathbf{R}_{k+1})^{-1}$$

For a small priori error, $\overline{\mathbf{P}}_{k+1/k}$, corresponding to a good system model, the gain K will be small. In this case the predicted measurement $\mathbf{H}_k\tilde{\mathbf{X}}_k$ is trusted more and KF uses the past estimate to form a new estimate. The actual measurement $\mathbf{Y}_k$ will be ignored. This means that if the a priori error estimate is good, there is little need to correct it.

For a large priori error, $\overline{\mathbf{P}}_{k+1/k}$, corresponding to a bad model, the gain $K = \frac{1}{H}$.

This means that the actual measurement $\mathbf{Y}_k$ is trusted more than the predicted measurement $\mathbf{H}_k\tilde{\mathbf{X}}_k$ and indicates a large error in the estimation process. The current measured value of the output is used to estimate the state and the priori estimate is ignored.

Furthermore, K can be expressed as $K \propto \frac{\mathbf{Q}}{\mathbf{R}}$ [82].

If **Q** is increasing the KF assumes that the model of the system is very inaccurate and makes the gain increase, the same effect is achieved by small **R**. On the other hand, as **Q** decreases (**R** increases), the KF assumes that the model of the system is more accurate and therefore prediction is given more attention than correction. Hence if the measurement noise is large, then the estimate will depend more upon the previous estimates and less dependent on the measurement. If by any chance **Q**=0, the gain will be zero and this may result in an open loop estimator and give poor performance. On the other hand, it is not possible to have **R**=0 since the calculation of the Kalman gain requires matrix inversion and **R** needs to have a finite value.

4.5 Motor Dynamic Model in Discrete Form

The continuous model of the IM has been discussed in the previous chapter. To be able to apply the recursive method, a discrete model of the IM is required. The process to be estimated and the output to be measured can be modelled in discrete time using the notation,

$$\mathbf{X}_{k+1} = \mathbf{\Phi}_k \mathbf{X}_k + \mathbf{\Gamma}_k \mathbf{U}_k$$

$$\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k \tag{4.13}$$

The discrete model is obtained using an approximate second order series expansion method.

$$e^{\mathbf{A}T} \approx 1 + T\mathbf{A} + \frac{T^2\mathbf{A}^2}{2!} \tag{4.14}$$

$$\mathbf{B} \approx T\mathbf{B} + \frac{\mathbf{A}\mathbf{B}T^2}{2!} \tag{4.15}$$

where T is the sampling time.

This second order technique gives a better approximation however it increases the computation time. Therefore to have short sampling times in the simulation, the last term of Eqn (4.14) and (4.15) was neglected. Using the time delay approximation method above together with Eqn (4.13) the IM model can be expressed as follows,

$$\mathbf{x}_k = \begin{bmatrix} i_{sD} \\ i_{sQ} \\ \psi_{rd} \\ \psi_{rq} \end{bmatrix} \quad \mathbf{Y}_k = \begin{bmatrix} i_{sD} \\ i_{sQ} \end{bmatrix} \quad \mathbf{u}_k = \begin{bmatrix} u_{sD} \\ u_{sQ} \end{bmatrix}$$

$$\mathbf{\Phi}_k = \begin{bmatrix} 1-\frac{K_2 T}{K_1} & 0 & \frac{L_m T}{K_1 L_r T_r} & \frac{\omega_r L_m T}{K_1 L_r} \\ 0 & 1-\frac{K_2 T}{K_1} & -\frac{\omega_r L_m T}{K_1 L_r} & \frac{L_m T}{K_1 L_r T_r} \\ \frac{L_m T}{T_r} & 0 & 1-\frac{T}{T_r} & -\omega_r T \\ 0 & \frac{L_m T}{T_r} & \omega_r T & 1-\frac{T}{T_r} \end{bmatrix} \quad \mathbf{\Gamma}_k = \begin{bmatrix} \frac{T}{K_1} & 0 \\ 0 & \frac{T}{K_1} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Therefore the output matrix is:

$$\mathbf{H}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{4.16}$$

4.6 Continuous and Discrete Model Comparison

The block diagram of the discrete model of IM given in Eqn (4.16) is shown in Figure 4.5. The model is exactly as the continuous model including the torque calculation except that the IM equations are in discrete form. To compare the continuous and discrete model of the IM, each model is run for 0.5s and a small sampling time of $T = 10us$ is required for the discrete model. For the discrete IM model, there is a total of 50 000 iterations. The sampling time seems to be very small, but [38] uses even smaller sampling times. Various values of T has been experiment and the smaller sampling time used, the slower the simulation will be.

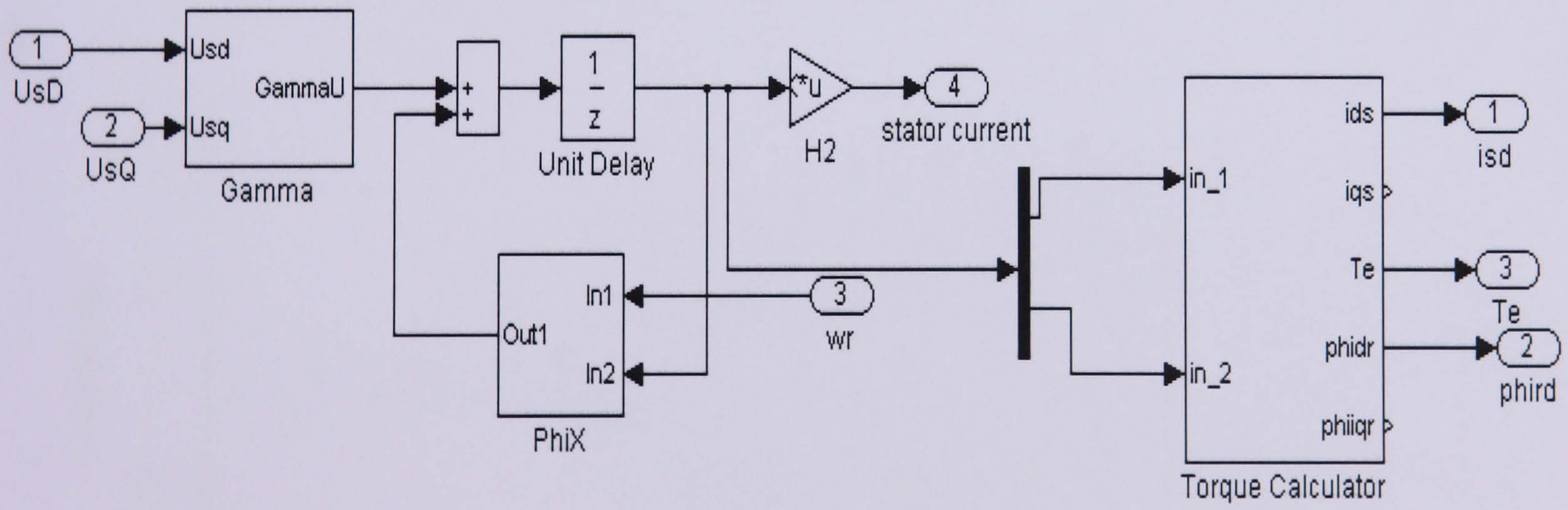
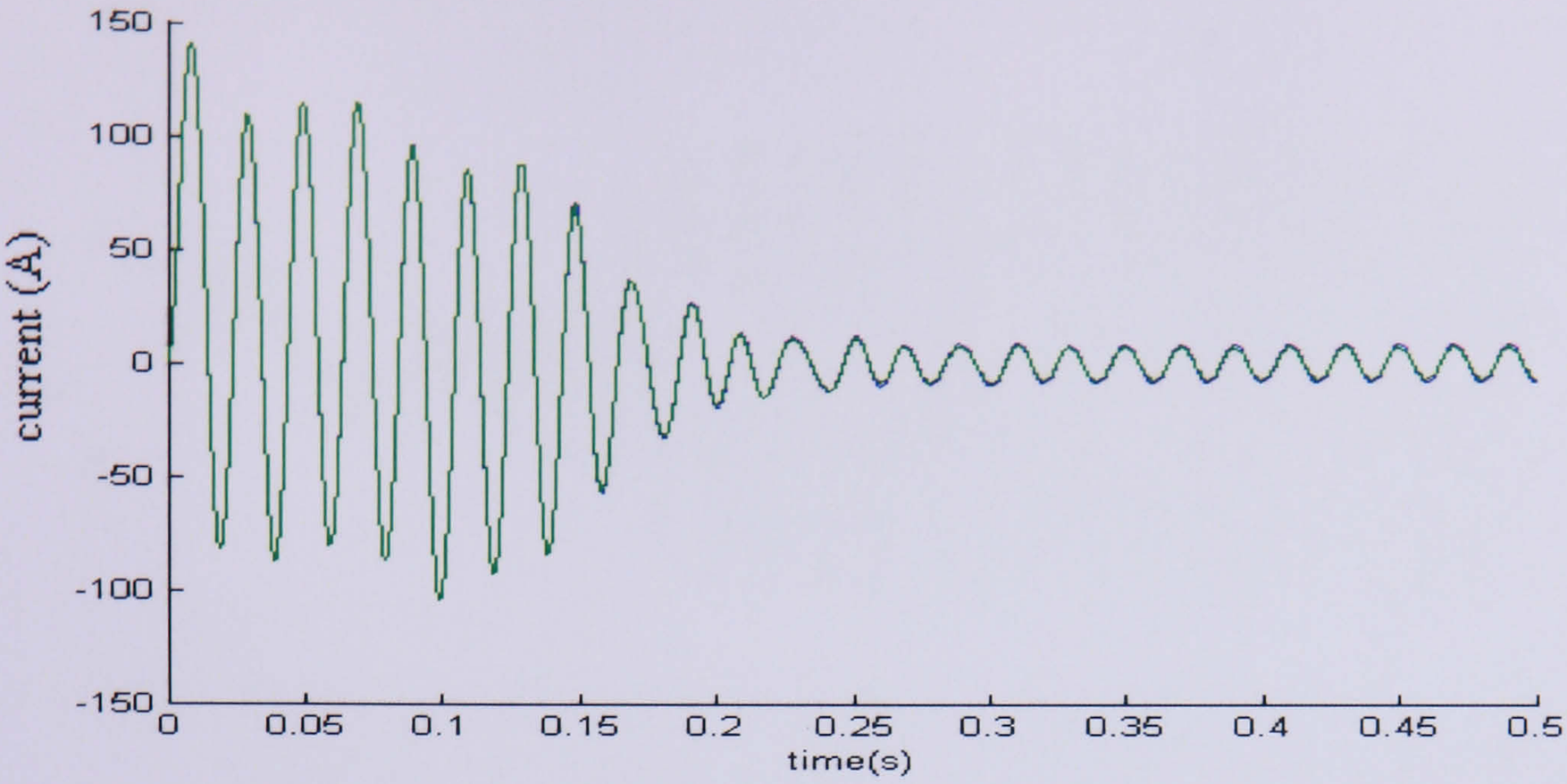
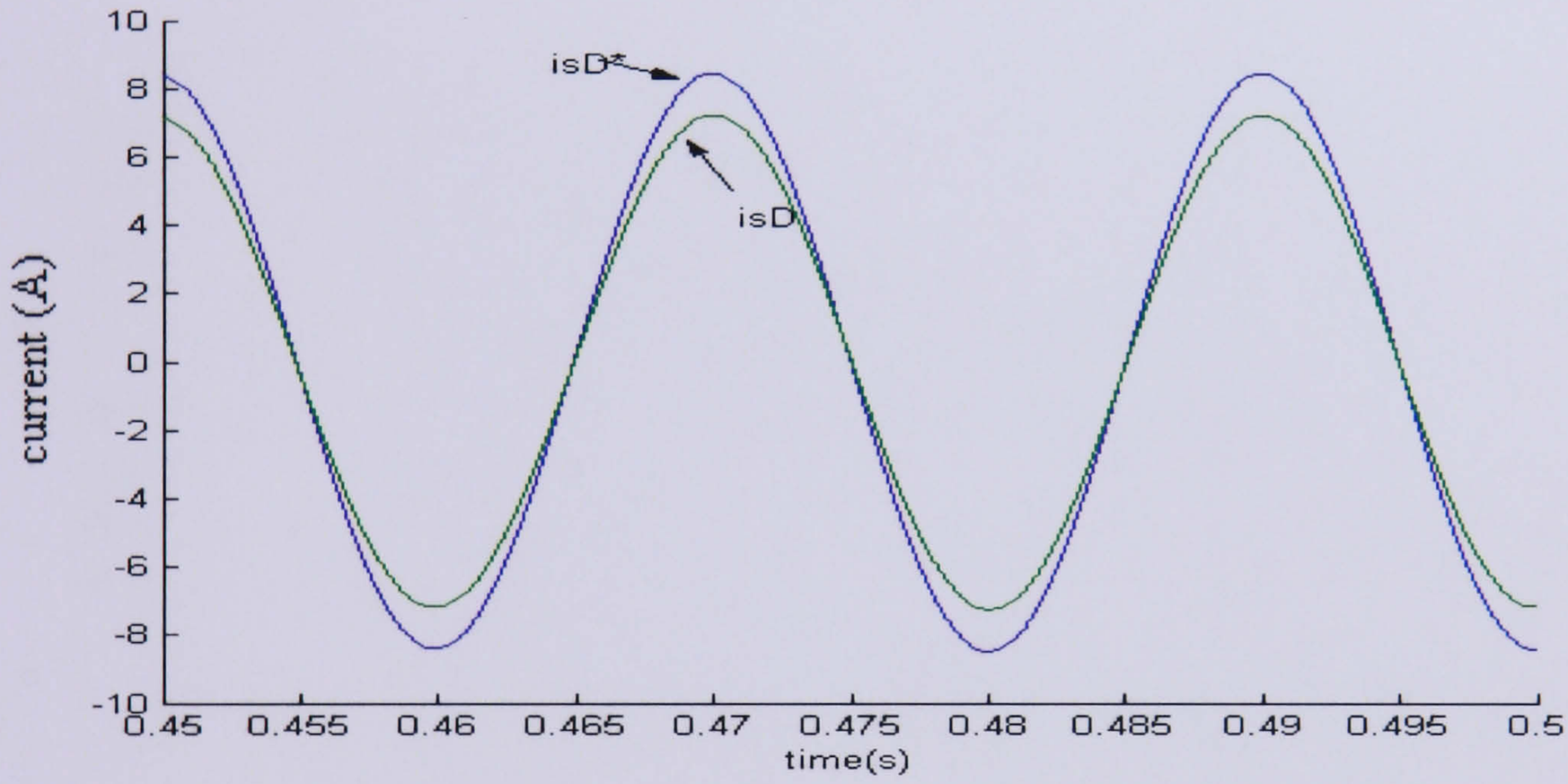


Figure 4.5: The discrete model of the IM

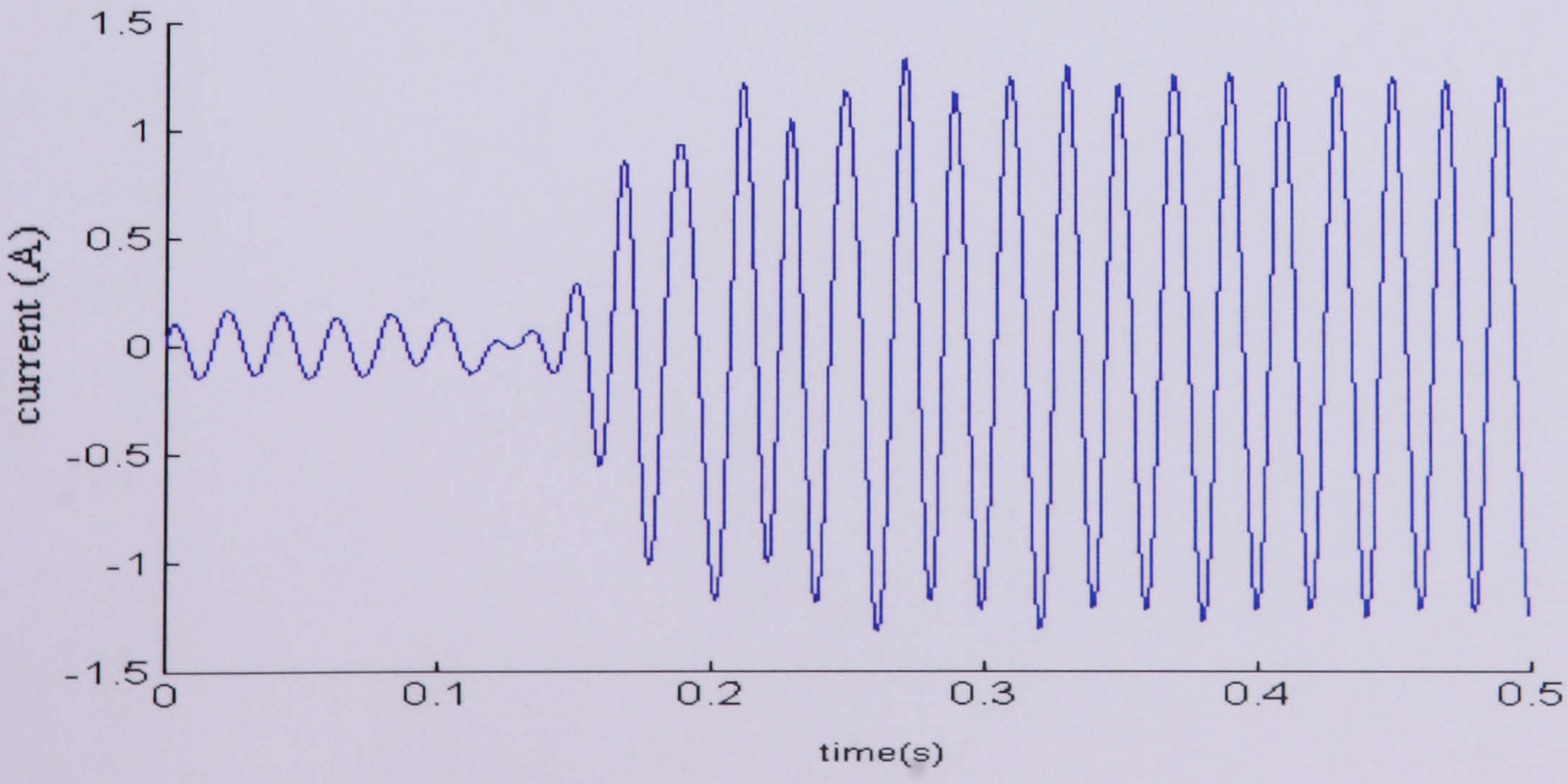
The continuous model with * and discrete model for the d axes stator current and rotor flux is plotted in the same graph and can be seen in Figure 4.6 and Figure 4.7. The simulation result has shown that the current has about 14% error while the rotor flux is 1.3% in error. Better result can be achieved if a smaller sampling time is used but this makes the simulation very slow.



(a)

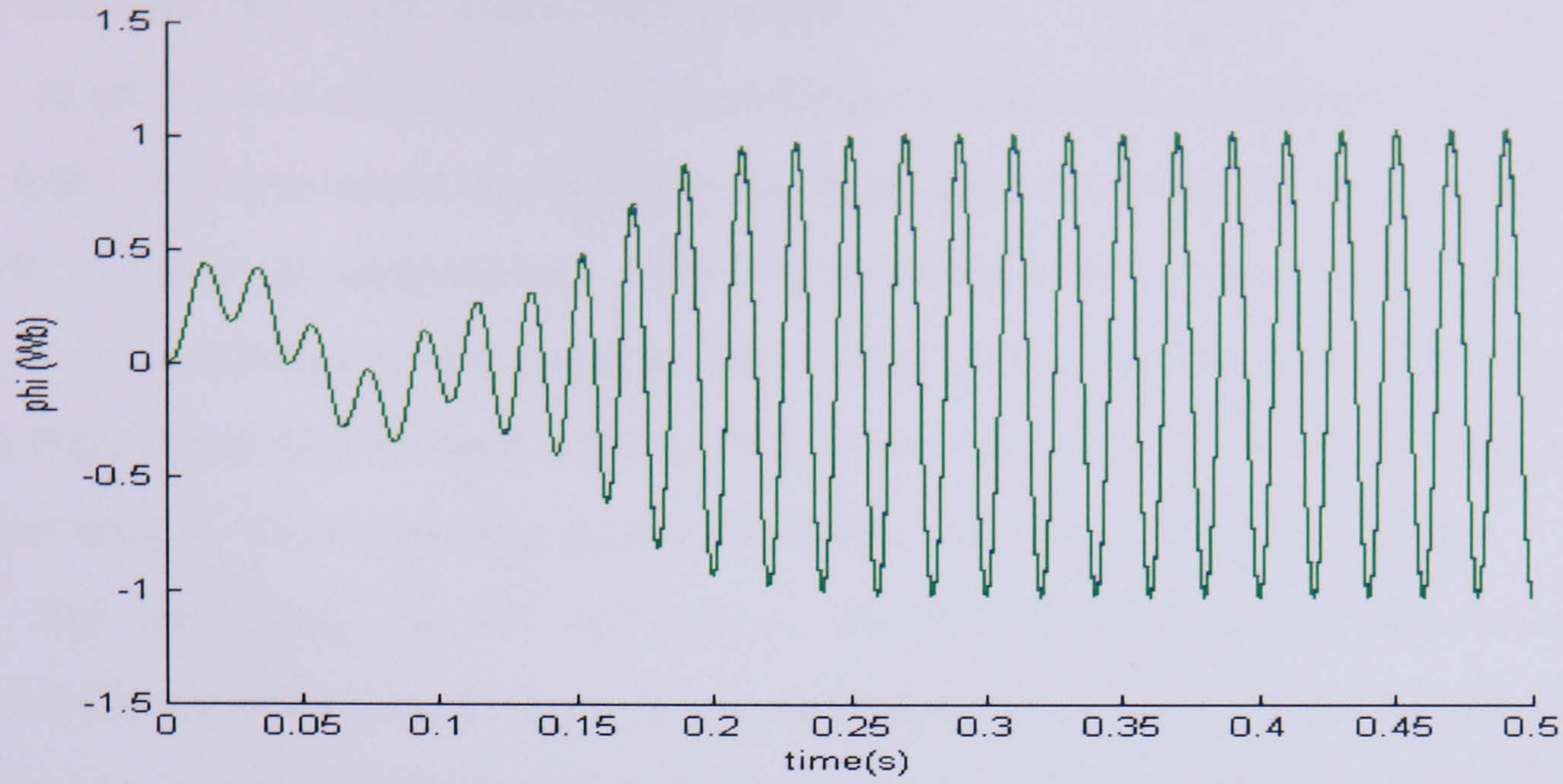


(b)

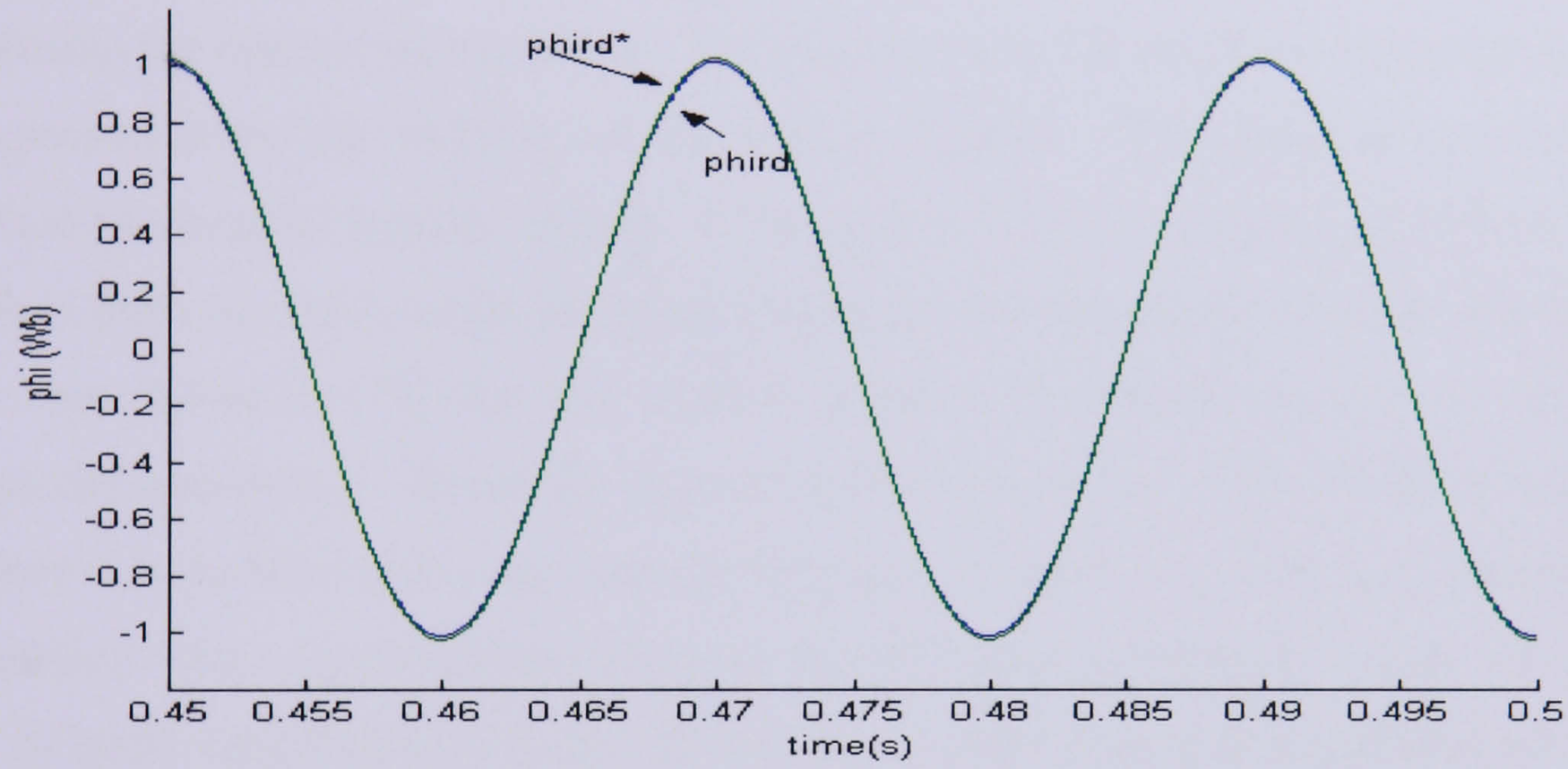


(c)

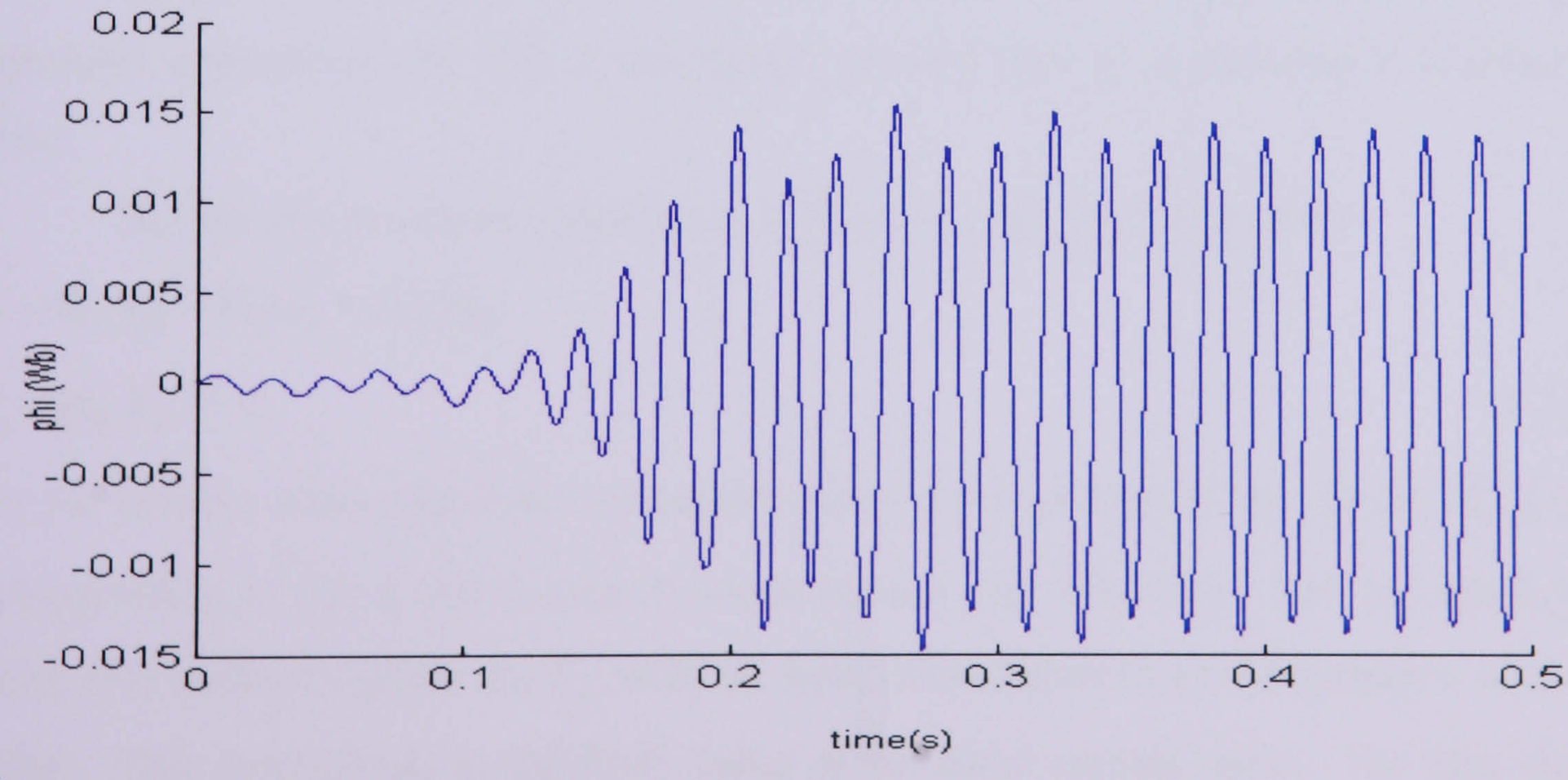
Figure 4.6: (a)The d-axis stator current for continuous and discrete in full figure (b) enlarged (c) the error between continuous and discrete



(a)



(b)



(c)

Figure 4.7: (a) The d-axis rotor flux for continuous and discrete in full and (b) enlarged (c) the error between continuous and discrete model

4.7 Uncertainties of the Induction Motor

In an IM, one of the factors that contribute to uncertainty is parameter variation. The stator and rotor resistances are perhaps the most important sources of uncertainty due to their dependence on temperature. This is termed process/ system uncertainty. Another factor that contributes to the uncertainties is the un-measurable states of the motor. To access these states they to need to be identified or measured and the sensor used will not be a perfect sensor. Such imperfect sensors contribute measurement uncertainties.

For simulating the KF application, the system and measurement noise which represent the uncertainties of the motor are considered to be in the form of white noise. The noise can be regarded as being inherent (e.g. supply voltage harmonics) or as being added from an external source [83]. This is shown in Figure 4.8. A random source block is used to represent the random uncertainties. The process noise, **Q**, and the measurement noise, **R**, are represented by the variance of the random signals. The variance is a very useful statistical property for random signals. If the variance is known it gives an idea of how far a sample is from the mean value and gives a sense of how much jitter or noise is in the signal. It has been shown in [79] that four random sources are required in order to represent the state model uncertainty. However, in practice, the uncertainty of the model is very general therefore it is possible to use one random source to represent it. If the four random sources are replaced with a single random source in the IM model as shown in Figure 4.8, it will not yield different amplitudes of noise. **Q** and **R** give the statistical description of the drive model. If **Q** is increased, the drive is in the condition of heavy system noise or increased parameter uncertainty and if **R** is increased, it means that the measurement is affected by noise.

The discrete stochastic equation as in Eqn (4.1) and (4.2) is applied:

$$\dot{\mathbf{X}} = \mathbf{\Phi}_k \mathbf{x}_k + \mathbf{\Gamma}_k \mathbf{u}_k + \mathbf{G}_k \mathbf{W}_k$$

$$\mathbf{Y} = \mathbf{H}_k \mathbf{X}_k + \mathbf{V}_k \qquad (4.17)$$

For the process noise, the four independent noisy sources namely are Q_1, Q_2, Q_3, Q_4, are corresponding to the d and q axes of stator current and rotor flux. For the measurement noise, two random signals R_1, R_2 with the same characteristics as the system's noise were added. This corresponds to the RMS value of the stator current noise. The four diagonal elements G_1, G_2, G_3, G_4 represent the gain weight of the state vector. The noise gain weight **G** used in the simulation is equal to the square root of variance **Q** which is known as the

standard deviation. Setting the variance **Q** to one and changing the gain **G** results in a change in the standard deviation of the random source. By setting the gain **G** to one the level of uncertainty of the model can be adjusted by changing the variance **Q** of the random source. Using this concept therefore some authors may use the gain weight [59,79] but some do not [38,51,52,58,84]. The covariance value of each random signal can be summarised as:

a) The process noise matrix:

$$Q = \begin{bmatrix} Q_{11} & 0 & 0 & 0 \\ 0 & Q_{22} & 0 & 0 \\ 0 & 0 & Q_{33} & 0 \\ 0 & 0 & 0 & Q_{44} \end{bmatrix}$$

b) The measurement noise matrix:

$$R = \begin{bmatrix} R_{11} & 0 \\ 0 & R_{22} \end{bmatrix}$$

c) The gain matrix:

$$G = \begin{bmatrix} G_{11} & 0 & 0 & 0 \\ 0 & G_{22} & 0 & 0 \\ 0 & 0 & G_{33} & 0 \\ 0 & 0 & 0 & G_{44} \end{bmatrix}$$

It was mentioned earlier that the noise is assumed to be totally uncorrelated. The off-diagonal matrix element describe the correlation between different noise sources in the state equations [85]. Therefore, it is common practice to assume that these matrices are diagonal [38,51,55,58] and lack of sufficient statistical information does not allow evaluation of their off-diagonal terms [86].

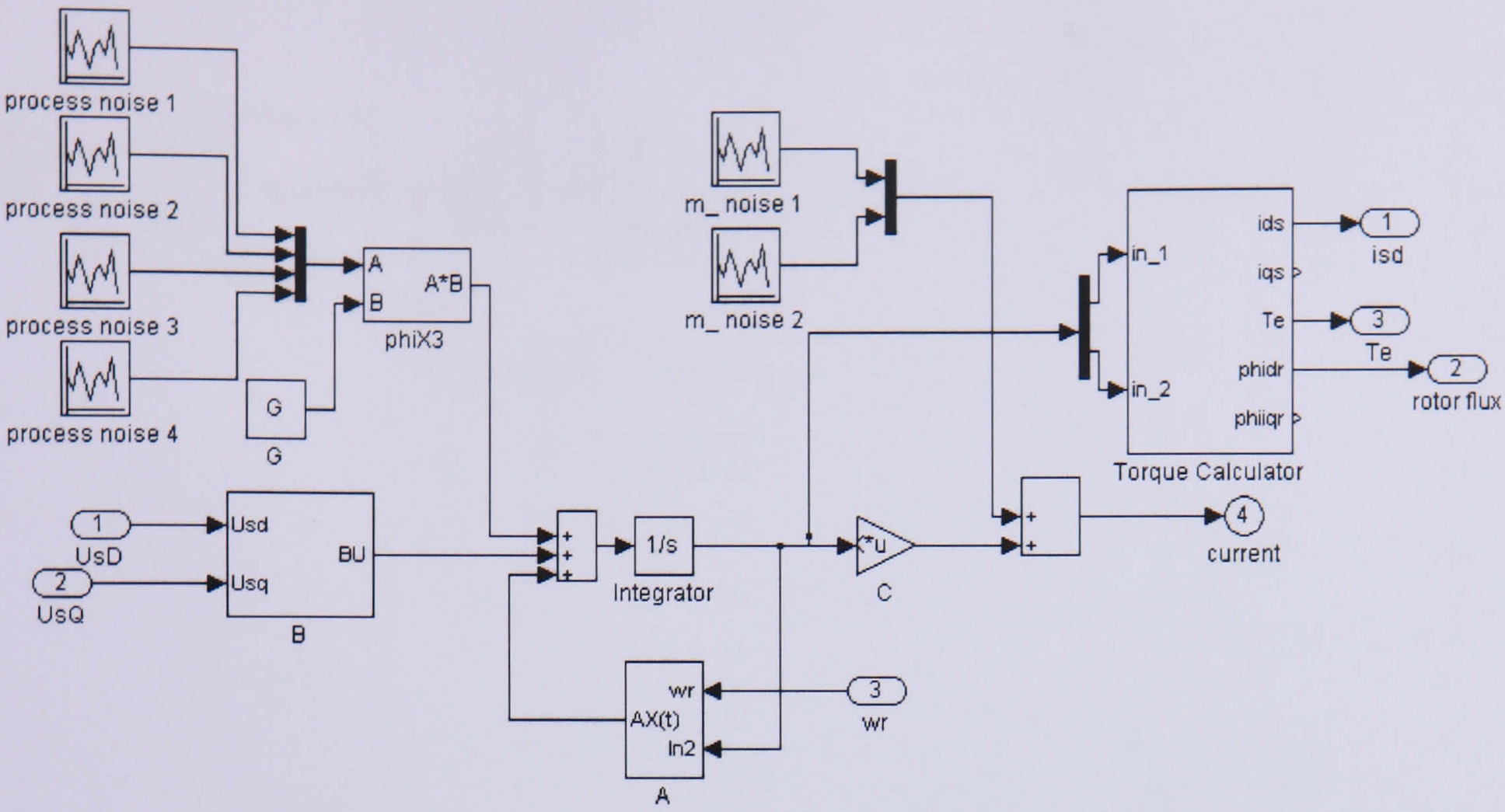


Figure 4.8: State space model of IM in the presence of noise

4.8 Observability

In an estimation application techniques such as the KF, the concept of observability is very important. The state space model of a plant system can be said to be observable if the measurement, $y(t)$, over a finite time, contains information which completely identifies the state $x(t)$. The observability matrix of a LTI system can be written as:

$$\left[C^T \quad A^T C^T \quad .. \quad .. \quad (A^T)^{n-1} C^T\right] \tag{4.18}$$

where **C** and **A** are the output and system matrices as already described. For the system to be observable, the rank of Eqn (4.18) must be equal to n where n is the number of the states.

4.9 Kalman Filter Estimator

Note that the KF is a model of the original system in discrete time and is driven by both the estimation error and the deterministic input. The observer has totally no idea about the process and measurement noise as in the model. Using that concept, the KF observer is modelled and shown in Figure 4.9. In the observer, the matrices **A** and **B** correspond to $\mathbf{\Phi}$ and $\mathbf{\Gamma}$ in Eqn (4.16). The gain, K needs to be optimised in order to give the optimum observer. In contrast to the common observer gain, the gain in KF is time varying. The subsystem K in Figure 4.9 contains the heart of the KF. The torque calculator block is necessary for providing information about the mechanical speed of the motor for use in the estimator.

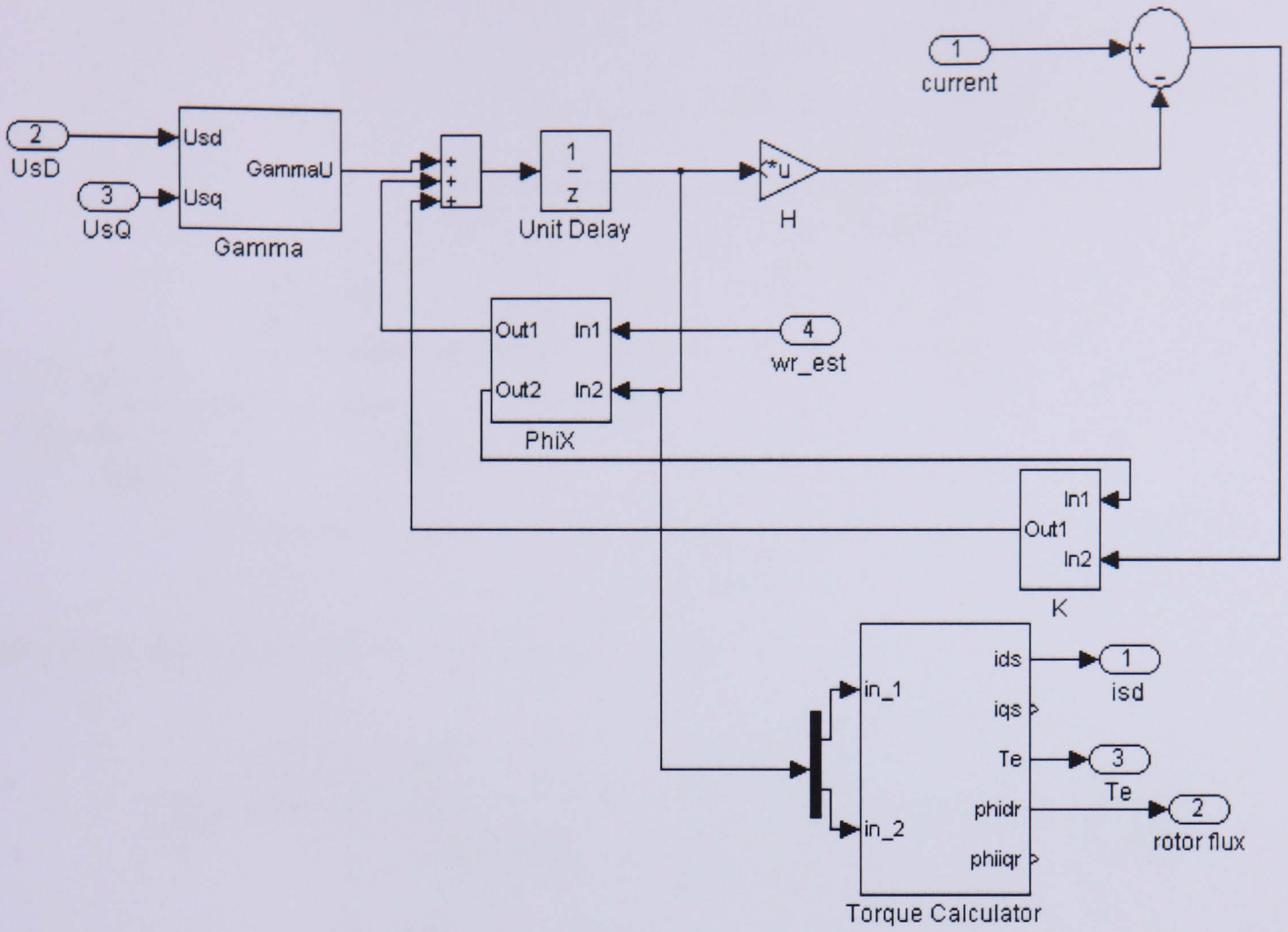


Figure 4.9: Simulation block for KF

The mask of the K subsystem is shown in Figure 4.10. This mask contains all the equations involved for filtering. Figure 4.11, Figure 4.12, Figure 4.13, and Figure 4.14 are the mask subsystems of Figure 4.10 and are based on Eqn (4.8)-Eqn (4.12).

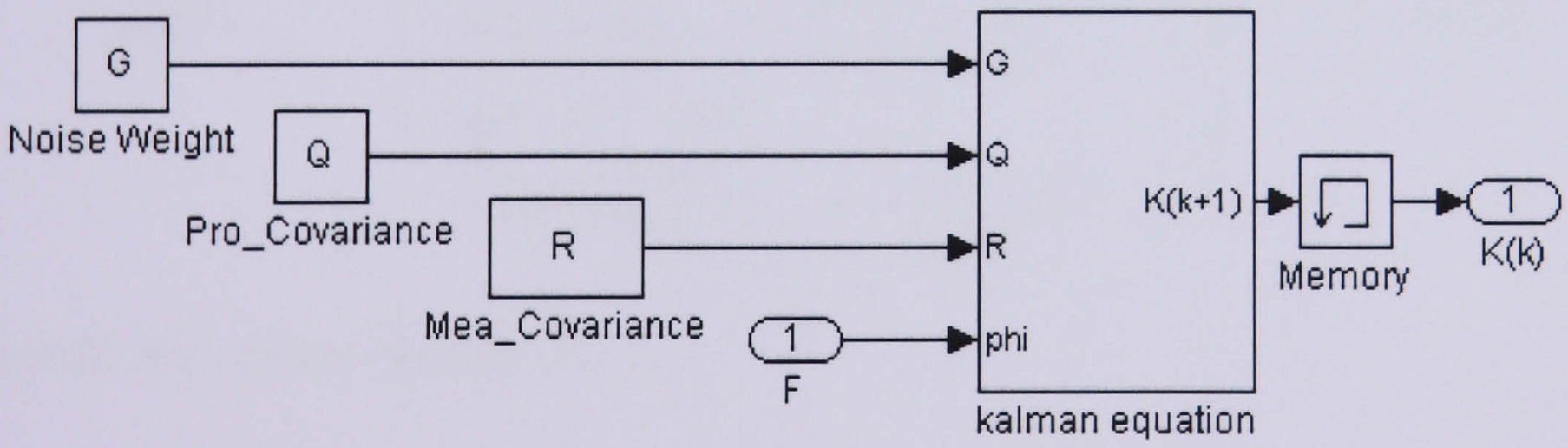


Figure 4.10: Simulation block of Kalman equation

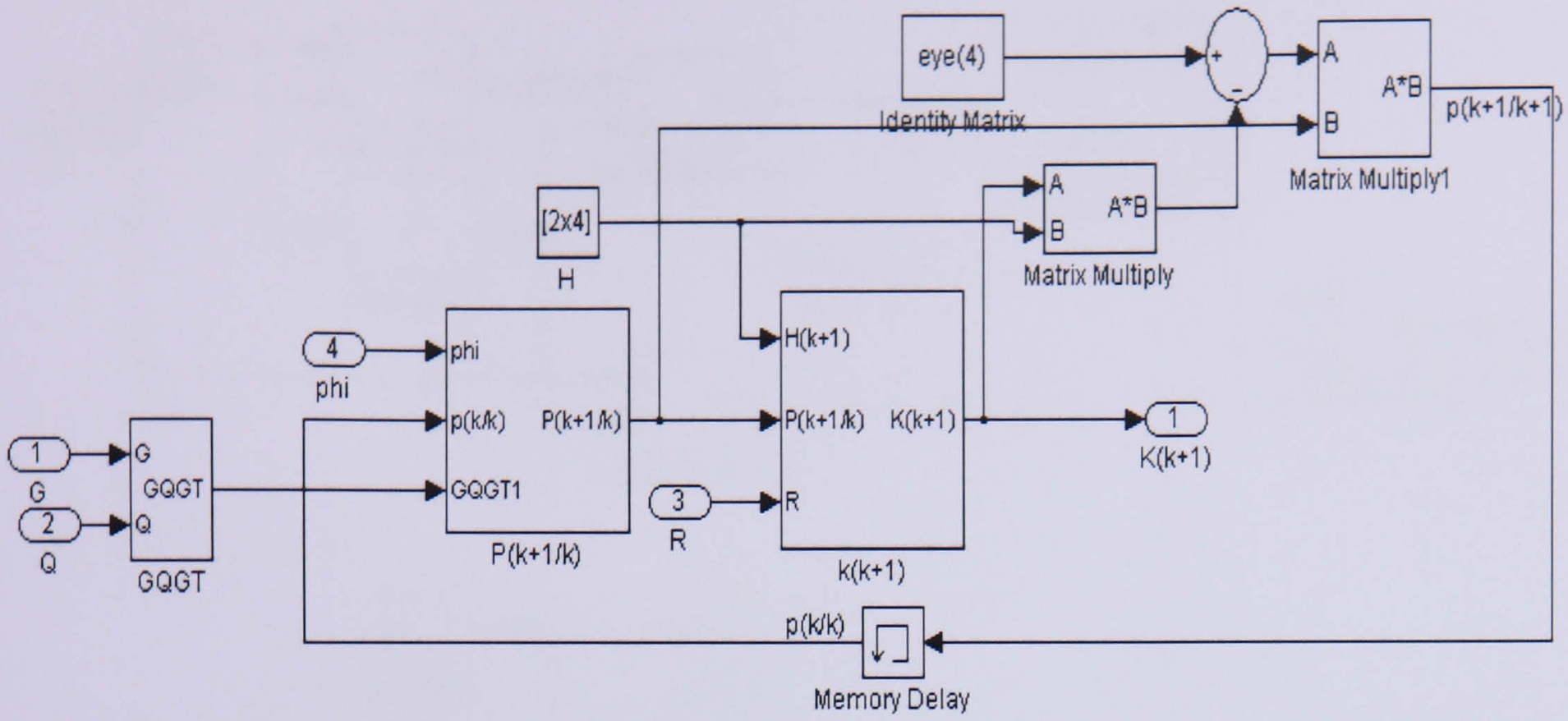


Figure 4.11: Simulation block of Kalman equation

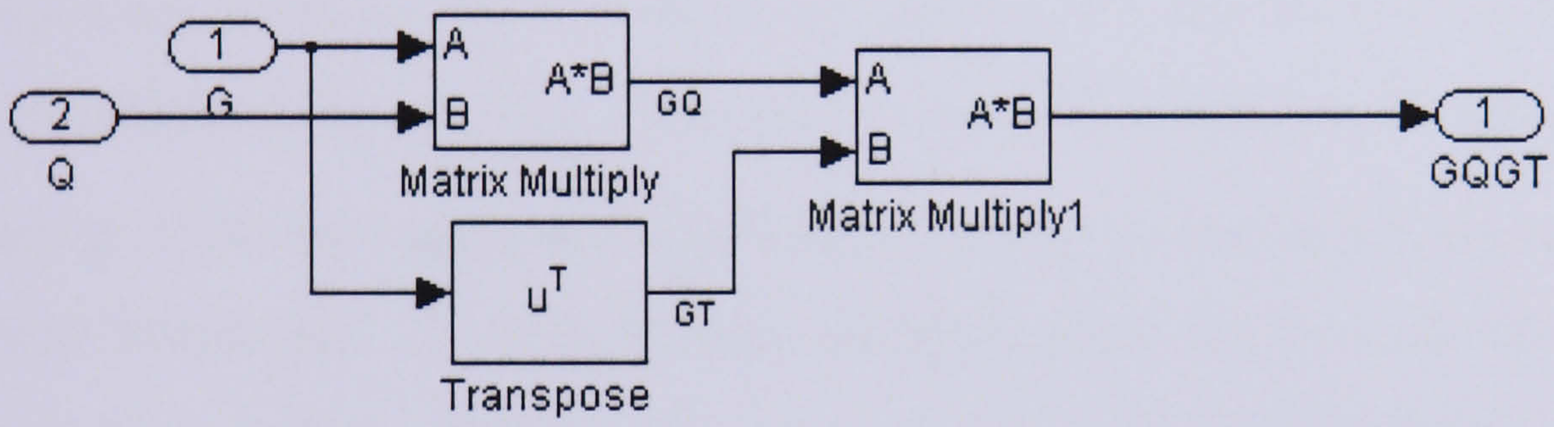


Figure 4.12: Simulation block of $\mathbf{GQG}^T$

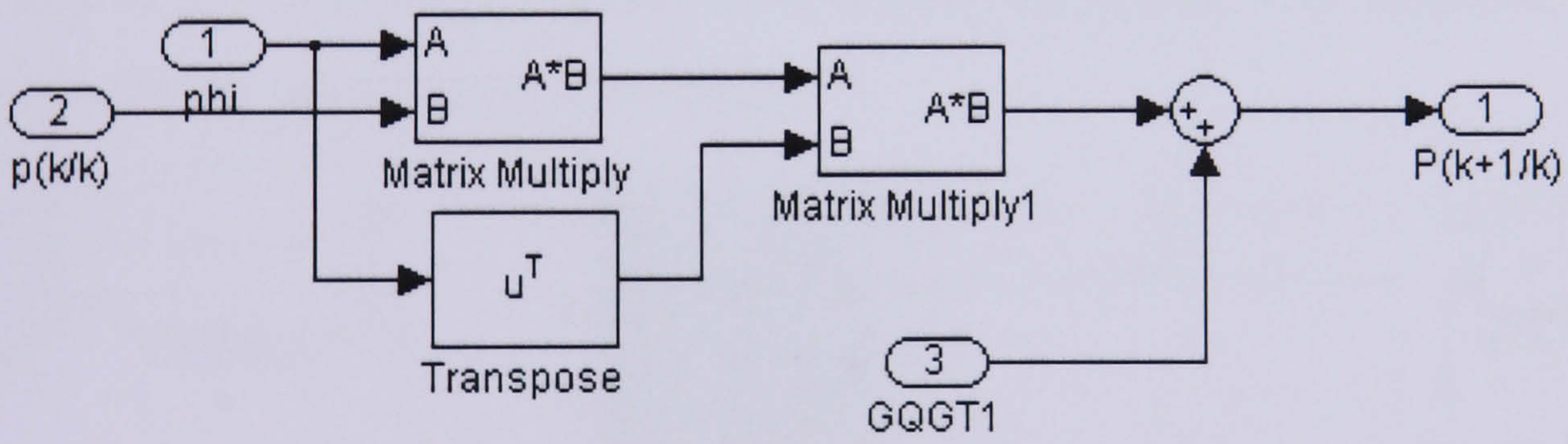
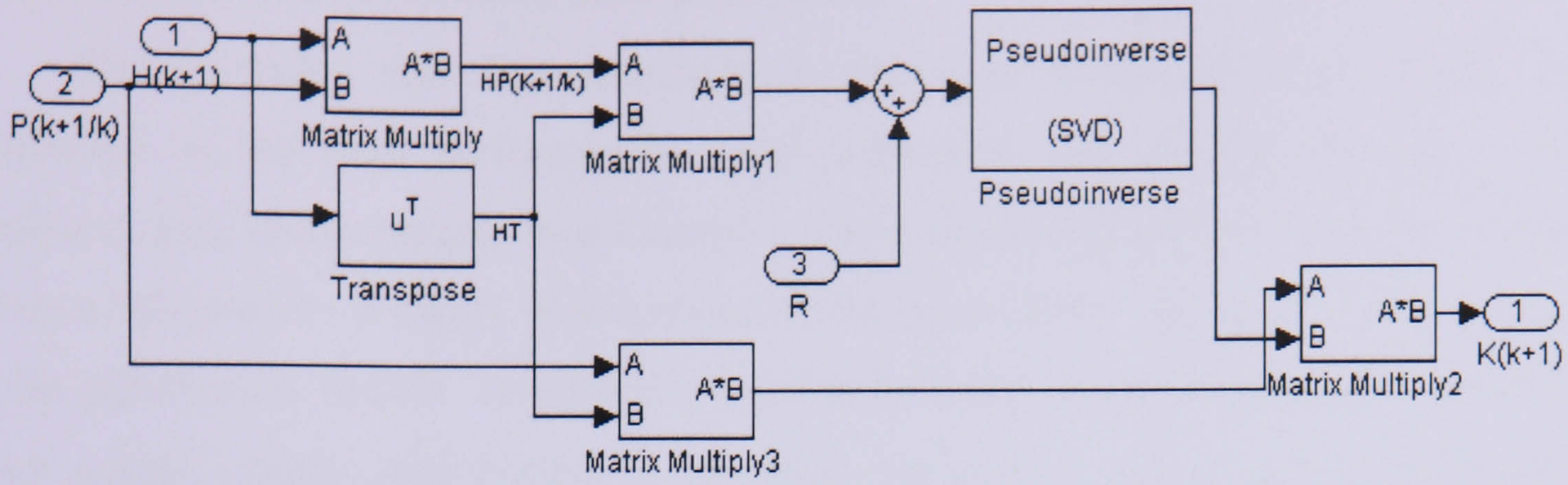


Figure 4.13: Simulation block of $\mathbf{P}_{k+1/k}$



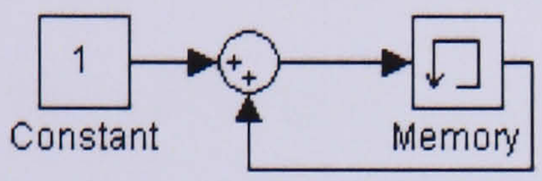


Figure 4.14: Simulation block of $\mathbf{K}_{k+1}$ with iteration number

The complete Simulink block is shown in Figure 4.15. The IM and the observer use the same control input U_{sD} and U_{sQ}. The electromagnetic torque can be calculated from the output current of the state space as in Eqn (3.14). Using the mechanical dynamics block, the speed can be determined. It would be more authentic to use ω_r from the IM model as if it was being measured by a sensor. However the KF is only claimed to be used for estimating currents and fluxes and not speed. Therefore the use of mechanical dynamics blocks 1 is only to give useful information for determine the speed. The estimated speed is then compared with the actual speed from the IM.

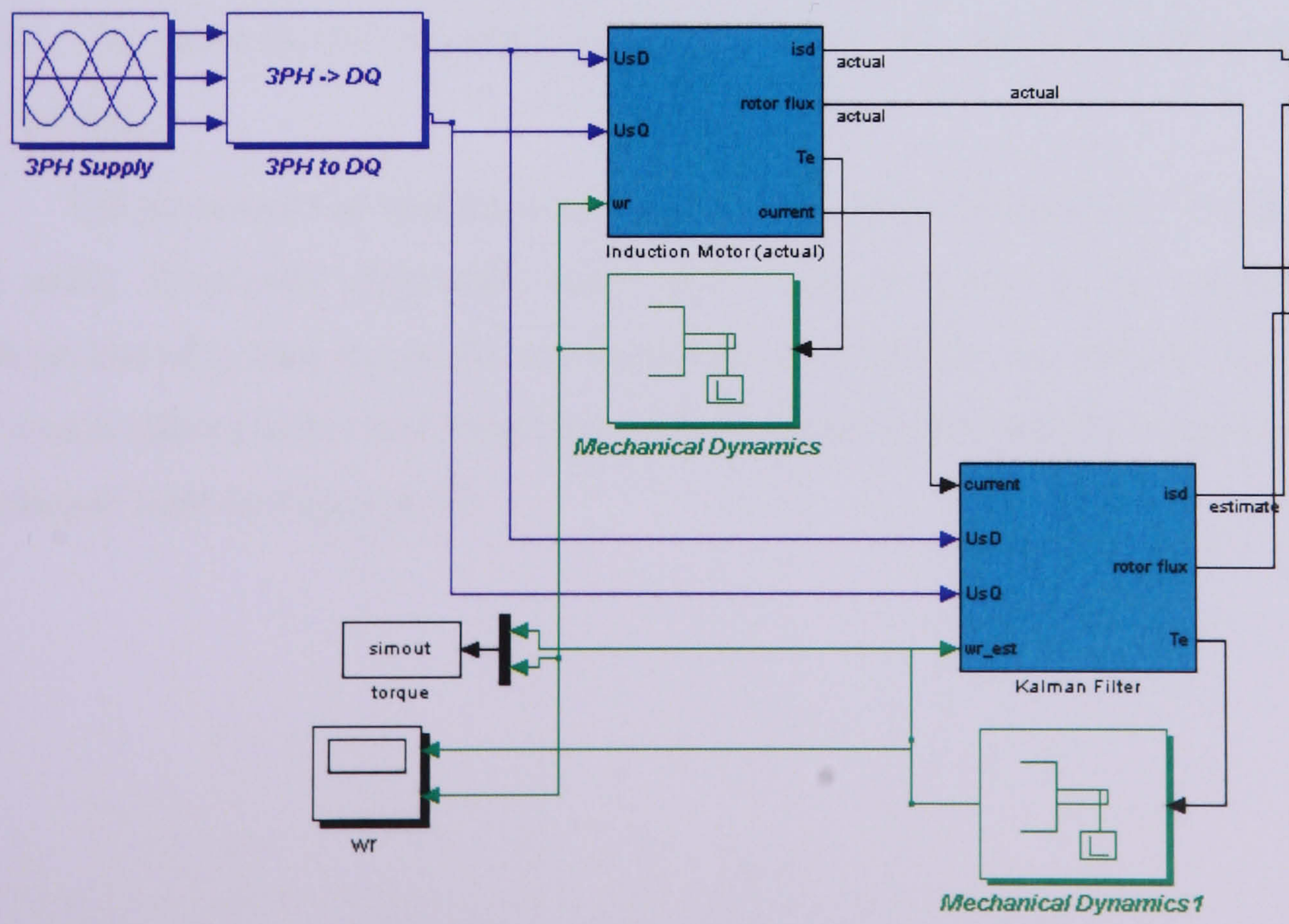


Figure 4.15: Simulation block of IM with KF Estimator

4.10 Simulation of Kalman Filter Estimator

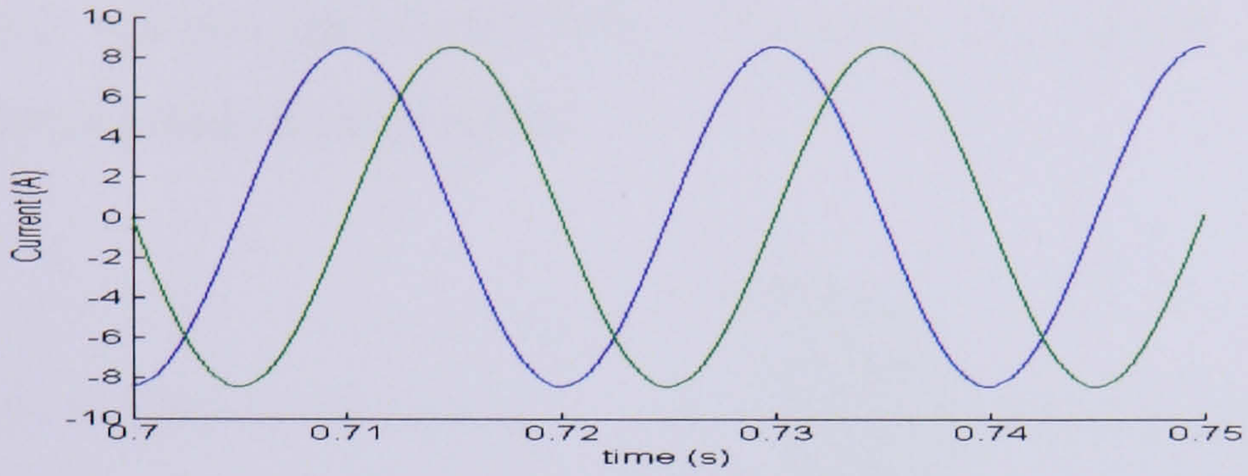
The algorithm was first simulated to determine what influences of the system parameters on the filter performance. The simulation was carried out over a 0.75 s simulation time using 3-phase direct supply. The voltage and current were for a transient which is followed by a steady speed; however the trace shown in Figure 4.16(a, b) is only for the steady state period. In addition, a Gaussian white noise is separately added to the motor model, voltage and current, to simulate the process and measurement noise and shown in Figure 4.16 (c-e). The process noise uses the corresponding noises covariance matrix $\mathbf{Q} = 4x10^{6}\mathbf{I}_4$, the measurement noise $\mathbf{R} = 0.01\mathbf{I}_2$, and the noise gain matrix used is $\mathbf{G} = 0.01\mathbf{I}_4$, where $\mathbf{I}$ is the 4x4 identity matrix. The measurement noise using was a typical value found [55].

The KF starts from rest with the IM accelerating from rest to 157 rad/s. The initial state condition is set to zero while the initial error covariance is $\mathbf{P_0} = 10\mathbf{I}_4$. Varying P yields different amplitudes of the transient however the duration of both the transient and steady state will not be affected [86].

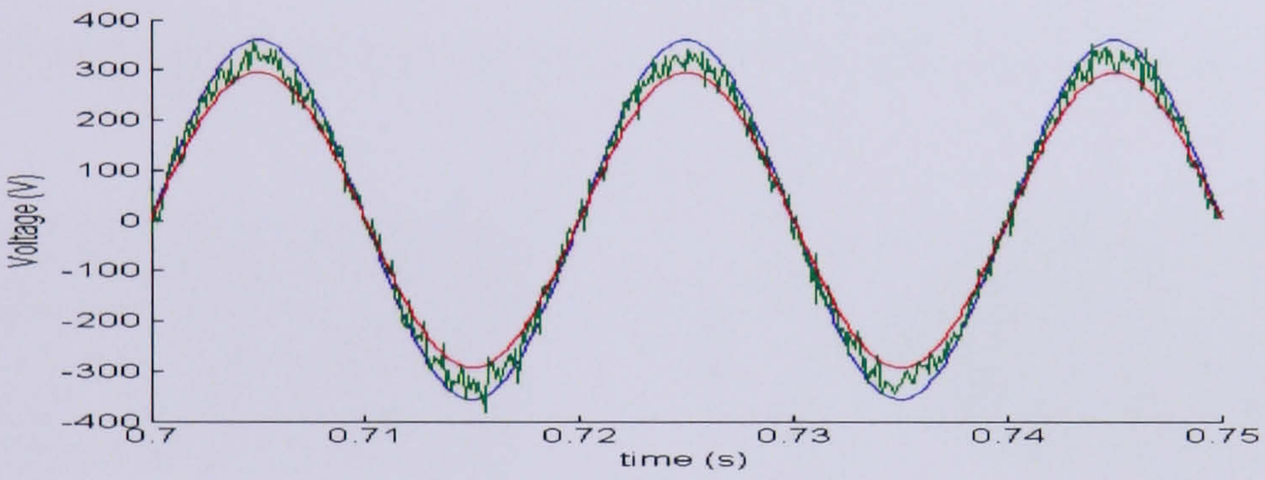
In the first simulation assumes that the motor is not disturbed with any noise. No noise sources, process and measurement are added to the motor model. In Figure 4.17(a-b) the current and flux, actual and estimated, are given together. To obtain the desired transient and steady state behaviour, the noise covariance matrix is tuning by Trial and Error. The value is chosen as $\mathbf{Q} = 0.0001\mathbf{I}_4$, $\mathbf{R} = 0.01\mathbf{I}_2$ and the noise gain matrix used is $\mathbf{G} = 0.01\mathbf{I}_4$.

The process noise and measurement noise are then added to the model, which in this case using 10 percent of process noise, and a typical value for the measurement noise. Without knowing that the model has increased uncertainties the estimator performed well. The d-axis stator current and rotor flux, both from the model and the estimated, can be seen in enlarged form in Figure 4.17.

(a)

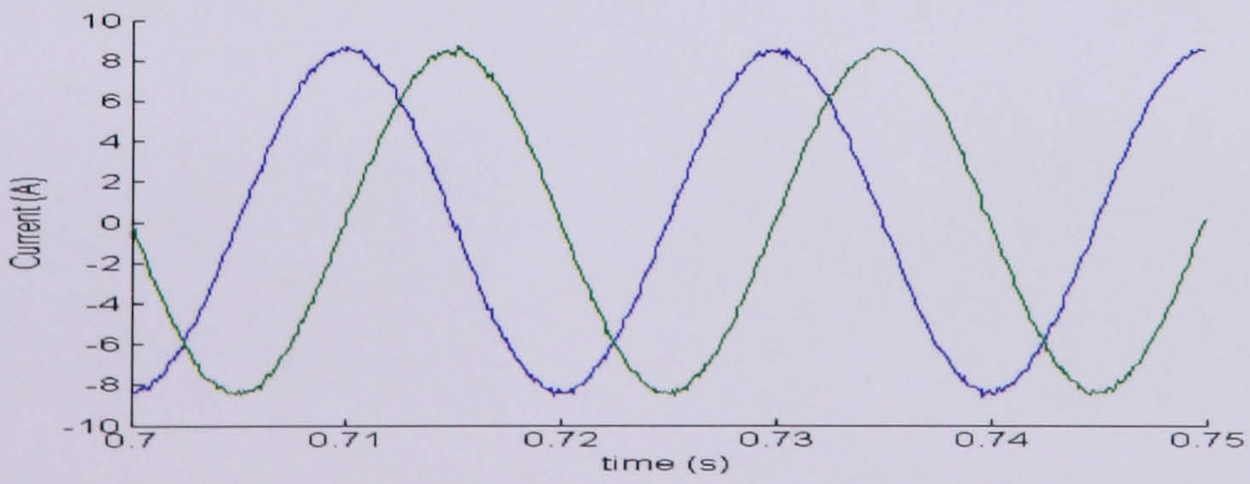


(b)



(c)

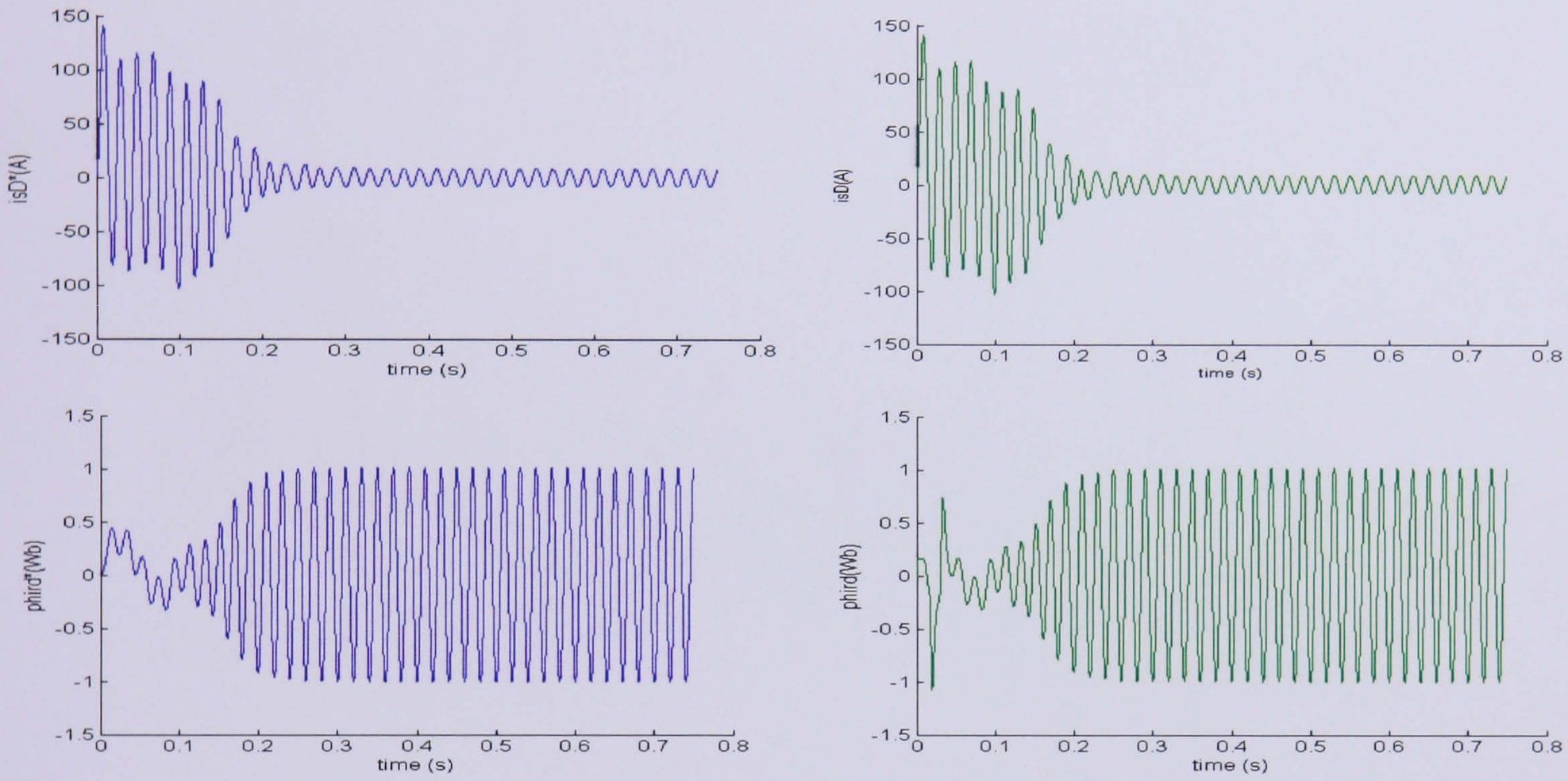
(d)



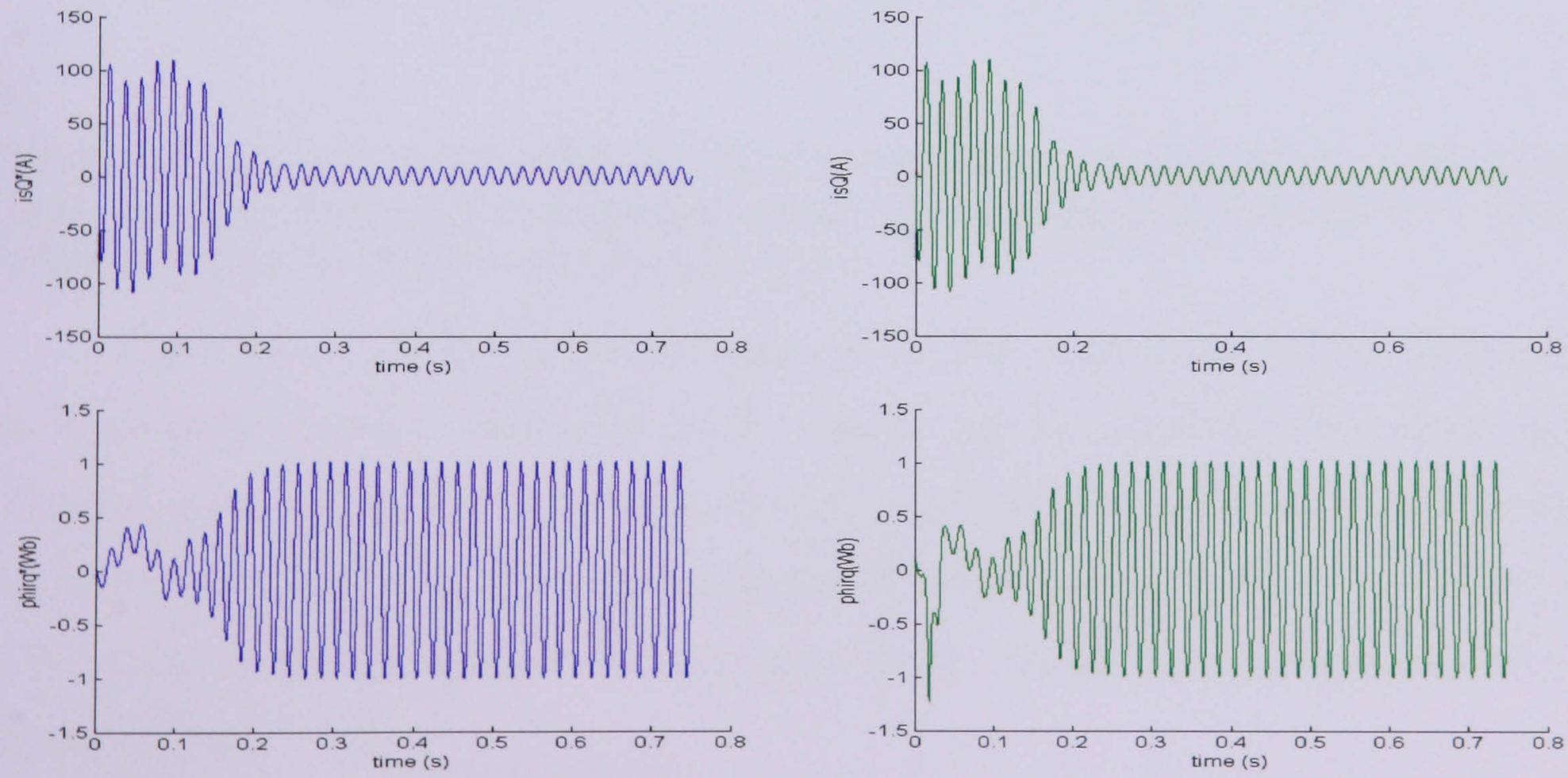
(e)

Figure 4.16: (a) Simulated voltages for IM running at steady speed, 157 rad/s. (b) Simulated current, ids, for IM running at steady speed, 157 rad/s. (c): The noise variation showing in the 10% envelope for the stator voltage representing the process noise, Q. (d) The stator current resulting from the process noise, Q. (e) The stator current resulting from the measurement noise, R

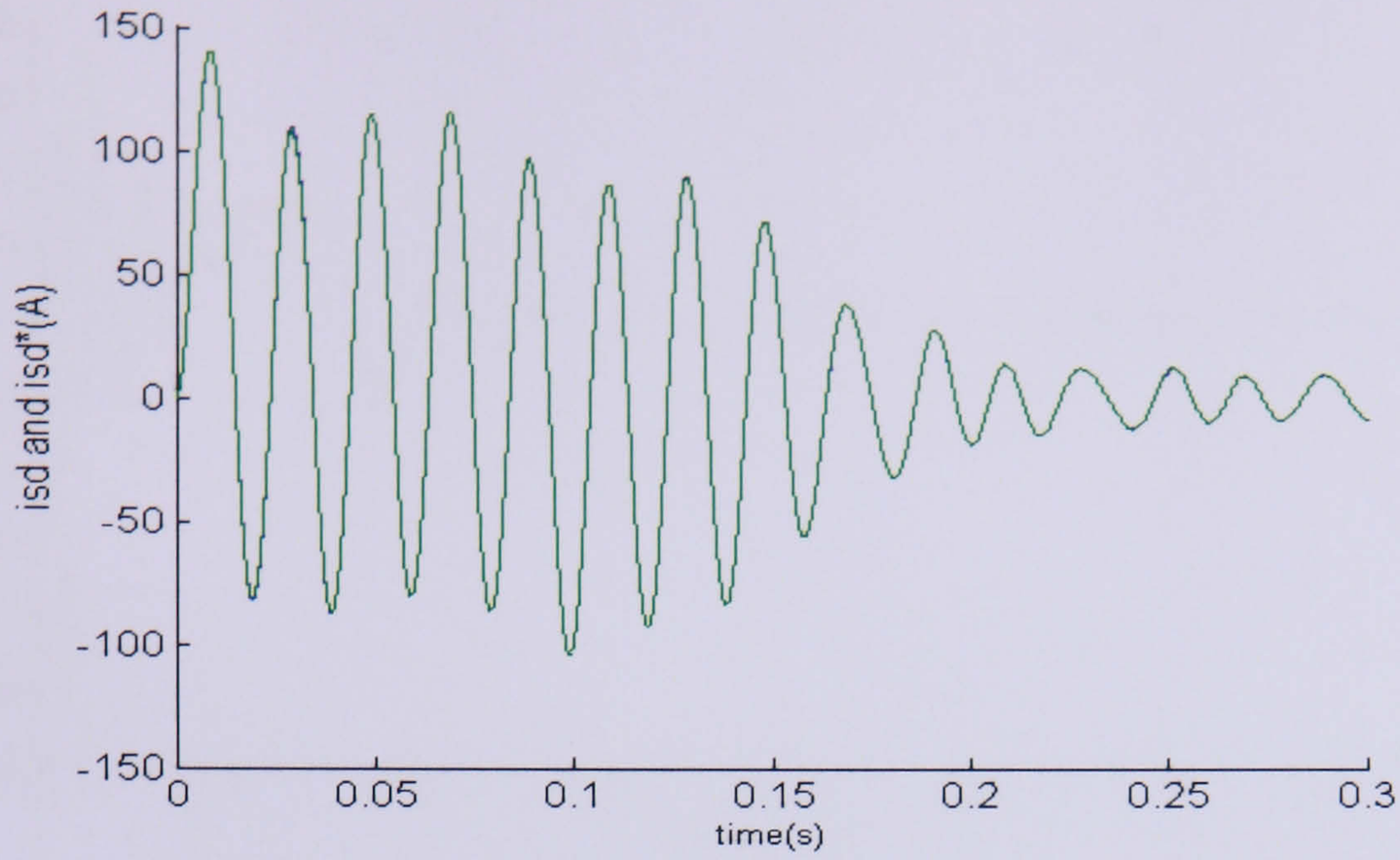
The closeness of the estimate to the actual states is shown in Figure 4.17(c,d). Due to the inertia of the motor from standstill, those current amplitudes increase during the transient and reach a steady state at about 0.25 s. The estimated currents also have the same motor transient condition as in actual currents. For flux estimation during the starting operation the rotor is initially stationary and this causes difficulty in the initial flux estimation. However, as the motor starts and the rotor begin to move the estimated flux almost immediately reaches the actual flux. Therefore the current and flux estimator responses can be considered as satisfactory.



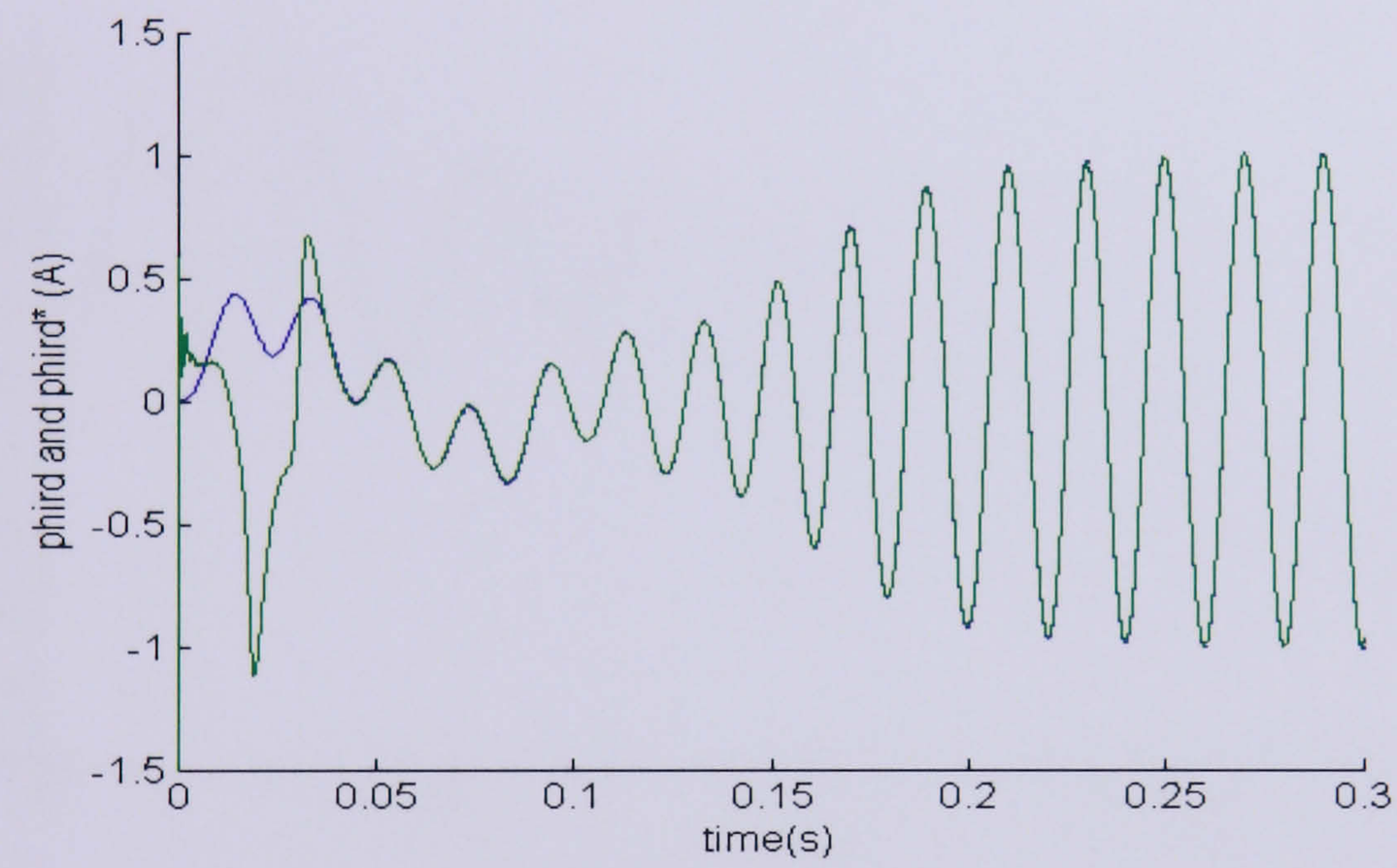
(a)



(b)



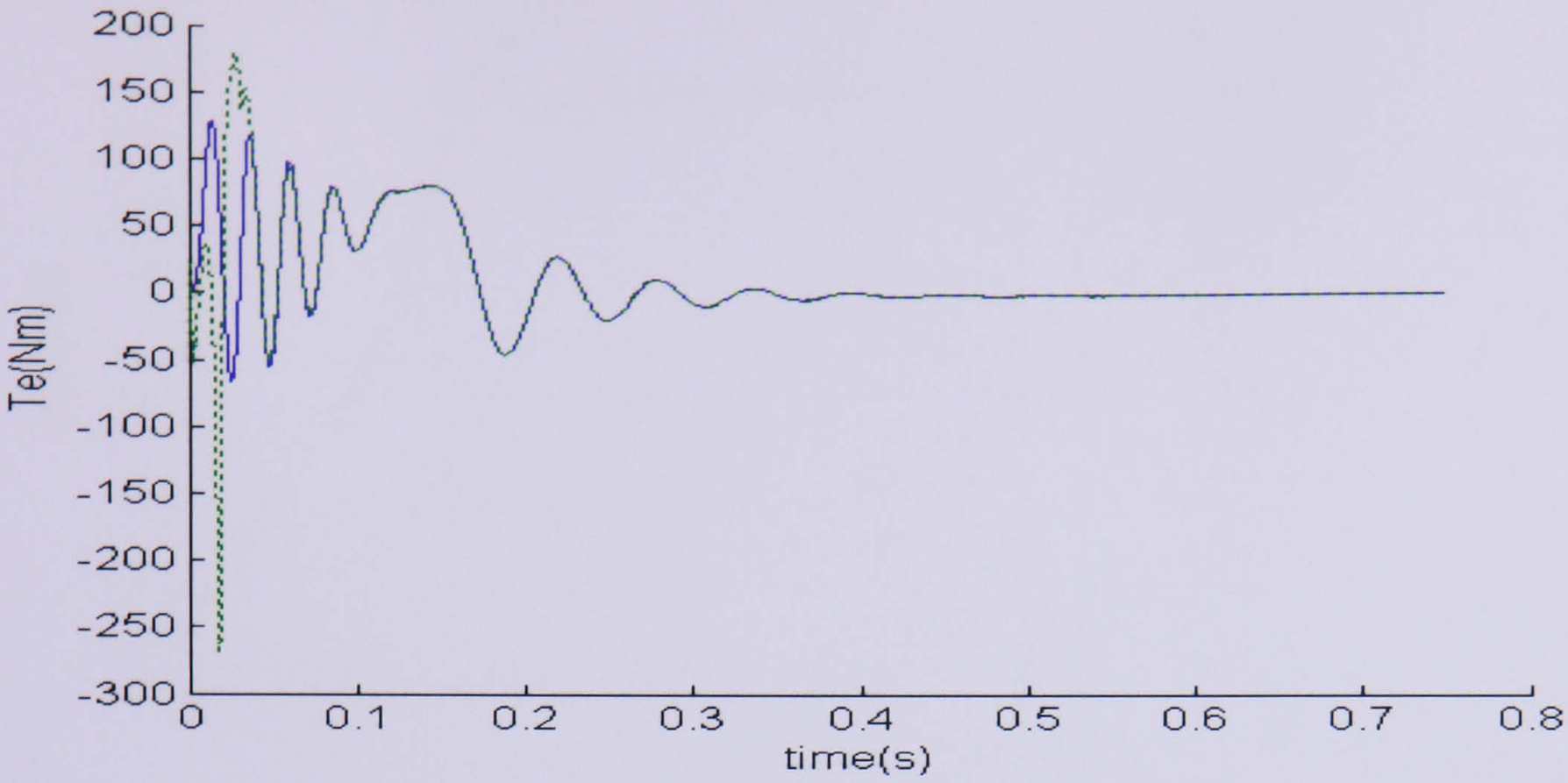
(c)



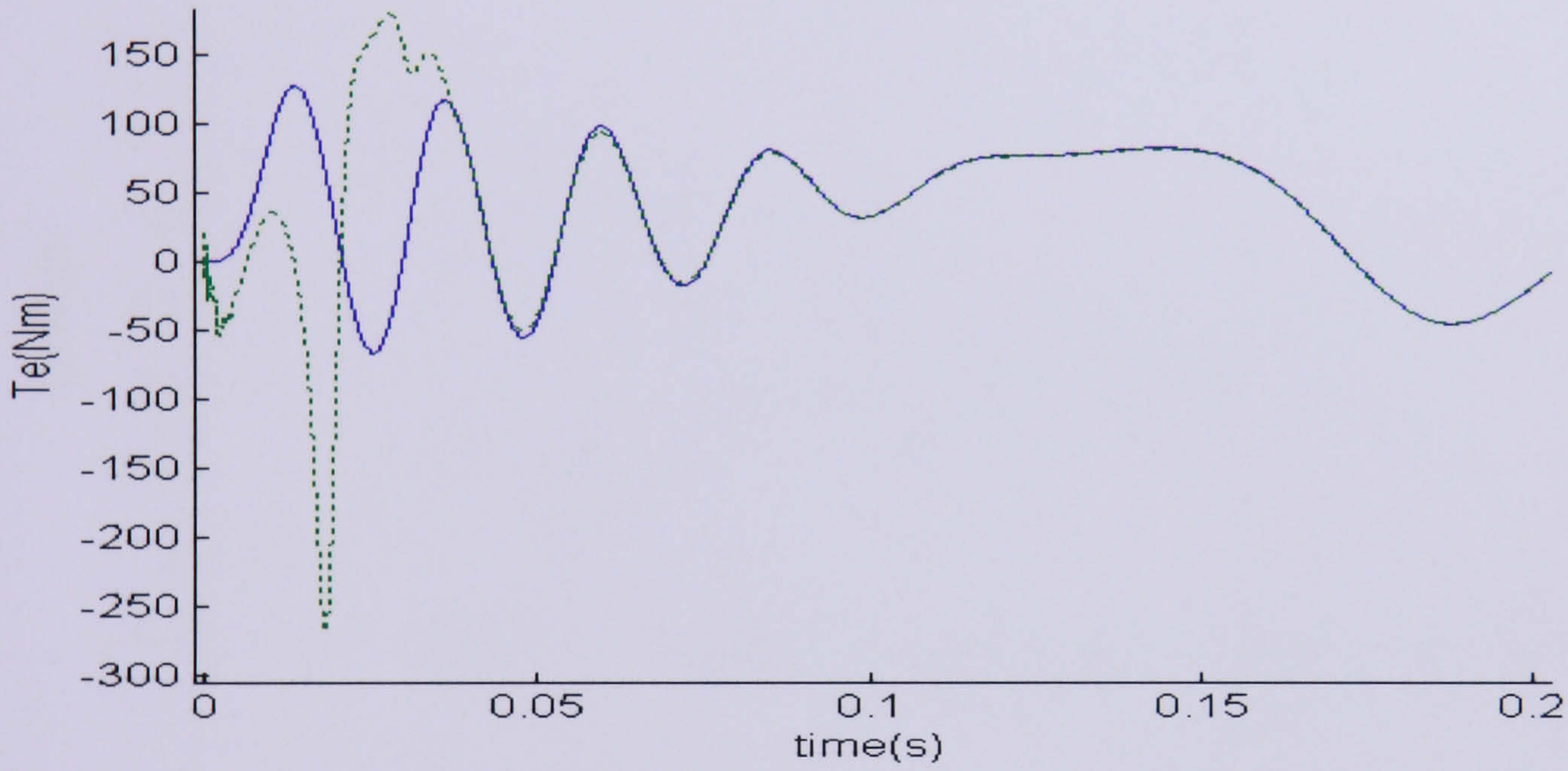
(d)

Figure 4.17: (a) d-axis of stator current and rotor flux with no uncertainty (b) q-axis of stator current and rotor flux with no uncertainty in motor model (c) and (d), the d-axis of stator current and rotor flux is shown in enlarged form

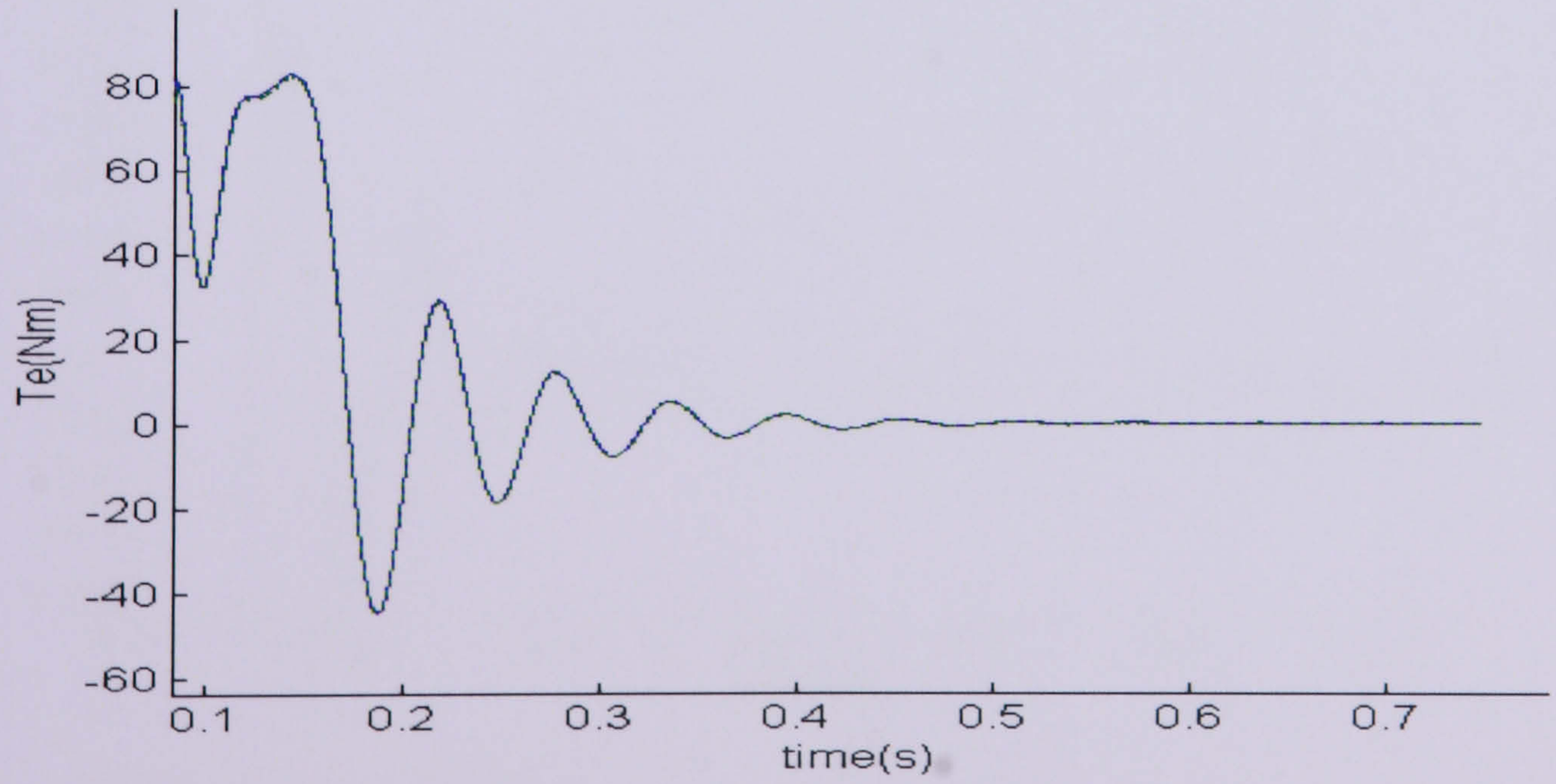
The torque response is shown in Figure 4.18. The large torque pulsations are due to the large motor inertia. Since the motor torque can be calculated, the speed can be estimated using Eqn (3.14). The estimated speed and reference speed are plotted together in Figure 4.19. It is also shown in the enlarged form that during transient and steady state there are only slight difference to the estimated speed. Hence at least in this idealised case the method works well.



(a)

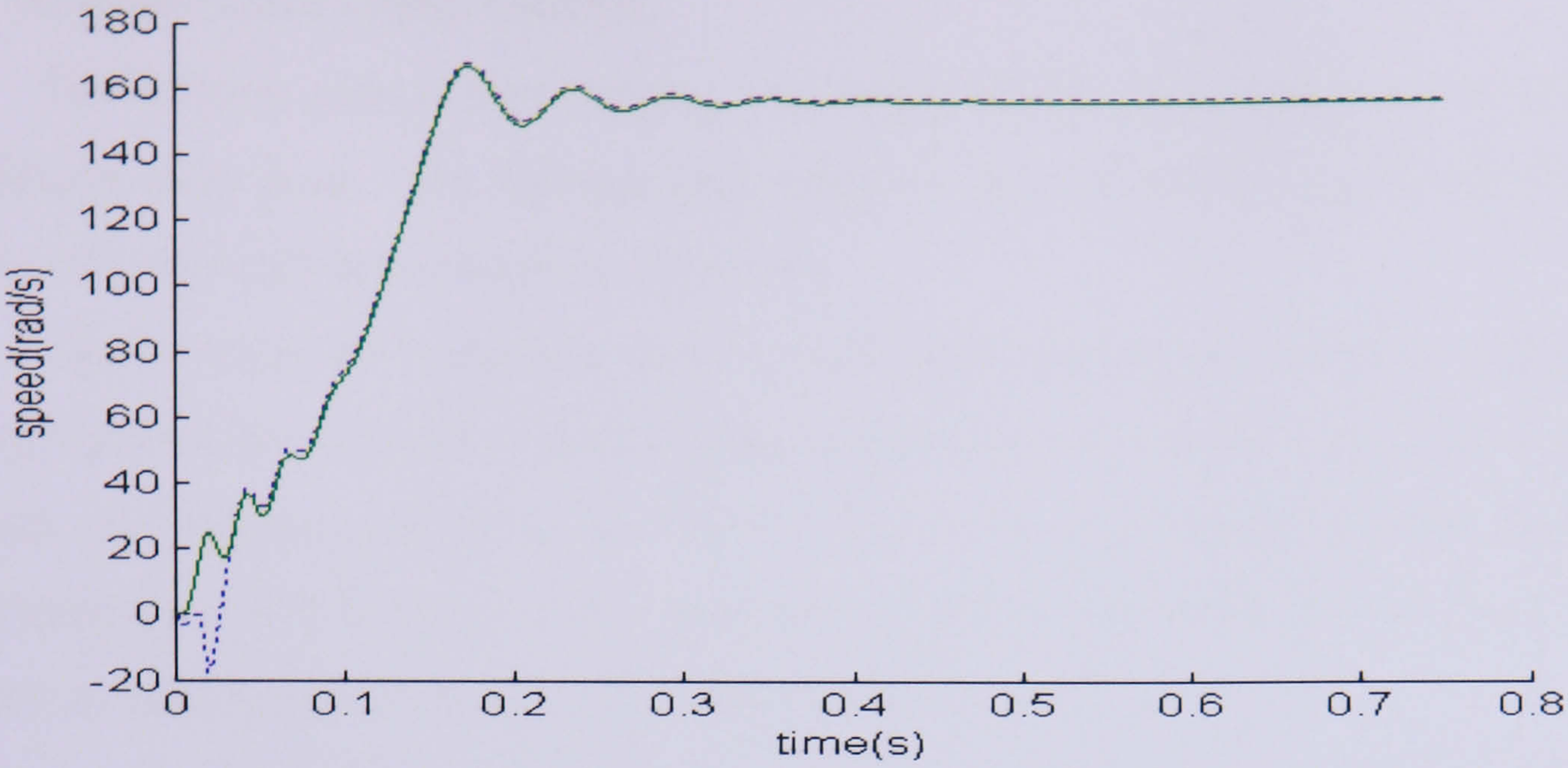


(b)

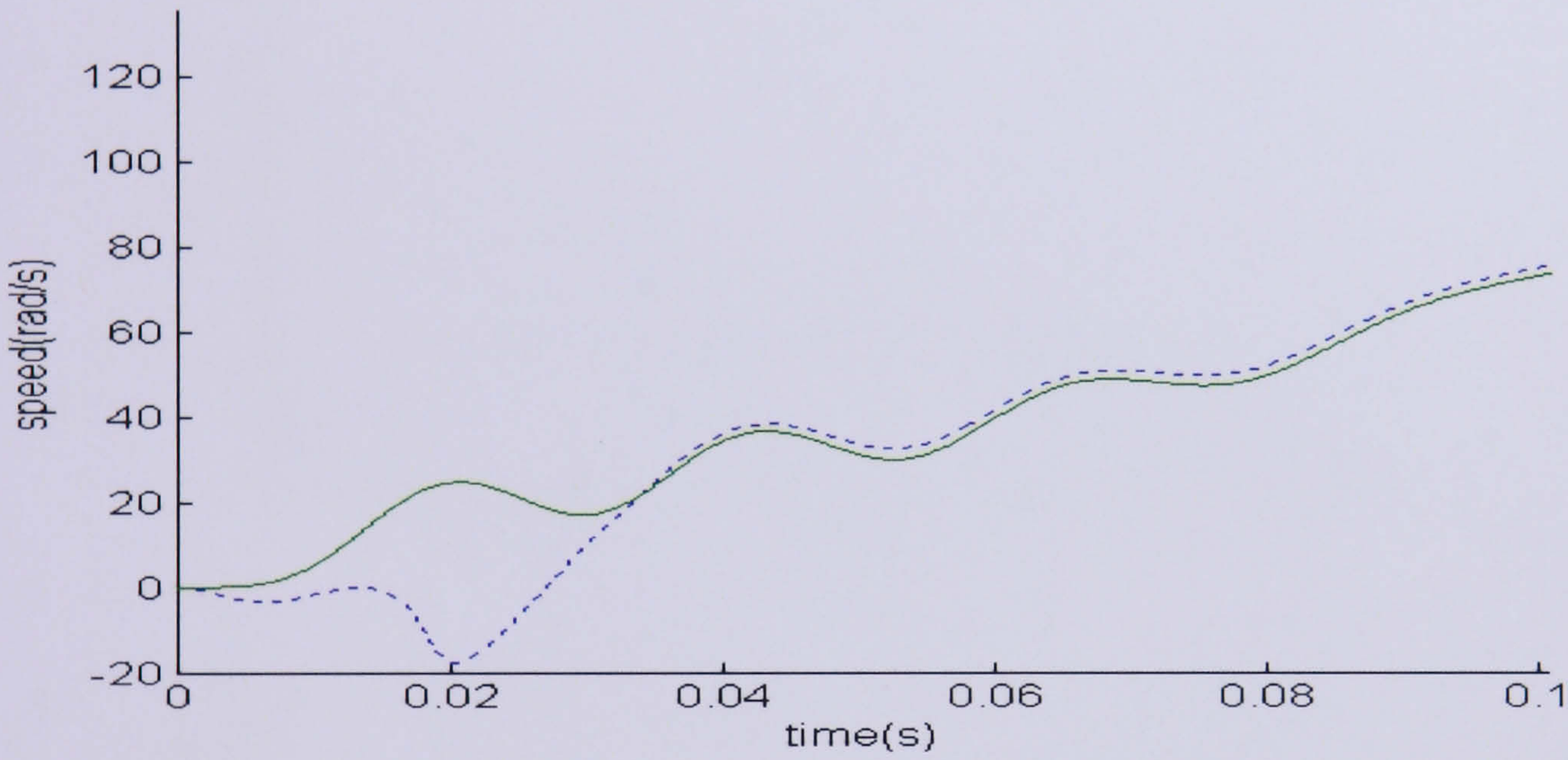


(c)

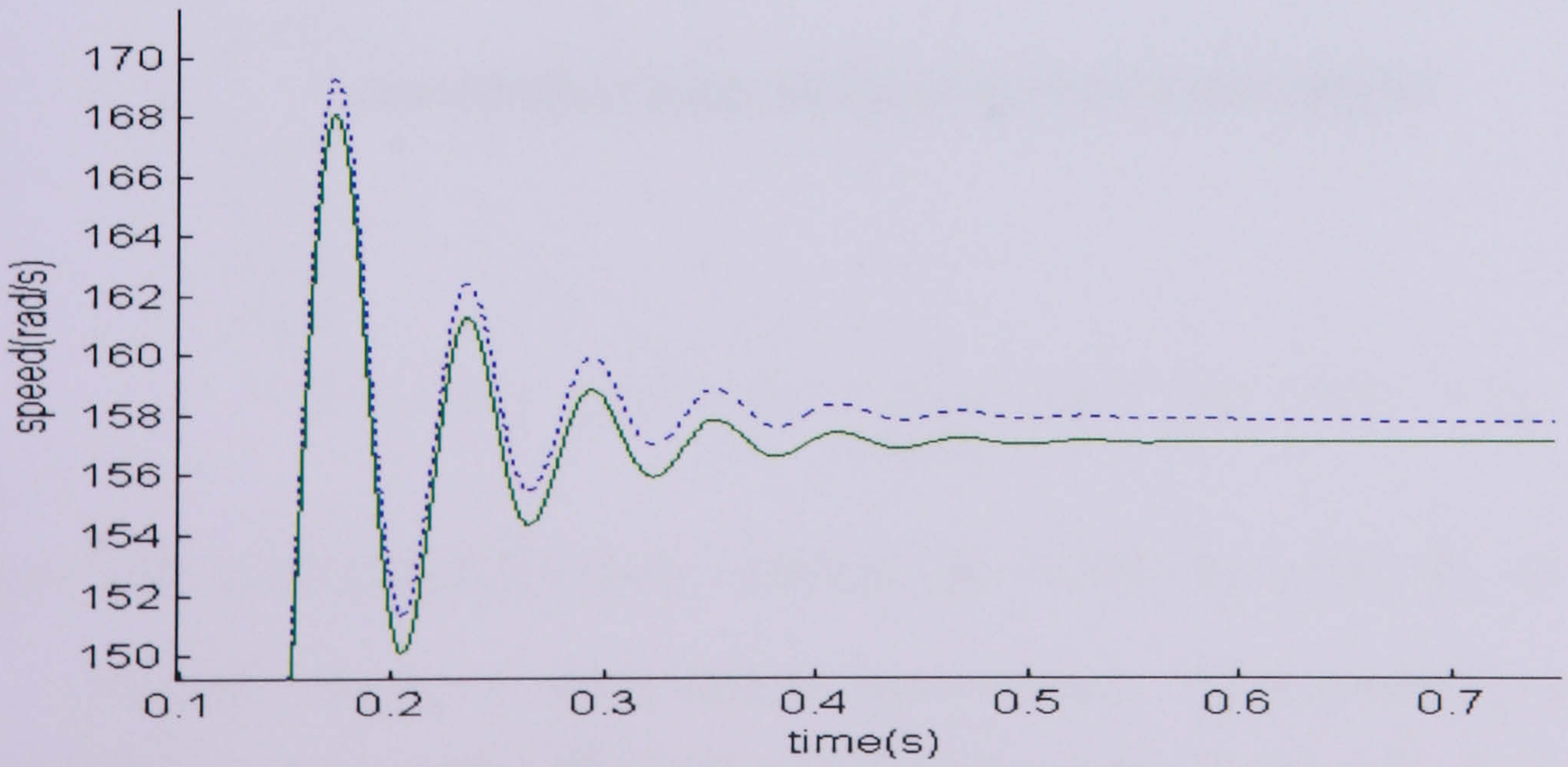
Figure 4.18: (a) Motor torque, (b, c) motor torque in enlarged form. The dotted line is the estimated and the straight line is the actual speed



(a)



(b)



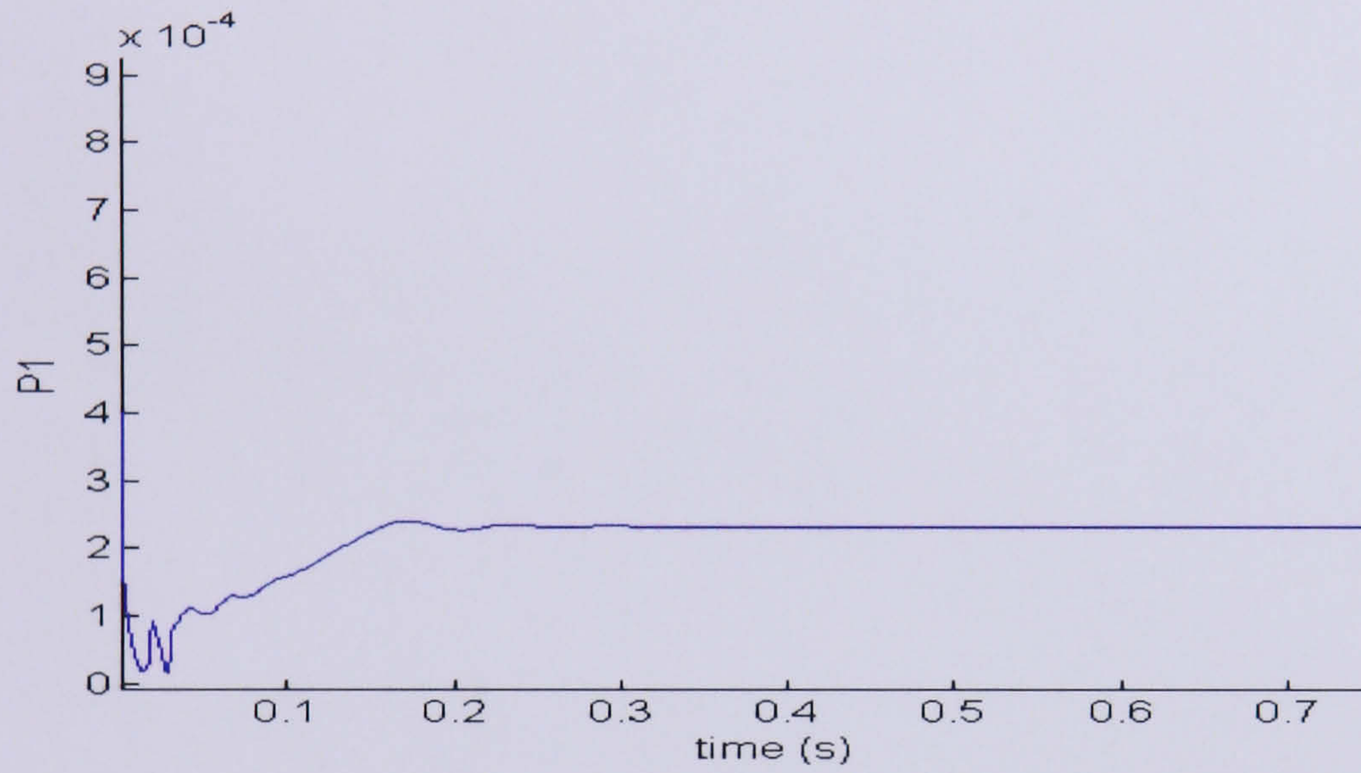
(c)

Figure 4.19: (a) Speed estimation of KF during transient and steady state (b) Speed estimation during early start up condition (c) Speed estimation approaching steady state condition. The dotted line is the estimated while the straight line is the actual speed

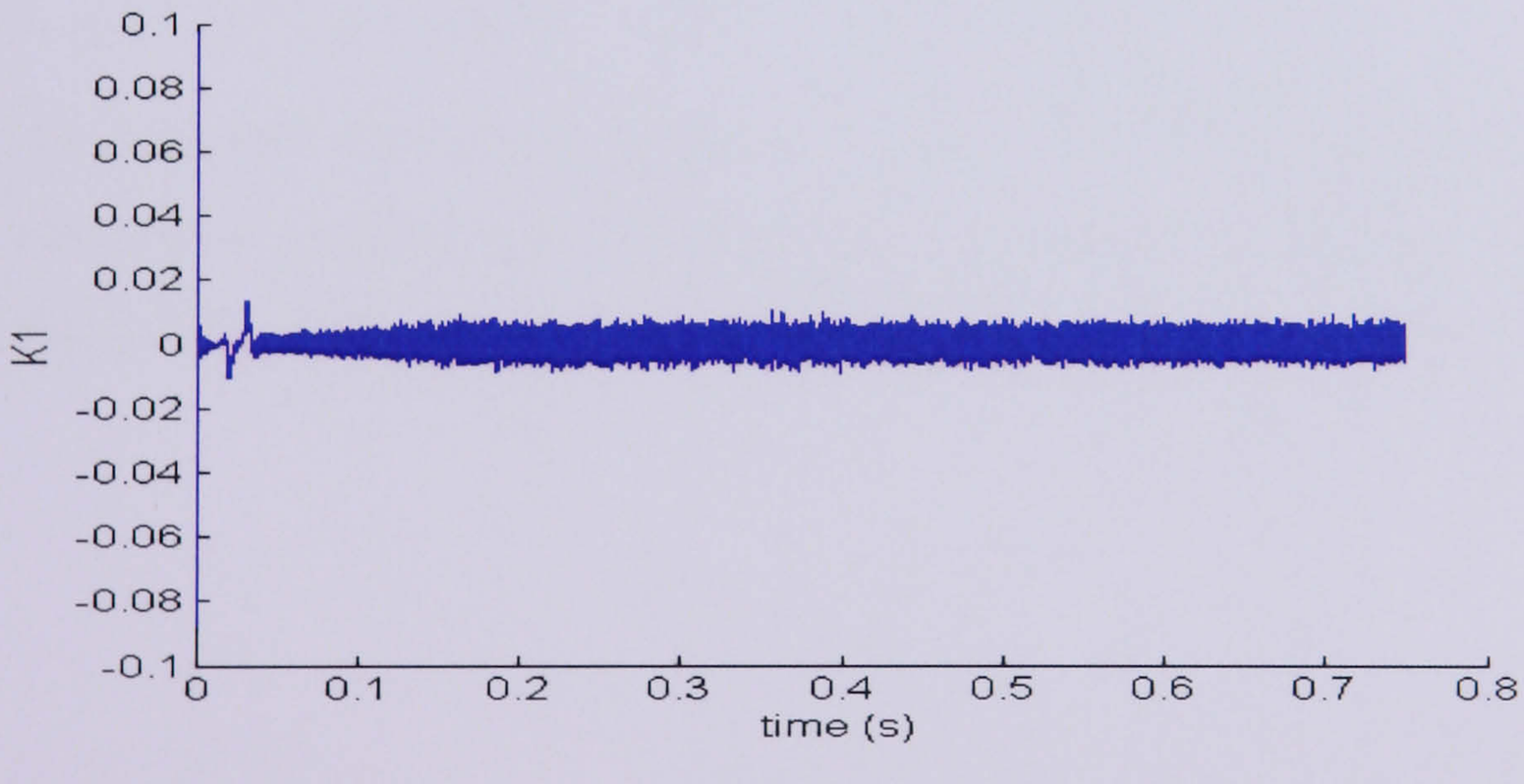
4.11 Kalman Gain Observations

The Kalman gain is time varying. To verify the KF gain feature, several additional simulations were done. The Kalman gain matrix is 4x2. However in this simulation only K_1 (the first row and first column) is observed.

Firstly, using the conditions such as initial state, initial error covariance, **Q**, **R** and **G** as stated before, the element of the KF gain is observed. As the error covariance is large at the start, the Kalman gain shown in Figure 4.20 is also starts large but then drops as the priori estimate is trusted more. The P_1 increases to reflect the correction of the error. It has reached an approximately steady state solution after about 0.2 s.



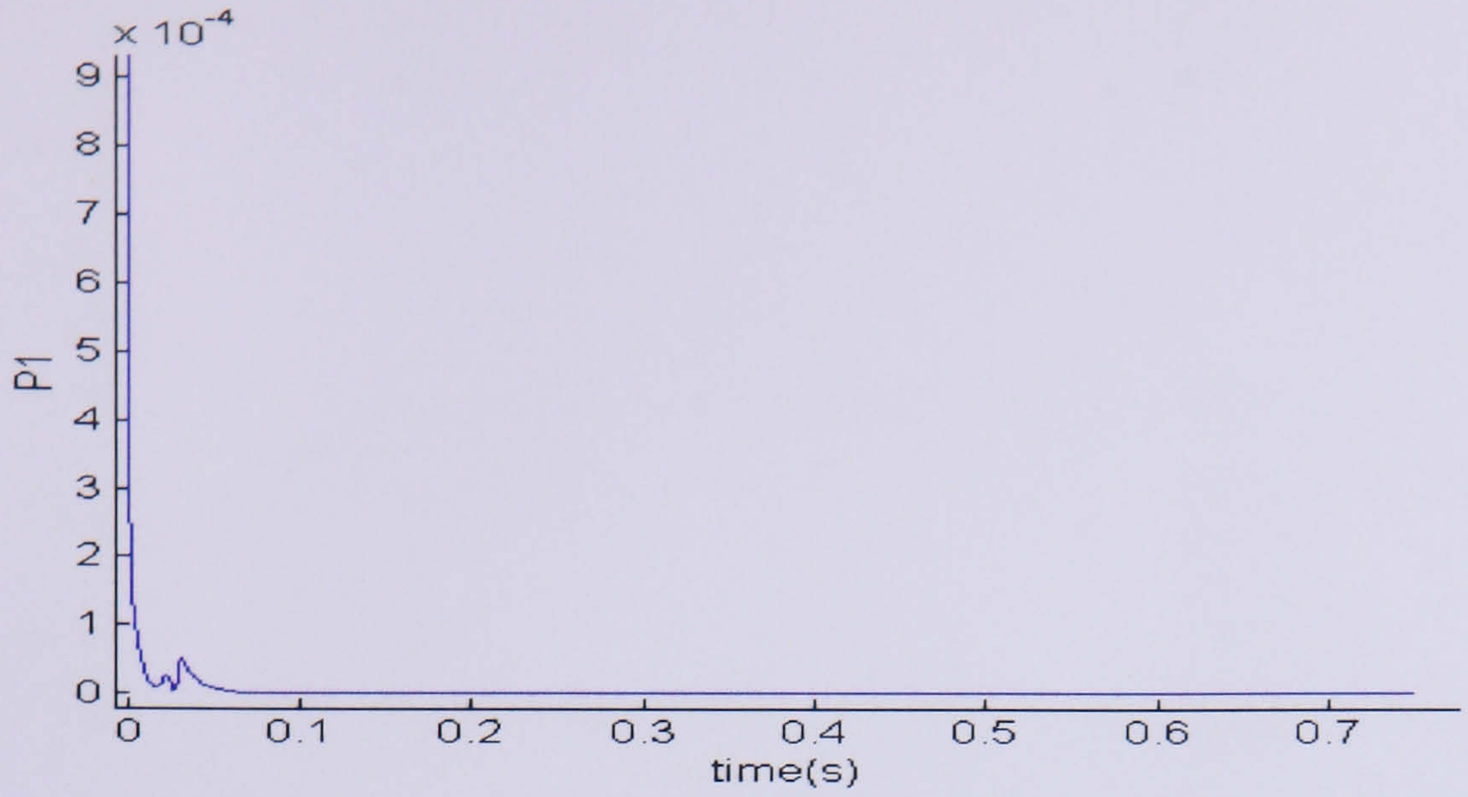
(a)



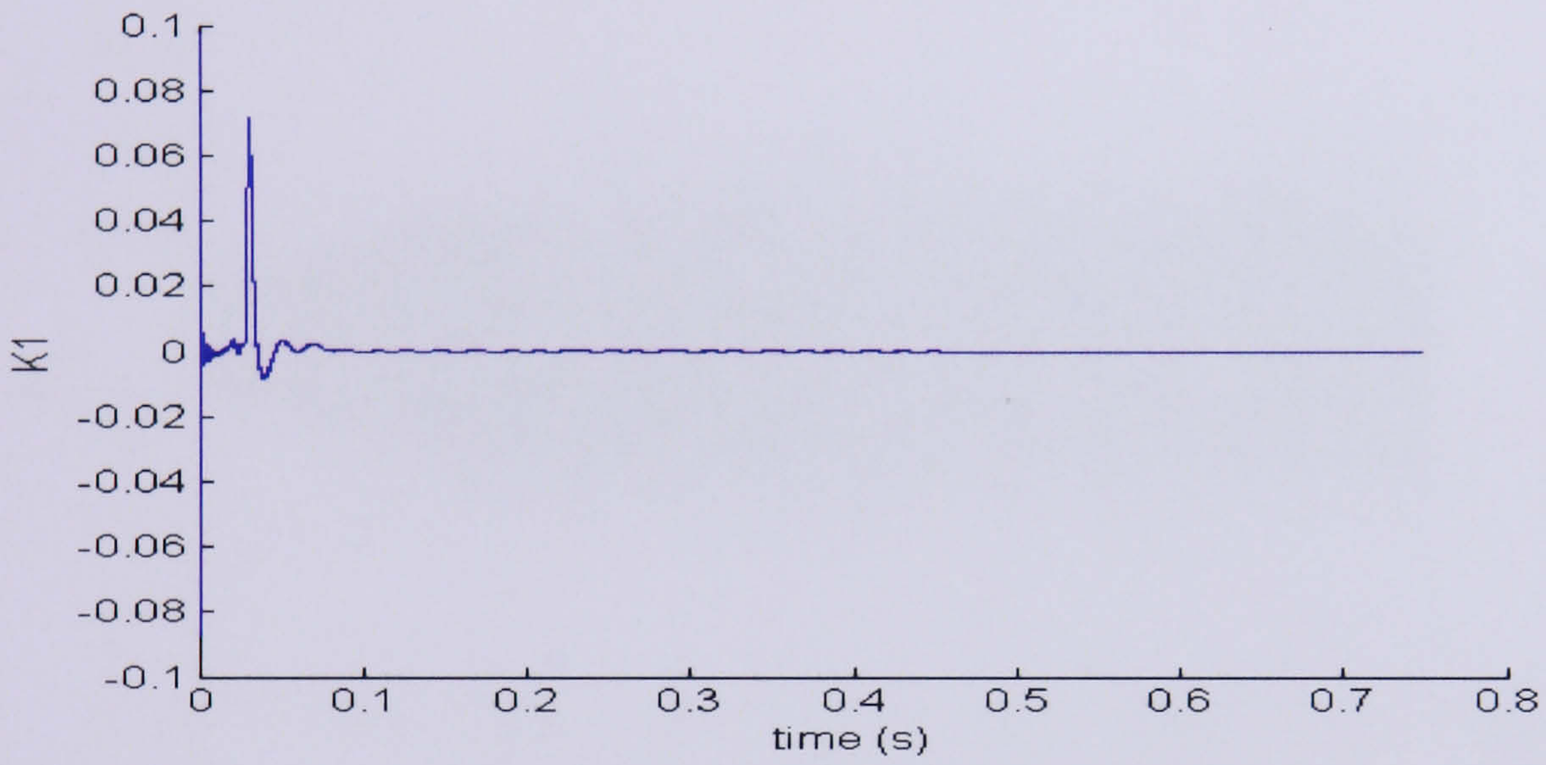
(b)

Figure 4.20: (a) P1 and (b) K1 for $\mathbf{Q} = 0.0001\mathbf{I}_4, \mathbf{R} = 0.01\mathbf{I}_2, \mathbf{P}_0 = 10\mathbf{I}_4\ \mathbf{X}_0 = 0$

Secondly, the KF is tested with no process noise. Other conditions such as initial state, initial error covariance, **R** and **G** were kept the same. Figure 4.21 shows that with no process noise added to the system, the error covariance starts as a large value but drops very quickly. As the priori estimate is trusted more, little needs to be done to correct the error. The Kalman gain drops to zero.



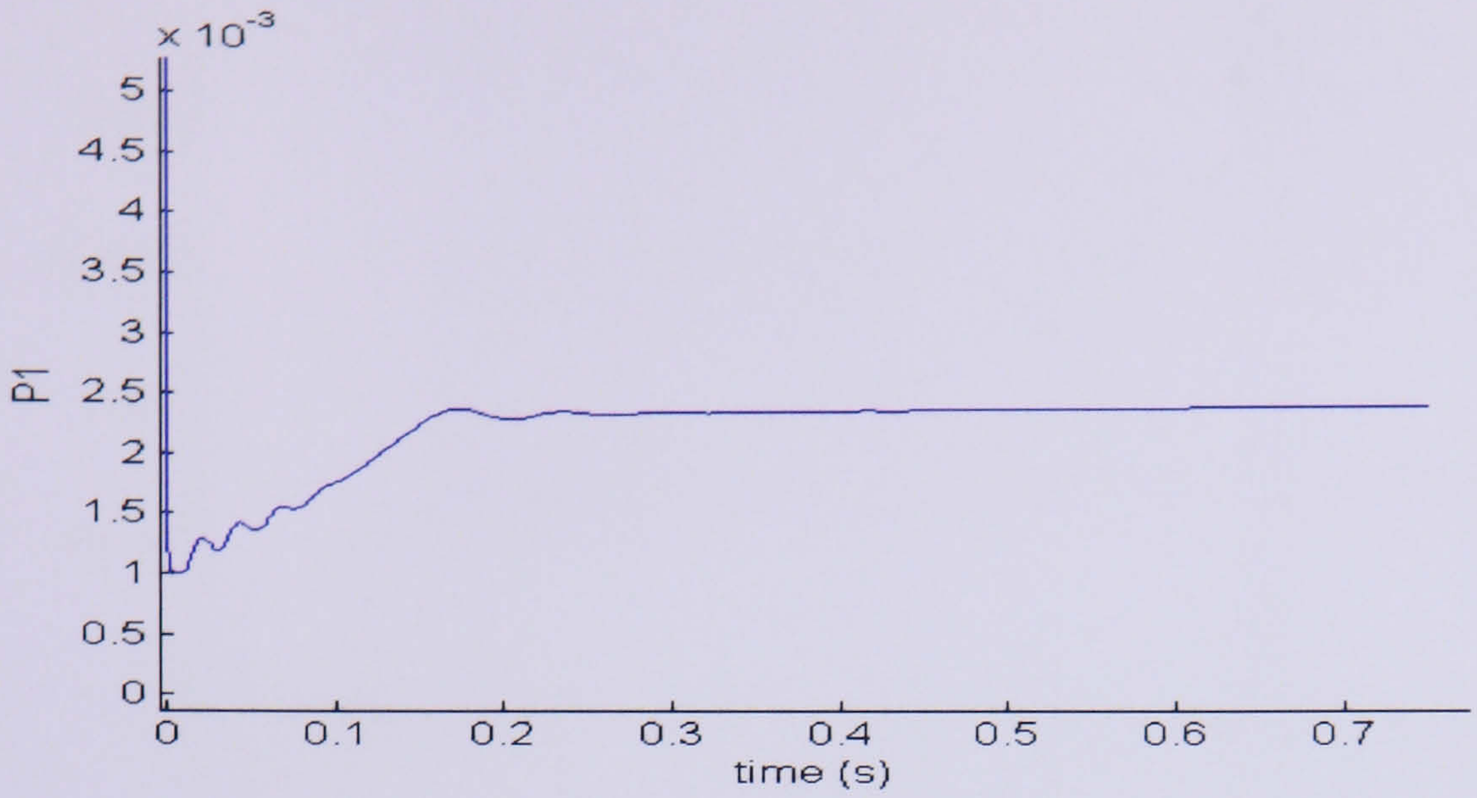
(a)



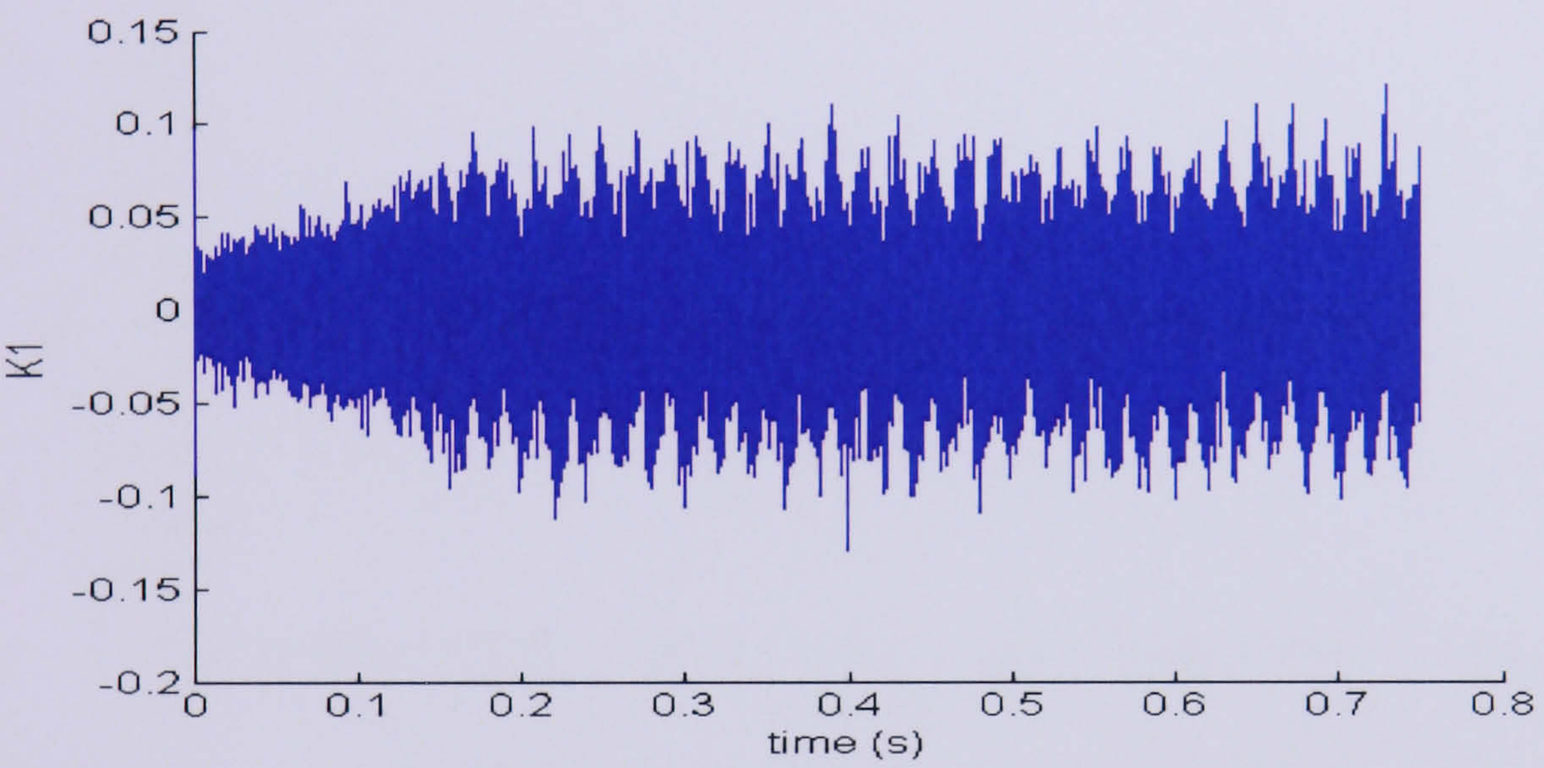
(b)

Figure 4.21: (a) P_1 and (b) K_1 for $\mathbf{Q} = 0, \mathbf{R} = 0.01\mathbf{I}_2$, $\mathbf{P}_0=10\mathbf{I}_4$, $\mathbf{X}_0=0$

In the third case the process noise is increased. The P_1 in Figure 4.22 is dropped as the priori estimate is trusted more but does not drop as low as the case without process noise. As the noise is added to the state at each step time, a larger correction is needed.



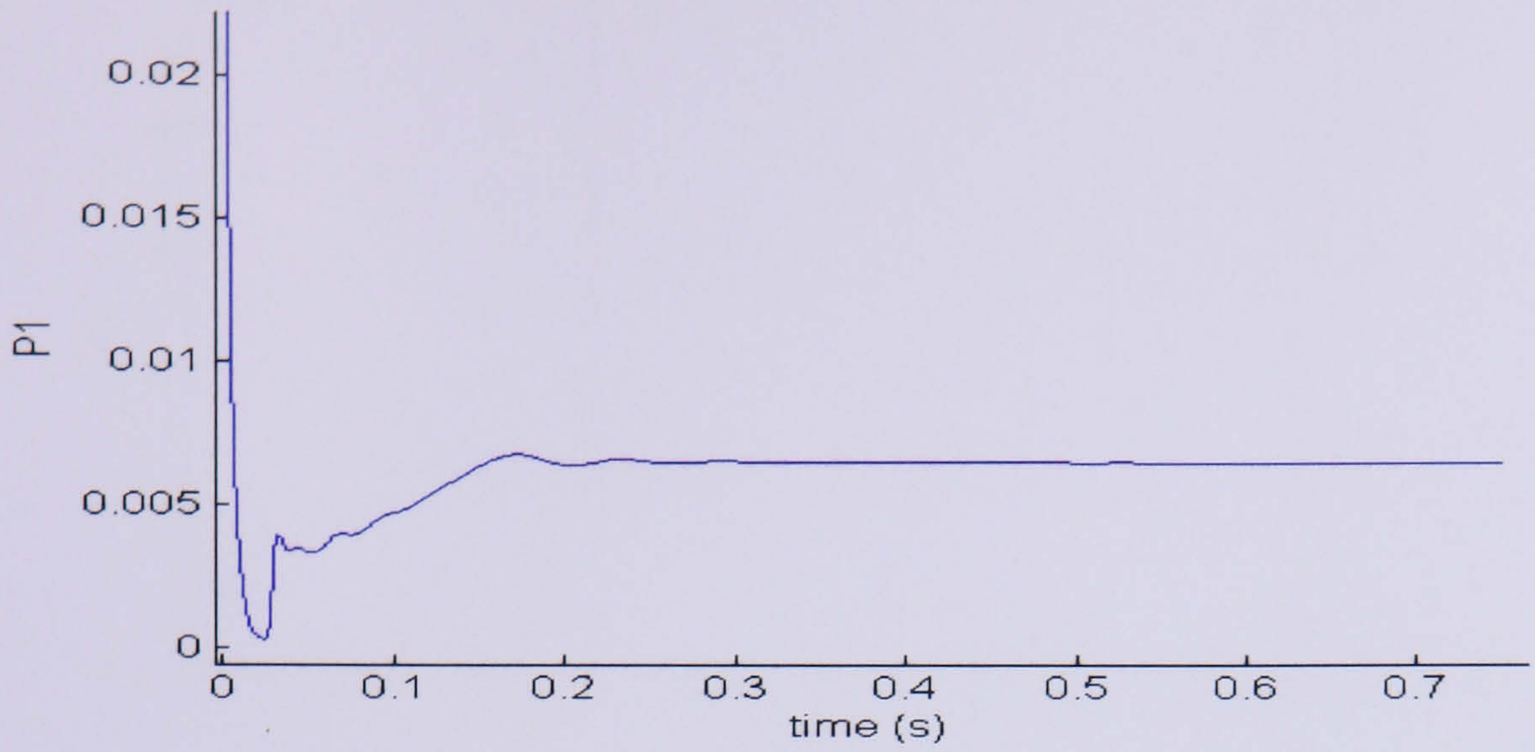
(a)



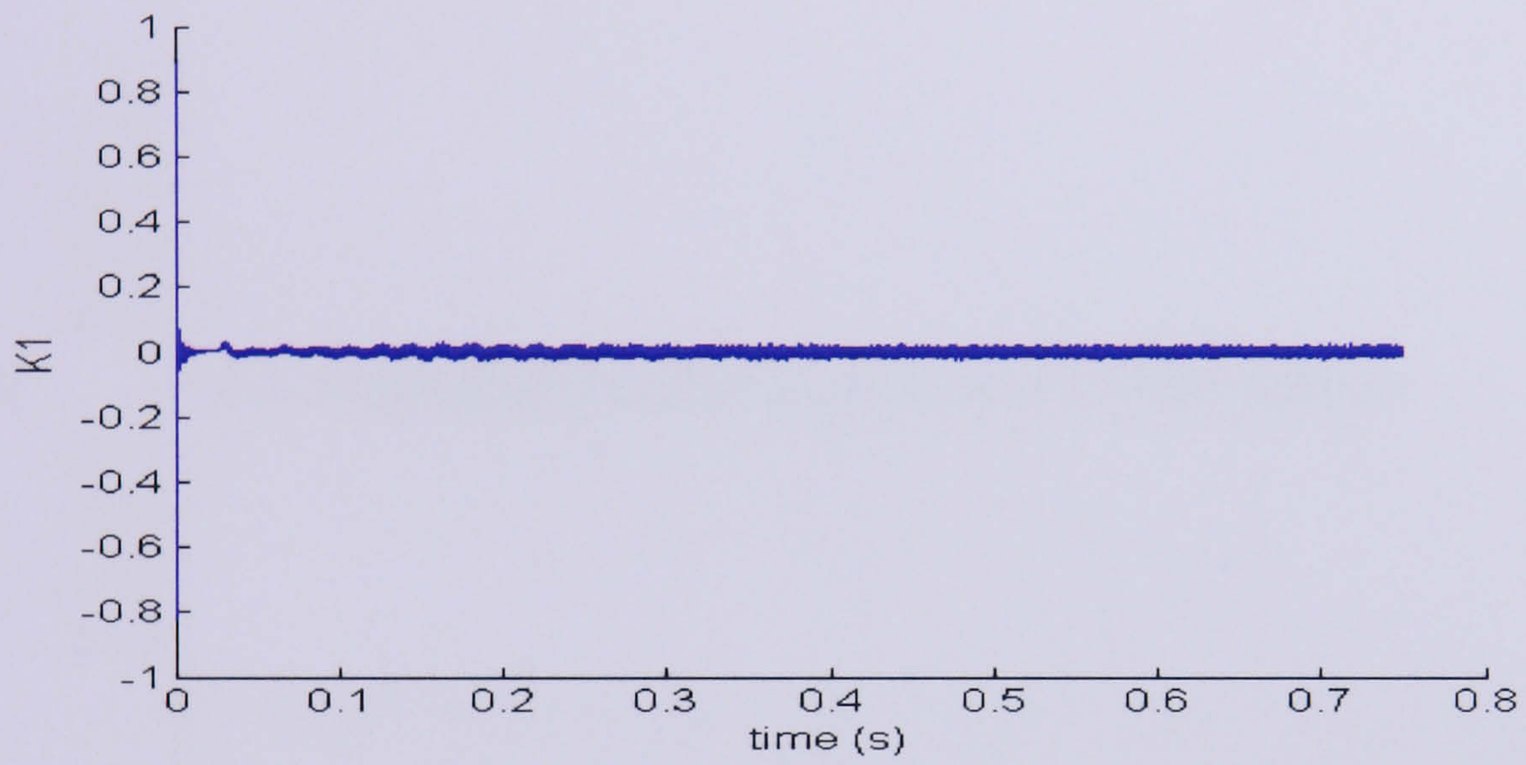
(b)

Figure 4.22: (a) P_1 and (b) K_1 for $\mathbf{Q} = \mathbf{I}_4$, $\mathbf{R} = 0.01\mathbf{I}_2$, $\mathbf{P}_0=10\mathbf{I}_4$, $\mathbf{X}_0=0$

Fourthly, when the measurement noise is increased in the noise model, the P_1 is rather high. This is shown in Figure 4.23. The KF puts more trust in the priori estimate (model) than the noisy measurement.



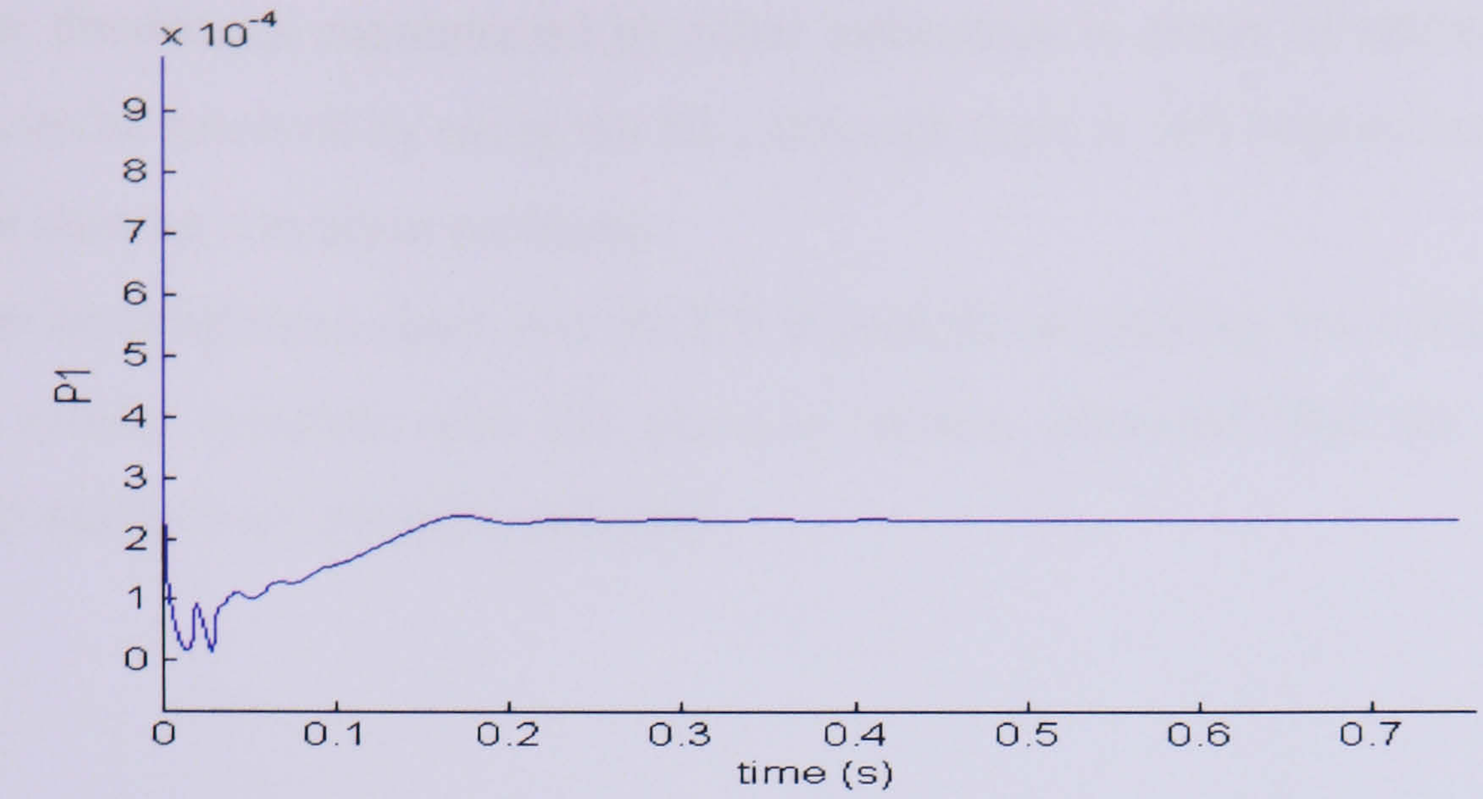
(a)



(b)

Figure 4.23: (a) P_1 and (b) K_1 for $\mathbf{Q} = 0.001\mathbf{I}_4$, $\mathbf{R} = \mathbf{I}_2$, $\mathbf{P}_0=10\mathbf{I}_4$, $\mathbf{X}_0=0$

And finally, when the KF is fed with different value of **Q** than the noise in the IM model, the Kalman gain is about the same as with the first case. Hence the variance used in both the model must not be the same with the KF. It also shows that the KF can be used to estimate the states without considering how large the noise on the IM model is. As long as the covariance matrices used for the estimator are properly selected, the KF can be used as an estimator. This is shown in Figure 4.24. The result is the same as in Figure 4.20.



(a)

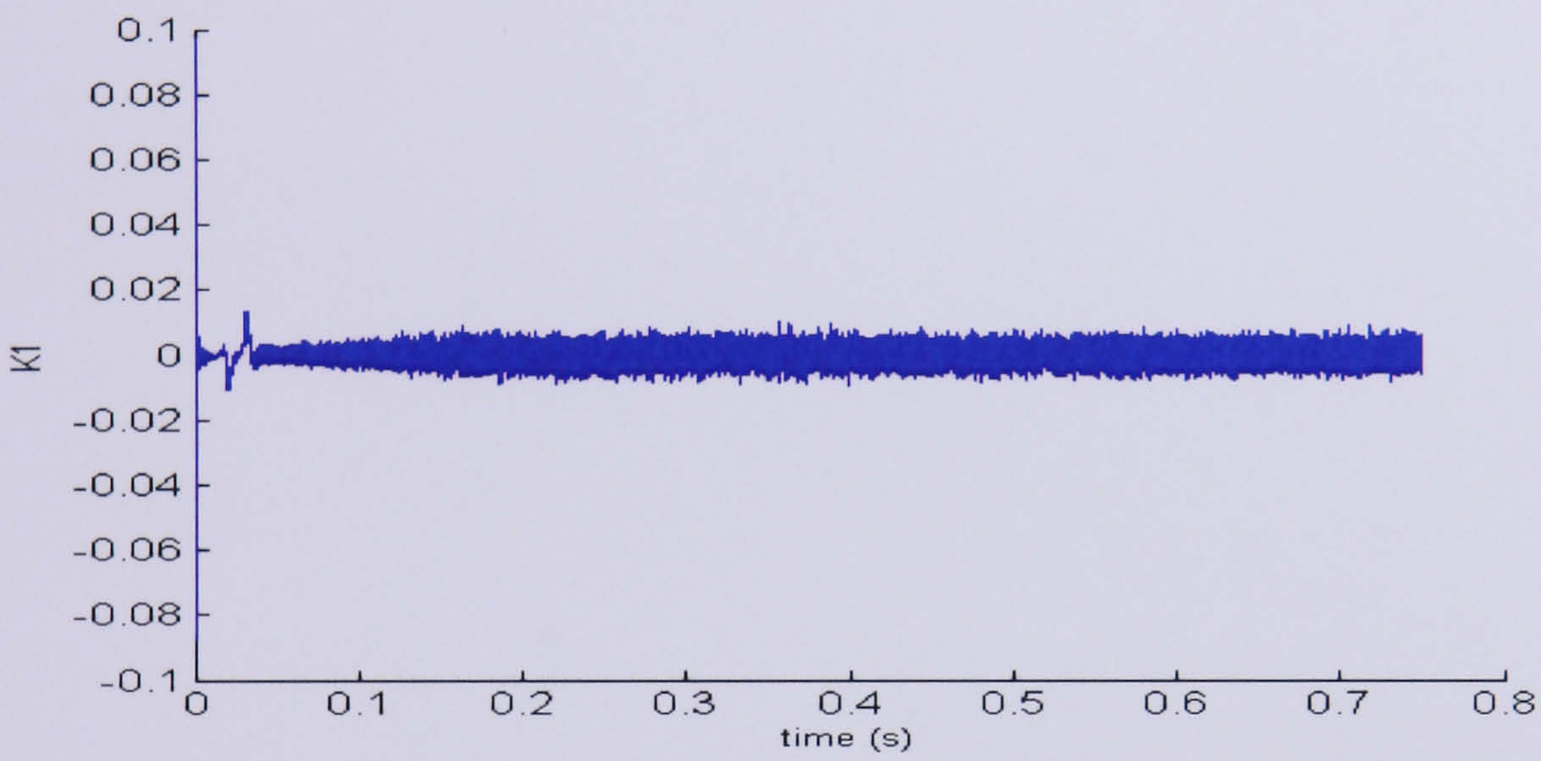


Figure 4.24: (a) P_1 and (b) K_1 for $\mathbf{Q} = 0.1\mathbf{I}_4$ of noise model, $\mathbf{Q} = 0.0001\mathbf{I}_4$ of KF, $\mathbf{R} = 0.01\mathbf{I}_2$, $\mathbf{P}_0$=$10\mathbf{I}_4$, $\mathbf{X}_0$=0

4.12 Conclusion

This chapter has presented a closed loop estimator for estimating the unmeasurable states of an IM. The derivation of KF algorithm has been presented in detail. The KF consists of two parts which are prediction and correction. It works by minimising the error covariance matrix. The features of KF have been described briefly and the Kalman gain characteristic has been observed in the simulation.

With no priori information in the initial estimated states and even in the presence of noisy measurement, it has been shown that the KF estimation performs quite well. Moreover, the KF is capable of estimating the rotor flux which is not accessible in the practical. However the estimating process requires significant computing time. The most difficult task faced when using KF as an estimator is to get correct noise covariance.

The drawbacks encountered by other estimators in terms of uncertainties in motor operation can be resolved by using the KF, although there is still improvement to be done on solving the start up condition problems.

The investigations show that the KF is capable of tracking the actual rotor speed and the speed model matches with the physical system provided that the elements of the covariance matrices are properly selected.

CHAPTER 5: SPEED ESTIMATION OF THE INDUCTION MOTOR

5.1 Introduction

The KF has been known as a stochastic observer. It can obtain high accuracy estimates of state variables in a noisy environment. The KF however applies to a linear system only. The application to a high performance sensorless drive requires knowledge of the state and the parameters to be estimated. The choice of using a separate observer for estimating the state and parameter is also given. Another way to find these values is by simultaneous estimation. To estimate the state and the parameters of the machine simultaneously, the selected parameters are treated as extra states in the state vector. The multiplication of the states in the equations then makes the system become nonlinear. For a nonlinear system, an EKF can be applied. It is one of the major approaches for speed estimation in IMs. The EKF is important since it has the capability to estimate the motor state and parameters with the advantage of being fairly simple to implement.

In this study, the application of the EKF to a nonlinear system which simultaneously estimates the rotor speed and the rotor flux of an IM is discussed. The linearization process and the associated EKF equations are given here. The application is verified with a direct supply connection to the IM and also with a constant V/f drive.

5.2 The Extended Kalman Filter

The linear estimator for a discrete system was derived in the previous chapter where the KF was used to estimate the rotor flux of the IM. Linear systems theory can be applied to nonlinear problems by a linearization method. For the application of the KF, the model must first be linearized about a nominal (reference) state trajectory. A flow chart of the EKF algorithm is shown in Figure 5.1 and the detail of the algorithm is given here. The discrete stochastic equation as in Eqn (4.1) and (4.2) is applied:

$$\dot{\mathbf{X}} = \mathbf{\Phi}_k \mathbf{x}_k + \mathbf{\Gamma}_k \mathbf{u}_k + \mathbf{G}_k \mathbf{W}_k$$

$$\mathbf{Y} = \mathbf{H}_k \mathbf{x}_k + \mathbf{V}_k \qquad (5.1)$$

The stochastic state space model based on Eqn (5.1) can be written in the form:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{G}_k \mathbf{w}_k$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \tag{5.2}$$

where f is a differentiable function of the state vector x and u and the function h is a differentiable function of the state vector x.

The process noise, $\mathbf{W}_k$ and the measurement noise, $\mathbf{V}_k$ are a white noise process with zero cross correlation as below:

$$E[\mathbf{W}_k] = 0$$

$$E[\mathbf{V}_k] = 0$$

$$E[\mathbf{W}_k \mathbf{W}_i^T] = \begin{cases} \mathbf{Q}_k & i = k \\ 0 & i \neq k \end{cases}$$

$$E[\mathbf{V}_k \mathbf{V}_i^T] = \begin{cases} \mathbf{R}_k & i = k \\ 0 & i \neq k \end{cases} \tag{5.3}$$

$\mathbf{W}_k$ and $\mathbf{V}_k$ are assumed to be separate additive terms and not included in the **f** and **h** function terms. Without those noises, the Eqn (5.2) becomes:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$$

$$\mathbf{y}_k = \mathbf{h}(x_k) \tag{5.4}$$

For linearization the nominal trajectory $\mathbf{x}^*_k$ is assumed to be a close solution to the actual trajectory $\mathbf{x}_k$. The EKF linearizes at each iteration around the estimated state and this is regarded as a major weakness of the EKF. The nominal trajectory $\mathbf{x}^*_k$ is obtained as a solution of the following equation without process noise and measurement noise:

$$\mathbf{x}^*_{k+1} = \mathbf{f}(\mathbf{x}^*_k, \mathbf{u}_k)$$

$$\mathbf{y}^*_k = \mathbf{h}(x^*_k) \tag{5.5}$$

The perturbations on the trajectory can be written as:

$$\delta\mathbf{x}_{k+1} = \mathbf{x}_{k+1} - \mathbf{x}^*_{k+1} \tag{5.6}$$

Assuming the perturbation is small enough, gives $\mathbf{x}_k = \mathbf{x}^*_k$, the Taylor series expansion, neglecting higher order terms, can be applied to Eqn (5.4):

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$$

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}^*_k, \mathbf{u}_k) + \frac{\partial \mathbf{f}(x,k)}{\partial x}\Big|_{\mathbf{x}=\mathbf{x}^*_k} \delta\mathbf{x}_k$$

$$\mathbf{x}_{k+1} = \mathbf{x}^*_{k+1} + \frac{\partial \mathbf{f}(x,k)}{\partial x}\Big|_{\mathbf{x}=\mathbf{x}^*_k} \delta\mathbf{x}_k \tag{5.7}$$

Recall Eqn (5.6) and replace with Eqn (5.7):

$$\delta\mathbf{x}_{k+1} = \frac{\partial \mathbf{f}(x,k)}{\partial x}\Big|_{\mathbf{x}=\mathbf{x}^*_k} \delta\mathbf{x}_k \tag{5.8}$$

where first order approximation coefficients are given by:

$$\mathbf{F}_k = \frac{\partial \mathbf{f}(.)}{\partial x}\Big|_{\mathbf{x}=\mathbf{x}^*_k} \tag{5.9}$$

The n x n matrix of first-order approximation coefficient $\mathbf{F}_k$ is known as the Jacobian matrix of partial derivatives and can be defined as:

$$\mathbf{F}_k = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} & \cdots & \frac{\partial f_3}{\partial x_n} \\ \vdots & \vdots & \vdots & \ddots & \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \frac{\partial f_n}{\partial x_3} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}\Bigg| \mathbf{x} = \mathbf{x}^*_k \tag{5.10}$$

Therefore, the linearized state equation about nominal values is:

$$\delta\mathbf{x}_{k+1} = \mathbf{F}_k \delta\mathbf{x}_k \tag{5.11}$$

The linearized measurement equation can be represented by the Taylor series for the Eqn (5.4):

$$\mathbf{h}_k = \mathbf{h}(\mathbf{x}^*_k) + \frac{\partial \mathbf{h}(.)}{\partial x}\Big|_{\mathbf{x}=\mathbf{x}^*_k} \delta\mathbf{x}_k \tag{5.12}$$

$$\delta\mathbf{y}_k = \frac{\partial \mathbf{h}(x,k)}{\partial x}\Big|_{\mathbf{x}=\mathbf{x}^*_k} \delta\mathbf{x}_k$$

where $\mathbf{H}_k = \frac{\partial \mathbf{h}(.)}{\partial x}\Big|_{\mathbf{x}=\mathbf{x}^*_k}$;

Therefore:

$$\delta\mathbf{y}_k = \mathbf{H}_k \delta\mathbf{x}_k \tag{5.13}$$

The m x n matrix of first-order approximation coefficient $\mathbf{H}_k$ is:

$$\mathbf{H}_k = \left[\begin{array}{ccccc} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} & \frac{\partial h_1}{\partial x_3} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_2} & \frac{\partial h_2}{\partial x_3} & \cdots & \frac{\partial h_2}{\partial x_n} \\ \frac{\partial h_3}{\partial x_1} & \frac{\partial h_3}{\partial x_2} & \frac{\partial h_3}{\partial x_3} & \cdots & \frac{\partial h_3}{\partial x_n} \\ \vdots & \vdots & \vdots & \ddots & \\ \frac{\partial h_m}{\partial x_1} & \frac{\partial h_m}{\partial x_2} & \frac{\partial h_m}{\partial x_3} & \cdots & \frac{\partial h_m}{\partial x_n} \end{array} \right] \Bigg|_{\mathbf{x} = \mathbf{x}^*_k} \tag{5.14}$$

When the actual trajectory $\mathbf{x}_k$ is sufficiently close to the nominal trajectory, then the higher order terms involve in the Taylor series approximation can be ignored. Therefore Eqn (5.11) and (5.13) can be used to transform the problem to a linear problem. Eqn (5.11) and (5.13) in the presence of noise is given as:

$$\delta\mathbf{x}_{k+1} = \mathbf{F}_k \delta\mathbf{x}_k + \mathbf{w}_k \tag{5.15}$$

$$\delta\mathbf{y}_k = \mathbf{H}_k \delta\mathbf{x}_k + \mathbf{v}_k \tag{5.16}$$

Although it has been shown to be a simple and effective approach, the major drawback of the EKF is that the computation of the unpredictable trajectory increases the computational burden and cost of the filter due to the real-time computation requirement. So, the standard recursive form of the EKF algorithm is expressed by the following equations:

a) The predicted state estimate:

$$\hat{\mathbf{X}}_{k+1/k} = \mathbf{f}(\hat{\mathbf{X}}_{k/k}, \mathbf{U}_k) + \mathbf{W}_k \tag{5.17}$$

where $\mathbf{f}(\hat{\mathbf{X}}_{k/k}, \mathbf{U}_k) = \mathbf{\Phi}_k \mathbf{X}_{k/k} + \mathbf{\Gamma}_k \mathbf{U}_k$ (5.18)

b) The a priori covariance matrix:

$$\overline{\mathbf{P}}_{k+1/k} = \mathbf{F}_k \overline{\mathbf{P}}_{k/k} \mathbf{F}_k{}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T \tag{5.19}$$

c) The Kalman gain:

$$\mathbf{K}_{k+1} = \mathbf{H}_{k+1}{}^T \overline{\mathbf{P}}_{k+1/k} (\mathbf{H}_{k+1} \overline{\mathbf{P}}_{k+1/k} \mathbf{H}_{k+1}{}^T + \mathbf{R})^{-1} \tag{5.20}$$

d) The posteriori estimator:

$$\hat{\mathbf{X}}_{k+1} = \hat{\mathbf{X}}_{k+1/k+1} = (1 - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\hat{\mathbf{X}}_{k+1/k} + \mathbf{K}_{k+1}\mathbf{Y}_{k+1} \tag{5.21}$$

e) The a posteriori covariance matrix:

$$\overline{\mathbf{P}}_{k+1/k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\overline{\mathbf{P}}_{k+1/k} \tag{5.22}$$

It is important to state that because the matrices $\mathbf{F}_k$ and $\mathbf{H}_k$ depend on previous state estimates, off-line computation is not applicable for the filter gain $\mathbf{K}_k$ and the matrices $\overline{\mathbf{P}}_{k/k}$ and $\overline{\mathbf{P}}_{k+1/k}$. In addition, since the EKF is implemented based on a set of approximations, it is therefore not an optimal filter. Thus, the covariance matrices $\overline{\mathbf{P}}_{k/k}$ and $\overline{\mathbf{P}}_{k+1/k}$ only represent the approximation and do not represent the true covariance of the state estimates.

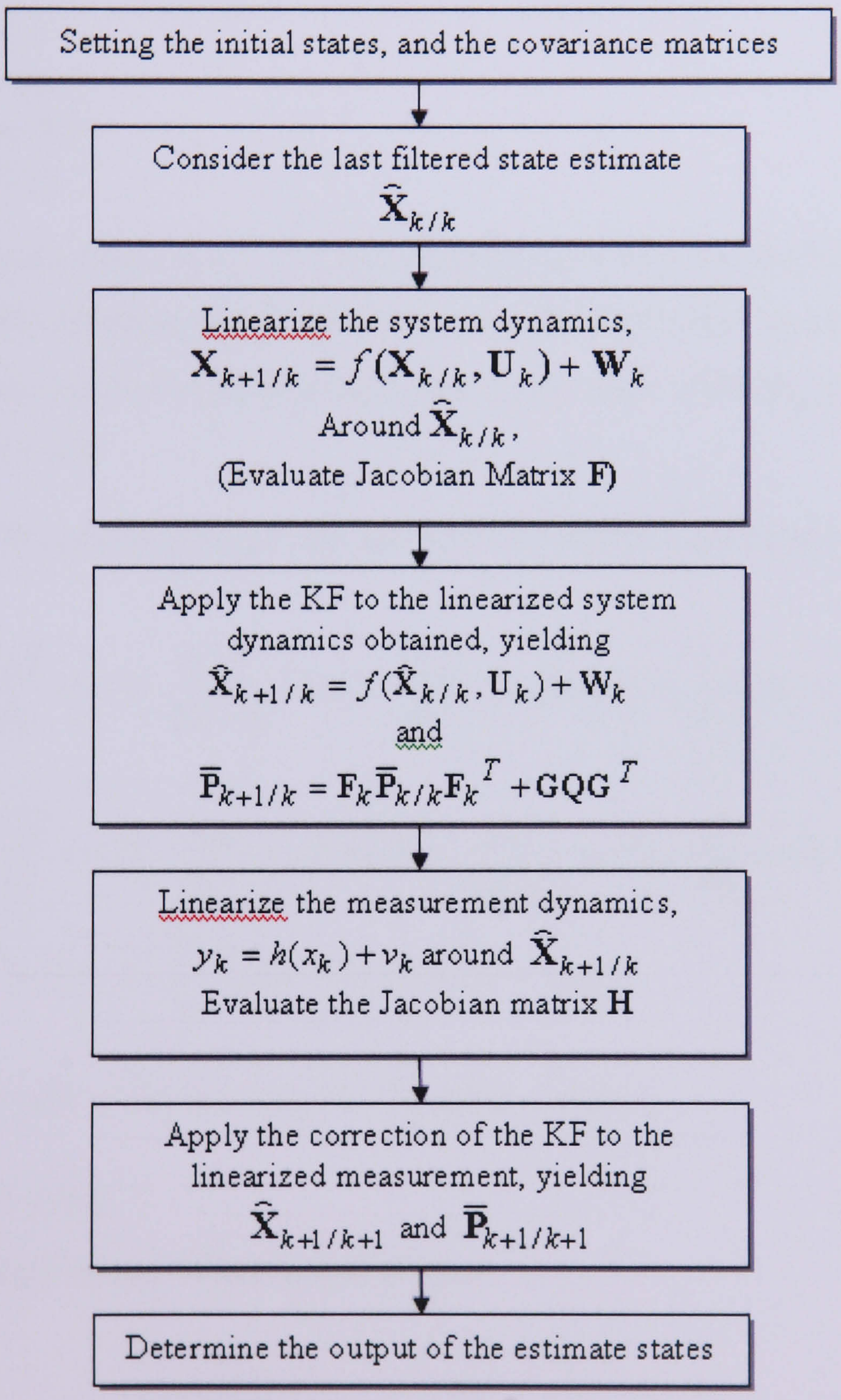


Figure 5.1: A flow chart of the EKF

5.3 Application to Speed Estimation

A linear IM model is derived in Eqn (3.13). Converting to state variable form and transforming to a discrete time model has yielded Eqn (4.16). The estimation of the rotor speed requires the speed to be treated as an extra state by forming an augmented state vector. The extended state vector with the rotor speed as an extra state variable is as follows:

$$\mathbf{x}_k = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} i_{sD}(k) \\ i_{sQ(k)} \\ \psi_{rd(k)} \\ \psi_{rq}(k) \\ \omega_r(k) \end{bmatrix} \tag{5.23}$$

It is assumed that the rotor speed varies as a randomly disturbed variable, therefore a component of white Gaussian noise $n(k)$ is added to give a revised or updated speed variable, giving an equation of the following form for the speed including noise:

$$\omega_{r(k+1)} = \omega_r(k) + n(k) \tag{5.24}$$

Therefore the fifth-order augmented state space model with the state vectors equations are as follows:

$$x_1(k+1) = 1 - \frac{K_2 T}{K_1} x_1(k) + \frac{L_m T}{K_1 L_r T_r} x_3(k) + \frac{L_m T}{K_1 L_r} x_4(k) x_5(k) + \frac{T}{K_1} u_1(k) + w_1(k)$$

$$x_2(k+1) = 1 - \frac{K_2 T}{K_1} x_2(k) - \frac{L_m T}{K_1 L_r} x_3(k) x_5(k) + \frac{L_m T}{K_1 L_r T_r} x_4(k) + \frac{T}{K_1} u_2(k) + w_2(k)$$

$$x_3(k+1) = \frac{L_m T}{T_r} x_1(k) + 1 - \frac{T}{T_r} x_3(k) - T x_4(k) x_5(k) + w_3(k)$$

$$x_4(k+1) = \frac{L_m T}{T_r} x_2(k) + T x_3(k) x_5(k) + 1 - \frac{T}{T_r} x_4(k) + w_4(k)$$

$$x_5(k+1) = x_5(k) + n(k) \tag{5.25}$$

The above equations can be written in matrix form:

$$\mathbf{\Phi}_k = \begin{bmatrix} 1-\frac{K_2T}{K_1} & 0 & \frac{L_mT}{K_1L_rT_r} & \frac{\omega_r L_mT}{K_1L_r} & 0 \\ 0 & 1-\frac{K_2T}{K_1} & -\frac{\omega_r L_mT}{K_1L_r} & \frac{L_mT}{K_1L_rT_r} & 0 \\ \frac{L_mT}{T_r} & 0 & 1-\frac{T}{T_r} & -\omega_r T & 0 \\ 0 & \frac{L_mT}{T_r} & \omega_r T & 1-\frac{T}{T_r} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{\Gamma}_k = \begin{bmatrix} \frac{T}{K_1} & 0 \\ 0 & \frac{T}{K_1} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{5.26}$$

As mentioned the state space model is no longer linear. Therefore the linearization procedure for the KF is applied. The Jacobian matrix of partial derivatives of $\mathbf{f}(\mathbf{x})$ is a 5 x 5 time varying matrix.

$$\mathbf{F}_k = \begin{bmatrix} 1-\frac{K_2T}{K_1} & 0 & \frac{L_mT}{K_1L_rT_r} & \frac{\omega_r L_mT}{K_1L_r} & \frac{L_mT}{K_1L_r}\psi_{rq} \\ 0 & 1-\frac{K_2T}{K_1} & -\frac{\omega_r L_mT}{K_1L_r} & \frac{L_mT}{K_1L_rT_r} & -\frac{L_mT}{K_1L_r}\psi_{rd} \\ \frac{L_mT}{T_r} & 0 & 1-\frac{T}{T_r} & -\omega_r T & -T\psi_{rq} \\ 0 & \frac{L_mT}{T_r} & \omega_r T & 1-\frac{T}{T_r} & T\psi_{rd} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.27}$$

The elements of $\mathbf{F}_k$ are calculated at each time step by substituting the most recent state estimate rotor fluxes and speed measurement into $x_3(k), x_4(k), x_5(k)$. The Jacobian matrix of partial derivatives of $\mathbf{H}$ which maps the state vector to the measurement vector with respect to x is however constant.

$$\mathbf{N}_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{5.28}$$

The disturbance noise weight matrix is:

$$\mathbf{G} = \begin{bmatrix} G_{11} & 0 & 0 & 0 & 0 \\ 0 & G_{22} & 0 & 0 & 0 \\ 0 & 0 & G_{33} & 0 & 0 \\ 0 & 0 & 0 & G_{44} & 0 \\ 0 & 0 & 0 & 0 & G_{55} \end{bmatrix} \tag{5.29}$$

The process noise covariance matrix is:

$$\mathbf{Q} = \begin{bmatrix} Q_{11} & 0 & 0 & 0 & 0 \\ 0 & Q_{22} & 0 & 0 & 0 \\ 0 & 0 & Q_{33} & 0 & 0 \\ 0 & 0 & 0 & Q_{44} & 0 \\ 0 & 0 & 0 & 0 & Q_{55} \end{bmatrix} \tag{5.30}$$

and the observation noise covariance matrix is given by

$$\mathbf{R} = \begin{bmatrix} R_{11} & 0 \\ 0 & R_{22} \end{bmatrix} \tag{5.31}$$

The initial error covariance $\mathbf{P}(0)$ is known where:

$$\mathbf{P} = \begin{bmatrix} P_{11} & 0 & 0 & 0 & 0 \\ 0 & P_{22} & 0 & 0 & 0 \\ 0 & 0 & P_{33} & 0 & 0 \\ 0 & 0 & 0 & P_{44} & 0 \\ 0 & 0 & 0 & 0 & P_{55} \end{bmatrix} \tag{5.32}$$

The diagonal elements for the above matrices correspond to

$i_{sD}, i_{sQ}, \psi_{rd}, \psi_{rq}, \omega_r$ respectively

Figure 5.2 shows the block diagram of the EKF. The EKF block uses the same motor parameters as in IM and is driven by deterministic current and voltage input. Without knowledge of the noise in the real motor, the EKF tries to guess the state matrices. The only information the estimator gets from the IM block is the stator currents which are usually measurable in practice, although will of course have noise. Based on the dynamic equations, the stator currents are calculated and these values are then used by the estimator. The IM model used in the estimator is in the discrete time form. The detail of the conversion from continuous to discrete time has been discussed in Chapter 4. The discrete model IM closely matches with the actual continuous time IM model provided that the discrete sampling time used is small.

In the subsystem block f, the $f(\ )$ function is performed and the Jacobian matrix is calculated at each time step by substituting the most recent state estimate and speed measurement into $x_3(k), x_4(k), x_5(k)$. Using the available covariance matrices and the initial state, the Kalman gain is calculated and the estimation error of the state is then corrected. This is done in the subsystem K(Y-Ye).

5.4 Simulation Using Direct Power Supply

The IM block together with the EKF estimator is shown in Figure 5.3. In this simulation block a direct supply from a 3-phase voltage is used. The transformation of 3-phase to dq axes is done in the 3PH to DQ transformation block. The IM block is based on the stator and rotor flux voltage equation given in Chapter 3. One of the outputs, the electromagnetic torque is used as an input to the mechanical block. The resultant speed from the mechanical block is used as the reference speed in the simulation. Using only information from the IM which is the stator current and using the same power supply output, the EKF estimator is trying to estimate the actual state of the IM model. The simulation study on the speed estimation has been done using Matlab/Simulink software. The simulation is run on an AMD Athlon XP at 2.09GHz, with 512 MB of RAM and Matlab 6.5. To test the performance of the estimation method, simulations were performed on an IM with parameters listed in Chapter 3.

The value of the covariance matrix elements has a considerable effect on the performance of the EKF estimation. In most cases, the covariance matrix of these noises is not known. In the simulation, the covariance matrices are tuned by a Trial and Error method to achieve the desired transient and steady state behaviour. The initial state vector for the estimator is

$$x(0) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$

The variances or mean squared errors in the EKF algorithm are represented by the diagonal state covariance matrix **P**(0). Varying **P**(0) produces rapid initial convergence [52] but yields different amplitude of the transient while both the transient duration and steady state conditions will be unaffected [86]. The initial value of the covariance matrix is not really a crucial factor in this simulation; the value chosen below is based on the satisfactory performance of the estimator.

$$P(0) = \begin{bmatrix} 20 & 0 & 0 & 0 & 0 \\ 0 & 20 & 0 & 0 & 0 \\ 0 & 0 & 20 & 0 & 0 \\ 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 0 & 20 \end{bmatrix}$$

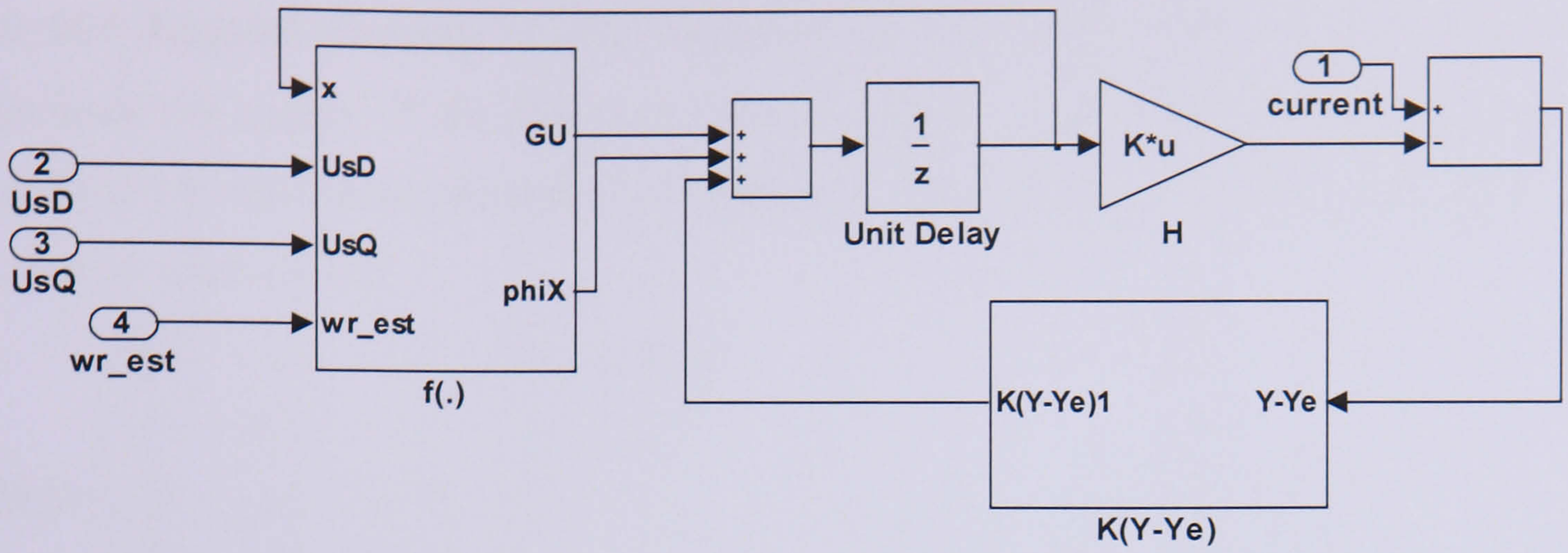


Figure 5.2: The structure of the EKF algorithm used for speed estimation in the IM

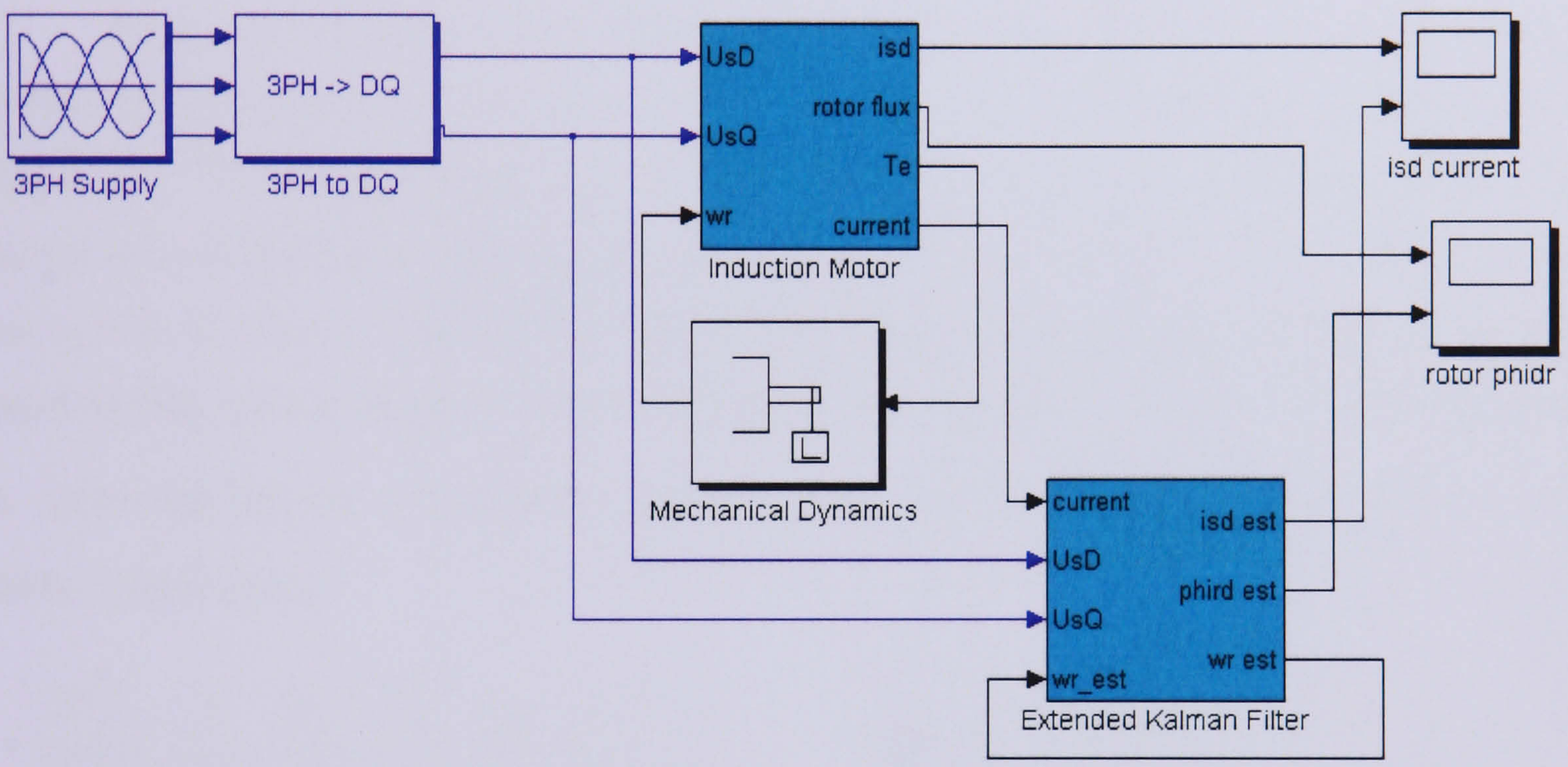


Figure 5.3: Simulation block of IM with KF estimator

Initially the motor model and the measurement were considered to be perfect. Later, assuming the model is under the influence of the noisy source, four independent, zero-mean, white noise signals were added to the system. $Q_{11}, Q_{22}, Q_{33}, Q_{44}$ correspond to the state uncertainty, d and q axes of stator current and rotor flux. The behaviour of the additional rotor speed as a state variable in the EKF corresponds to the fifth element of noise covariance which is Q_{55}. To simulate the measurement error, since only stator currents can be easily measured in practice, two random signals R with the same characteristics as the system's noise were added. The measurement noise covariance matrix is

$$\mathbf{R}(k) = \begin{bmatrix} 0.02 & 0 \\ 0 & 0.02 \end{bmatrix}$$

The noise weight matrix **G** also influences the performance of the estimator. The first four diagonal elements G_{11}-G_{44} represent the state vector while the fifth element G_{55} represents the speed. If the first four element different with the fifth element G_{55}, larger process noise covariance element Q_{55} is required. Initially, the value of the element is set to a constant value of 0.01.

$$\mathbf{G}(k) = \begin{bmatrix} 0.01 & 0 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0.01 \end{bmatrix}$$

To see the actual IM model characteristics used in the simulation, the model was run without the presence of the disturbance, with **Q** and **R** set to zero. The results are shown in Figure 5.4. The voltage supply used is 326.6 Vp-p, 50Hz as shown in the first trace. The second trace is the d-axis stator current which settled to 8.42A at about 0.2s. The rotor flux can be seen in trace 3. The electromagnetic torque with the zero load torque gives the rotor speed settling time at about 0.2s with the steady state value of 157 rad/s. In Figure 5.4 and the following figures, the reference speed and the estimated speed are indicated by ω_r^* and ω_r respectively.

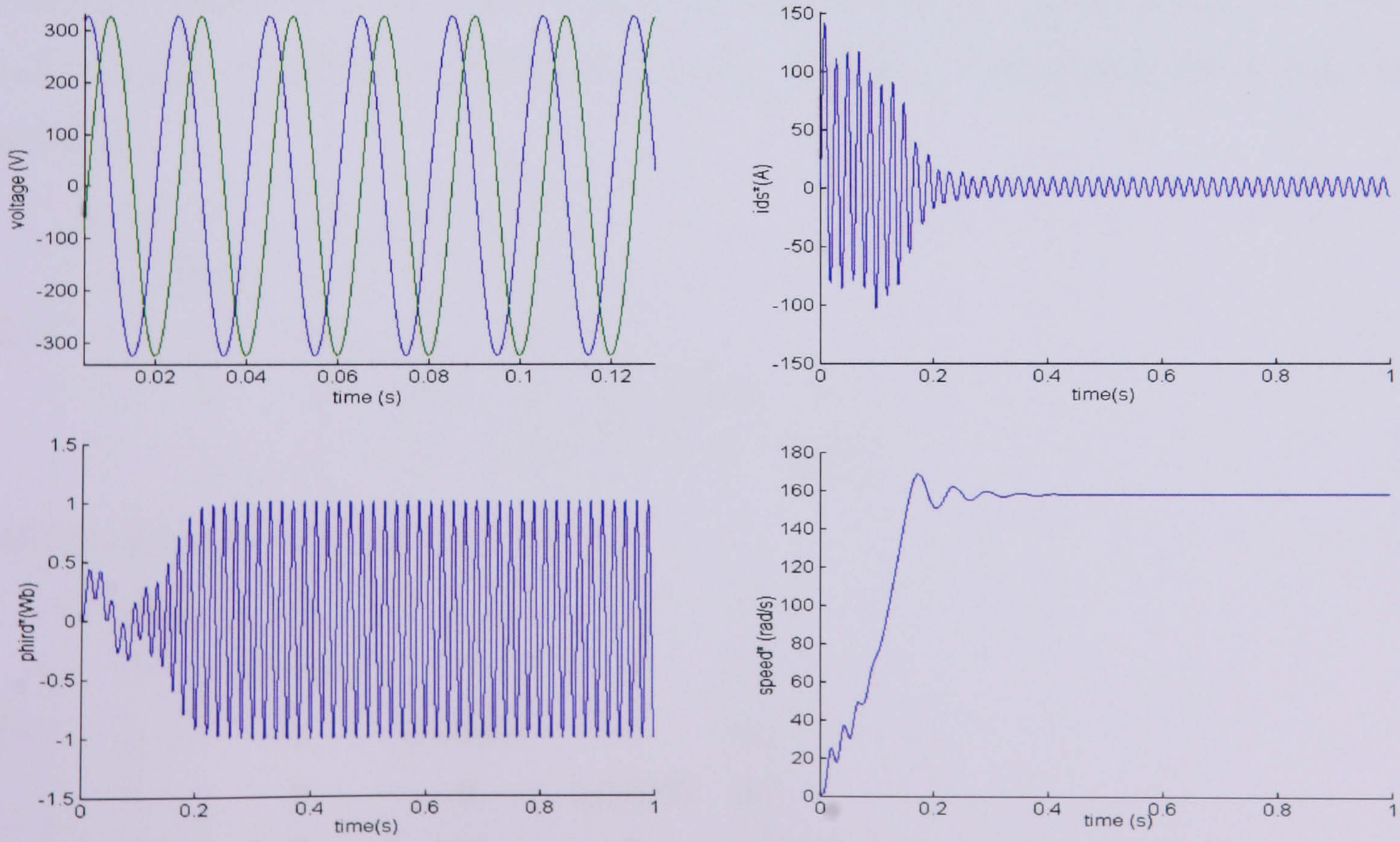


Figure 5.4: The voltage, current, flux and speed reference applied in simulation

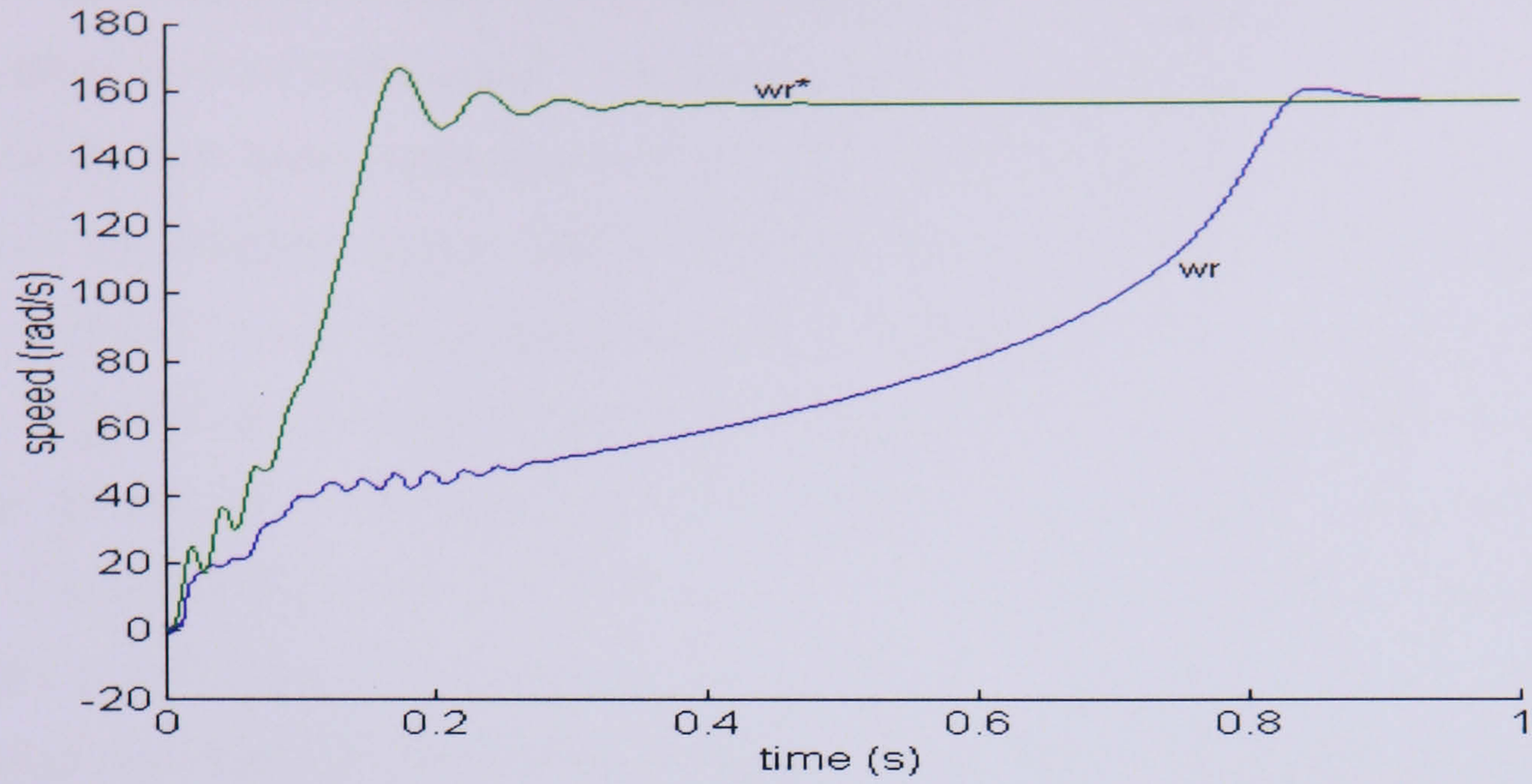
Setting the process noise covariance Q to zero yields an open loop estimator and gives poor performance. This is shown in Figure 5.5(a). The covariance matrix Q plays an important role in the behaviour of the algorithm. Q_{55} in particular will control directly the estimated speed in the EKF algorithm. To see the effect of Q_{55}, the process noise covariance matrix needs to be evaluated differently for Q_{11}-Q_{44} and Q_{55}. The values in matrix Q are assumed to be $\mathbf{Q} = Diag[\alpha \quad \alpha \quad \alpha \quad \alpha \quad \beta]$. If parameters α and β are equal this results in poor speed estimation performance.

Figure 5.5 (b) shows the typical EKF performance obtained by a Trial and Error method for Q_{55}. By selecting larger values of β the performance of algorithm is improved. When Q_{55} is unity, the estimated speed does achieve a steady state value at about 0.4s which is close to the actual speed. However to reduce the settling time some overshoot may be allowed, perhaps with Q_{55}=10 or bigger. Using Q_{55}=10 or higher the estimated speed converges quickly to the motor speed. The bigger Q_{55} the more jittery the estimated speed. Based on this evaluation, good estimation performance can be obtained by setting the value of α=0.01 and β=100. Using these values, the result of the rotor speed can be seen clearly in Figure 5.6. The choice of covariance stated above seems satisfactory enough in estimating the speed however the error covariance P_{55} in Figure 5.7 shows that using a combination of those covariances result in a 9.65 $(rad/s)^2$ error. Using the same initial state, $\mathbf{X}(0)$ and error covariance matrix $\mathbf{P}(0)$ but with a new combination of covariance matrix of $\mathbf{R}$=0.01I_2, $\mathbf{G}$=0.01I_5 and α=0.00001 and β=1 gives better accuracy. The covariance matrix can be written as:

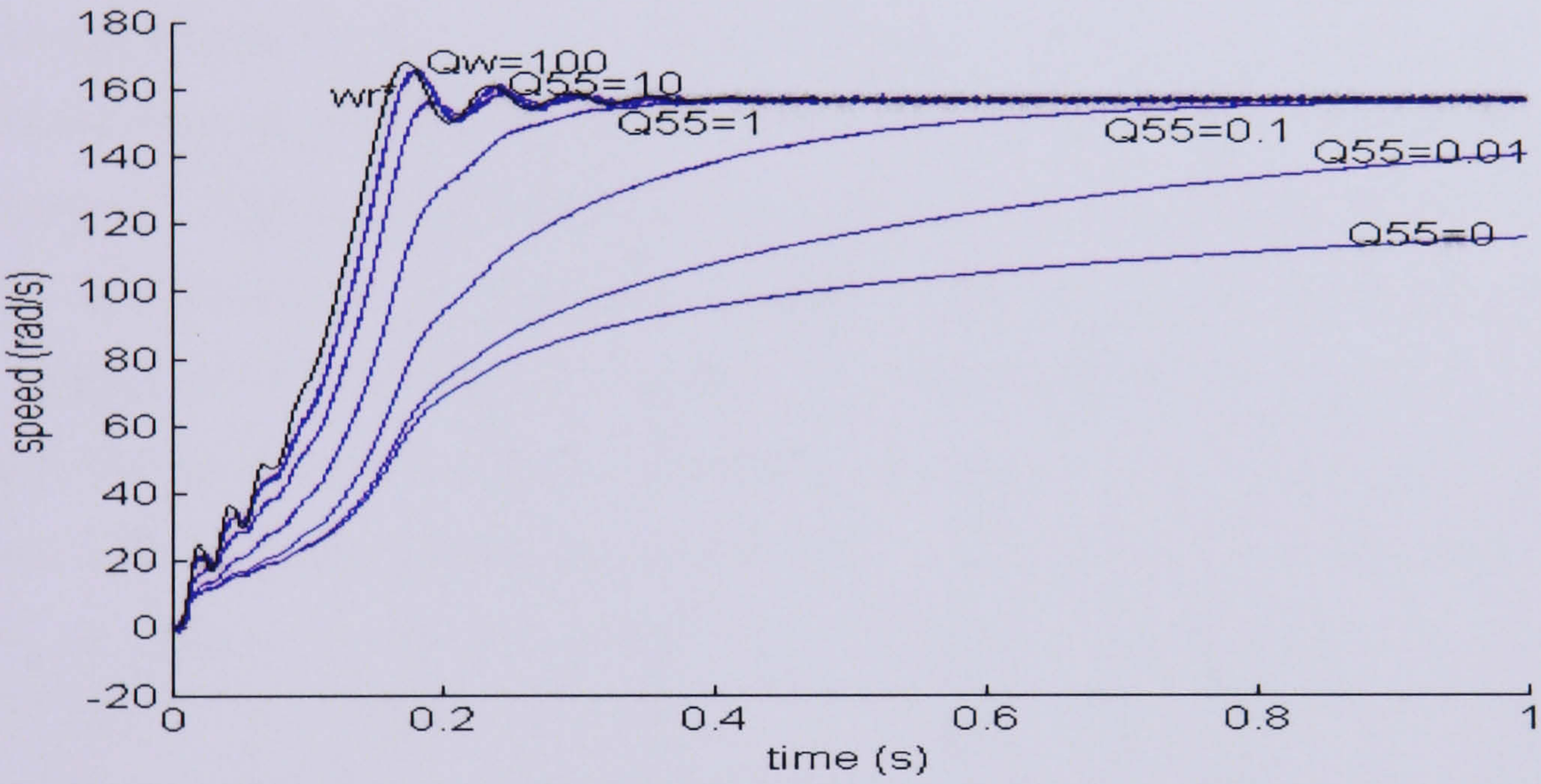
$$\mathbf{G} = \begin{bmatrix} 0.01 & 0 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0.01 \end{bmatrix} \quad \mathbf{R} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

The process noise covariance matrix is:

$$\mathbf{Q} = \begin{bmatrix} 0.00001 & 0 & 0 & 0 & 0 \\ 0 & 0.00001 & 0 & 0 & 0 \\ 0 & 0 & 0.00001 & 0 & 0 \\ 0 & 0 & 0 & 0.00001 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$



(a)



(b)

Figure 5.5: (a) Results of setting **Q**=zero (b) Performance of EKF with various covariance $\mathbf{Q}_{55}$

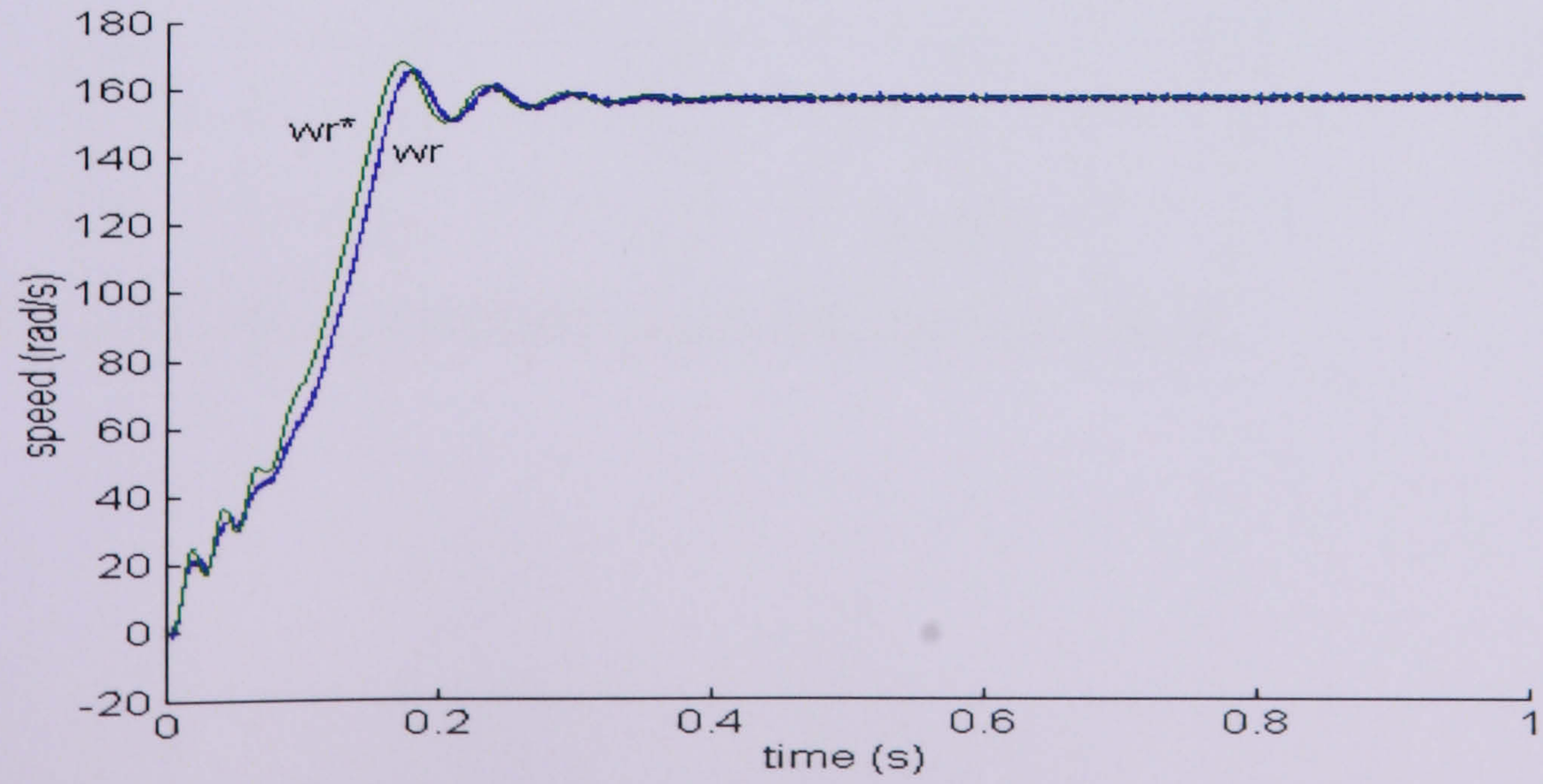


Figure 5.6: Estimated speed with covariance $\mathbf{Q}_{55}$=100

The results are closely similar showing that the estimator is working well with the small speed error of 0.04 $(rad/s)^2$. The mean squared error obtained is summarised in Table 5.1 with the best mean squared error E_n of 4.40. The result as shown in Figure 5.8 clearly indicates an excellent match, during transient response and steady state operation. To achieve the best result the covariance matrices G_{55} and Q_{55}, which correspond to the noise of the estimated speed, need to have larger values than the other covariance matrices. It is shown that the estimated speed experiences greater variations than the other state variables [59]. The speed error is shown in Figure 5.9. The d-axis stator currents and rotor fluxes between estimated and reference are shown in Figure 5.10 and 5.11 Figure 5.12 shows the estimated values are very close to the reference value for both current and rotor flux.

After finding the right covariance, the performance of the estimator is verified using higher uncertainties in the model. The process noise of the IM model was increased to a 10% noise variation, which is $Q= 4x10^6$, while maintaining the same covariance matrices in the estimator. The results in Figure 5.13 and Figure 5.14 shows that, without knowing the increase of these values, the estimator is capable of estimating the speed and produces the same small error of $0.04(rad/s)^2$. Figure 5.14, when compared to Figure 5.9, shows that although the same small variance is obtained, the response has slight small perturbations because with the uncertainties have increased. The bigger the uncertainties, the more "jittery" the results. In this case another set of covariance matrices needs to be selected.

G_{1-4}	G_5	R_{12}	Q_{1-4}	Q_5	E_n
0.01	0.01	0.02	0.00	0.00	7960
0.01	0.01	0.02	0.01	0.01	3670
0.01	0.01	0.02	0.01	1.00	619
0.01	0.01	0.02	0.01	10.0	154
0.01	0.01	0.02	0.01	100	28
0.01	0.01	0.01	0.00001	1	4.4

Table 5.1: Performance of the EKF using a Trial and Error method

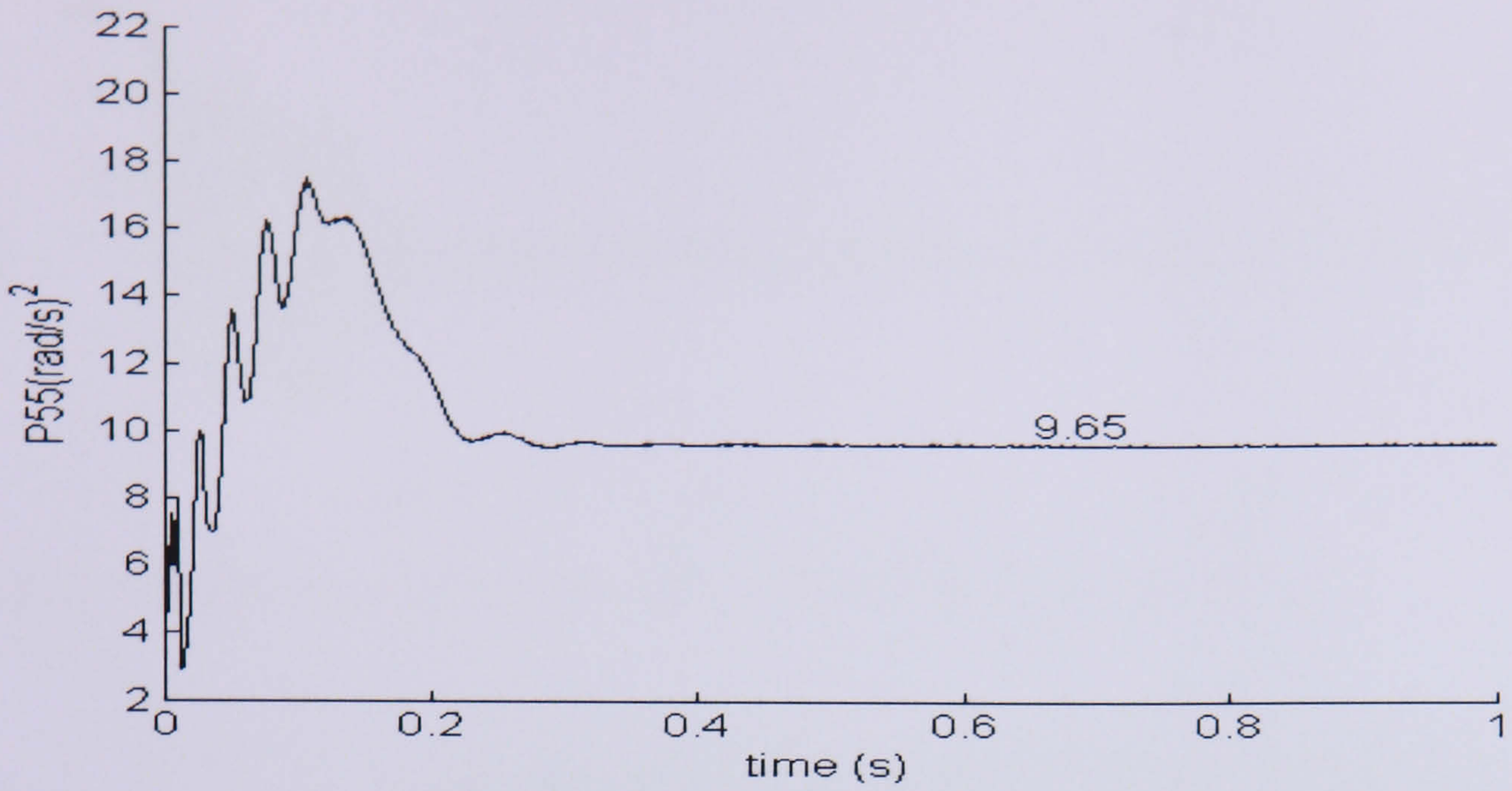


Figure 5.7: The error covariance of speed estimation

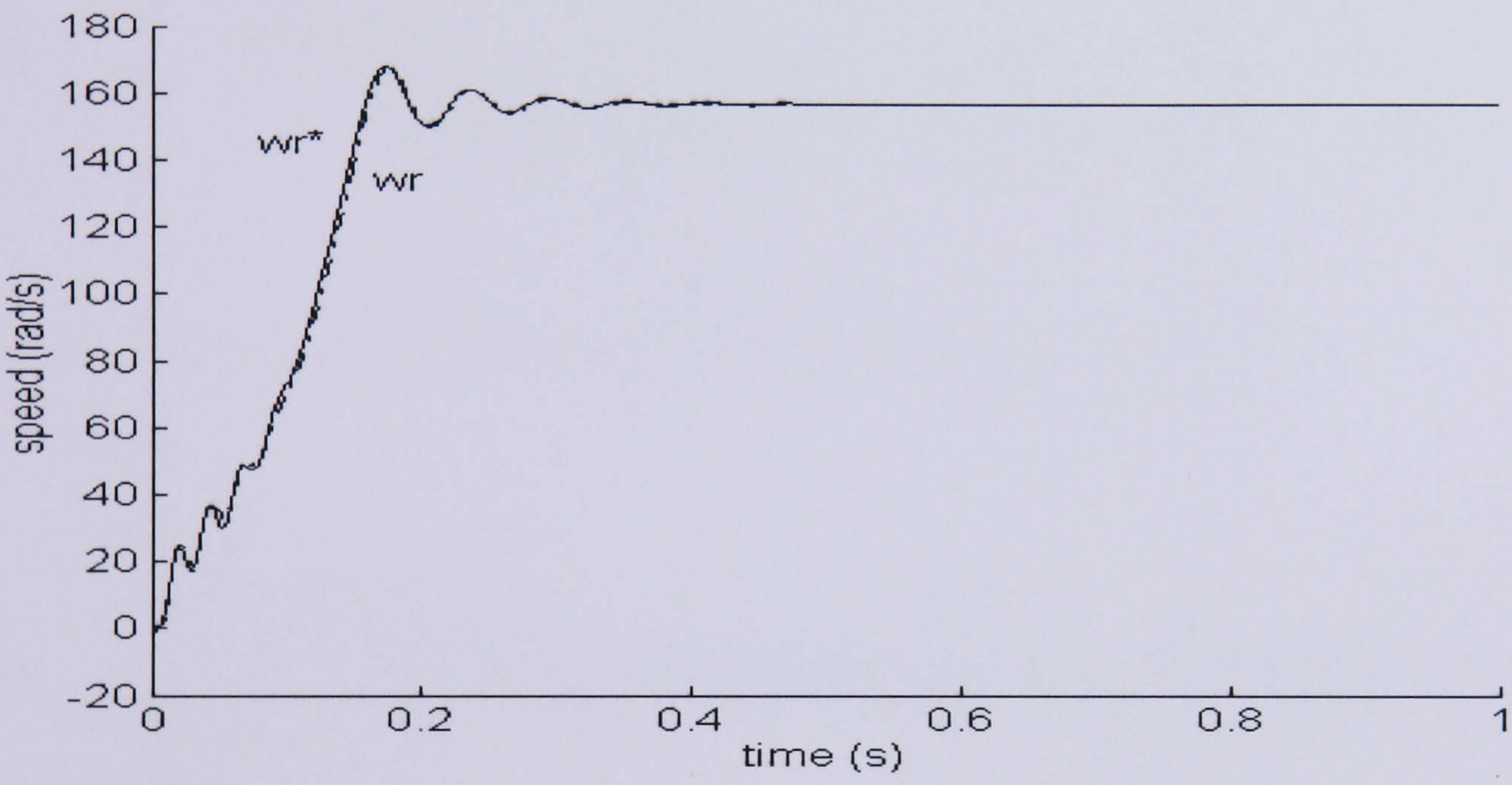


Figure 5.8: Speed during transient and steady state

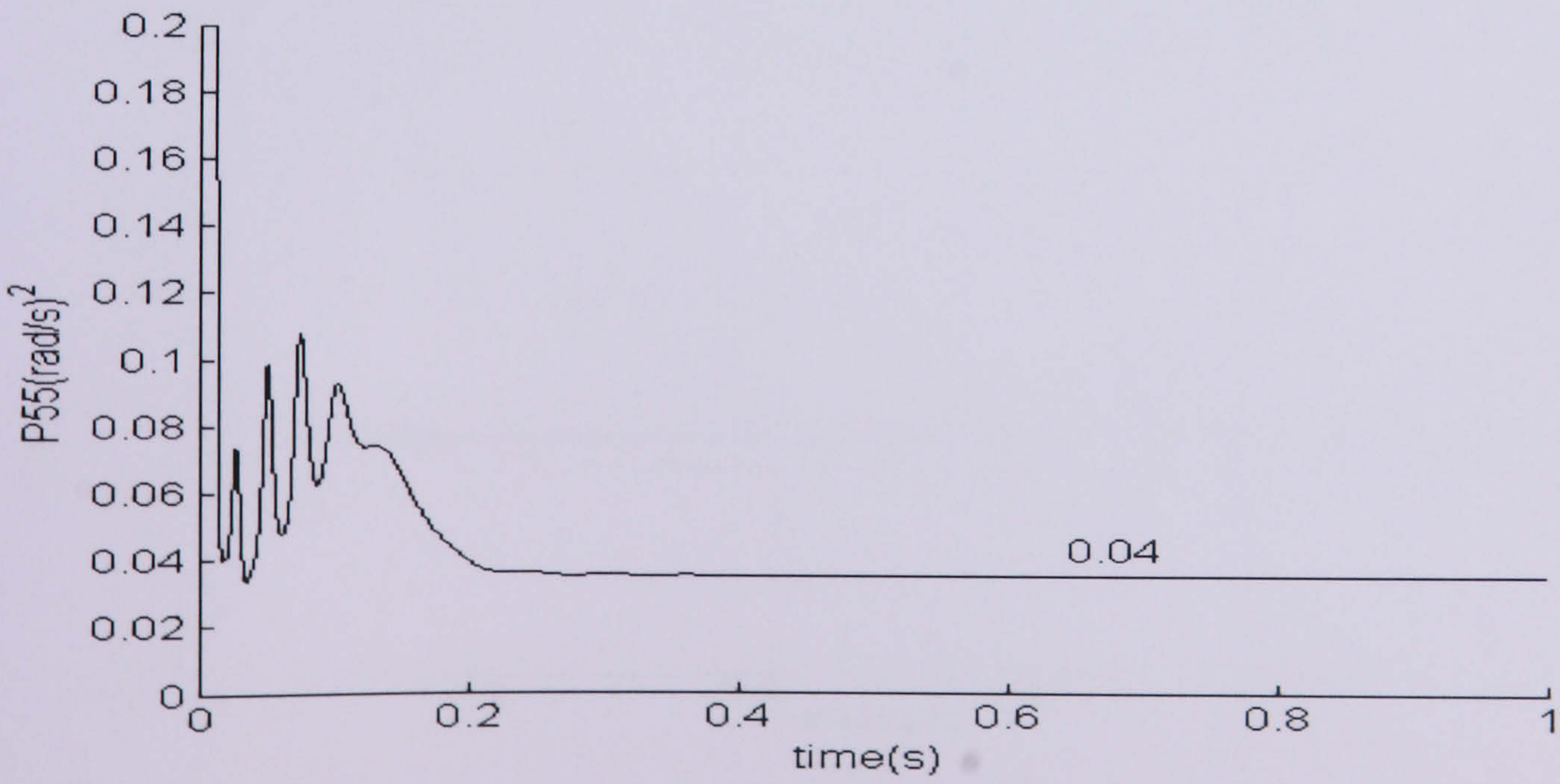


Figure 5.9: P55, error of reference and estimated speed

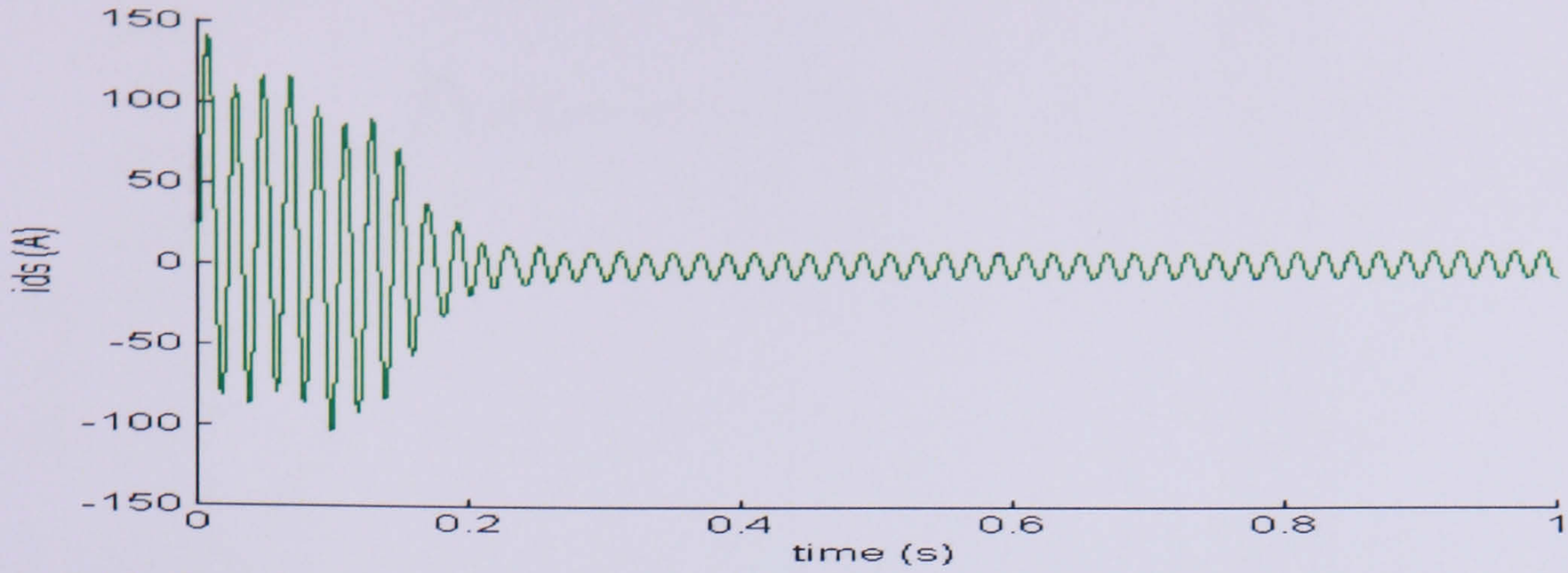


Figure 5.10: D-axis stator current for estimated and reference respectively

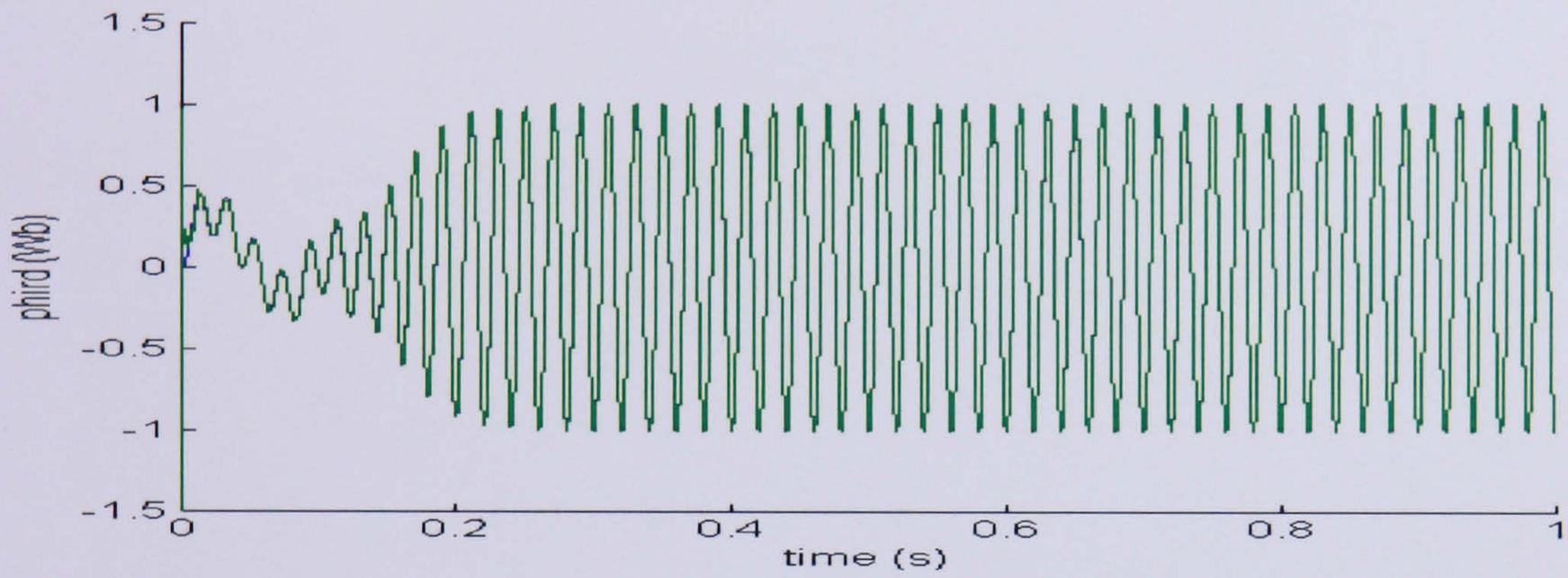
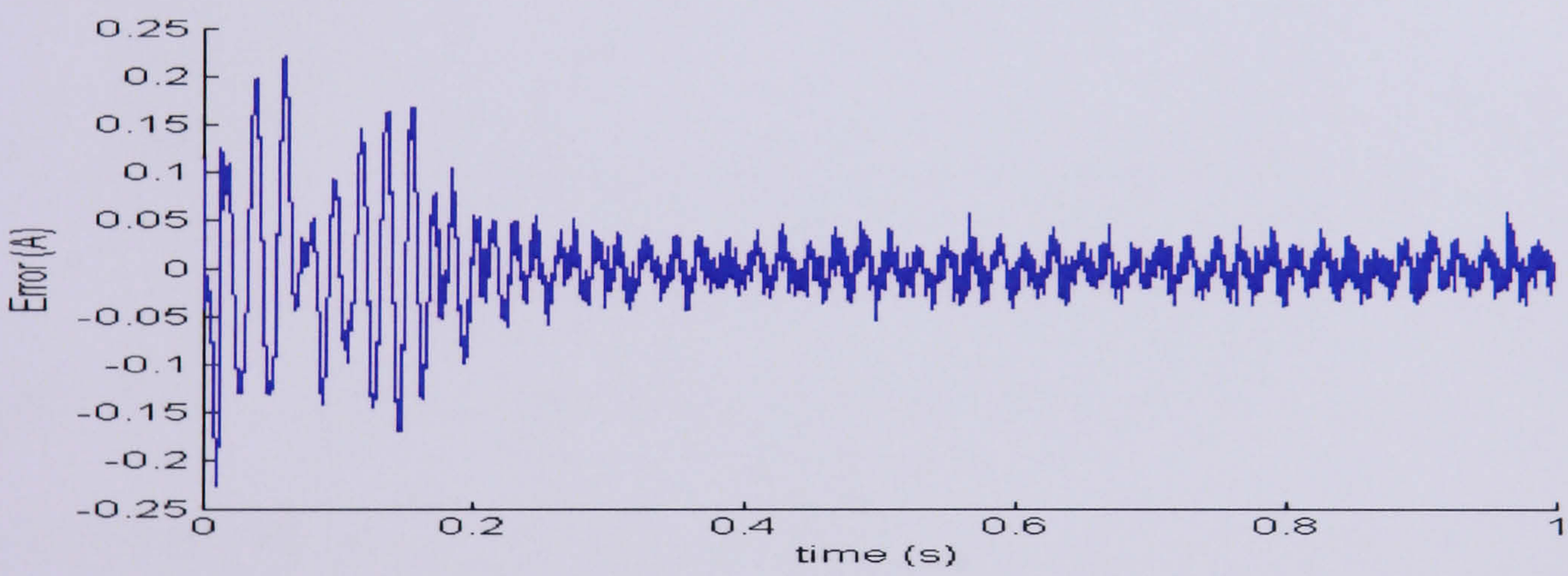
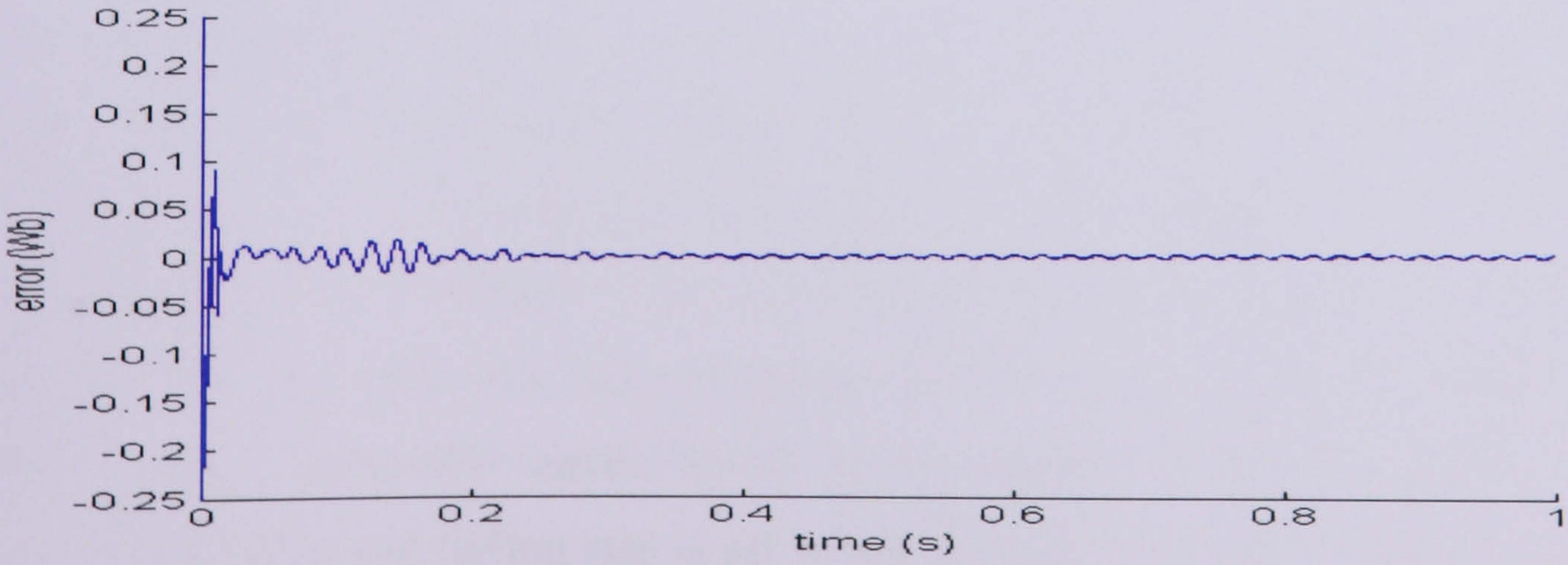


Figure 5.11: D-axis rotor flux for estimated and reference respectively



(a)



(b)

Figure 5.12: (a) D-axis stator current error for estimated and reference (b) D-axis rotor flux error for estimated and reference

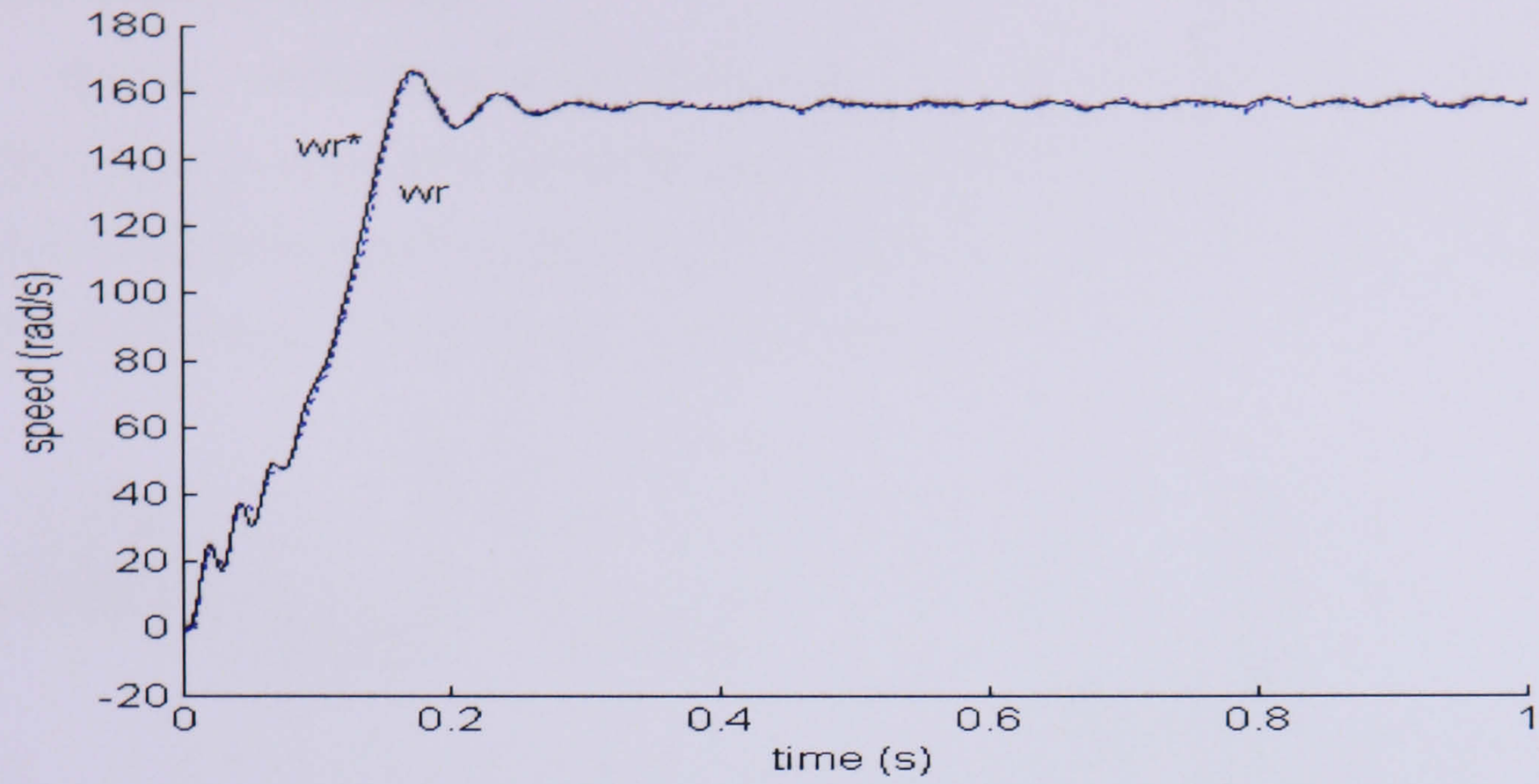


Figure 5.13: Speed responses when process noise is 10% increased

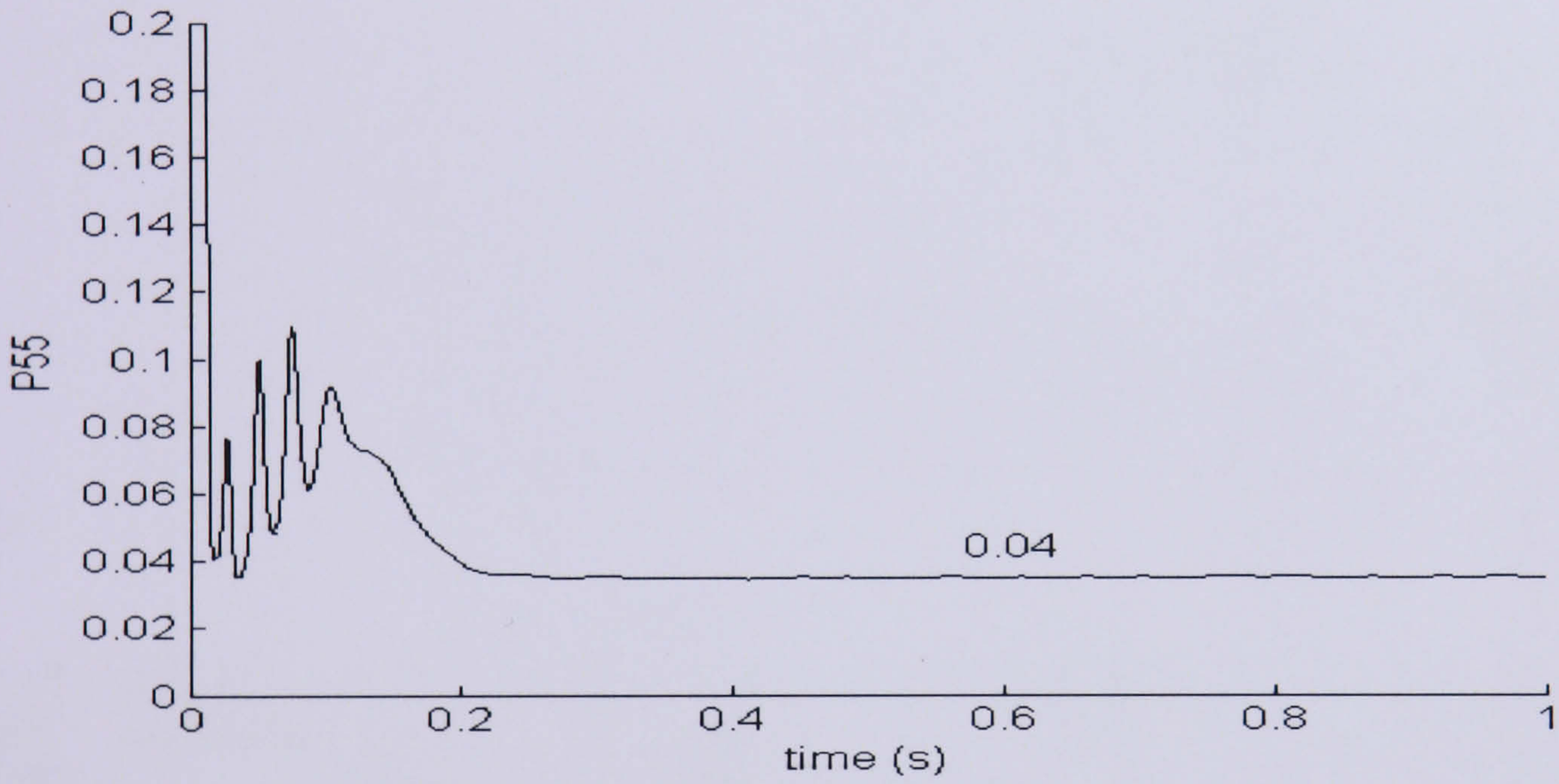


Figure 5.14: Speed responses when process noise is 10% increased

5.5 Simulation using a constant V/f Drive

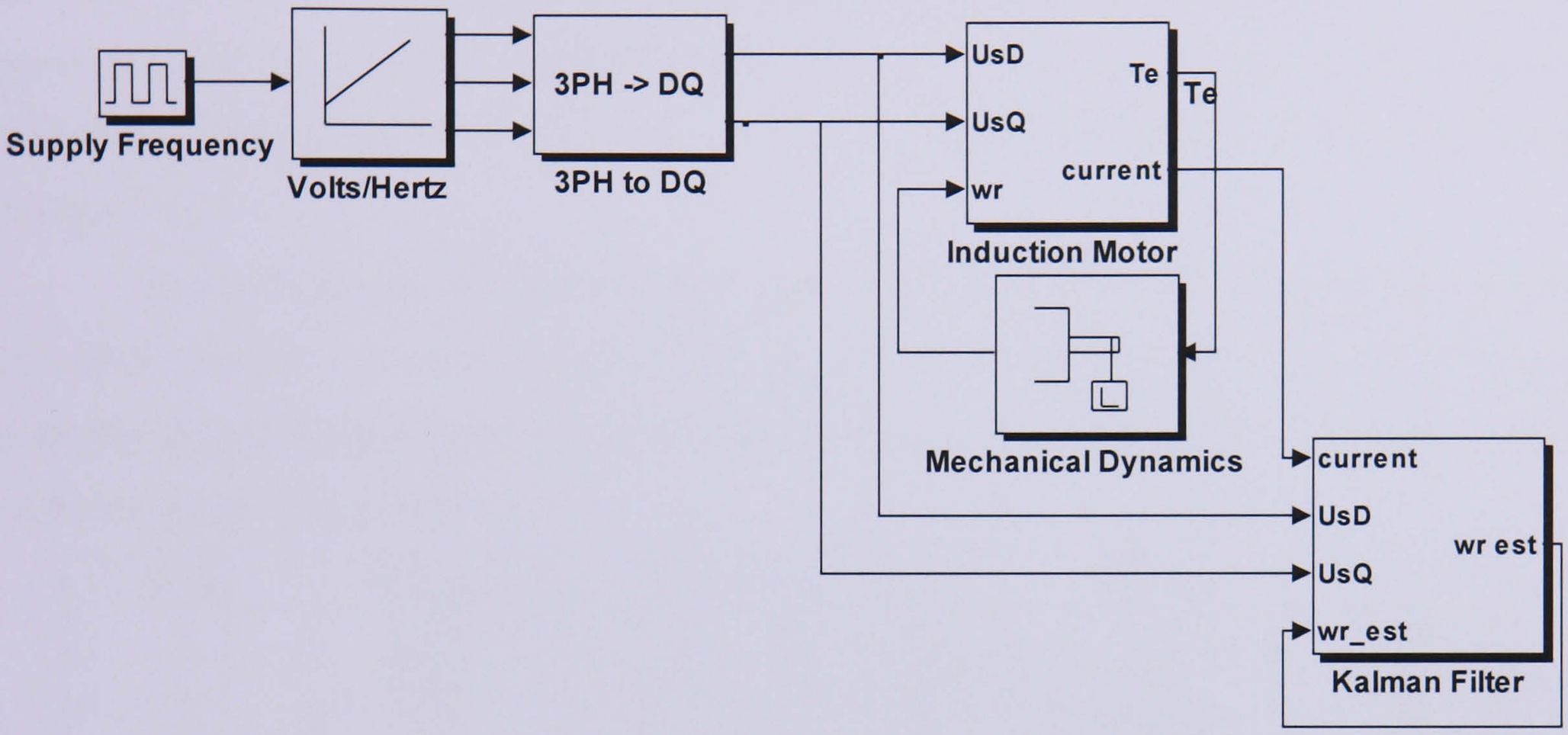
To confirm the validity of the speed estimator, a computer simulation was implemented using an open loop V/f supply. In this simulation, the constant 50Hz supply is replaced with a variable frequency supply based on a constant V/f characteristic. This is shown in Figure 5.15. The demanded frequency is applied to the stator of the motor by a pulse generator in the V/f block. The pulse generator combined with a switch is used for giving the 314 rad/s and -314 rad/s frequency demand. The V/f block integrates the rate limiter to ensure acceptable current levels in the machine by limiting the rate of change demand. The rising and falling rate is set at 600 $(rad/s)^2$. The three output blocks produce the phase A, B and C of voltage demand using equations:

Phase A= cos(u[1])

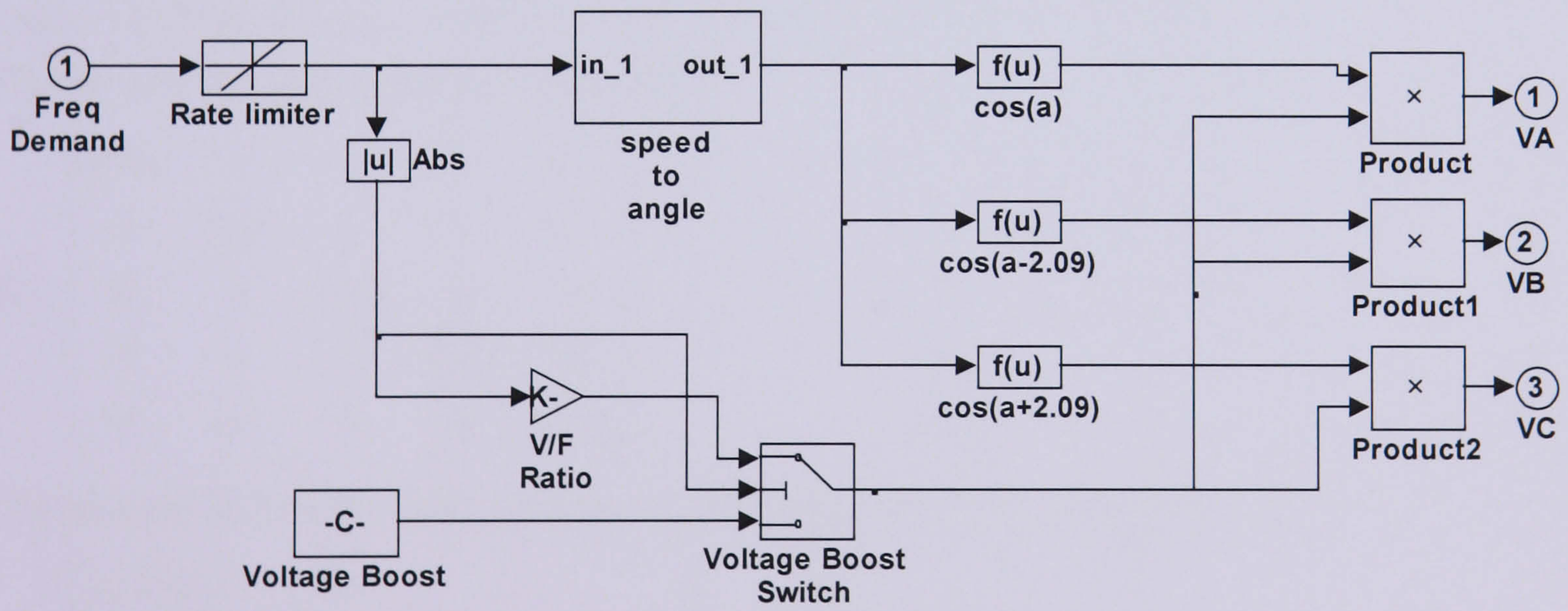
Phase B=cos(u[1]-(2*pi /3))

Phase C=cos(u[1]+(2*pi/3)) (5.33)

u[1] is a radian value obtained by converting the speed demand to angle where the integrator is reset every time the angle reaches a multiple of pi. The magnitude of voltage demand is then obtained by multiplying it with volts/hertz ratio, 0.79V/Hz. A voltage boost of 20V is introduced if the frequency demand drops below a set value.



(a)



(b)

Figure 5.15: (a) Constant V/f of IM drive with EKF estimator. (b) V/f simulation block

The simulation period is set to 2.5s, enough time for the estimator to complete the series of different speed conditions including transient and steady state, acceleration and deceleration, low speed as well as zero speed. The results using a Trial and Error method are summarised in Table 5.2. To achieve the best result the covariance matrices G_{55} and Q_{55}, which correspond to the noise of the estimated speed, need to have larger values than the other covariance elements. It is seen that the estimated speed experiences greater variations than the other state variables [59]. Assumed to be noise free the estimated speed is shown as in Figure 5.16. For the first 5 trials, the performance of the estimator is shown in Figure 5.17.

After a few trials, the best solution was obtained in trial 6 with a mean squared error of 1.0527 and the error percentage of 0.17%. The result as shown in Figure 5.18 clearly indicates that an excellent result, during transient response and steady state operation, can be obtained using Trial and Error.

Trials	$G_{11\text{-}44}$	G_{55}	$R_{11,22}$	$Q_{11\text{-}44}$	Q_{55}
1	0.01	0.01	0.02	0	0.00
2	0.01	0.01	0.02	0.01	0.01
3	0.01	0.01	0.02	0.01	1.00
4	0.01	0.01	0.02	0.01	10.0
5	0.01	0.01	0.02	0.01	100
6	0.01	0.01	0.01	0.00001	1

Table 5.2: Performance of the EKF using a Trial and Error

The set of covariance matrices obtained is:

$$\mathbf{G} = \begin{bmatrix} 0.01 & 0 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0.01 \end{bmatrix}$$

The process noise covariance and the observation noise covariance are given by:

$$\mathbf{Q} = \begin{bmatrix} 0.00001 & 0 & 0 & 0 & 0 \\ 0 & 0.00001 & 0 & 0 & 0 \\ 0 & 0 & 0.00001 & 0 & 0 \\ 0 & 0 & 0 & 0.00001 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

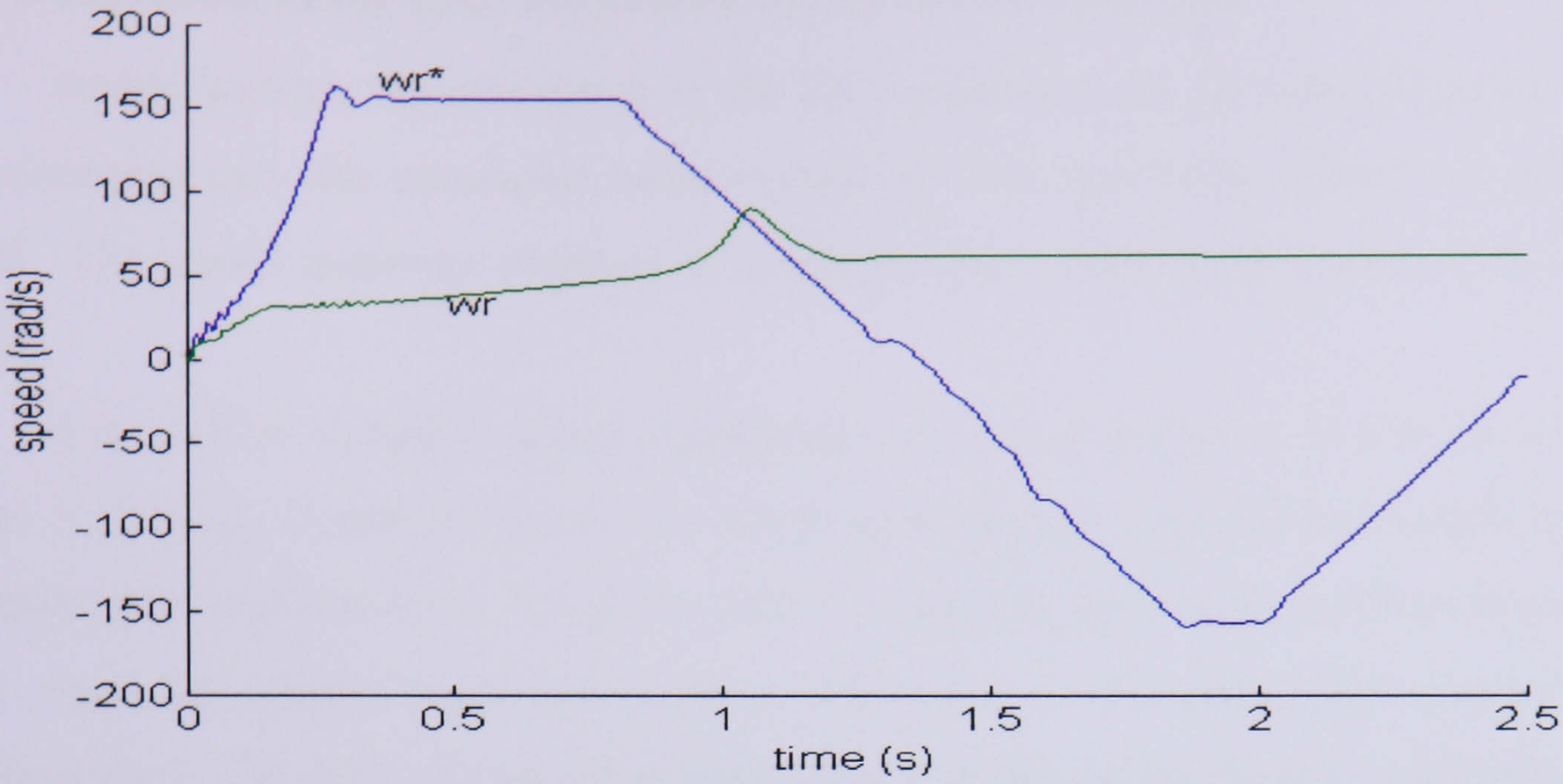


Figure 5.16: Motor and estimator assumed to be noise free

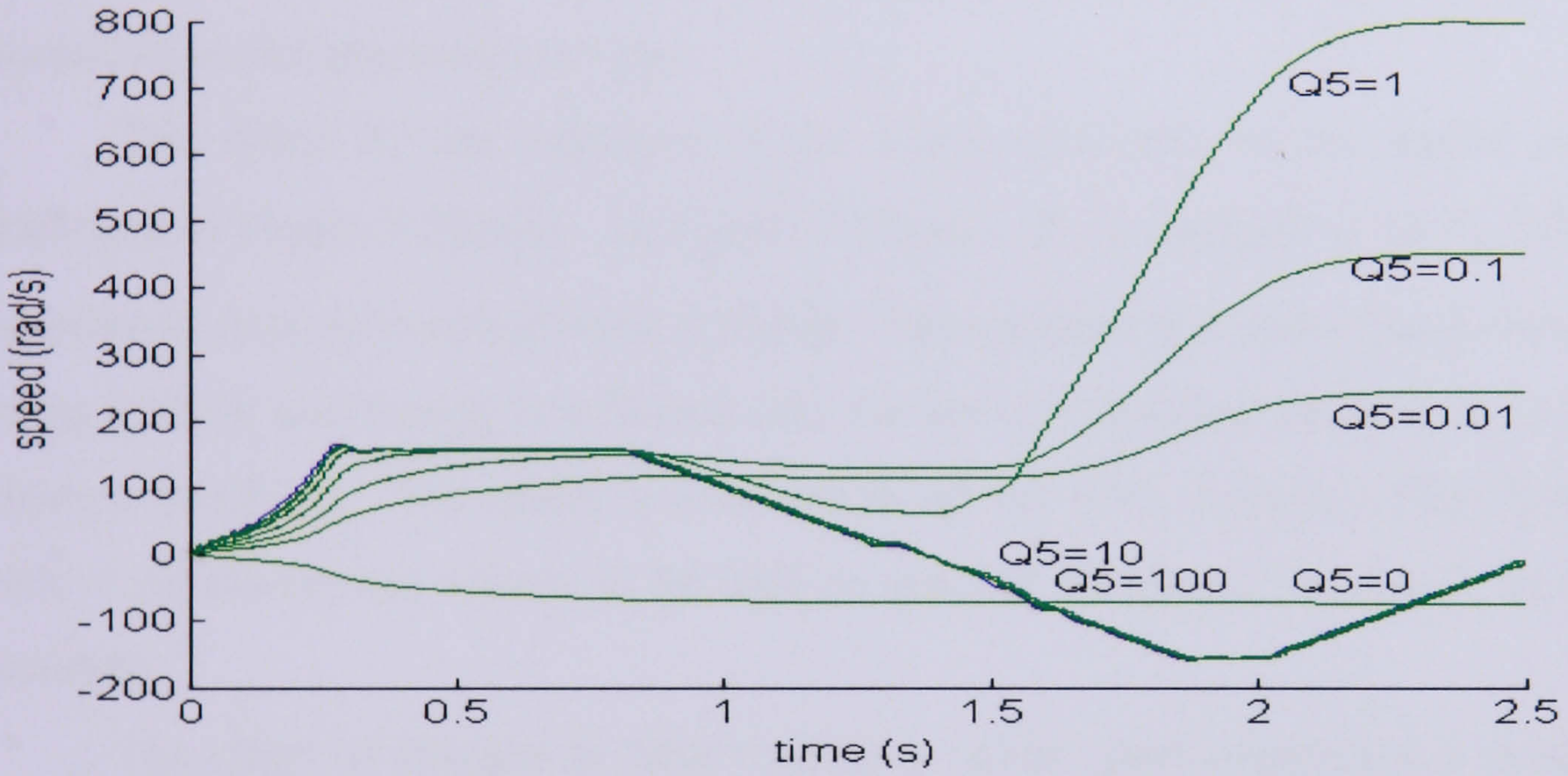


Figure 5.17: The performance of the estimator obtained using various Q_{55}

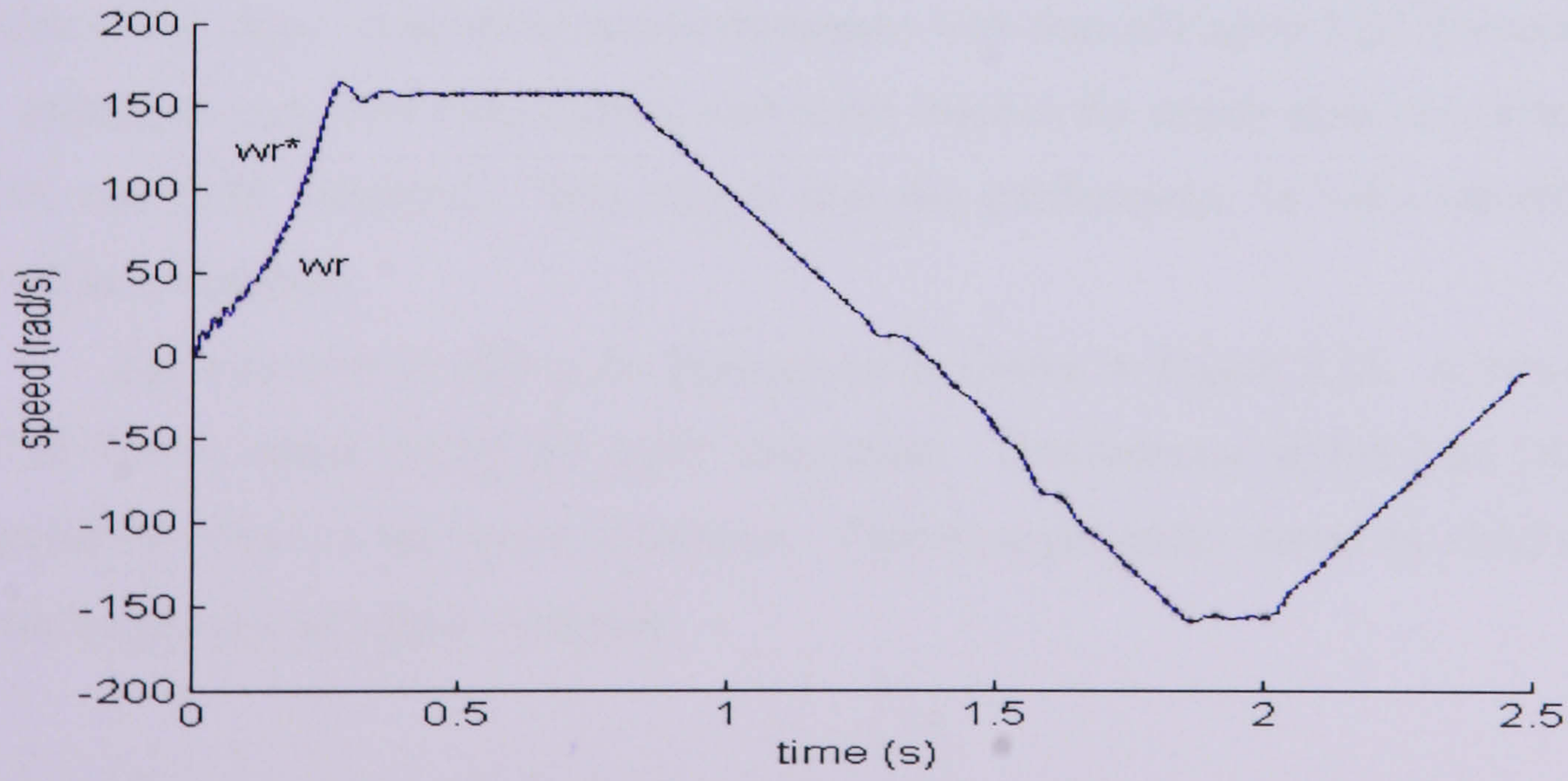


Figure 5.18: Best solution of estimated speed obtained from a Trial and Error method

5.5.1 Robustness of the EKF Parameter on the Speed Response

In this section, the robustness of the EKF parameter on the speed response is given. The estimator uses the parameter values tuned by Trial and Error method to estimate the speed. The speed response obtained is using the best solution discussed in the previous section.

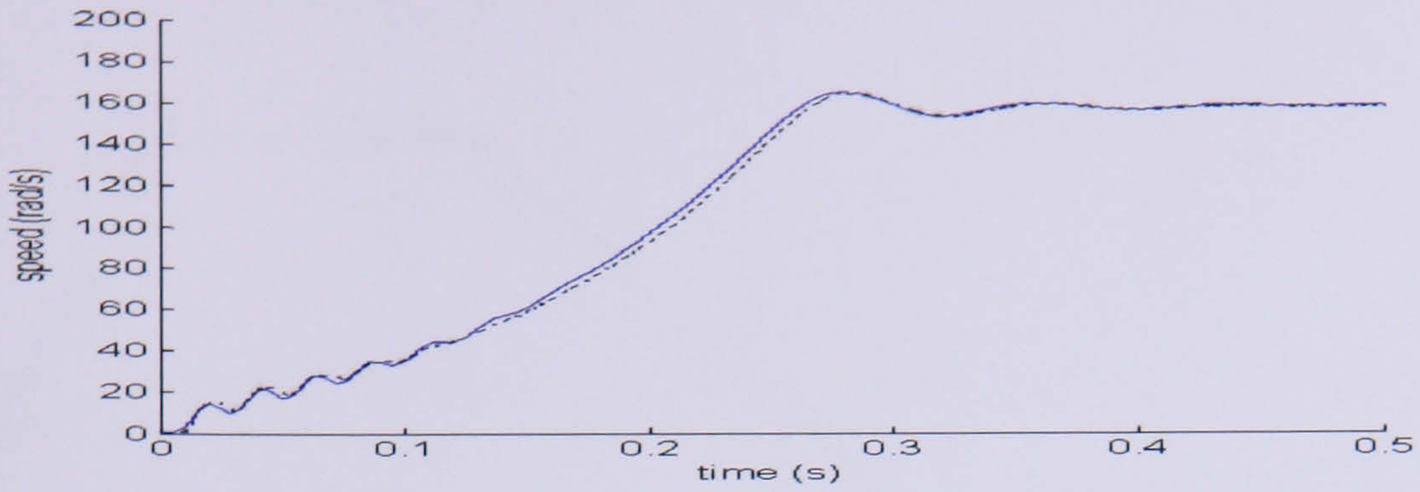
For various dynamic speed conditions, the speed response is shown in detail in Figure 5.19 (a-e). Trace (a) shows the motor speed during transient and steady state, trace (b) shows the acceleration to 160 rad/s, trace (c) showing speed reversal from 6 rad/s to -53 rad/s, trace (d) shows acceleration from -30 rad/s to -52 rad/s and trace (e) shows deceleration to -58 rad/s. From these traces, the two speeds are close to each other in both steady state and transient performance. So this suggests that the speed estimate is insensitive to the operating envelope.

The effect of the variation of the stator resistance in the speed estimation is illustrated in Figure 5.20(a-c). In Figure 5.20(a-c), R_s is changed to 10 %, 25% and 50% respectively from its nominal value 0.6 ohm. This mismatch of motor parameters during the tuning process and testing is reflected into the speed estimation particularly in the start-up transient condition. The effect is even worse on the 50% variation. This Trial and Error method parameter has shown to be able to recover the speed estimation in steady state condition.

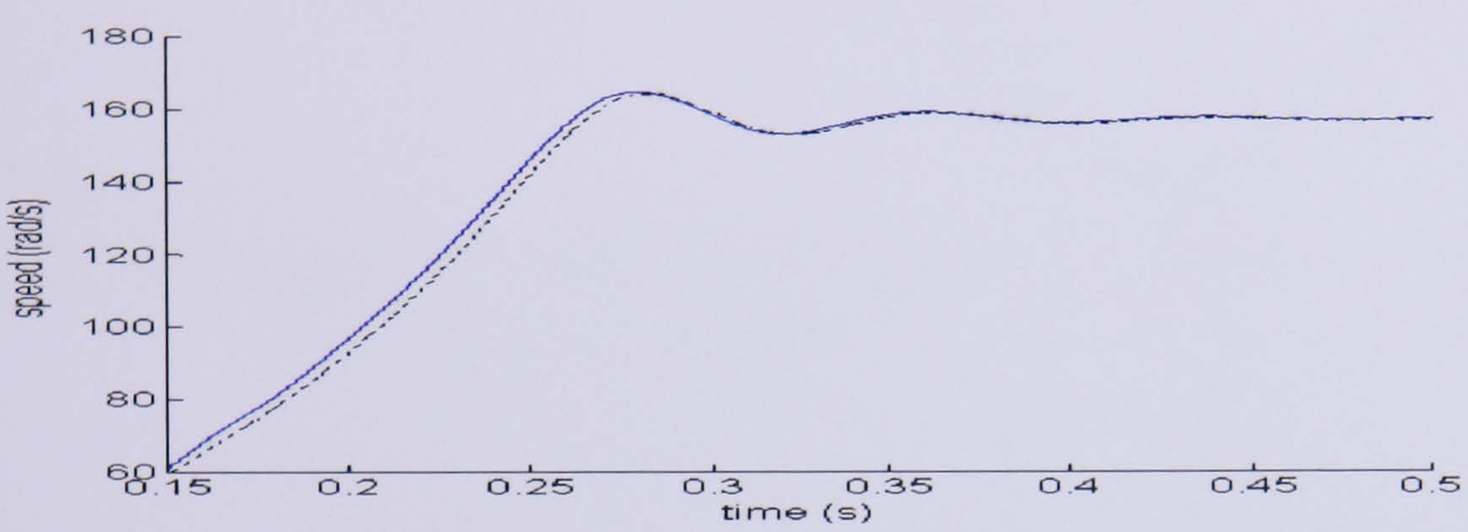
The effect of changes in rotor resistance on the speed estimation is demonstrated in Figure 5.21(a-c). The rotor resistance is increased to 10 %, 25% and 50% from its nominal value of 0.4 ohm. Comparing this performance with that of Figure 5.20, the speed estimate is undergoes very slow convergence and never reaches the steady state condition during the 10% and 50% variation. This shows that the performance is very sensitive to rotor resistance variation.

The outcome of adding the load torque is shown in Figure 5.22. A step load torque of 20 Nm is added during the 0.65s simulation. The increase in the load torque has no significant effect on the speed estimation. Therefore parameter tuned by Trial and Error is robust against load torque variation.

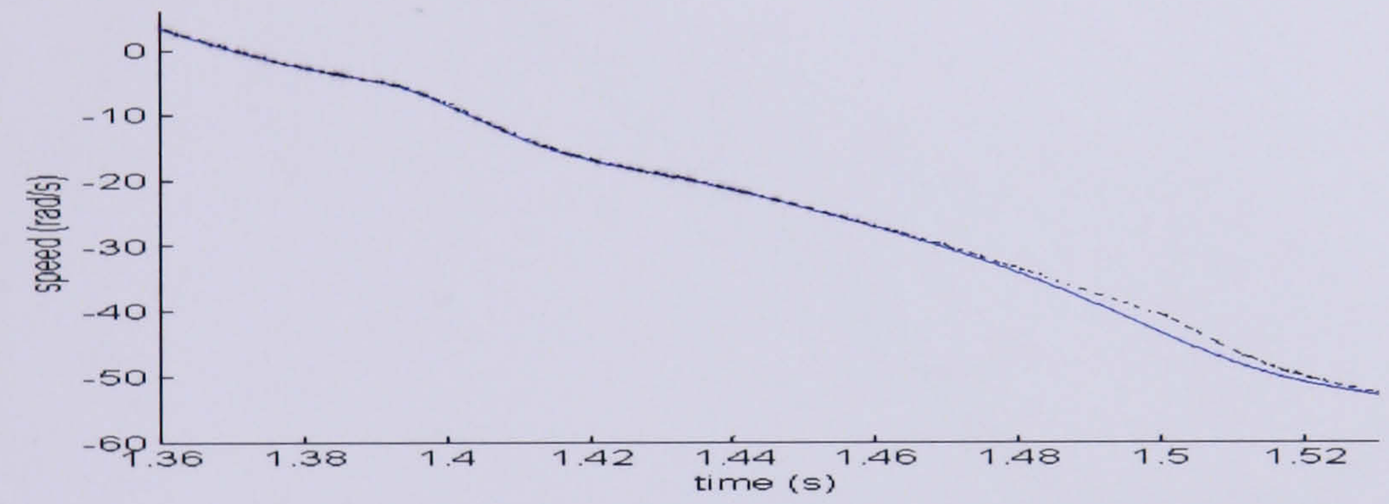
A. Various dynamic speed conditions



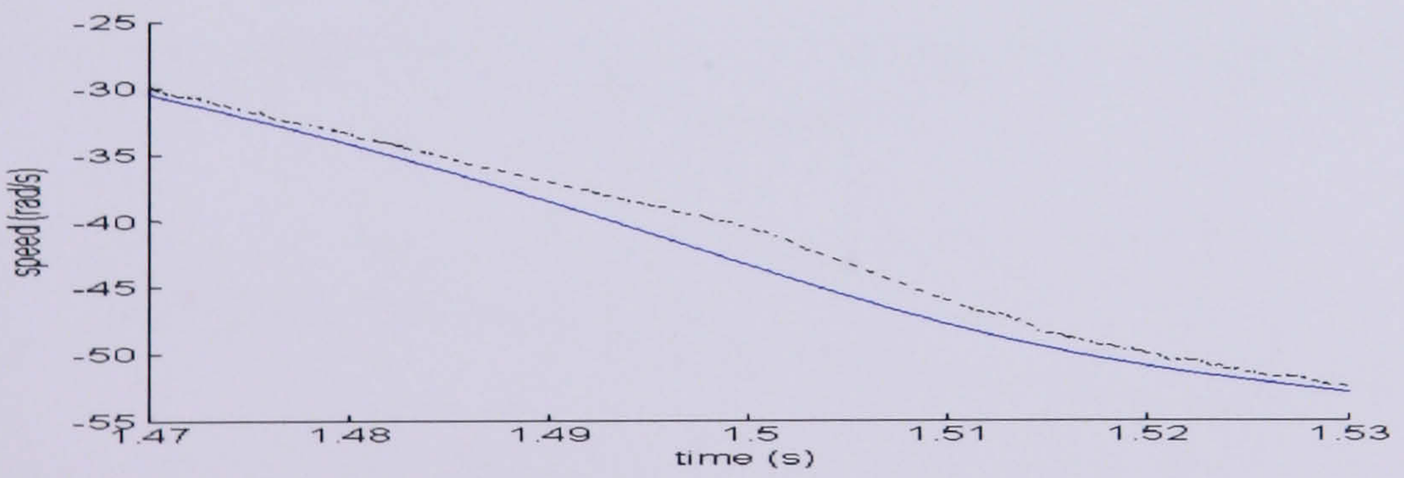
(a)



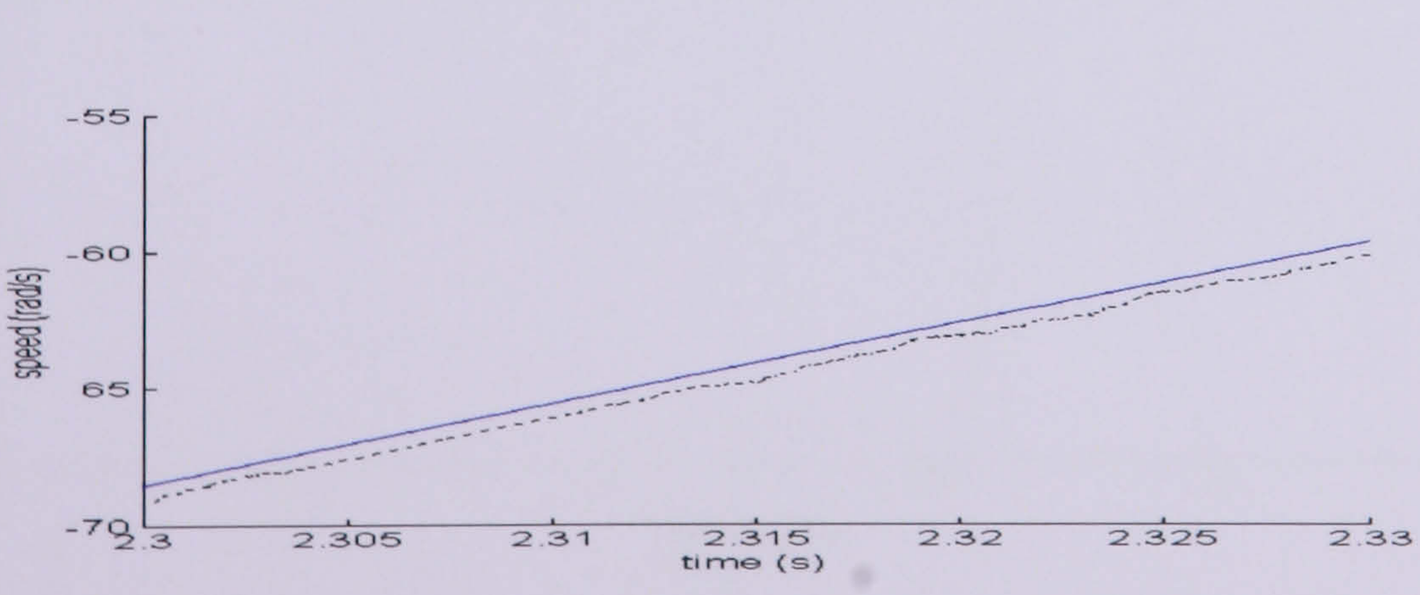
(b)



(c)



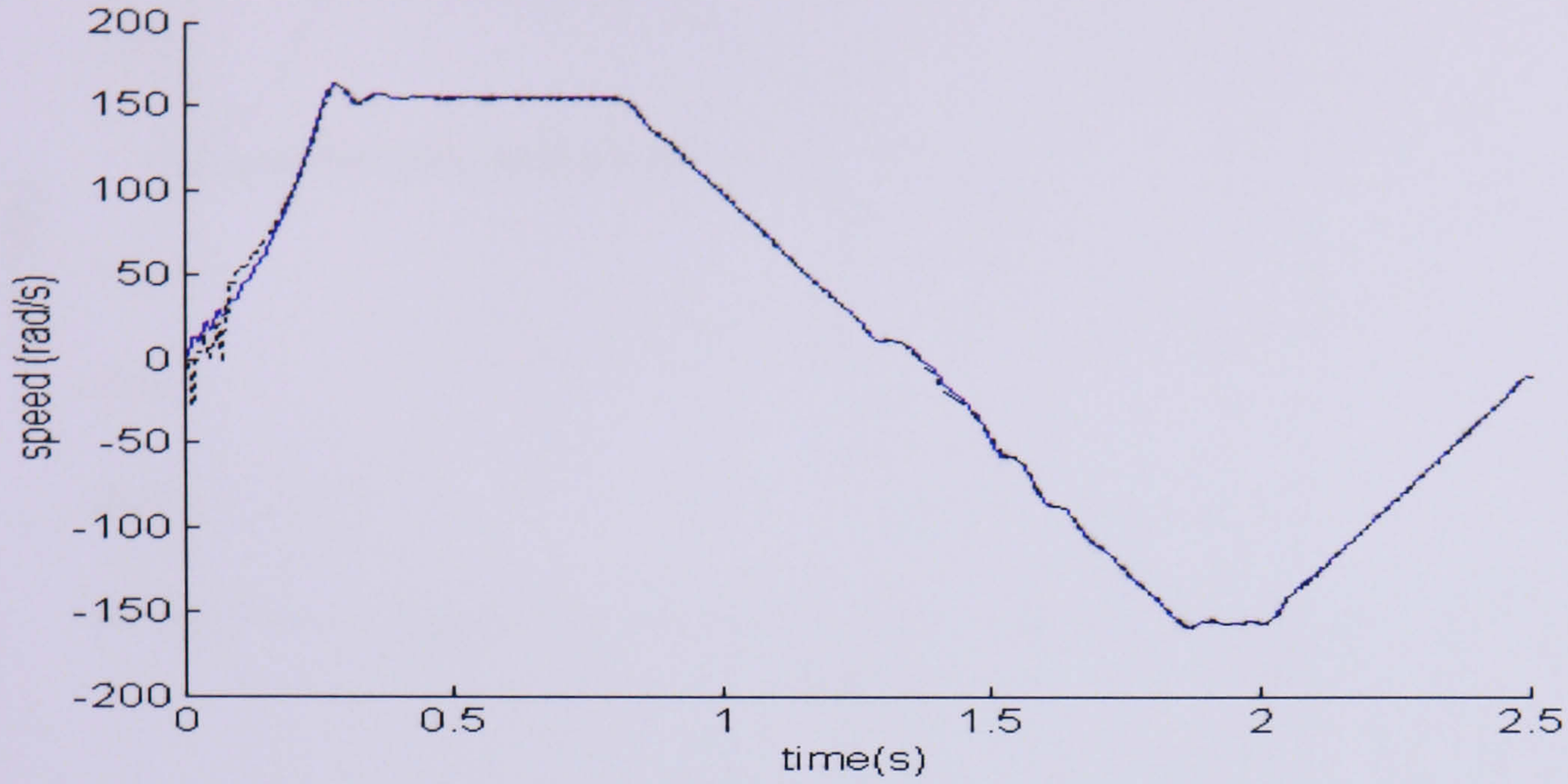
(d)



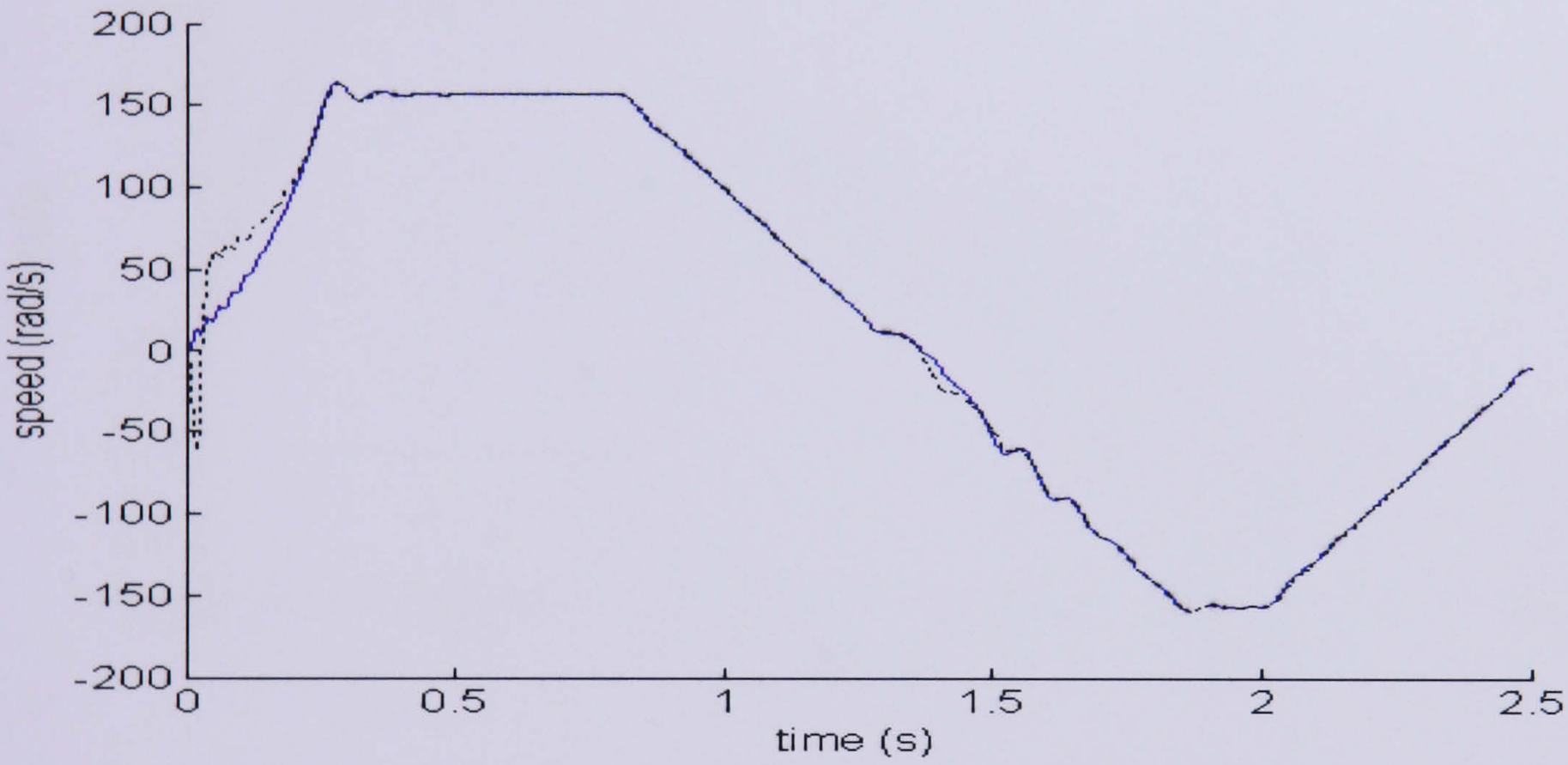
(e)

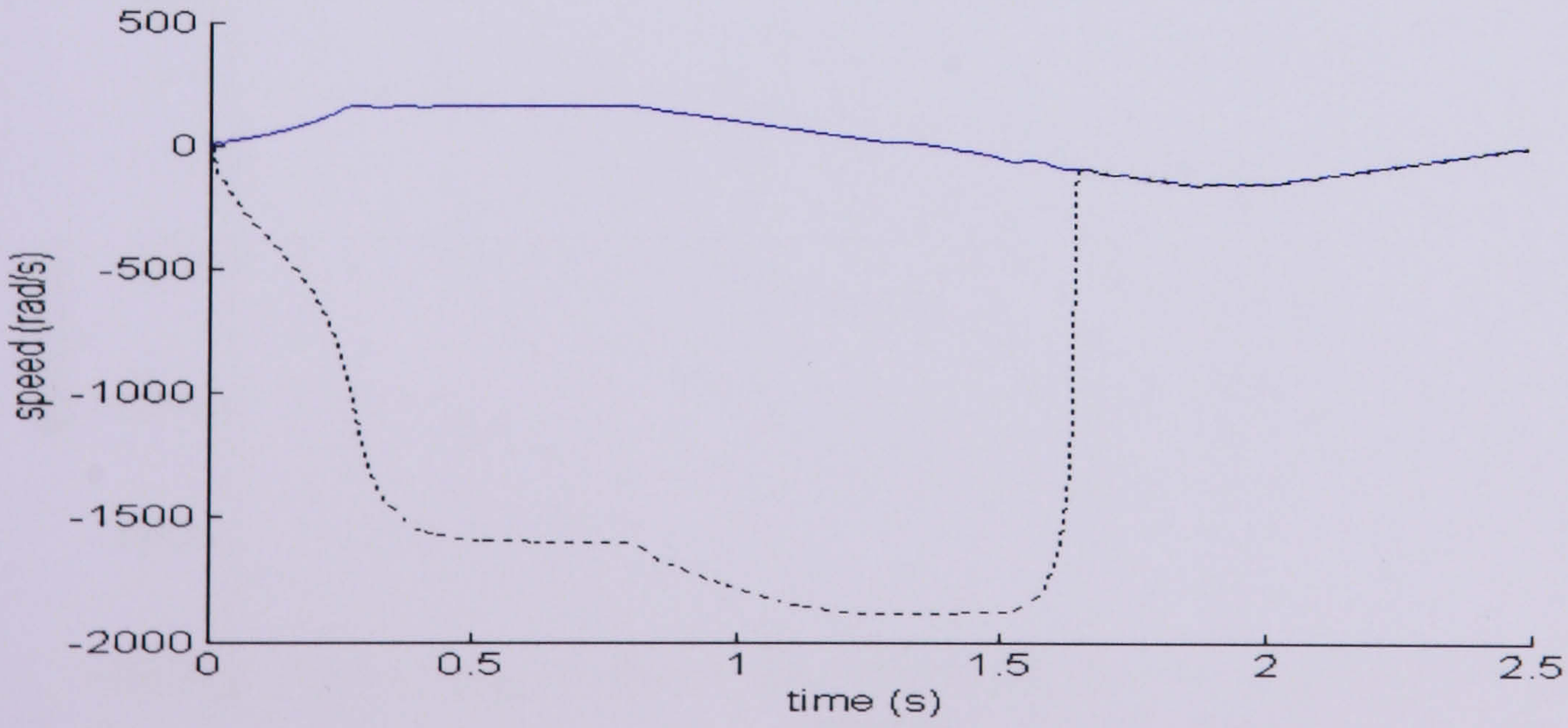
Figure 5.19: (a) Motor speed during transient and steady state (b) Acceleration to 160 rad/s (c) Speed Reversal from 6 rad/s to -53 rad/s (d) Acceleration from -30 rad/s to -52 rad/s (e) Deceleration to -58rad/

B. Sensitivity of the tuning parameter to parameter variation.



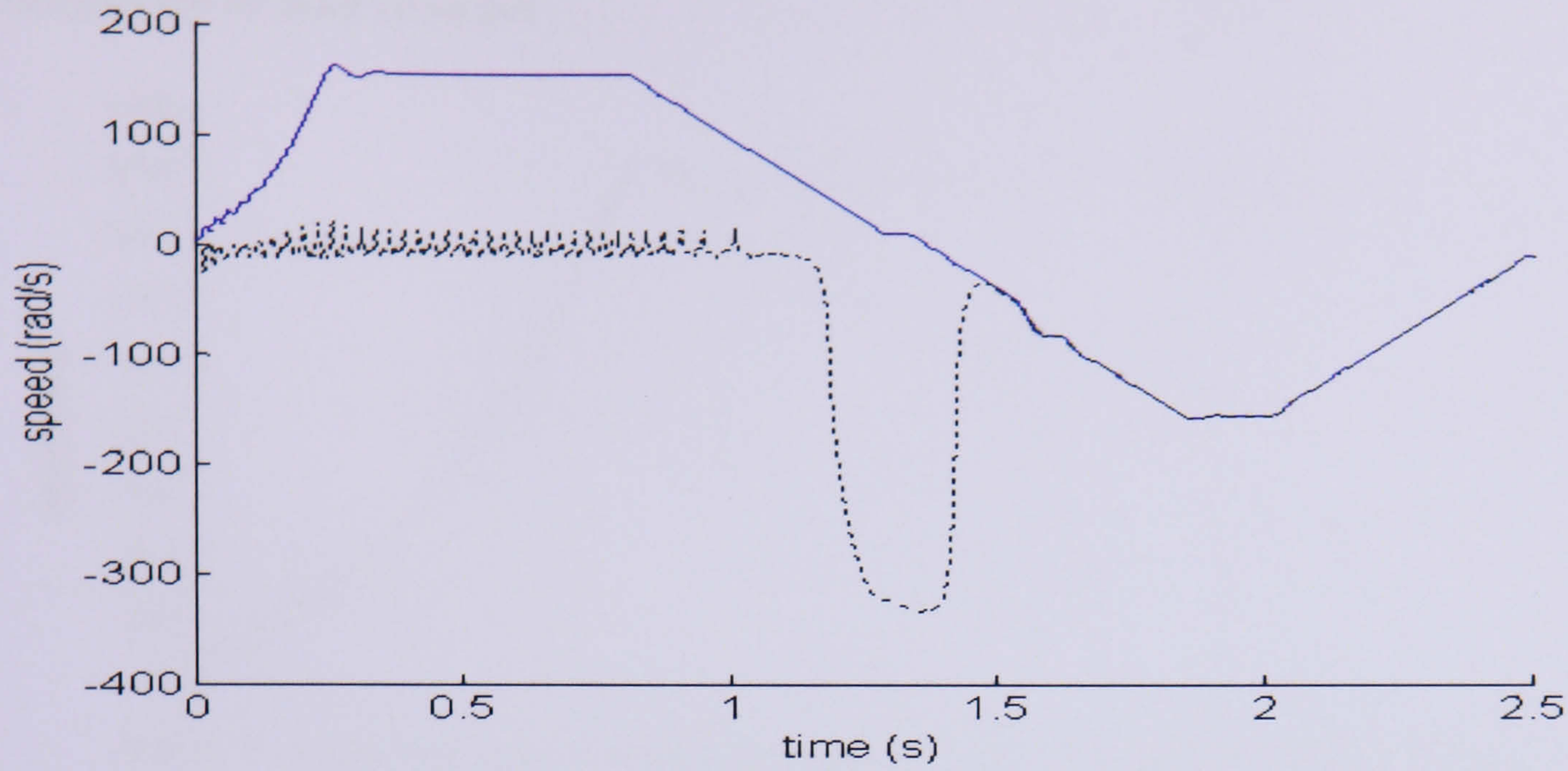
(a)



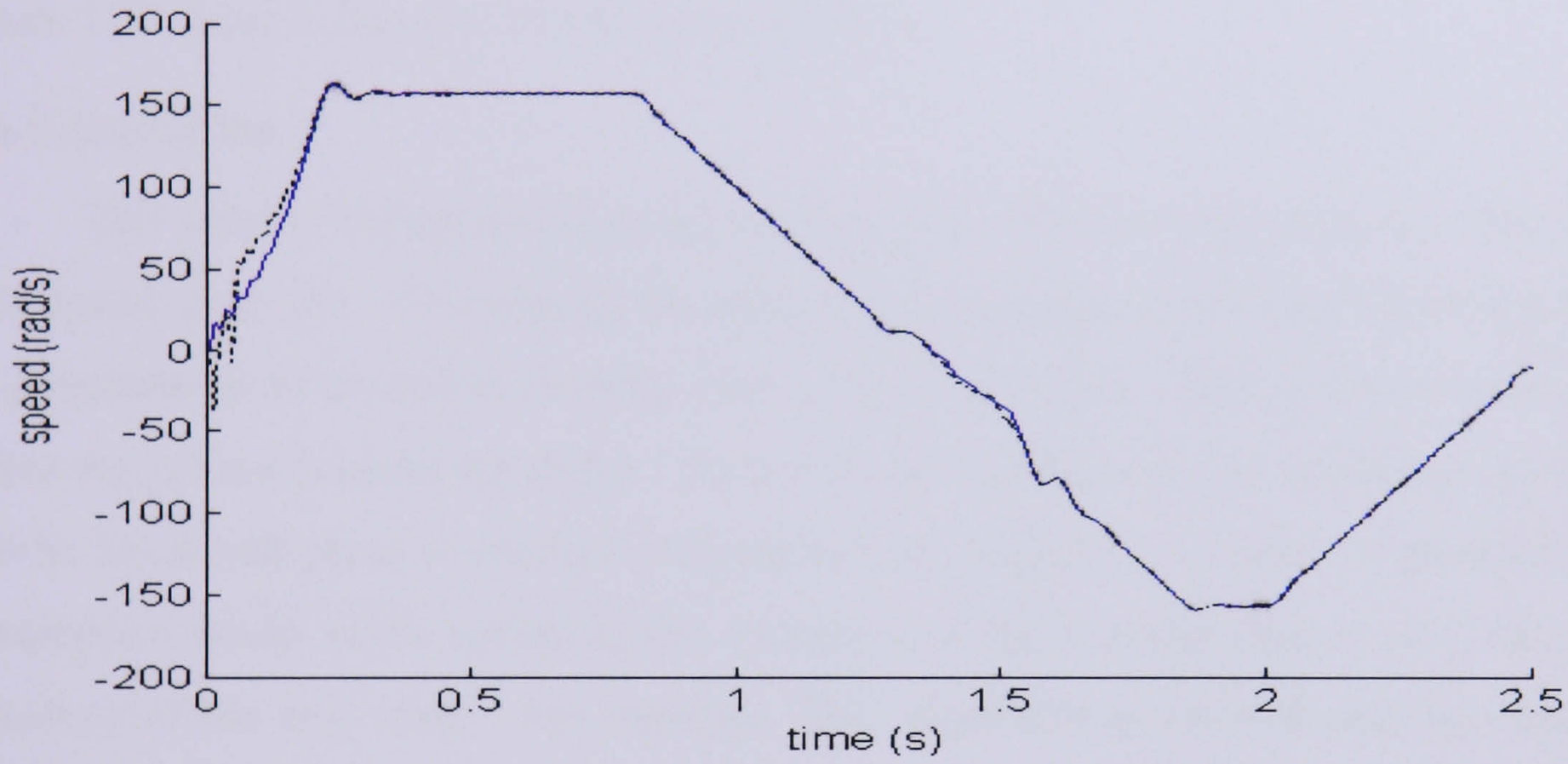


(c)

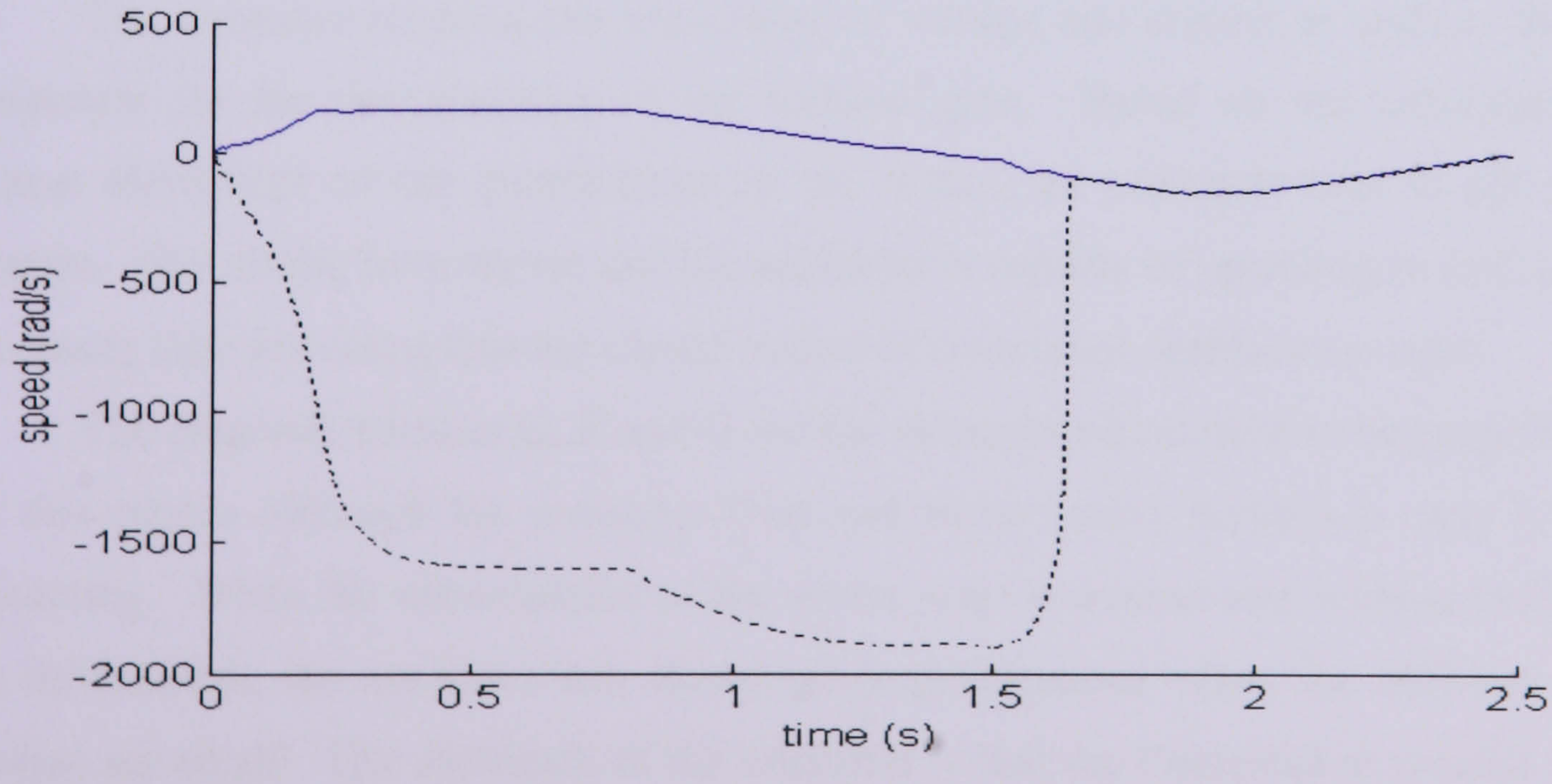
Figure 5.20: (a) Stator resistance variation increased by 10% (b) Stator resistance variation increased by 25% (c) Stator resistance variation increased by 50%



(a)



(b)



(c)

Figure 5.21: (a) Rotor resistance variation increased by 10% (b) Rotor resistance variation increased by 25% (c) Rotor resistance variation increased by 50%

C. Sensitivity to load changes

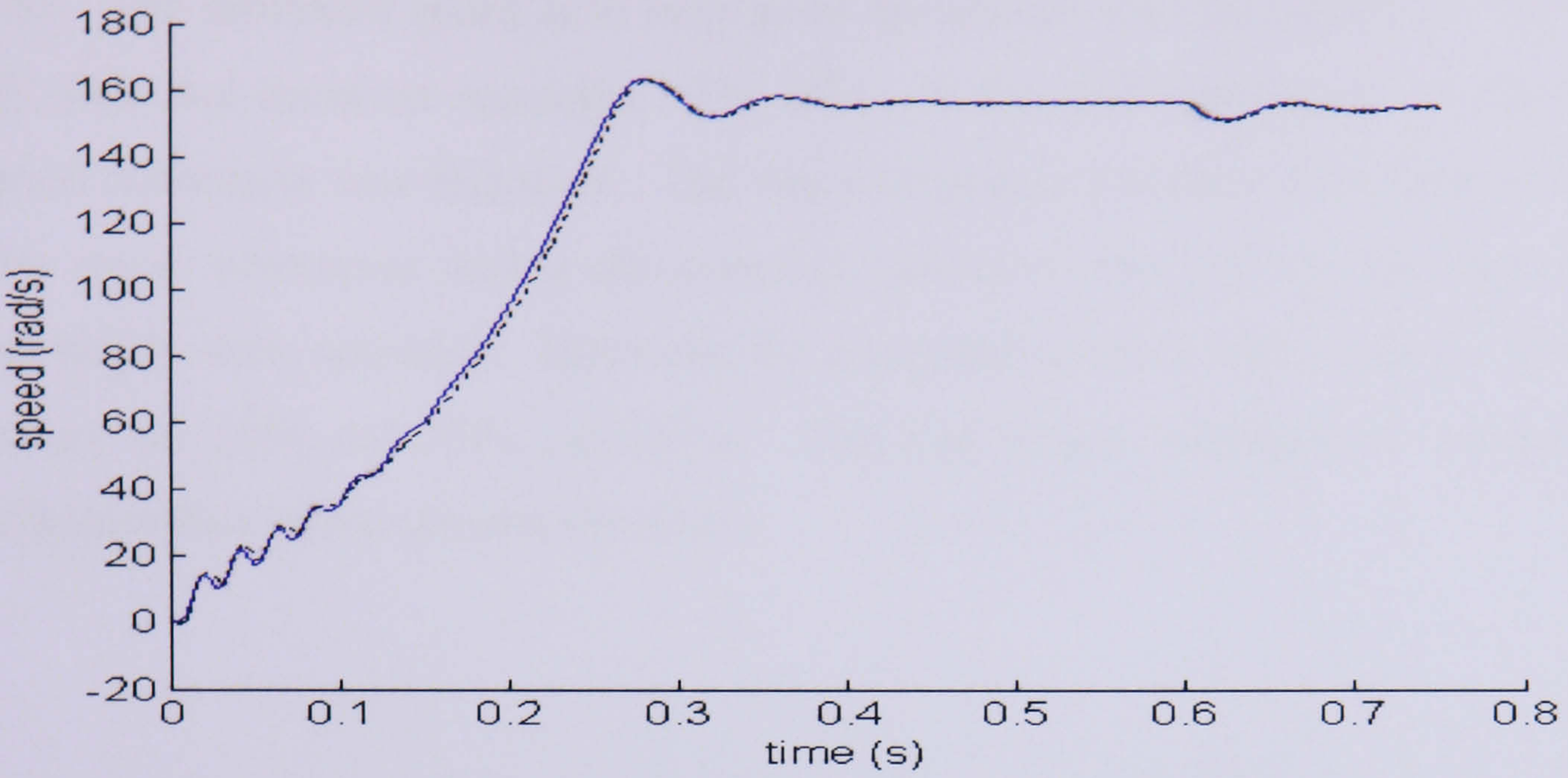


Figure 5.22: Load Torque of 20 Nm added at 0.65s

5.6 Conclusion

This chapter has presented an application of an EKF for estimating the rotor flux and rotor speed of an IM. To estimate the motor states and parameter simultaneously requires the parameter to be treated as an extra state in the state vector. This multiplication of states makes the system become nonlinear. To use the KF application, the nonlinear system must first be linearized about a nominal (reference) state trajectory in order to produce a linear perturbation model while assuming the trajectory of the nominal state is very close to the trajectory of the real state. The standard EKF algorithm is derived and then applied in simulations.

The estimator required the knowledge of voltage and current as well as the motor parameters for the determination of the Kalman gain. Based on this information and without knowledge of the uncertainties of the motor, the estimator tries to get the best estimate. The results have shown that the estimator is capable of operating in both transient and steady state providing that the correct values of covariance matrices are used.

The diagonal matrices Q, R and G are the elements which need to be properly tuned. For this reason although the common Trial and Error tuning method is easy it is time consuming. When the uncertainties of the motor were increased and without being given this information, the estimator has shown good performance when the differences from nominal are small. The drawback of the estimator is that the linearization process requires the gain to be computed in real-time and this yields long computation times.

The speed response obtained using the EKF is shown in detail for various operating regions. The estimated speed is in very good agreement with the reference speed in both steady state and transient operation. The effect of the EKF parameters on the parameter variation robustness was discussed. The stator resistance variation does have a little effect on the speed estimation during the transient operation, but good results were obtained during steady state operation. However, the estimated speed is badly affected by the rotor resistance for 10% and 50% variations. The load torque variation by contrast has no significant effect on the speed estimation.

CHAPTER 6: SIMULATED ANNEALING ON DIRECT SUPPLY AND CONSTANT V/f

6.1 Introduction

The EKF performance strictly depends on the noise covariance matrices, and so these matrices need to be properly selected. Conventionally Trial and Error schemes dominate EKF controller tuning [55, 58]. However, Trial and Error methods depend on personal skill and although this method is easy to use, it can be time consuming. In addition, the selected value matrices may not be accurate and optimal. A few methods have been proposed to replace the Trial and Error method. They include the GA [59], simplified GA [60], normalisation of the EKF algorithm [61], fuzzy logic [62], Taylor series expansion and by Monte Carlo simulations [63].

SA which chooses a path through the design space, has been successfully applied to many system optimisation problems. SA is a viable approach to finding optimal, or near optimal solutions for large scale problems. In this chapter, a new approach for tuning the EKF is proposed using SA. The attractive feature of SA is that it is very easy to program and the algorithm typically has few parameters that require tuning. The performance of the proposed algorithm on the EKF is demonstrated through simulation. It is expected that the robustness of the new proposed method will give a better result or will be as good as the conventional method. In this new approach, the SA is first investigated using a direct sinusoidal supply and then, with the same parameters, is applied to a constant V/f induction drive.

6.2 Simulated Annealing

This method was first introduced by Kirkpatrick et al [87]. Annealing is the process of heating up a solid and then cooling it down slowly until it crystallises. The atoms in the material have high energies at high temperatures and have more freedom to arrange themselves. As the temperature is reduced the atomic energies decrease. A crystal with regular structure is obtained in the state where the system has minimum energy [88].

The algorithm of the SA mainly consists of repeating a sequence of iterations. Given an optimisation problem, at a selected initial temperature, the SA starts off with the

initial solution: current and trial, randomly selected from two points within the search space. Two energy level sets of current solution and trial solution, E_i and E_j respectively, are obtained. The Metropolis algorithm, generation and acceptance, based on Metropolis et al [89] is then applied. In the acceptance rule, if $E_j - E_i < 0$, then the trial solution is accepted and replaces the current solution. Otherwise, the trial solution is accepted based on Boltzmann's probability, which is

$$Pa = \exp(-\frac{E_j - E_i}{kT}) \tag{6.1}$$

where T denotes the temperature and k is known as Boltzmann constant.

If the probability acceptance is higher than R, where R is a random value [0, 1], the trial solution is accepted and replaces the current solution. If the probability acceptance is less than R then the current solution remains and a new trial solution is generated. The generation mechanism and acceptance criterion are then repeated. After certain iterations the temperature is reduced. Given reduced temperature, these two processes are repeated again until the criterion of execution is achieved.

6.3 The Application of the Simulated Annealing to EKF

This section proposes the application of the SA algorithm to solve the combinatorial optimisation of Q, G and R parameters of the EKF. The general step of the SA algorithm is described as a flowchart in Figure 6.1 and can be described as follows:

Step 0: Initialize the initial temperature, T_{start} and counter.

Step 1: Generate some initial random solution, X_i.

Step 2: Evaluate the objective function value E_i of the current solution.

Step 3: If the temperature condition is satisfied go to step 10.

Step 4: If the iteration condition is satisfied go to step 9.

Step 5: Generate new solution, X_j.

Step 6: Evaluate the objective function value E_j of the trial solution.

Step 7: Acceptance test: if E_j-E_i<0, then store the new trial solution into the best. Otherwise calculate the Boltzman probability (Pa) with the current temperature. If the Pa > [0,1], store the new trial solution and into the best. Otherwise, the current solution and solution remain the same.

Step 8: If the objective function remains the same for consecutive times, go to step 9 Otherwise go to step 4.

Step 9: Reduce the temperature and go to step 3.

Step 10: Stop if termination criteria is applied.

6.3.1 Representation

The parameters to be optimised can be represented by two common methods. The first method uses a binary string and the second uses a vector of integers or real values. These two methods might cause differing performance in terms of accuracy and computation time [88] of optimisation. The performance of the filter is very much dependent on the right noise levels of the covariance matrices, **Q**, **G** and **R**, where

$$Q = diag[Q_1 \quad Q_2 \quad Q_3 \quad Q_4 \quad Q_5]$$

$$G = diag[G_1 \quad G_2 \quad G_3 \quad G_4 \quad G_5]$$

$$R = diag[R_1 \quad R_2]$$

In this algorithm, the parameters are represented by short format real value numbers, and the example of the random initial solution used is given as follows:

$$X_i = [0.0074 \quad 0.0098 \quad 0.0049 \quad 0.0000 \quad 0.0078 \quad 0.0035$$
$$0.0030 \quad 0.0083 \quad 0.0010 \quad 0.0037 \quad 0.0064 \quad 0.0003]$$

where $X_i = [Q_1 \quad Q_2 \quad Q_3 \quad Q_4 \quad Q_5 \quad G_1 \quad G_2 \quad G_3 \quad G_4 \quad G_5 \quad R_1 \quad R_2]$

6.3.2 Creation of an Initial Solution

The SA requires an initial solution at the start of optimisation. The initial solution can be formed in two ways: either by a random generator number, or by using an approximation [88]. The advantage of using the first method is that the initial solution is produced without a prior knowledge of the optimisation problem. The approximation method on the other hand employs a prior knowledge of the algorithm based on certain requirements and starts the optimisation with an approximate solution in hand. The advantage of this method is it converges to an optimal solution in less time than the former method. In this study the initial solution is tested via both methods, starting with use of the random number and followed by an approximation method.

6.3.3 Objective Function

The mean squared error is the expected value of the square of the error. This error is the difference between the speed from the motor and the estimator. In this work, the term

objective function is used for the mean squared error. The objective function is calculated at all sampling time periods.

$$E = \frac{1}{n}\sum_{i=0}^{n}(\omega_{ri} * - \omega_{ri})^2 \tag{6.2}$$

where $\omega_r * = actual\_speed$ $\omega_r = estimated\_speed$

6.3.4 Generation Mechanism of Trial Solution (Neighbour)

The generation mechanism corresponds to the perturbation mechanism, whereby the mechanism is chosen so that the new solution in the neighbour can be obtained by a simple re arrangement that is computed rapidly. The neighbour should be randomly generated, feasible, and span the problem space as much as possible. Some of the generation mechanisms used by researchers are from the common perturbation mechanism such as the roulette wheel [90] and stochastic universal sampling [59]. Some have used their own perturbation method that suits their algorithm [64,91,92]. In this application, a new solution is employed from a neighbour using a random value of (0-1).

6.3.5 Cooling Schedules

The cooling schedule consists of the initial temperature, temperature decrement control, iterations at each temperature and final temperature. The temperature must be controlled so that it is large enough to move off the local minimum, but small enough to move off the global minimum. The initial temperature must be high enough to allow a move of each solution to almost any neighbourhood state space. The main concern for not using a high initial temperature is that there will be little change in the solutions, and it will end up with the final solution staying very close to, or similar to the initial solution. Several temperature decrement controls have been proposed. In [90,91,93], the temperature is lowered in series, $T=\alpha T$. The typical value of α lies between 0.8-0.99. Another decrement technique proposed by [94] makes the temperature inversely proportional to the number of potential solutions investigated. At each temperature, the temperature is held constant for an appropriate period of time, or a number of iterations in order to allow the algorithm to reach a steady state point or equilibrium state [91-93]. For the final temperature, although the annealing process can keep decreasing until the temperature reaches zero, it may take a while. To reduce the computation time, some applications set the maximum amount of temperature reduction [64] and continue decreasing the temperature until a stable code

solution was obtained [91], or until the current solution remains unchanged for a number of consecutive chains [92].

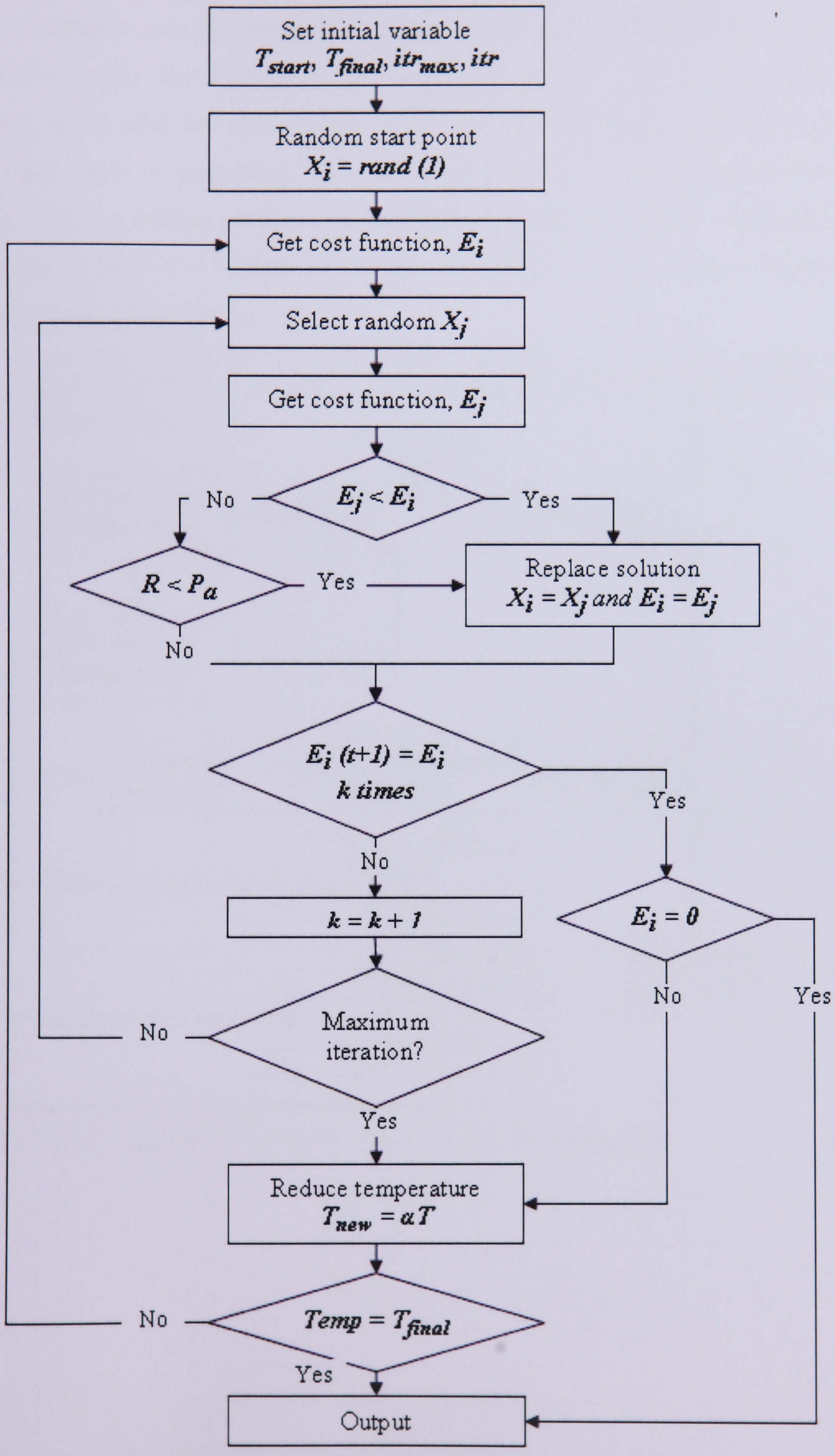


Figure 6.1: Flowchart of the SA Algorithm

6.4 Drive Structure

The proposed tuning scheme with the control system diagram is shown in Fig.6.2. The EKF estimator uses the measured voltage and current provided from the drive. The speed of the motor has been used as a reference speed. The motor model with the application of the EKF has been implemented and discussed in the previous chapter. The SA algorithm starts by generating the initial solution, which is the diagonal elements of Q, G, and R. These solutions are then transferred to the estimator and the objective function is then calculated from the difference between the speeds. The cooling schedule and the acceptance test are then carried out.

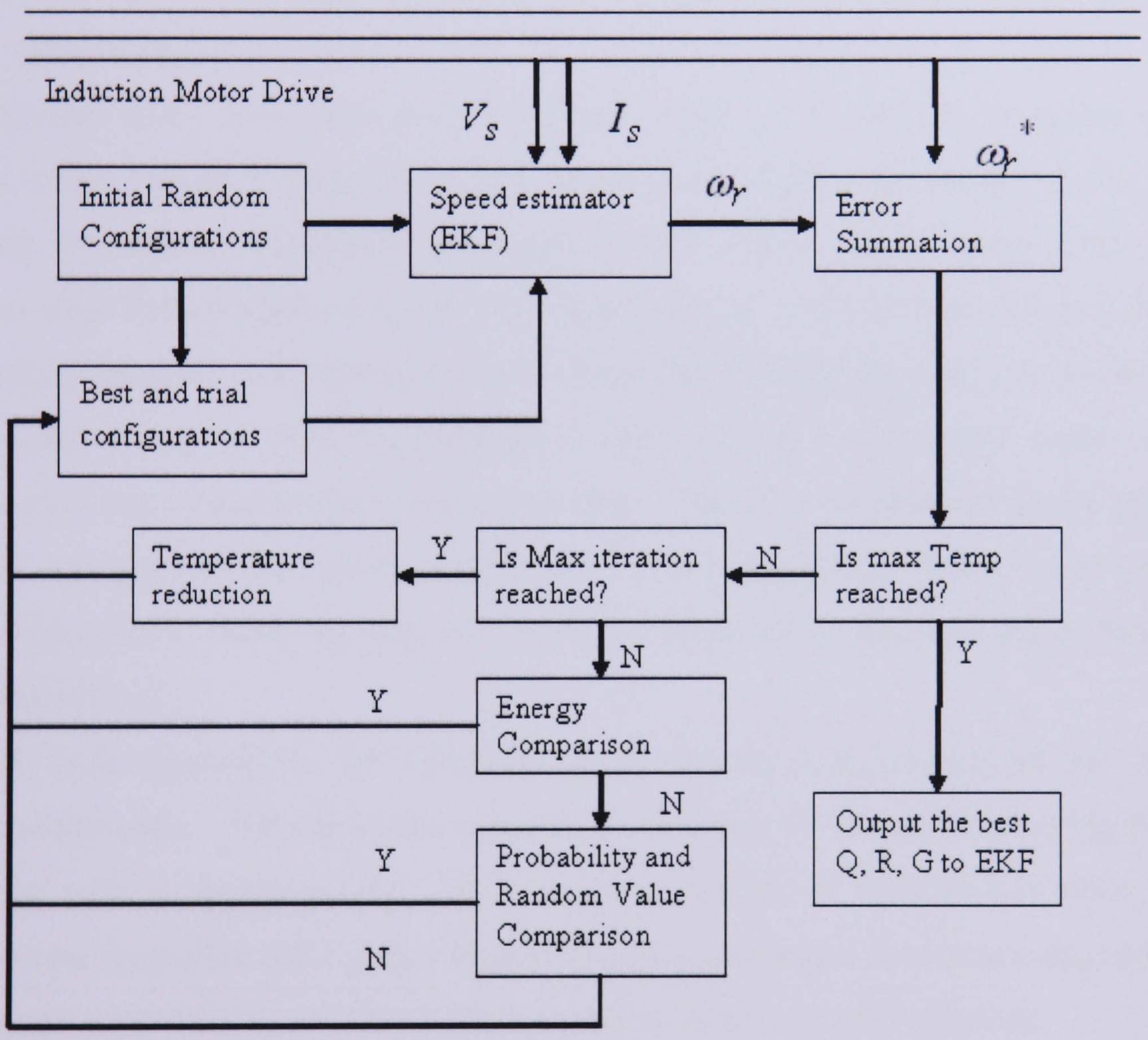


Figure 6.2: Block diagram showing the tuning of the EKF using SA

6.5 Simulation Results using Direct Power Supply

The SA automatically tunes the EKF with the aim of minimising the mean squared error of the speed. The SA has been studied using three approaches for the initial solution: In the first approach, the SA is studied based on a small space. The solution is searched in the 0-0.01 area. In the second approach, it is proposed that the SA uses an approximation for the initial solution. The speed parameter however is given a bigger value than the other parameters. This is done by normalising the Q_{55} parameter in the code before the simulation, to give a 0-1 search space and leaving the other parameters with a [0-0.01] search space. In the final approach, the initial solution is searched in a huge space with a random value of [0-1].

Several trials have been performed and Table 6.1 shows the summary of using different values of initial temperature, final temperature, maximum number of iteration and maximum consecutive number of unchanged codes, initial random value space and the minimum objective function obtained. The best solution obtained using SA is 2.2651. As discussed earlier, the initial temperature used must be high enough, where as at this stage all the proposed solutions will be accepted. Case 2 and 5 show that using too high temperatures only increases the computation time. The final temperature on the other hand is not as important as the initial temperature. When the system freezes, and no further energy drops occur, either the desired solution is achieved or not, and the system can be terminated [87,91].

At each temperature, the simulation must proceed long enough for the system to reach a steady state. The maximum number of iterations is set to avoid having too many iteration at each temperature step. The maximum number of saturation is set so that the inner loop can terminate early if the solution remains unchanged for some consecutive time. Case 2 and 3 show that if a smaller limit is used it results in a worse solution.

The area for the parameters is set to [0-1] or [0-0.01] for all the parameters, or [0-1] for Q_{55} and [0-0.01] for other parameters, thus allowing the rotor speed to have a greater influence. Case 1 and 7 show that the larger the space area that needs to be explored, the longer the computation time that is needed. Other cases show that using an approximation leads to better results. Also, cases 2 and 5 show that the estimator can take a variety of initial points of Q, G, R, and initial temperature, and still converge to the same solution 2.5163.

Case No.	Temp start	Temp final	Temp Coeff.	Iteration max	Saturation Max	Q,G,R	Q_{55}	En
1	15	7	0.9	15	15	0-0.01	0-0.01	13.64
	20	7	0.9	15	15	0-0.01	0-0.01	4.14
	40	7	0.9	15	15	0-0.01	0-0.01	52.17
	90	7	0.9	15	15	0-0.01	0-0.01	106.14
2	80	7	0.9	15	10	0-0.01	0-1	2.27
	80	7	0.9	15	10	0-0.01	0-1	2.52
	80	7	0.9	15	10	0-0.01	0-1	2.27
	80	7	0.9	15	10	0-0.01	0-1	2.52
3	80	7	0.9	15	5	0-0.01	0-1	5.84
	80	7	0.9	15	5	0-0.01	0-1	3.08
	80	7	0.9	15	5	0-0.01	0-1	6.01
4	80	7	0.9	30	15	0-0.01	0-1	4.17
	80	7	0.9	30	15	0-0.01	0-1	5.66
5	100	7	0.9	15	10	0-0.01	0-1	2.52
	100	7	0.9	15	10	0-0.01	0-1	2.52
	100	7	0.9	15	10	0-0.01	0-1	4.72
	100	7	0.9	15	10	0-0.01	0-1	5.23
6	200	100	0.9	15	10	0-0.01	0-1	2.98
	200	100	0.9	15	10	0-0.01	0-1	12.24
7	100	7	0.9	15	10	0-1	0-1	102
	200	7	0.9	15	10	0-1	0-1	70

Table 6.1: Performance of the EKF using SA algorithm

The details of the best solution will now be discussed. To get the best solution, the SA was randomly started with these initial solutions:

$$Q = diag[0.0092 \quad 0.0074 \quad 0.0018 \quad 0.0041 \quad 0.94]$$

$$G = diag[0.0092 \quad 0.0041 \quad 0.0089 \quad 0.0006 \quad 0.0035]$$

$$R = diag[0.0081 \quad 0.0001]$$

The solutions which converge to the minimum objective function of 2.2651 are as follow:

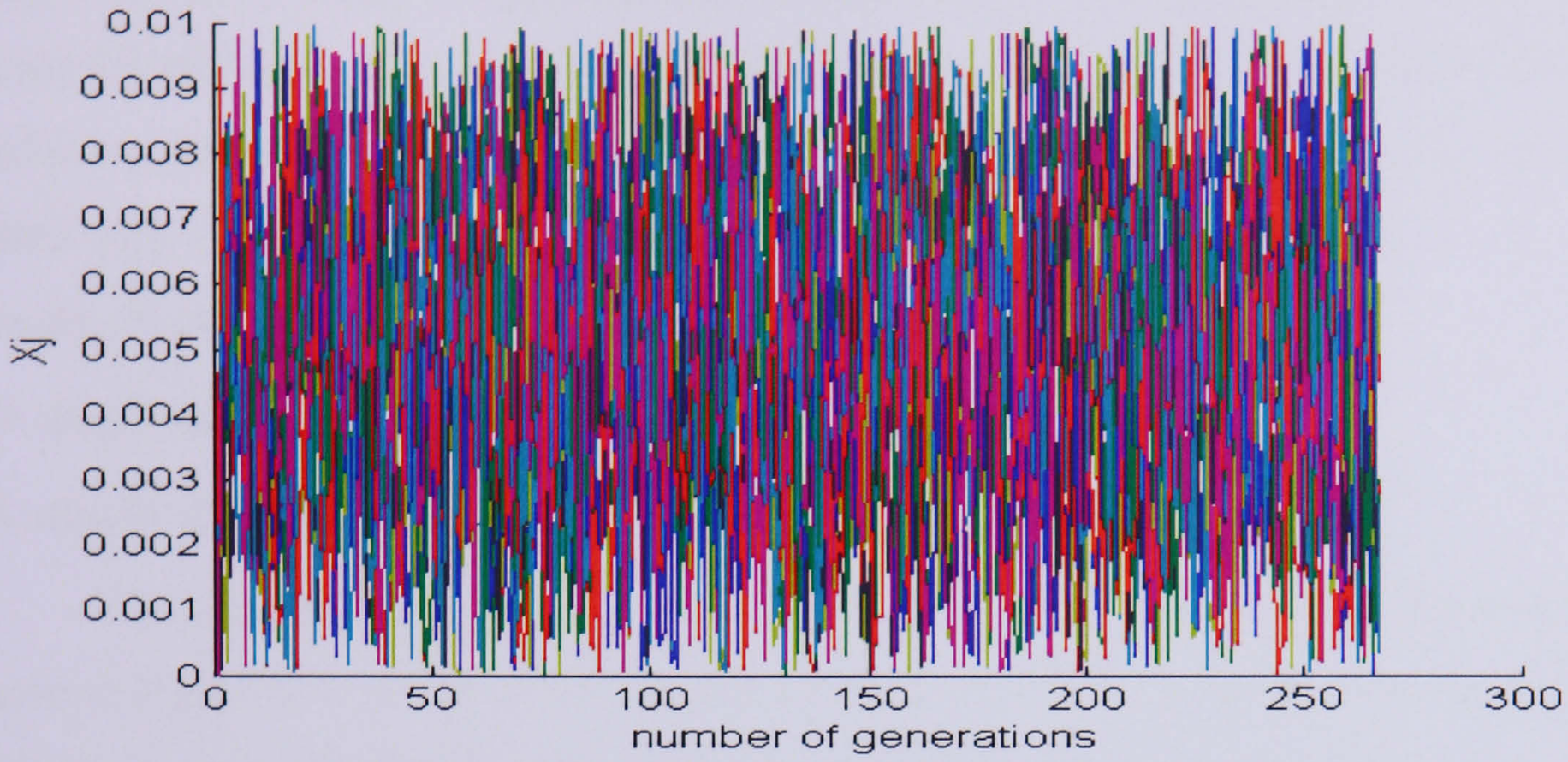
$$Q = diag[0.0097 \quad 0.0013 \quad 0.0025 \quad 0.0091 \quad 0.68]$$

$$G = diag[0.0062 \quad 0.0051 \quad 0.0000 \quad 0.0023 \quad 0.0098]$$

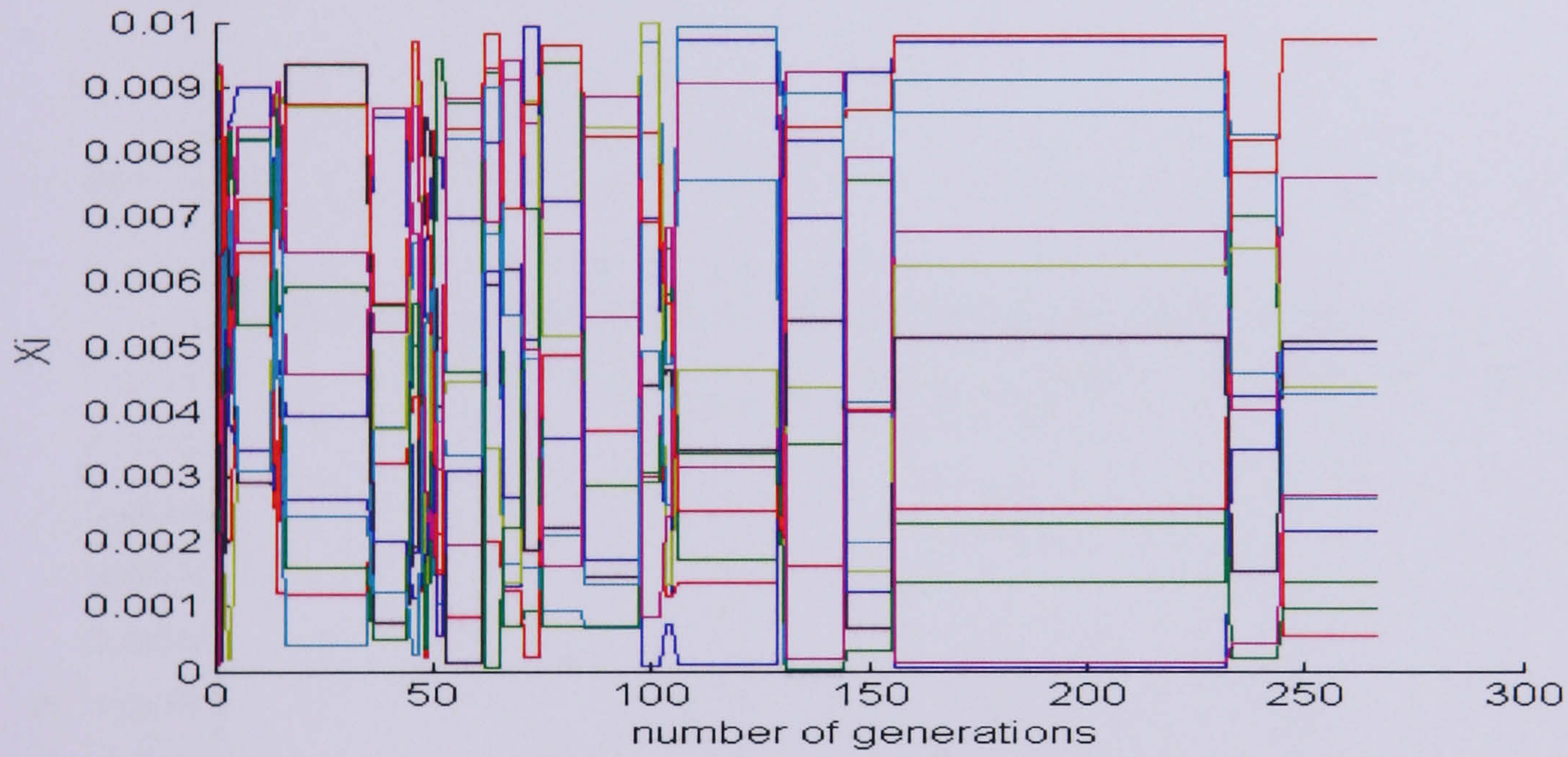
$$R = diag[0.0086 \quad 0.0001]$$

The initial temperature has been set to 80; the simulation time is set to 0.5 sec. The minimum temperature has been set to 7 and gives a total of t_s=336 iterations, with the initial setup of t_s=2 to occupy the matrix complication in Matlab programming. For the scenario of an unchanged solution in 10 consecutive times, the iteration is terminated early; the temperature is reduced and proceeds with another iteration. The SA finally made t_s=267

iterations in total. Figure 6.3(a) and (b) show the overall solutions randomly selected in the search space and the accepted solutions respectively:



(a)



(b)

Figure 6.3: (a) The search space of twelve configurations of Q, G, and R (b) The accepted configurations of Q, G, and R

The investigated solutions obtained from randomly selected solutions are shown in Figure 6.4(a). The accepted possible solutions are shown in Figure 6.4(b), while the best accepted solution is shown in an enlarged scale in Figure 6.4(c). The best solution has been generated at t_s=156-232. As mentioned before an E_j, which is larger than the E_i, will be accepted provided that Pa, is larger than the random value R [0, 1]. Figures 6.5(a) has shown that at high temperature, the chances of the solutions being accepted is more than during the cool temperature. Figure 6.5(b) shows that R gives a higher value than Pa during the cooling time.

The speed estimate obtained using the proposed method is shown in Figure 6.6. The reference, initial and final speed are plotted on the same graph. The initial solution is created randomly using an approximation method and gives a good result in a steady state, but not the transient. By using SA, the final result shows that both transient and steady state conditions follow the reference speed very closely. In another simulation, the SA was started with a different initial solution as follows:

$$Q = diag[0.0085 \quad 0.0053 \quad 0.0020 \quad 0.0067 \quad 0.84]$$

$$G = diag[0.0002 \quad 0.0068 \quad 0.0038 \quad 0.0083 \quad 0.0050]$$

$$R = diag[0.0071 \quad 0.0043]$$

The same solution as in the previous simulation is obtained and the final result is shown in Figure 6.7. In this case, the best solution might not be always the same due to the randomness value that is used to determine the parameter in the search space.

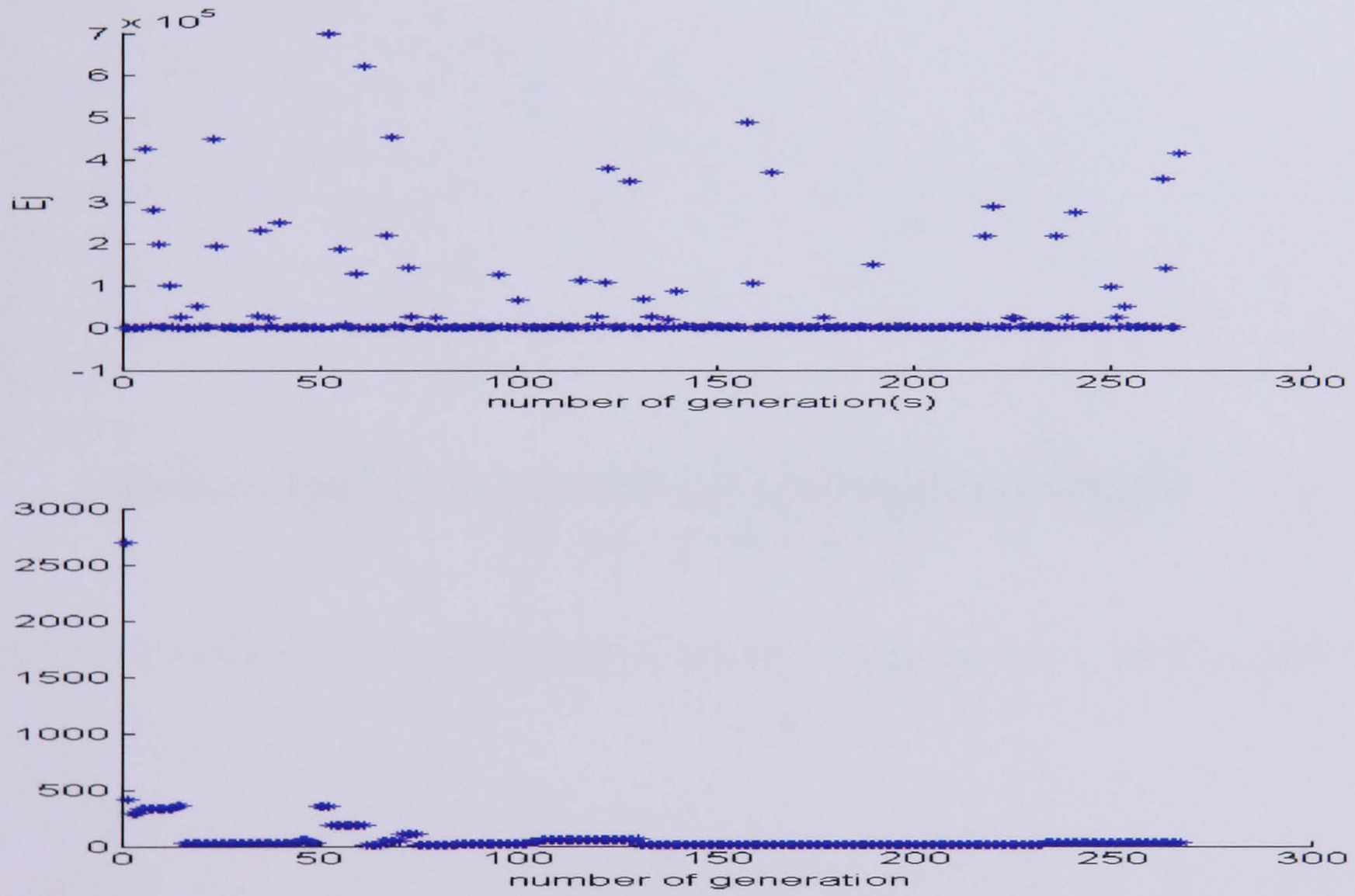
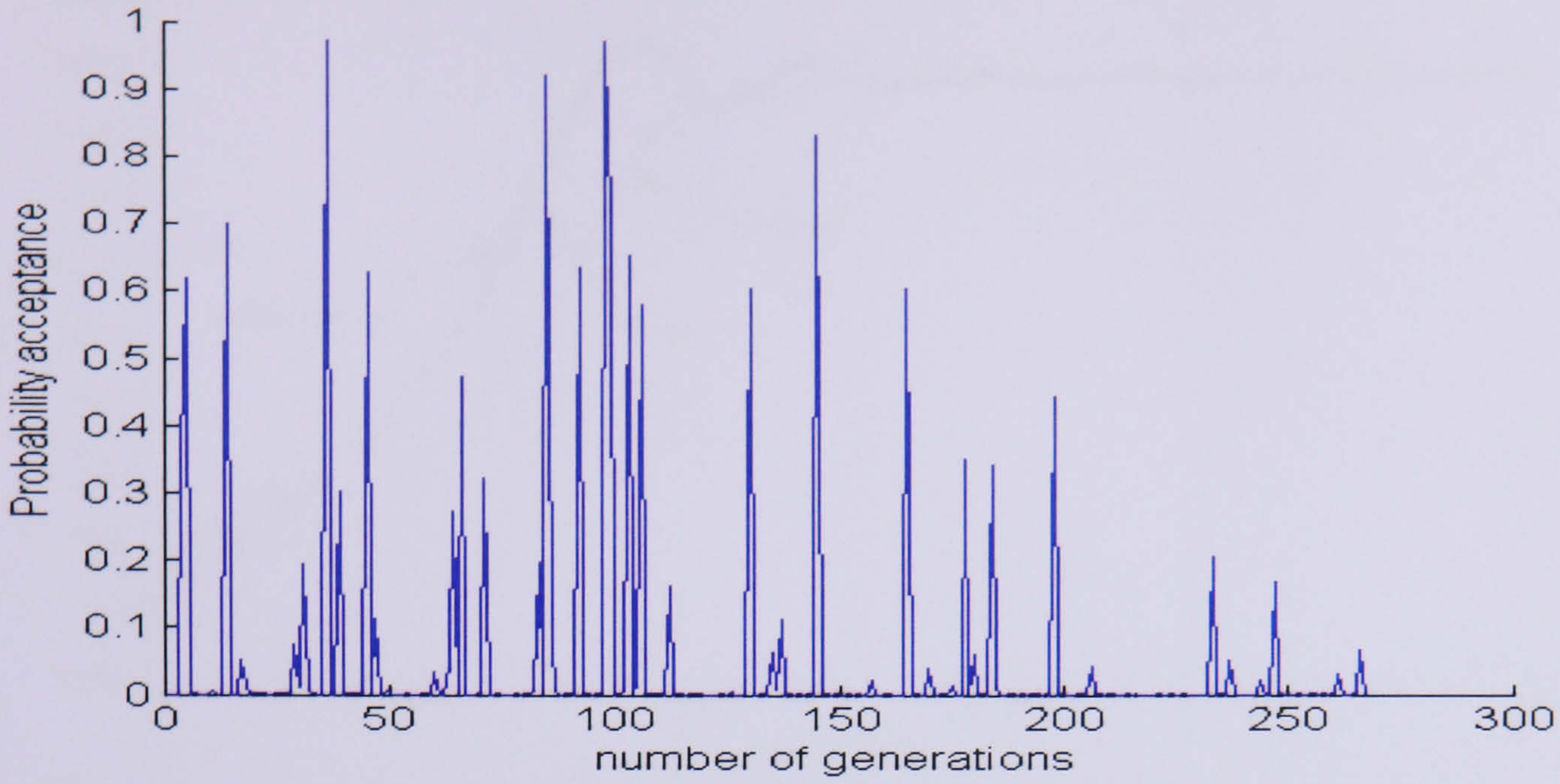
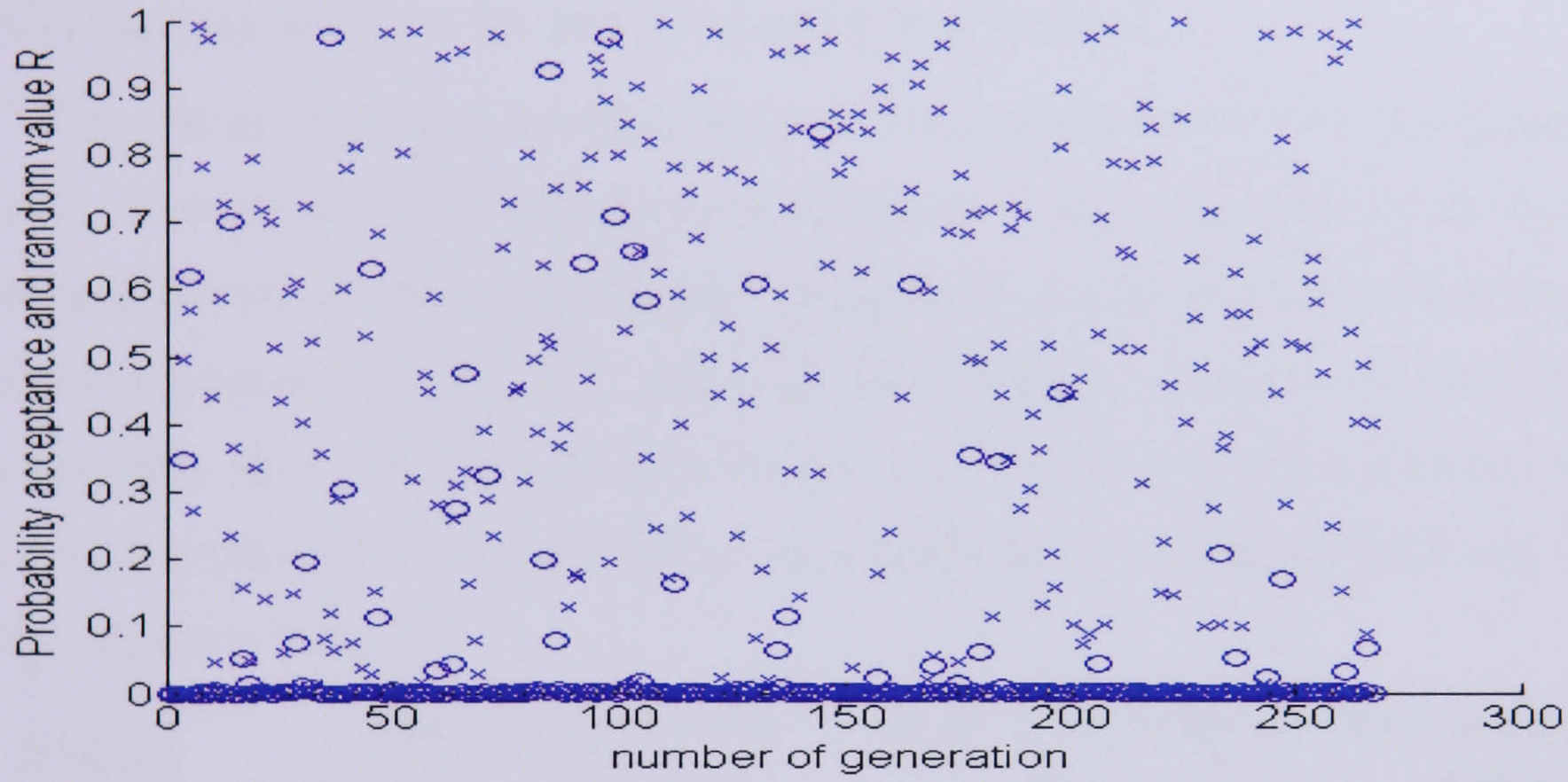


Figure 6.4: (a) The investigated solutions (b) The accepted solution (c) The accepted solutions in an enlarged scale



(a)



(b)

Figure 6.5: (a) Probability acceptances (b) Probability acceptances ‘o’ and random value R ‘x’

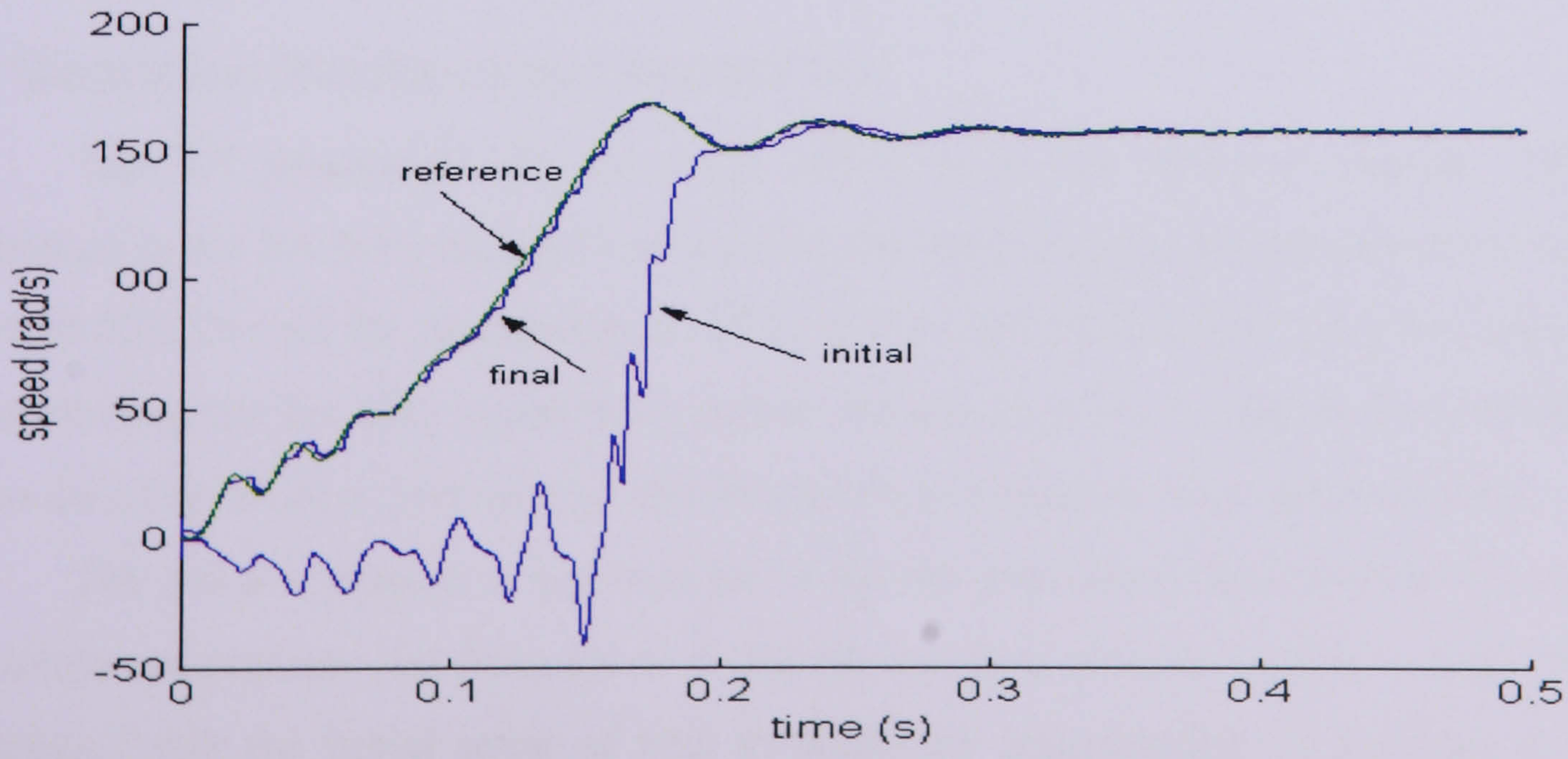


Figure 6.6: Reference and estimated speed

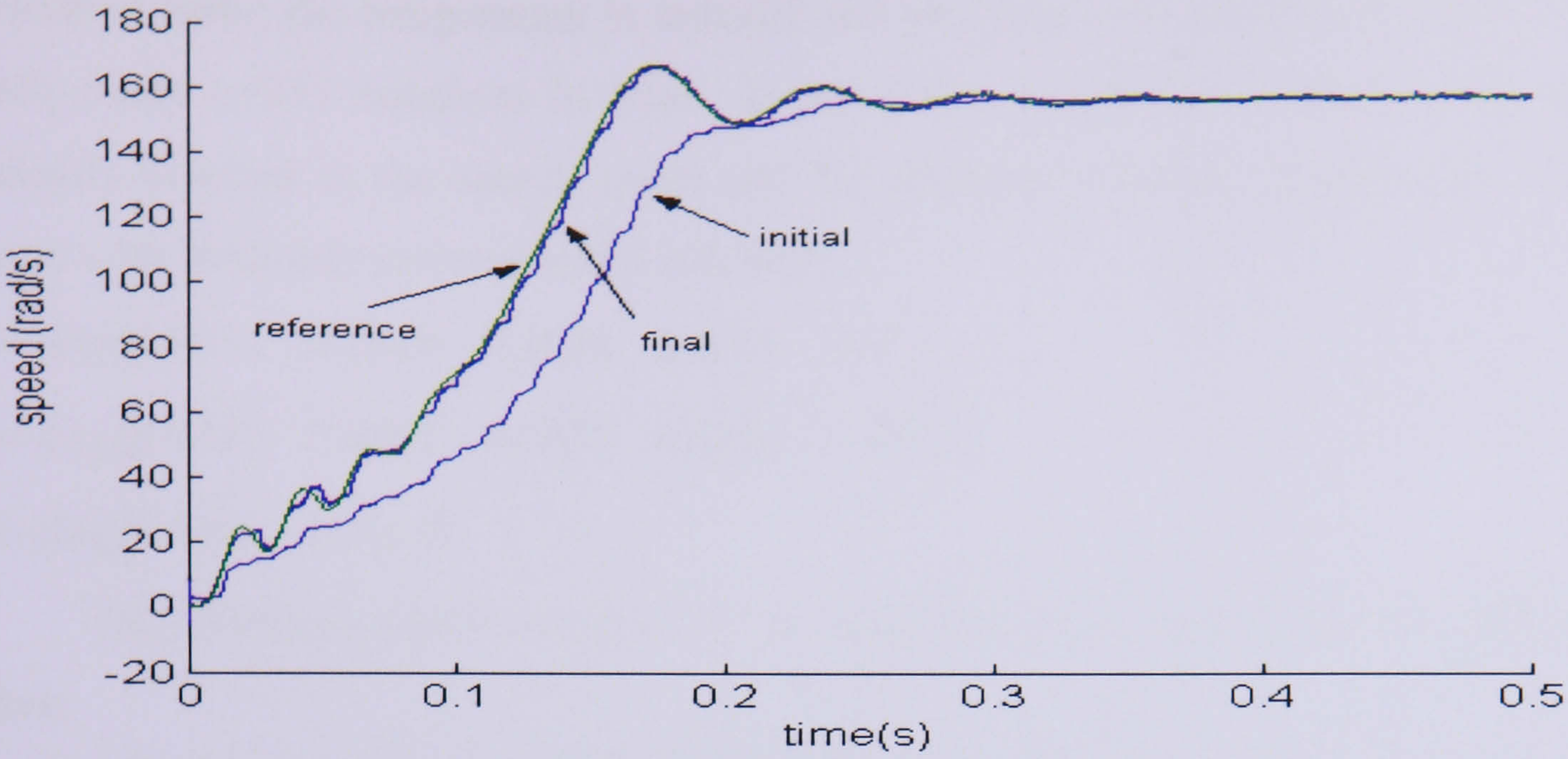


Figure 6.7: Reference and estimated speed using different initial solution

6.5.1 Comparison between SA and Trial and Error Method

This section compares the EKF estimator tuned with SA with the one tuned by Trial and Error. Comparing the percentage error difference, based separately on the steady state performance and the transient performance using the best solution from each method, gives the results summarised in Table 6.2. Although the difference between each method is small, the comparison shows the improved performance of the proposed SA method, both in a steady and transient state. Hence SA has the capabilities of tuning the EKF and replacing the Trial and Error method.

Method	Mean Squared Error En	Steady State Error (%)	Transient and Steady State (%)
Trial and Error	4.3994	0.114	0.52
SA	2.2651	0.12	0.36

Table 6.2. Comparison table using Trial and Error and SA

6.6 Simulation Results using Constant V/f

The V/f parameters are the same values as in the previous chapter. With no difference in the SA from the previous section, the searh area for the parameters is set to [0-1] or [0-0.01] for all the parameters, or [0-1] for Q_{55} and [0-0.01] for the other parameters, remembering that the rotor speed has a greater influence on Q_{55}. It shows that the larger the space area that needs to be explored, the longer the computation time that is needed.

The initial temperature has been set to 80; the simulation time is set to 2.5 sec. The minimum temperature has been set to 7 and the solution initially require a total of t_s=336 iterations, with the initial setup of t_s=2 to avoid the complication of matrices in Matlab programming. With an unchanged solution for 10 consecutive times, the iteration is

terminated early; the temperature is reduced and proceeds with another iteration. The SA finally made t_s=255 iterations in total. Figure 6.8(a) and (b) show the overall solutions randomly selected in the search space and the accepted solutions respectively. The SA started with randomly selected initial solution:

$$Q = diag[0.0014 \quad 0.0020 \quad 0.0020 \quad 0.0060 \quad 0.27]$$

$$G = diag[0.0020 \quad 0.0002 \quad 0.0075 \quad 0.0045 \quad 0.0093]$$

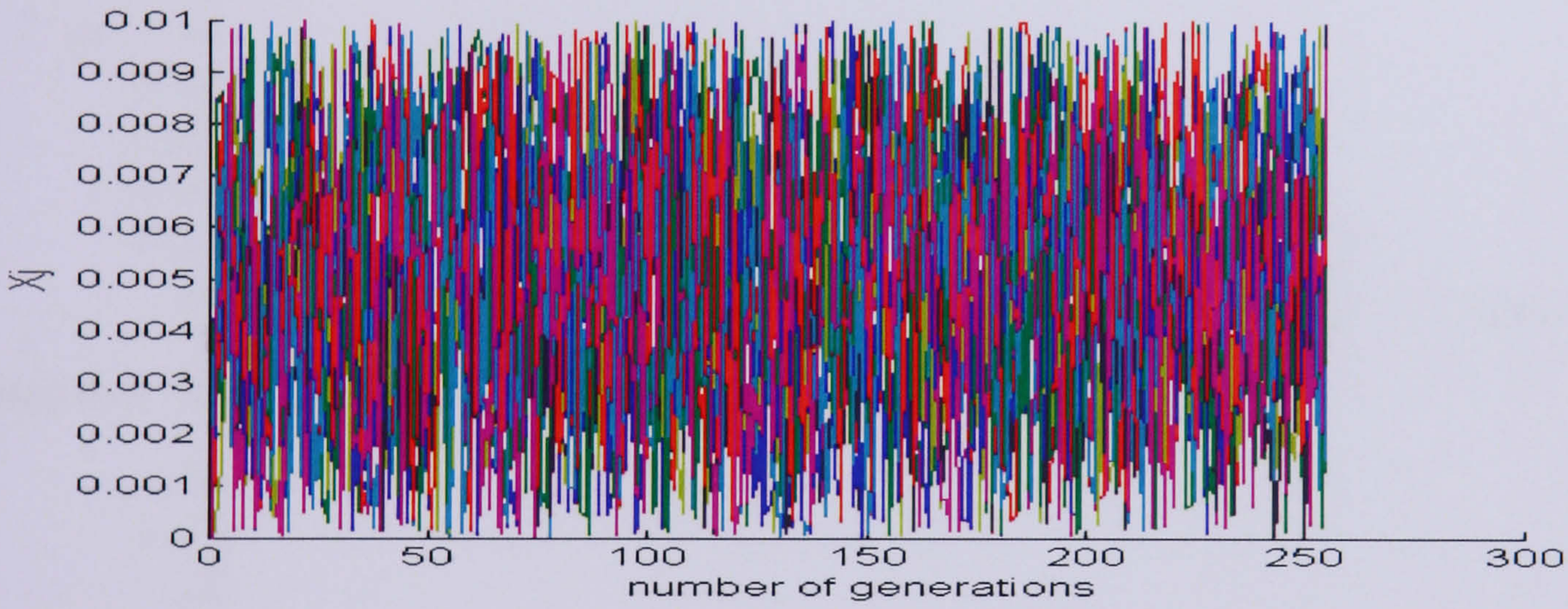
$$R = diag[0.0047 \quad 0.0042]$$

The solutions, which converge to the minimum objective function of 0.5707, are as follow:

$$Q = diag[0.0096 \quad 0.0081 \quad 0.0026 \quad 0.0021 \quad 0.93]$$

$$G = diag[0.0039 \quad 0.0006 \quad 0.0004 \quad 0.0001 \quad 0.0082]$$

$$R = diag[0.0092 \quad 0.0074]$$



(a)

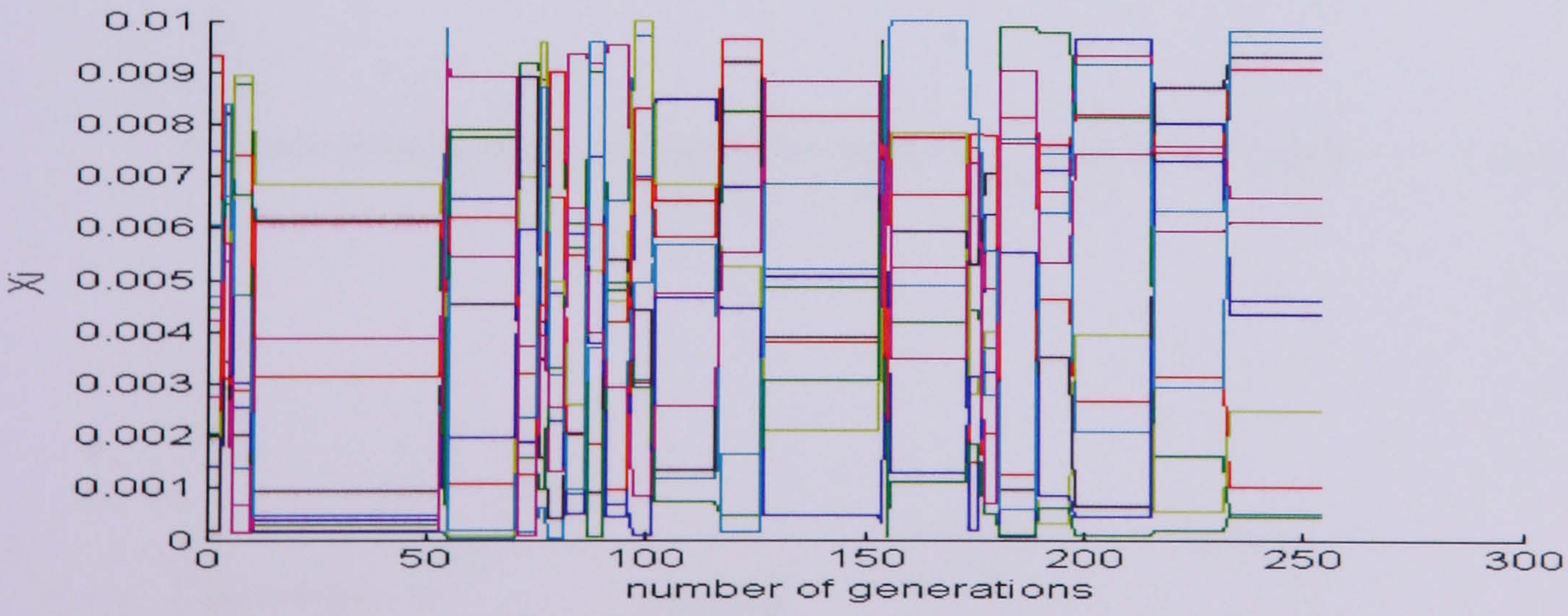


Figure 6.8: (a) Search space of twelve configurations of Q, G, and R (b) Accepted configurations of Q, G, and R

The investigated solutions obtained from randomly selected solutions are shown in Figure 6.9. The accepted possible solutions are shown in Figure 6.10(a), while the best

accepted solution is shown in an enlarged scale in Figure 6.10(b). The best solution has been generated at ts=198-215.

As mentioned, E_j, which is larger than E_i, will be accepted on condition that Pa, must be larger than the random value R [0, 1]. In Figure 6.11(a) and (b), at higher temperature, the chances of the solutions being accepted are more than at cool temperatures. R gives a higher value than Pa during the cooling time. The higher the initial temperature used, the bigger is the difference between using the probability acceptance or not.

With a low mean squared error of 0.57, a good estimate for the rotor speed in the nominal condition is to be expected. The best result is shown in Figure 6.12 and confirms this.

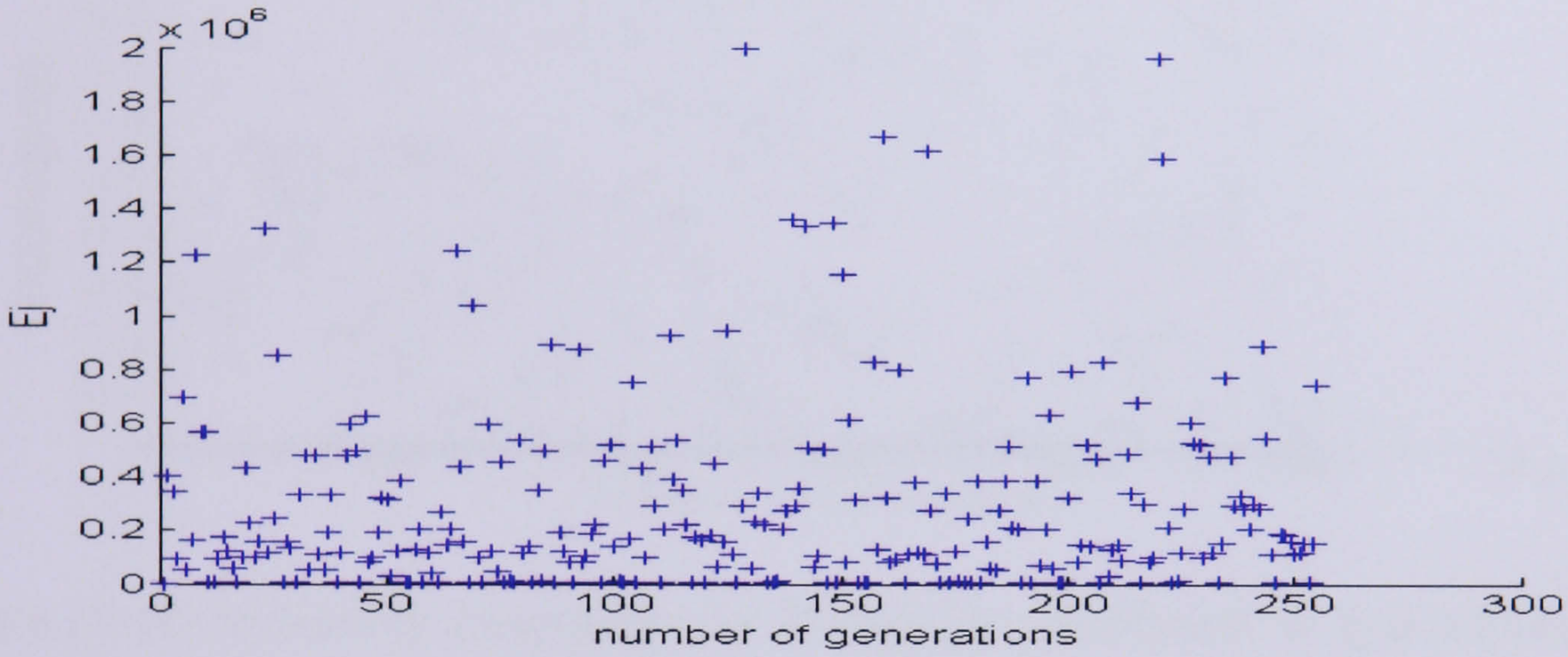
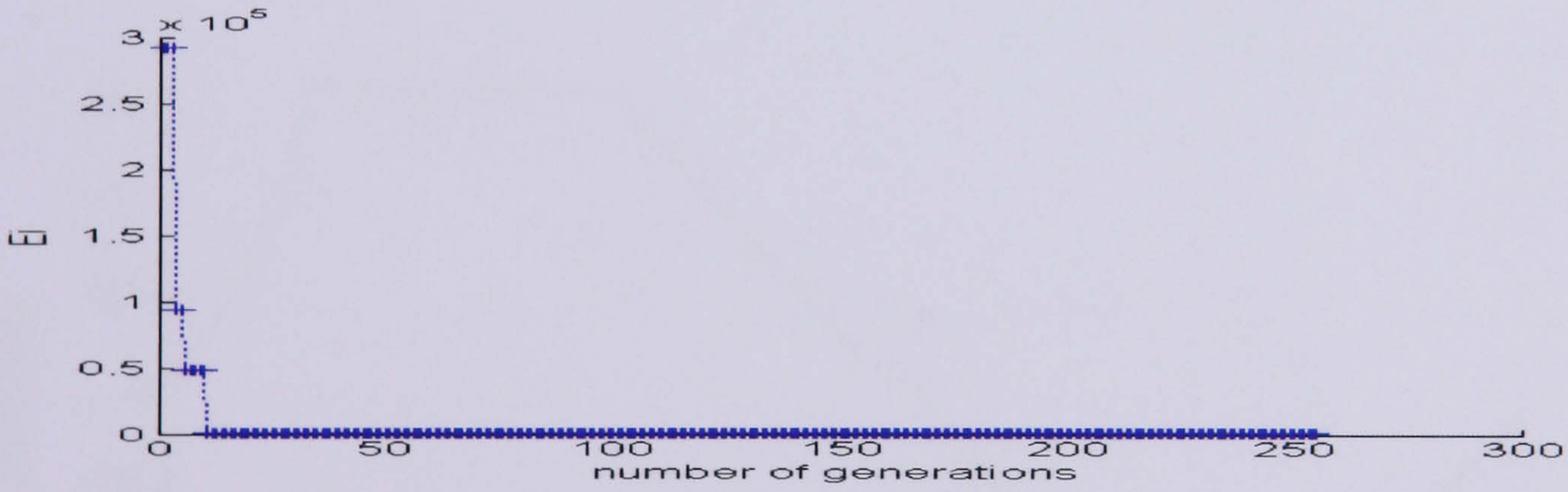
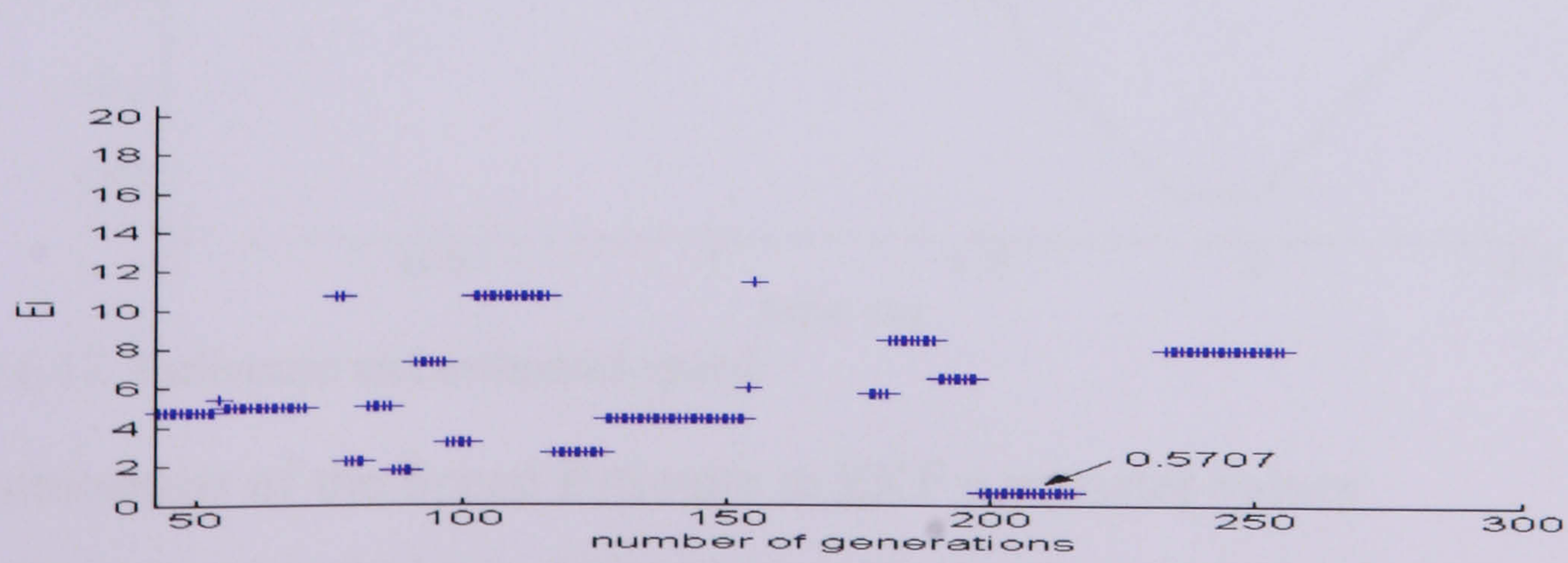


Figure 6.9: The investigated solutions

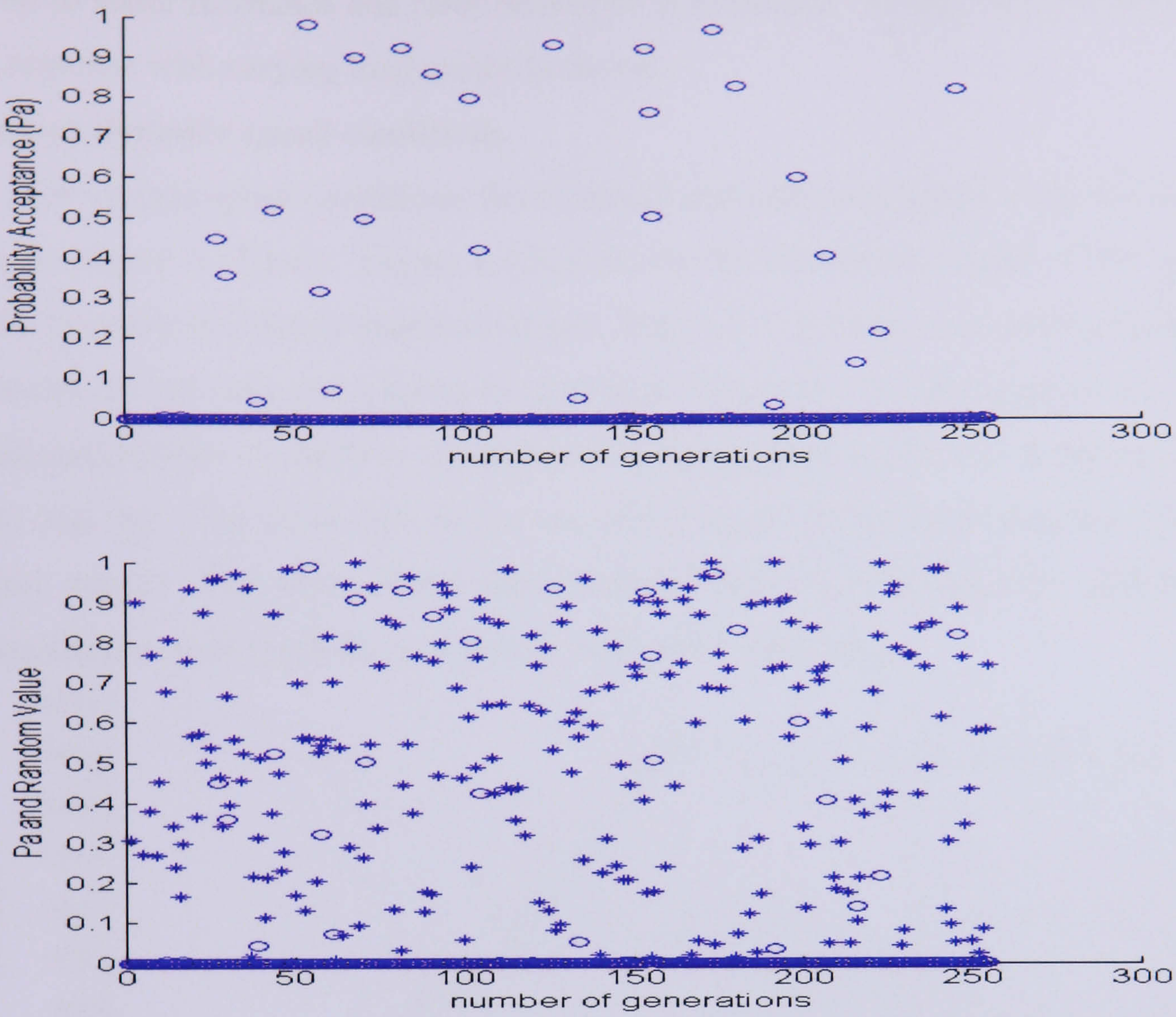


(a)



(b)

Figure 6.10: (a) The accepted solution (b) The accepted solutions on an enlarged scale



(a)

(b)

Figure 6.11: (a) Probability acceptances (b): Probability acceptances ‘o’ and random value R ‘x’

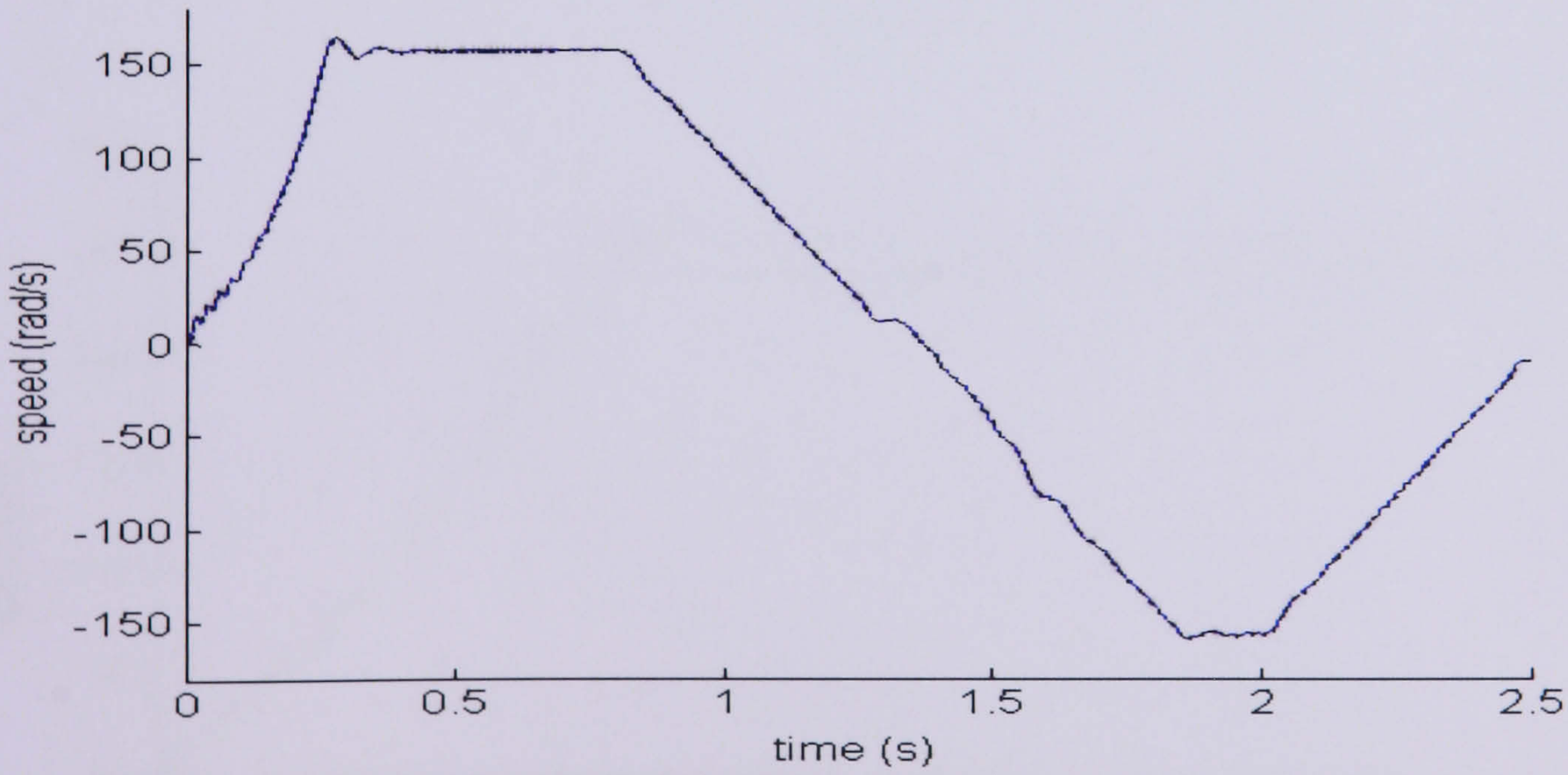


Figure 6.12: Reference and estimated speed

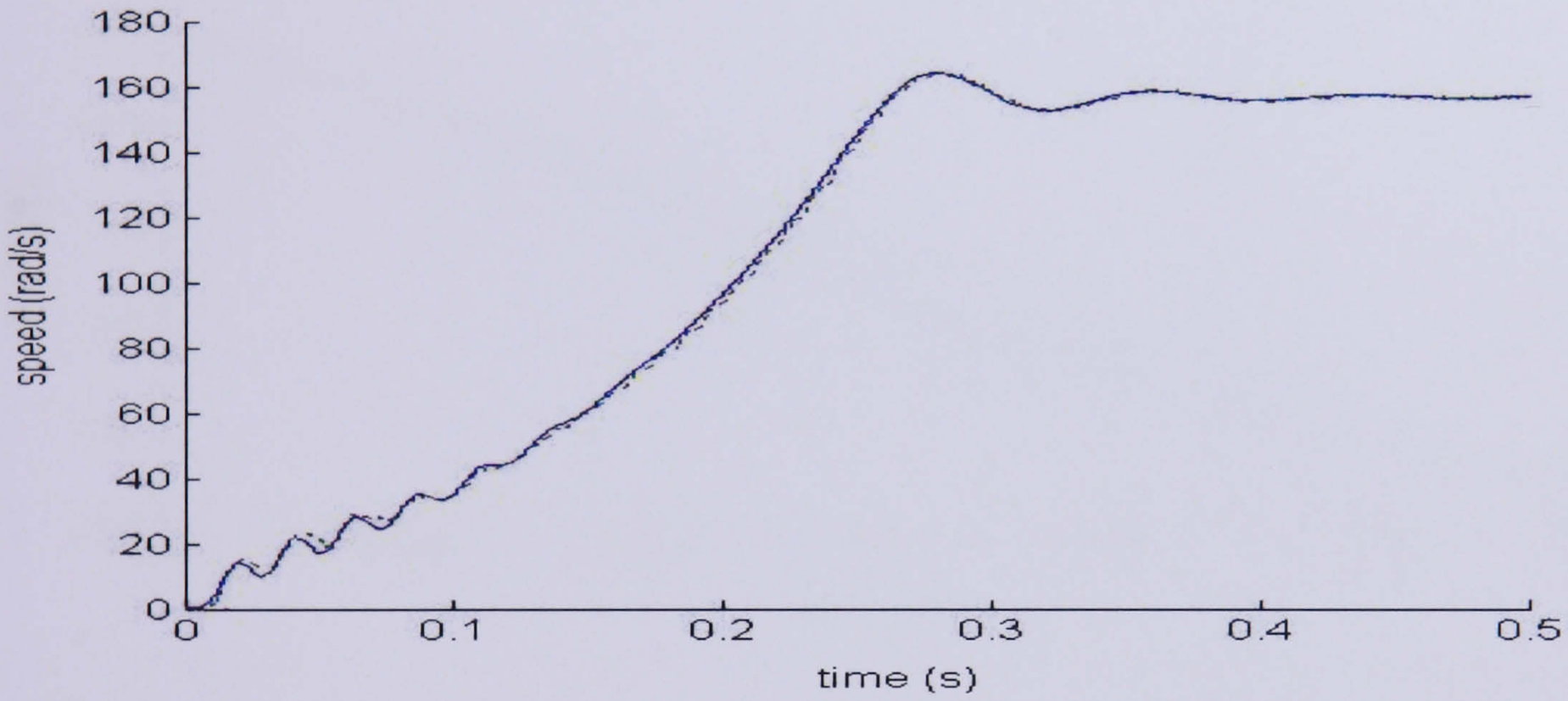
6.7 Robustness of the Speed Estimate to EKF parameter values

In this section, the best solution discussed in the previous section is used. The speed response for different operating conditions is simulated. Then the sensitivity of the speed

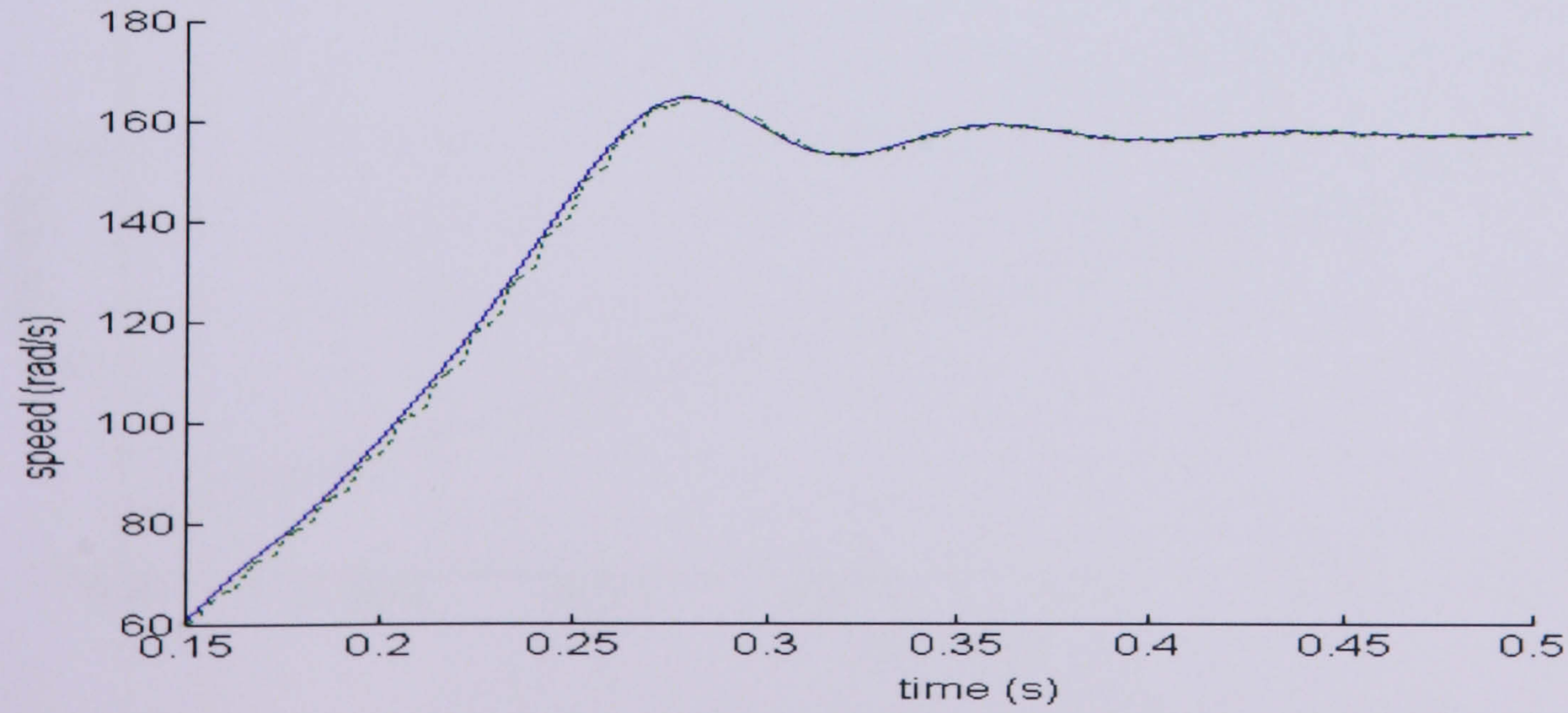
response to stator resistance and rotor resistance is discussed. Finally, the robustness of the speed response with varying load torque is shown.

A. Various dynamic speed conditions

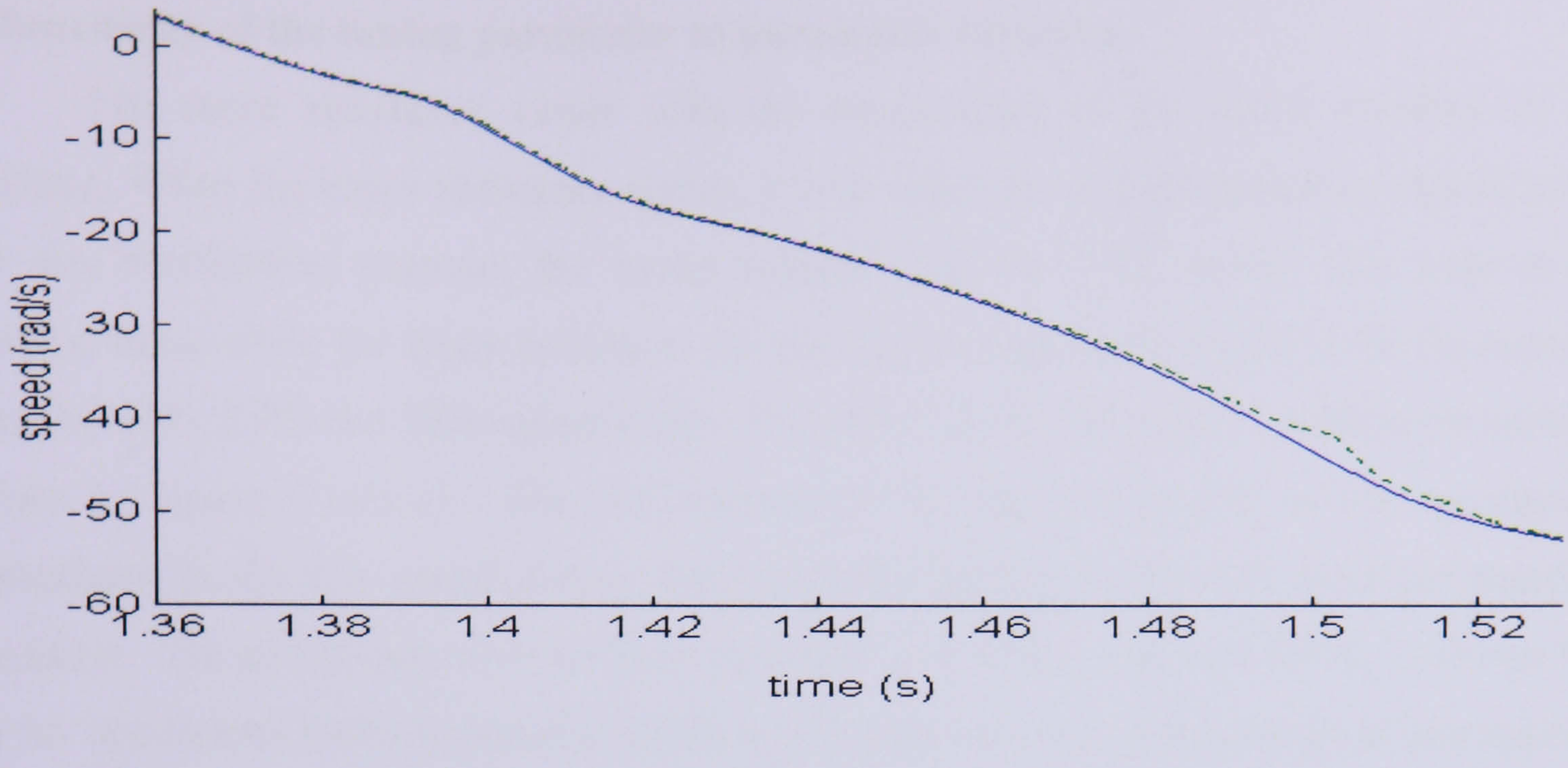
For various speed conditions, the reference and estimated speed using SA are shown clearly in Figure 6.13(a-e). Figure 6.13(a) shows the simulation results of the optimised EKF for transient and steady state conditions. Figure 6.13(b) and (c) show the results for an acceleration to 160 rad/s and a speed reversal from 6 rad/s to -53 rad/s respectively. Finally, an acceleration from -30 rad/s to -52 rad/s and a deceleration to -58rad/s is shown in Figure 6.13(d) and (e). The solid line shows the actual speed while the dotted line shows the reference speed. The result shows that reference speed is followed very closely by the estimated speed in all operating conditions with only 0.14% error.



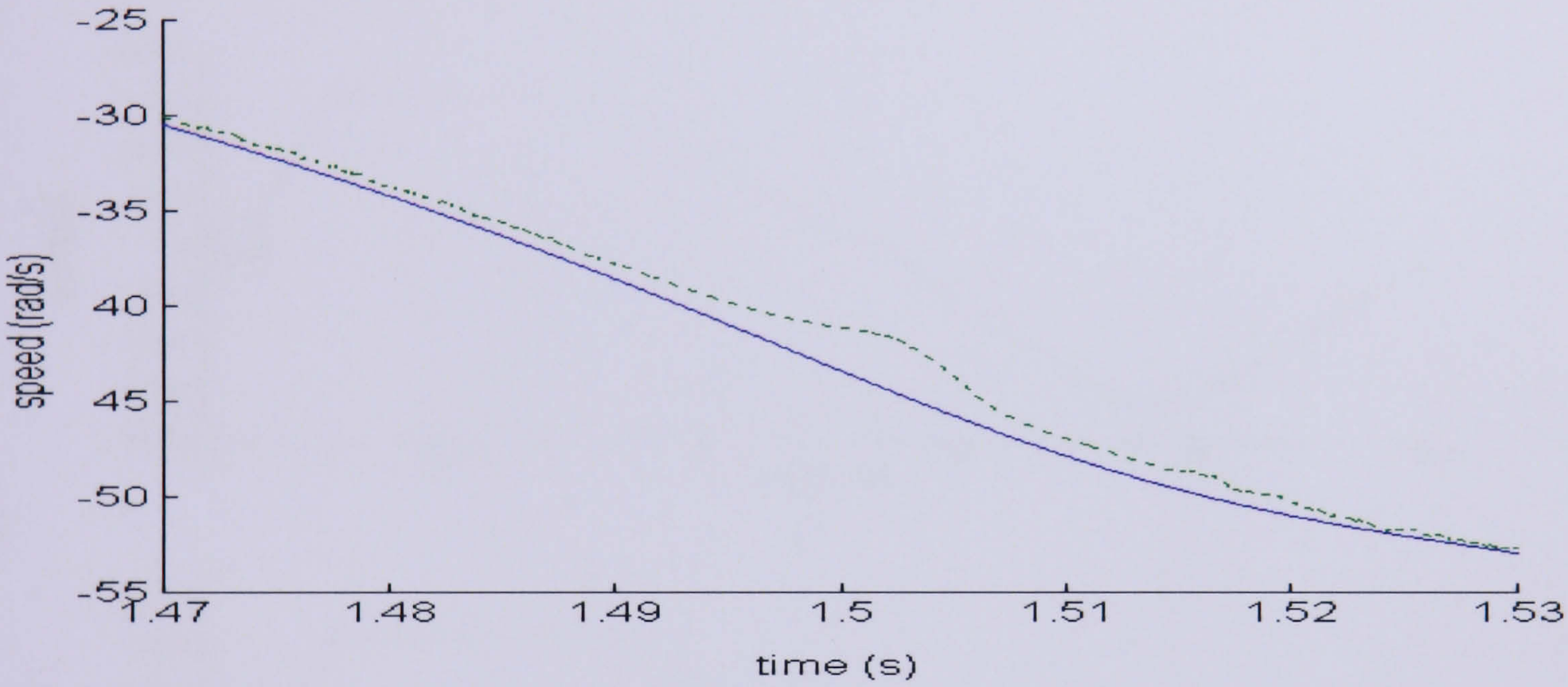
(a)



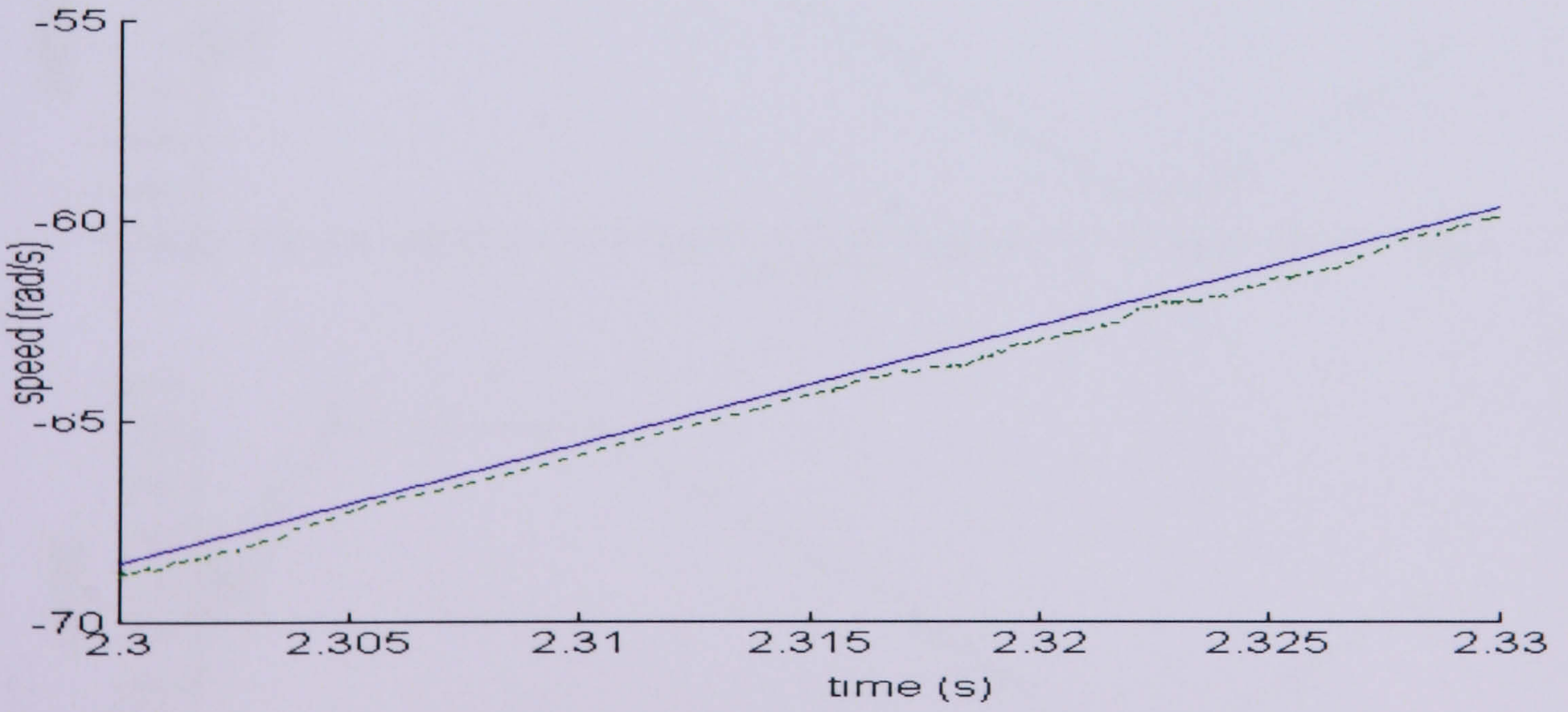
(b)



(c)



(d)

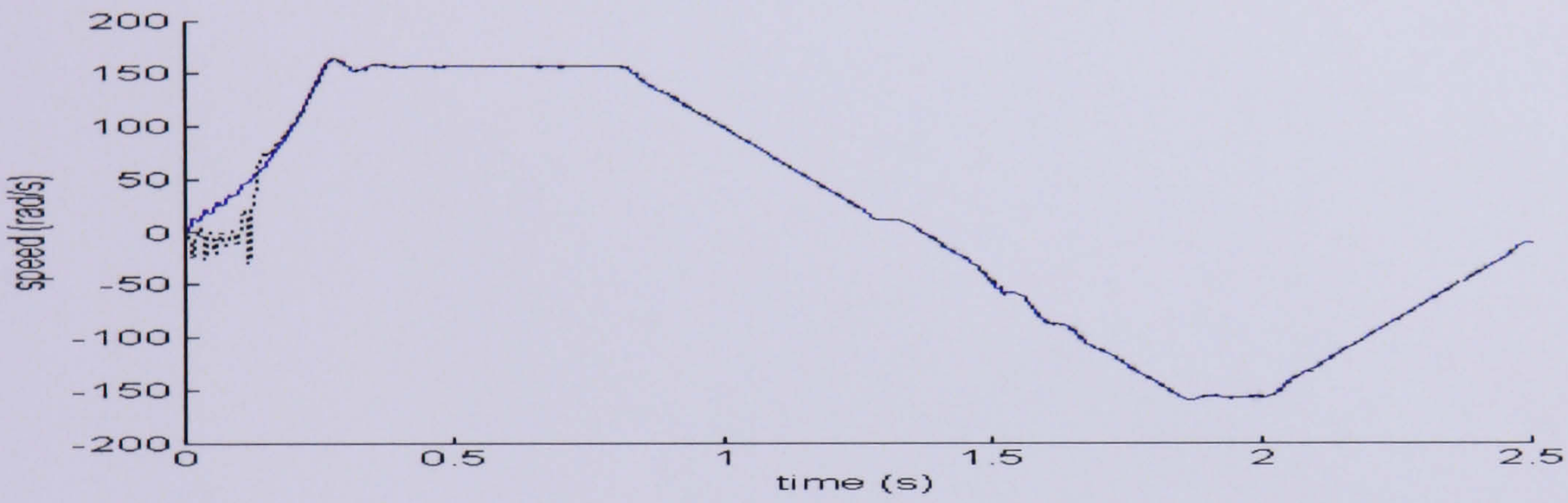


(e)

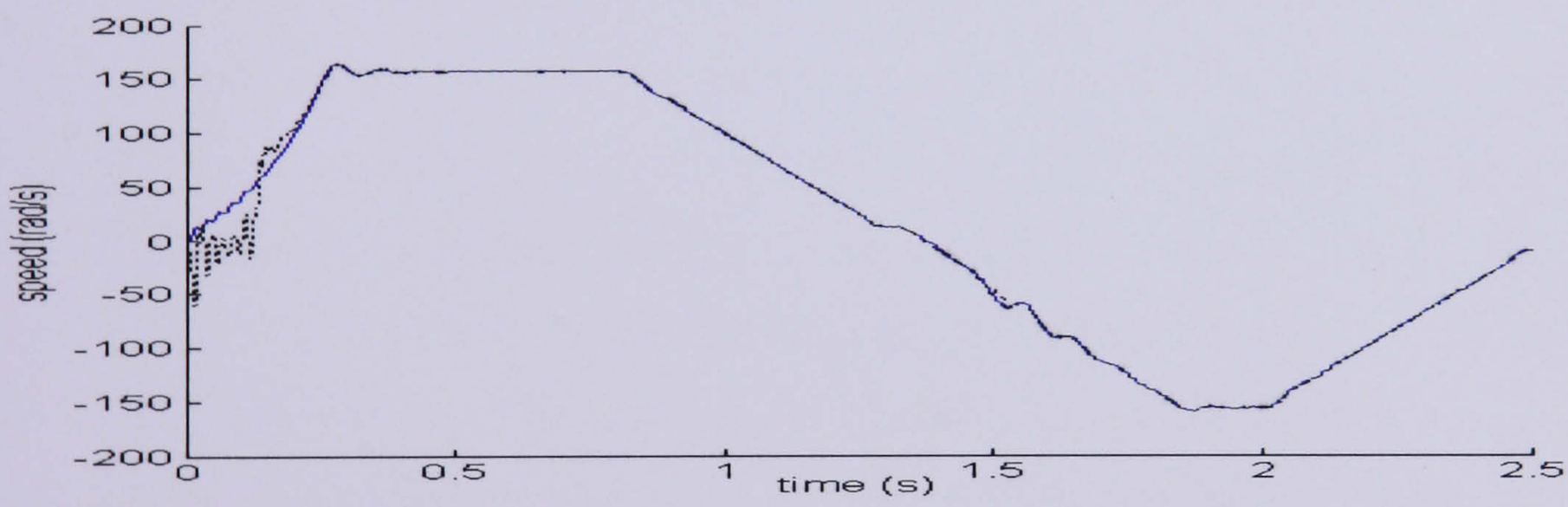
Figure 6.13: (a) Motor speed during transient and steady state (b) Acceleration to 160 rad/s (c) Speed Reversal from 6 rad/s to -53 rad/s (d) Acceleration from -30 rad/s to -52 rad/s (e) Deceleration to -58rad/s

B. Sensitivity of the tuning parameter to parameter variation

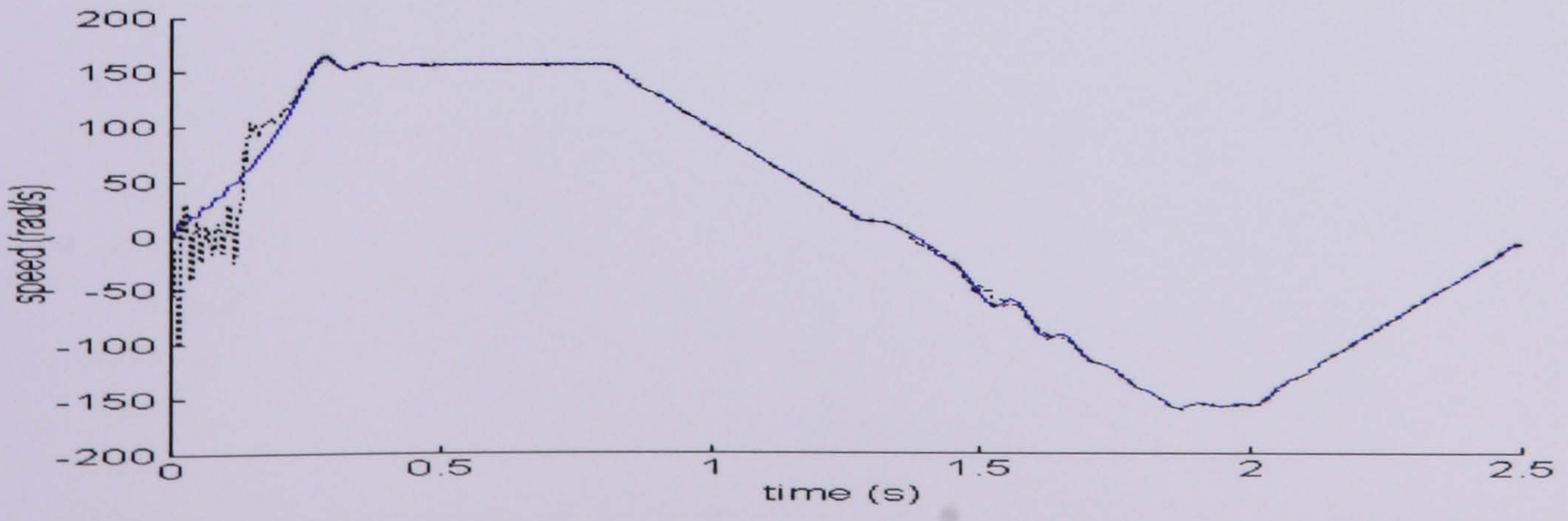
The stator resistance varies with the temperature of the stator winding of the machine. When the stator resistance varies, it will affect the speed estimation significantly. For this verification purpose, the motor resistance of the EKF model was kept at the nominal value while the stator resistance for the motor model is increased from the nominal value by 10%, 25% and 50% respectively. The effect of this increment on the rotor speed is shown in Figure 6.14(a-c). The performance of the speed estimator is not satisfactory particularly in the low speed during start-up condition but works well after the transient condition. The percentage error for each variation is 0.16%, 0.06% and 0.4%. It shows that the SA is robust to stator resistance variations but does limit the performance at low speed.



(a)



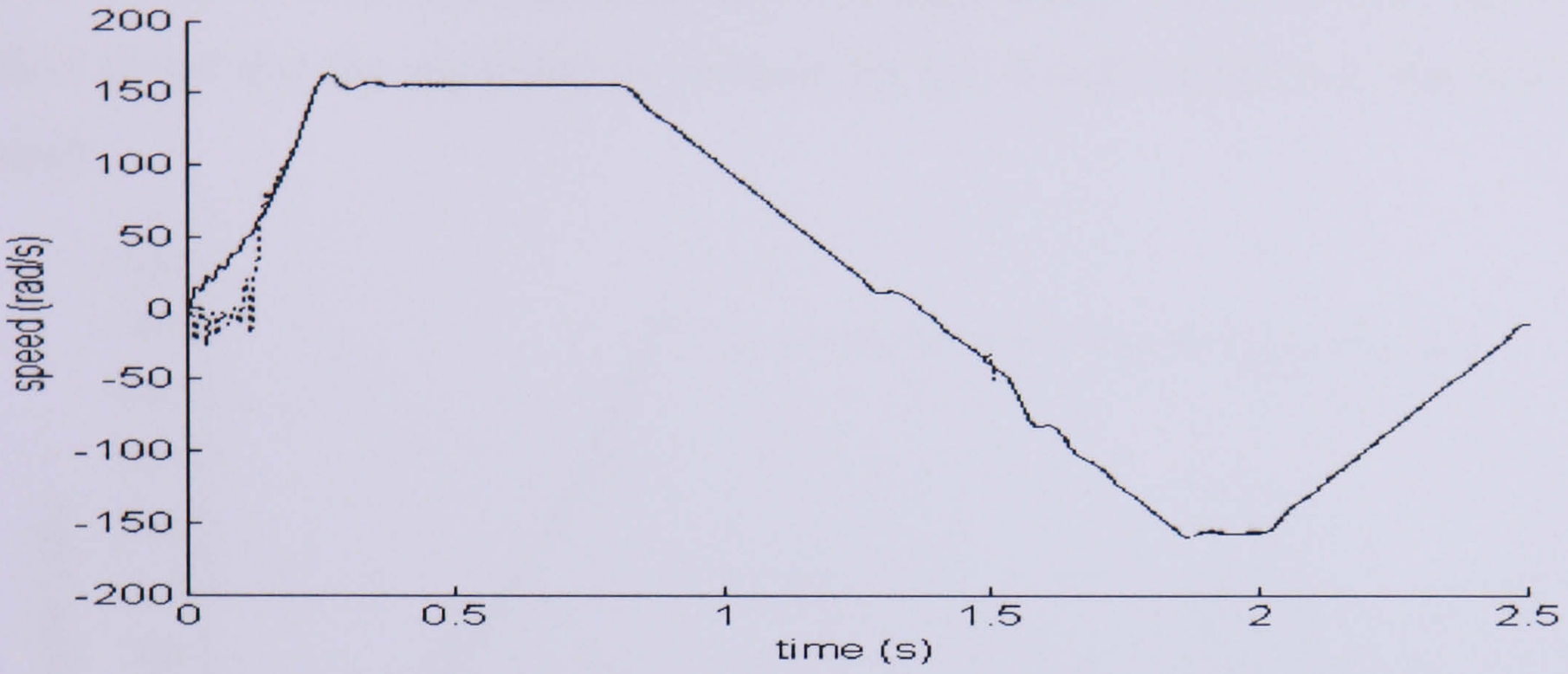
(b)



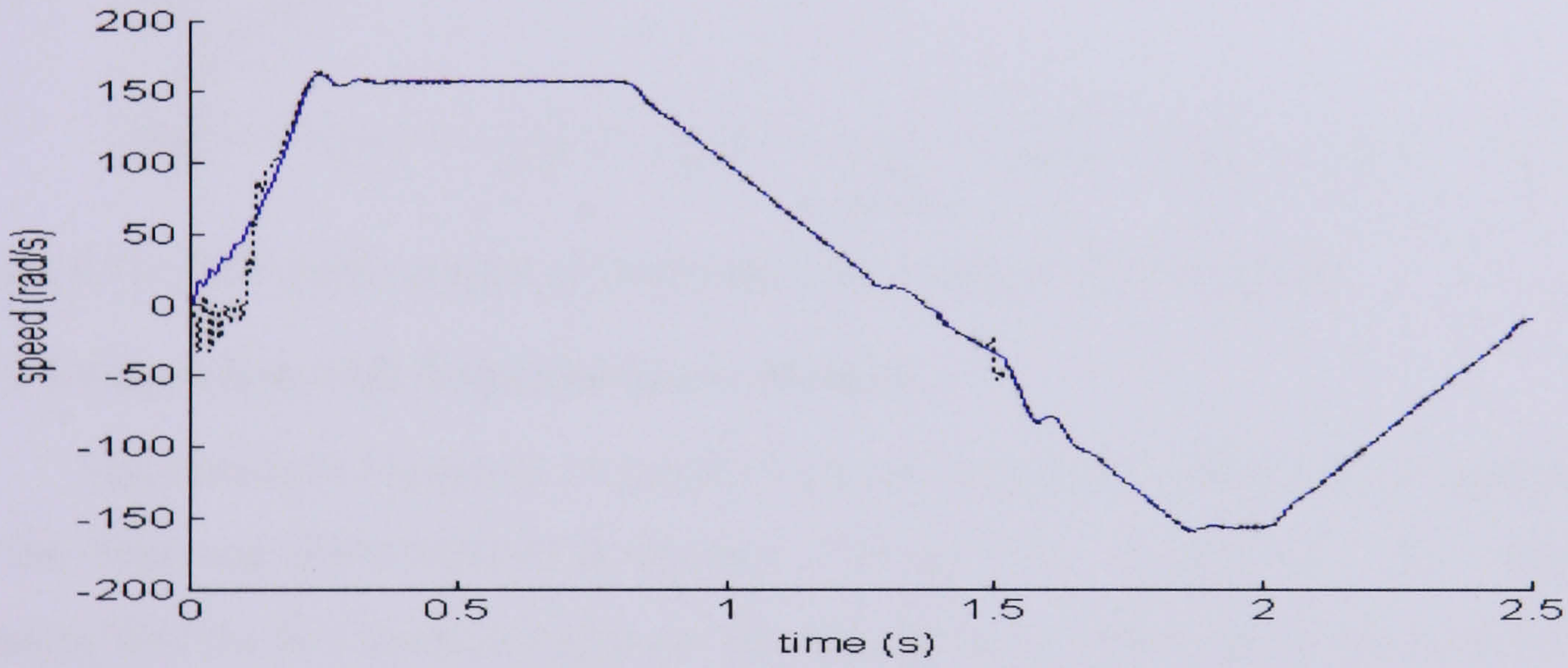
(c)

Figure 6.14: (a) Stator resistance increased by 10% (b) Stator resistance increased by 25% (c) Stator resistance increased by 50%

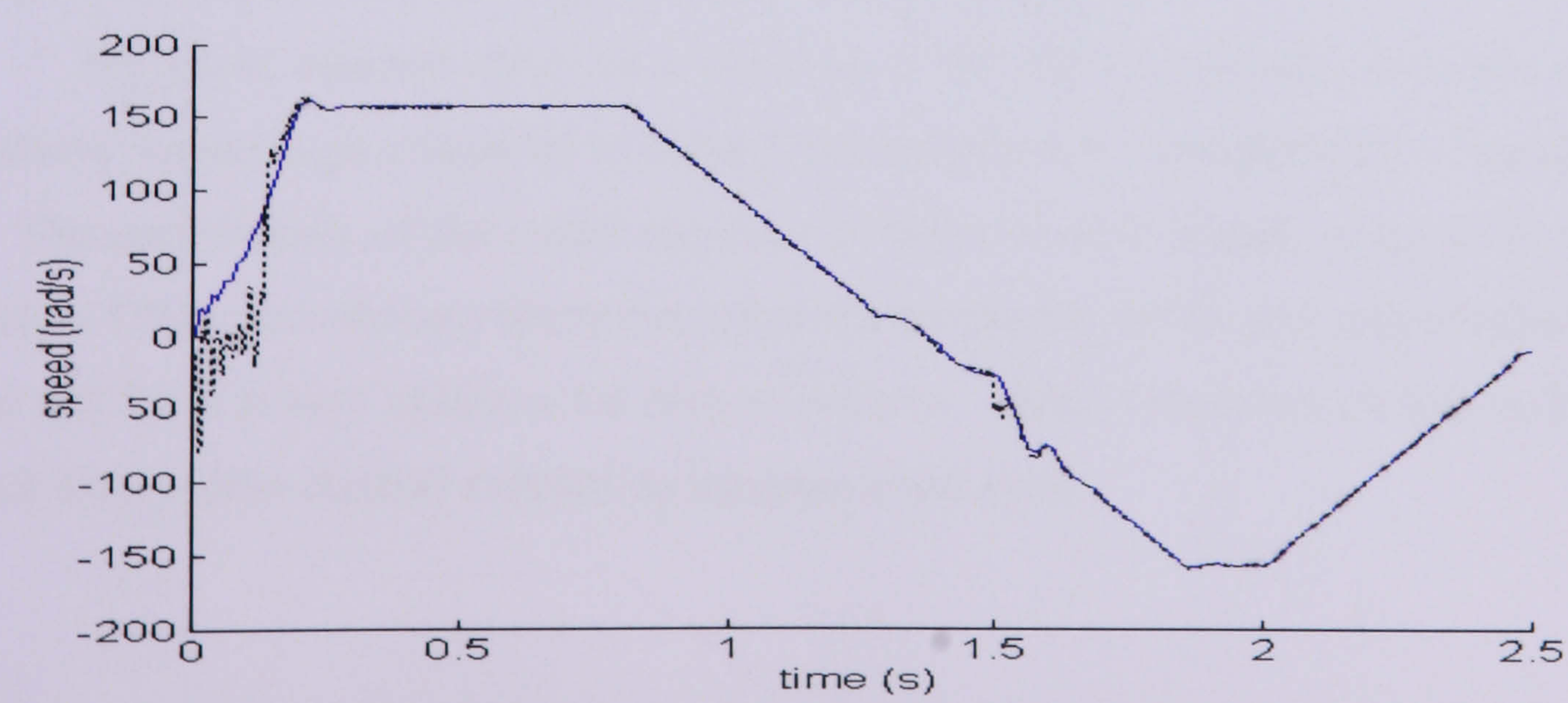
Another test is carried out to illustrate the sensitivity of the tuning parameter to rotor resistance variations. In this test R_r is also increased to 10%, 25% and 50 %. Figure 6.15 (a-c) shows the result of each variation of the rotor resistance. The contribution to error is higher in the transient condition but very low in the steady state.



(a)



(b)



(c)

Figure 6.15: (a) Rotor resistance increased by 10% (b) Rotor resistance increased by 25% (c) Rotor resistance increased by 50%

C. Sensitivity to load changes

To illustrate more of the potential of using the tuning method for the EKF in a sensorless IM drive, the performance of the EKF for load changes is also investigated. A step response is used to represent the load changes. The load torque is increased from no load to 20 Nm at 0.6s with the motor drive at high speed. As shown in Figure 6.16, the method shows that the capability to estimate the speed with variation of the load torque is retained.

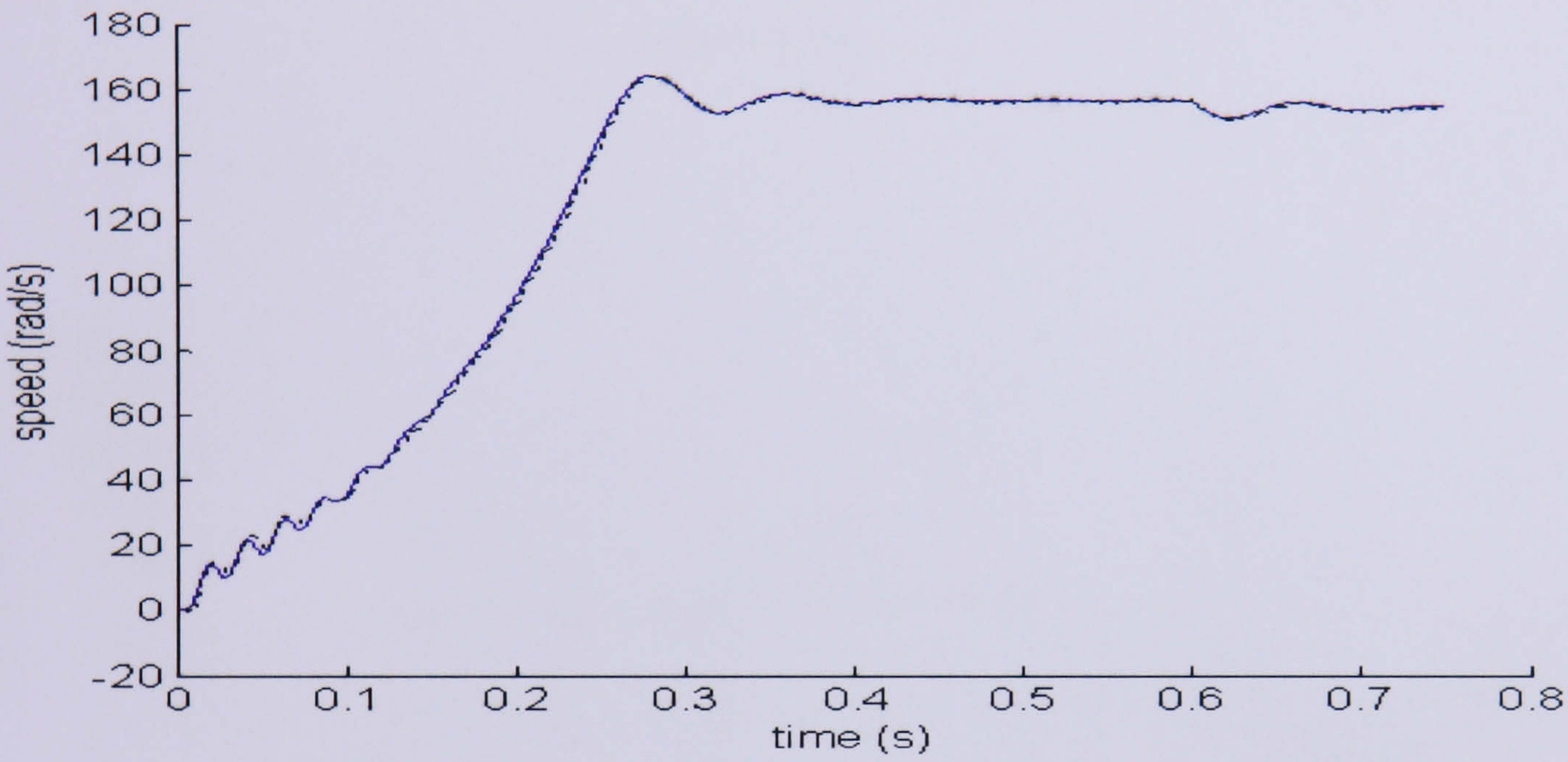
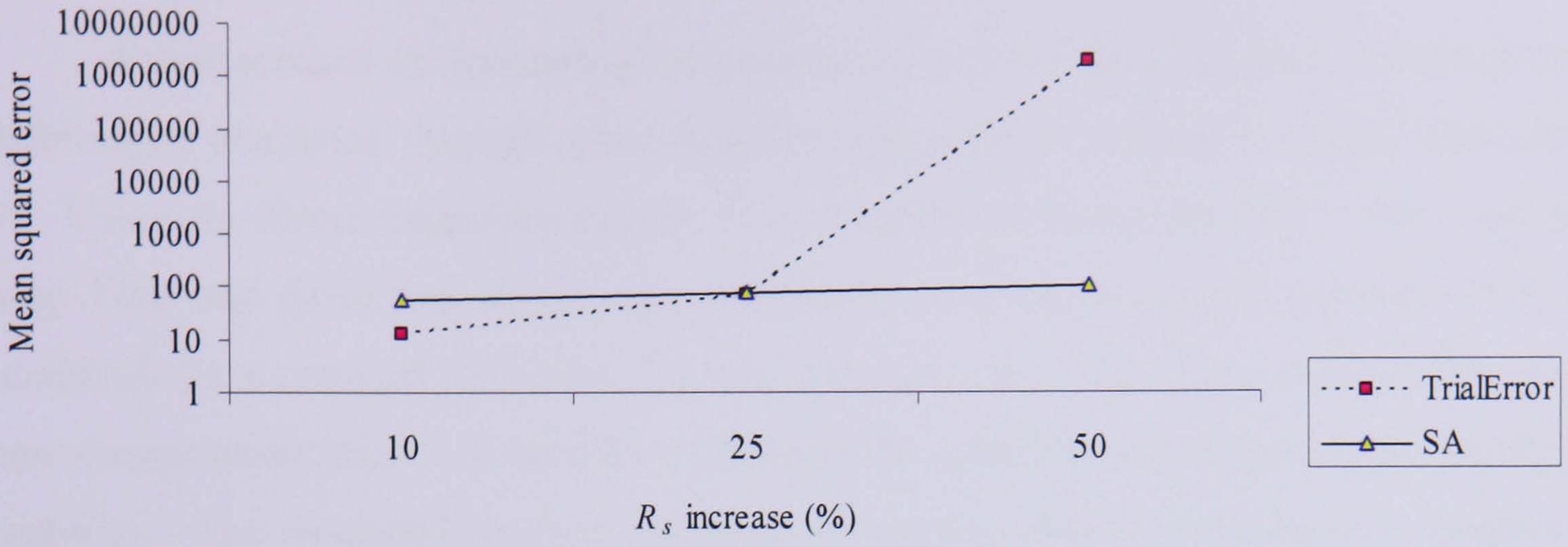


Figure 6.16: EKF performance of increased load torque to 20 Nm at 0.6s

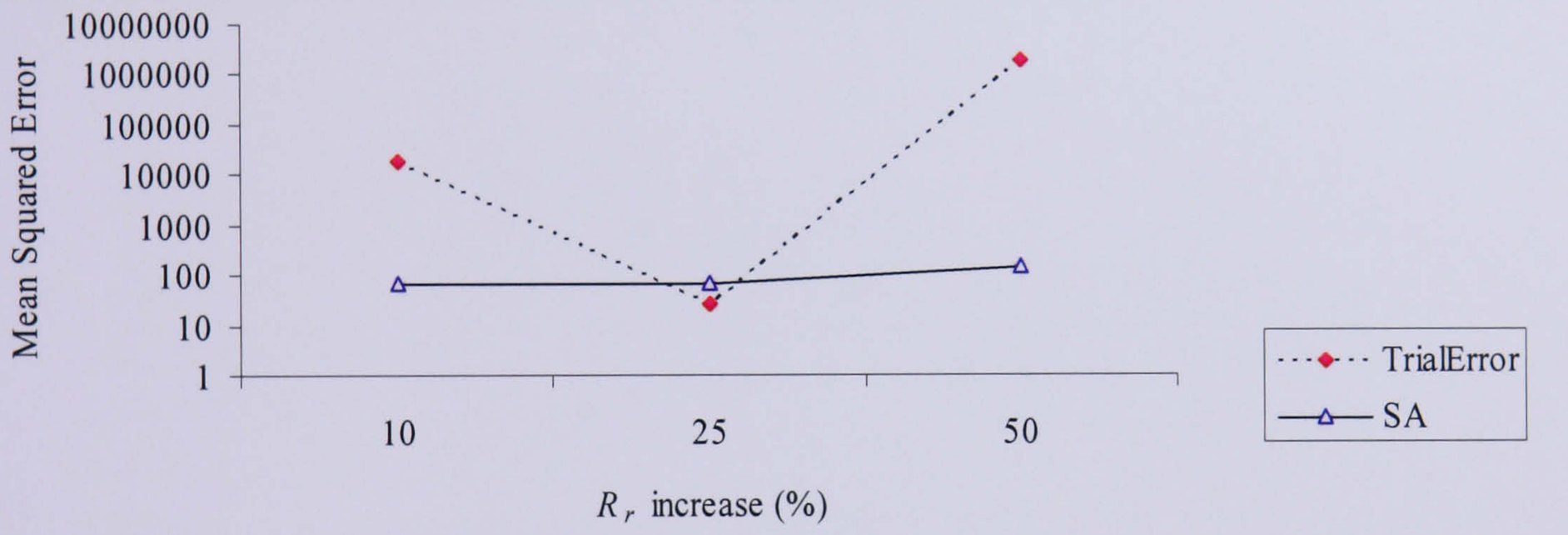
6.7.1 Comparison with Trial and Error Method

The results in Figures 6.14 and 6.15 can be compared with the corresponding results for the Trial and Error method in Figures 5.20 and 5.21, respectively. It is immediately apparent that the SA method improves the robustness to variations of the motor resistance and this is potentially a very valuable feature for the estimator.

The mean squared errors as a function of the stator resistance variation and rotor resistance variation as compared to using Trial and Error are summarised in Figure 6.17(a, b). The performance of the speed response in terms of load torque variation is shown in Figure 6.17(c). It is shown clearly that when the parameter varies; the speed response using Trial and Error is very sensitive for both resistances. When load torque is added during the simulation, neither method exhibits an increase in the error.



(a)



(b)

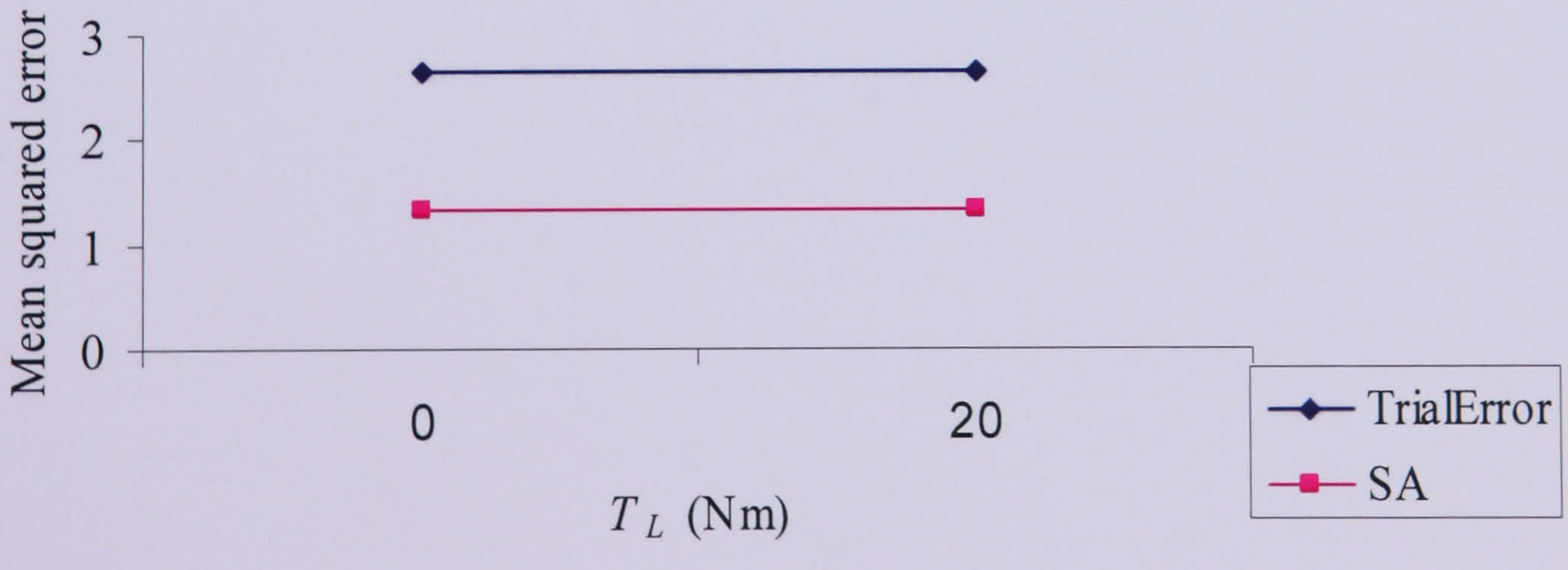


Figure 6.17: (a) Mean squared error with stator resistance variation (b) Mean squared error with rotor resistance variation. (c) With additional load torque of 20 Nm at 0.6s simulation

6.8 Conclusion

A new method of optimising the performance of the EKF has been proposed and its performance evaluated through simulation for both a direct sinusoidal supply and constant V/f. Using the direct sinusoidal supply, SA is capable of tuning the EKF. The comparison using Trial and Error has shown that SA results in a smaller mean squared error. For simulation on a constant V/f drive, SA can also tune the EKF. Even though SA requires some computation time, it is capable of tracking the speed in both transient and steady state condition. The proposed method performs effectively under conditions of constant and variable speeds, including the start up condition. The robustness of the proposed method against parameter variations has been verified. The new SA method gives better results in tuning the EKF as well as being robust to parameter variation.

CHAPTER 7: SIMULATED ANNEALING ON VECTOR CONTROL OF IM

7.1 Introduction

VC or field oriented control is widely used for IM drives. This scheme however requires knowledge of the amplitude and angular position of the rotor flux and in a speed control application, accurate mechanical speed is required. The determination of the speed can be performed by measurement and a transducer such as a tachogenerator or encoder can be used. For sensorless control studies such as this the IM drive can be assumed to have a transducer mounted only for the estimated speed to be compared with the measured.

The main intent of this chapter is to present the application of the EKF in VC, with optimisation of the EKF using the proposed method, first without speed feedback from the estimator, and then with speed feedback from the estimator. The performance, difficulties and problems arising during tuning are also demonstrated and discussed. For sensorless control the robustness of the proposed method especially in the closed loop estimator is also presented.

7.2 Controller

Proportional-Integral-Derivative (PID) controllers are widely used in industry due to their simplicity and ease of retuning online. Several tuning methods of the PI gains have been proposed [66,95]. The aim here is to concentrate on the optimisation of the EKF and to reduce the difficulties in tuning the controller; the gains of this controller are tuned using a simple approach using successive trials.

The guideline for tuning the PID gains on a closed loop response is given by [96]. Table 7.1 gives an idea on how to fine tune the controller. The example given shows that while the gains of the integral and derivative, K_i and K_d respectively, are fixed, increasing the proportional gain, K_p alone can decrease rise time, increase overshoot, slightly increase settling time, decrease the steady state error, and decrease stability margins. Although many controllers have been proposed in recent years for the application of the sensorless VC, the PI controller is still a preferred method [54,97] and has been shown to be a successful controller without the need of the derivative gain. The general block diagram of a

proportional plus integral controller is shown in Figure 7.1. In the controller, if $e(t)$ is the signal input to the PI controller then the output is given by:

$$y(t) = K_p e(t) + K_i \int_0^t e(t)dt \tag{7.1}$$

	Rise Time	**Overshoot**	**Settling Time**	**Steady State Error**	**Stability**
Increasing K_p	Decrease	Increase	Small Increase	Decrease	Degrade
Increasing K_i	Small Decrease	Increase	Increase	Large Decrease	Degrade
Increasing K_d	Small Decrease	Decrease	Decrease	Minor Change	Improve

Table 7.1: The effects of independent P, I, and D tuning on closed loop response

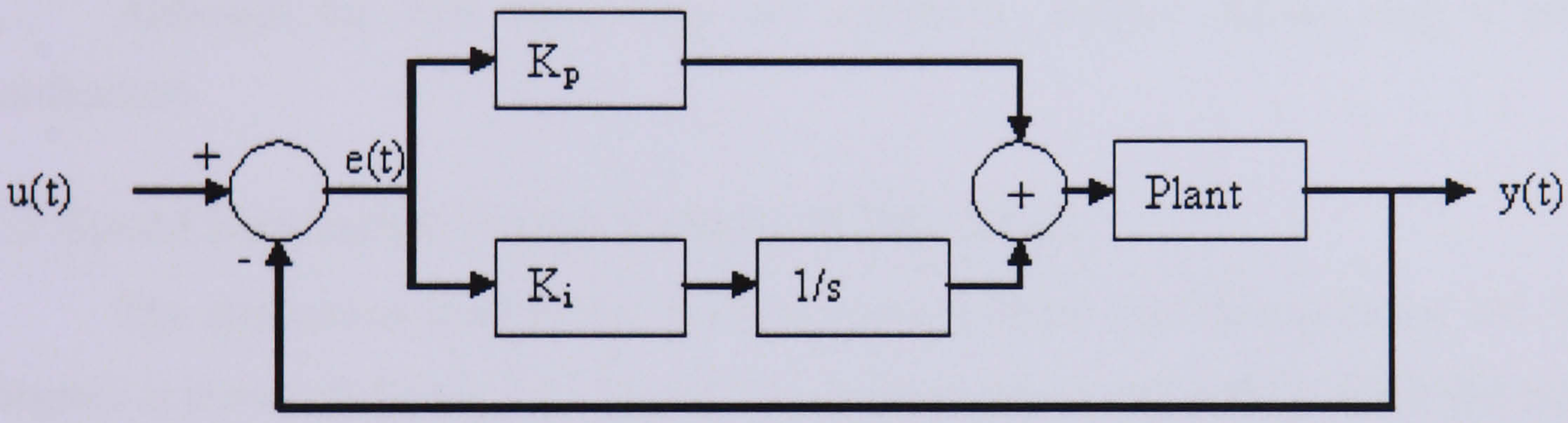


Figure 7.1: General block diagram of the Proportional –Integral controller

In this VC application, three P-I controllers are implemented, where all the references or commands are indicated by using superscript '*'. The speed reference ω_r^* is compared with the speed response either from the estimate or from the transducer, to create the speed error. Any change in the rotor speed due to a change in the torque will be compensated by the PI controller and generates the required command for the torque producing current component i_{qse}^*. The constant flux command is used and any difference between the reference and the estimated current i_{dse}^* will have an affect on the developed torque. The correction is made by the PI controller. The gains are represented by DK_p, DK_i, QK_p and QK_i for the proportional and integral of d q axes respectively. PI controllers used for tracking the current are as follows:

$$u_d = DK_p(i_{dse}^* - i_{dse}) + DK_i \int (i_{dse}^* - i_{dse})dt \tag{7.2}$$

$$u_q = QK_p(i_{qse}^* - i_{qse}) + QK_i \int (i_{qse}^* - i_{qse})dt \tag{7.3}$$

The rotor flux is kept at a constant value and the PI controllers used for tracking the speed are as follows:

$$i_{qse}^{*} = QK_p(\omega_r^{*} - \omega_r) + QK_i \int (\omega_r^{*} - \omega_r)dt \tag{7.4}$$

Although, many methods and theories have been proposed, here a straight forward method is used for tuning the gain for the entire PI controller:

1. Consider DK_p, QK_p for both controllers to be zero.
2. Increase DK_p, QK_p gradually and simulate the system until the overshoot gets to its optimum.
3. Using the best set of DK_p, QK_p obtained, the DK_i, QK_i is then increased gradually.
4. Using the best gain of DK_i, QK_i obtained, the DK_p, QK_p is once again varied until satisfactory results achieved.

Although the first three steps are sufficient, another further step is taken for verification.

7.3 Speed Estimation Vector Control of IM Drives

The application used in this study involves a direct rotor flux oriented VC. The VC diagram is shown in Figure 7.2. Figure 7.2(a) shows the torque control of the IM running at constant speed and Figure 7.2(b) shows closed loop speed control of the IM. The rotor flux magnitude and position is obtained by taking a readily measured value from the stator current and measured or estimated rotor speed. The rotor flux developed in the dynamic flux model is based on the d q axes equation:

$$\frac{d\psi_{dr}}{dt} = -R_r\left(\frac{\psi_{dr} - L_m i_{ds}}{L_r}\right) - \omega_r \psi_{qr} \tag{7.5}$$

$$\frac{d\psi_{qr}}{dt} = -R_r\left(\frac{\psi_{qr} - L_m i_{qs}}{L_r}\right) + \omega_r \psi_{dr} \tag{7.6}$$

The rotor flux magnitude and position is obtained using the following equation:

$$|\psi_r| = \sqrt{\psi_{dr}^{2} + \psi_{qr}^{2}} \tag{7.7}$$

$$\arg\psi_r = \tan^{-1}\frac{\psi_{qr}}{\psi_{dr}} \tag{7.8}$$

The transformation from the synchronous frame to the stationary reference frame is given by:

$$\begin{bmatrix} i_{dss} \\ i_{qss} \end{bmatrix} = \begin{bmatrix} \cos\theta_e & -\sin\theta_e \\ \sin\theta_e & \cos\theta_e \end{bmatrix} \begin{bmatrix} i_{dse} \\ i_{qse} \end{bmatrix} \tag{7.9}$$

While the transformation from the stationary reference frame to the synchronous frame is:

$$\begin{bmatrix} i_{dse} \\ i_{qse} \end{bmatrix} = \begin{bmatrix} \cos\theta_e & \sin\theta_e \\ -\sin\theta_e & \cos\theta_e \end{bmatrix} \begin{bmatrix} i_{dss} \\ i_{qss} \end{bmatrix} \tag{7.10}$$

The rotor flux can be determined as the product of the magnetising inductance and d-axis current.

$$\psi_{dr} = L_m i_{dse} \tag{7.11}$$

The steady state torque, T_e in the synchronous reference frame controlled only by the q-axis stator current can be expressed as follows:

$$T_e = \frac{3}{2}\frac{P}{2}\frac{L_m}{L_r}\psi_{dr} i_{qse} \tag{7.12}$$

or can be written as:

$$T_e = K_T \psi_{dr} i_{qse} \tag{7.12a}$$

where $K_T = \frac{3}{2}\frac{P}{2}\frac{L_m}{L_r}$

The mechanical dynamics, neglecting friction, windage and core losses is calculated using:

$$T_e - T_L = J\frac{d\omega_r}{dt} \tag{7.13}$$

The study is carried out to access the speed estimation first with the loop closed by a speed sensor and then closed with the estimated speed. The speed estimation with a sensor and without a sensor is shown in Figure 7.3 and 7.4 respectively. In Figure 7.3, the EKF estimator calculates the rotor speed from the stator voltage and current components but is not used for control; instead a feedback speed signal for the flux observer and the speed controller are employed from the IM. This is also known as an open loop estimator. Alternatively, the speed signal feedback to the speed controller is the speed obtained from the estimator, which is also used for determining the flux position. This is known as speed estimation without a speed sensor, or closed loop estimator, and the diagram is shown in Figure 7.4.

For application of speed estimation in the VC, the tuning of the PI controller is complex and time consuming while its proper selection is extremely important for good performance. In this section, a different set up of the controller is used. The first simulation

shows the effect of a constant speed supplied to the IM. Without the EKF, the current controller used the gains of DK_p=500 DK_i=500 and QK_p=500 QK_i=500. For speed estimation with the sensor and without the sensor, SET A uses a speed controller gain of QK_p=8 QK_i=20 and current controller DK_p=200 DK_i=3 and QK_p=500 QK_i=3 while SET B uses a speed controller gain of QK_p=8 QK_i=20 and current controller DK_p=200 DK_i=3 and QK_p=100 QK_i=3. The difference between SET A and B is the value of QK_p in the current controller representing the torque producing current. The outcome of using the controller gains in each type will be discussed in detail.

An IM model based on the stationary two axes (d,q) reference frame is used. The motor ratings and parameters are given in Chapter 3. To determine the torque and flux producing current, at rated speed, the RMS rotor flux obtained is 0.6848 and calculated from the steady state equivalent circuit. With the conversion of the real three phase motor into a two phase equivalent model, a coefficient 1.5 is introduced and since the vectors are calculated from the time function of the physical quantities, the peak value is normally used rather than RMS [4]. The flux is calculated as:

$$\psi_r^* = 1.5(\sqrt{2})0.6848 = 1.45Wb \tag{7.14}$$

Using Eqn (7.11), the flux command ψ_r^* corresponds to a constant current command in the synchronous frame i_{dse}* of 12.1 A.

The torque producing current i_{qse}* is calculated based on Eqn (7.12) and gives about 11.9 A. Based on Eqn (7.12):

$$K_T = \frac{3}{2}\frac{4}{2}\frac{0.12}{0.1274} = 2.83 \tag{7.15}$$

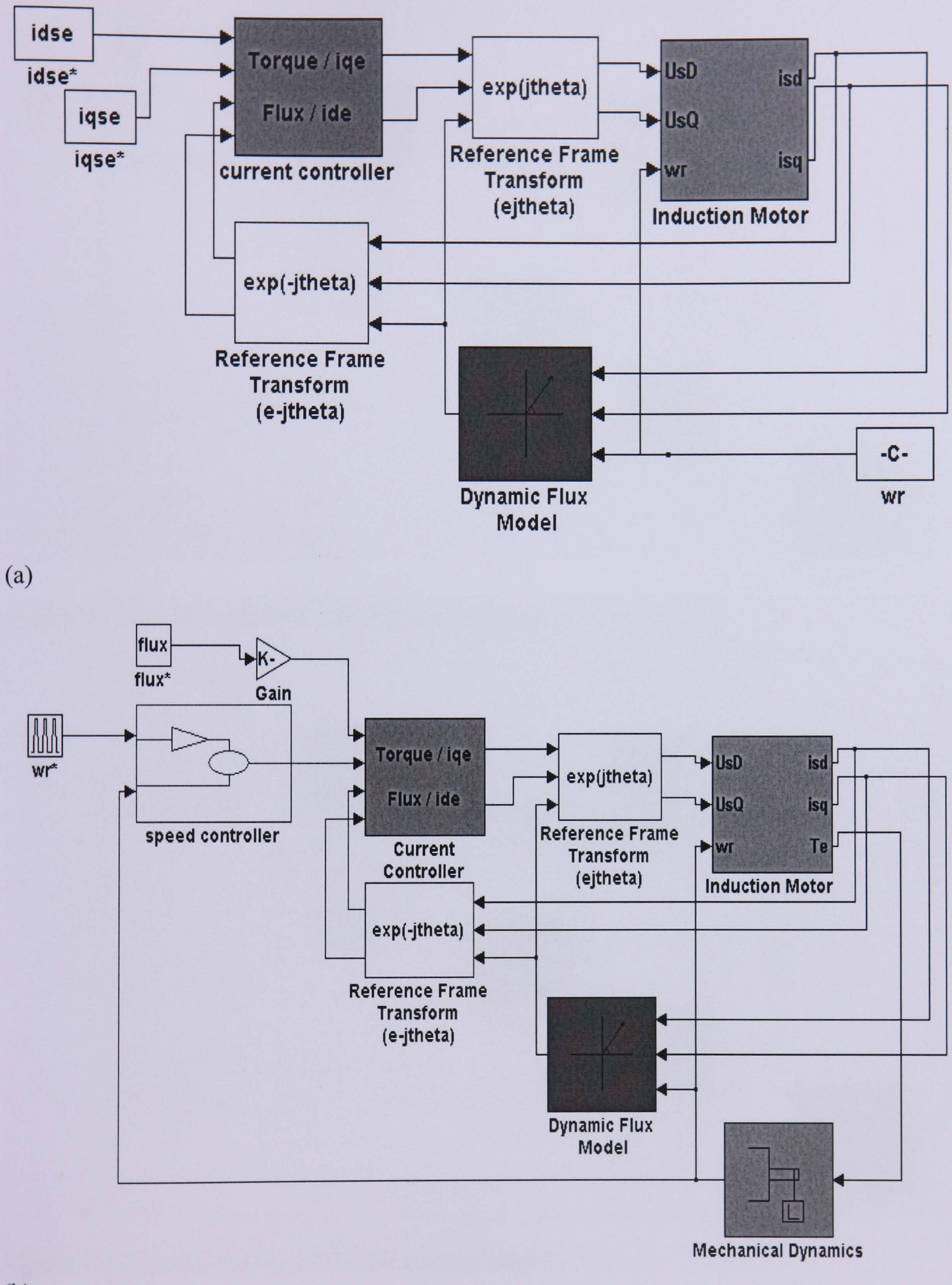
$$T_e = K_T \psi_{dr} i_{qse} = 48.8Nm \tag{7.16}$$

The speed estimator is based on the EKF, implemented in the stationary reference frame. The corresponding noise covariance matrices for **Q**, **R** and **G** are based on the Trial and Error method.

$$Q = diag[0.00001 \quad 0.00001 \quad 0.00001 \quad 0.00001 \quad 1]$$

$$G = diag[0.01 \quad 0.01 \quad 0.01 \quad 0.01 \quad 0.01]$$

$$R = diag[0.01 \quad 0.01]$$



(a)

(b)

Figure 7.2: (a) Torque control of IM architecture (b) Speed control of IM architecture

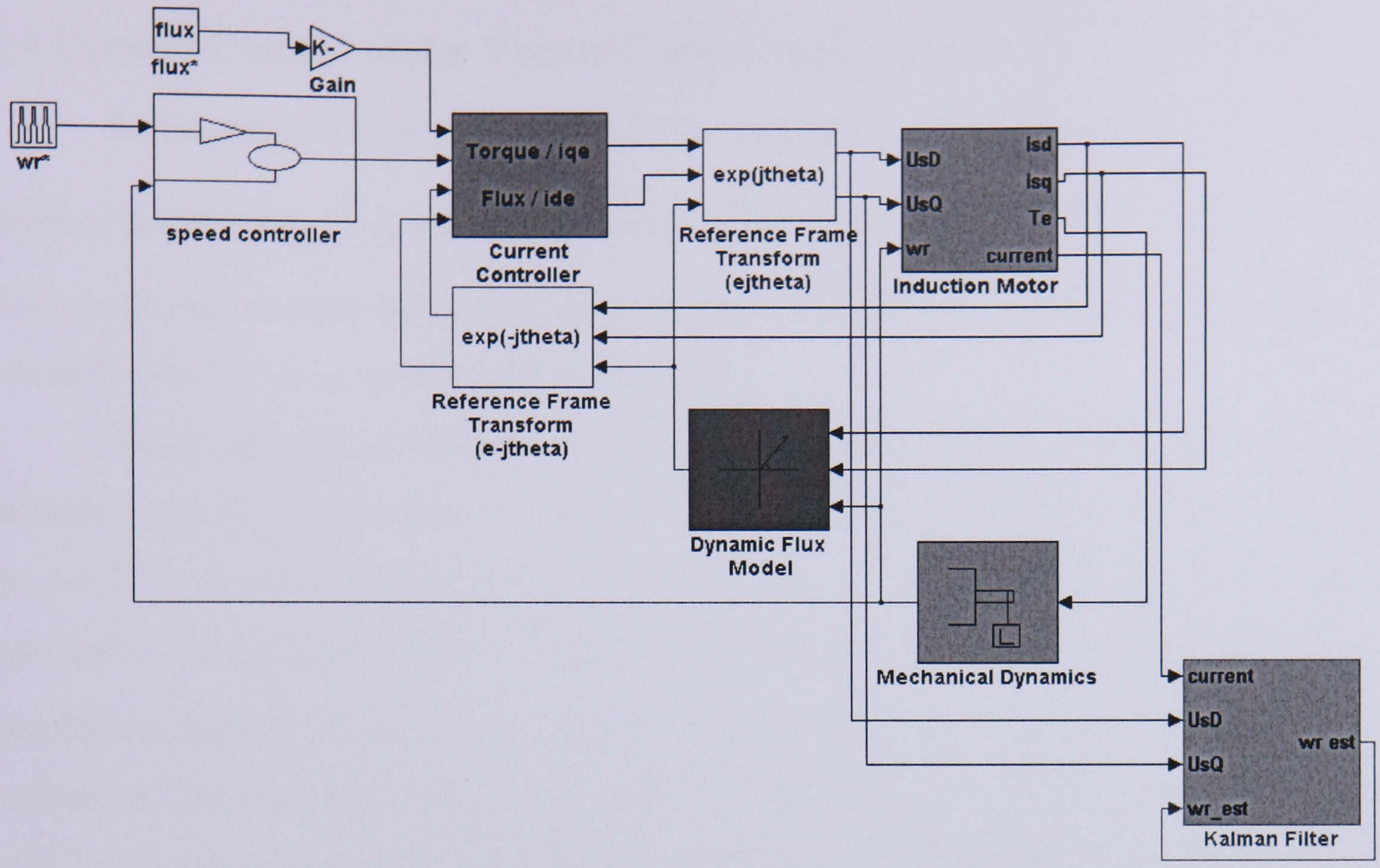


Figure 7.3: Speed Estimation with a speed sensor block diagram

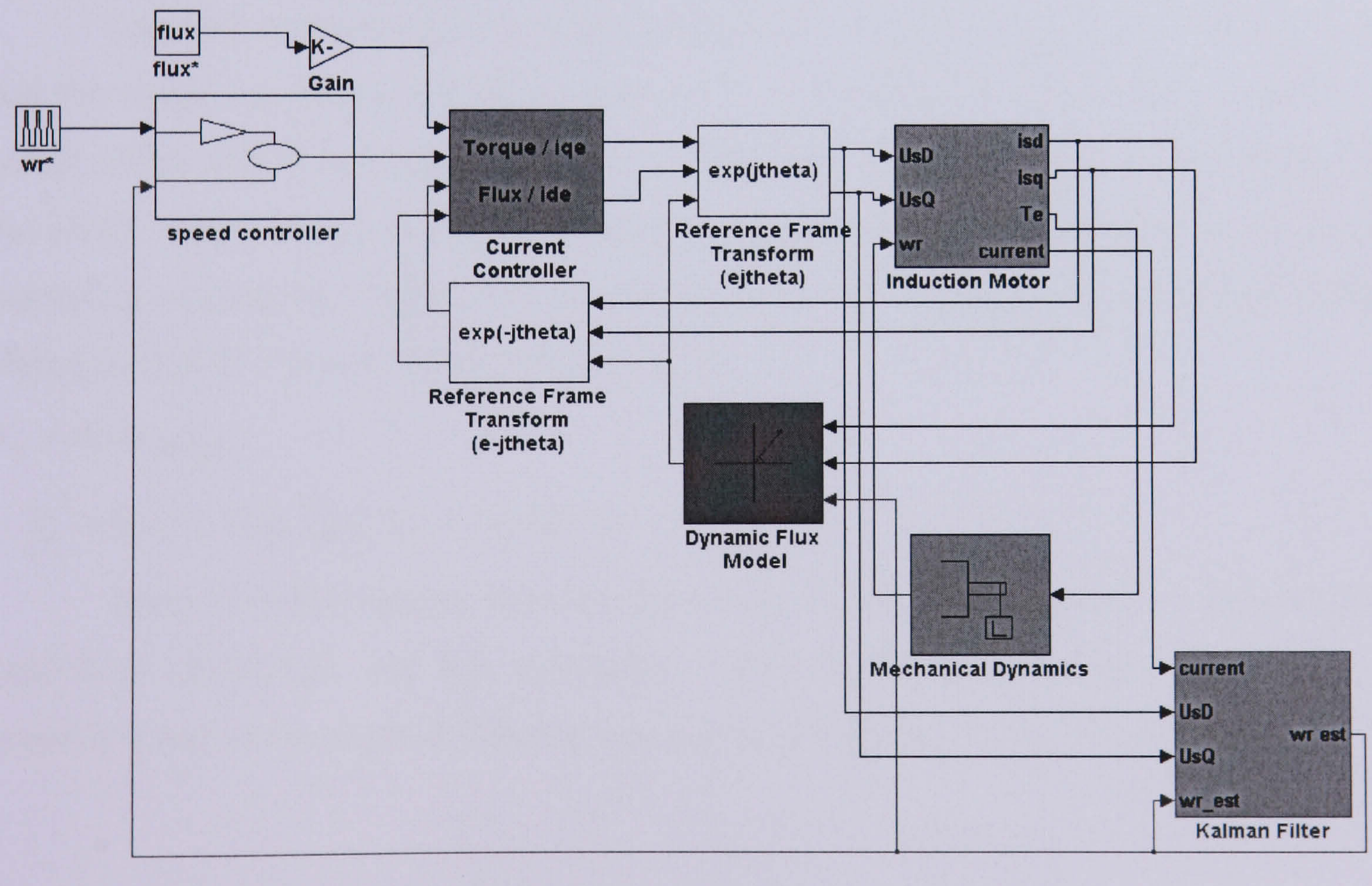


Figure 7.4: Speed Estimation without a speed sensor

7.4 Current Control of the Vector Control application

As shown in Figure 7.2(a), the IM is running at a constant speed ω_r. The torque producing current i_{qse}^* is calculated based on Eqn (7.12) and gives about 11.9 A. The rotor flux producing current calculated at rated speed is given in the previous section. The constant speed is set at rated which is 154 rad/s.

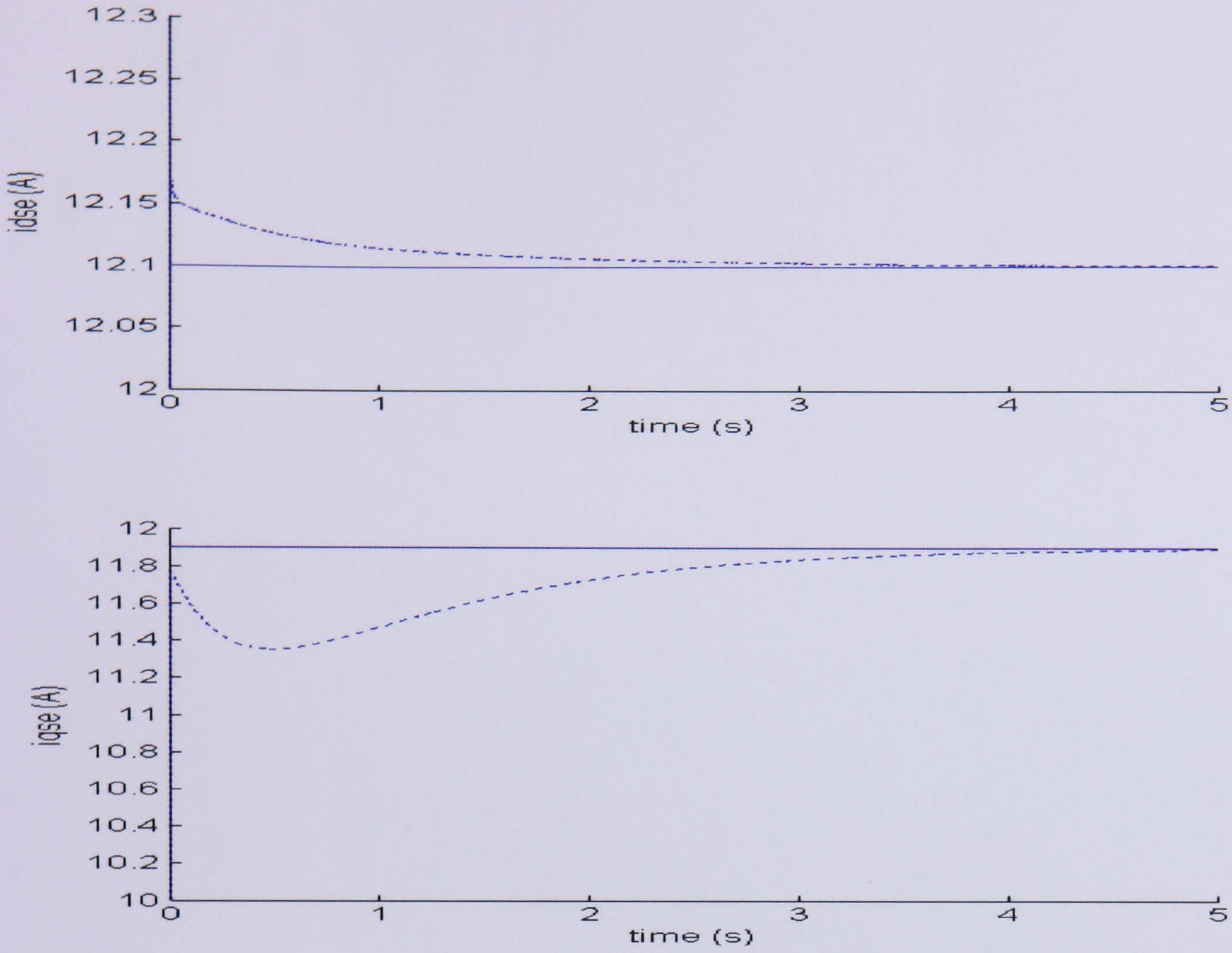
Using gains DK_p=500 DK_i=500 and QK_p=500 QK_i=500, the currents are close to the reference current and are shown in Figure 7.5(a, b). The developed torque is expected to be the same as the torque calculated as shown in Eqn (7.12) and from the equivalent circuit calculation which is 48.8 Nm. As indicated in Figure 7.6(a, b), T_e is increased gradually and settles at about 48 Nm with a rotor flux of 1.45Wb at 5s. Figure 7.7(a, b) shows stator currents in the stationary reference frame. Figure 7.8 (a, b) shows there is no difference with 101% of constant speed and only a slight difference at 70% of constant speed. The test has shown that the VC is capable of maintaining the torque at different constant speeds.

Imposed constant speed is now removed by using the mechanical dynamics block and the torque producing current is changed to a repeating sequence input function. The actual torque and flux producing current can be seen in Figure 7.9(a, b). In Figure 7.9(b), the initial torque during the first transient is not fully developed therefore is not used for numerical calculation. Figure 7.10 shows the motor speed and torque developed relations. Based on Eqn (7.13) and Figure 7.7(a) at 0.25s,

$$T_e = K_T \psi_{dr} i_{qse}$$

$$\therefore T_e = K_T (11.4)(0.12)(12.1) = 46.84 Nm$$

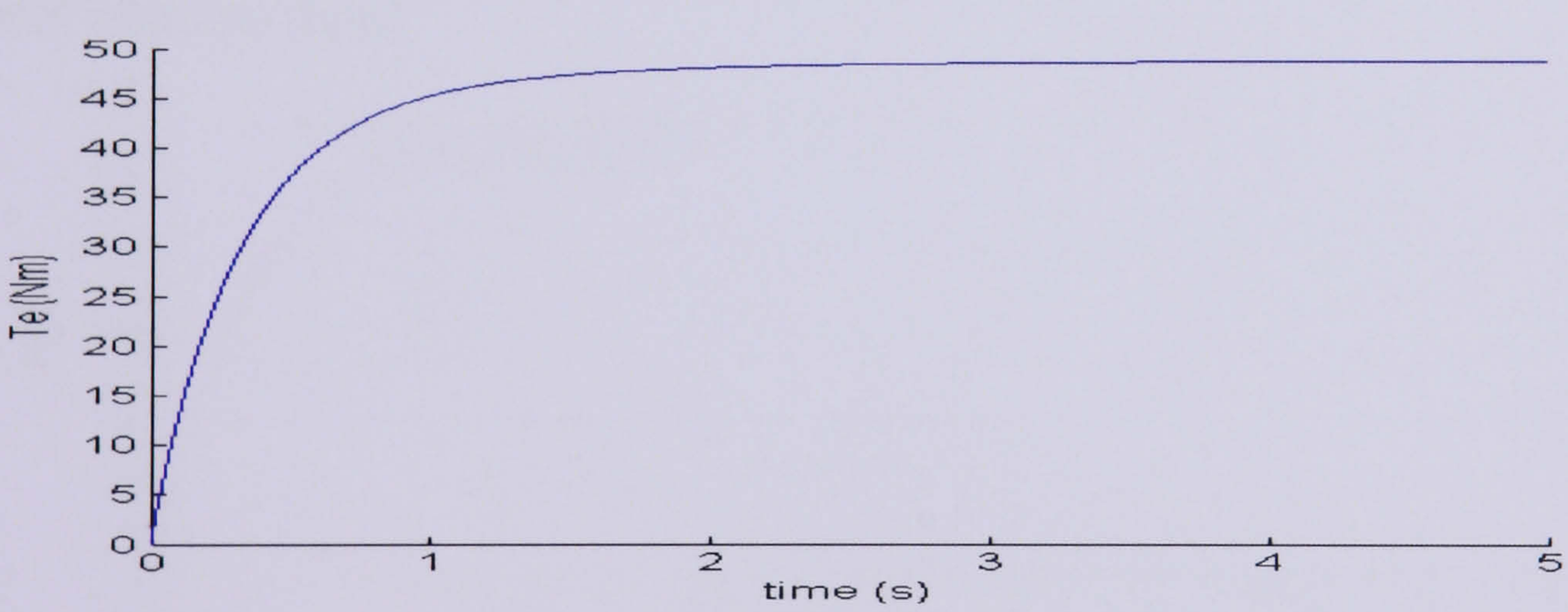
There is a discrepancy between the values of steady state torque calculated by the equivalent calculation and this simulation. This shows that the controller needs to be properly tuned as the slight difference may not be giving optimal developed torque.



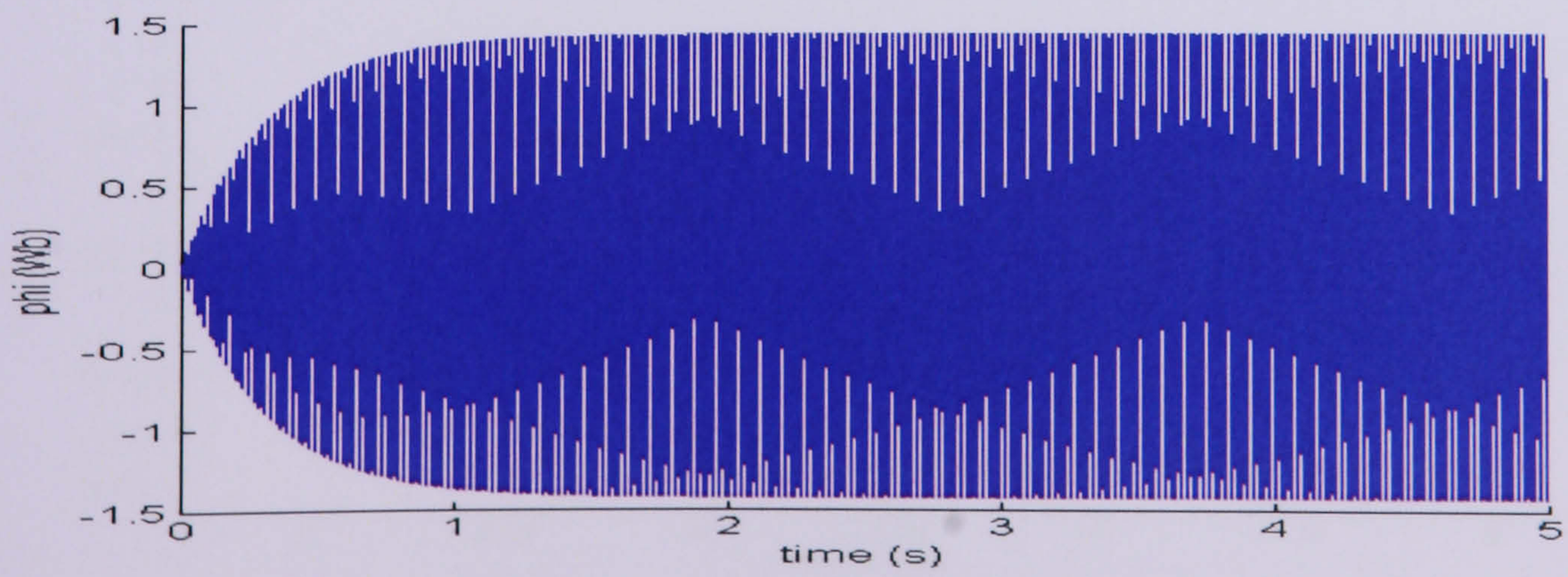
(a)

(b)

Figure 7.5: (a) Flux producing current (b) Torque producing current

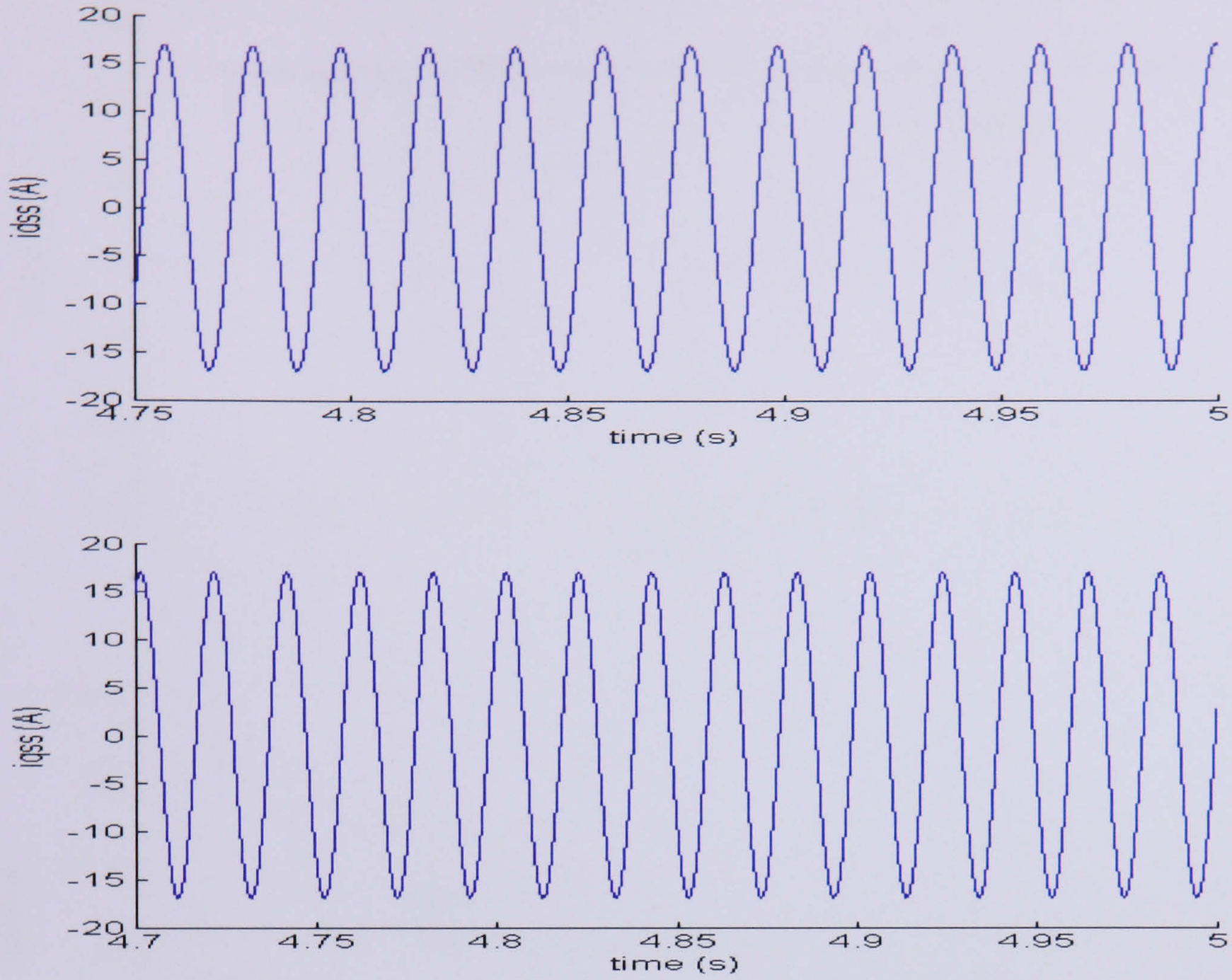


(a)



(b)

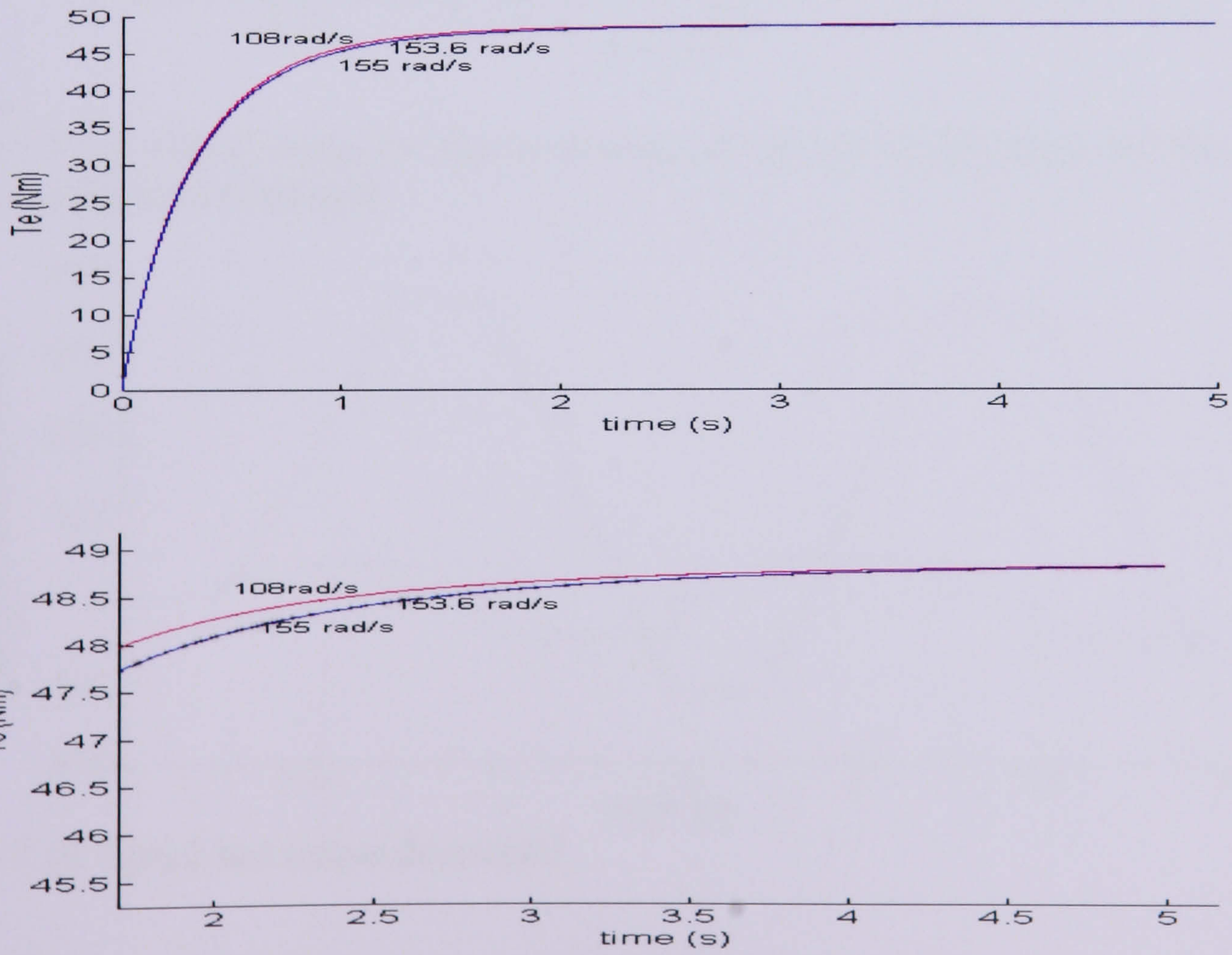
Figure 7.6: (a) Developed Torque (b) Rotor flux of the IM



(a)

(b)

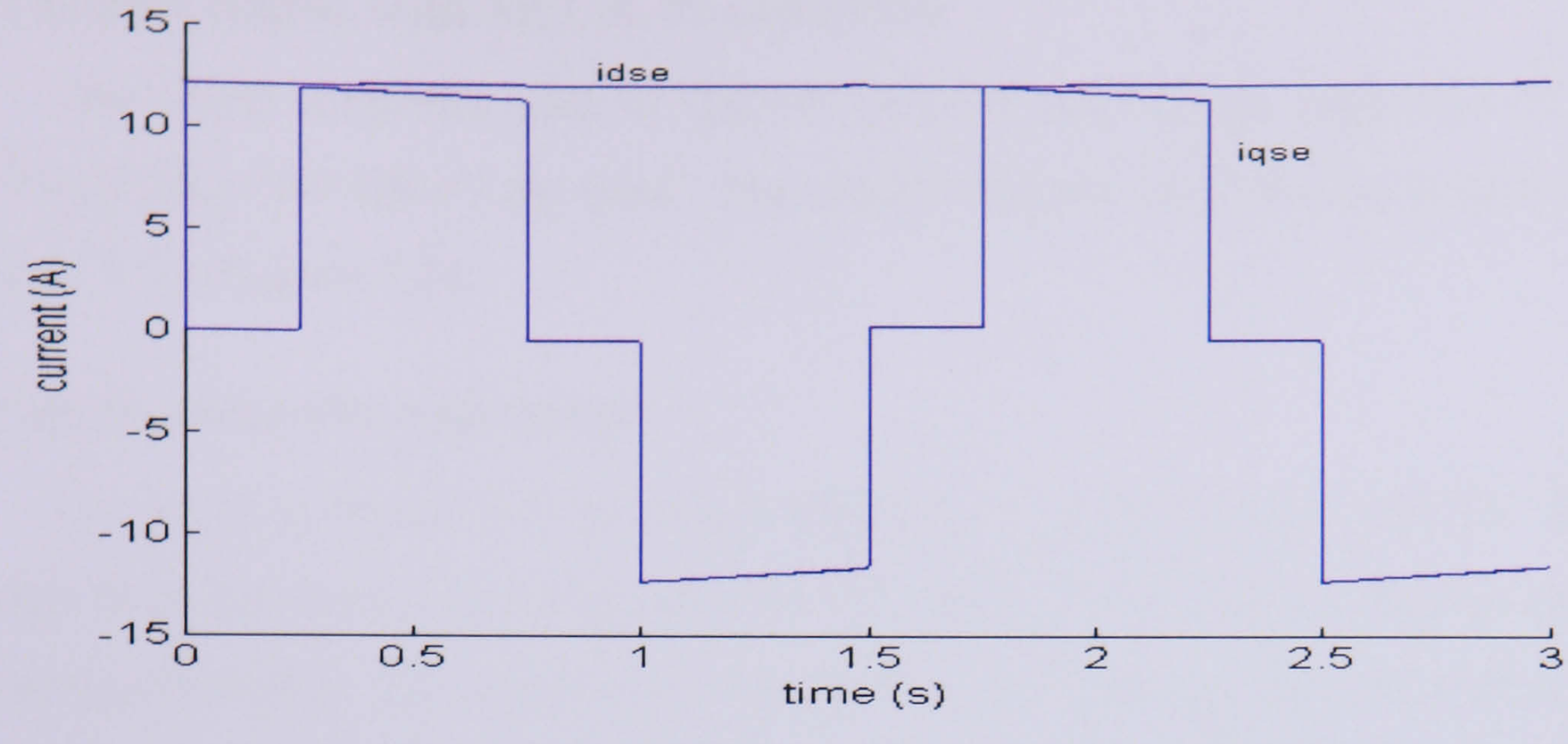
Figure 7.7: (a) Stator current in stationary reference frame (b) Stator current q-axis in the stationary reference frame



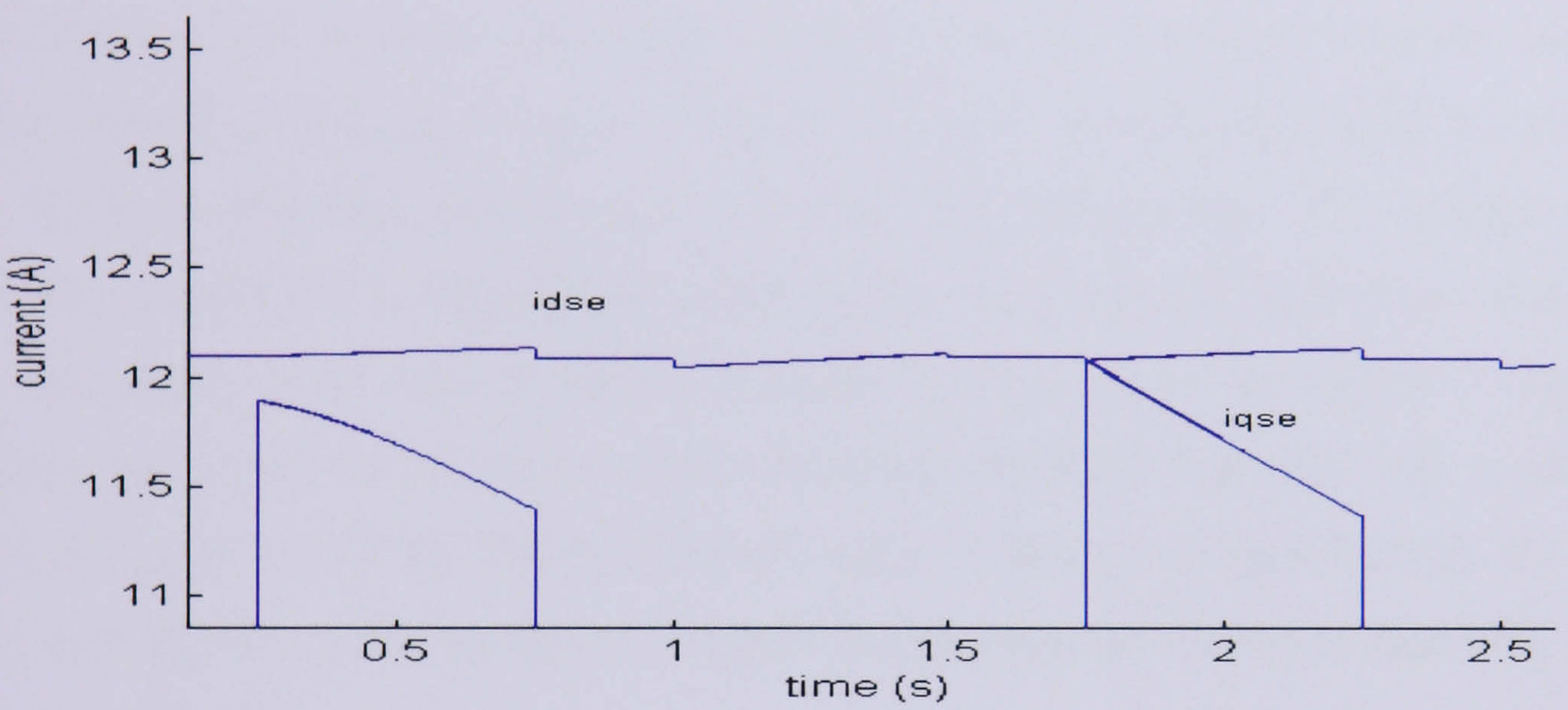
(a)

(b)

Figure 7.8: (a) Developed torque at constant speed 153.6 rad/s, 155 rad/s and 108 rad/s (b) Developed torque at constant speed 153.6 rad/s, 155 rad/s and 108 rad/s in enlarged form



(a)



(b)

Figure 7.9: (a) Actual torque and flux producing current (d) Actual torque and flux producing current in enlarged

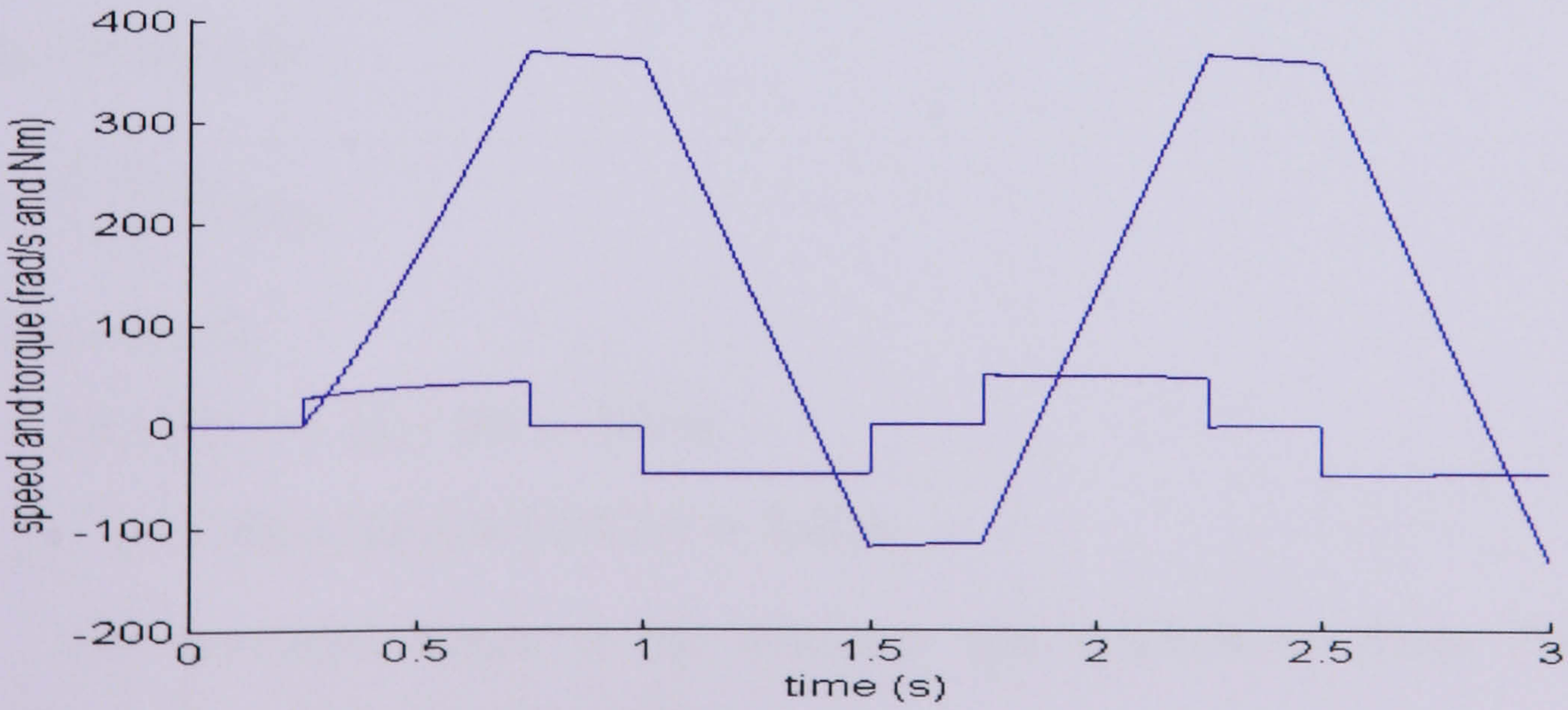


Figure 7.10: Speed and torque developed

7.5 Vector Control with SET A PI controller

The speed controller gain of QK_p=8 QK_i=20 and current controller of DK_p=200 DK_i=3 and QK_p=500 QK_i=3 are used. The results obtained from the simulation are shown in Figure 7.7 to Figure 7.16.

7.5.1 Speed estimation with sensor

As shown in Figure 7.3, the actual speed from the IM is fed to the VC. The results obtained from the simulation are shown below. Figure 7.11 (a, b) show that the current loop is properly controlled. The estimator is capable of tracking the actual speed with almost zero error during steady state operation. This is shown in Figure 7.12(a, b). Without the use of power electronic ie. inverter, the supply voltage to the IM is directly from the output of the current controller. This can be seen in Figure 7.13(a, b). Initially the speed is zero until 0.2s after which it increases slowly until it reaches the steady state. The change in torque-producing current has a slight effect on the initial frequency of the voltage. Both d,q axes show the steady state current settled at about 12A, as shown in Figure 7.14(a, b). The estimator has shown to perform well in estimating the rotor flux with only small errors as shown in Figure 7.15(a, b). The developed torque is shown in Figure 7.15(c). By using Eqn (7.13) and referring to the section of constant deceleration between 0.6s and 0.8s:

$$T_e = 0.05(\frac{0-155}{0.8-0.6}) = -38.75Nm$$

Comparing with the Eqn (7.12) and (7.12a) and using the simulated values of ψ_{dr} and i_{qse} at 0.6s and 0.8s,

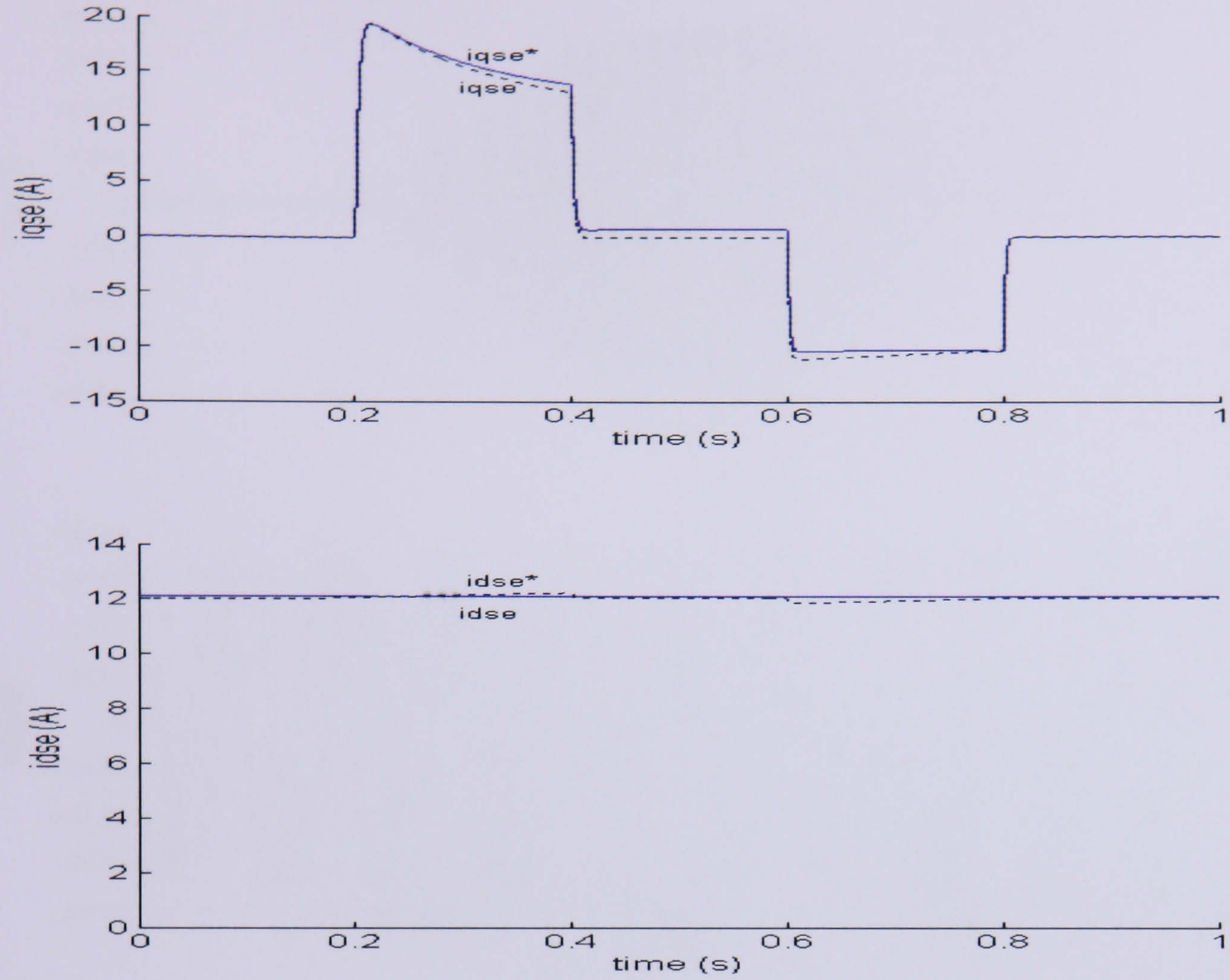
$$T_e = \frac{3}{2}\frac{P}{2}\frac{L_m}{L_r}\psi_{dr} i_{qse}$$

Therefore at 0.6s,

$$T_e = 2.83(0.12)(-11.2)(11.88) = -45Nm$$

At 0.8s, $T_e = 2.83(0.12)(-10.4)(12.05) = -42Nm$

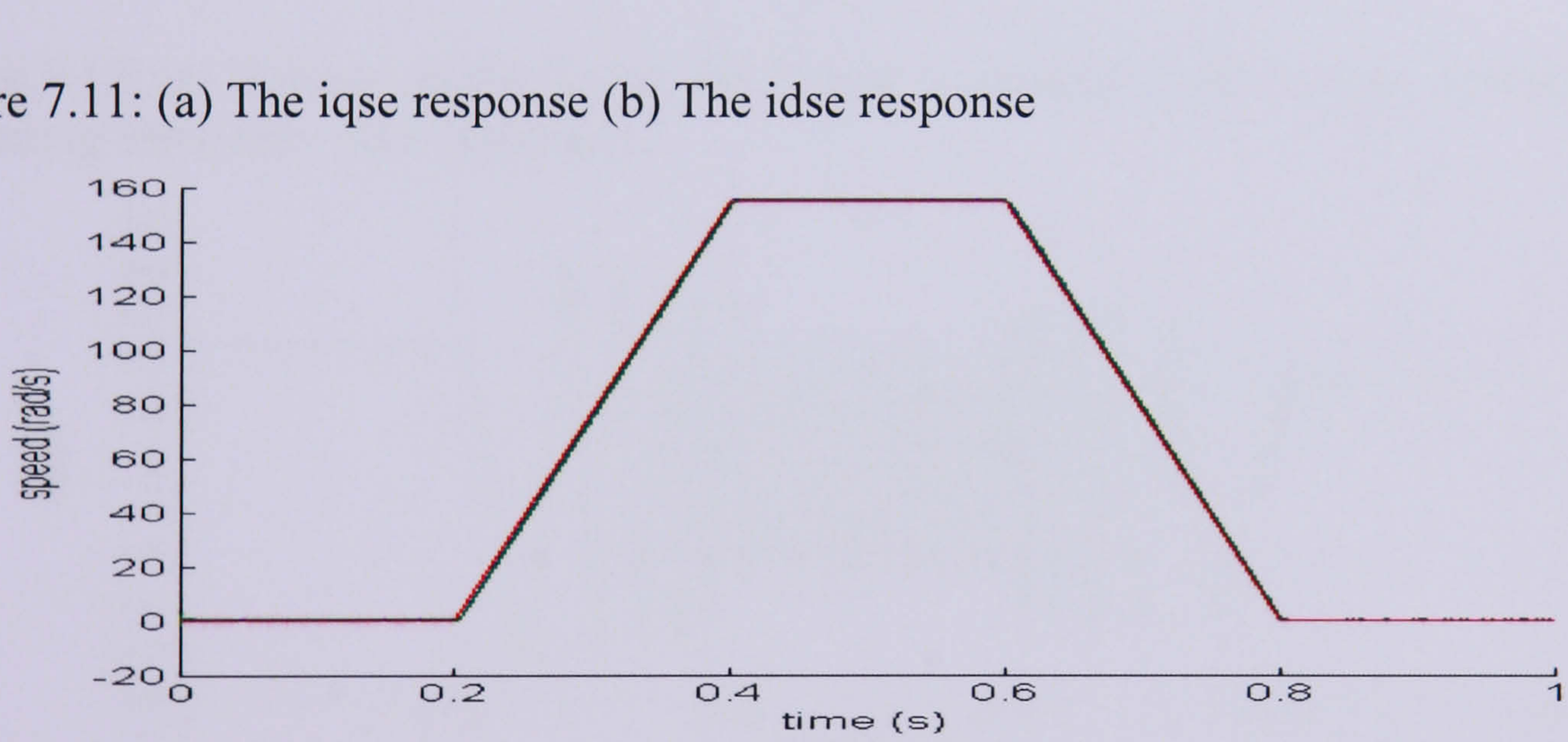
The developed torque at the stationary and synchronous frame has a small discrepancy due to the controller setting.



(a)

(b)

Figure 7.11: (a) The iqse response (b) The idse response



(a)

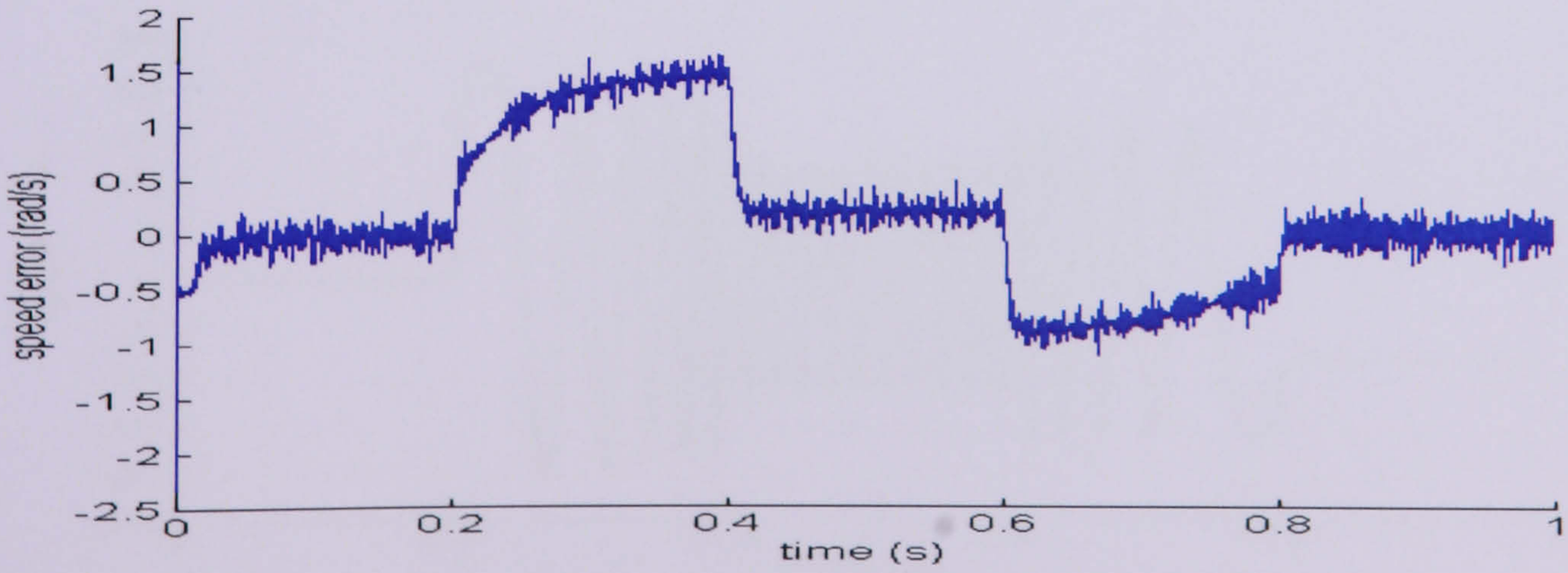
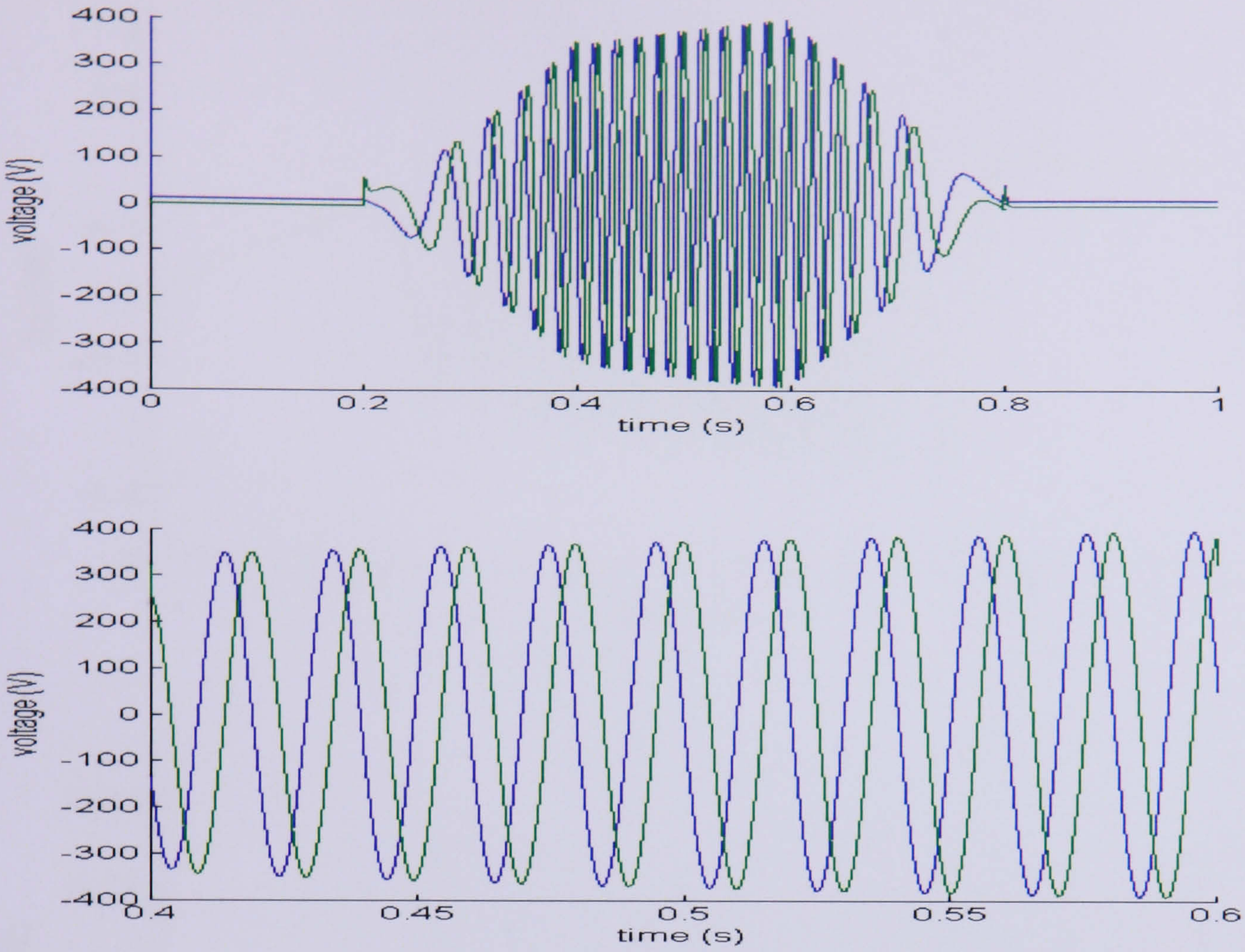


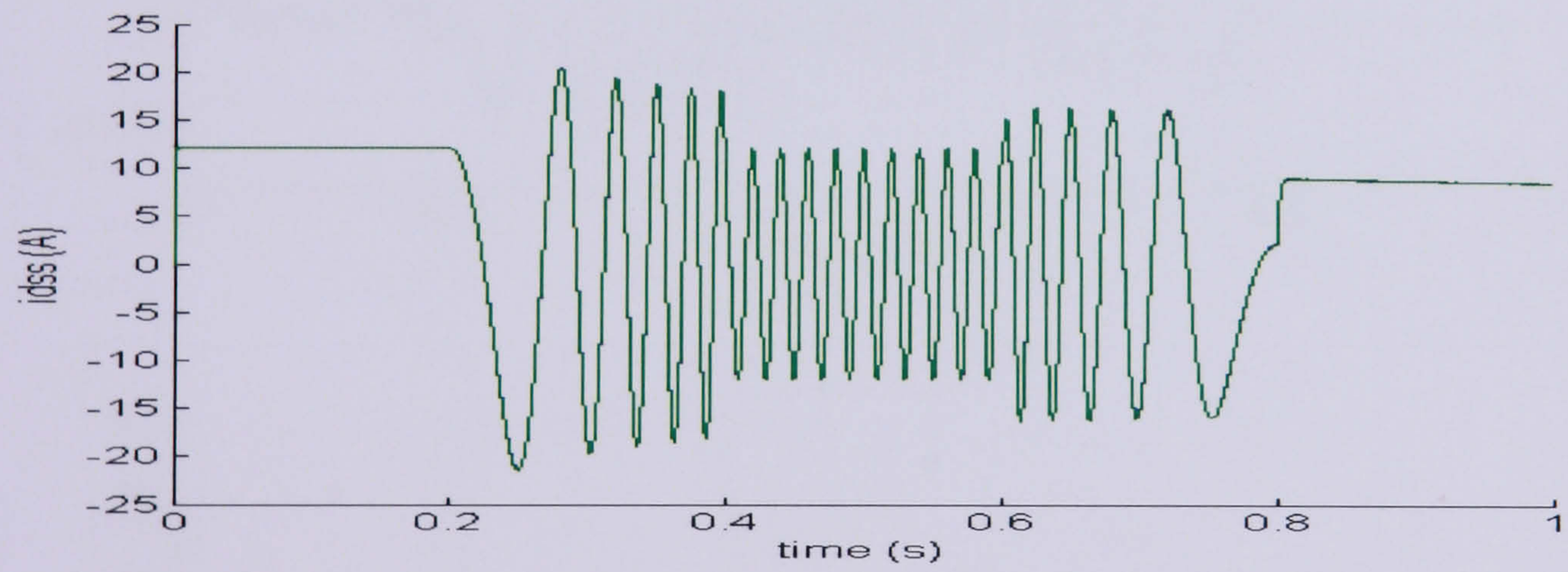
Figure 7.12: (a) Speed response of reference, actual and estimated (b) Error between actual and estimated speed



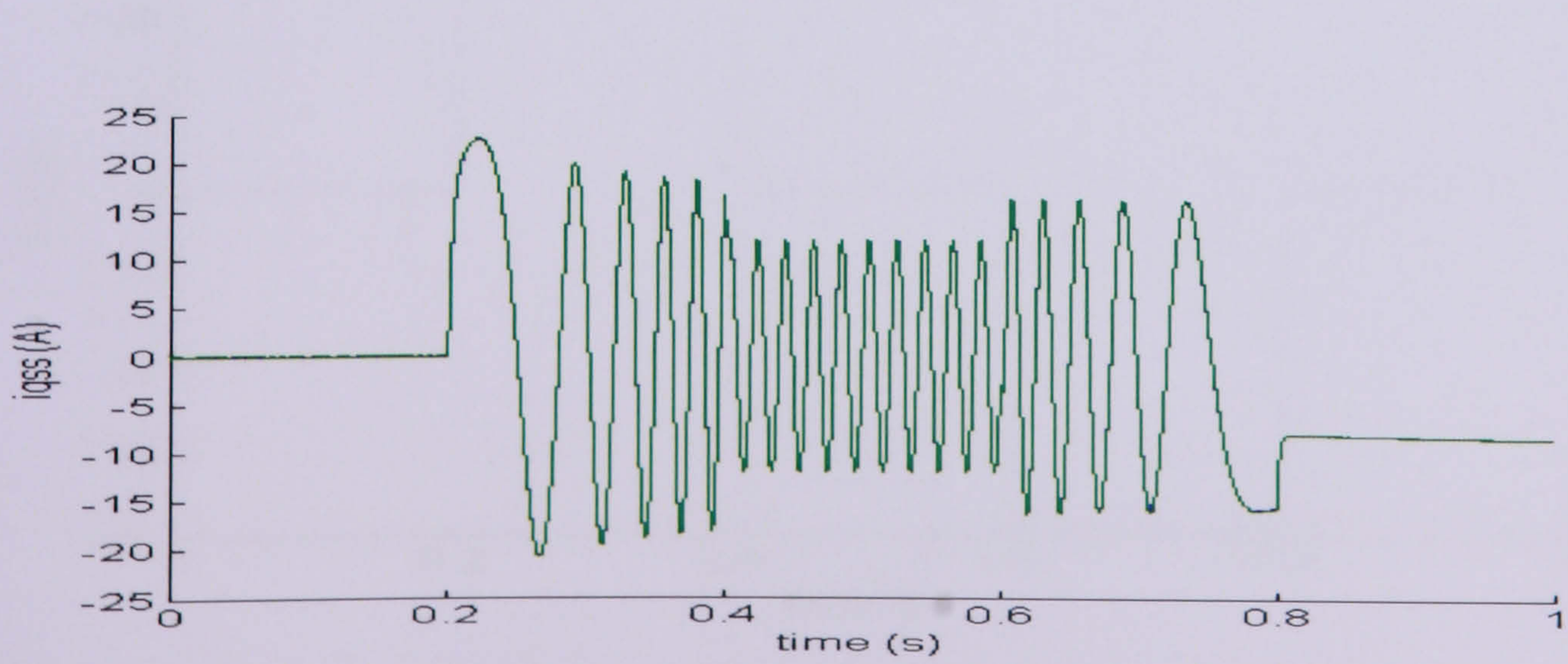
(a)

(b)

Figure 7.13: (a) Voltage applied to the IM during 1s simulation (b) Voltage applied to the IM during the steady state operation

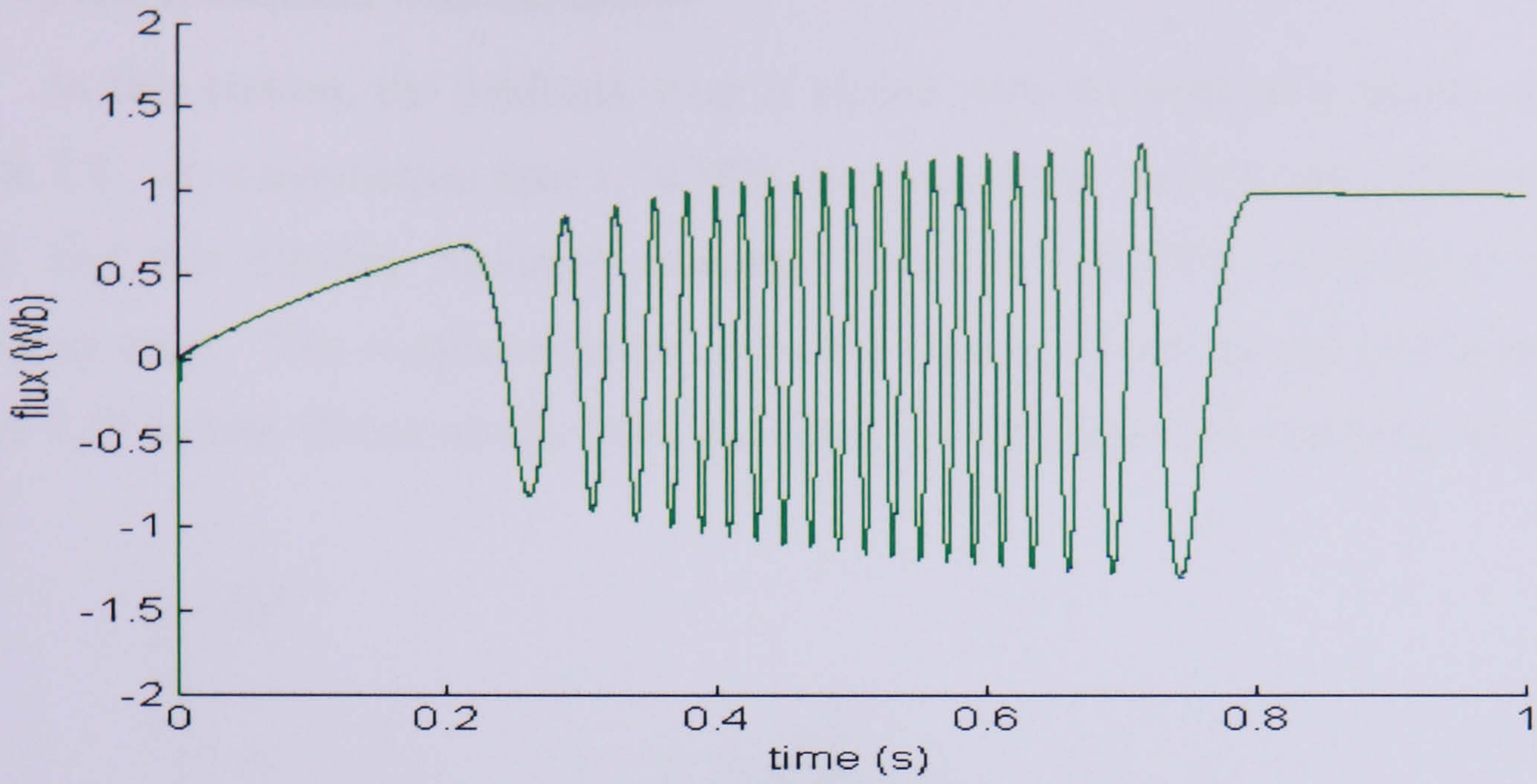


(a)

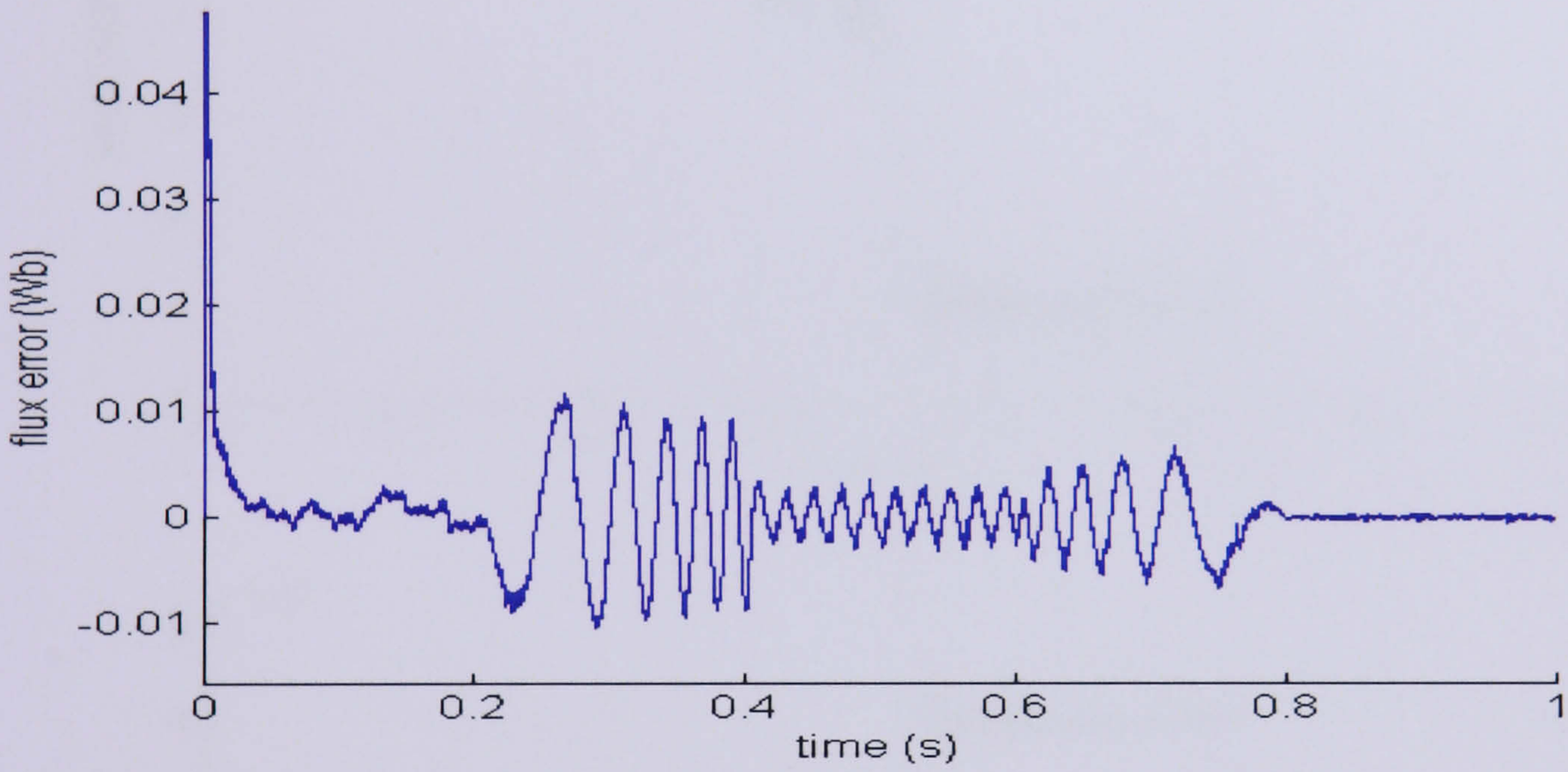


(b)

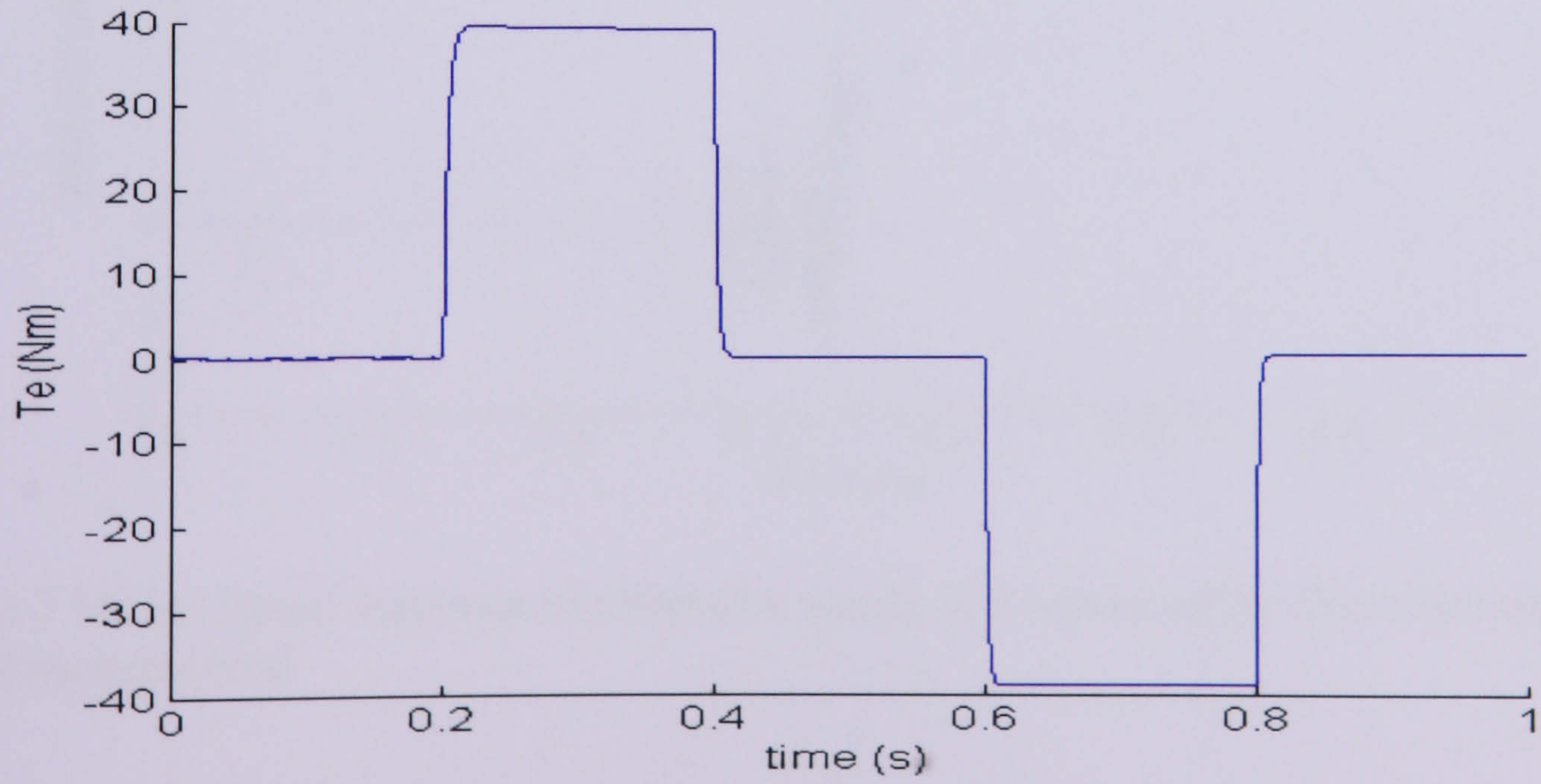
Figure 7.14: (a) The d-axis stator current of actual and estimated (b) The q-axis stator current of actual and estimated



(a)



(b)

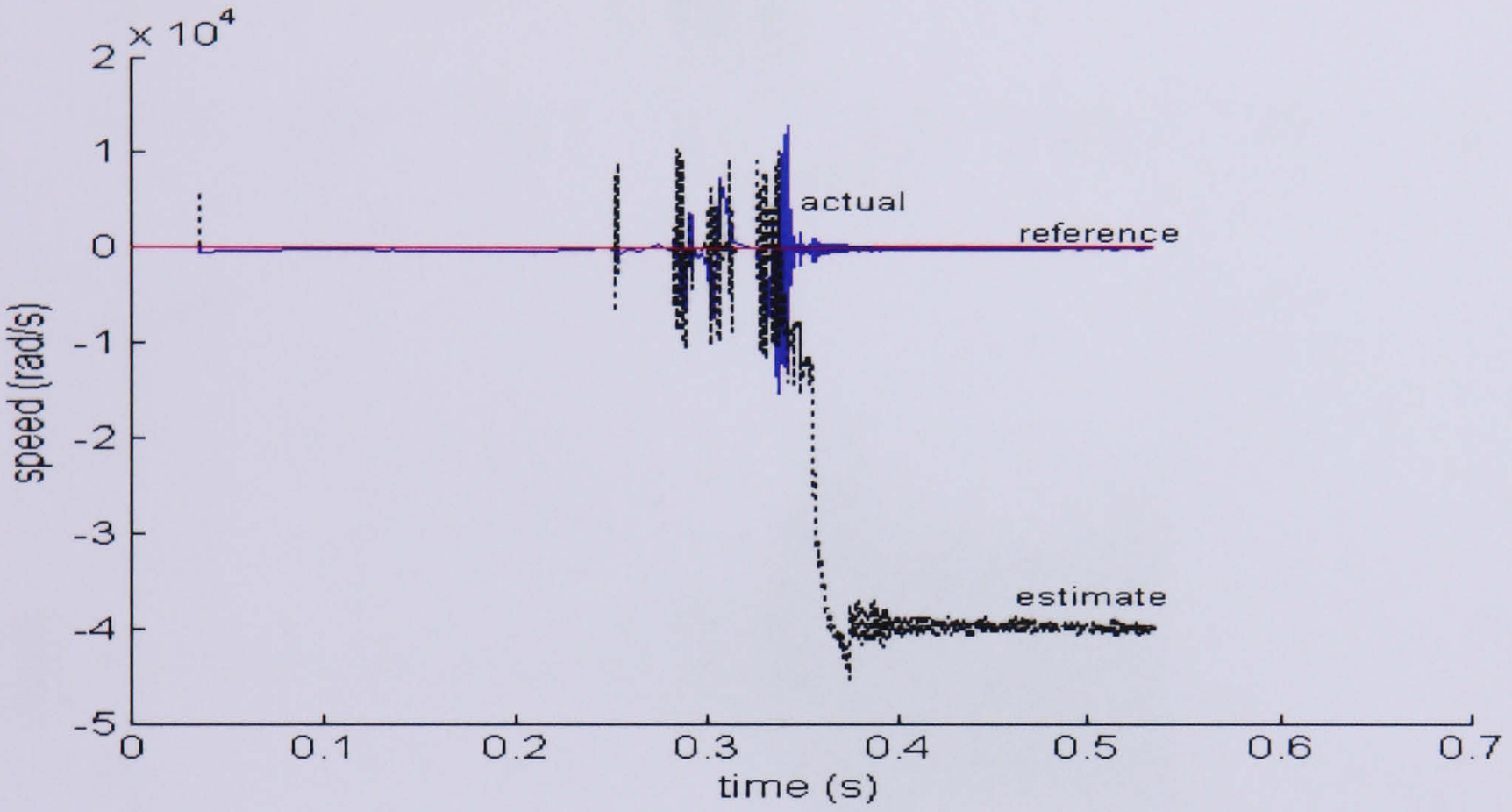


(c)

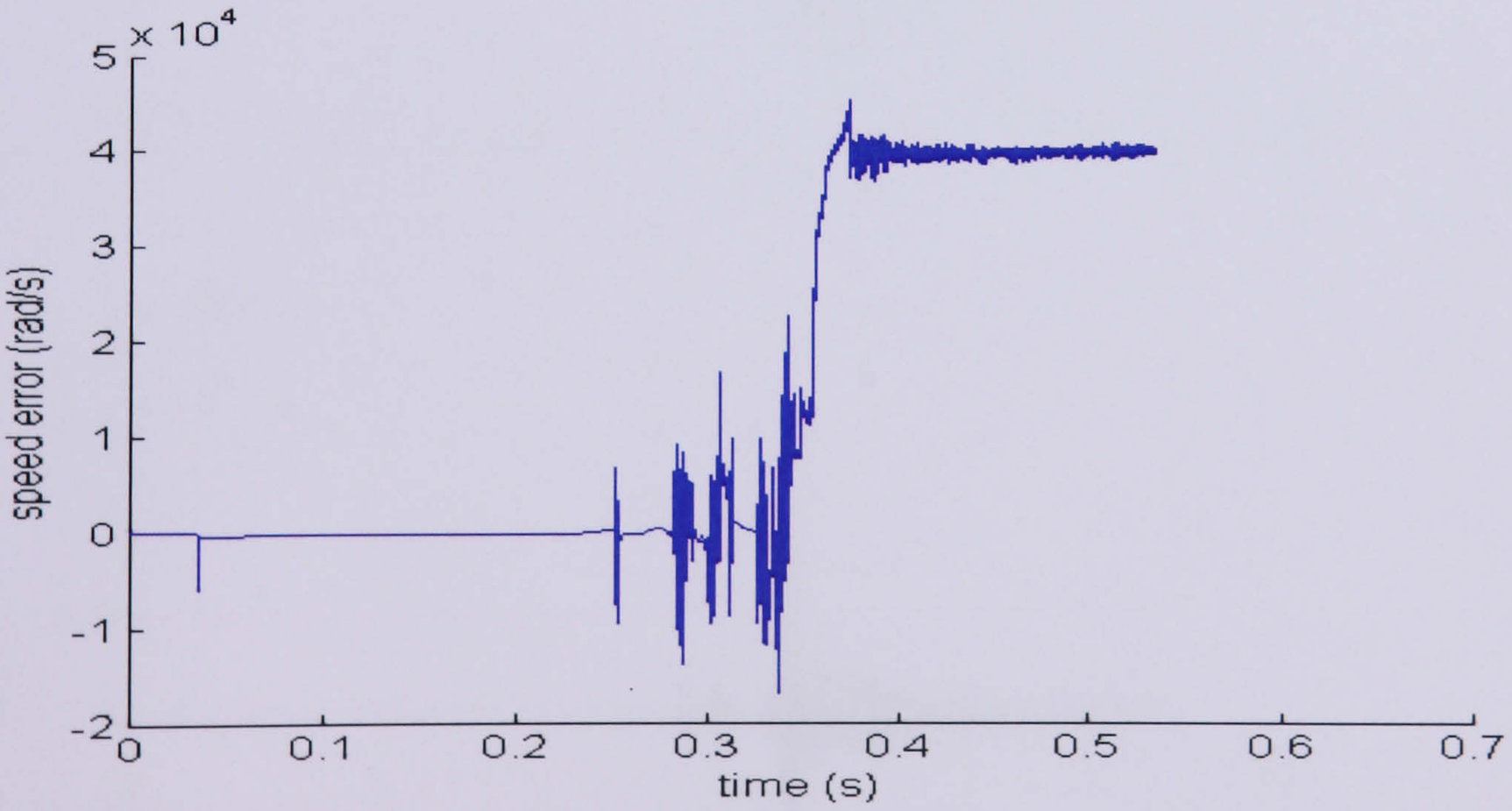
Figure 7.15: (a) The d-axis rotor flux of actual and estimated during the operation module (b) The d-axis rotor flux of actual and estimated error (c) Developed motor torque

7.5.2 Speed estimation without sensor

In this section, the feedback loop is closed with the estimated speed, as shown in Figure 7.4. At a simulation time t_s=0.383s, the simulation starts to run rather slower than usual; and this running becomes extremely slow at t_s=0.533s resulting in a memory allocation error. The results obtained from the simulation are shown in Figure 7.16 and Figure 7.17 below. These results indicated there are difficulties in choosing the controllers gains.

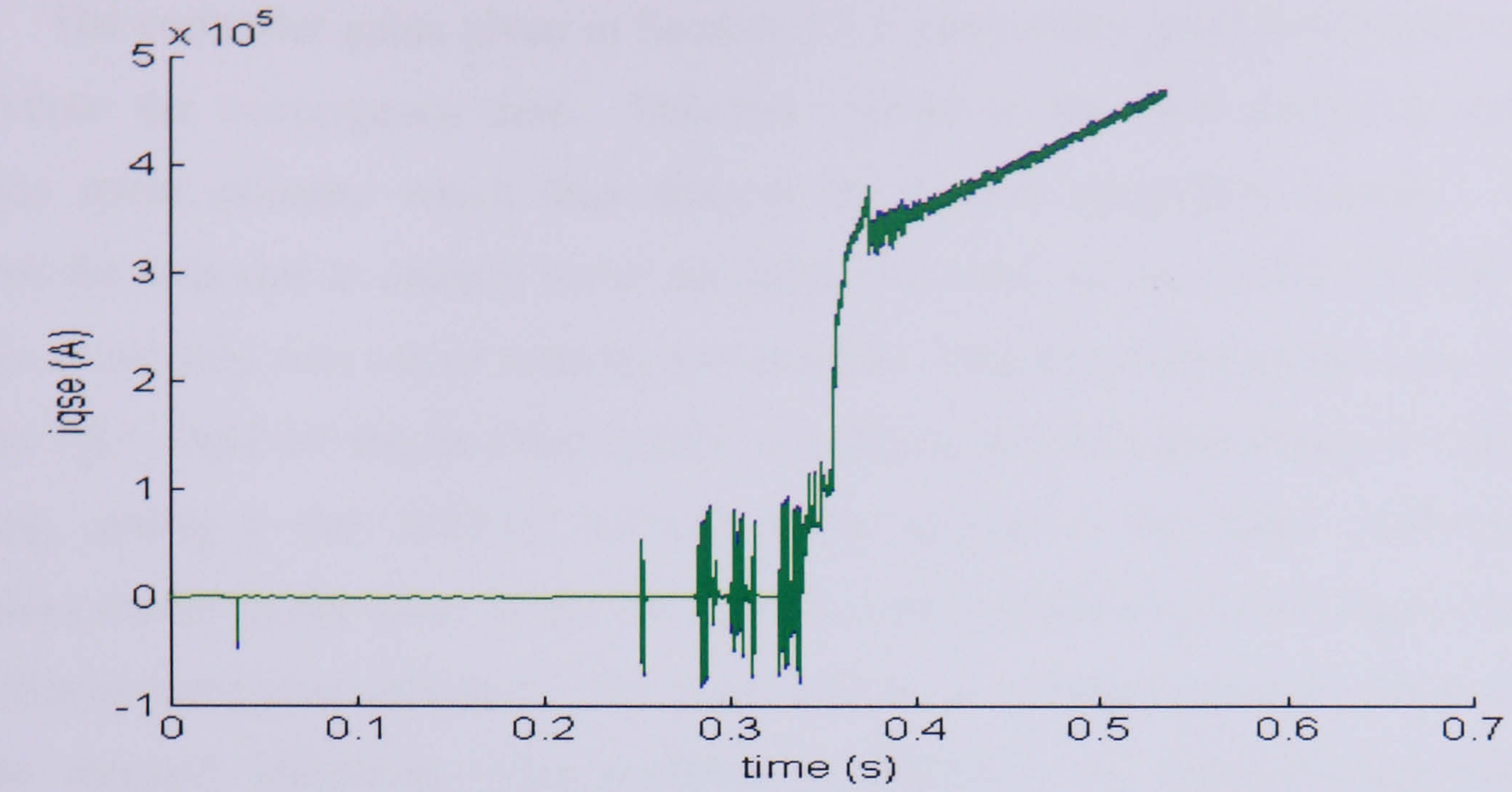


(a)

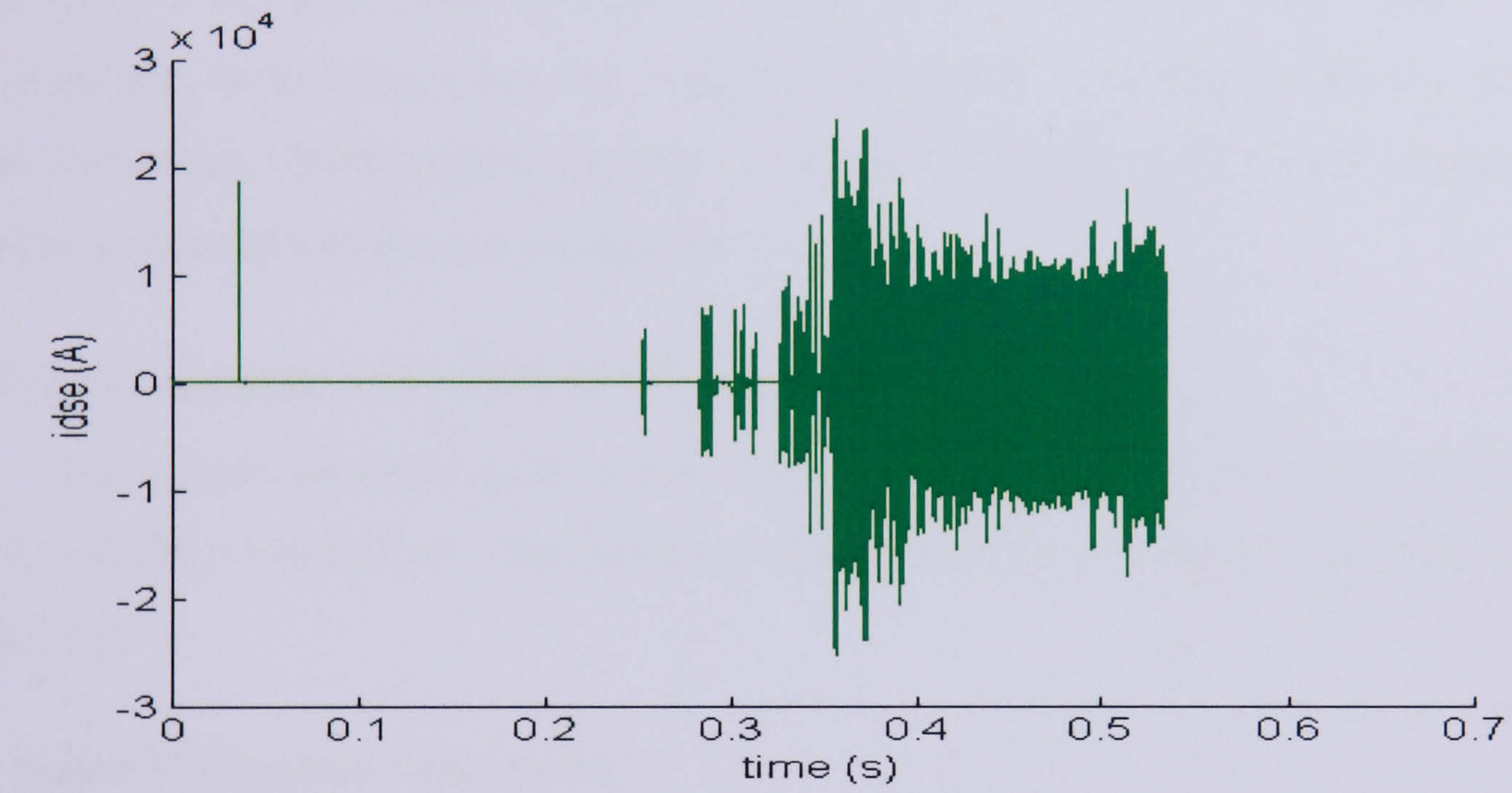


(b)

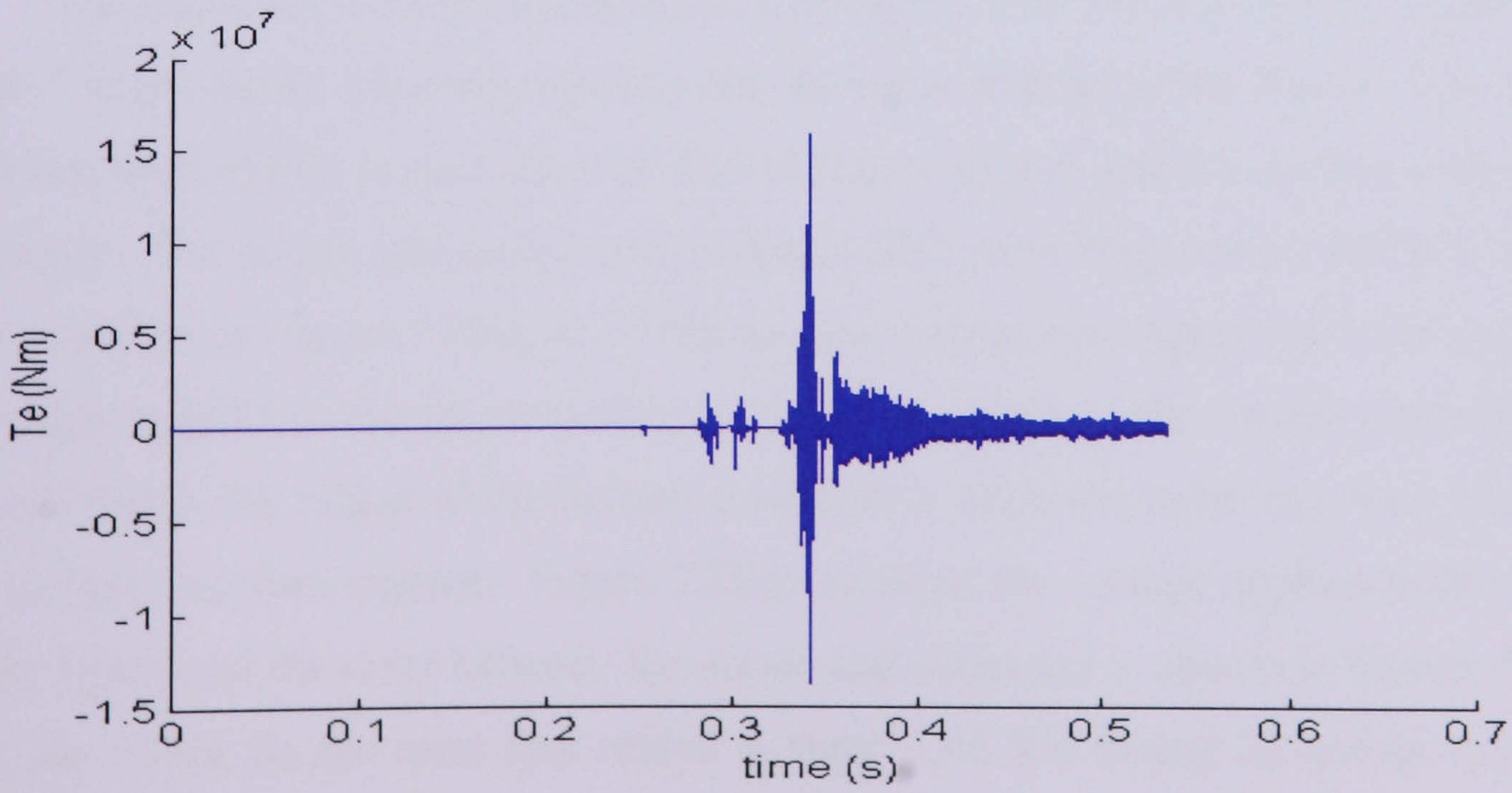
Figure 7.16: (a) Speed response of reference, actual and estimated (b) Error between actual and estimated speed



(a)



(b)



(c)

Figure 7.17: (a) The iqse response (b) The idse response (c) Developed motor torque

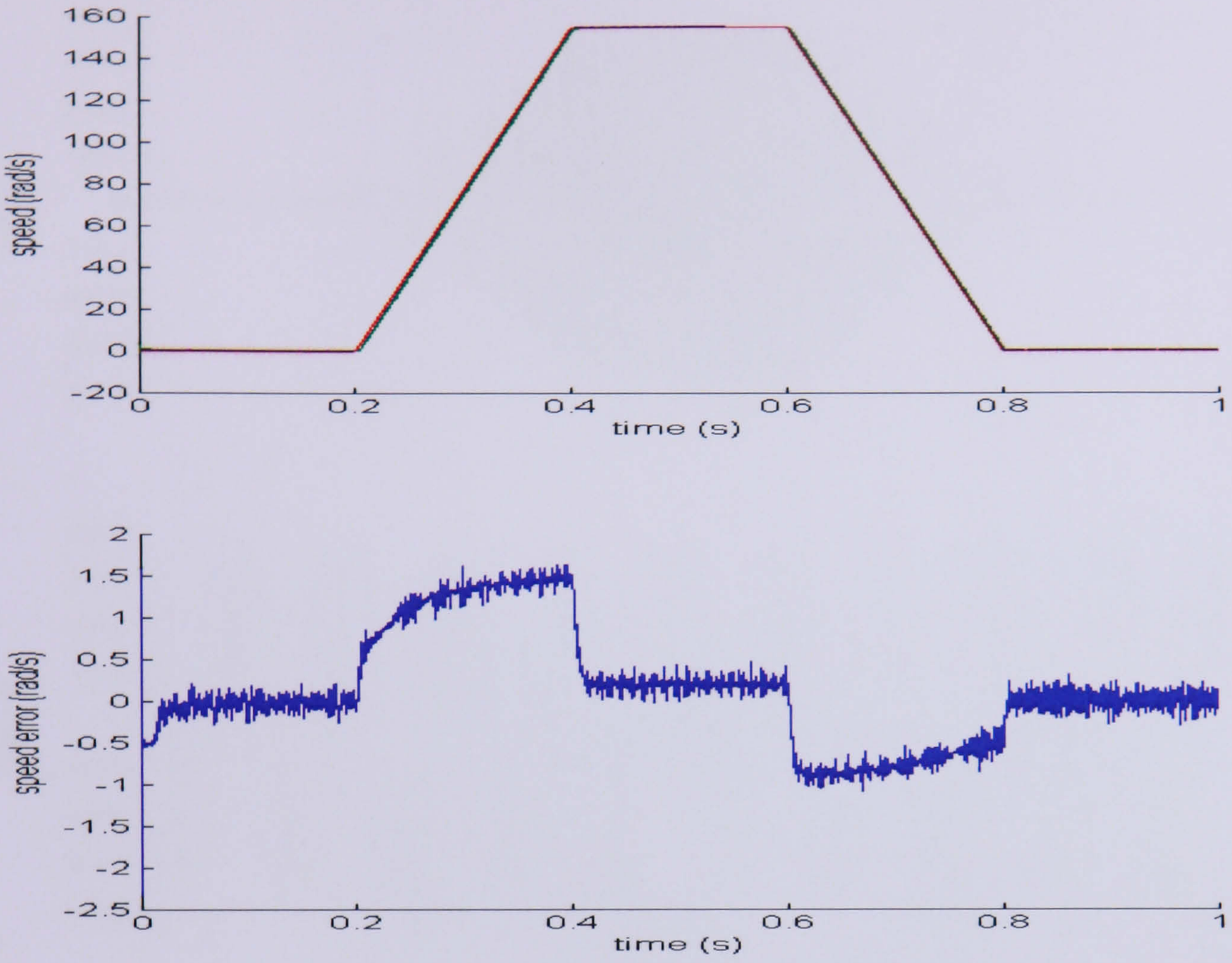
The controller gains given in Section 7.5.1 give a very good convergence response but reduce the convergence time. This has resulted in the EKF diverging and gives a tenuous speed estimate which then disturbs the Matlab integration routine. This then reduces the step size to attempt better estimated accuracy, so the routine starts to run very slowly, eventually runs out of memory and crashes. This long computation time is required for the EKF itself for the Jacobian matrix calculation and this contributes to much slower running, giving a very difficult scenario when applied to the speed estimation. The slowness results in the delay in the EKF for the speed response and the high overshoot of the estimated response obtained. This high error in the estimated response then contributes to the memory allocation error problem in Matlab. To overcome the problem, a compromise is required when choosing the gain of the controller. The complexity of the EKF algorithm in VC drives has also been discussed [49]. The author describes the Digital Signal Processing (DSP) used to implement the EKF as not being powerful enough and that therefore a reduced sample rate is required.

7.6 Vector Control with SET B PI controller

The speed controller gain is now QK_p=8 QK_i=20 and current controller DK_p=200 DK_i=3 and QK_p=100 QK_i=3. The results obtained from the simulation are shown in Figures 7.18 to 7.22.

7.6.1 Speed Estimation with sensor

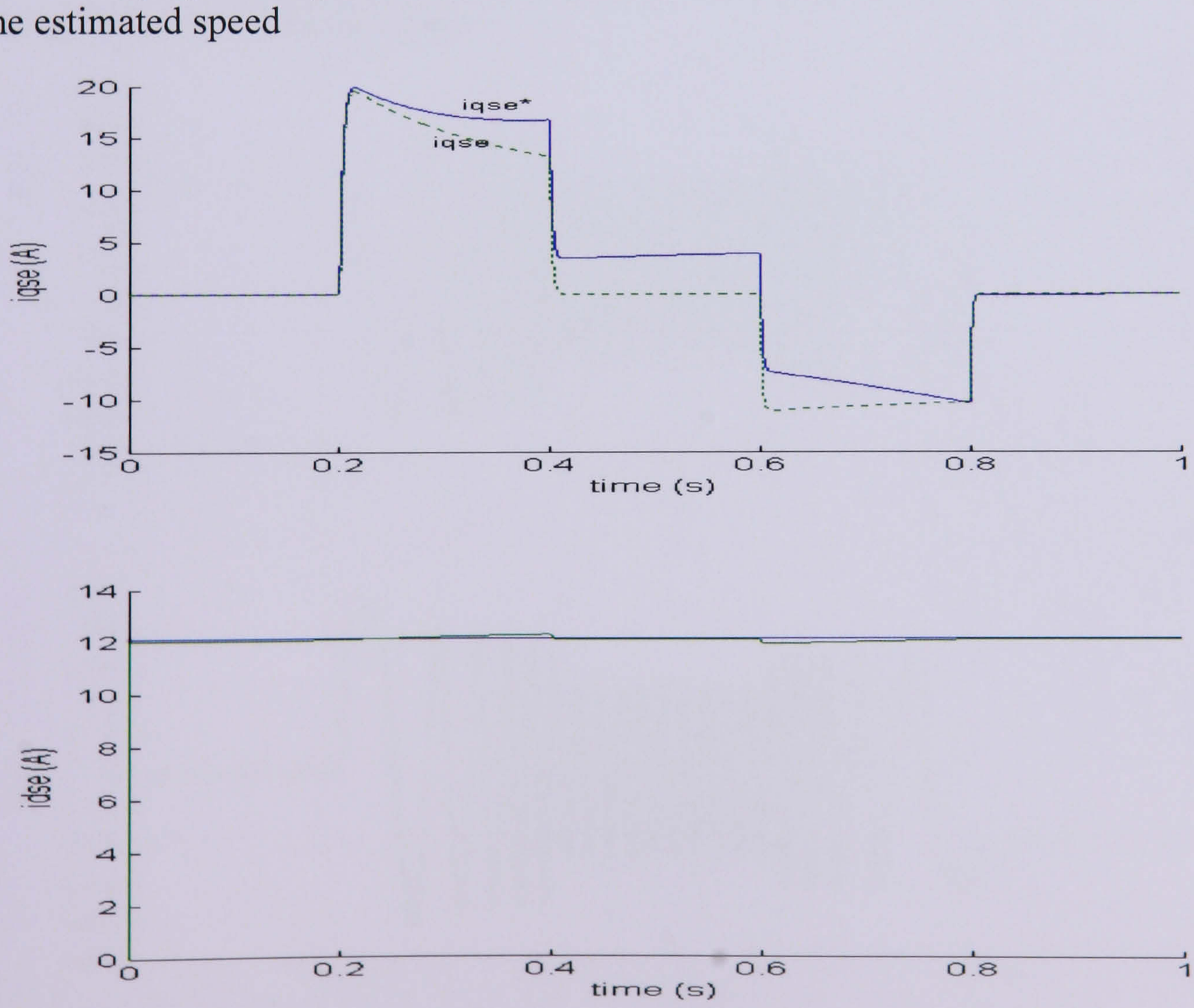
The reference, actual and estimated rotor speed obtained is shown in Figure 7.18(a). Figure 7.18(b) shows the error between the reference and estimated speed. The estimated speed has been shown to perform well and indicates a good speed tracking performance of the system. The torque producing current and the flux producing current in the synchronous frame is shown in Figure 7.19(a, b). Without power electronics application such as inverter, the voltage applied to the motor is directly from the output of the current controller. At 0-0.2s and 0.8-1s the output of the current controller is expected to be zero thus should yield zero voltage and zero current. Figure 7.20(a, b) show the voltage applied to the IM. The current figure and the error between the actual and estimated is shown in Figure 7.21(a, b). In Figure 7.22(a, b), the rotor flux settled at about 1.45 Wb during the steady state and the errors between actual and estimated are shown. The torque developed is shown in Figure 7.22(c). These results shows that the estimator still behave reasonably apart from i_{qse}.



(a)

(b)

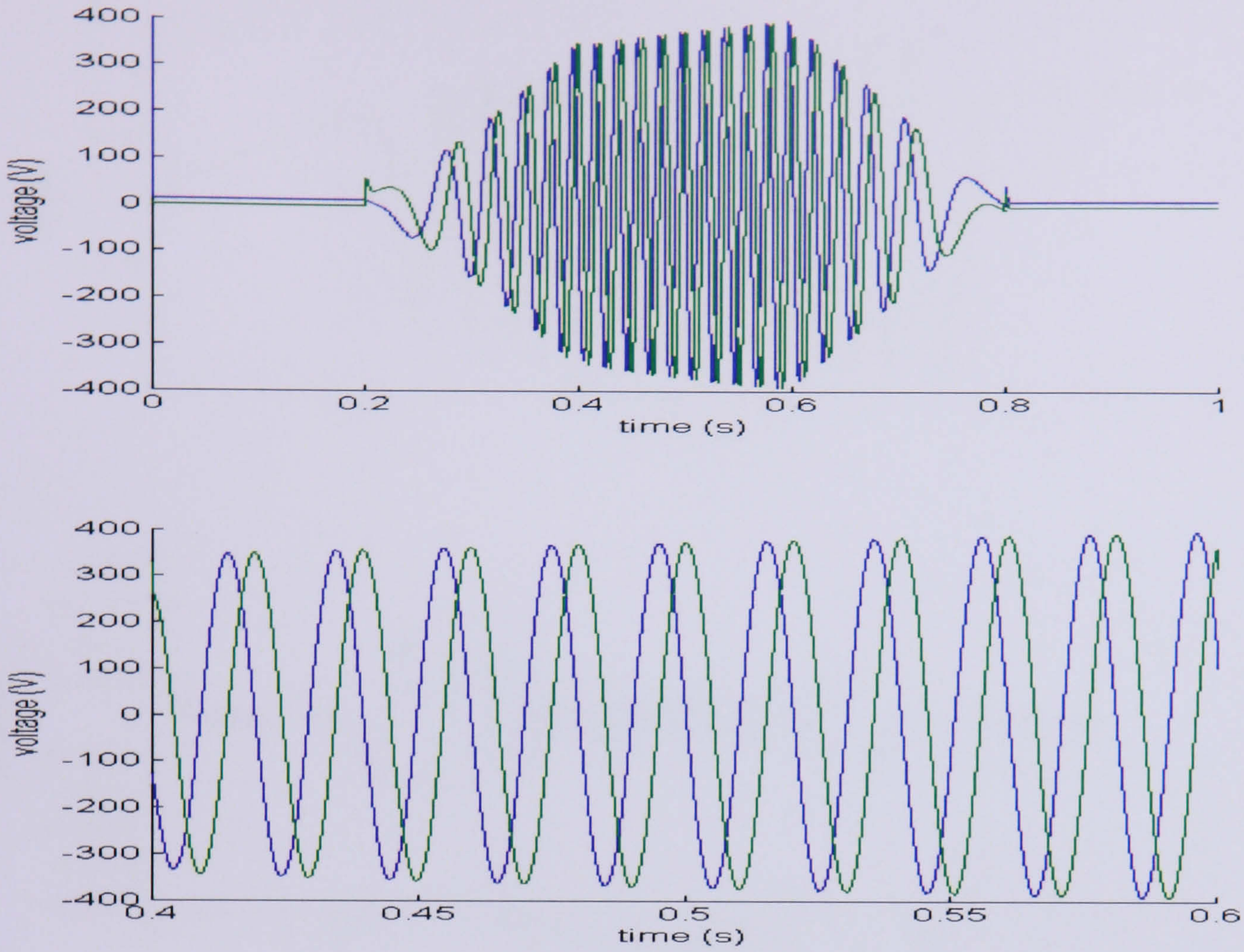
Figure 7.18: (a) Speed response of reference, actual and estimated (b) Error between actual and the estimated speed



(a)

(b)

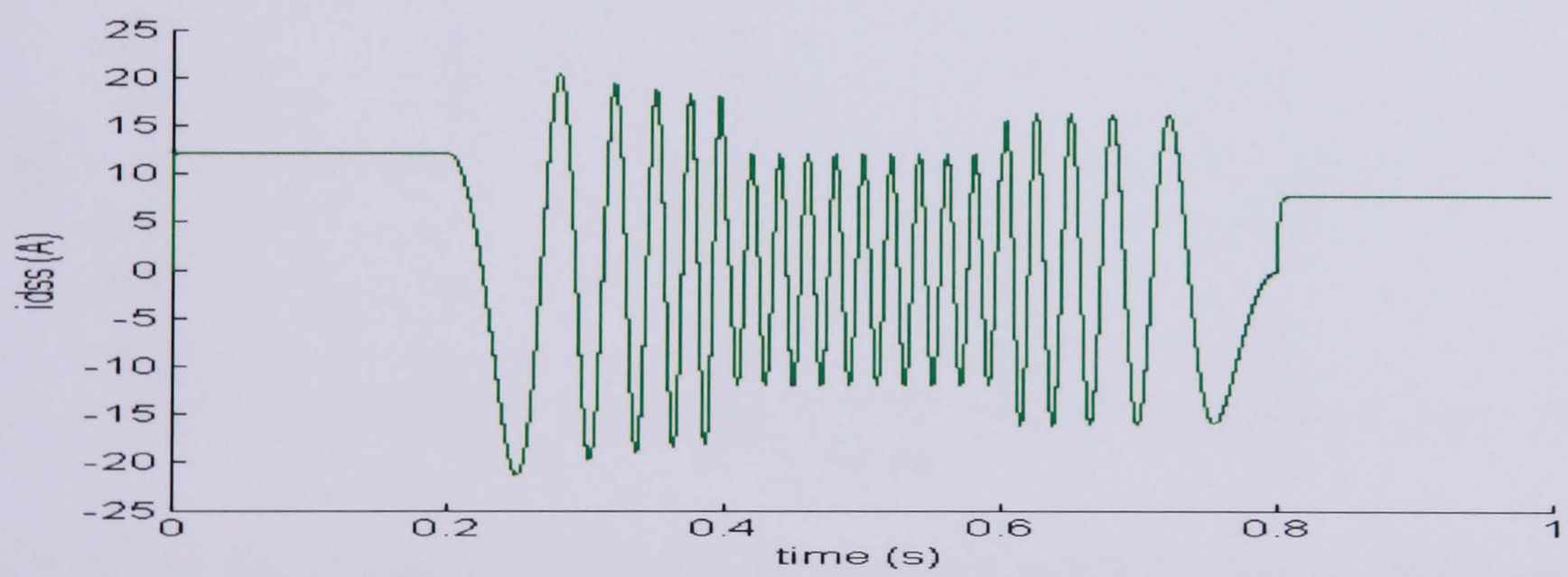
Figure 7.19: (a) The iqse response (b) The idse response



(a)

(b)

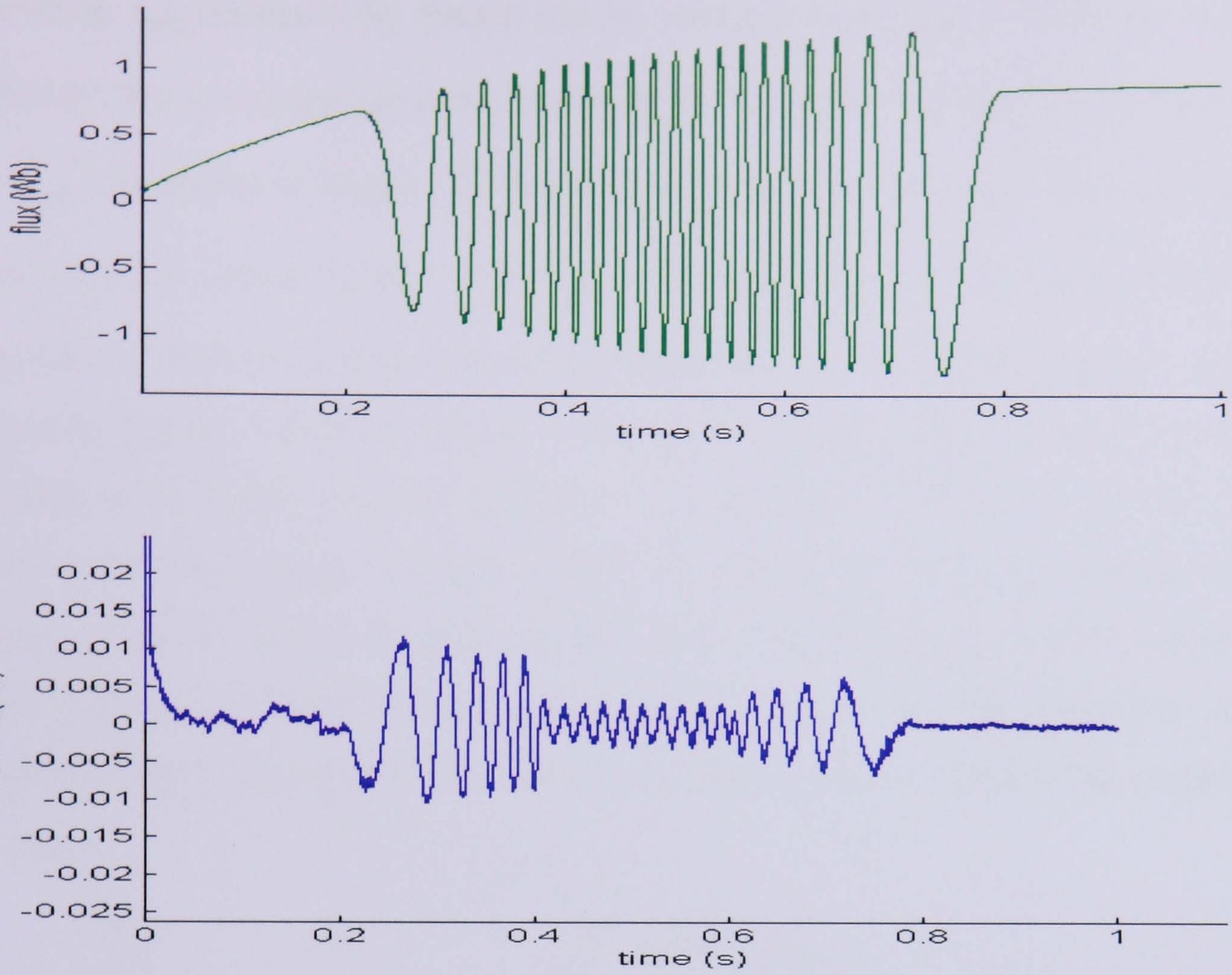
Figure 7.20: (a) Voltage applied to the IM during 1s simulation (b) Voltage applied to the IM during the steady state operation



(a)

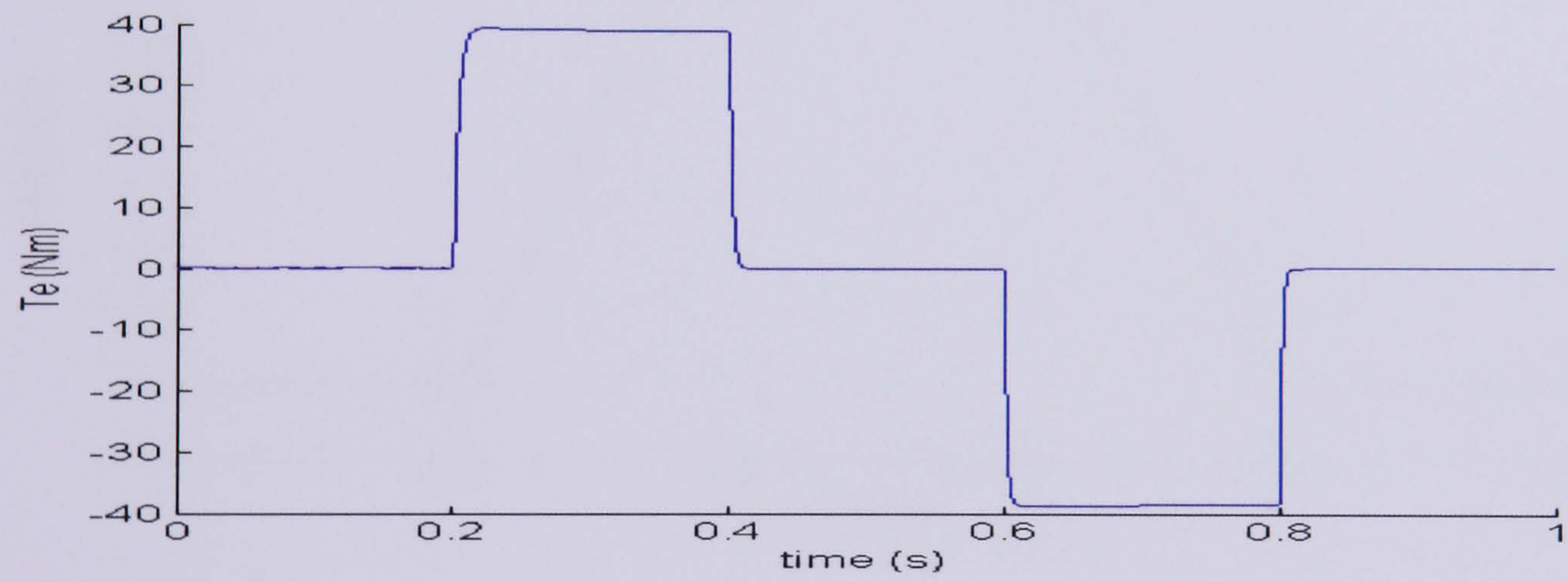
(b)

Figure 7.21: (a) The d-axis stator current of actual and estimated (b) The q-axis stator current of actual and estimated



(a)

(b)



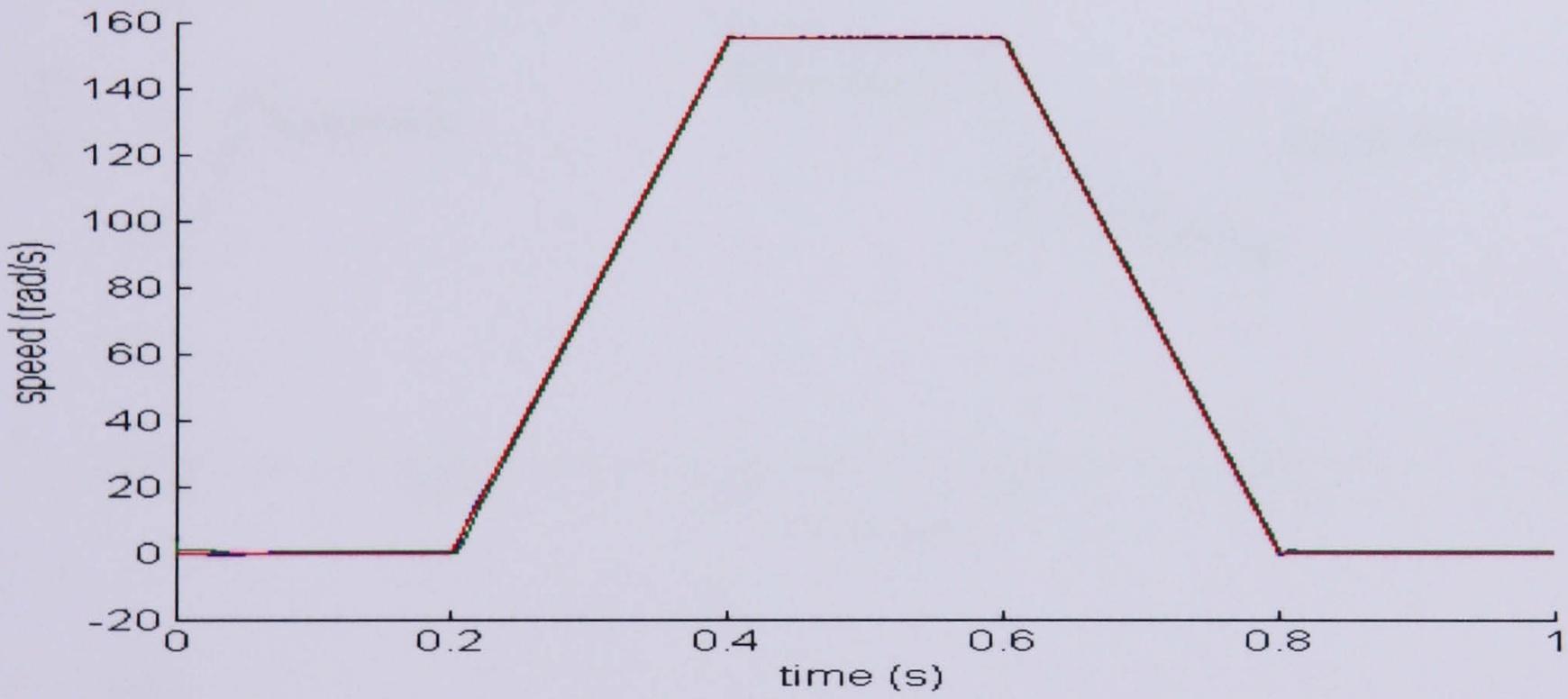
(c)

Figure 7.22: (a) The d-axis rotor flux of actual and estimated during the operation module (b) Error between actual and estimated d axis rotor flux (c) Developed motor torque

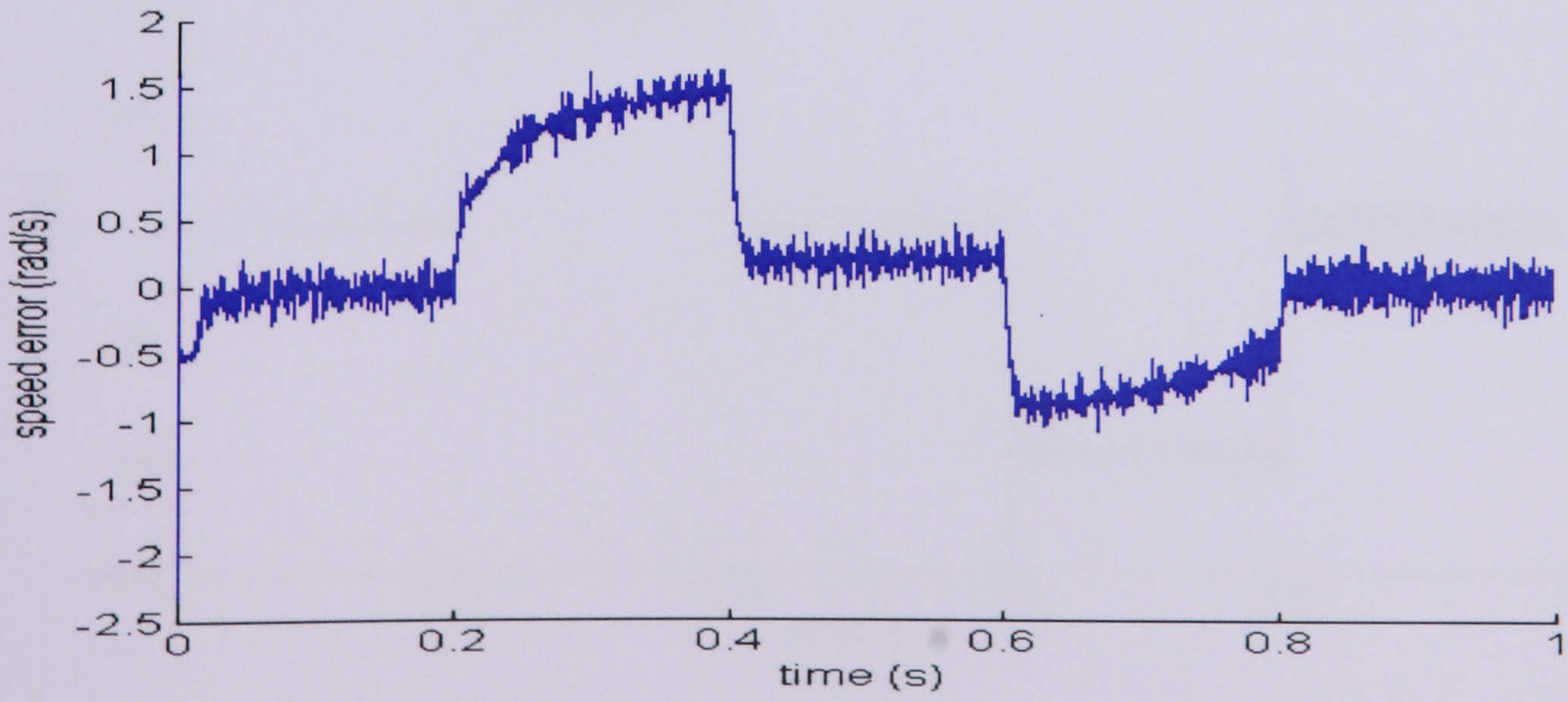
7.6.2 Speed Estimation without sensor

In this section, closing the speed loop with the EKF output is investigated with reduced gain as in SET B. The results are presented in Figures 7.23-7.26. The closed loop estimator produces a smaller mean squared error than the open loop estimator shown in the previous section. The mean squared speed error obtained is 0.7856. The reference, actual and estimated speeds are shown in Figure 7.23 (a) while the error between the actual and estimated speed is shown in Figure 7.23 (b). The speed of the motor reaches 155 rad/s at 0.4s. Figure 7.24(a, b) shows the reference and actual currents i_{dse} and i_{qse} respectively.

By holding i_{dse} constant, the torque can be changed using i_{qse}^{*}. Once the motor flux is established, the motor performs the acceleration, transient and steady state operation. The trace i_{qse}^{*} as shown in Figure 7.24(b) is expected to have the same form as the developed torque shown in Figure 7.24(c). The i_{qse} and torque traces show that a sudden change in the reference produces a high overshoot and this affects the speed response significantly. Comparing Figure 7.24(b) to Figure 7.19 (a), the closed loop system is now stable but noticeably more 'noisy' than the case with a speed sensor. Without the inverter, the voltage applied to the IM is shown in Figure 7.25(a, b). The d-axis and q-axis current in stationary reference frame is shown in Figure 7.26(a, b). The maximum current shown in Figure 7.26(b) is about 16A RMS during the transient and 12A during the steady state which when compared to the q current in the synchronous reference frame, verifies that proper VC of IM does occur.

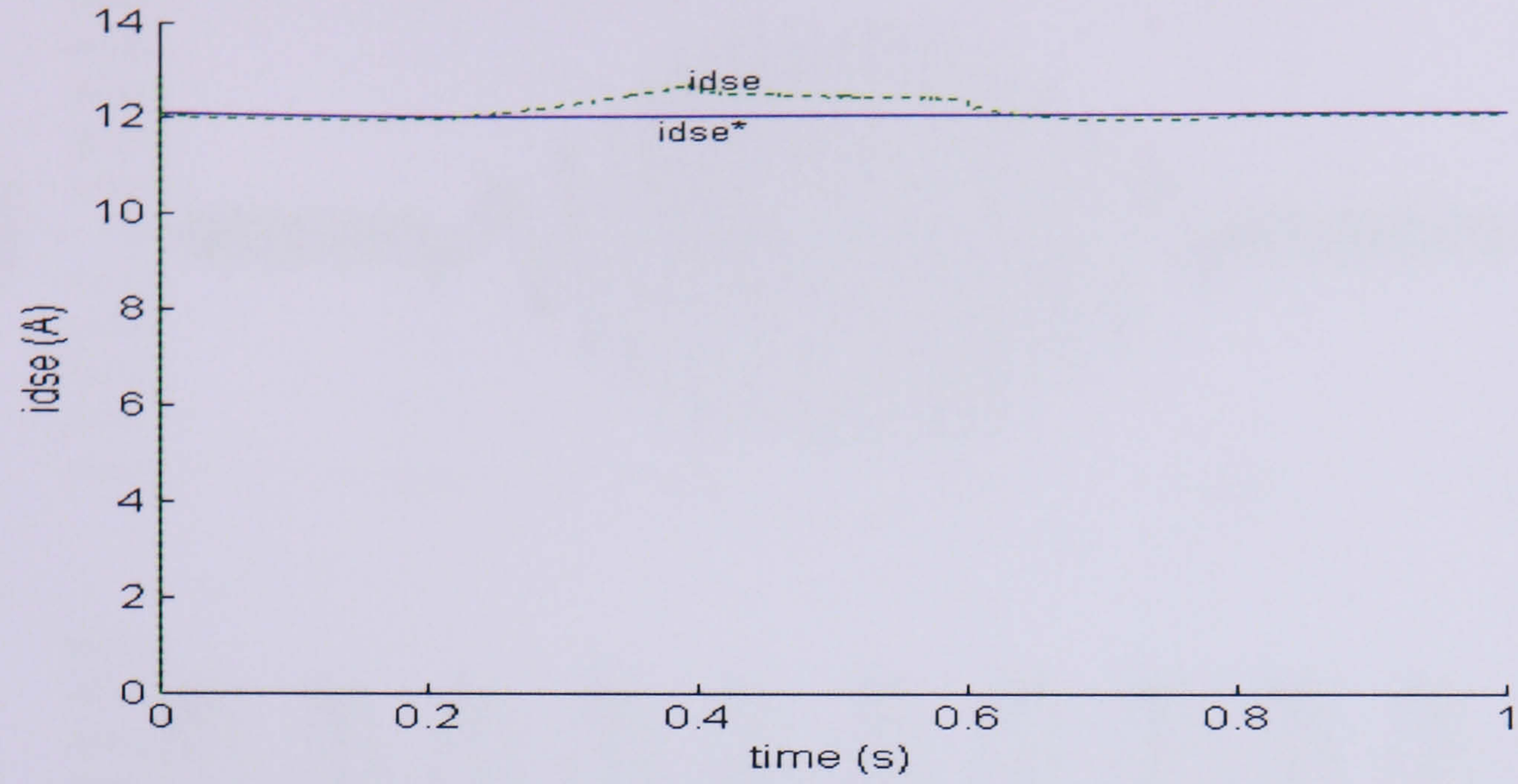


(a)

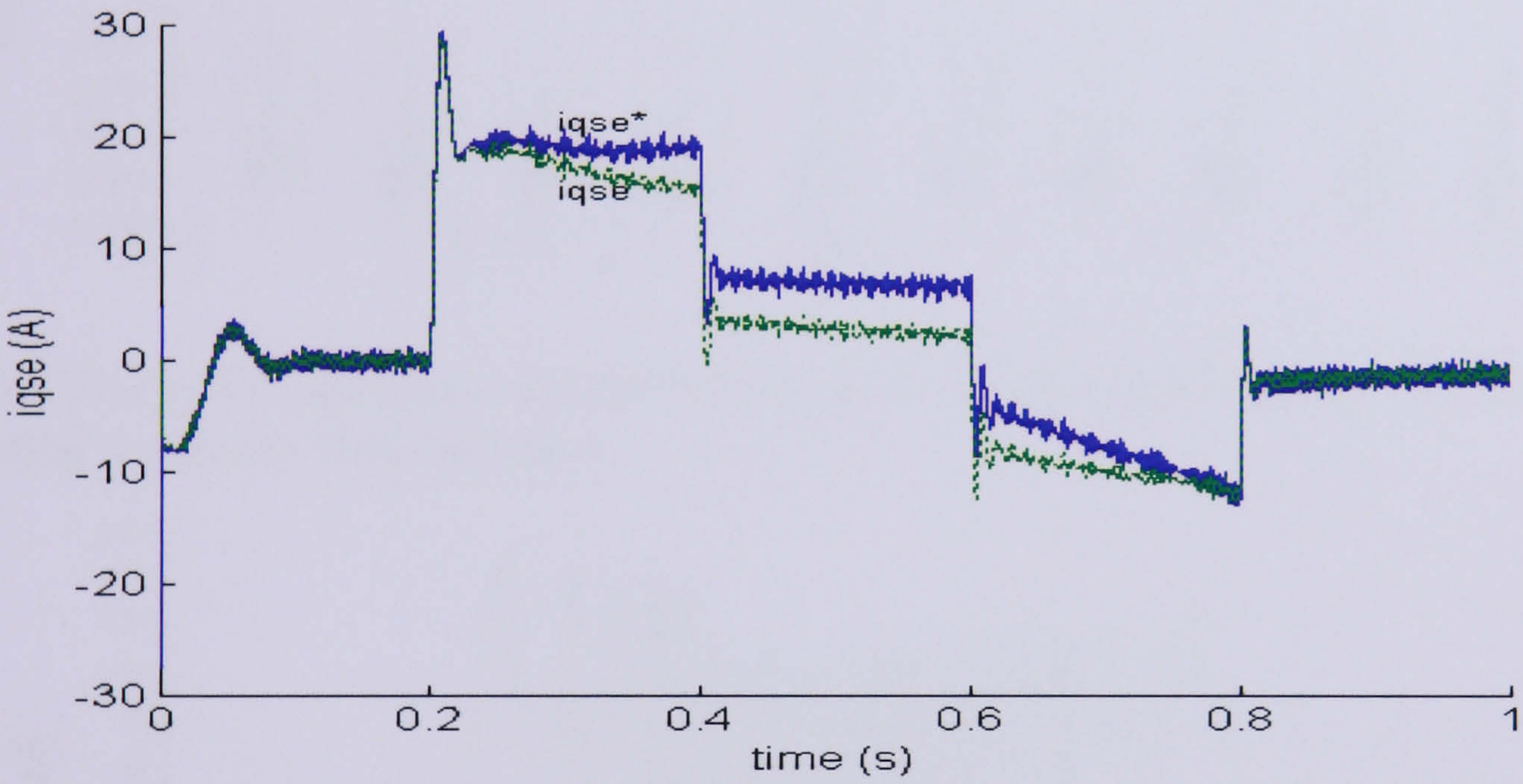


(b)

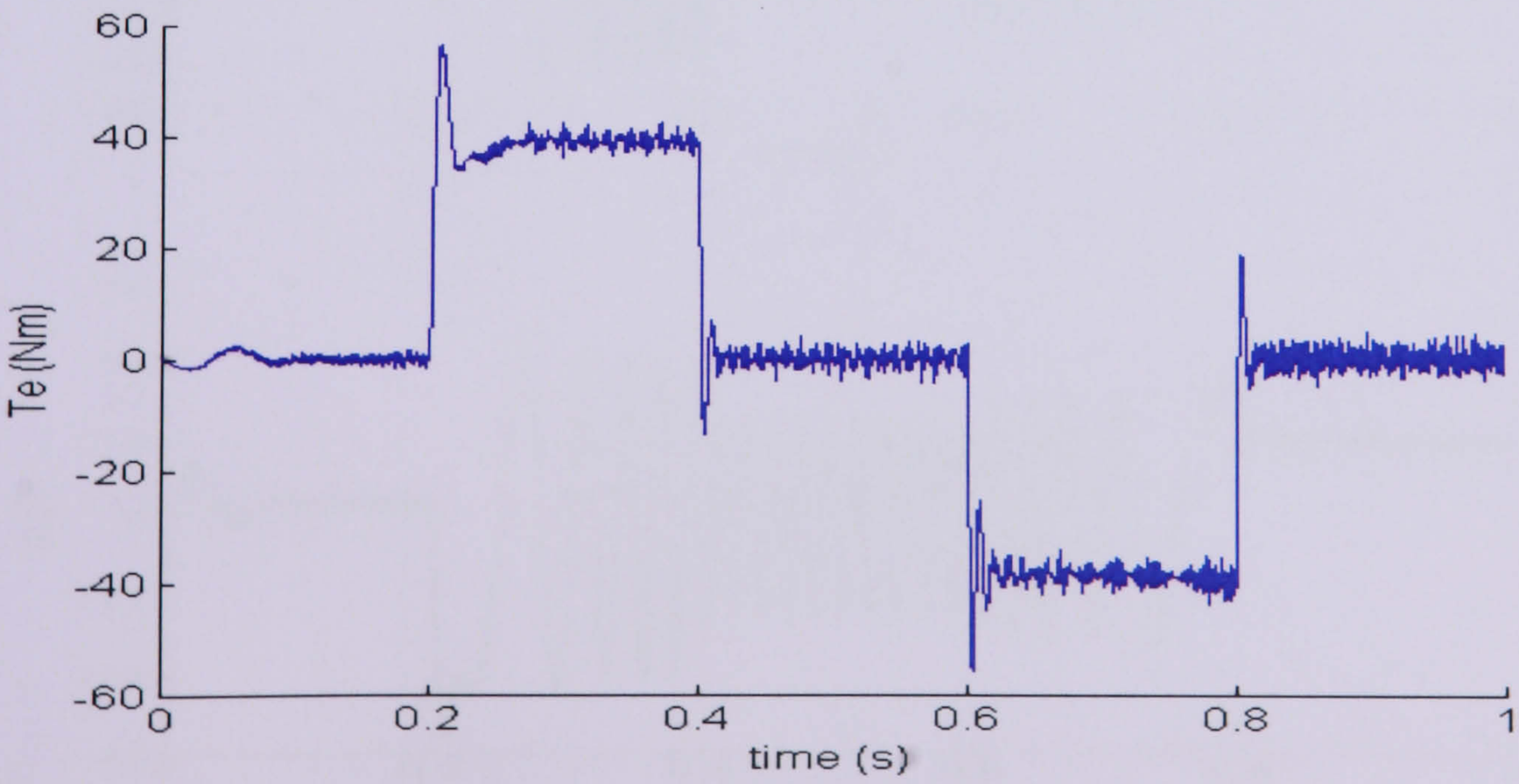
Figure 7.23: (a) Reference (red), estimated (green) and actual (blue) rotor speed (b) Error between actual and the estimated speed



(a)

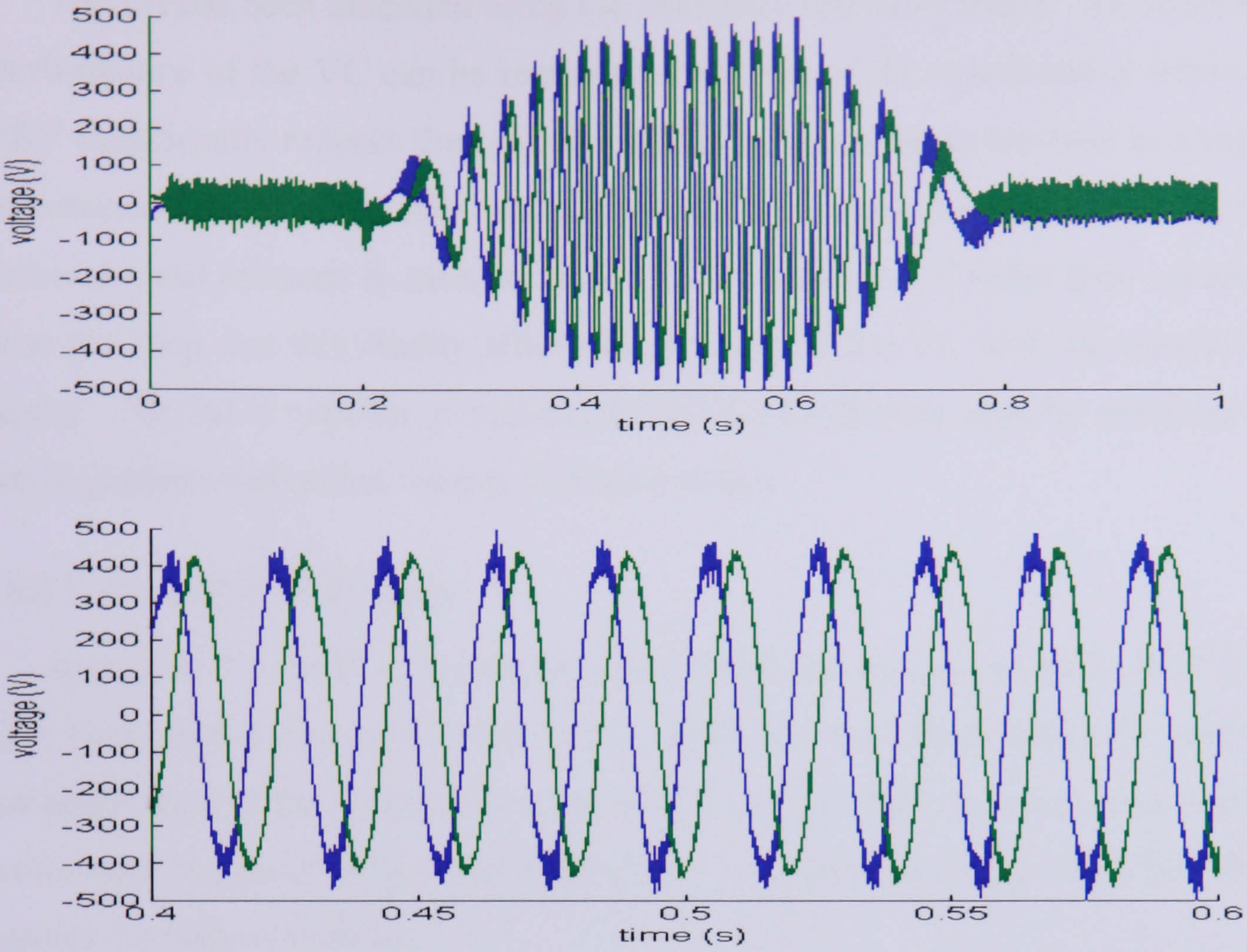


(b)



(c)

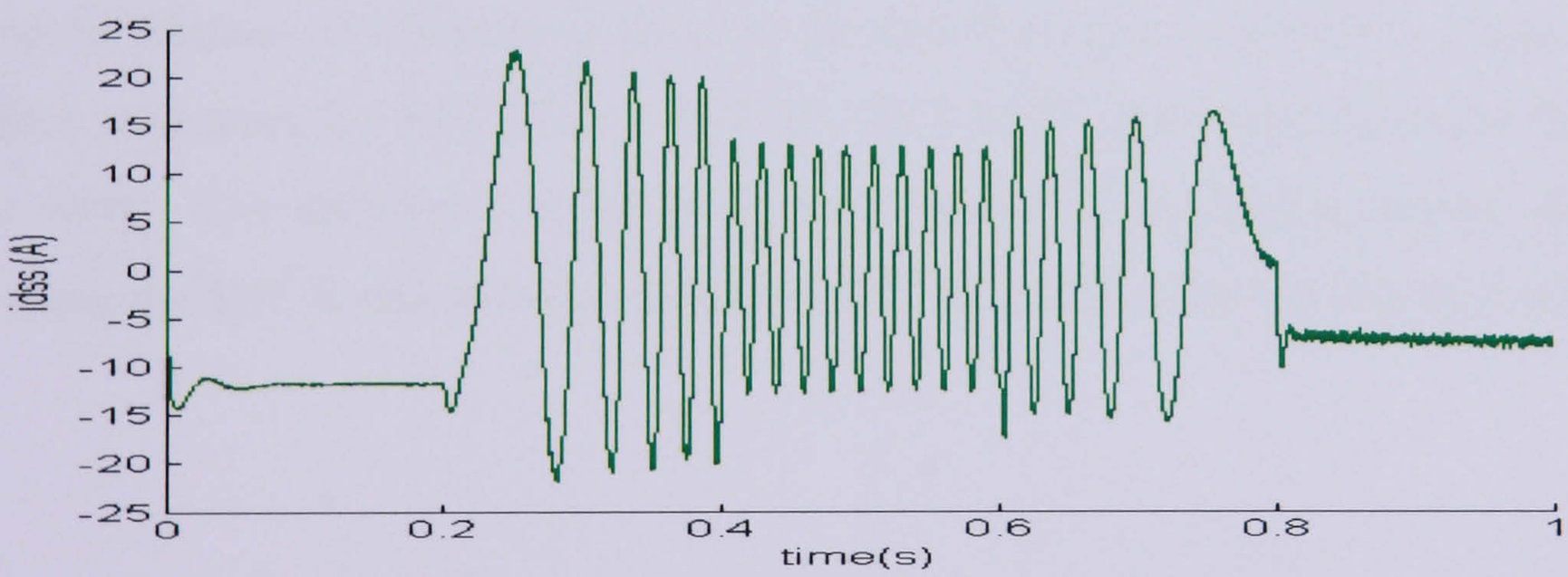
Figure 7.24: (a) The reference and estimated flux producing current (b) The reference and estimated torque producing current (c) Developed motor torque



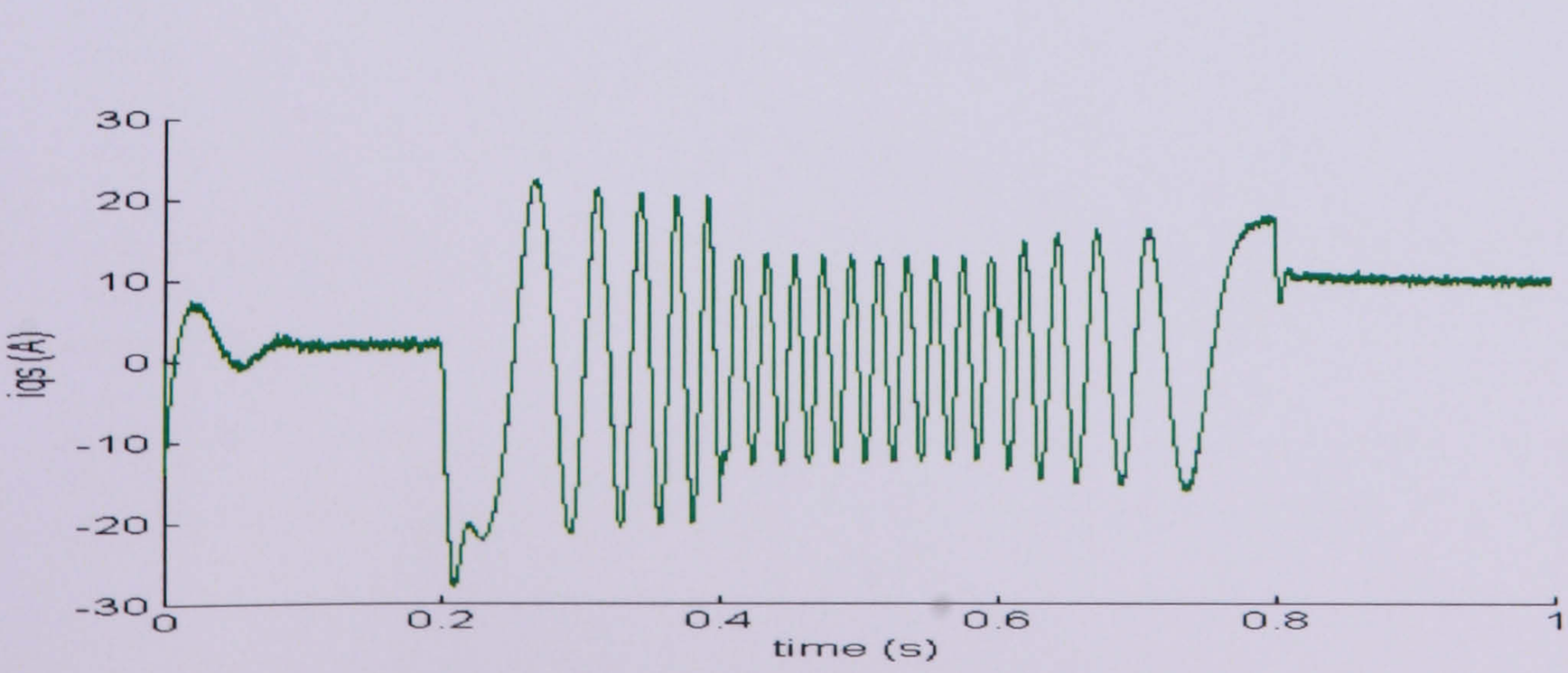
(a)

(b)

Figure 7.25: (a) Voltage applied to the IM during 1s simulation (b) Voltage applied to the IM during the steady state operation



(a)



(b)

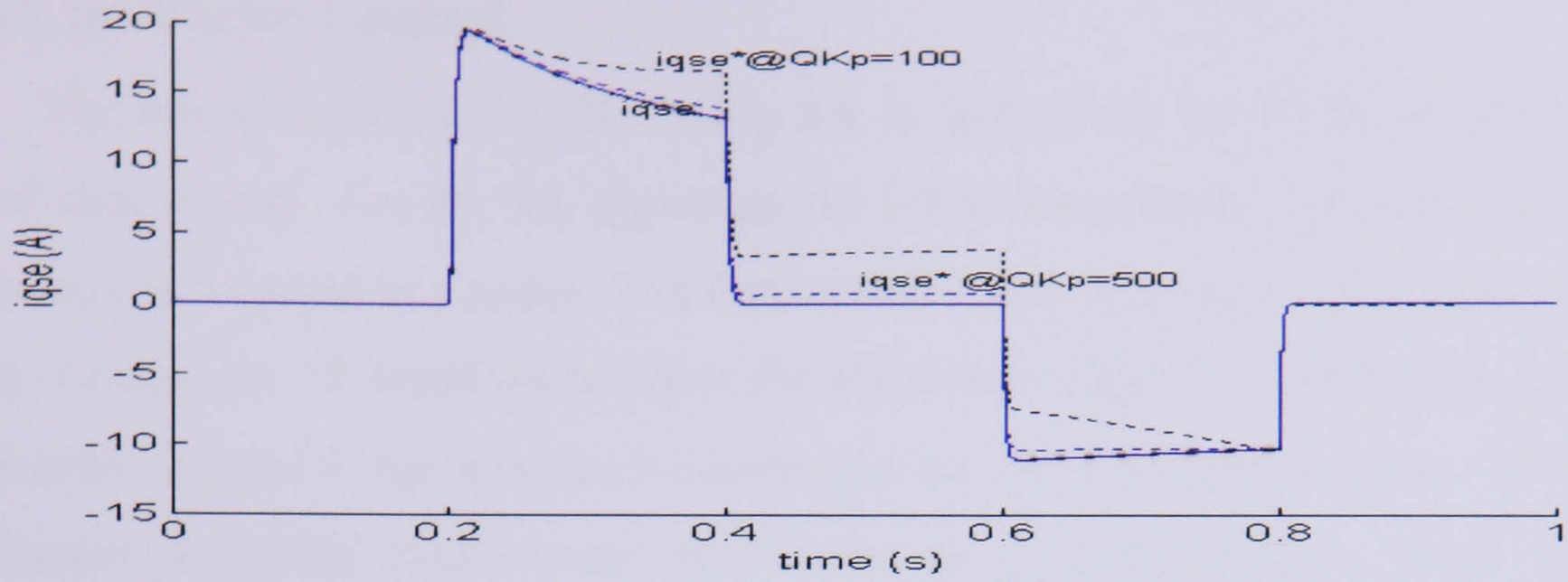
Figure 7.26: (a) The d-axis stator current of actual and estimated (b) The q-axis stator current of actual and estimated

The IM has been modelled using the stationary reference frame. As claimed by [98], the performance of the VC can be improved. The use of the synchronous frame model in the EKF significantly reduces the estimation error and requires no increase in computational effort compared to the commonly used stationary frame implementation. The error between the reference and estimate in each controller can be improved if better gain values are used to close the loop, but this clearly affects the stability of the VC with the closed loop EKF estimator. The IM is capable of running at the speed reference and the estimator has been shown to perform well either with or without a sensor.

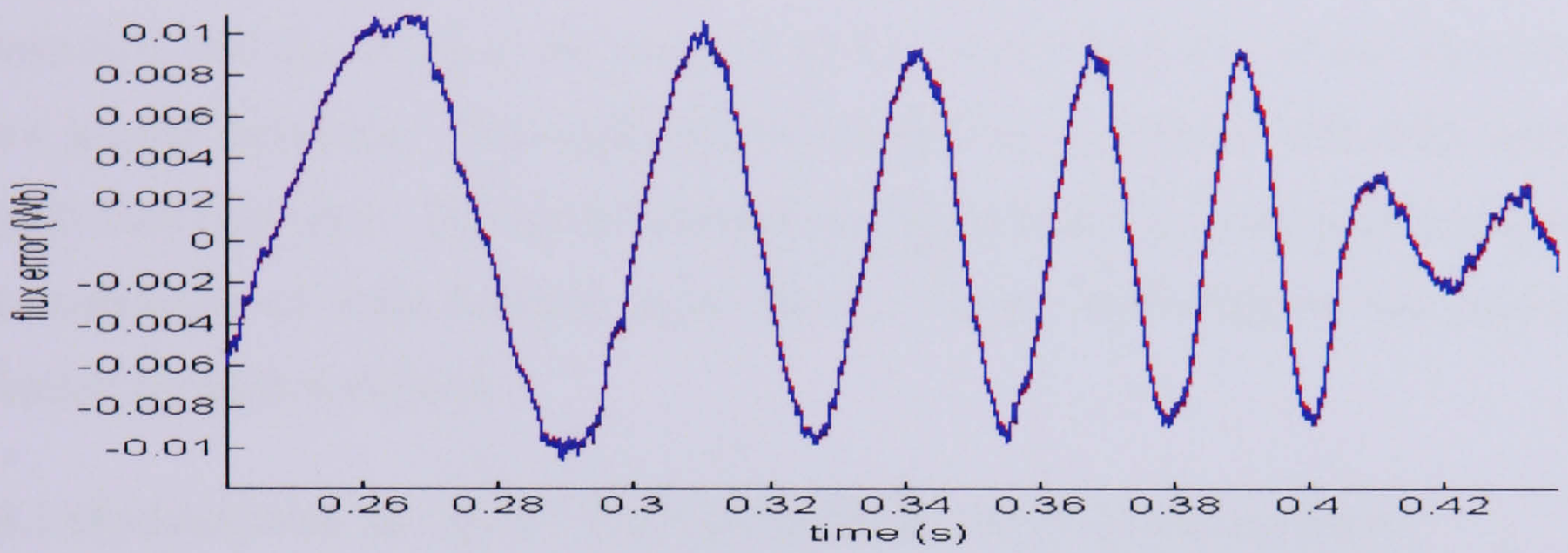
7.7 SET A or SET B PI gain

In section 7.3 the PI controller has been tuned and two sets of gains, SET A and SET B, have been used in the application of VC. Both sets have been tested for sensorless and sensor application of the estimator in sections 7.5 and 7.6. SET A can be used with the open loop estimator but gives difficulties in the closed loop estimator case, while SET B has been successful for both conditions.

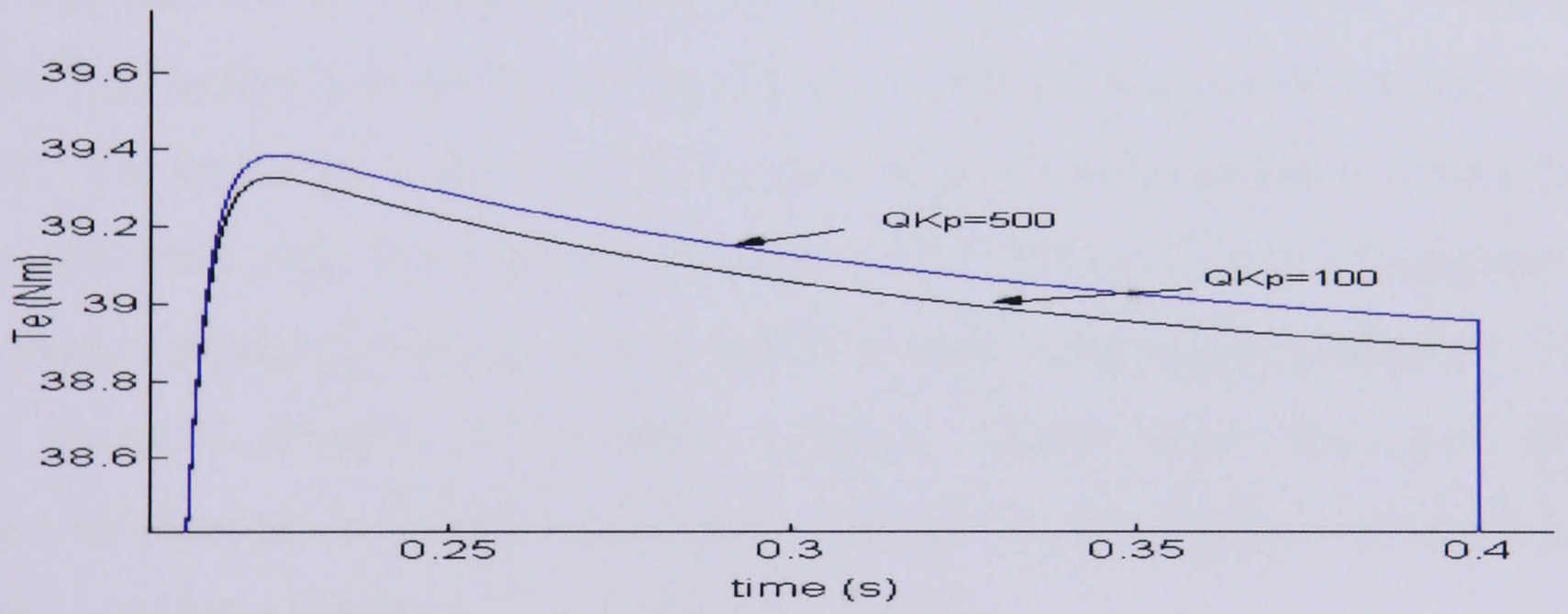
In the case of the open loop estimator, it makes little difference whether the PI gain controller for SET A and SET B is used (500 for SET A and 100 for SET B), provided the same speed demand of 155 rad/s is used for the specified speed controller. Figure 7.27(a-d) compares responses for SET A and SET B. SET B PI gains will therefore be used for further work. The difference in QK_p has made the torque producing current iqse* higher when using the SET B and remains the same with the actual torque producing current.



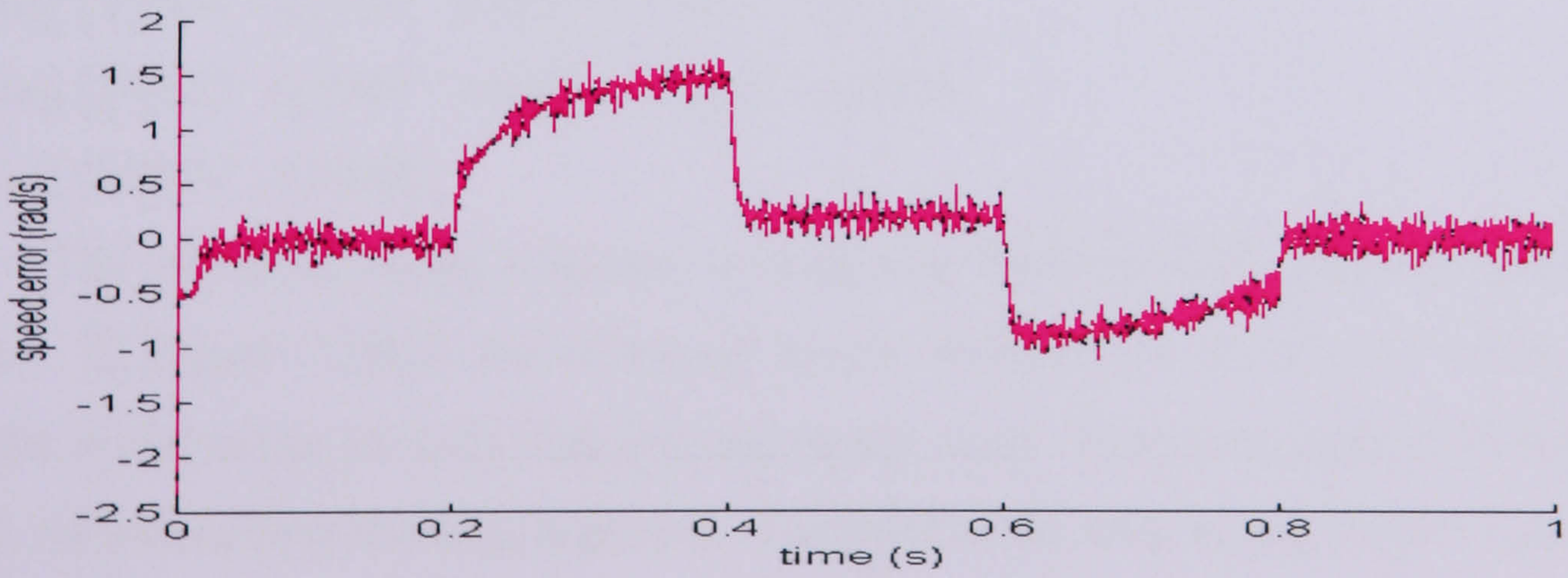
(a)



(b)



(c)



(d)

Figure 7.27: (a) Torque producing current (b) Flux error (c) Developed torque (d) Speed error between the actual and estimated speed

7.8 SA on Vector Control

The optimisation of the EKF using SA is carried out for 101% of rated speed and 70% of rated speed. For the SA algorithm the initial temperature has been set to 80. The temperature is lowered in a series, $T=\alpha T$ where the value of α used is 0.9. The temperature is held constant for 15 iterations to allow the algorithm to reach a steady state point and the temperature reduced if the solution is unchanged for 10 consecutives times. At the start of optimisation the initial solutions are initialised over a suitable range. Based on Trial and Error tuning, the configuration space for the solutions is set to [0-0.01] for the rest of the parameters and Q_{55} is set in the range of [0-1]. This bigger Q_{55} allows the rotor speed to have greater influence. The optimisation is done on the system with both open loop and closed loop estimator. The speed response is evaluated over a 1 sec time window; therefore the evaluation of total iteration takes about 5 hours depending on the total number of iteration for each condition.

7.8.1 Optimisation on Open Loop Estimator for 70 % of Rated Speed

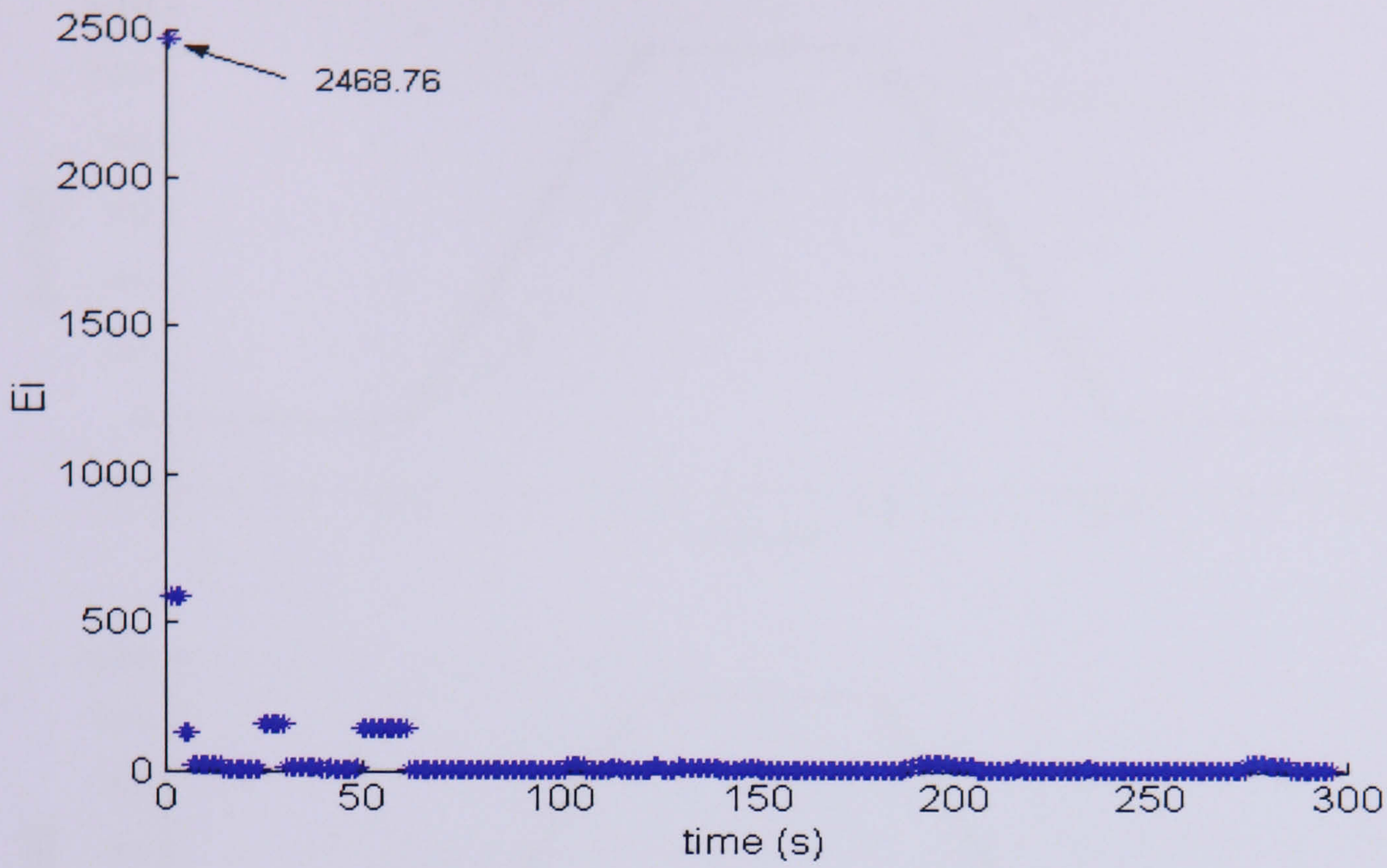
In this test, the SA is used to tune the EKF for the speed demand of 70% of rated speed. The total generation for the tuning process is 296 generations and lasted for 4 hour 56 minutes. The objective functions as a function of generation number is shown in Figure 7.28(a, b). The best objective function obtained is E_i=0.2994 at 172 to 173 generation, the biggest accepted objective function is E_i=2468.76 found during initial generation, while the frequently accepted solution is E_i=1.4687. Figure 7.29(a) shows the speed responses associated with these three objective functions. The best speed response is shown in Figure 7.29(b) and is obtained with the noise covariance matrix:

Q =diag [0.0076 0.0044 0.0035 0.0007 0.8518]

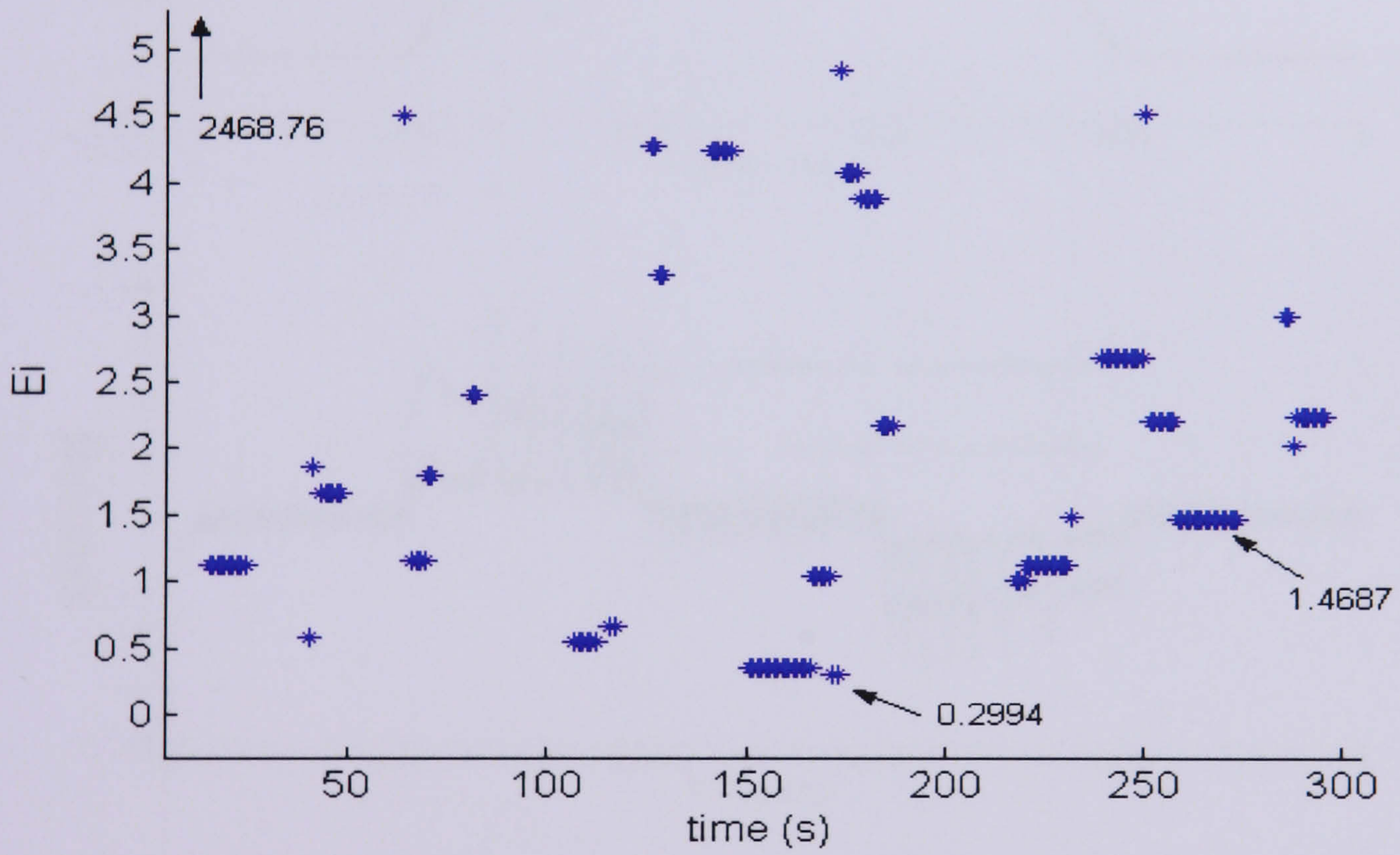
G =diag [0.0023 0.0080 0.0025 0.0.003 0.0097]

R= diag [0.0074 0.0068]

The estimated speed response is compared between the reference and the actual speeds. In Figure 7.29(c), the estimated speed exhibits very small error when compared with the actual motor for both transient and steady state. When compared with the reference speed, the closed loop tracking is poorer. The jitter in the error is due to the PI gain used.

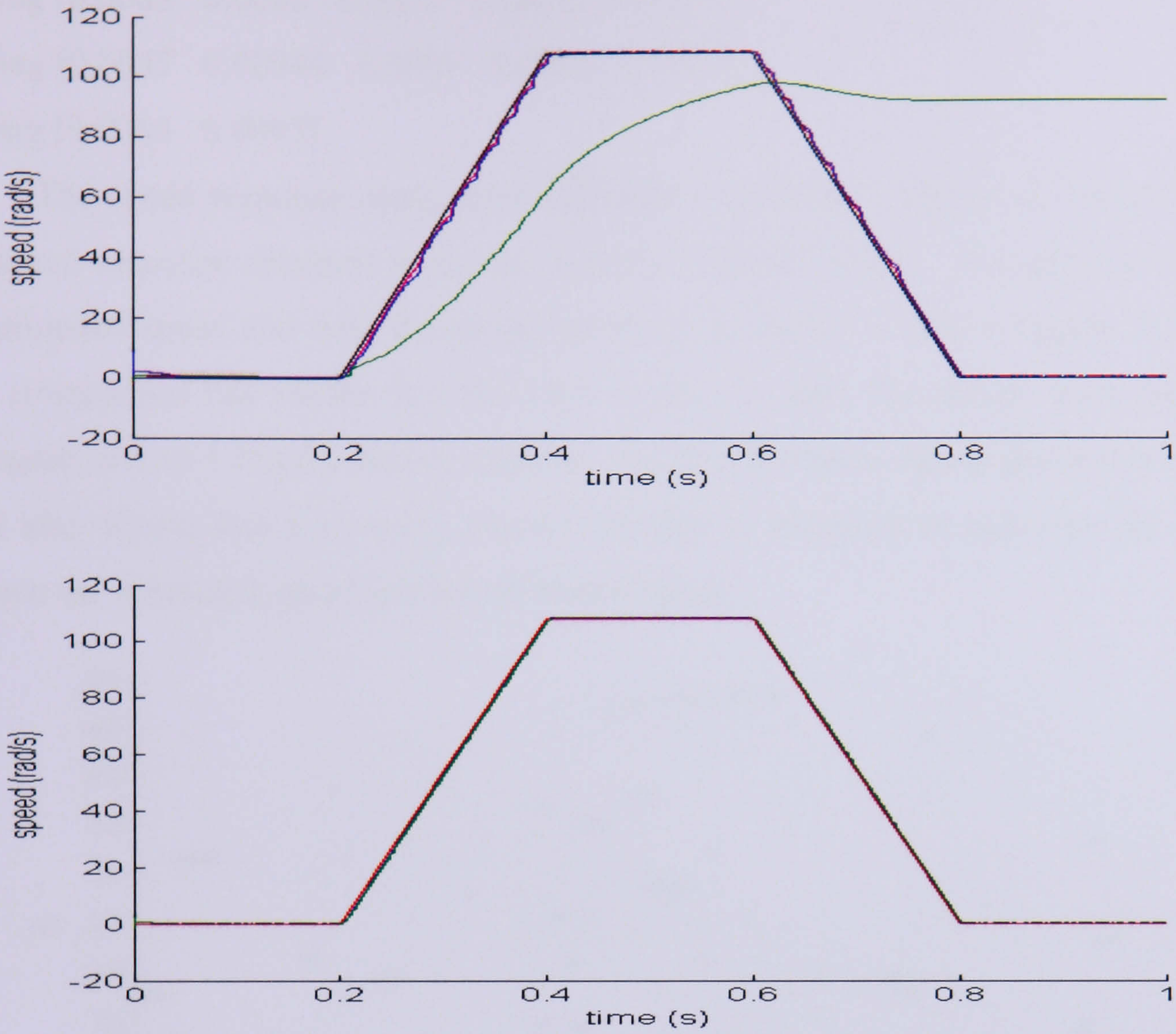


(a)



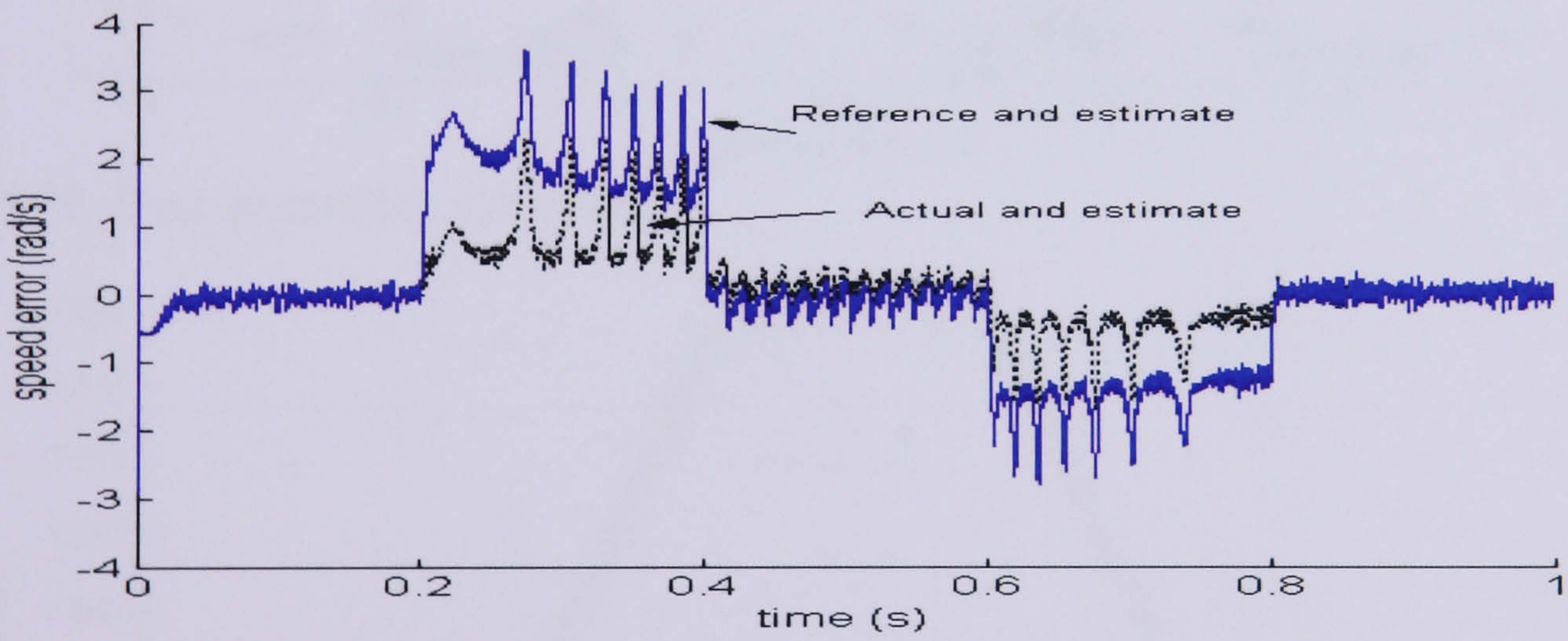
(b)

Figure 7.28: (a) Best generation obtained (b) Best generation obtained in enlarged scale



(a)

(b)



(c)

Figure 7.29: (a) Reference and speed responses using different objective functions obtained during transient and steady state operation (b) Best speed response tuned by SA (c) Speed error obtained between reference, actual and estimated speed

7.8.2 Optimisation on Open Loop Estimator for 101 % of Rated Speed

This test aims to find any difference in using a higher speed demand as an input. To complete the whole generations of 297 iterations, takes about 5 hours (4 hour 57 minutes). The best objective function obtained is 0.4050 at ts=172-181 shown in Figure 7.30 and using the covariance matrix:

Q =diag [0.0083 0.0030 0.0071 0.0007 0.7502]

G =diag [0.0047 0.00940 0.0001 0.0005 0.0074]

R= diag [0.0064 0.0093]

The speed response using three difference solutions is shown in Figure 7.31. The best speed response obtained is shown clearly in Figure 7.32(a). The speed error between the estimated speed and with the actual speed of the motor is shown Figure 7.32(b). The error comparison has shown that the EKF is able to track the actual rotor speed with a difference of 1 to 1.5 rad/s during transient and 0 to 0.3 rad/s during the steady state. The result also shows that a properly tuned controller is essential to reducing the high error between the estimated speed and the reference speed.

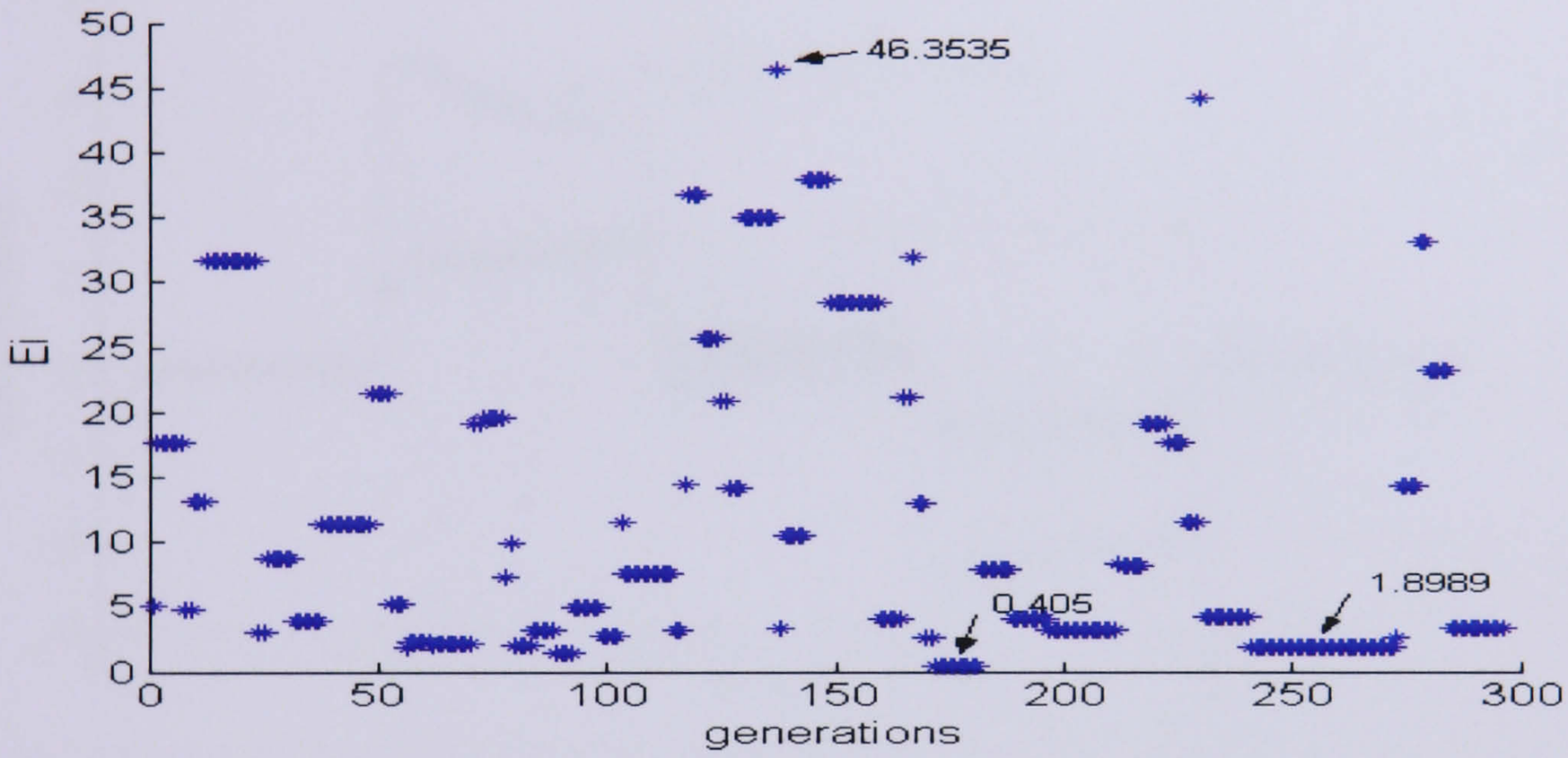


Figure 7.30: Best generation obtained

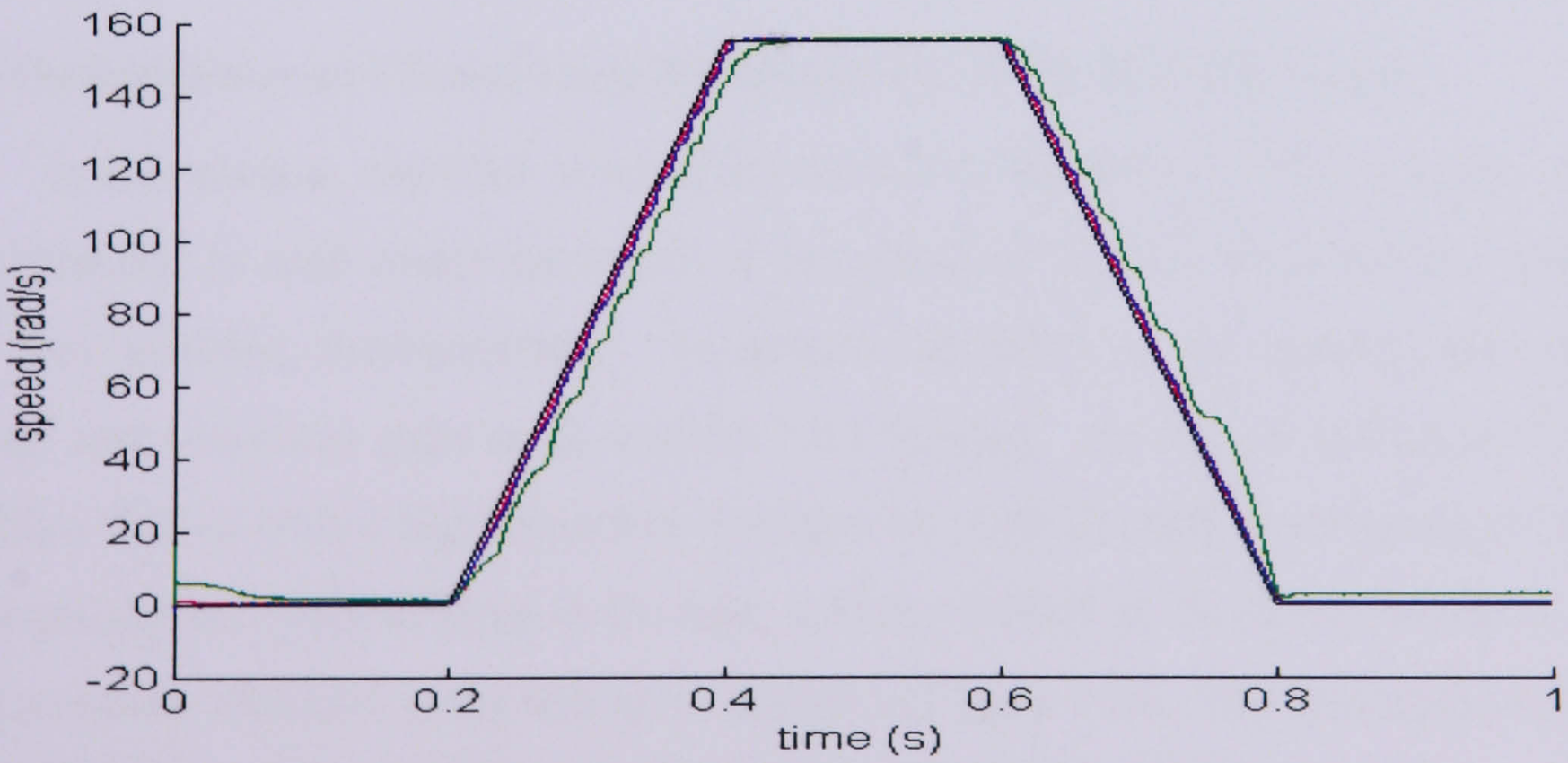
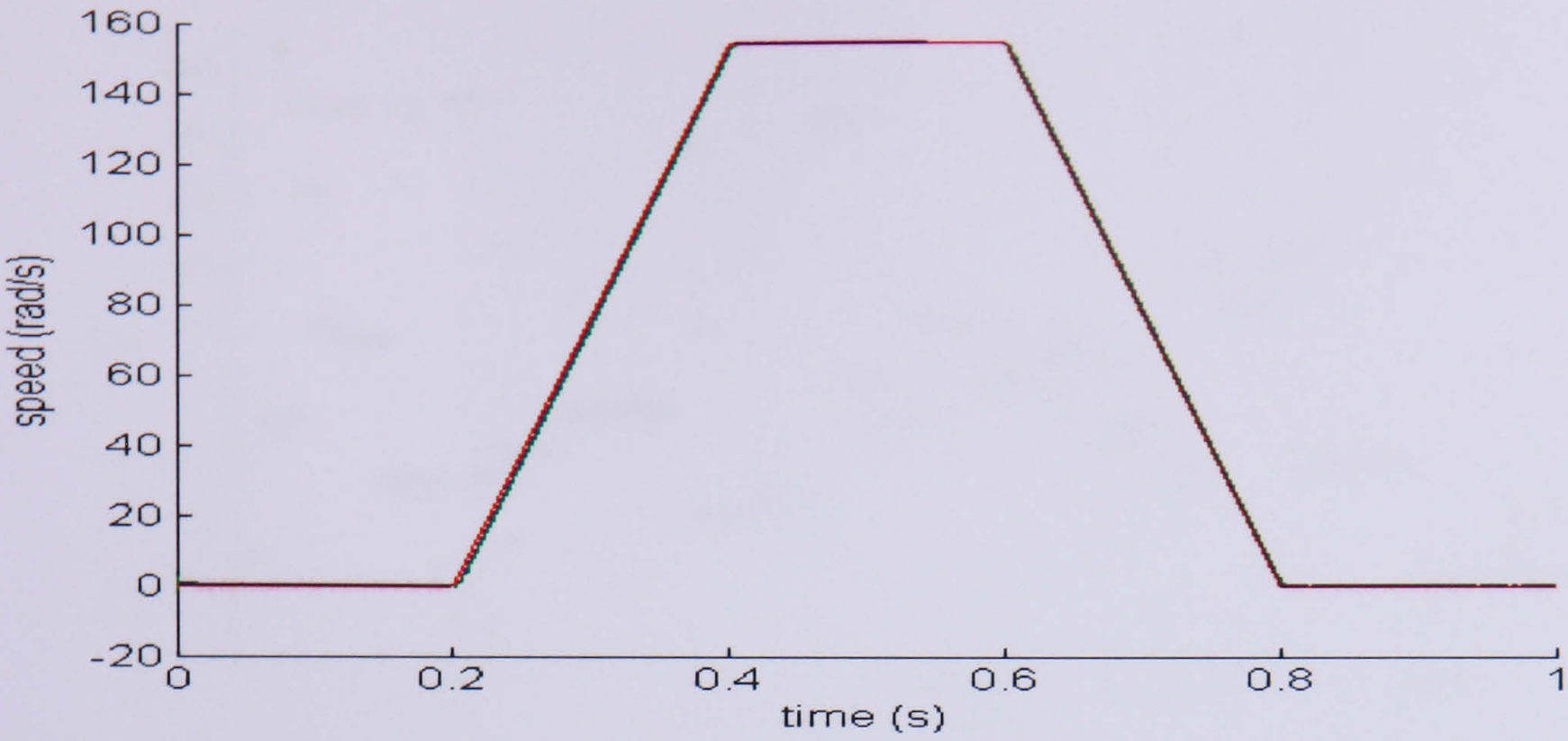
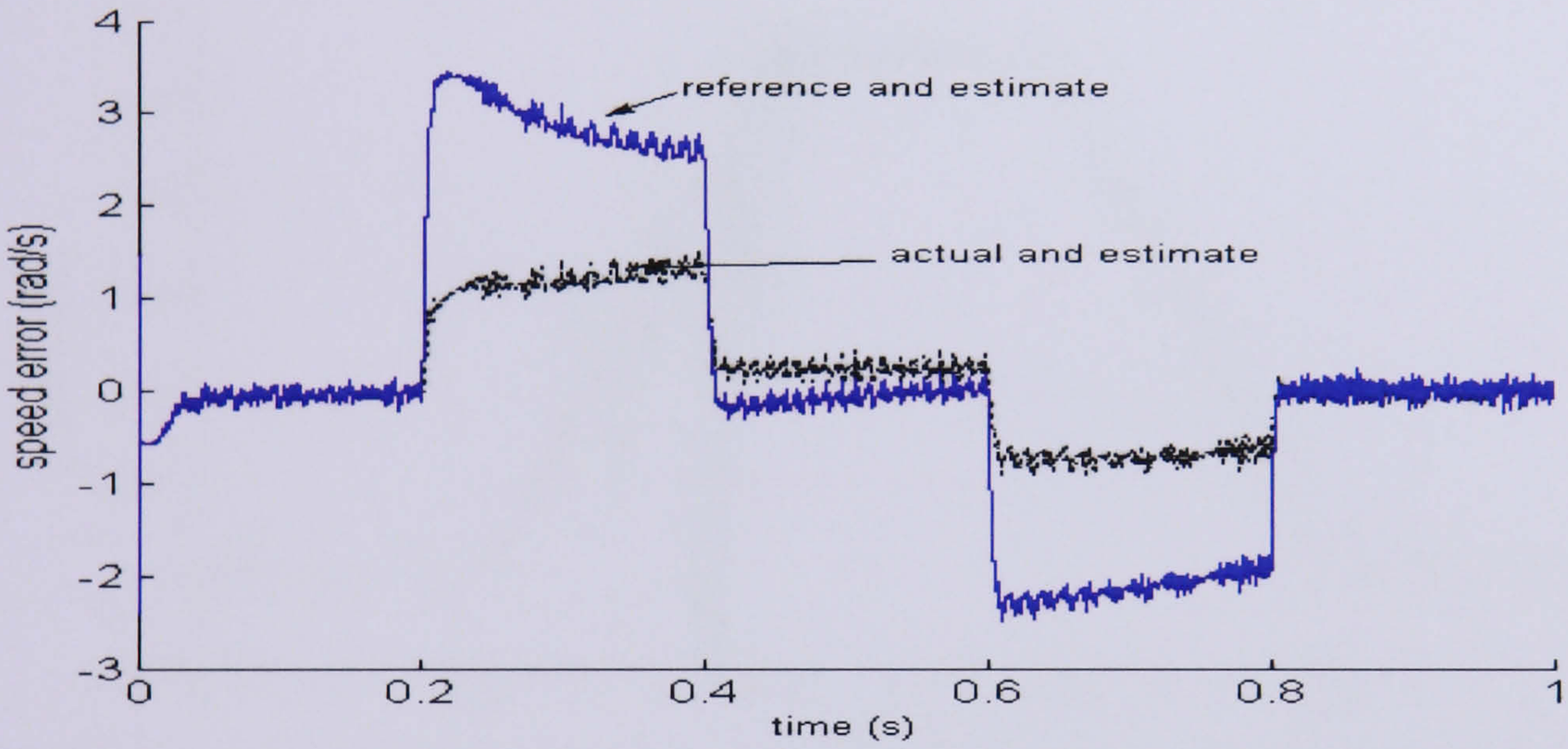


Figure 7.31: Reference and speed responses using different objective functions; the reference in black, the best objective function Ei=0.405 in magenta, biggest objective function Ei=46.35 in blue and most accepted objective function Ei=1.899 in green trace



(a)



(b)

Figure 7.32: (a) Best speed response tuned by SA (b) Error between speed responses

7.8.3 Optimisation on Closed Loop Estimator for 70 % of Rated Speed

In this section, the EKF is tuned by the SA at demand of 70% of rated. The closed loop estimator is used where the speed of the estimator is used for controller feedback and rotor flux position determination. To assure reliability of the results, the same speed demand and controller gain as in section 7.8.2 is used. As shown in Figure 7.33(a), the algorithm started with a high objective function of 1158.12 until ninth iteration. In this test the frequently accepted solution is the best, with E_i=0.9607 at the iterations 246 to 279. The speed response obtained using this set is shown in Figure 7.34. The best speed response and error are shown in Figure 7.35 (a, b) with the covariance matrix of:

Q =diag [0.0030 0.0057 0.0022 0.0014 0.9786]

G =diag [0.0021 0.0014 0.0005 0.0026 0.0076]

R= diag [0.0044 0.0051]

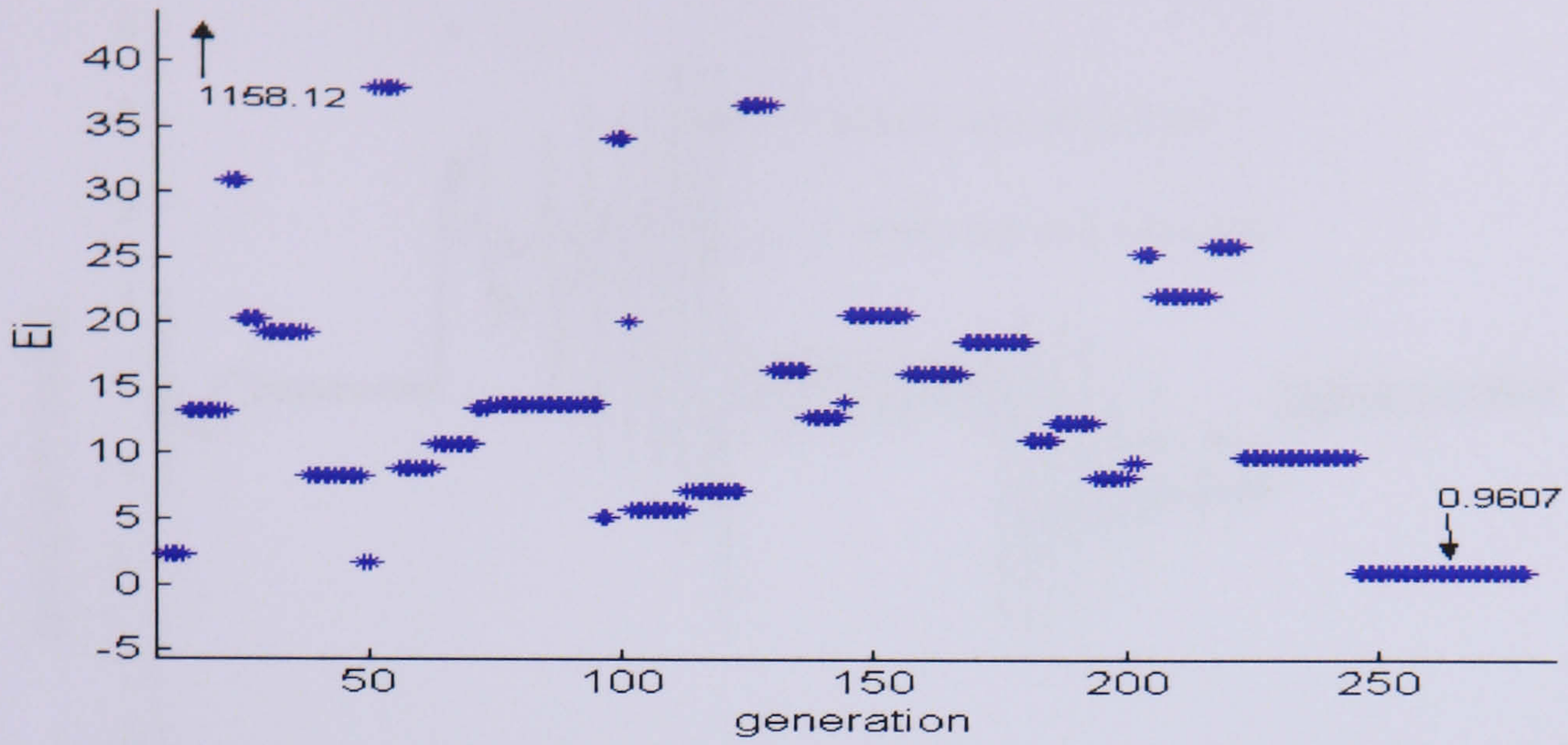


Figure 7.33: Objective function obtained

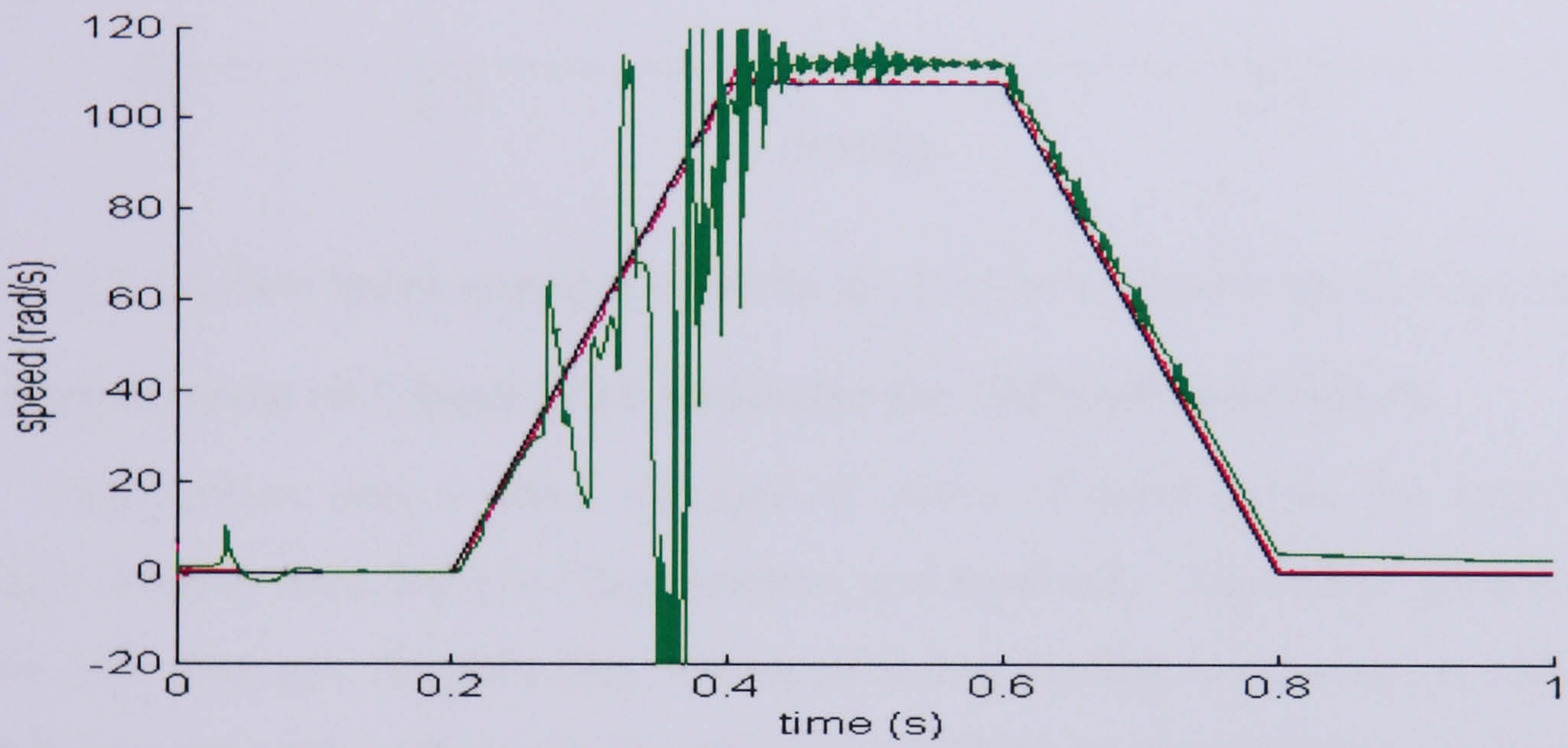
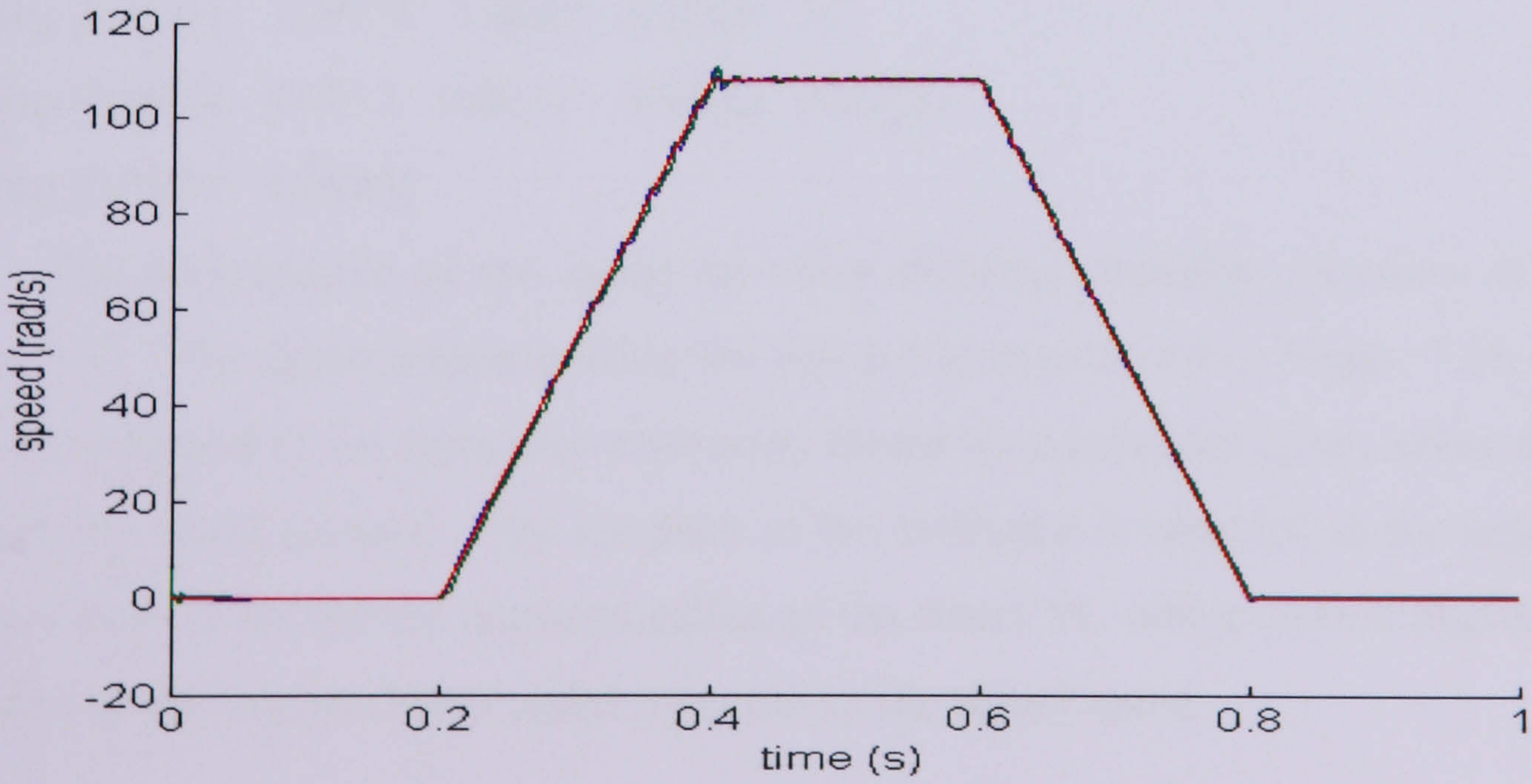
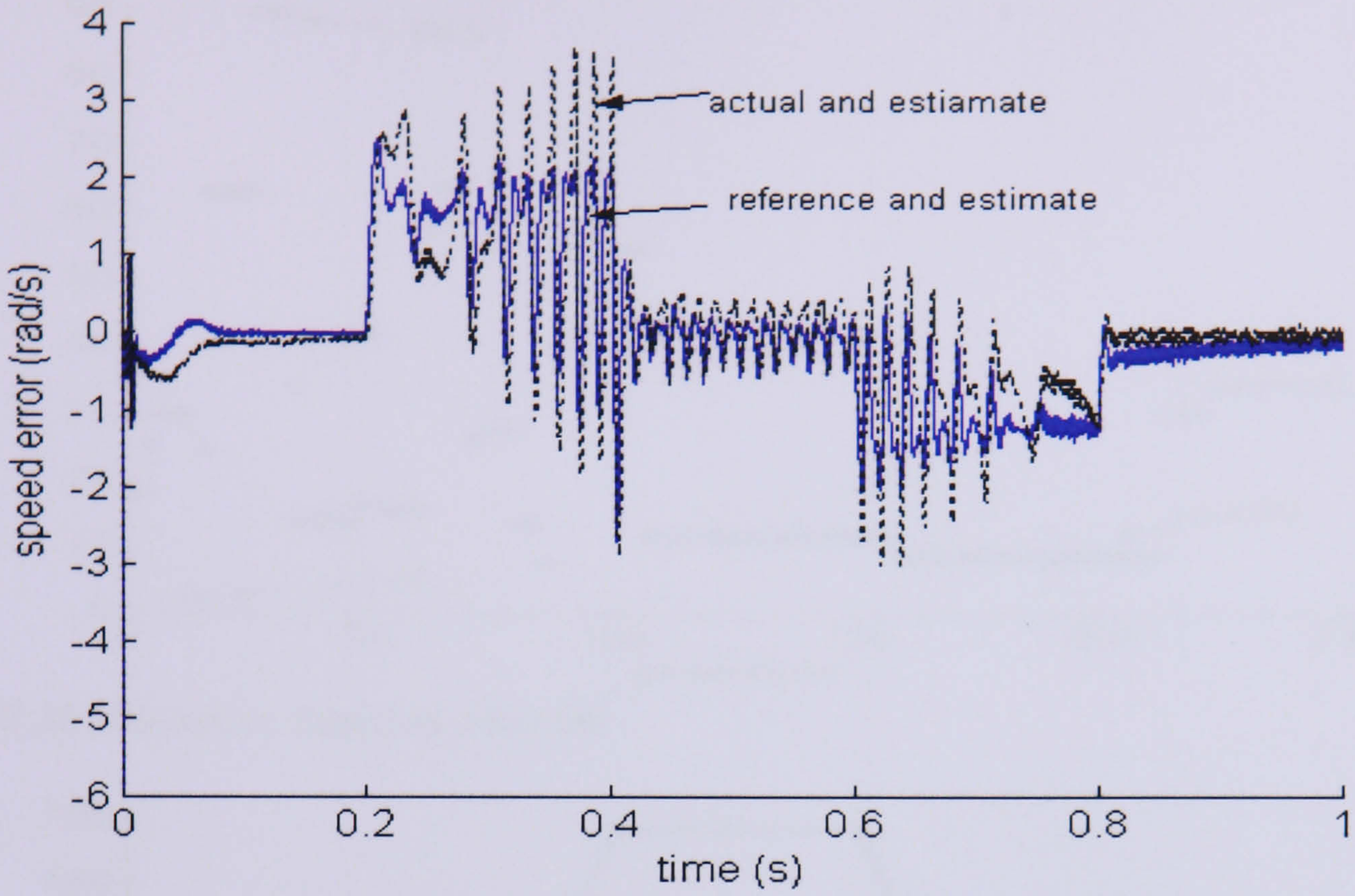


Figure 7.34: Reference and speed responses using different objective functions obtained during transient and steady state operation



(a)



(b)

Figure 7.35: (a) Best speed response tuned by SA (b) Error between speed responses

7.8.4 Optimisation on Closed Loop Estimator for 101% of Rated Speed

This section uses a speed demand of 101% of rated speed, the speed from the estimator is again used for rotor flux position and feedback. The initial generation started up with low objective function then increased before finding the minimum solution. The best solution obtained as shown in Figure 7.36 is 1.88 during the iteration of 14 and 15. The noise covariance matrix obtained is:

Q =diag [0.0065 0.0075 0.0009 0.0010 1]

G =diag [0.0078 0.0015 0.0011 0.0022 0.0071]

R= diag [0.0034 0.0023]

The performance of the speed for three different objective functions is shown in Figure 7.37. The speed response using the best solution is shown in Figure 7.38. In Figure 7.39, as compared to the open loop estimator, closed loop estimator given better response in tracking the speed demand. The accuracy of the estimator is essential in the application of the sensorless of IM for the implementation of the direct VC where correct alignment of the rotor flux position gives better speed response of the actual speed.

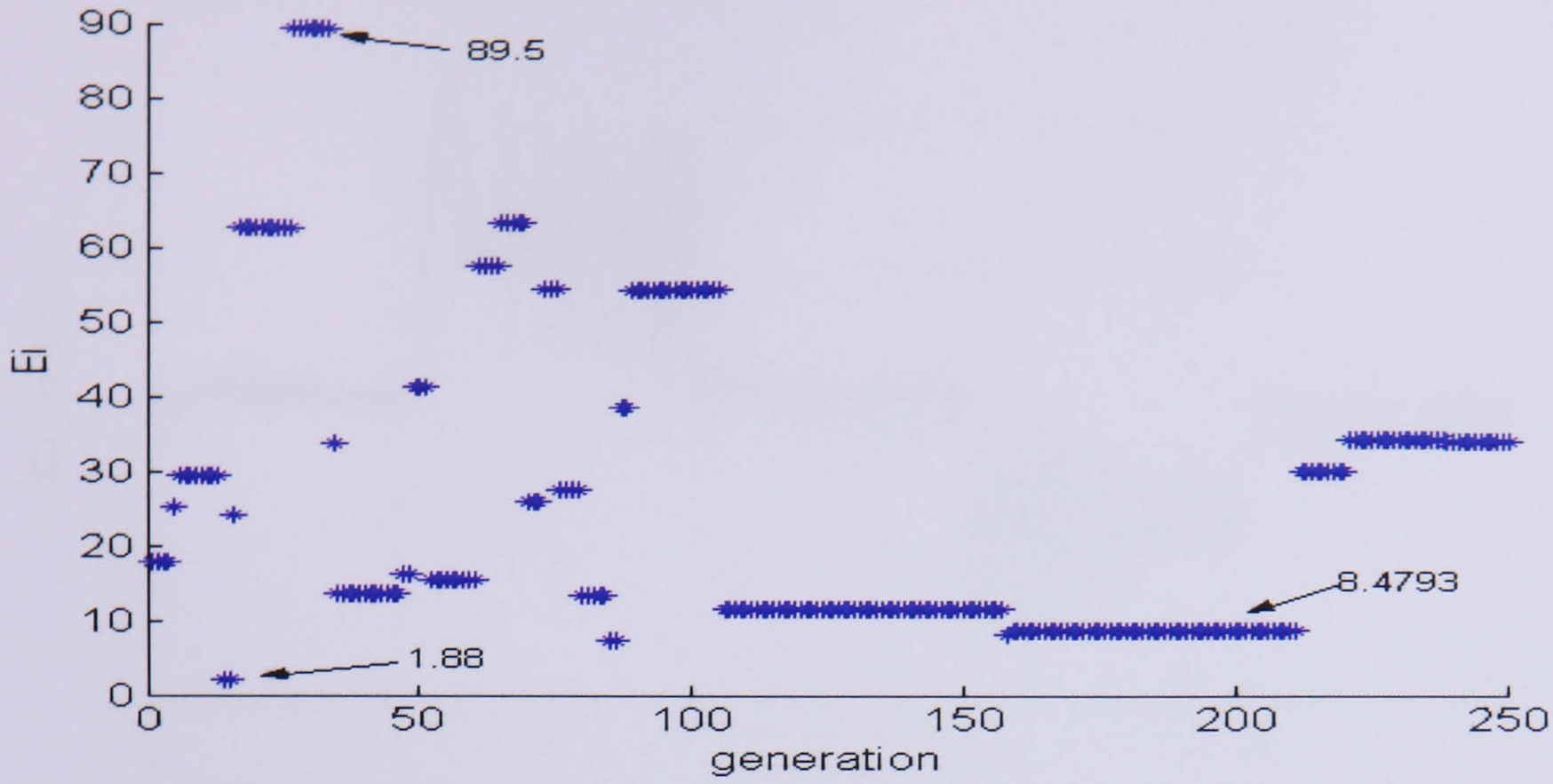


Figure 7.36: Objective function obtained

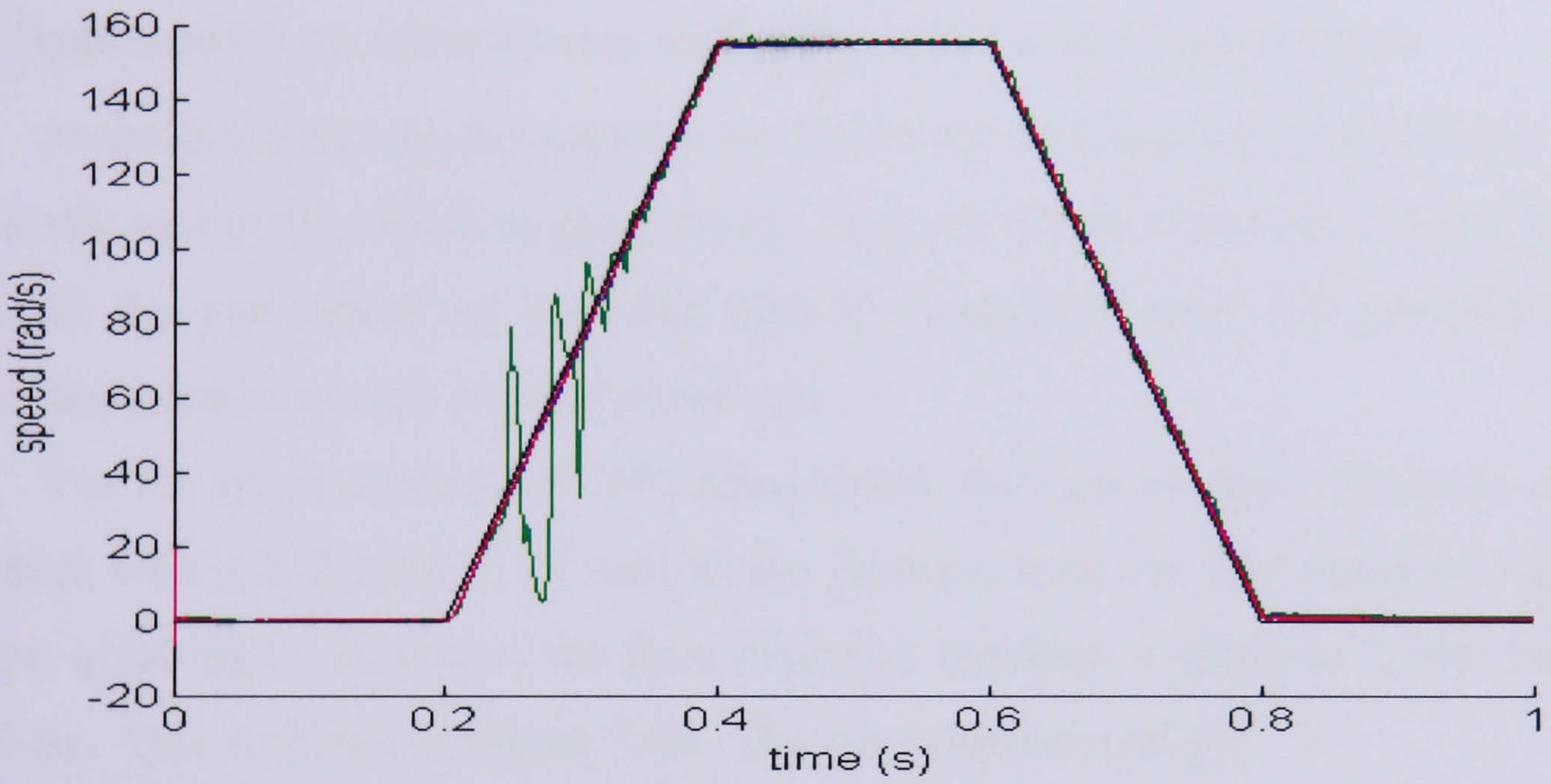


Figure 7.37: Reference and speed responses using different objective functions obtained during transient and steady state operation

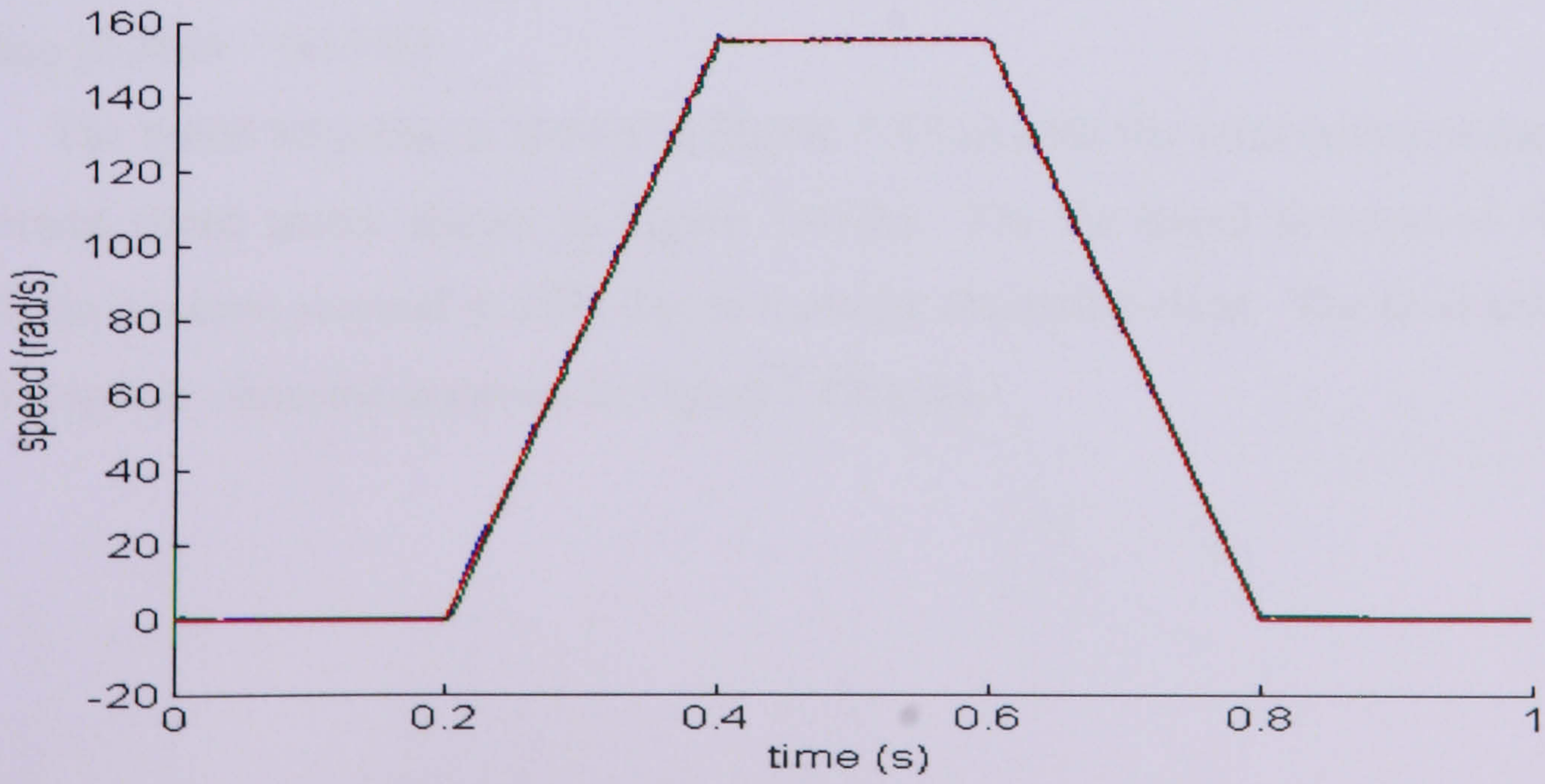


Figure 7.38: Best speed response tuned by SA

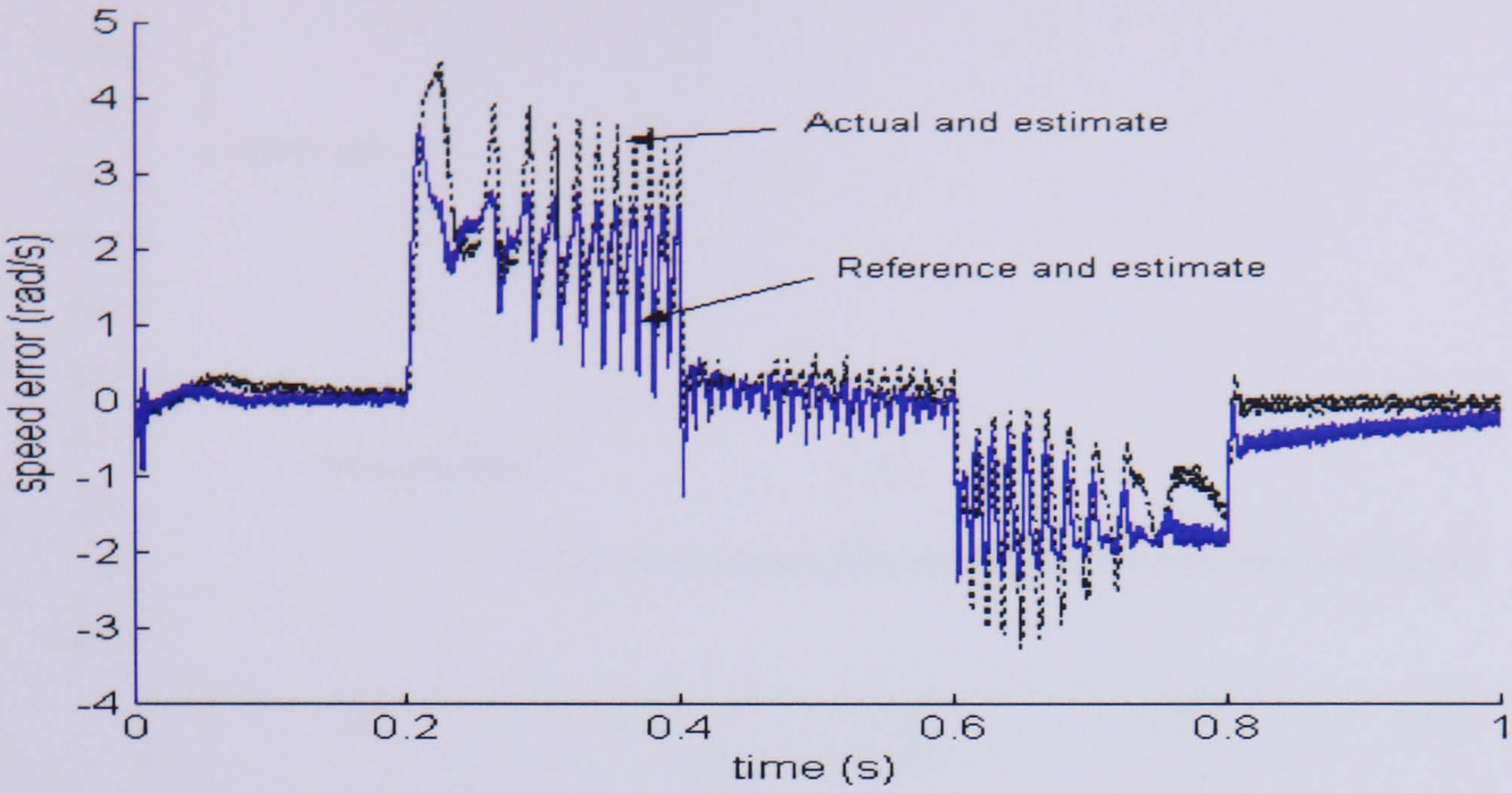


Figure 7.39: Error between speed responses

7.8.5 Optimisation on Closed Loop Estimator with Larger Search Area

In the previous test, the search area for the SA was specified for (0-0.01) for all the parameters except Q_{55} which is given (0-1). Now the SA is tested on a bigger search area where all the parameters are searched over (0-1) and the other SA parameters such as control temperature are the same as previously.

For the speed demand of 70% rated speed, the first objective function obtained is very high which is 2.2085×10^7 and as the iteration goes on, the objective function still remains quite high. However, the final objective function is stable at 2.096 from 80-256 iterations. This is shown in Figure 7.40. The covariance matrix is:

Q =diag [0.9230 0.7435 0.3657 0.0200 0.9239]

G =diag [0.1108 0.1138 0.0019 0.0169 0.2525]

R= diag [0.3824 0.1745]

The speed response is shown in Figure 7.41(a) with the error between the reference and actual speed motor shown in Figure 7.41(b). For the speed demand of 101% rated speed, the iteration stopped at 207s due to memory allocation error. The final and optimum speed response obtained is shown in Figure 7.42(a, b).

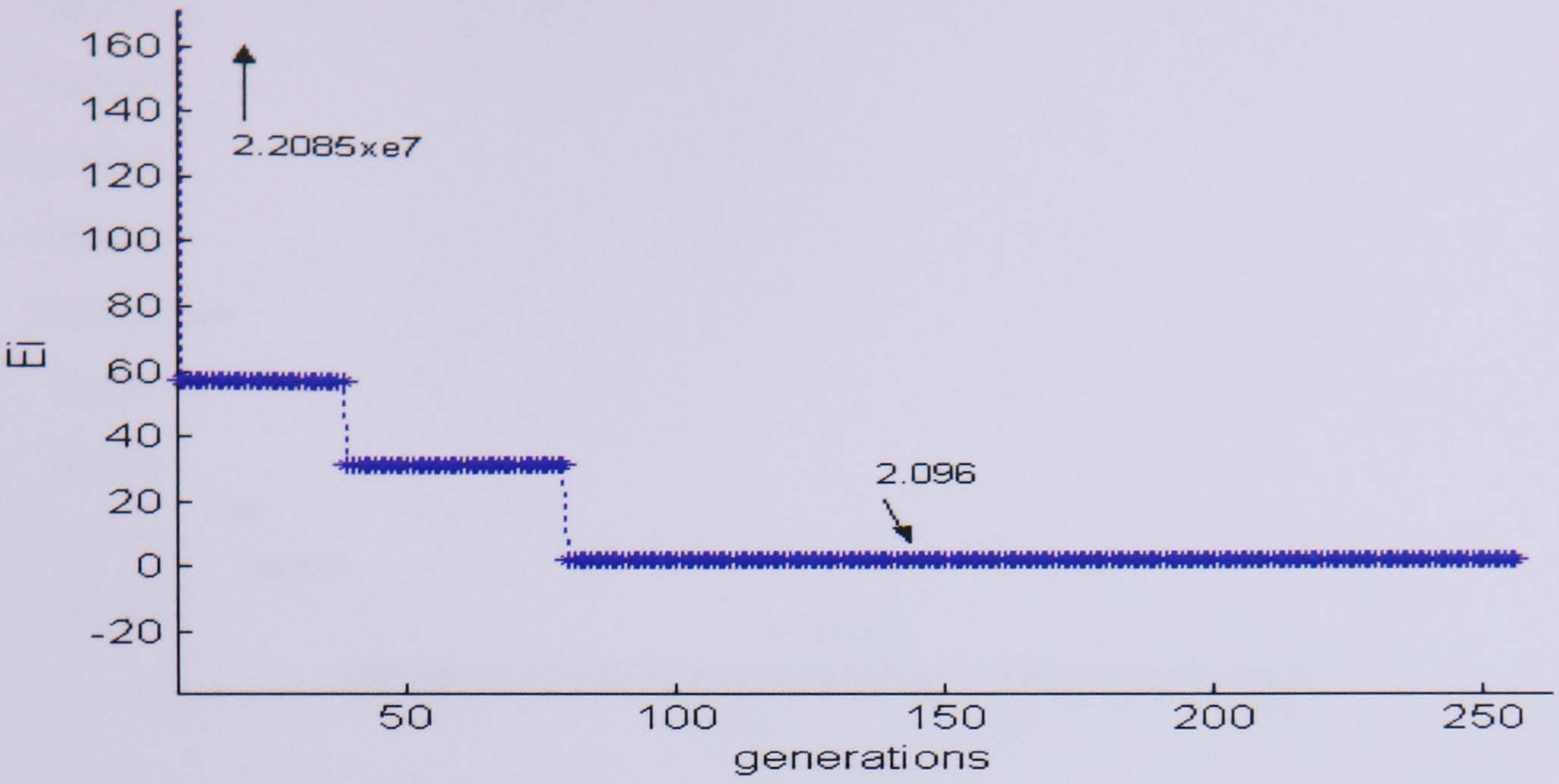
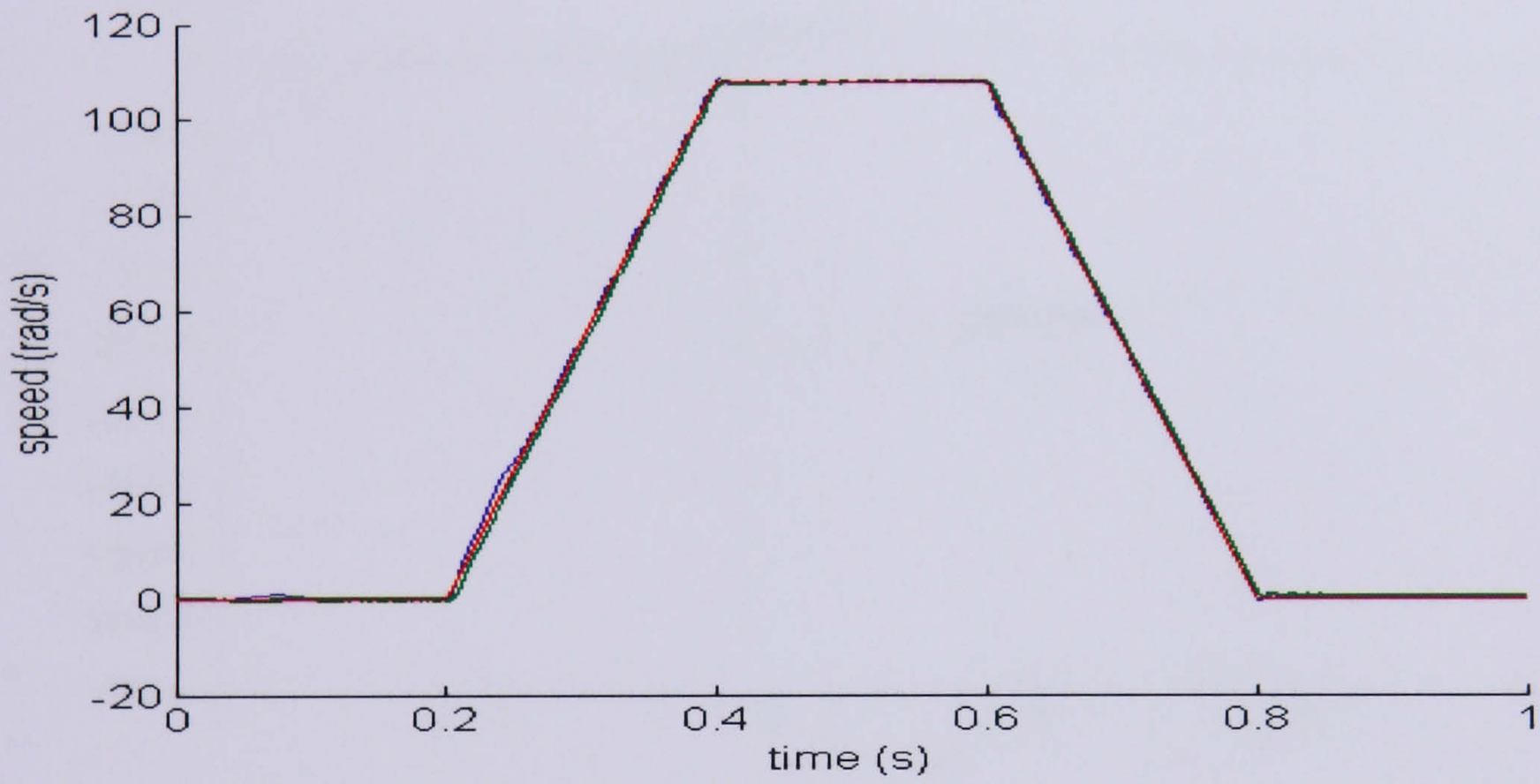
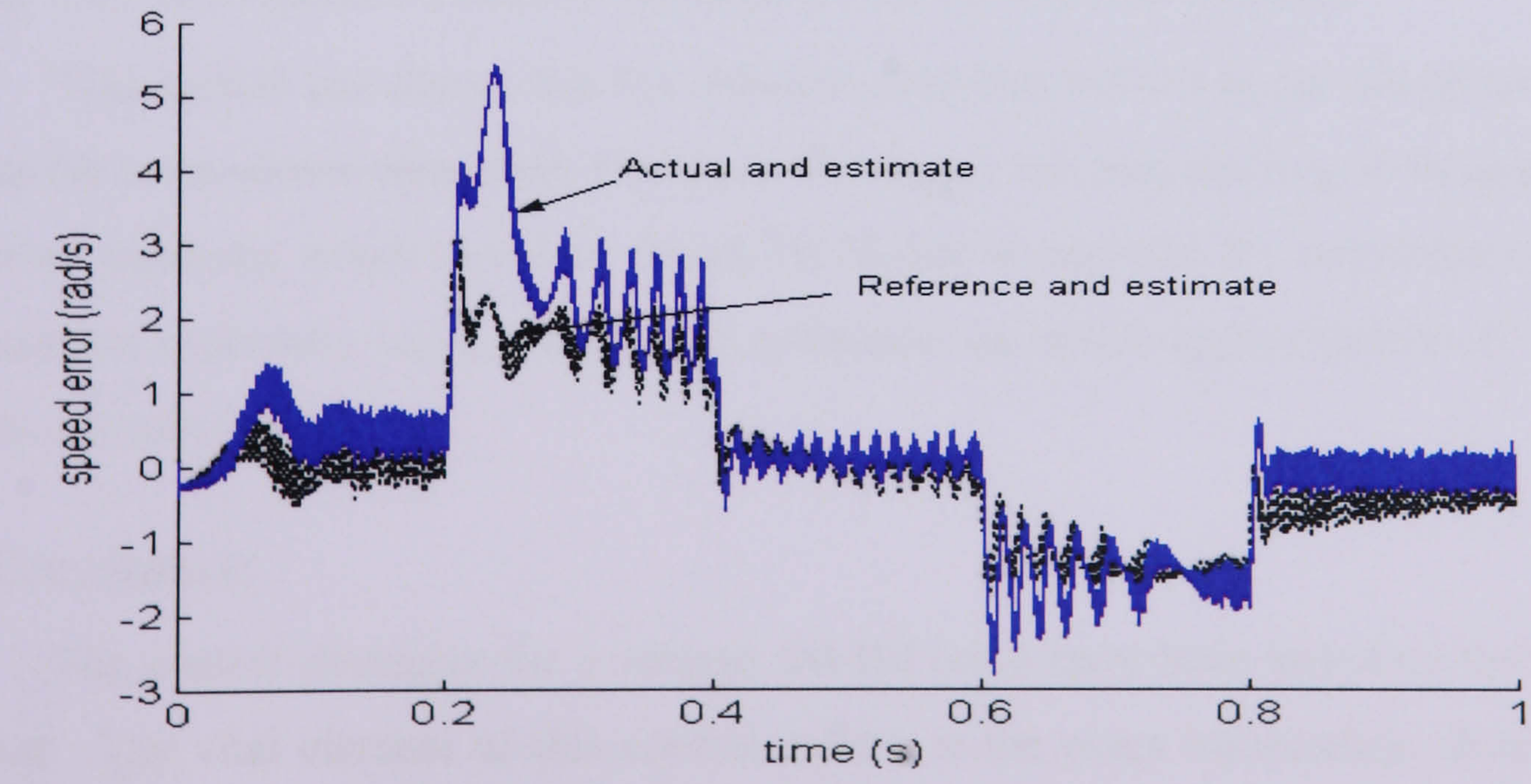


Figure 7.40: Objective Function

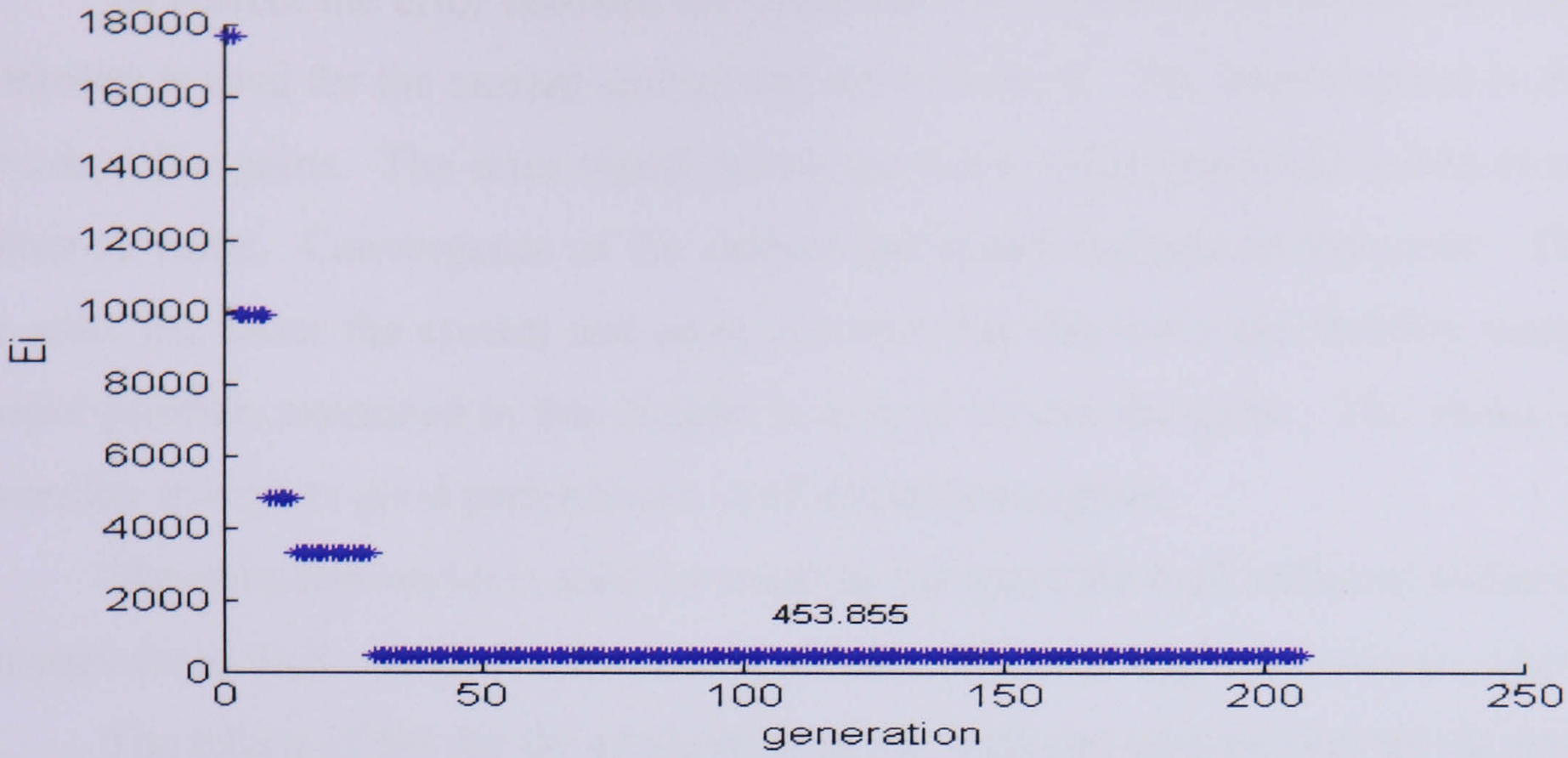


(a)

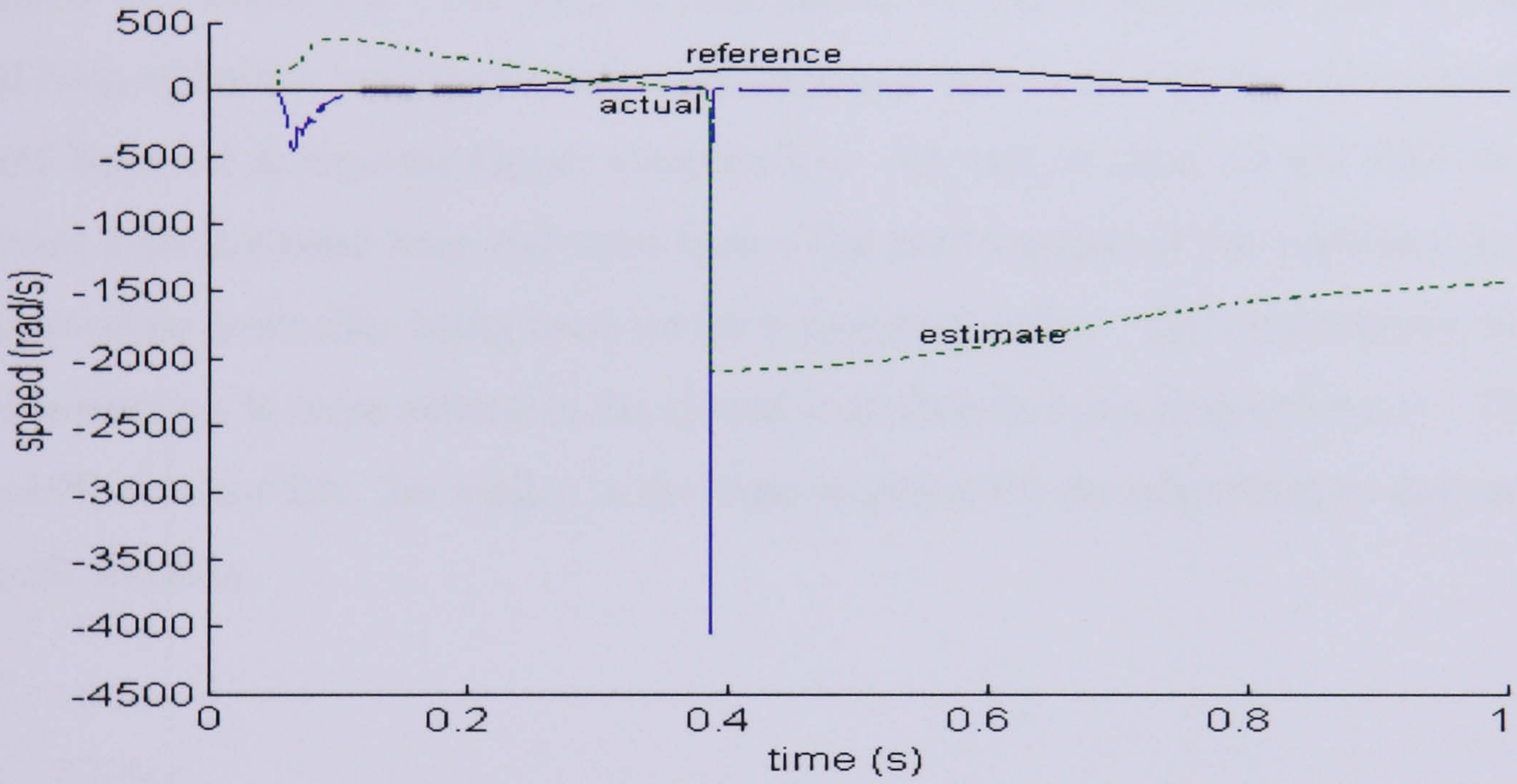


(b)

Figure 7.41: (a) The speed response (b) Error between speed responses



(a)



(b)

Figure 7.42: (a) Objective Function obtained (b) Speed response obtained

This section has shown that the objective function reduces as the iterations increased and as the temperature decreased. However, the bigger the area needing to be searched and the more solutions which could be found the longer is required for convergence. For VC applications especially using closed loop estimator, an initial approximation of the search area is recommended.

7.9 Conclusion

The control strategies for a voltage fed IM drive have been based on the direct VC method. The vital element of this control system is the exact information about the rotor flux angle as well as amplitude. The rotor flux model is formulated based on the stator current and rotor speed.

To correct the error between the predicted and measured values of the system, a PI controller is used for the current control and speed control. The improvement is defined by the controller gains. The error signal drives the value of the predicted values closer to the reference value. Convergence of the current and speed depends on the gains. The higher the gain, the faster the system and more accurate, but the lower the stability margin. The crucial problem examined in this chapter is how to choose the gains. The choice of a gain controller that gives good performance at all situations is given.

The estimator which is used for tracking the speed for both with and without sensors, has performed well. However, the accuracy of the estimator depends on its parameters.

The tuning of SA for the application of VC with and without sensors for determining the speed was discussed. For each configuration the same controller gain is used. The closed loop estimator has shown better performance than open loop in estimating the speed demand however it requires longer computation. SA can be used for the EKF to tracking the speed both in closed loop and open loop. The performance of the estimator depends on the appropriate controller being used for each speed reference. The convergence to the best possible solution is more settled in the closed loop than in open loop estimator. The bigger the search area for SA, the longer is the time required for the algorithm to converge to an optimum solution.

CHAPTER 8: SIMULATED ANNEALING AND GENETIC ALGORITHM IN COMPARISON

8.1 Introduction

The GA is a direct random search technique to find a global optimal solution in a complex search space. It was first invented in 1970's by Holland [99]. GA is modelled on the natural biological evolution process. It operates on a population of potential solutions or individuals over several generations to gradually improve on their fitness. At each generation, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain, and breeding them together using genetic operators.

The potential solution for a problem is an individual known as a chromosome. The chromosomes can be represented by strings of numbers, normally but not necessarily, binary numbers. After deciding on the chromosome representation, it is possible to access the performance and filter the individual members of a population. The individuals are evaluated according to the objective and fitness function. The best individuals are selected to mate and generate offspring. Then, a new generation is created and the most fit individuals are selected to replace the least fit individuals of the previous generation while keeping the same population size. Through an iterative process, the population evolves towards better regions of the search space. The algorithm then converges to the best chromosome which represent the optimal or near optimal solution for the problem. The pseudo code for a GA and the flowchart for the algorithm are presented in Figure 8.1 and Figure 8.2.

```
begin
        t = 0
        initialize P(t)
        evaluate P(t)
        while (t < max # of generations) do
        begin
                t = t + 1
                select P(t) from P(t -1)
                reproduce pairs in P(t)
                evaluate P(t)
        end
end
```

Figure 8.1: Pseudo code of GA given by Chipperfield [100]

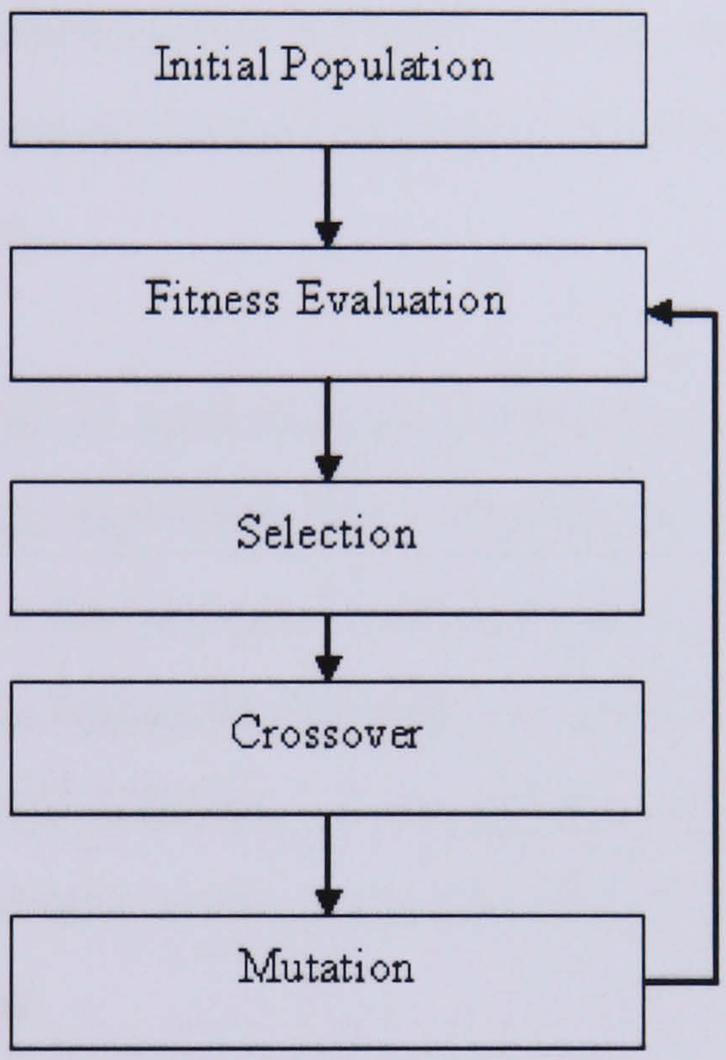


Figure 8.2: Flowchart of a standard GA based on [101]

8.2 Genetic Algorithm Operations

GA can be divided into eight main areas: population representation, initialization, objective function and fitness function, selection, crossover, mutation, reinsertion and termination.

1. Population Representation

The most commonly used representation of the individual or a chromosome are binary or bit strings {0, 1}. Another representation uses a vector of integers or real valued numbers.

2. Initialisation

The initial solution for the GA can be created by using a random number generator which uniformly distributes numbers in the desired range. The use of a random number

generator gives the advantage of accessing the performance of the algorithm without the use of a priori knowledge of the problem. Alternatively it can be created by testing a number of random initialisations for each individual, that the best performance is chosen to be the initial population. Another method is by giving the initial population some individuals that are known to be in the area of the global minimum.

3. Evaluation of Objective and Fitness Function

The objective function is responsible for measuring an individual's performance in the problem domain. This function acts as an interface between the GA and the optimisation problem. It indicates that the best individuals will have the lowest value of the objective function in the minimization problem. Once the objective function has been evaluated, it is transformed into a fitness function to provide a measure of the individual's relative or scaled performance.

4. Selection

The selection algorithm is used to select individuals for reproduction based on their relative fitness. The selection algorithm has a significant influence on driving the search towards a promising area and finding good solutions in a short time while maintaining the range of the population. The selection algorithm is used to avoid premature convergence and to reach the global optimal solution. Two common selection methods, roulette wheel selection and stochastic universal sampling, are described below:

4a. Roulette Wheel Selection

A roulette wheel mechanism is commonly used in selection techniques to probabilistically select individuals based on some measure of their performance. A real valued interval, Sum, is determined from the sum of the fitness values over all the individuals in the current population or the sum of the individuals' expected selection probabilities. The individuals are then mapped one to one into contiguous segments in the range [0, Sum]. Each individual's segment corresponds to the fitness value of the associated individual. To select an individual, a random number known as pointer is generated in the interval [0, Sum]. The individual whose segment spans the random number is selected. This process is repeated until the desired number of individuals is obtained. Figure 8.3 shows the circumference of the roulette wheel with the sum of all six individuals' fitness values. Individual 6 is the most fit individual and occupies the largest interval, while individual 1 is the least fit and for this reason occupies the smallest interval within the roulette wheel.

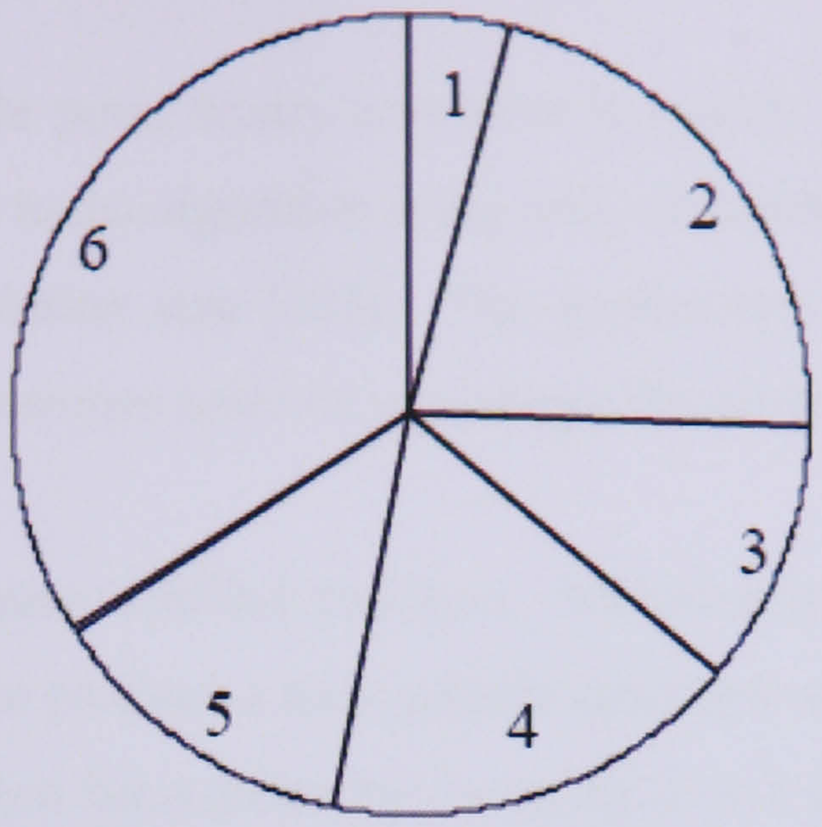


Figure 8.3: Roulette Wheel Selection

4b. Stochastic Universal Sampling

The stochastic universal sampling (SUS) algorithm is based on a variation of the roulette wheel method. Instead of the single pointer used in roulette wheel methods, the SUS method uses N equally spaced pointers, where N is the number of individuals to be selected and the distance between the pointers are 1/N. The population is shuffled randomly in the range of [0, 1/N] and this gives the position of the first pointer. Individuals whose positions span the positions of the pointers are selected.

5. Crossover

Crossover is the basic operator for producing new chromosomes in GA. The crossover produces new individual by selecting parts of the parents' genetic material to form the chromosomes of the children. The simplest form of crossover is single point crossover. In this operation, a pair of chromosome is selected and a crossover point is selected at random in the range of [1, L – 1] where L is the length of the string. To generate the offspring, the left strings of the crossover point of one parent are then combined with the right string of the crossover point of the other parent. The methodology of single point crossover can be illustrated by an example of 6-bit binary strings of two parents (A1, A2) shown below:

$$A1 = [1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1]$$

$$A2 = [1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0]$$

If the random crossover point was selected to be 3, then the two offspring A1' and A2' would become:

$$A1' = [1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0]$$

$$A2' = [1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1]$$

In this case, the single point binary crossover is exactly the same as real crossover [102]. A crossover rate used in the algorithm is the ratio of number of offspring produced in each generation to the population size [103]. The application of the crossover rate is to control the number of chromosomes selected to undergo the crossover operation [104].

6. Mutation

Another genetic operator is called mutation. Mutation is a random process where a gene is replaced by another to produce a new genetic structure with some probabilistic rule. In a binary string, the mutation takes place by changing 0 to 1 and 1 to 0. An example of mutation in the fourth bit of A1' is shown below:

$$A1' = [1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0]$$

7. Reinsertion

Once a new population has been produced, the fitness of the new individuals is determined. A generation gap may occur where fewer individuals are produced and this may cause a difference between the old and new population sizes. To maintain the original population size, some of the new individuals need to be reinserted in the old population. For the situation where not all of the new individuals are to be used then a reinsertion scheme is required to determine which individuals need to be present in the new population. To replace members of the old population, the least fit members or the oldest members of the population should be taken into consideration. This is to ensure the individual must be sufficiently fit for reproduction in future generations.

8. Termination

The fitness of a population may remain static for a number of generations before a better individual is found. This has made the convergence criterion of a stochastic search method difficult to specify. It is a common practice to terminate the GA after a pre-specified number of generations. The best member of the population is tested against the problem definition. When an acceptable solution is found the GA is terminated otherwise the GA need to restart a new search.

The above operation, solutions, methods and the sequence of them are directly taken from GA Toolbox for use with Matlab User's Guide, as created by [105] with additional information from [101].

8.3 Genetic Algorithm and its application to EKF

After describing the principles of GA, the next section concentrates on finding the optimum settings for the parameter of EKF for a sensorless IM drive. For comparison purposes, the genetic operators used are as proposed by [59]. The performance of the filter is very much dependent on the right noise levels of the covariance matrices, Q, G and R. By considering the diagonals of these matrices, a total of 12 parameters need to be adjusted. The chromosome can be represented by binary or real numbers. The use of real numbers gives an advantage of easy understanding as compared to binary numbers. The 12 parameters are therefore the genes of the chromosome and can be written as:

$$Chromosome = [Q1 \quad Q2 \quad Q3 \quad Q4 \quad Q5 \quad G1 \quad G2 \quad G3 \quad G4 \quad G5 \quad R1 \quad R2]$$

Where: **Q**= process noise;

R=measurement noise;

G=Noise weight;

The GA tuning of an EKF for Sensorless IM control is shown in Figure 8.4. The EKF estimator has used the measured voltage and current provided from the drive. The speed of the motor has been used as a reference speed. The combined motor model and EKF has been implemented and discussed earlier in Chapter 5. The GA starts by generating an initial solution, which is the diagonal elements of Q, G, and R. These solutions are then transferred to the estimator and the objective function is then calculated from the difference between the speeds. The GA operations are then carried out. The entire process is completely automatic without the need for manual interference. To compare the general dynamic conditions of the drive, different speed values are tested including transient, steady state and low speed as well as zero speed.

At the start of optimisation, the initial solutions are initialised over a suitable range. Based on Trial and Error tuning, the range for the solutions is set to [0-0.01] for the rest of the parameters and Q_{55} is set in the range of [0-1]. Bigger Q_{55} allows the rotor speed to have greater influence. The number of individuals in the population and the generation number however are different from one problem to another problem.

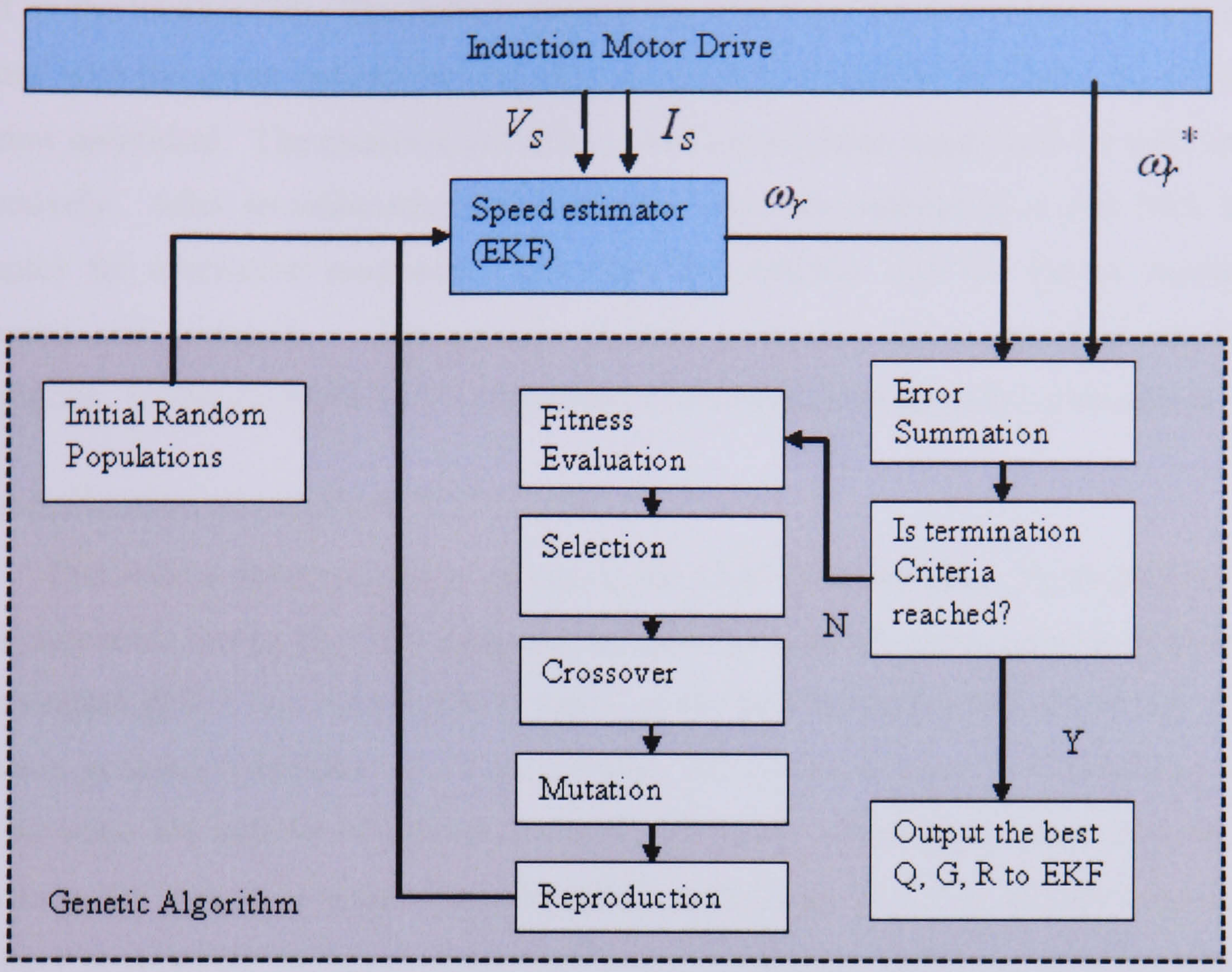


Figure 8.4: IM drive system

The objective function provides a measure of an individual's performance and here the objective function is based on minimizing the mean squared error of the estimated speed at all sampling time periods:

$$E = \frac{1}{n}\sum_{i=0}^{n}(\omega_{ri}^{*} - \omega_{ri})^2 \tag{8.1}$$

where :

$\omega_r^* = actual\_speed$;

$\omega_r = estimated\_speed$;

The fitness function which is used to transform the objective value into a measure of relative fitness is done using the linear ranking method. In this method, the most fit individual is given a fitness value of 2 and the least fit individual a fitness value of 0.

The selection of parents to form new offspring for the reproduction process is done by stochastic universal sampling. Once the parent structure has been selected, single point crossover is used to recombine the individuals from a population with the rate of 0.8 by exchanging the information between two chromosomes. The recombined chromosomes are then returned as new chromosomes. The mutation of the real valued population is done

based on the breeder GA. The current individuals which are the real valued population are mutated with the given probability and after mutation is complete; it returns the individual as a new individual. The mutation probability and the mutation range used are 0.01 and 0.1 respectively. After recombination and mutation, each new individual is sent back to the estimator for evaluation purposes. The objective function and the fitness function is measured and assigned. The process is then continued for a specified number of generations. In this study the GA is terminated at the number of specified generations.

8.4 Application to constant V/f drives

This section presents results of tuning the EKF using the GA. To avoid confusion when comparing tuning the EKF using the GA and SA later, the term guess is used instead of generation [93]. The comparison is based on the best solution achieved by SA. Using the same generation number, of 15 generations, the GA must have 21 individuals which give the same 336 number of guesses as the SA including the initial possible solutions. An open loop V/f supply is used. The simulation time is set to 2.5 s, enough time for the estimator to complete the series of different speed conditions including, transient state and steady state, acceleration and deceleration, low speed as well as zero speed zone. Using an AMD Athlon XP with 2.09Ghz, 512 MB of RAM and Matlab 7.0, the simulation runs for about 24 hours for 336 guesses including initial solution. The use of the EKF requires each guess to run for about 4 minutes.

The GA iterates around the loop until the maximum number generation is reached. After several trials, the best objective function achieved is 0.7676 with the best set chromosomes of:

Chrom= [0.0039 0.0051 0.0026 0.0007 1.0000 0.0017 0.0055
0.0001 0.0001 0.0093 0.0056 0.0007]

The objective function for the best chromosomes produced at each generation can be seen in Figure 8.5. The speed response obtained using the best achieved chromosome is shown in Figure 8.6. The estimated speed error in rad/s is shown in Figure 8.7 and 8.8. In Figure 8.8, the absolute speed error shows some overshoot during the transient period and the speed error settled from 0.4s onwards. The GA has been shown to be very good at optimising the EKF. The problem at the initial low speed is mostly not because of the GA itself, but due to the initial convergence of the EKF combined with a rapidly changing speed.

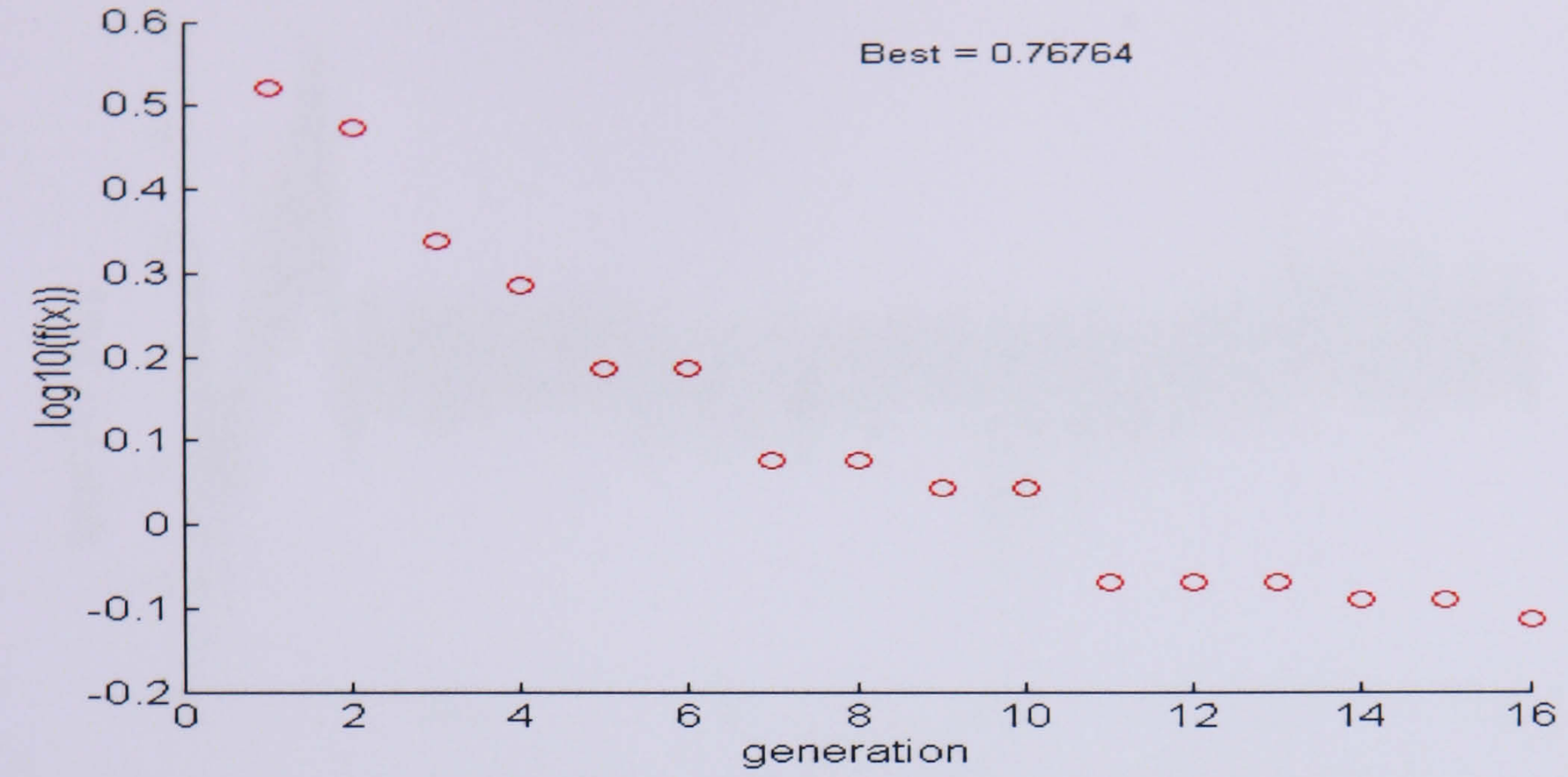


Figure 8.5: Objective Function for the best chromosome of each generation

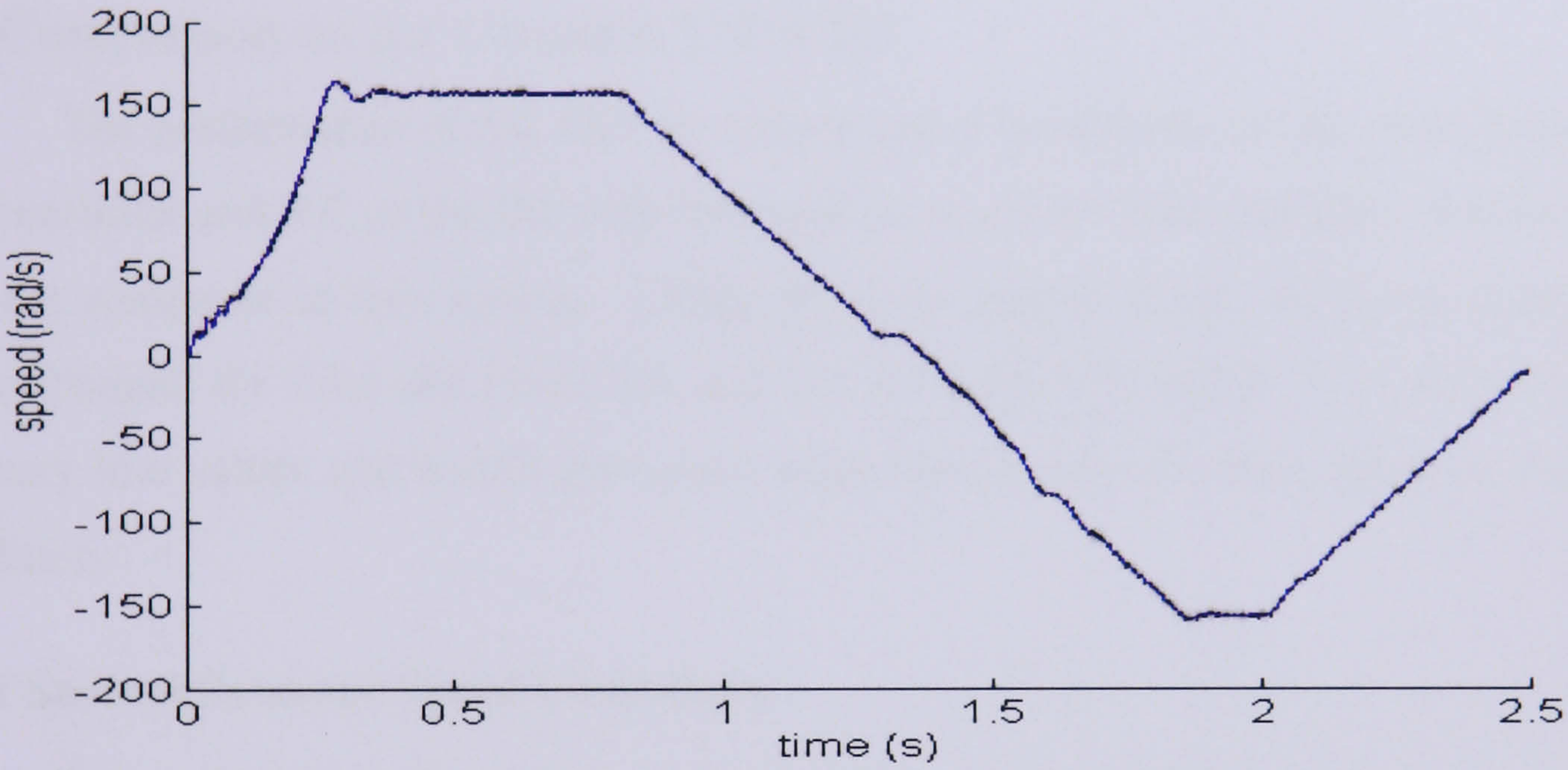


Figure 8.6: Estimated and reference speed

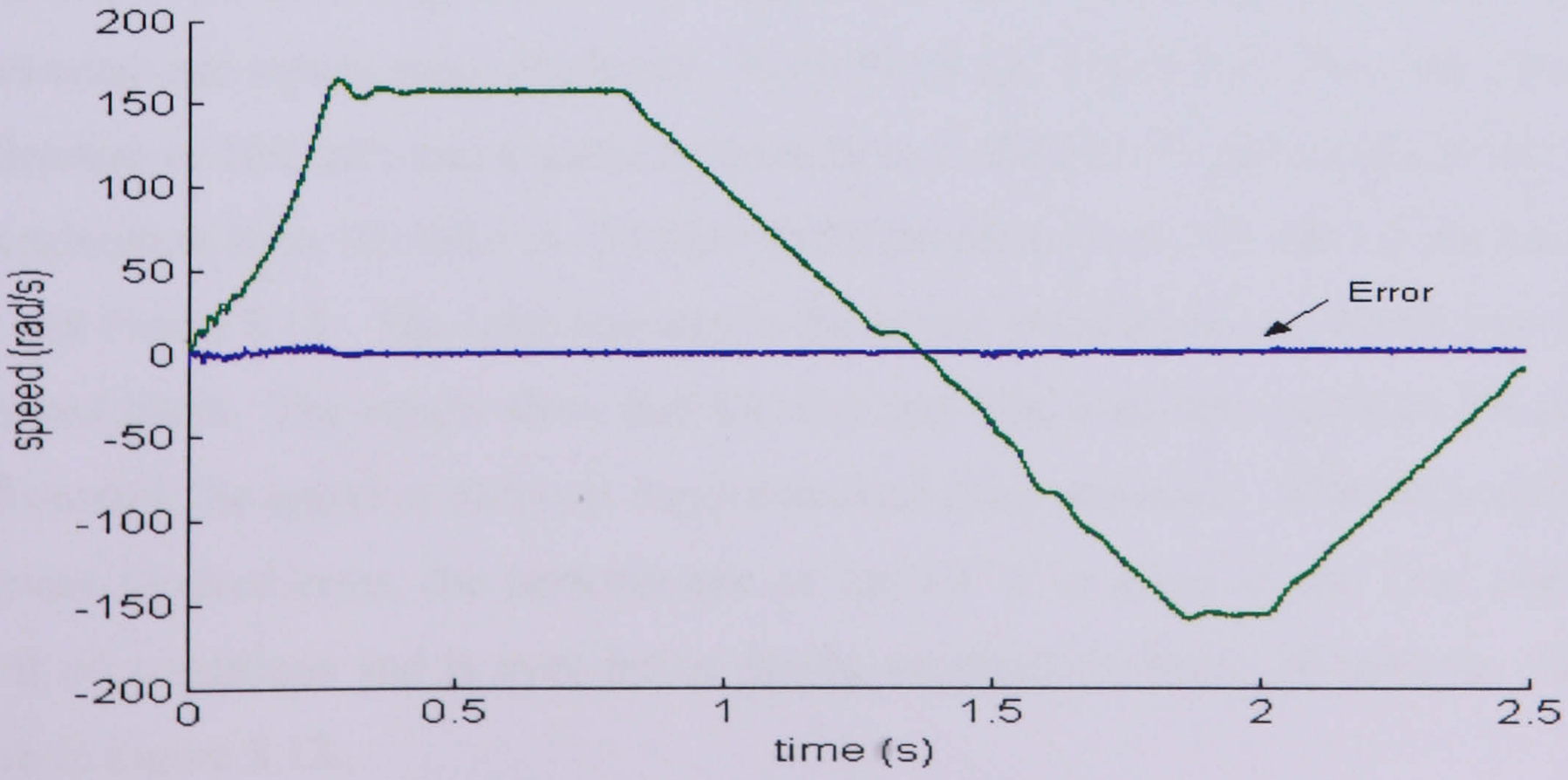


Figure 8.7: Estimated and reference speed and the error between these speeds

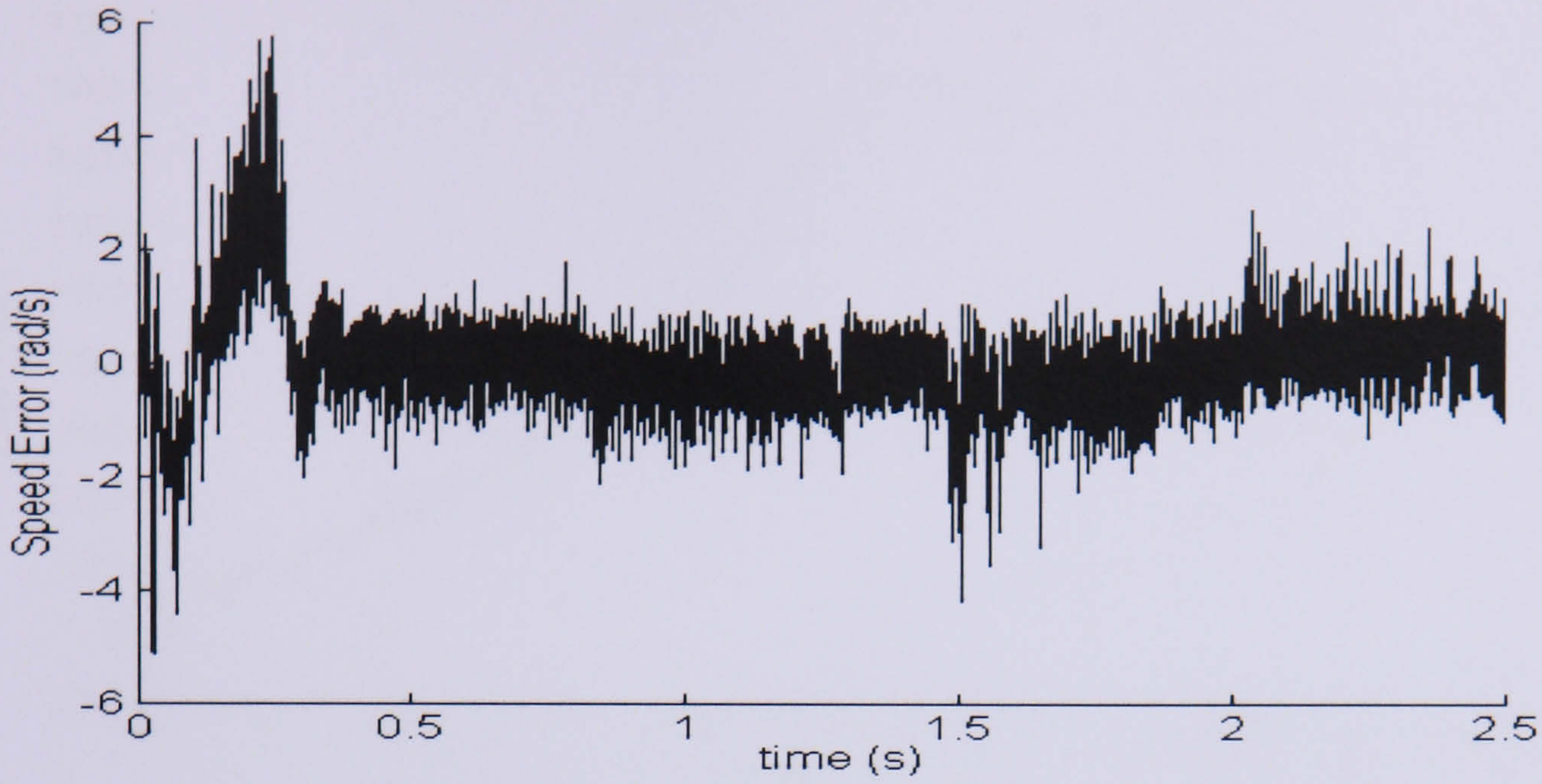


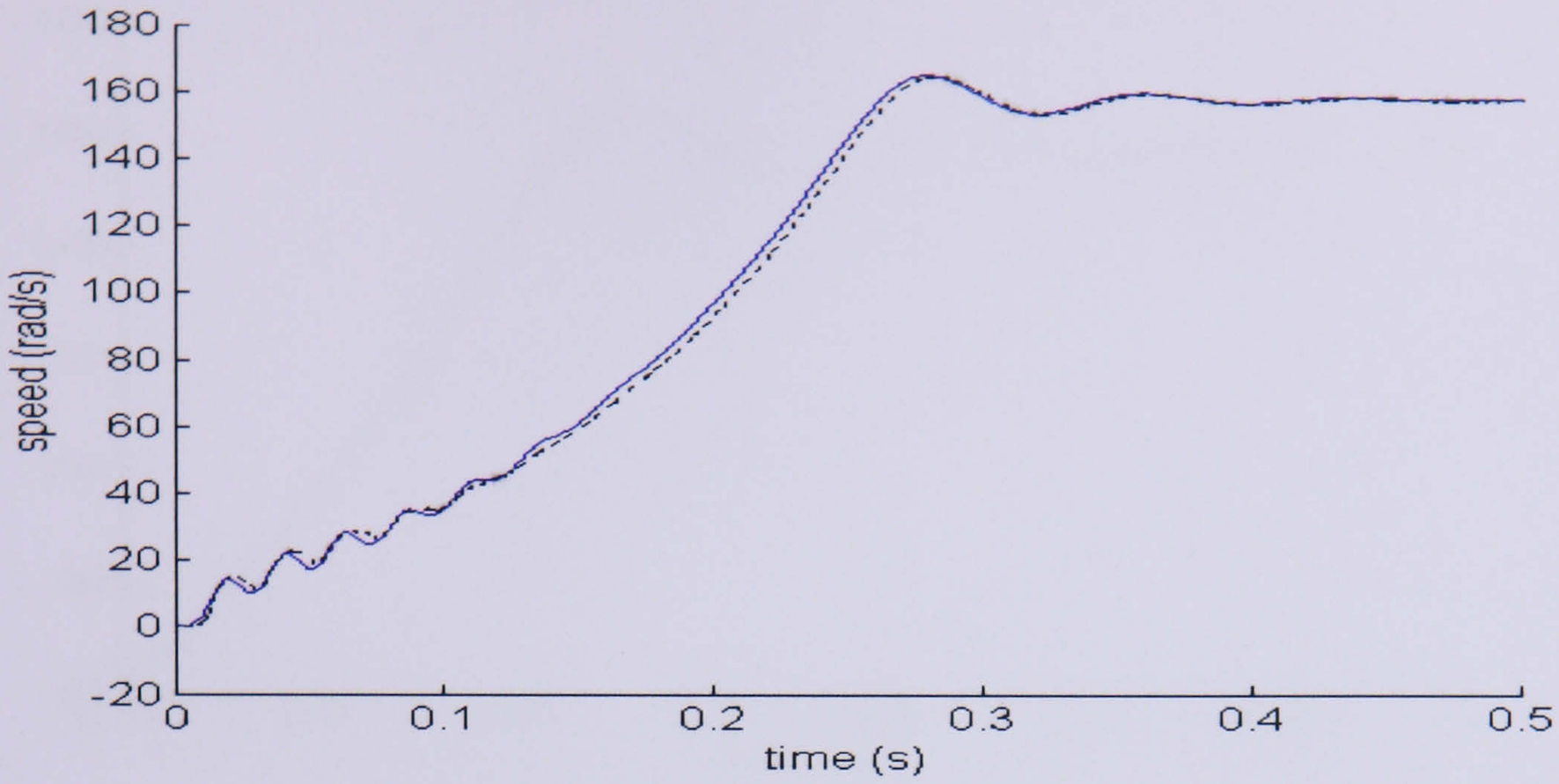
Figure 8.8: Error between estimated and reference speed in enlarged scale

8.5 Comparison on the Constant V/f of IM

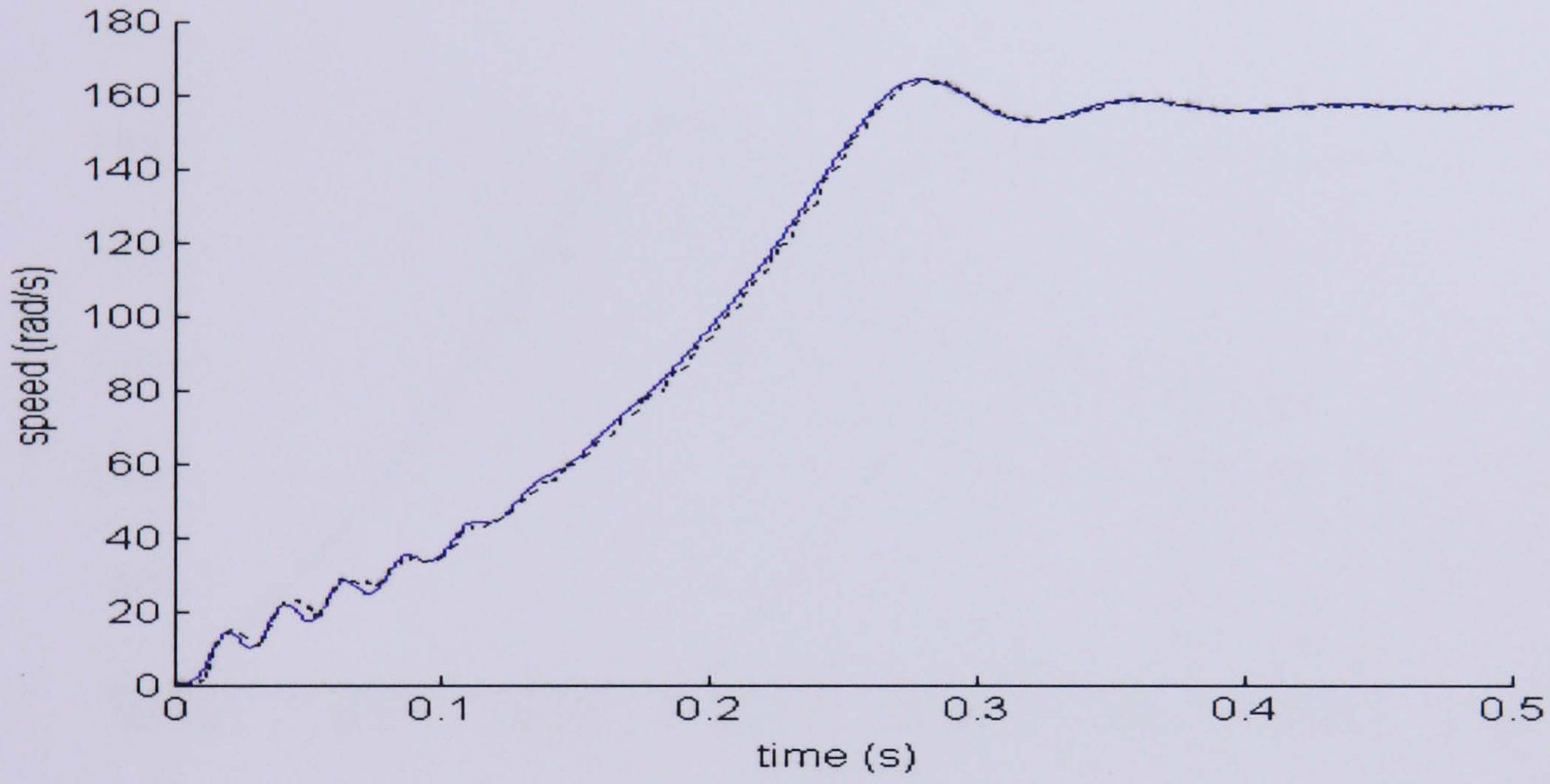
The performance of the EKF at various speed conditions for the open loop constant V/f controller and VC of the IM with the matrices tuned by Trial and Error method, SA and GA are compared in this section. Using the same control cycle, the mean squared speed error obtained for Trial and Error, SA and GA are 1.05, 0.57 and 0.77 respectively. These are very low values and would give good estimation results for rotor speed at the nominal condition.

8.5.1 Several Dynamic Speed Conditions

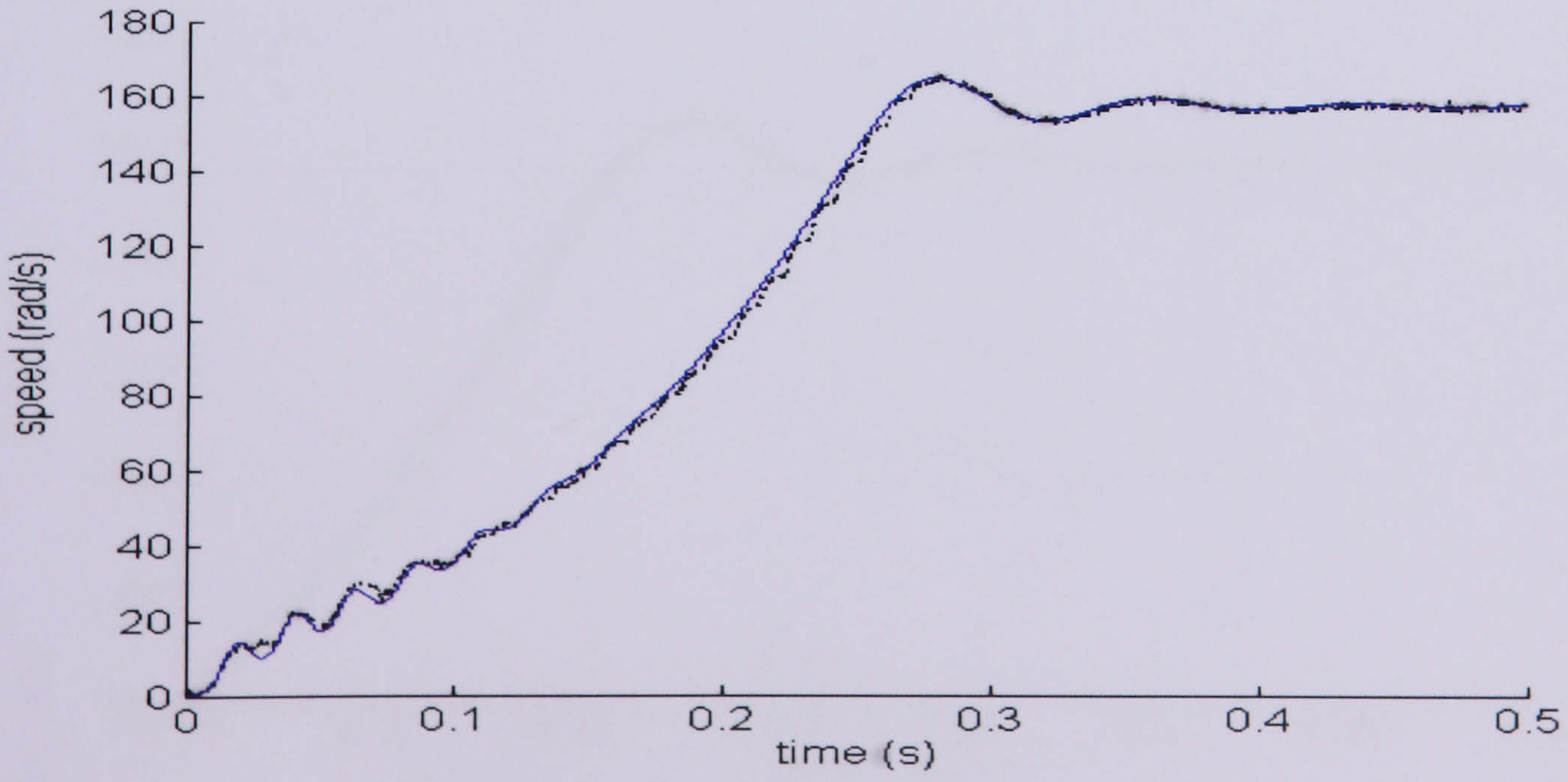
For various speed conditions, the reference and estimated speed using these three methods are shown in Figure 8.9-8.13. Figure 8.9 shows the results of the optimised EKF for transient and steady state conditions. Figure 8.10 and Figure 8.11 show the results for an acceleration to 160 rad/s and a speed reversal from 6 rad/s to -53 rad/s respectively. Finally, an acceleration from -30 rad/s to -52 rad/s and a deceleration to -58 rad/s is shown in Figure 8.12 and Figure 8.13. The solid line shows the actual speed while the dotted line shows the estimated speed. The results show that SA, GA and Trial and Error methods are all capable of estimating the speed at different conditions including standstill. With little difference in the mean squared error, the performance of the SA is as good as the Trial and Error in almost all conditions and is even better during acceleration from -30 rad/s to -52 rad/s as shown in Figure 8.12.



(a)

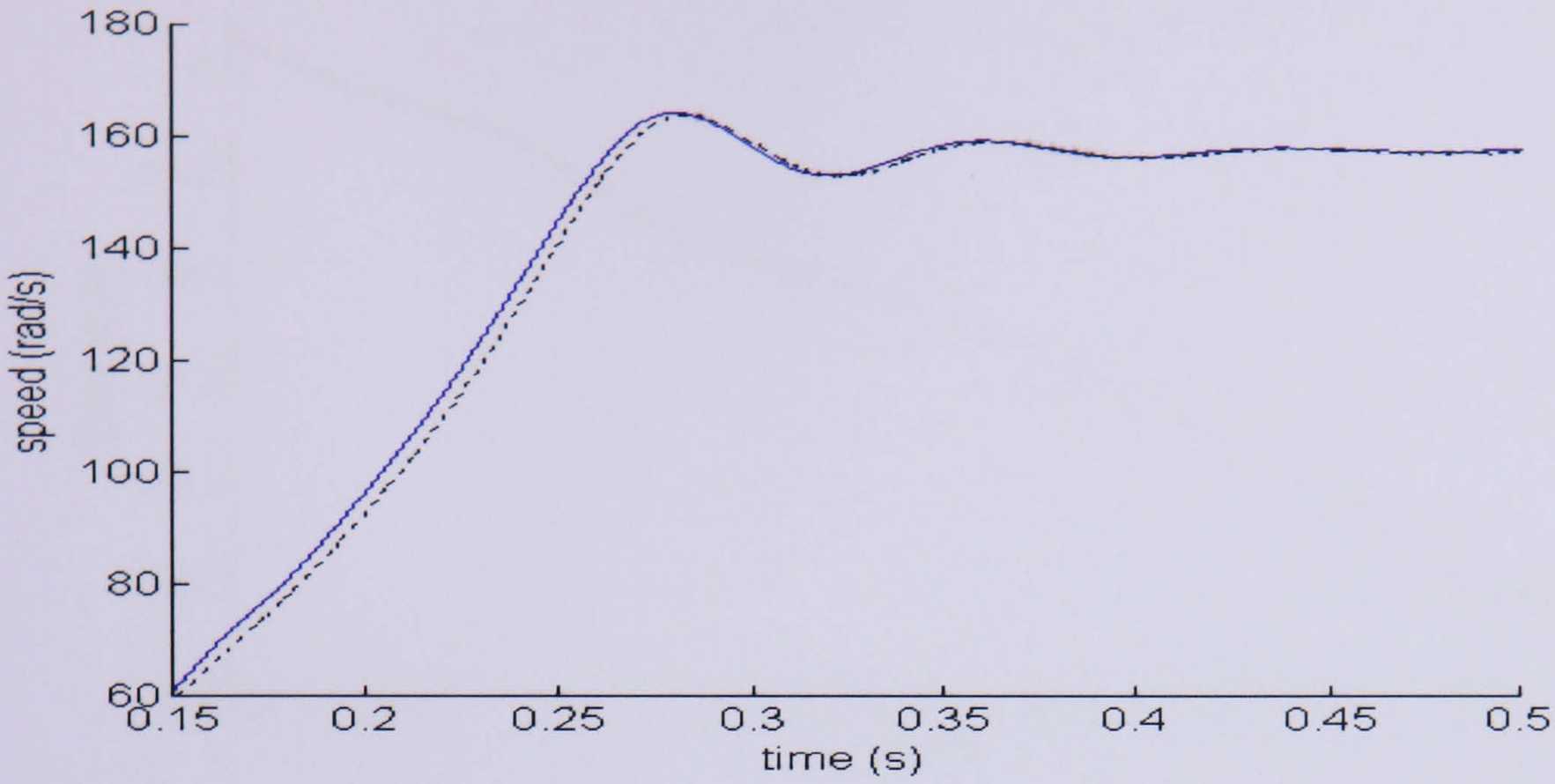


(b)

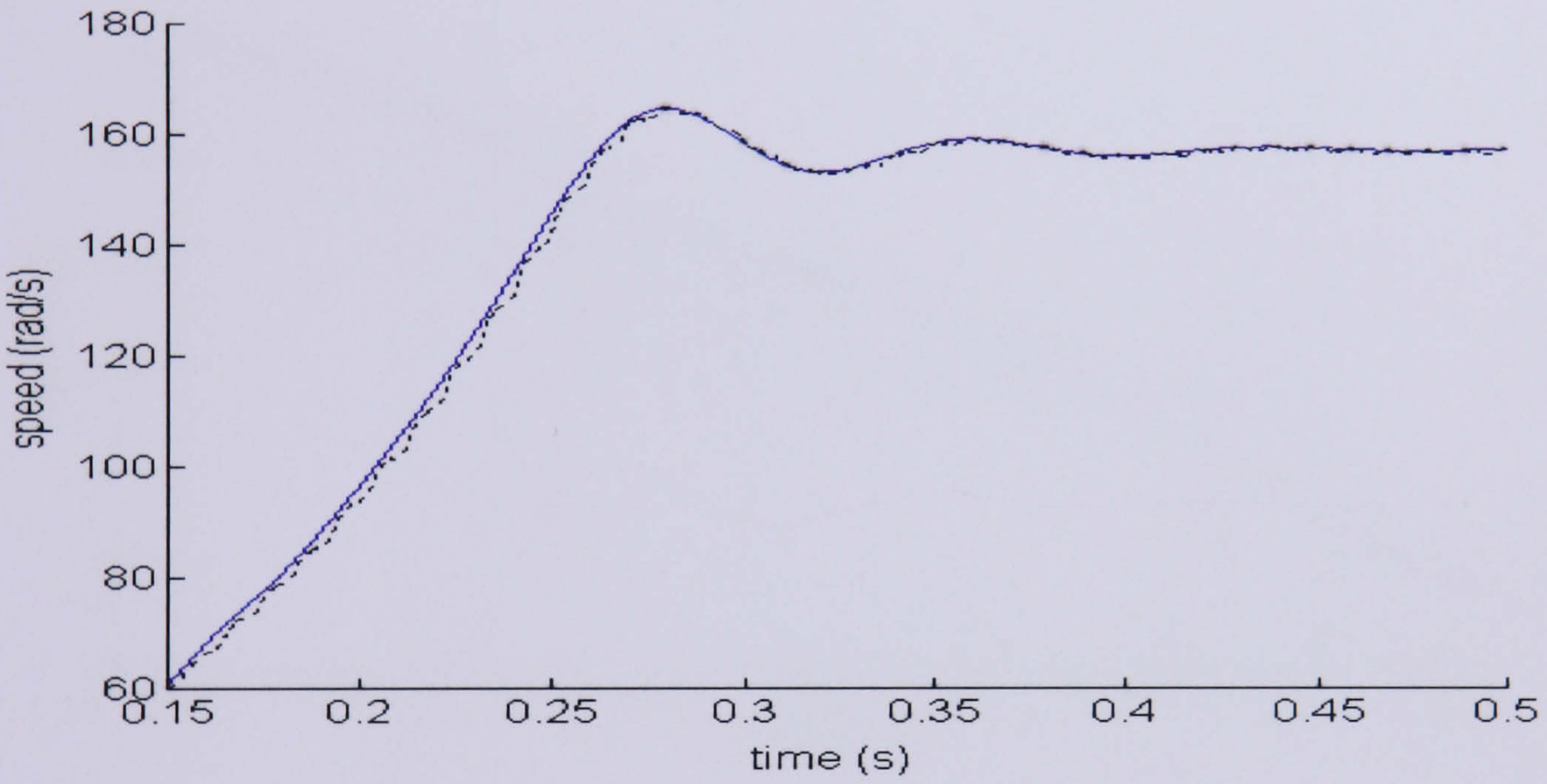


(c)

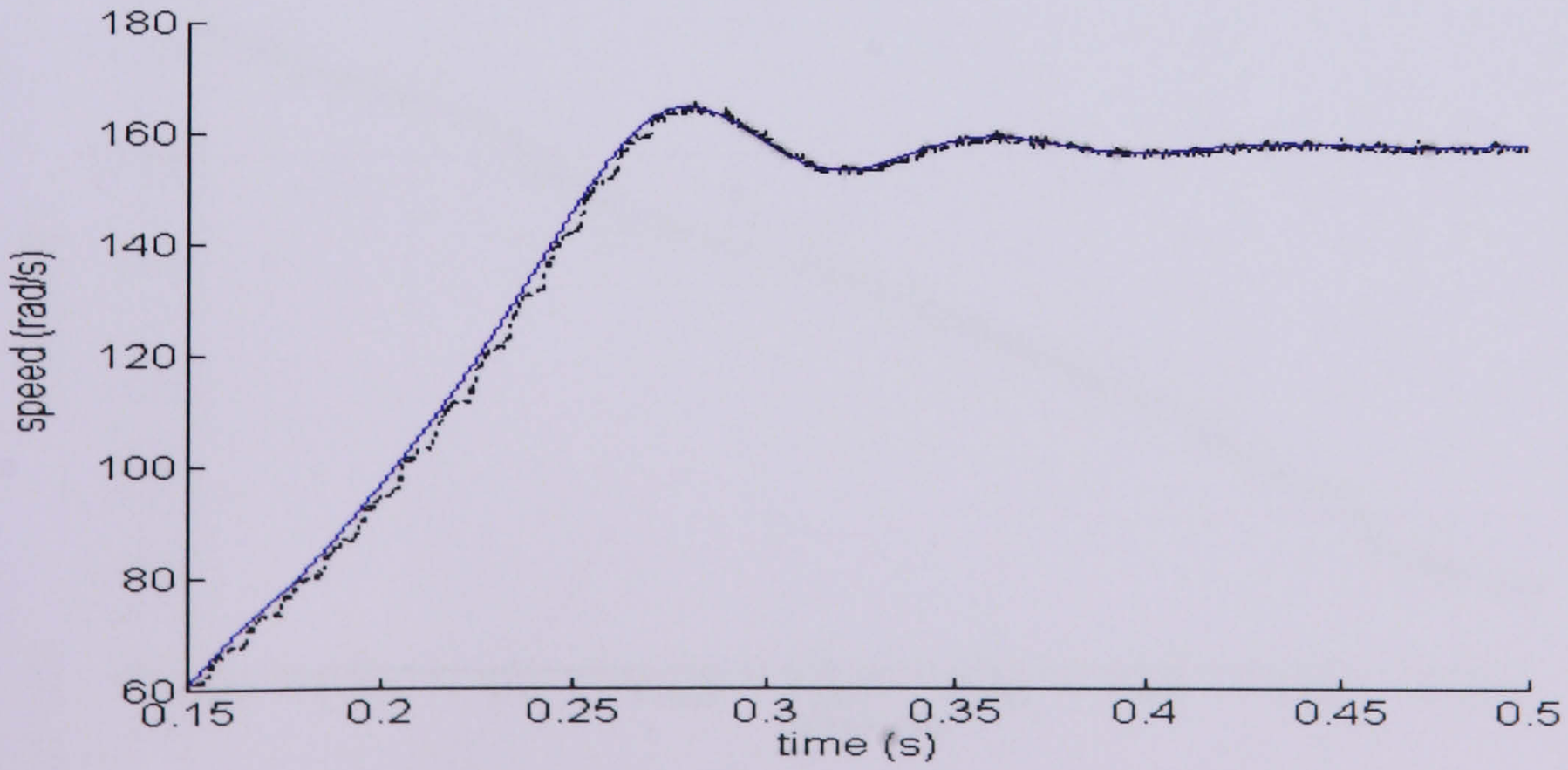
Figure 8.9 The transient and steady state tuned by (a) Trial and Error, (b) SA and (c) GA respectively.



(a)

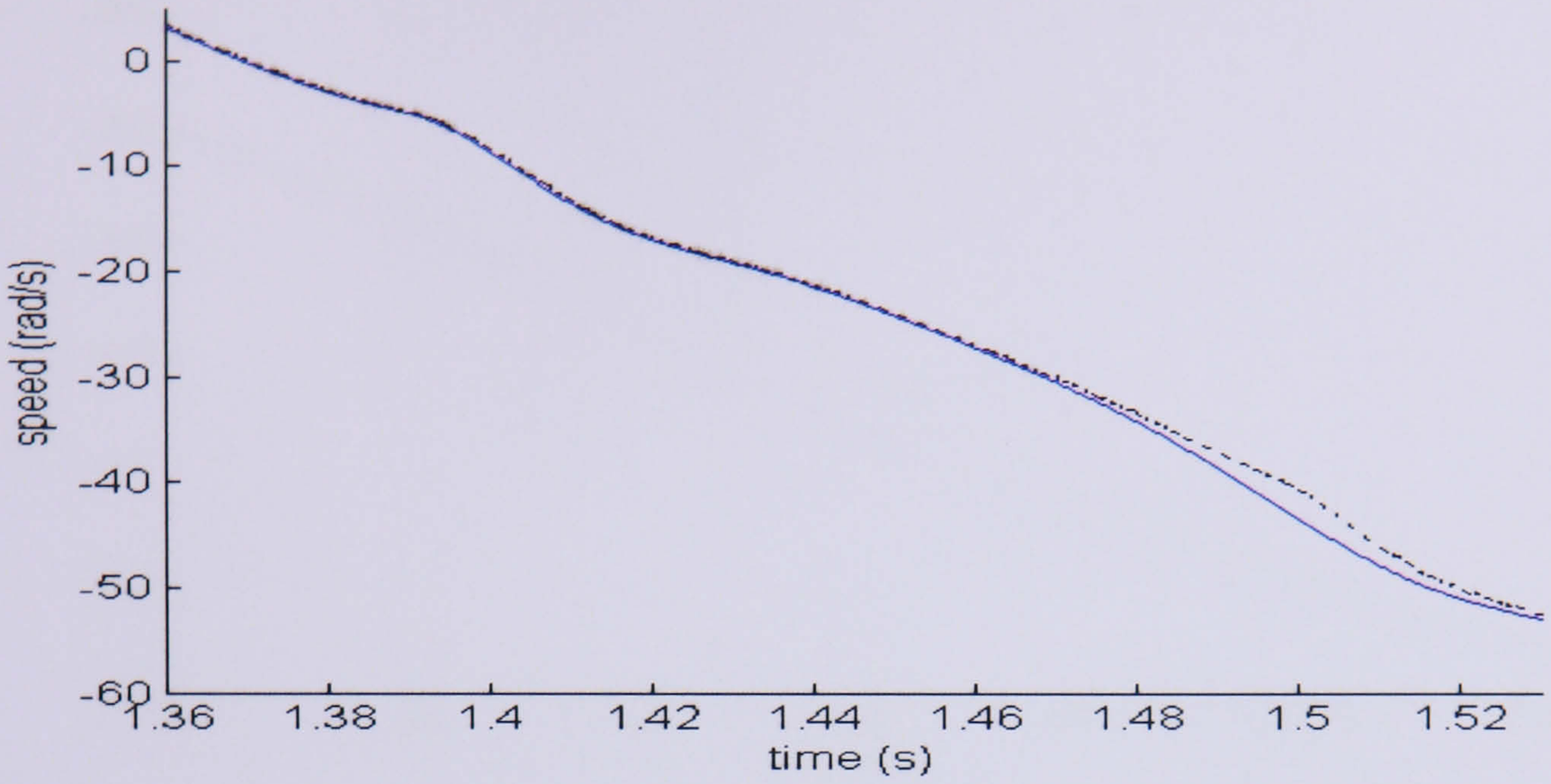


(b)

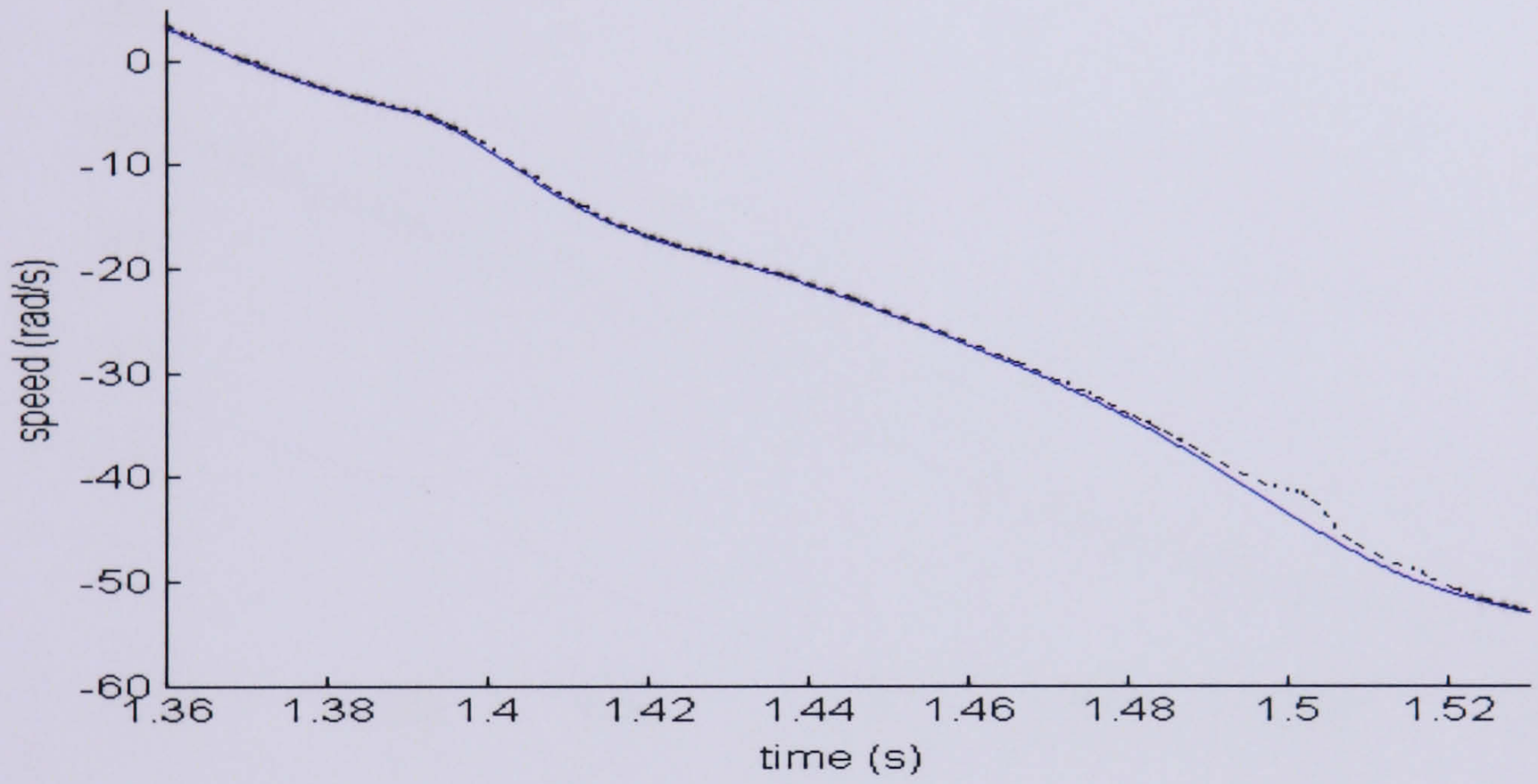


(c)

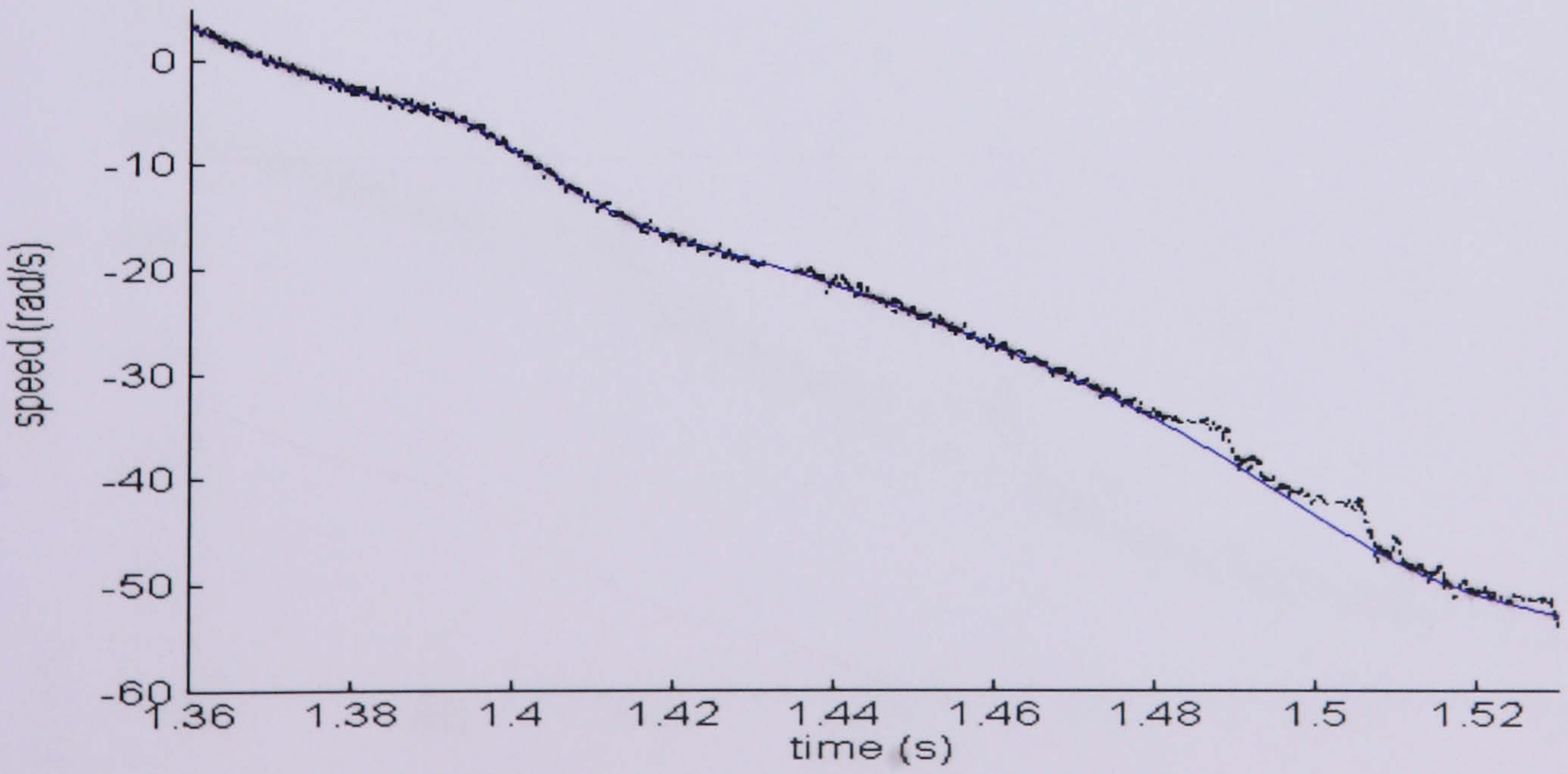
Figure 8.10: Acceleration to 160 rad/s tuned by (a) Trial and Error, (b)SA and (c) GA respectively



(a

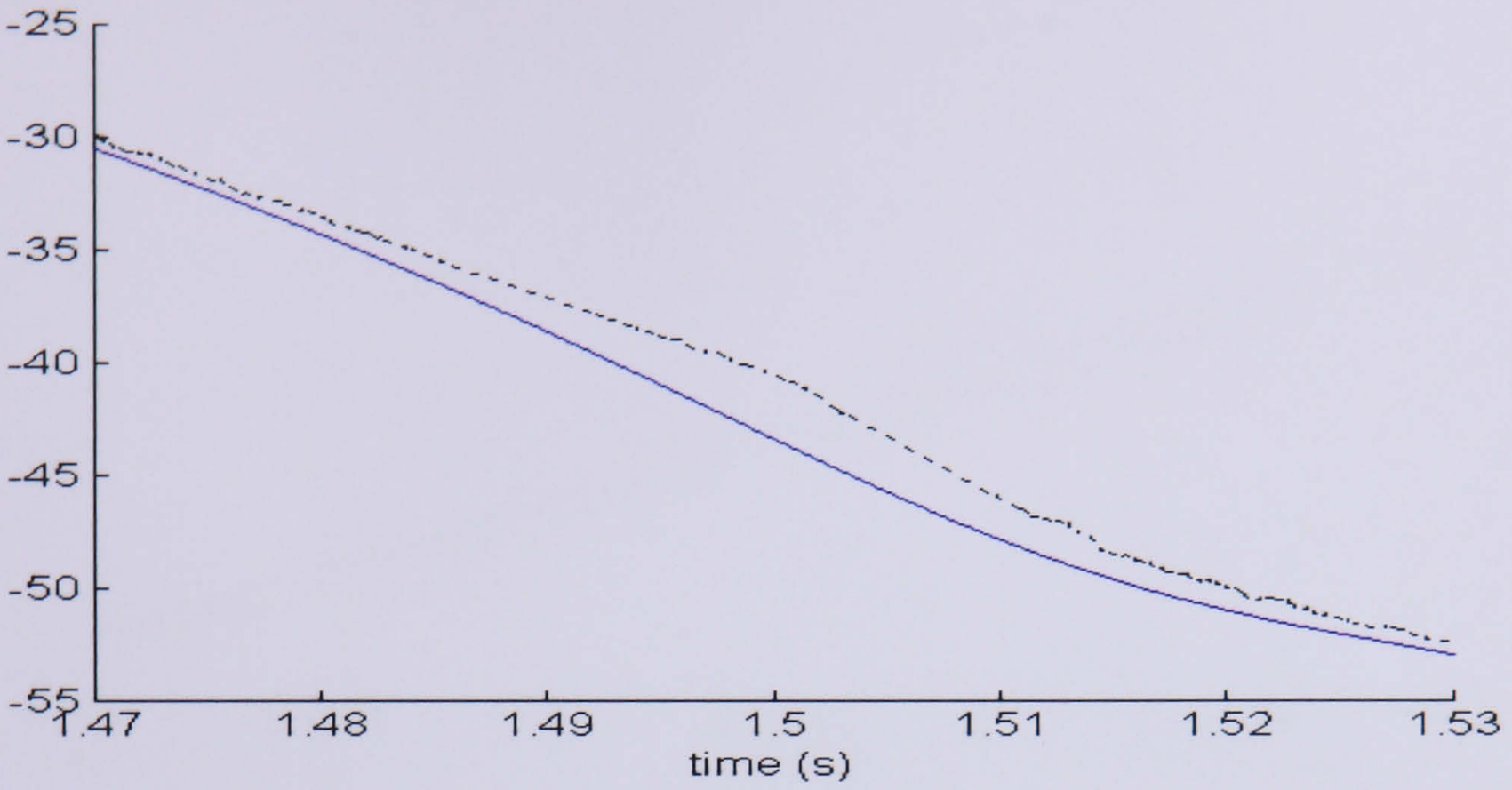


(b

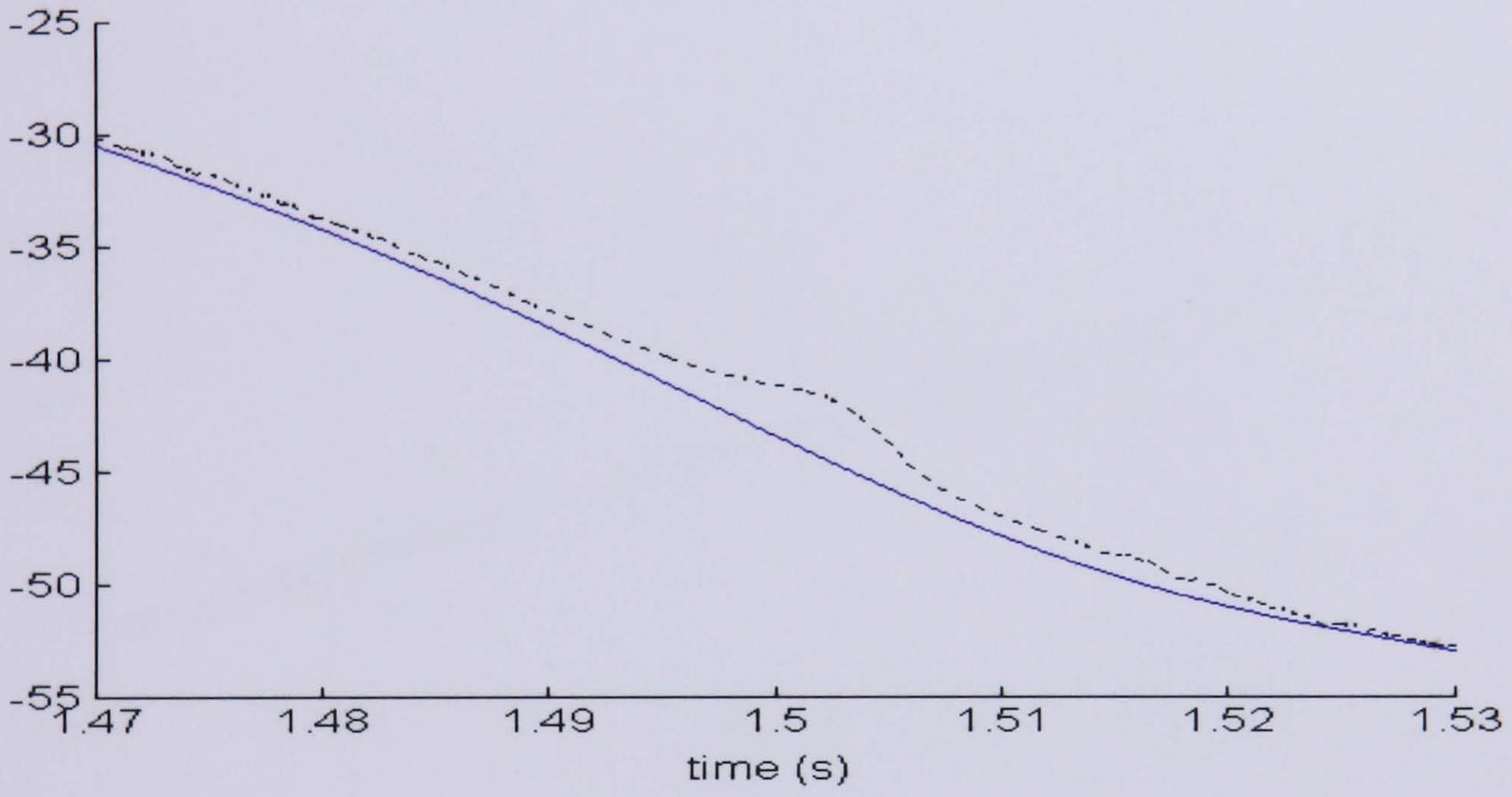


(c)

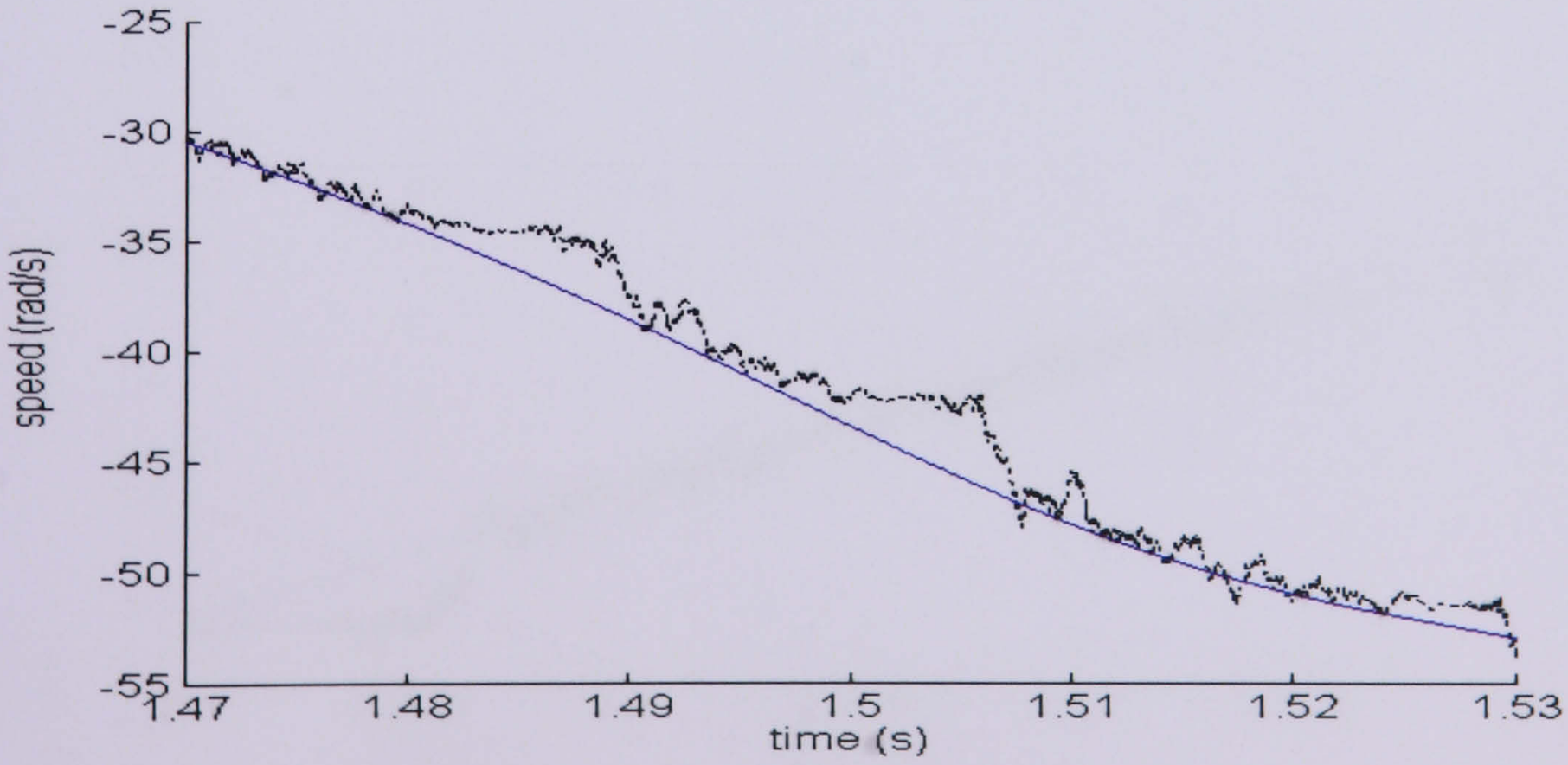
Figure 8.11: Speed Reversal from 6 rad/s to -53 rad/s tuned by (a) Trial and Error, (b) SA and (c) GA respective



(a)

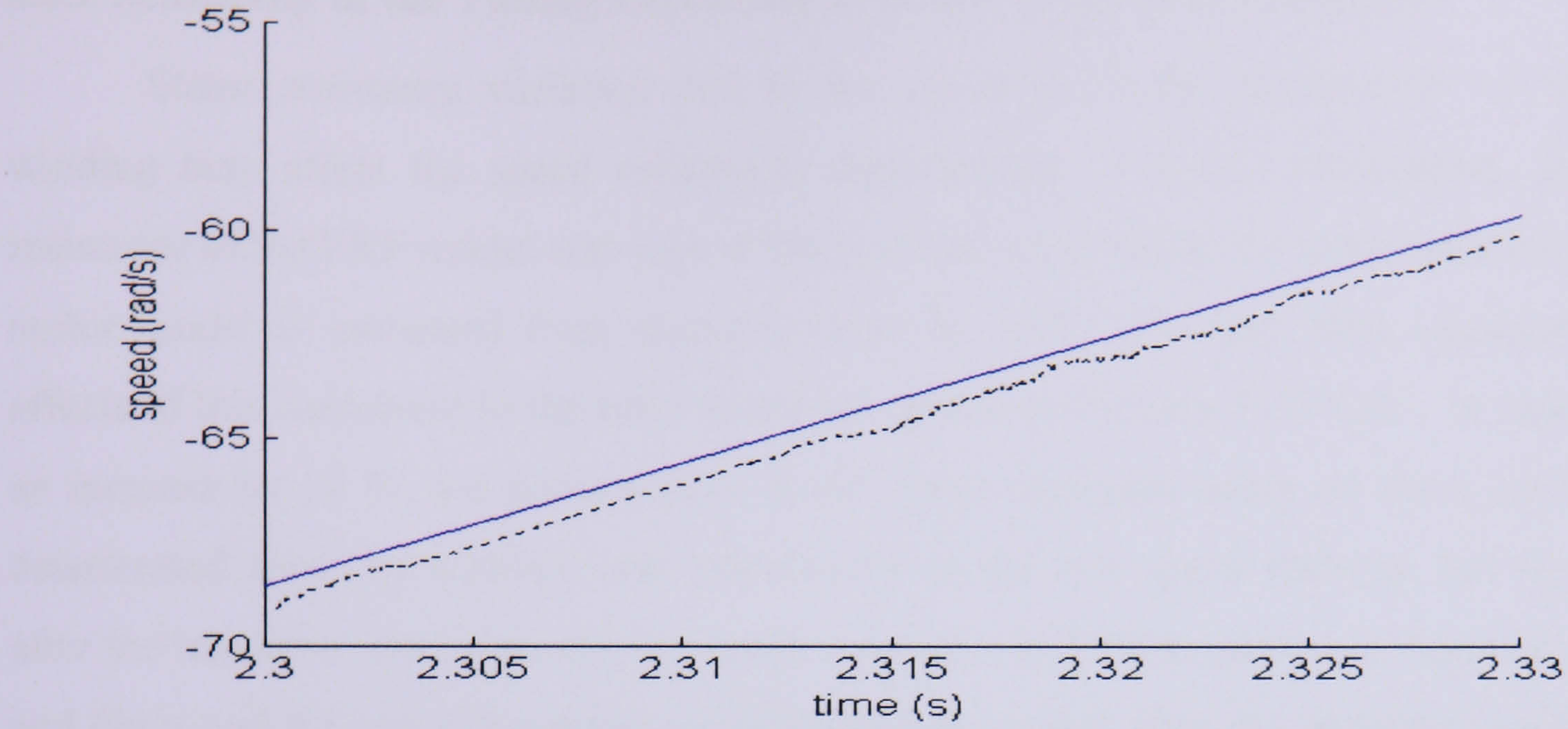


(b)

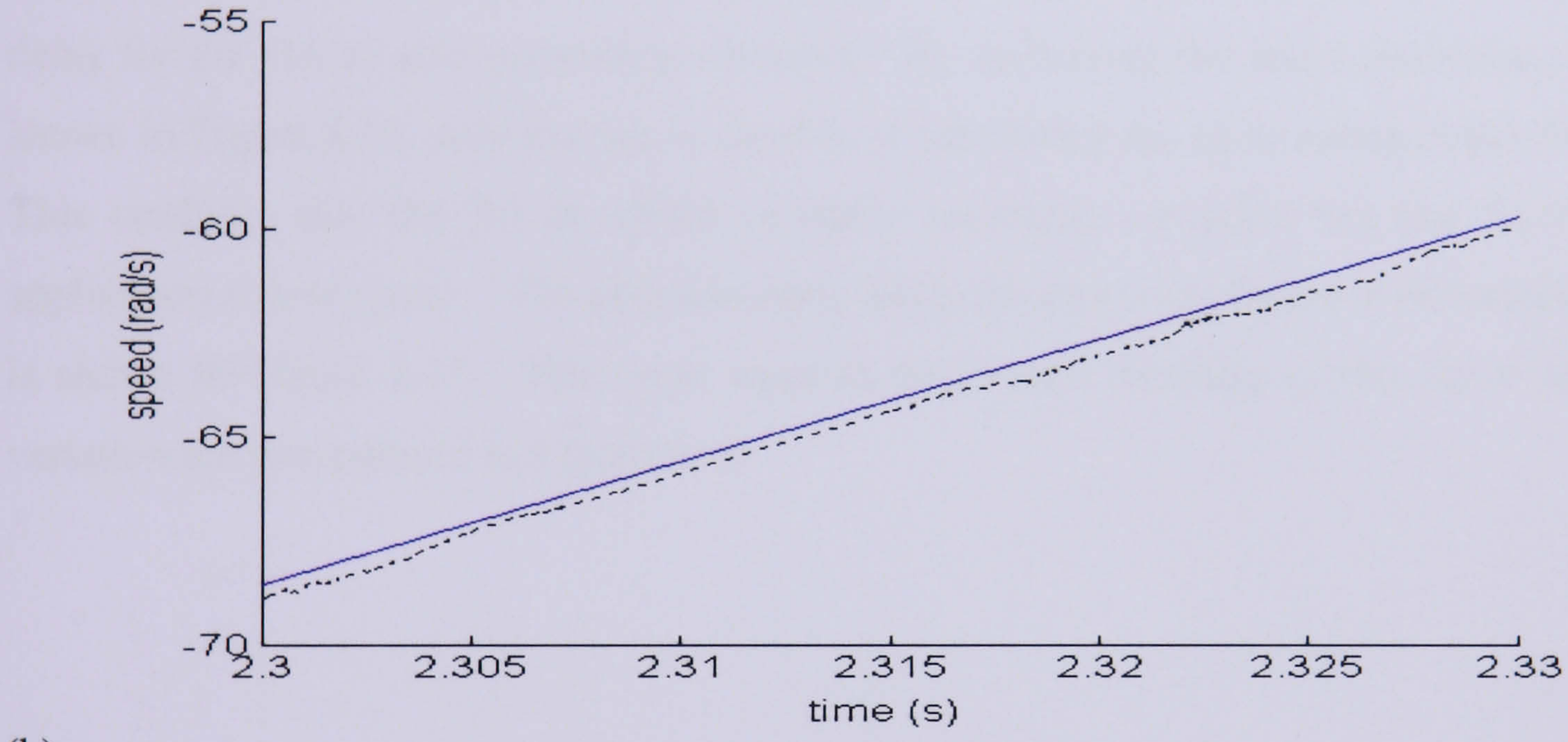


(c)

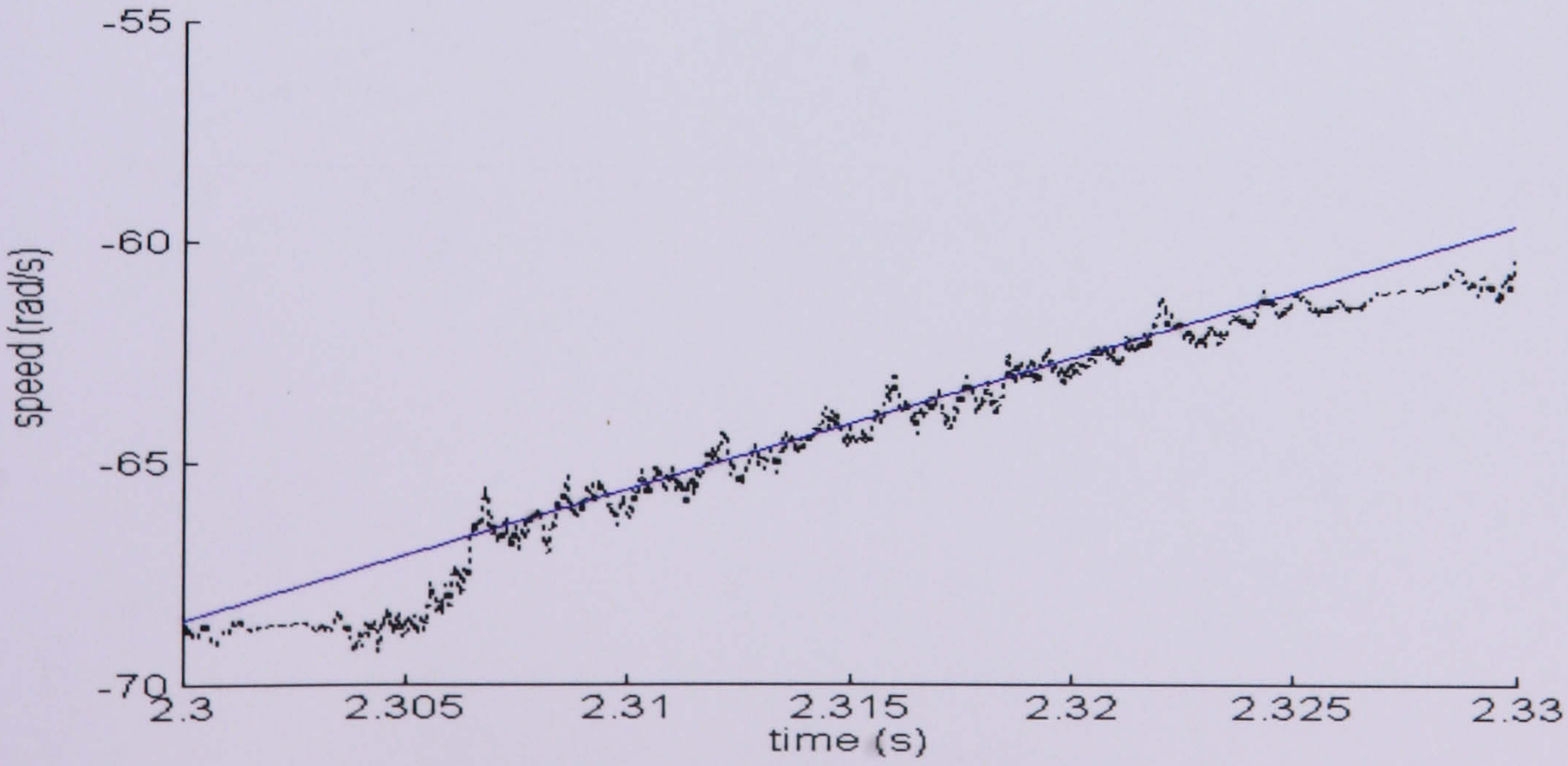
Figure 8.12: Acceleration from -30 rad/s to -52 rad/s tuned by (a) Trial and Error, (b) SA and (c) GA respectively



(a)



(b)

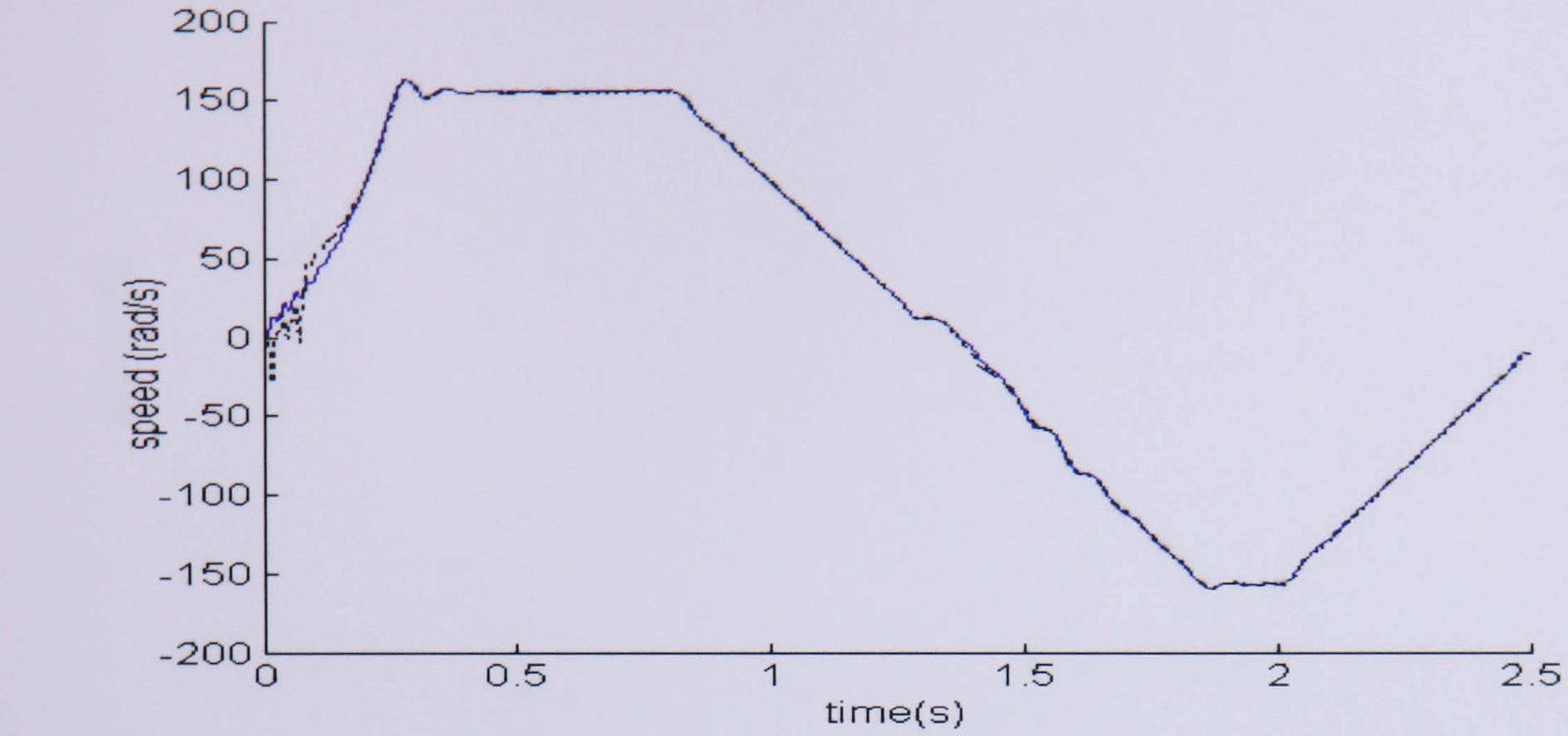


(c)

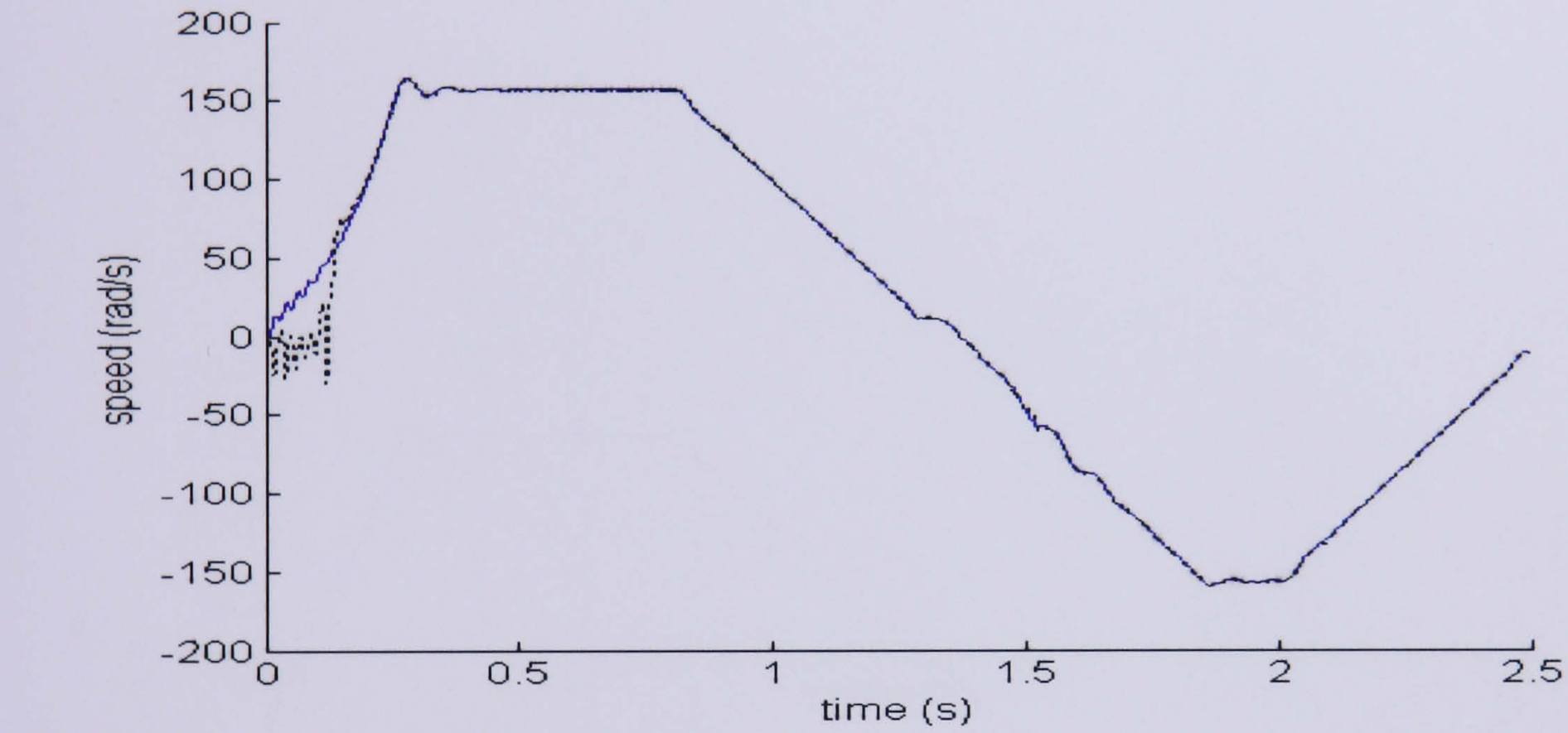
Figure 8.13: Deceleration to -58 rad/s tuned by (a) Trial and Error, (b) SA and (c) GA respectively

8.5.2 Sensitivity of the Tuning Parameter to Stator Resistance Variation

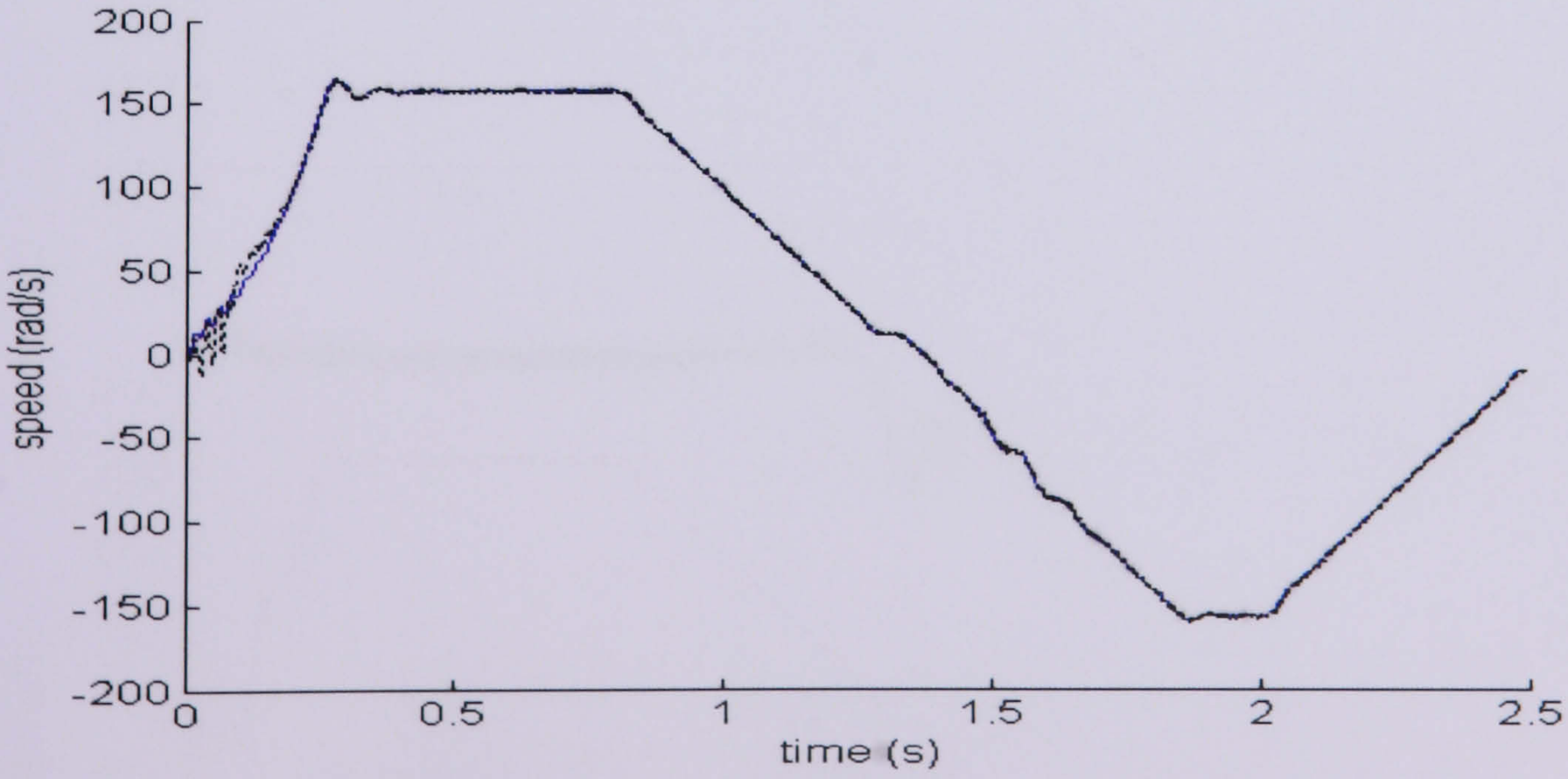
Stator resistance variation due to the variation of the temperature of the stator winding may affect the speed estimation significantly. For this verification, the motor resistance of the EKF model was kept at the nominal value while the stator resistance for the motor model is increased from nominal value by 10%, 25% and 50% separately. The affects of this increment to the rotor speed are shown in Figures 8.14-8.18. In Figure 8.14, an increase by 10 %, the performance of the speed estimator using all three methods has deteriorated from the nominal case particularly in the low speed start-up, but works well after the transient. By increasing the stator resistance to 25% as shown in Figure 8.15, Trial and Error and SA are still capable of estimating the speed after the transient condition but only start to estimate properly when they get a 'second chance'. It requires about a 1.5 s delay for the GA to start estimating correctly. By increasing the stator resistance by 50%, shown in Figure 8.16, only the SA is capable of estimating the rotor speed in the first 1.5 s. This confirms that the SA is robust to stator resistance variation but has limits on its application at low speed. The absolute error when the stator resistance is increased by 10% is shown in Figure 8.17. The mean squared errors as a function of the stator resistance variation are summarised in Figure 8.18.



(a)

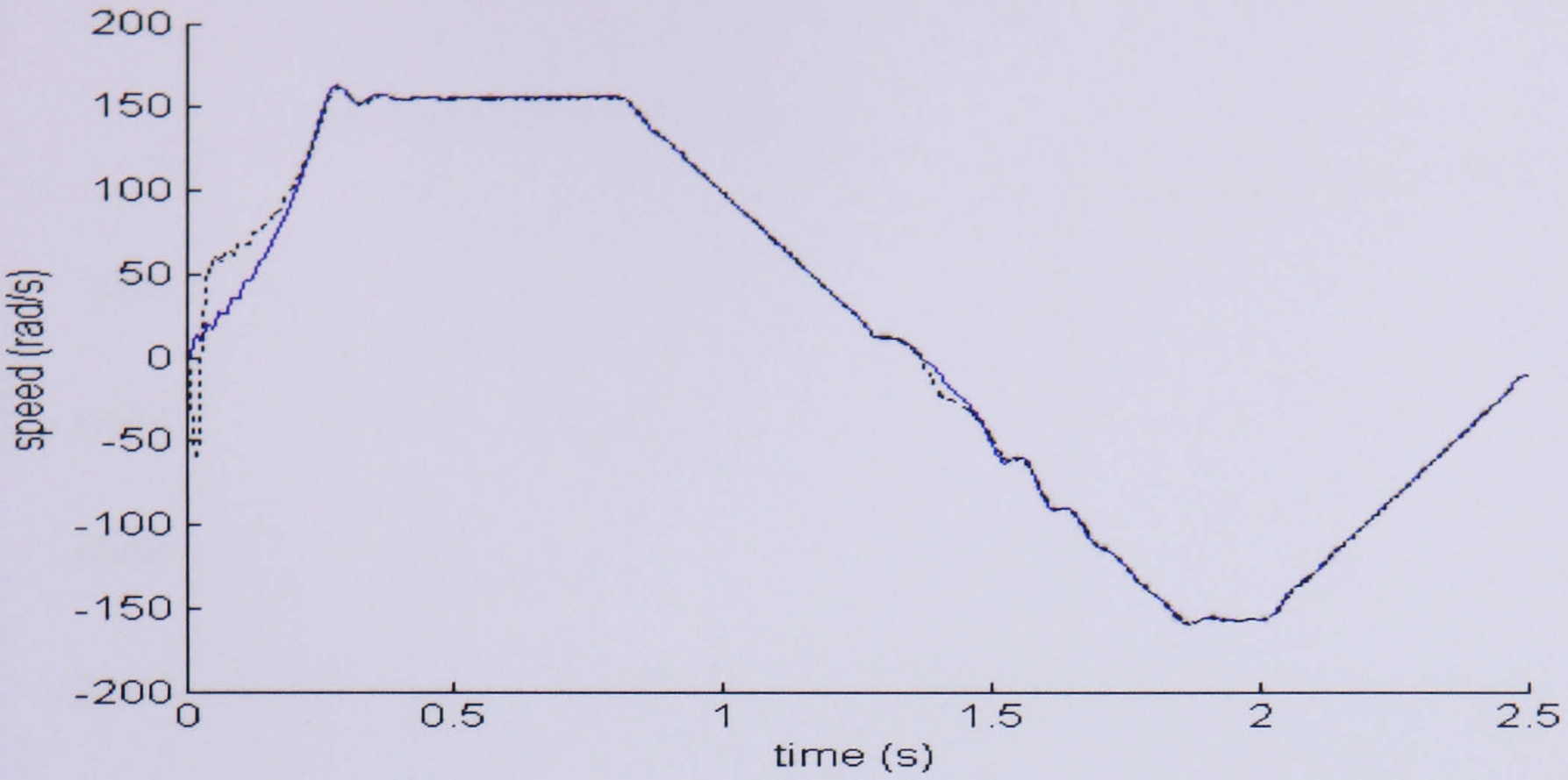


(b)

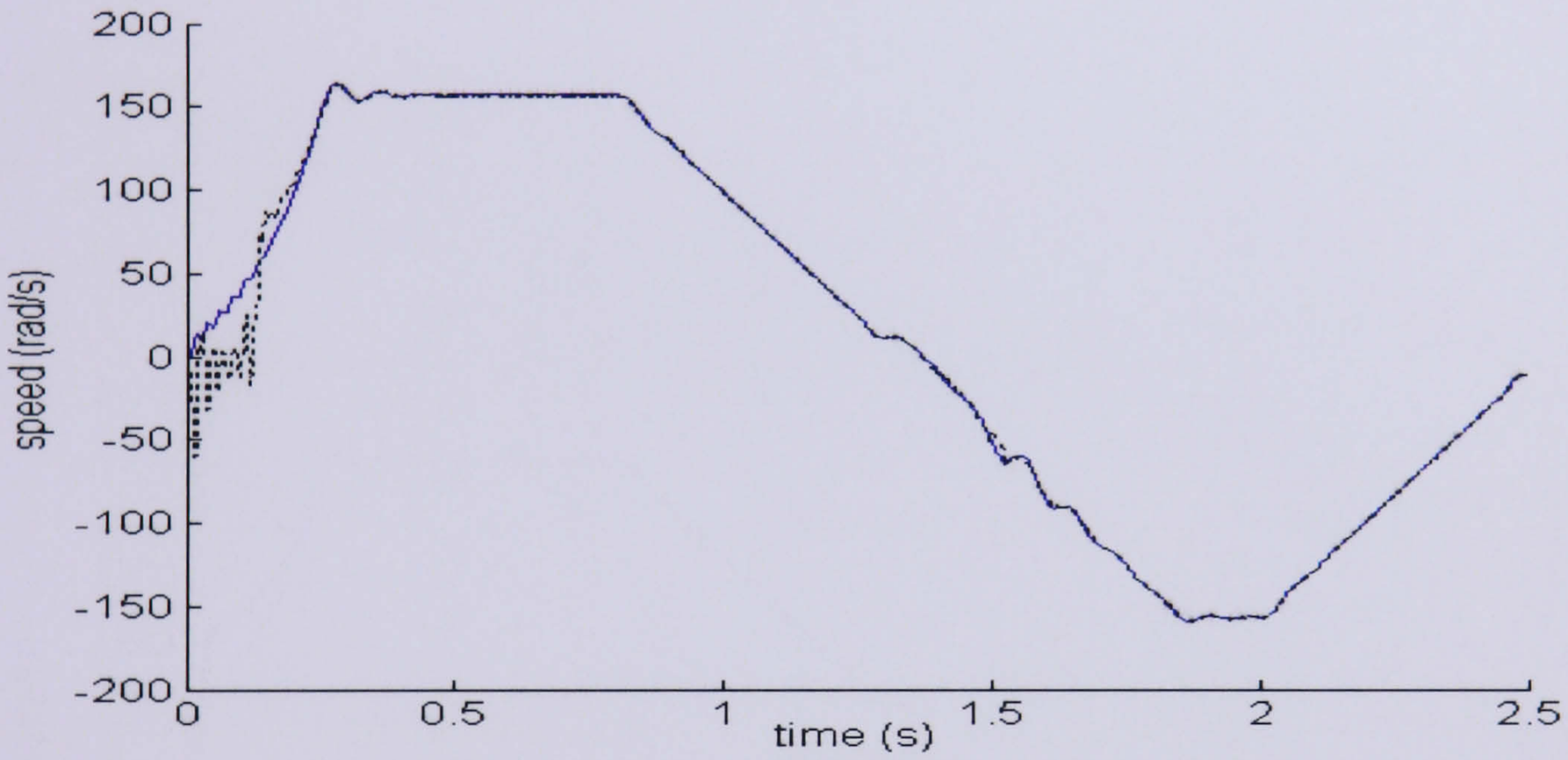


(c)

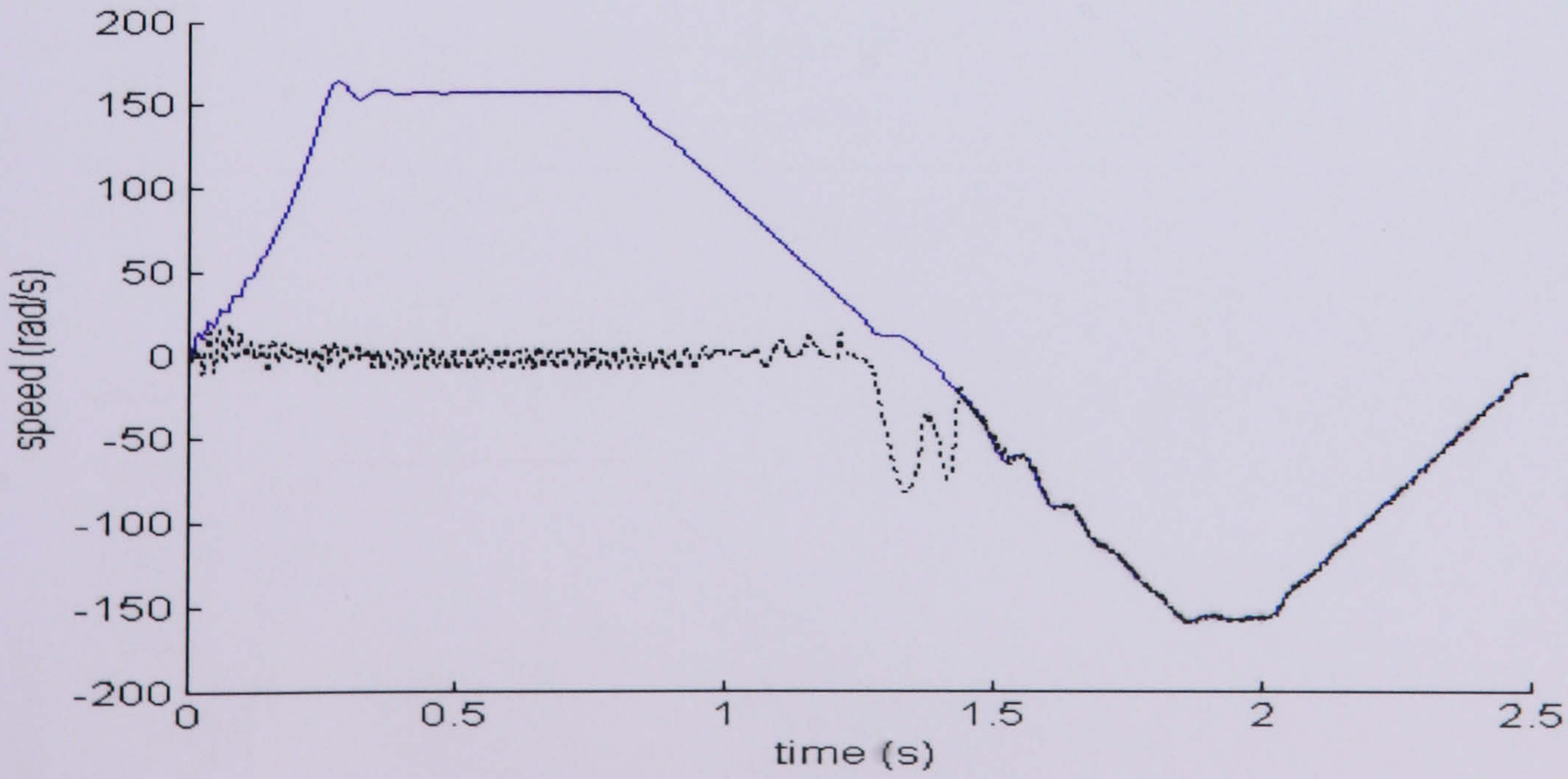
Figure 8.14: Speed comparison using (a) Trial and Error, (b) SA and (c) GA respectively when the stator resistance of the motor model is increased by 10%



(a)

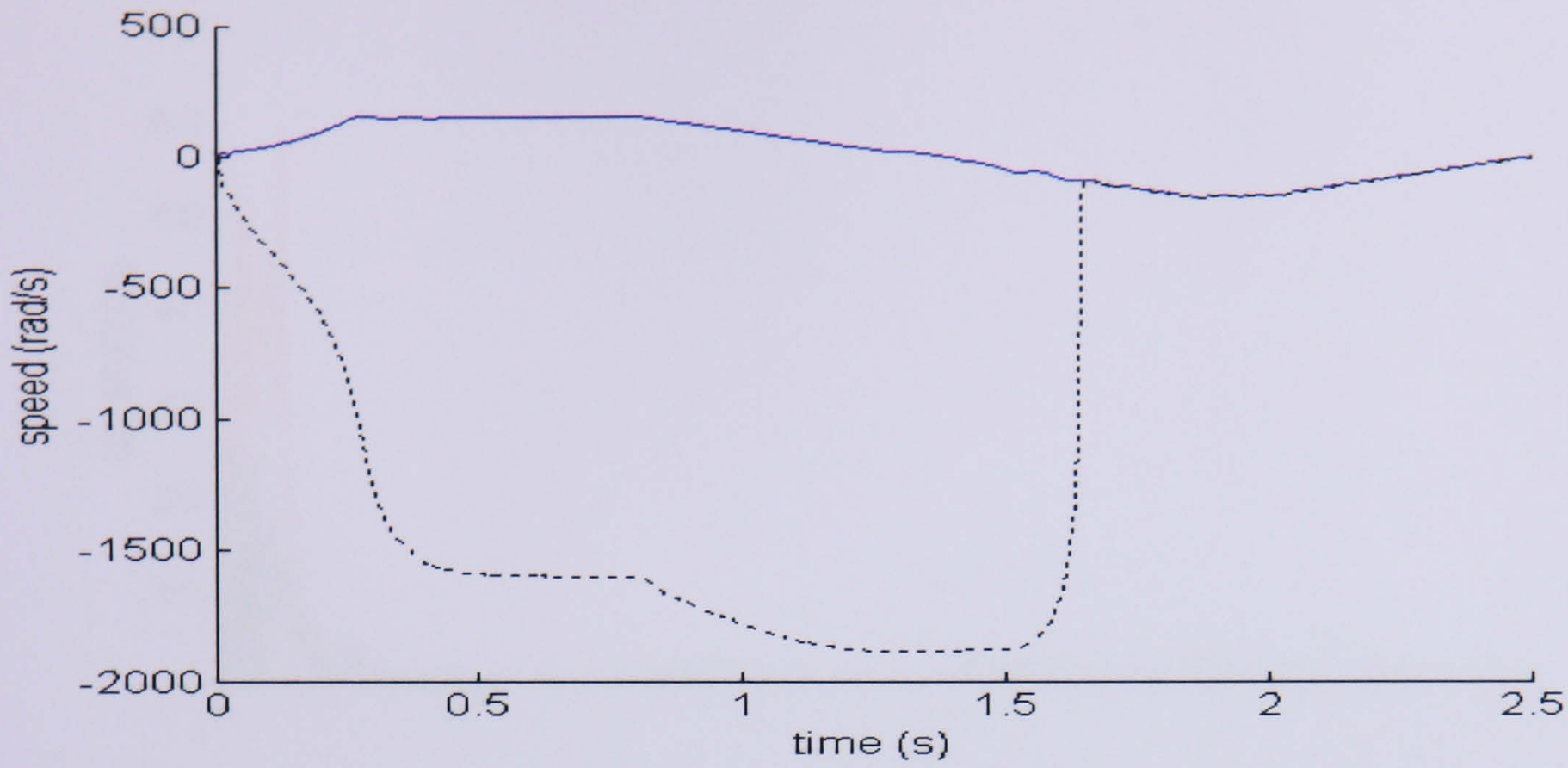


(b)

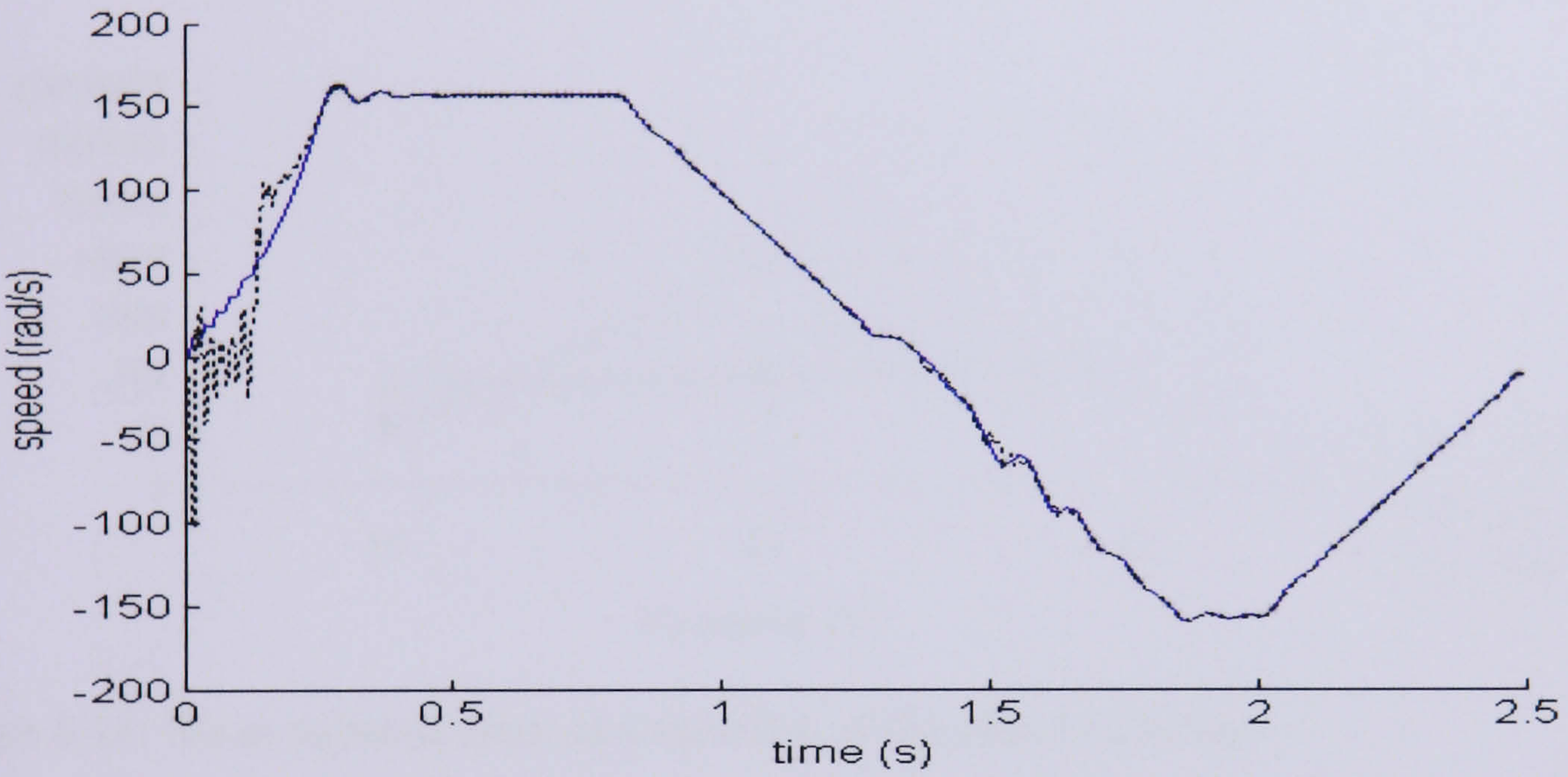


(c)

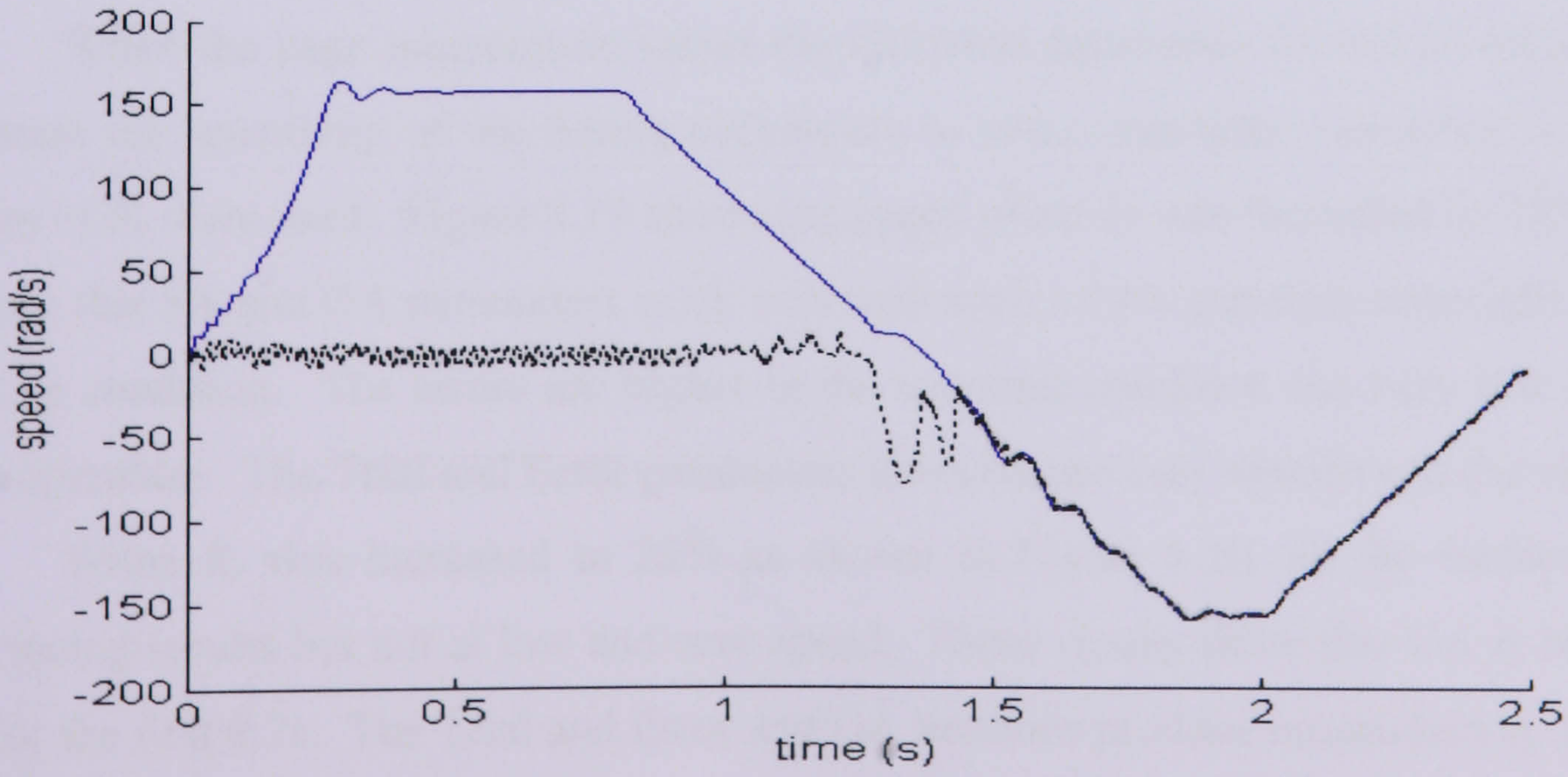
Figure 8.15: Speed comparison using (a) Trial and Error, (b) SA and (c) GA respectively when the stator resistance of the motor model is increased by 25%



(a)



(b)



(c)

Figure 8.16: Speed comparison using (a) Trial and Error, (b) SA and (c) GA respectively when the stator resistance of the motor model is increased by 50%

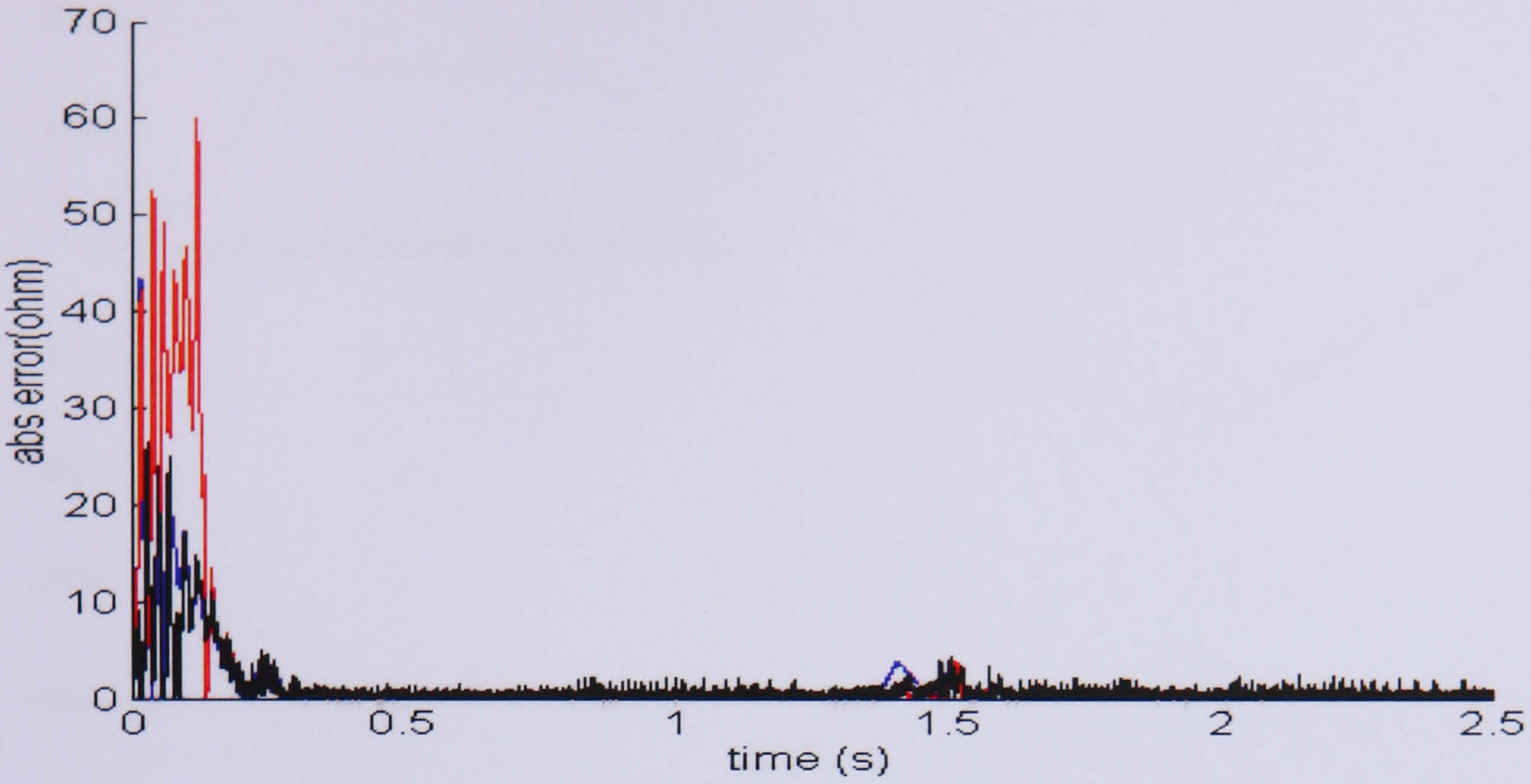


Figure 8.17: Absolute speed error when the stator resistance is increased by 10%

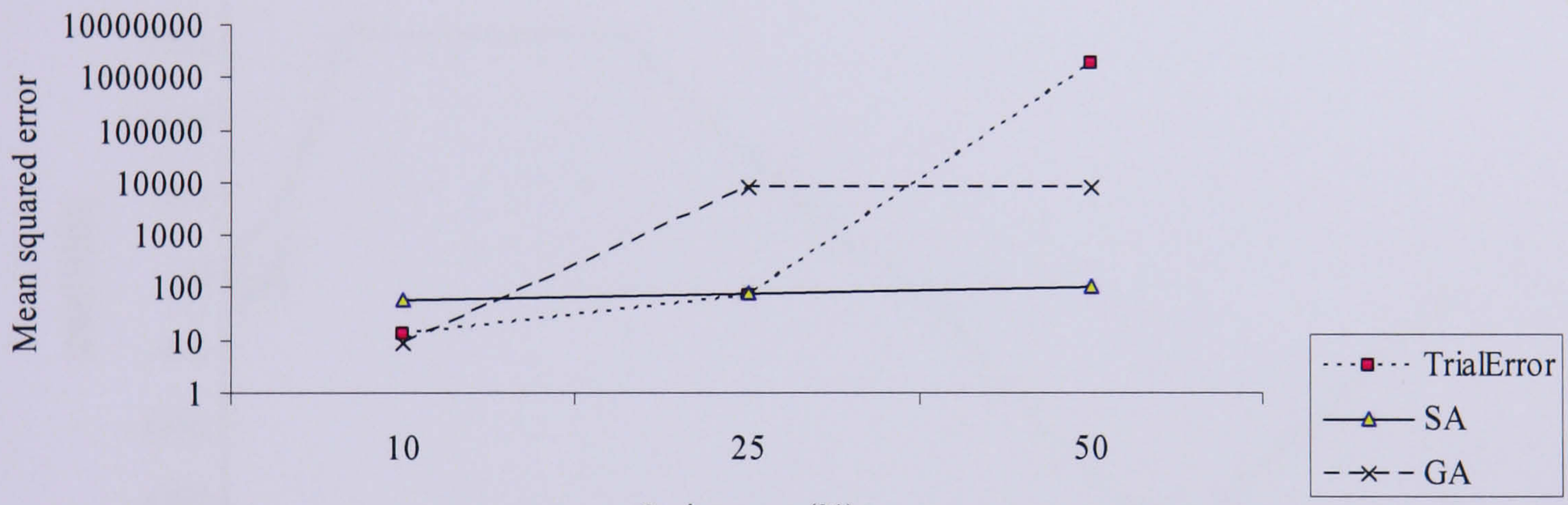
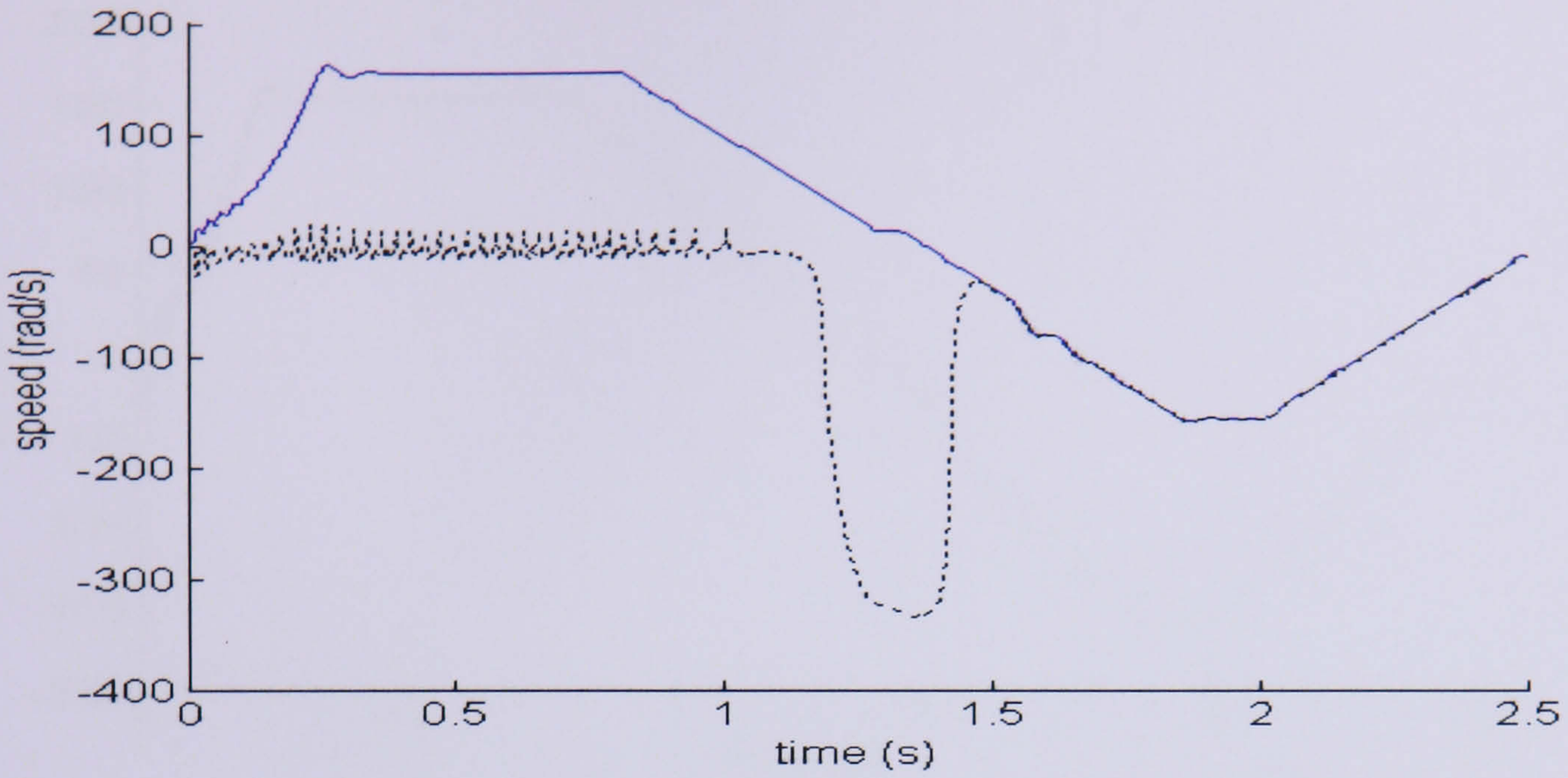


Figure 8.18: Mean squared error as a function of the stator resistance

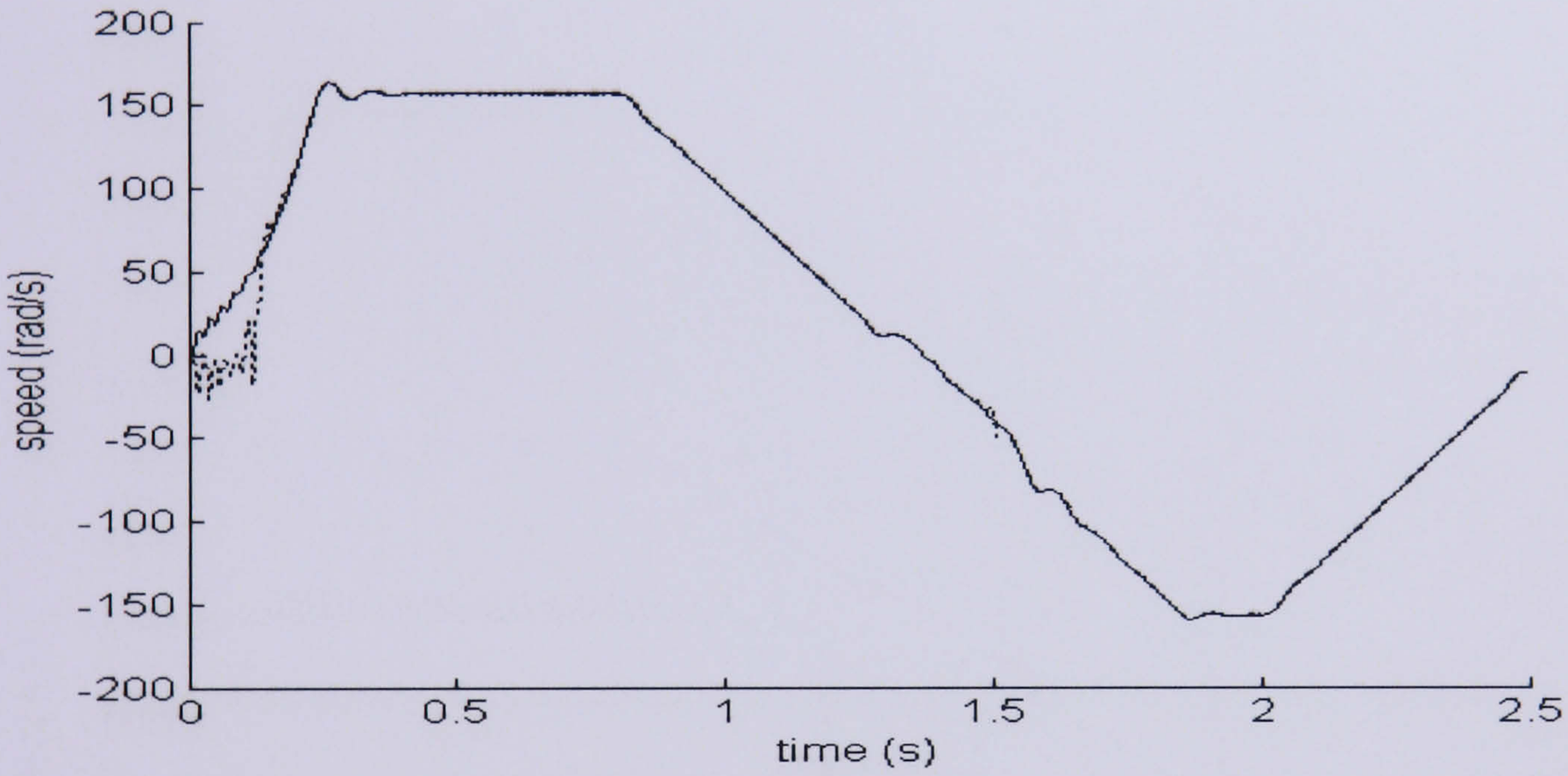
8.5.3 Sensitivity of the Tuning Parameter to Rotor Resistance Variation

When the cage temperature varies the electrical parameter R_r will be affected. To illustrate the sensitivity of the tuning parameters to rotor resistance variations, 3 different values of R_r were used. Figure 8.19 shows the speed when R_r was increased by 10%. It can be seen that SA and GA parameters work well with such a 10% variation especially after the start up condition. The errors are higher in the transient condition but very low in steady state operation. The Trial and Error parameters however are very sensitive to the variation.

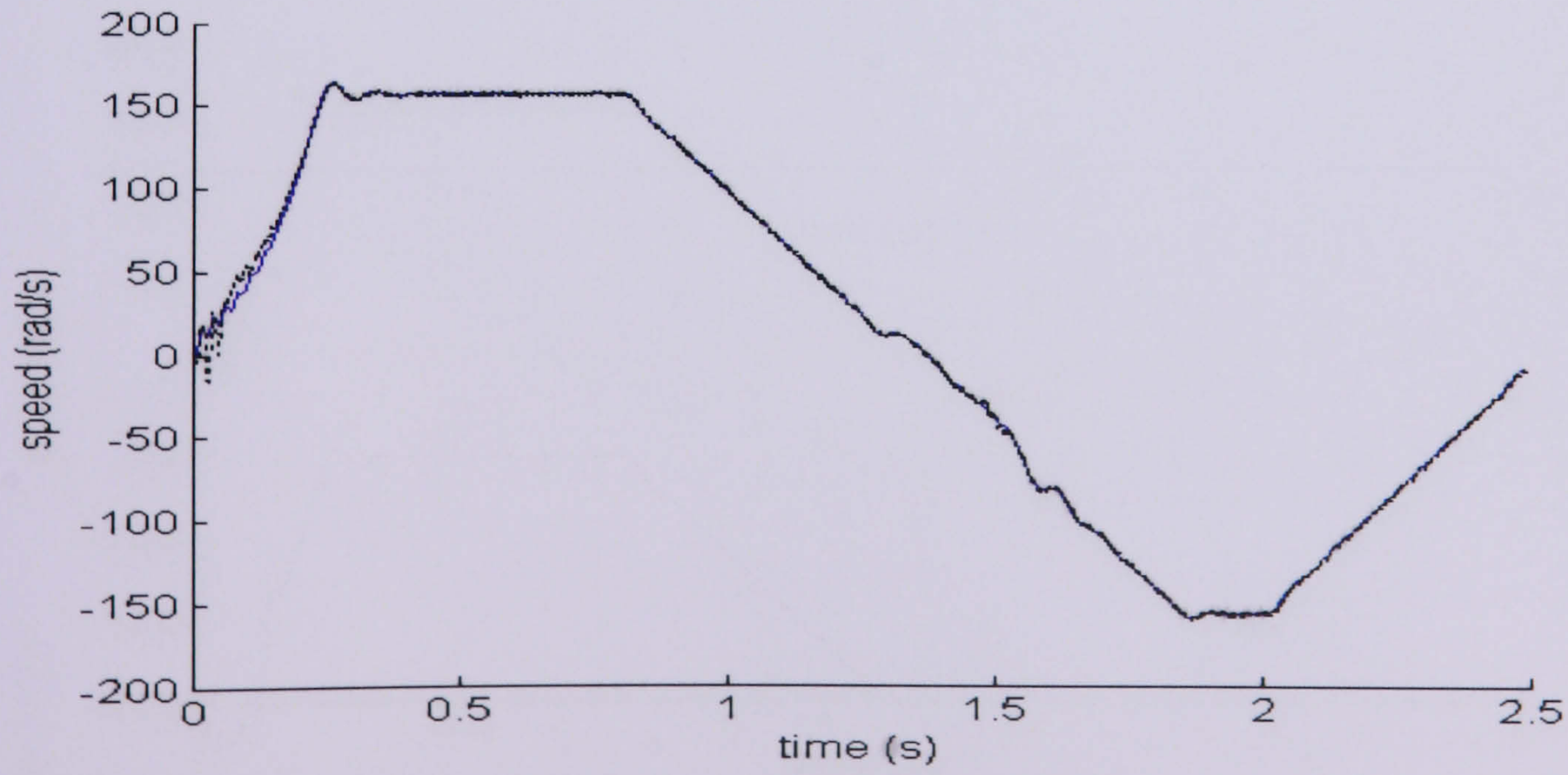
When R_r was increased to 25% as shown in Figure 8.20, all the methods shows convincing results but not at low and zero speed. These results show that GA is better than SA for the first 0.2s. The Trial and Error and GA methods produce unsatisfactory responses when estimating the speed with 50% increased in R_r as shown in Figure 8.21. The absolute speed error with an increase to 25% is shown in Figure 8.22. The mean squared errors as a function of the rotor resistance variation are summarised in Figure 8.23.



(a)

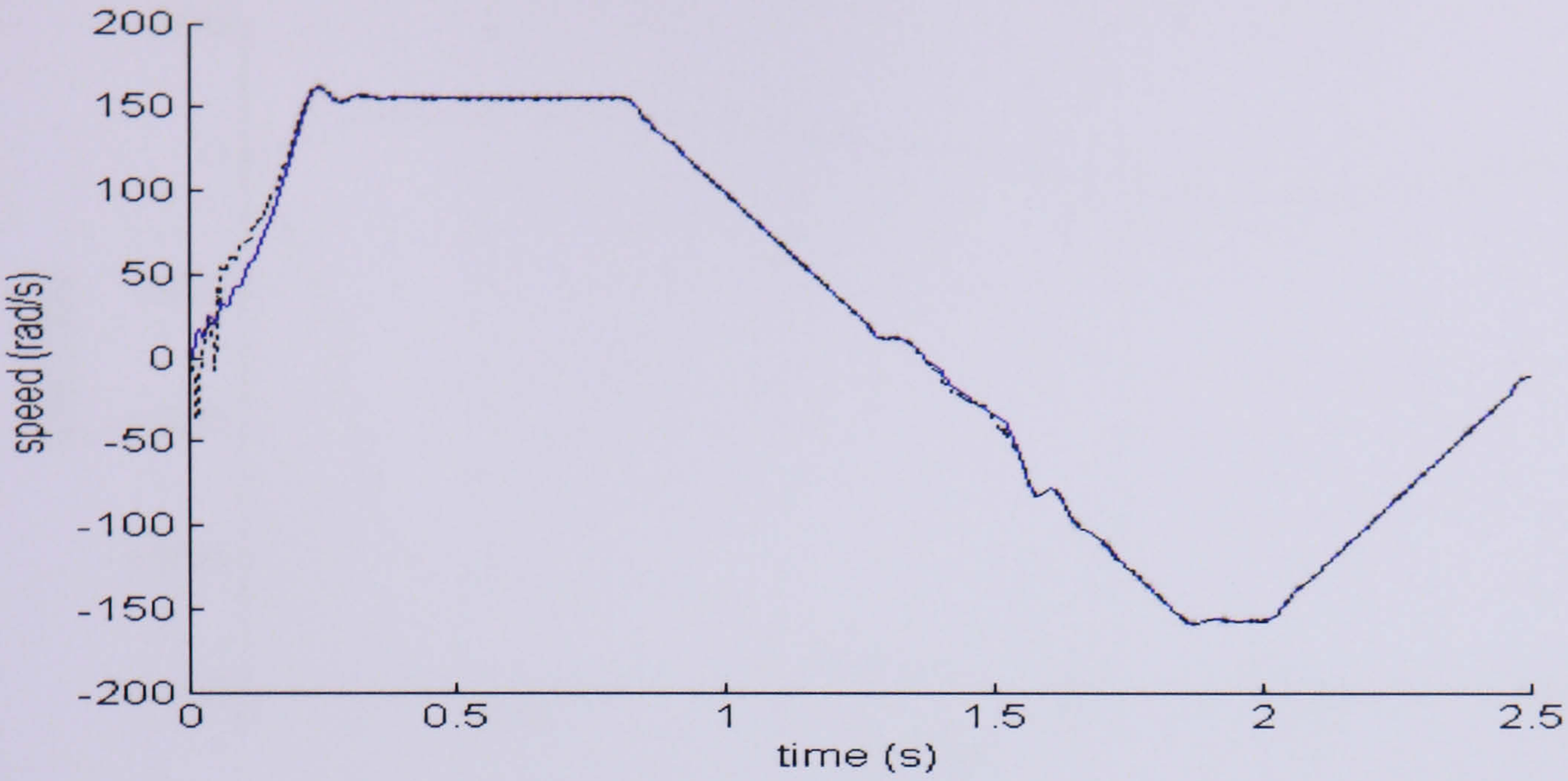


(b)

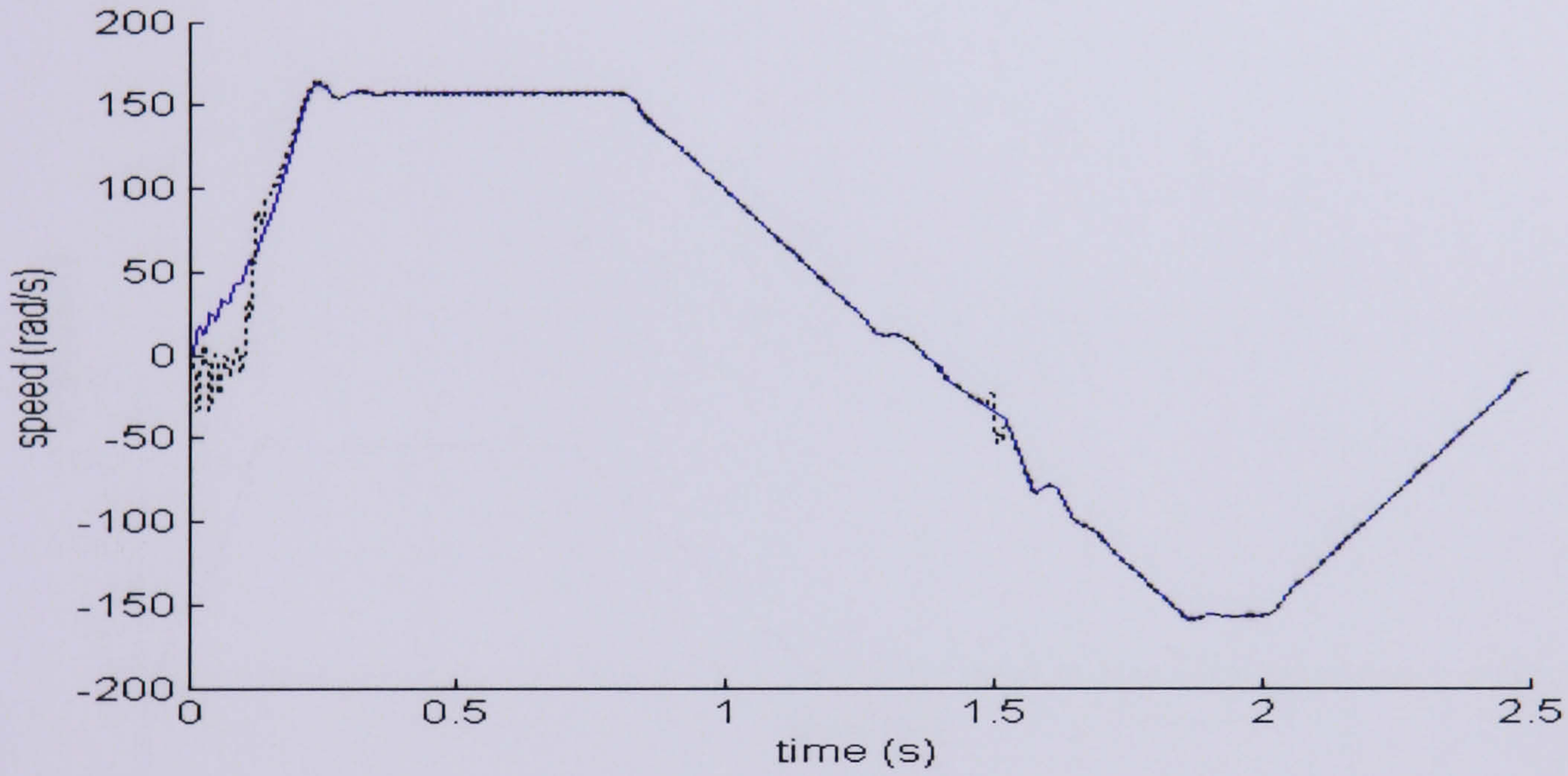


(c)

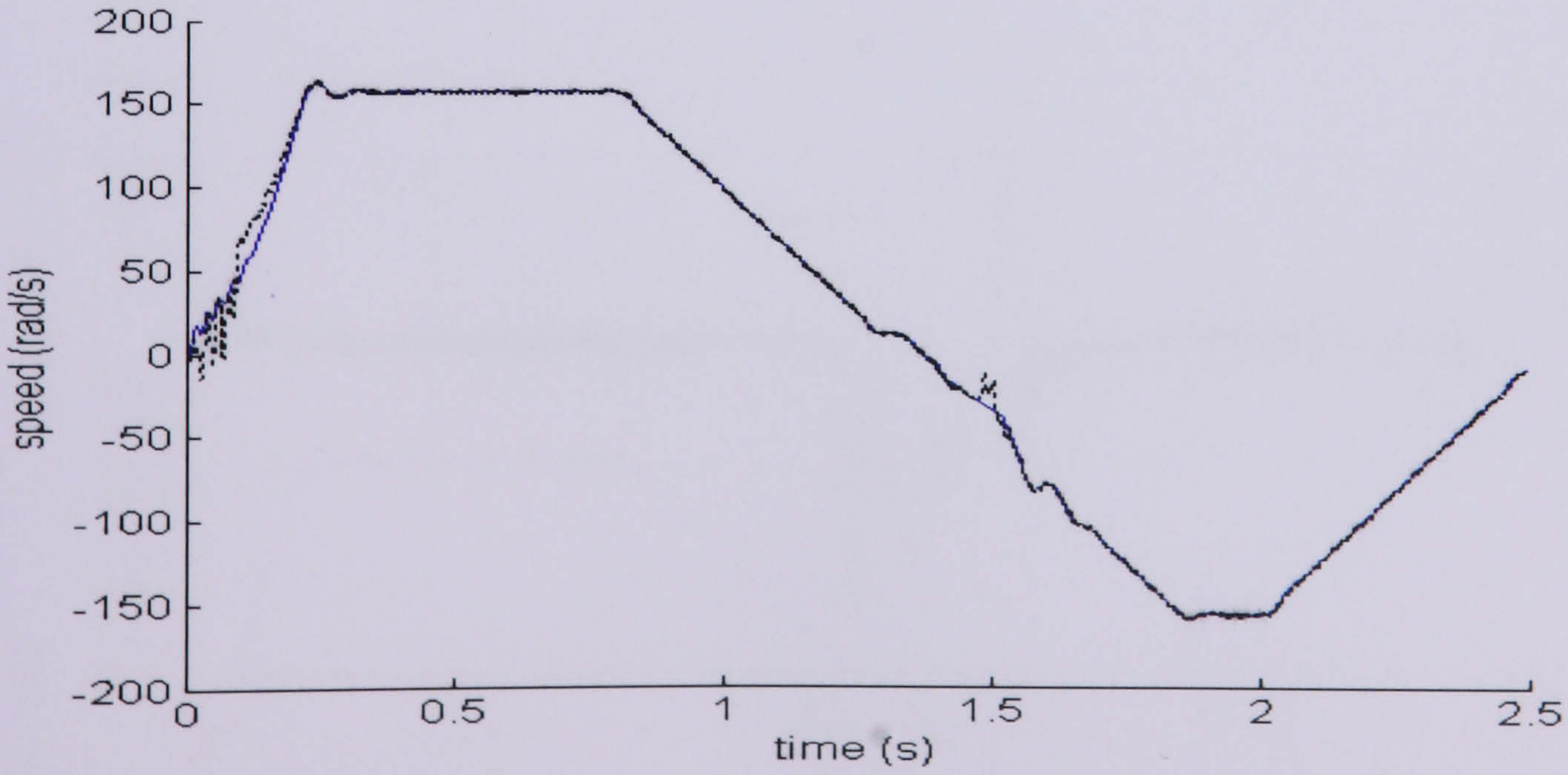
Figure 8.19: (a) Speed comparison of using (i) Trial and Error, (ii) SA and (iii) GA respectively when the rotor resistance of the motor model is increased by 10%



(a)

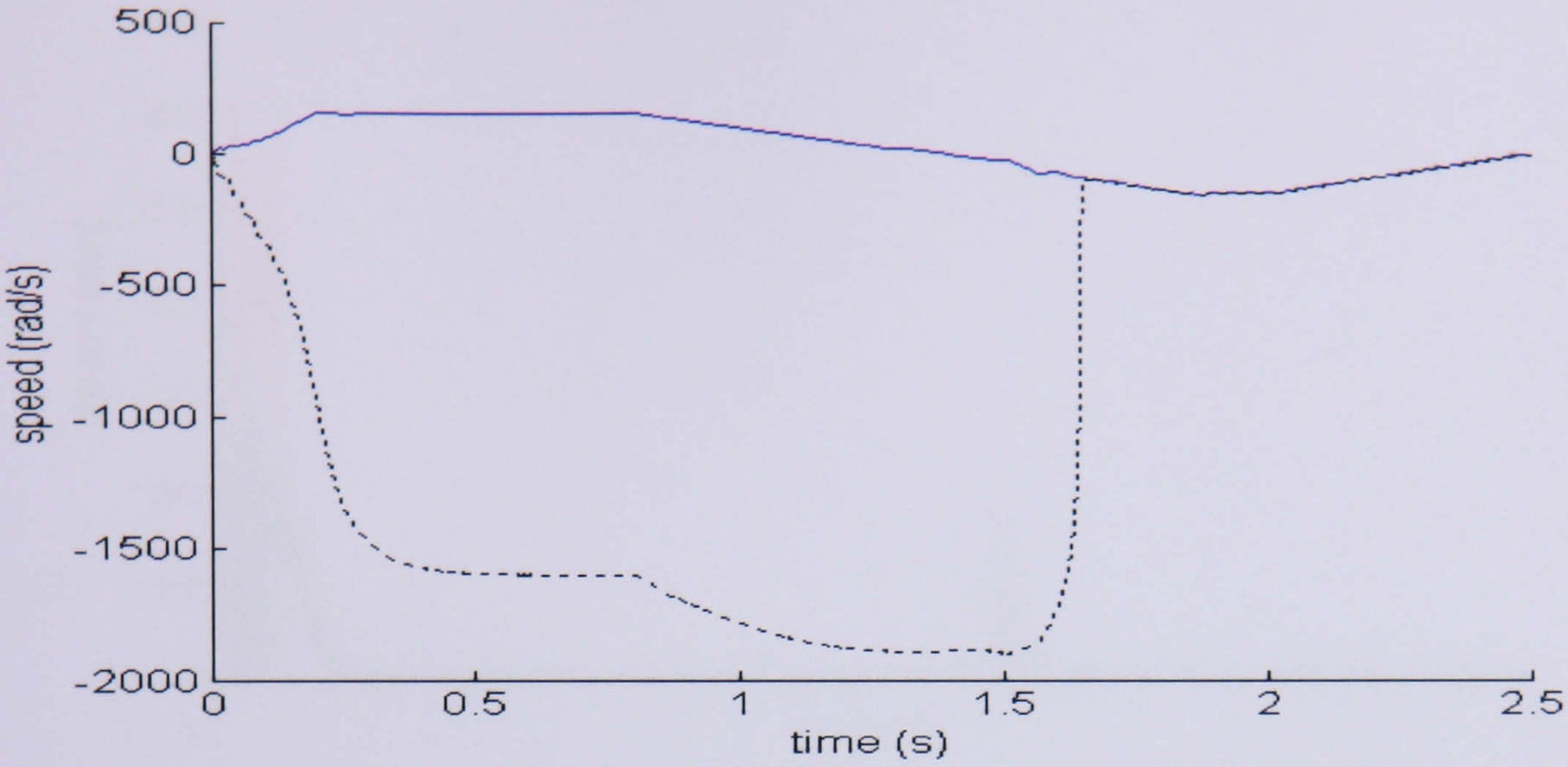


(b)

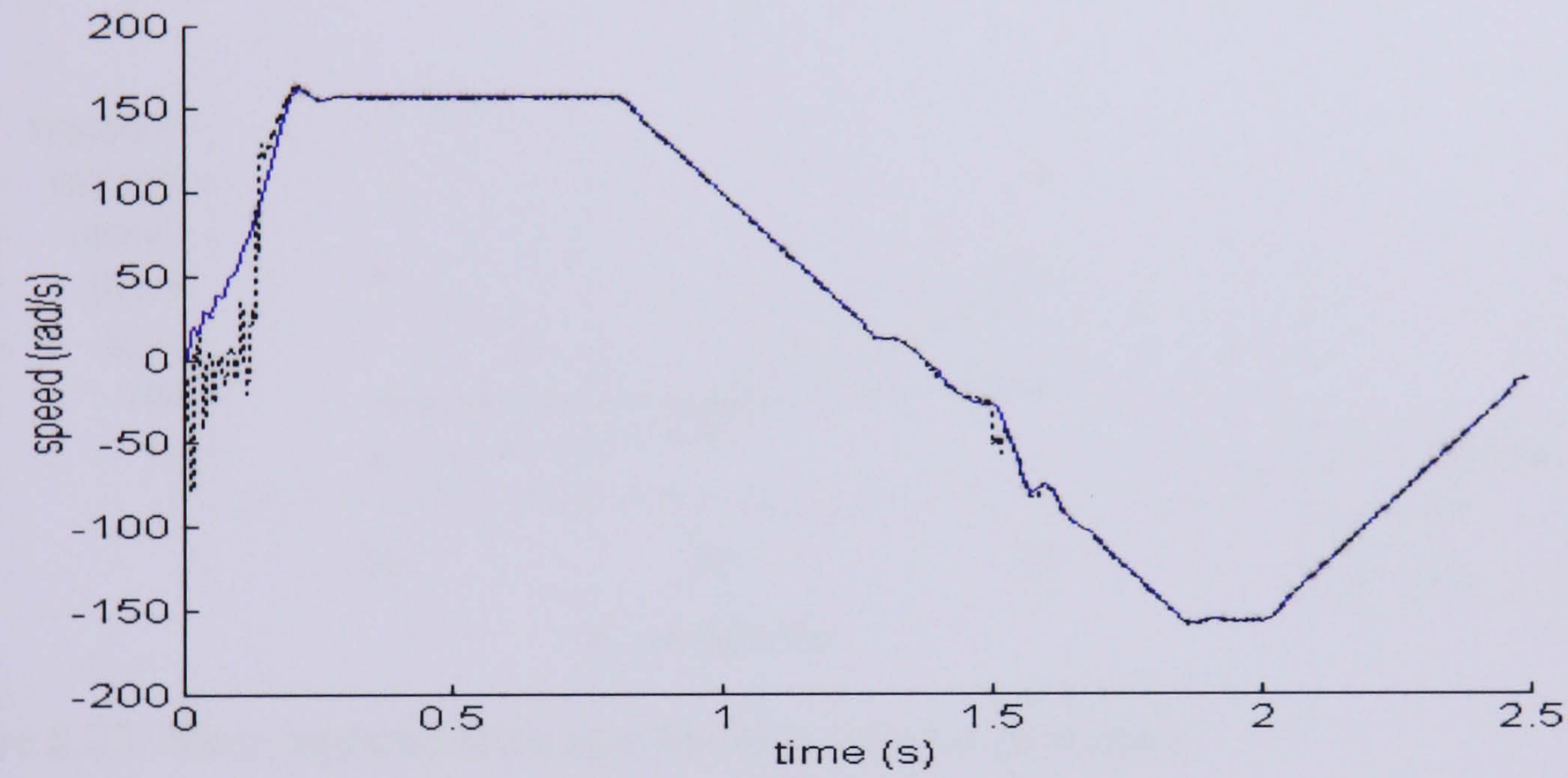


(c)

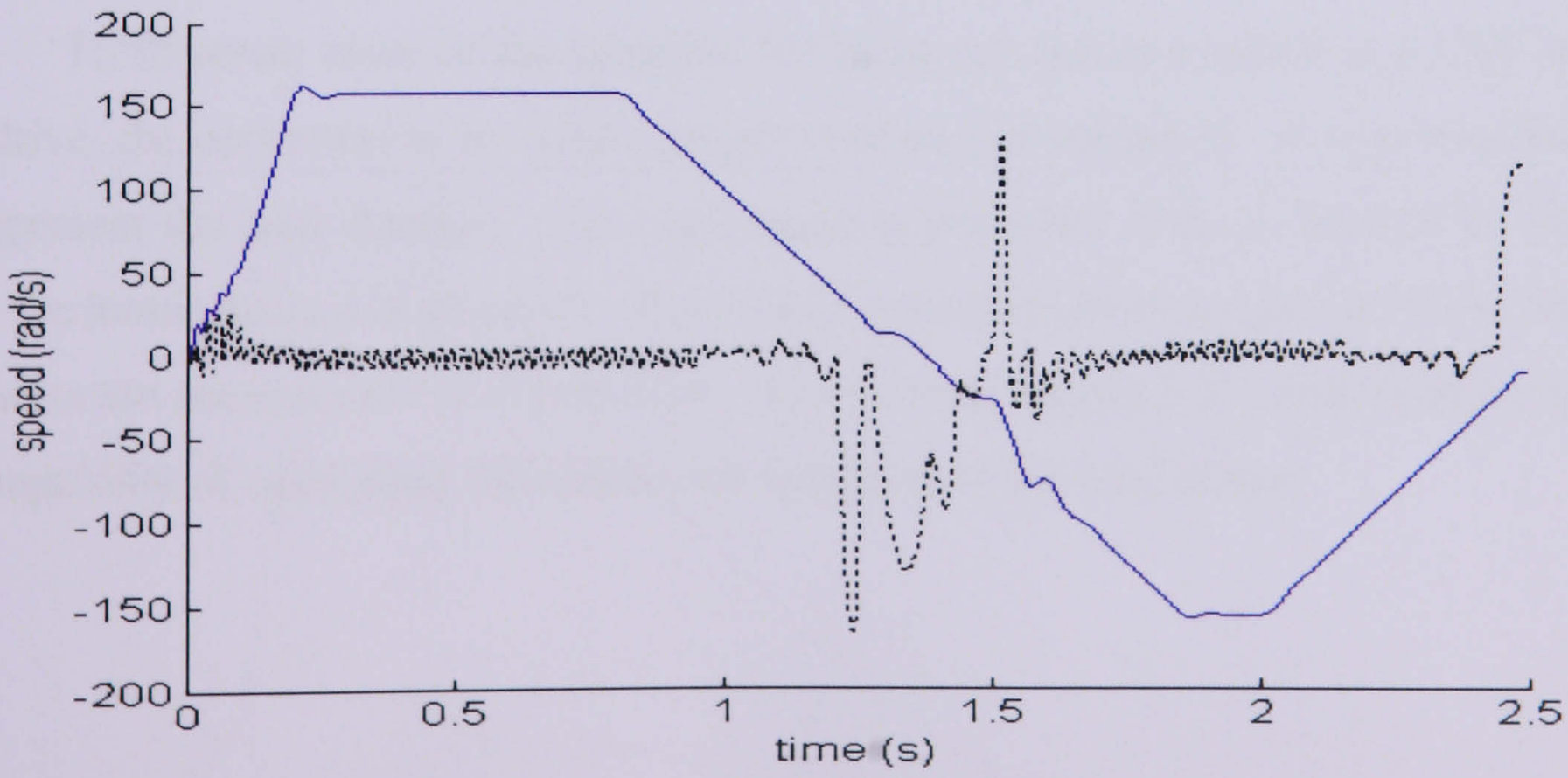
Figure 8.20 Speed comparison of using (a) Trial and Error, (b) SA and (c) GA respectively when the rotor resistance of the motor model is increased by 25%



(a)



(b)



(c)

Figure 8.21: Speed comparison of using (a) Trial and Error, (b) SA and (c) GA respectively when the rotor resistance of the motor model is increased by 50%

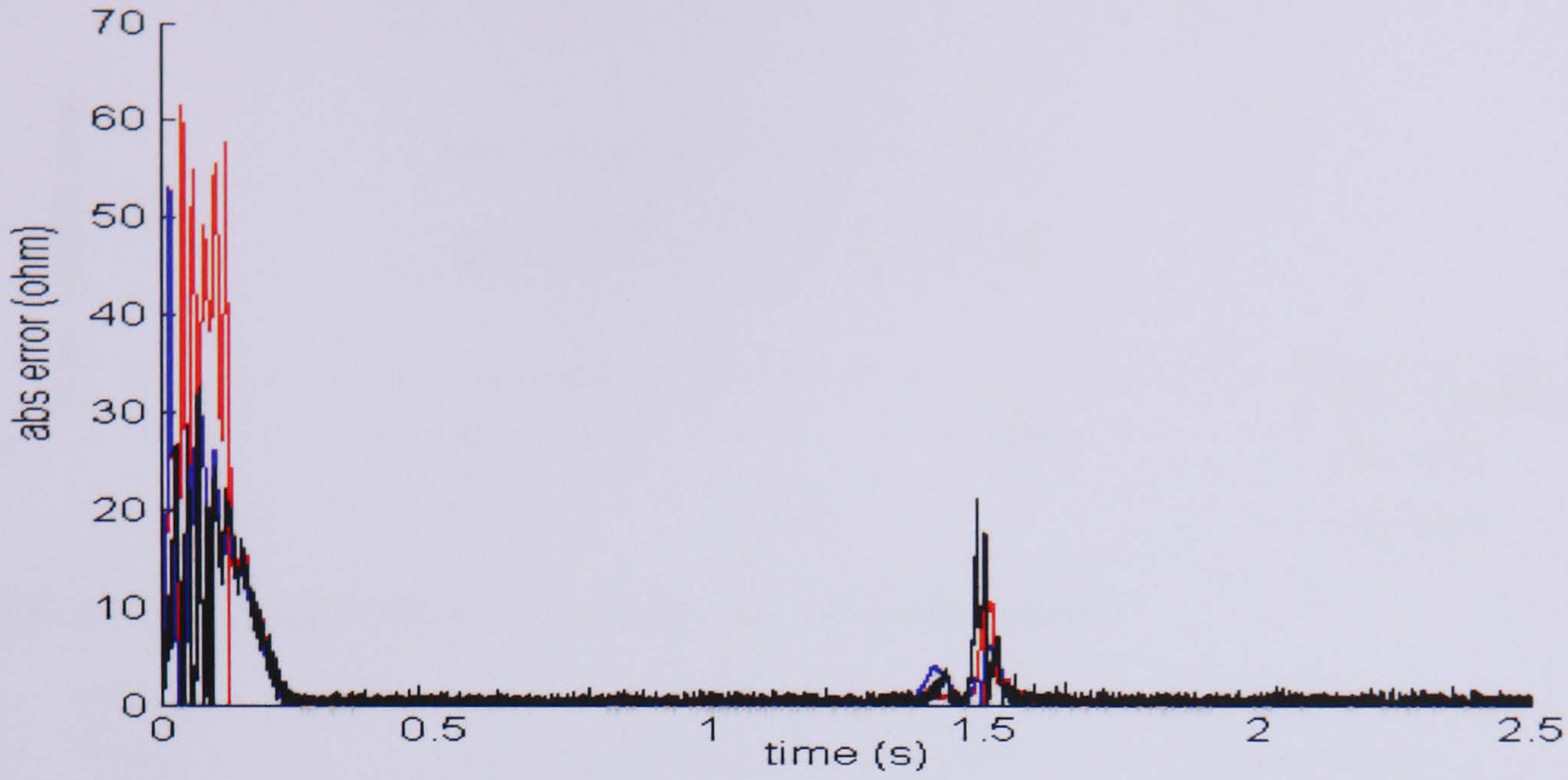


Figure 8.22: Absolute speed error when the rotor resistance increased to 25%

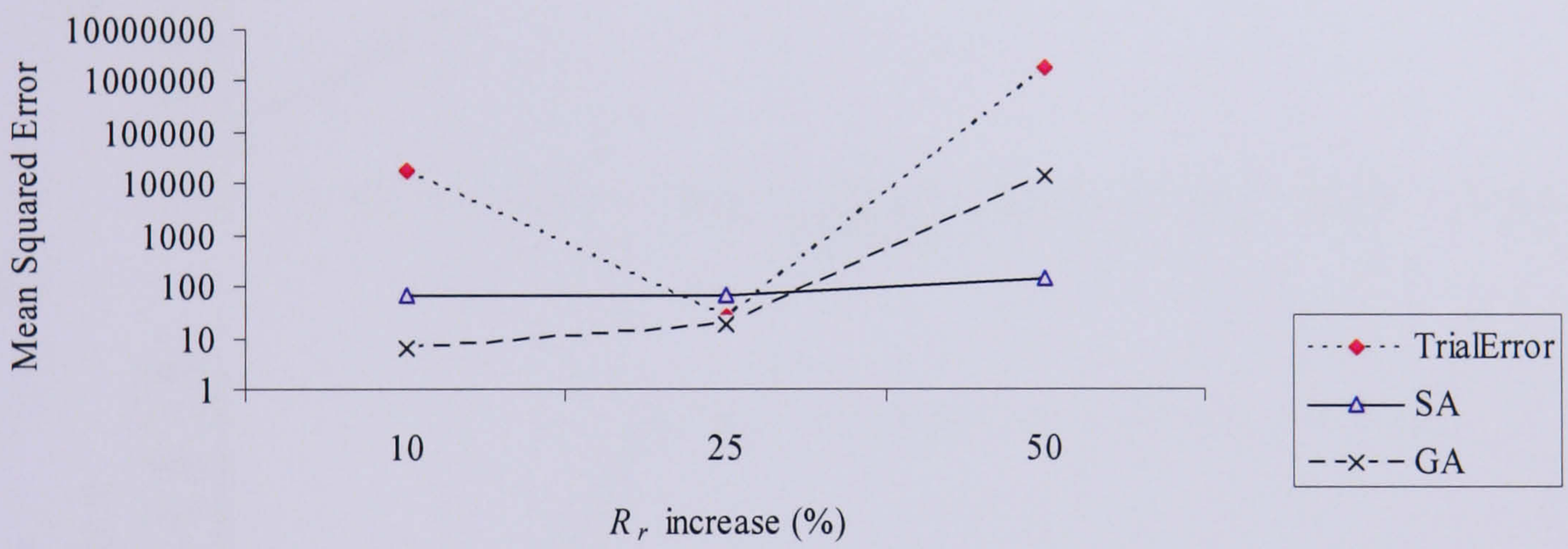


Figure 8.23: Mean squared error as a function of rotor resistance

8.5.4 Sensitivity of the Tuning Parameter to Load Changes

To illustrate more of the potential for using the tuning method in a EKF sensorless IM drive, the performance for load changes was also investigated. A step response is used to represent the load changes. The load torque is increased from no load to 20 Nm at 0.6s with the motor drive at high speed. The mean squared errors as a function of the load torque variation are summarised in Figure 8.24. As shown in Figure 8.25, each method has shown the capability of estimating the speed with variation of the load torque.

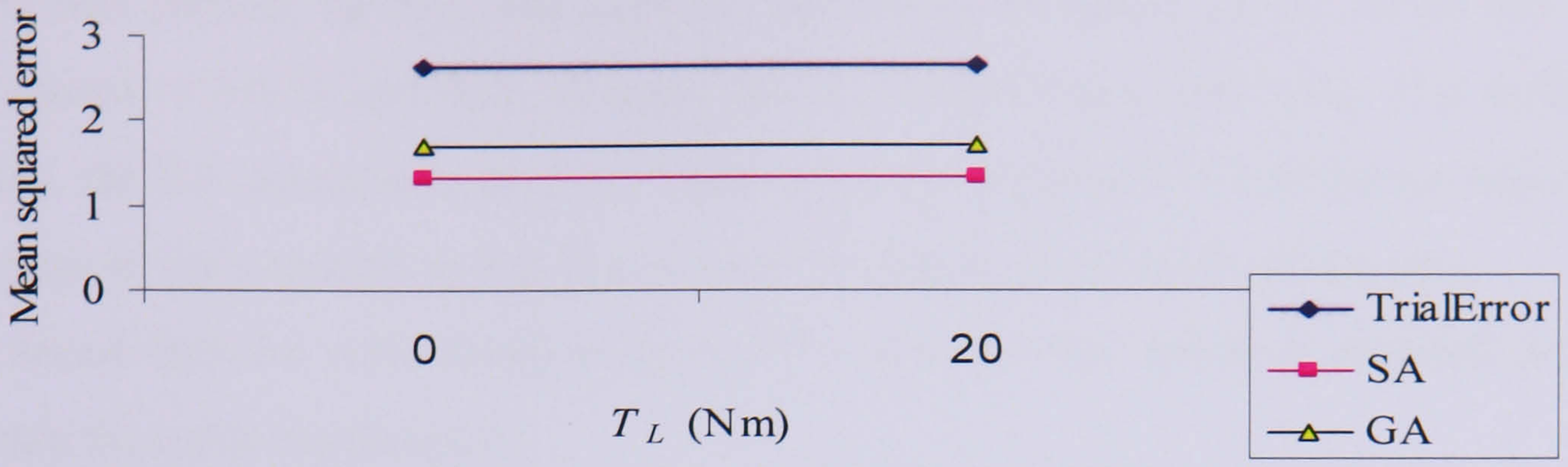
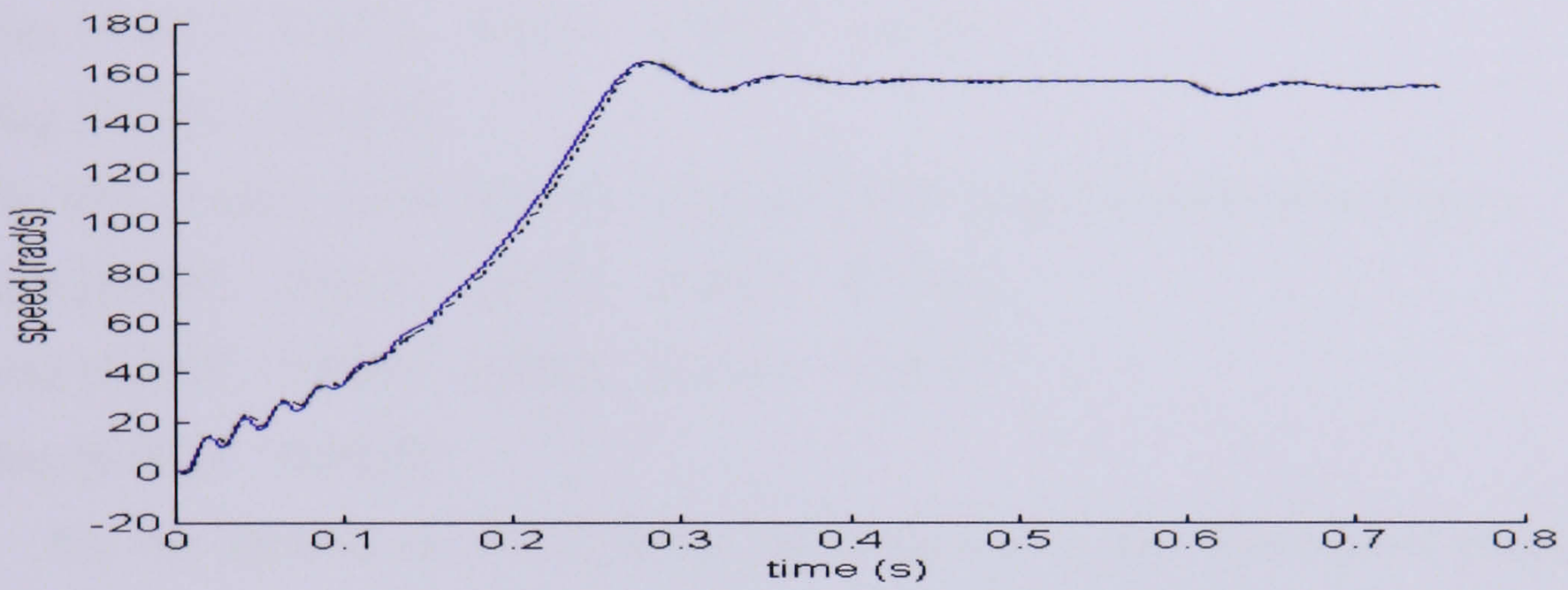
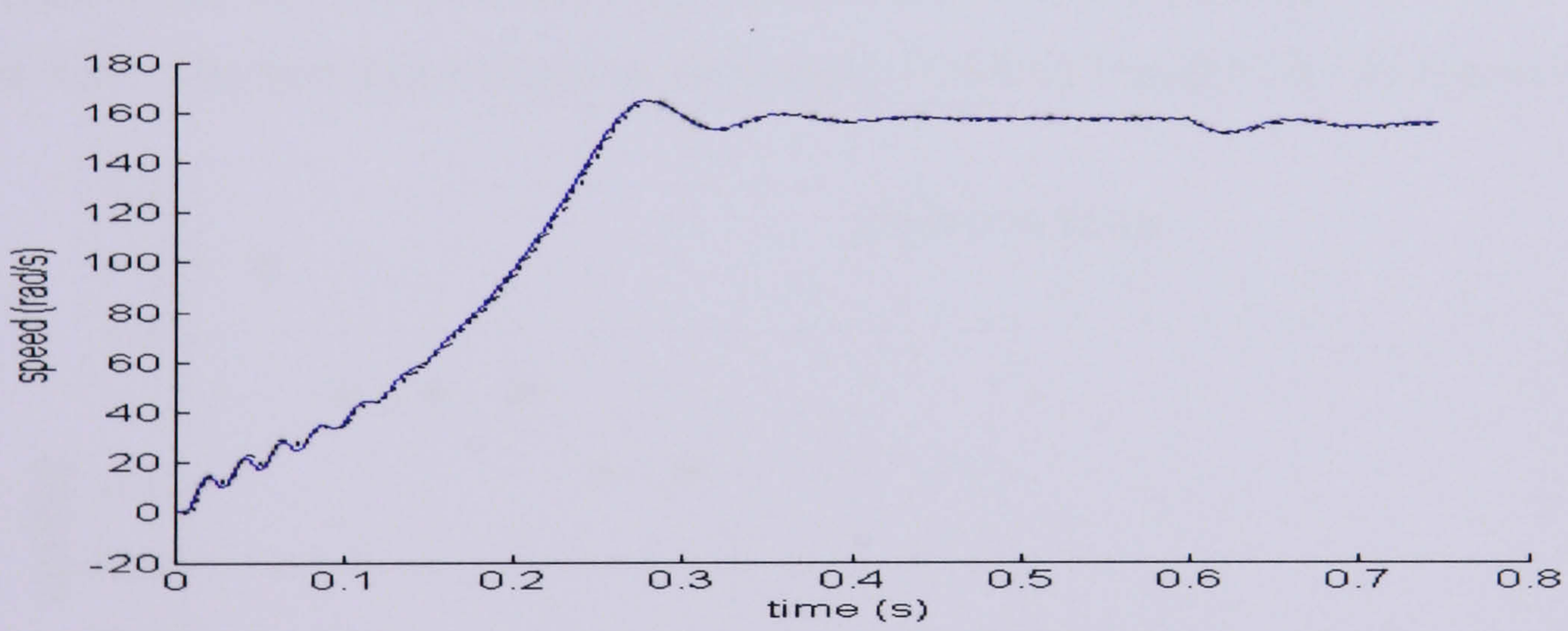


Figure 8.24: Mean squared error as a function of load torque



(a)



(b)

(c)

Figure 8.25: Performance of the EKF when the load torque increased at 20 Nm at 0.6s using (a) Trial and Error, (b) SA and (c) GA respectively

8.6 Application on Vector Control Drives

In this section, the GA was used for optimising the EKF for VC of the IM for the speed demand of 101% and 70% of rated speed, 155 rad/s and 108 rad/s. For both speed references, the GA parameters used are identical to the constant V/f application where there is no change in the population size, the generation number and the searching area.

For the speed demand with steady state of 155 rad/s, the best solution obtained is 3.0939 which uses the covariance matrix:

Q =diag [0.0024 0.0020 0.0019 0.0017 0.5200]

G =diag [0.0077 0.0055 0.0011 0.0024 0.0100]

R= diag [0.0092 0.0056]

For the 70% of rated speed, the best solution 0.2629 using the covariance matrix:

Q =diag [0.0055 0.0072 0.0039 0.0042 0.7700]

G =diag [0.0007 0.0005 0.0001 0.0001 0.0094]

R= diag [0.0032 0.0023]

For the application of VC drive, the evolution of the optimisation process which converges to the best solution for both speed references is shown clearly in Figure 8.26 and Figure 8.27. The best speed response obtained is shown in Figure 8.28 and Figure 8.29.

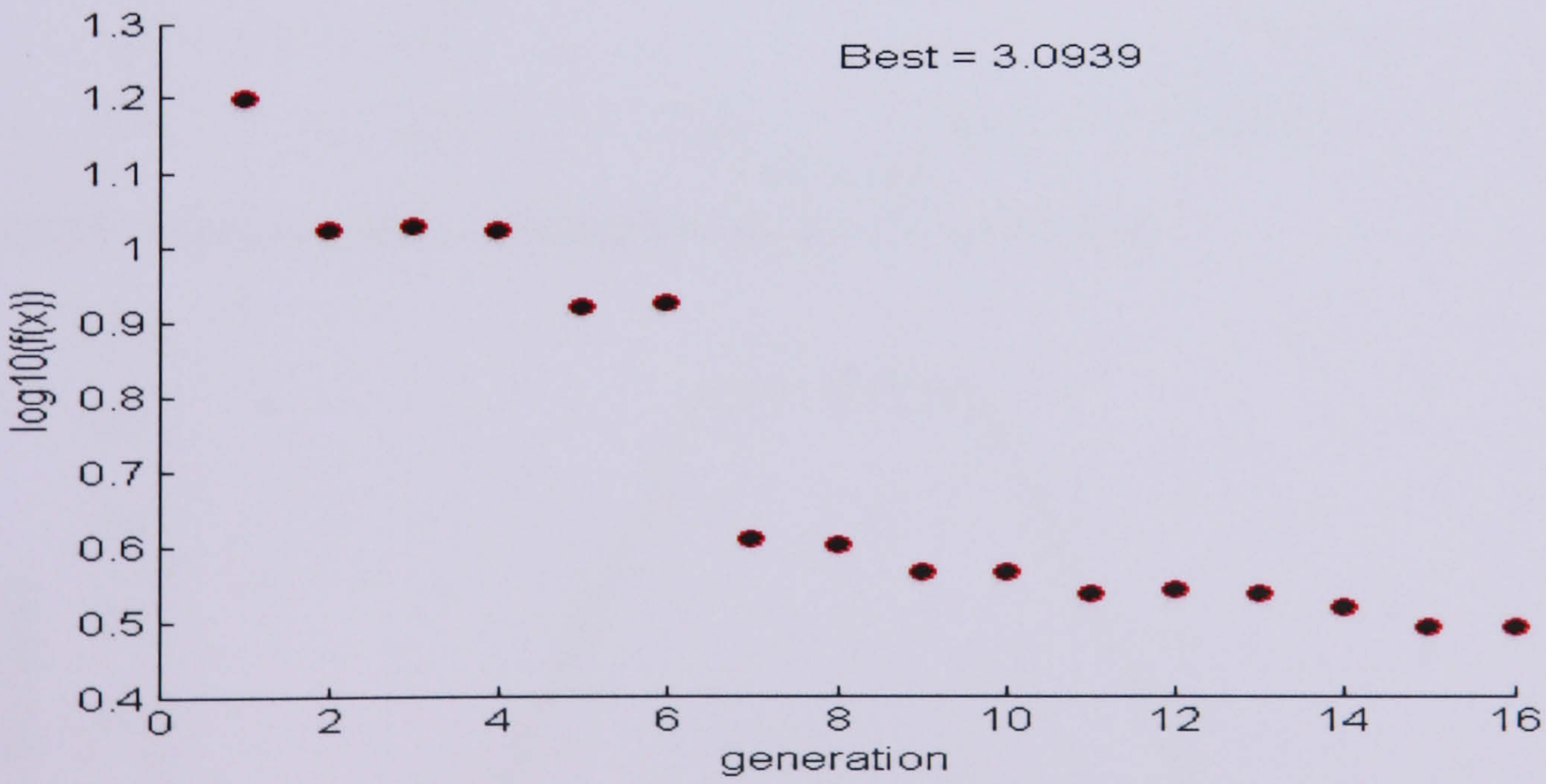


Figure 8.26: Evolution of the optimisation process for 155 rad/s

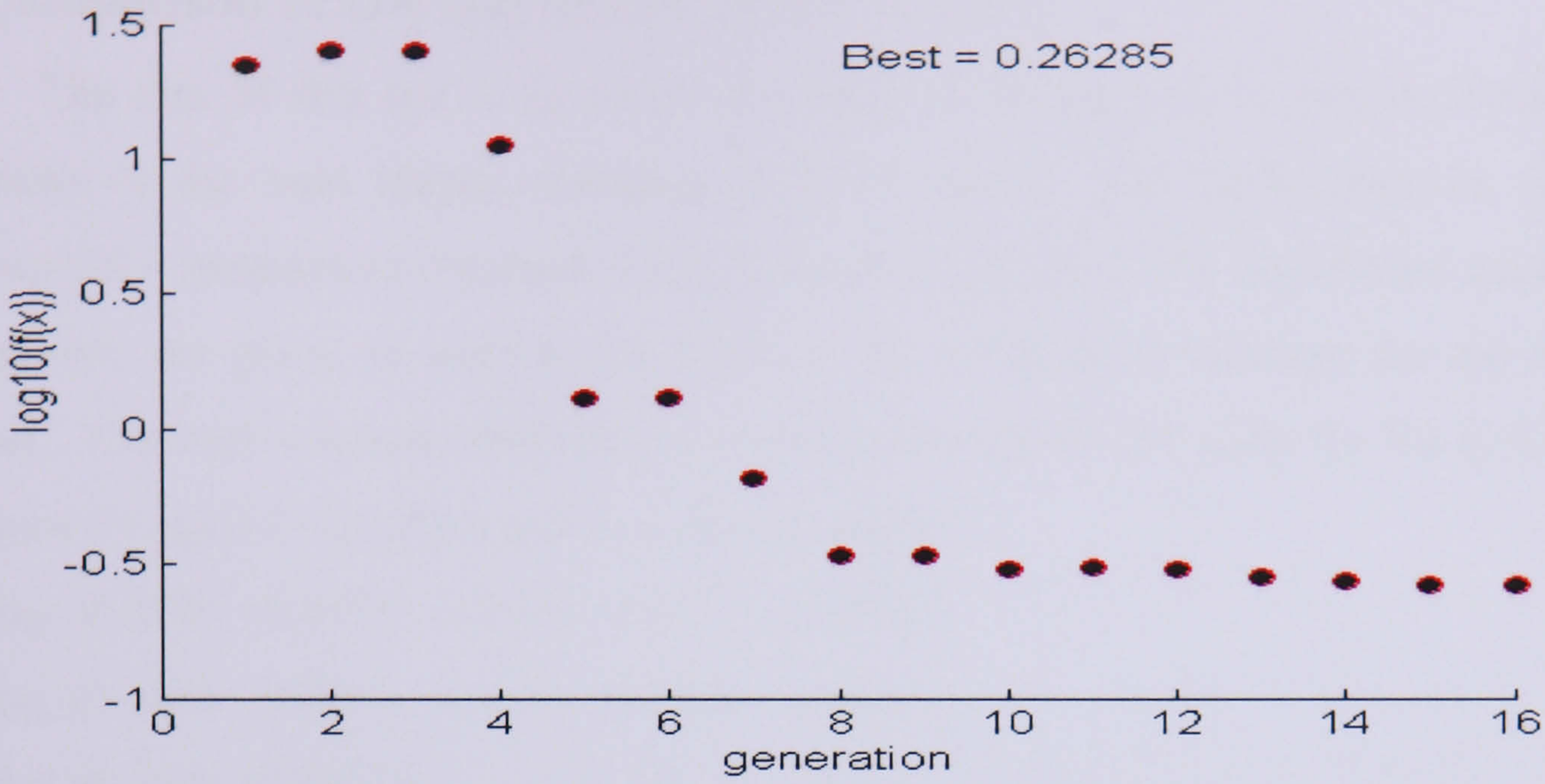


Figure 8.27: Evolution of the optimisation process for 108 rad/s

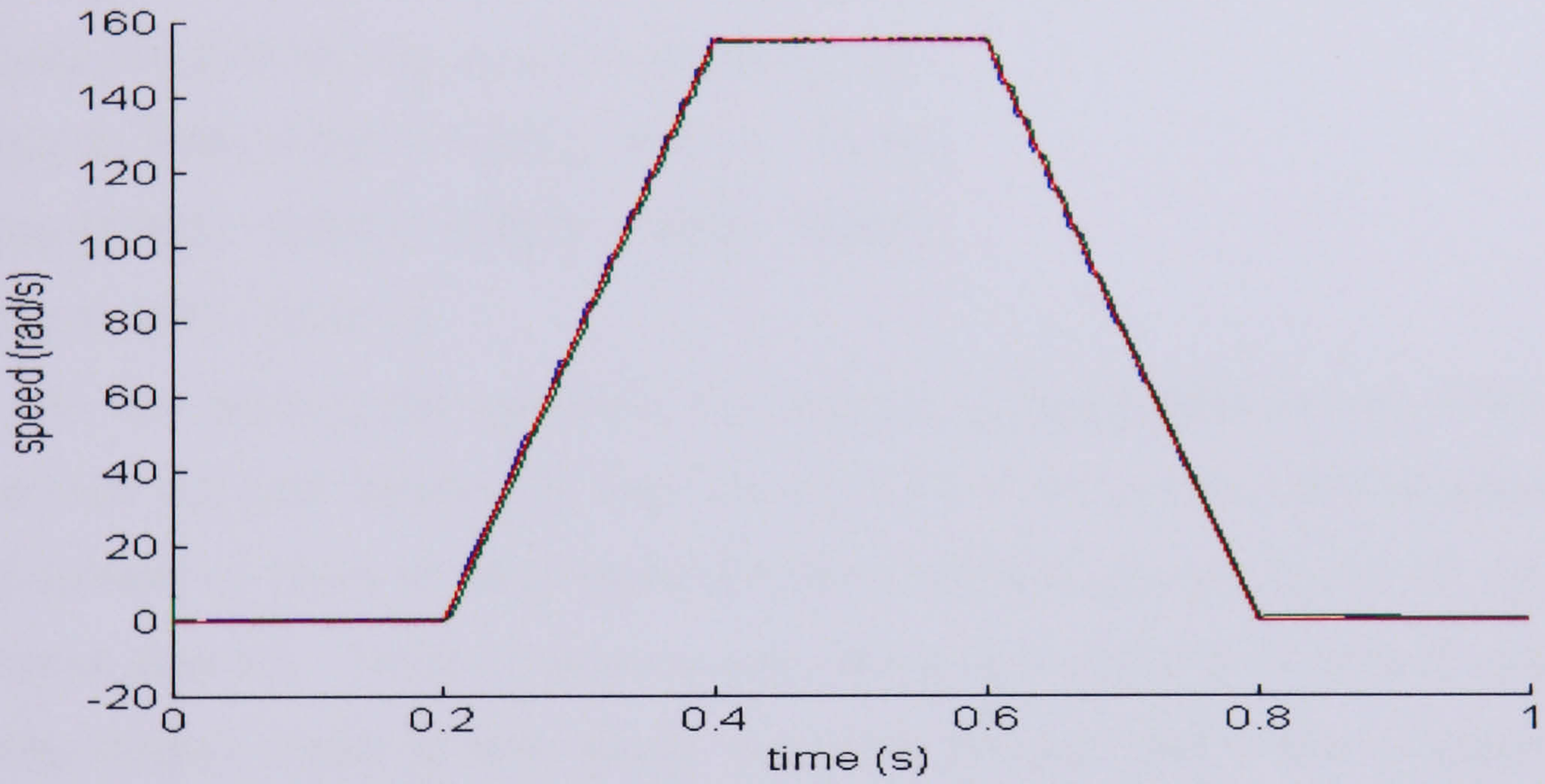


Figure 8.28: Speed response obtained by GA for 155 rad/s ramp

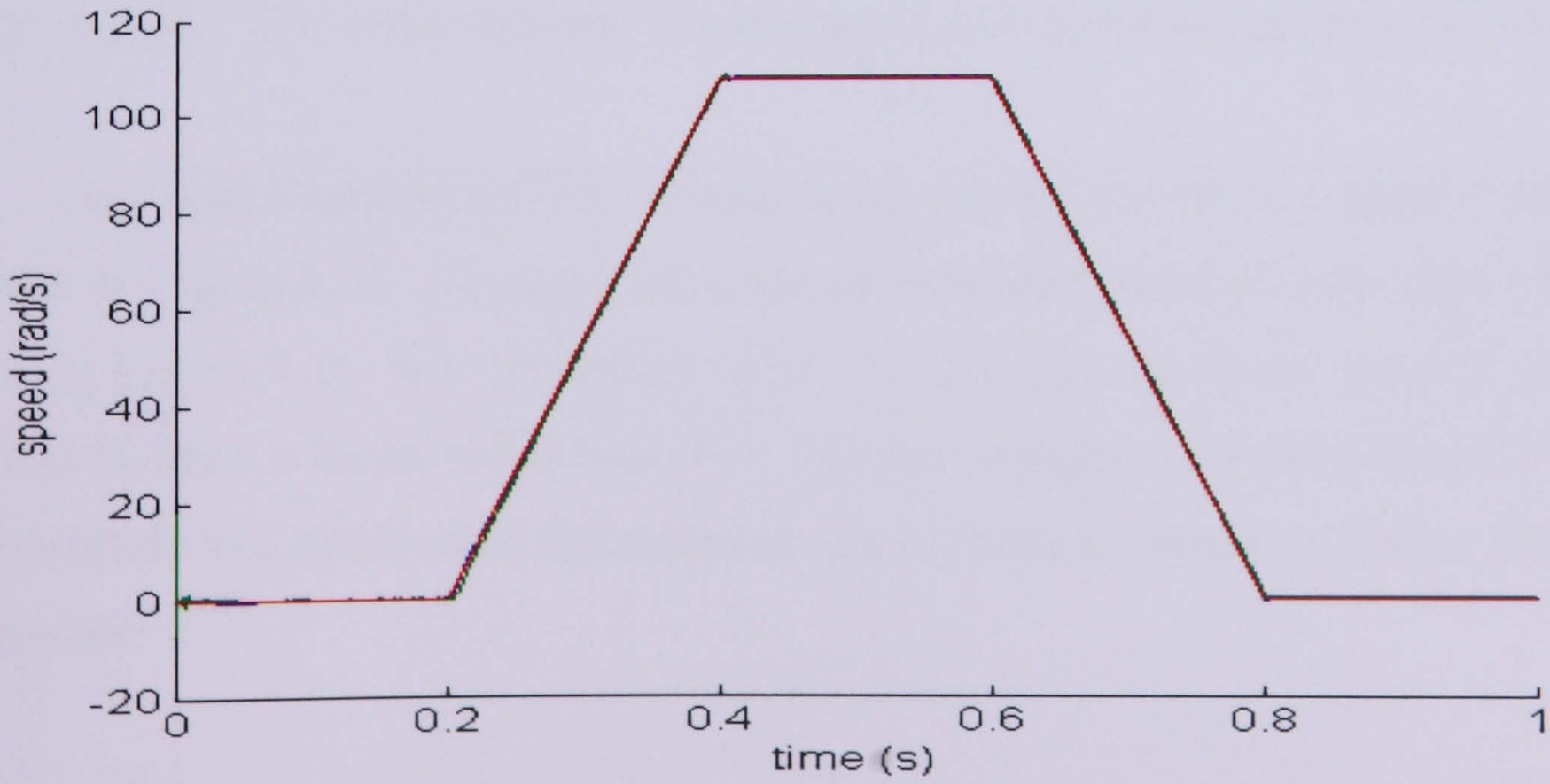


Figure 8.29: Speed response obtained by GA for108 rad/s ramp

8.7 Comparison of GA and SA on Vector Control

The aim of this test is to access the EKF in the case of the speed demand, motor parameter or the load torque changing in a VC drive. The comparison is carried out between EKF parameters obtained using SA and using GA. The covariance matrices used for the GA are given in section 8.6 whereas the covariance matrices for the SA are as follows. The best solution obtained for a speed demand of 155 rad/s for SA is 1.88 during iterations 14 and 15 with the noise covariance matrix:

Q =diag [0.0065 0.0075 0.0009 0.0010 1.0000]

G =diag [0.0078 0.0015 0.0011 0.0022 0.0071]

R= diag [0.0034 0.0023]

The best solution obtained for a speed demand of 108 rad/s for SA is 0.9607 during iterations 246-279 with the noise covariance matrix:

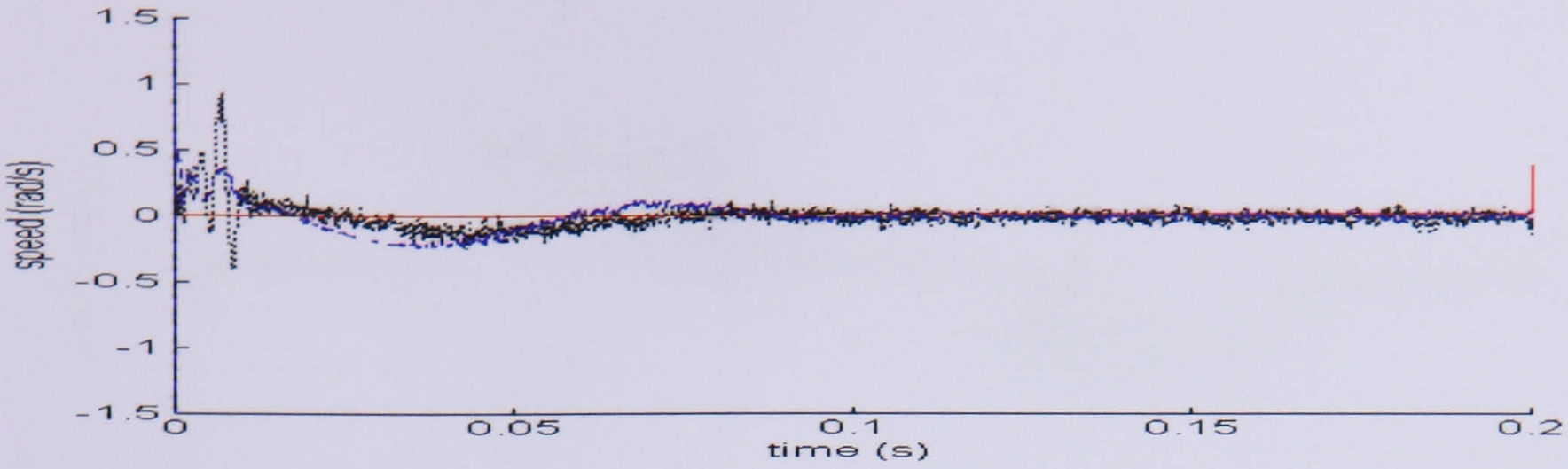
Q =diag [0.0030 0.0057 0.0022 0.0014 0.9786]

G =diag [0.0021 0.0014 0.0005 0.0026 0.0076]

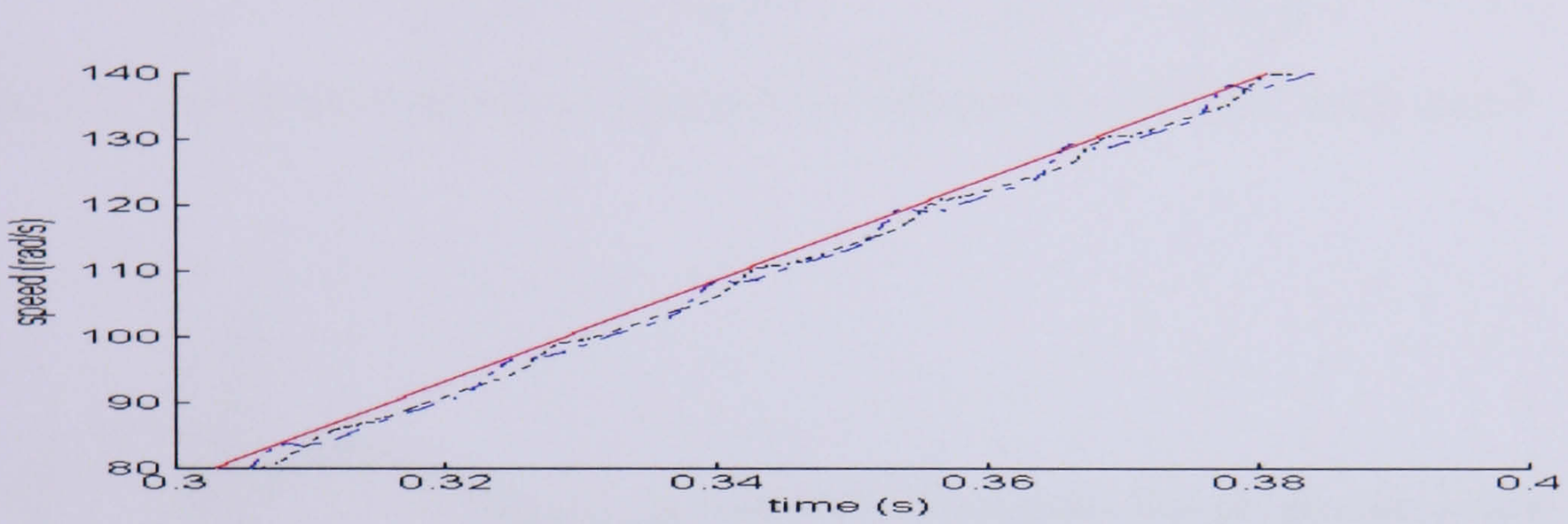
R= diag [0.0044 0.0051]

In this section, the reference, SA and GA is represented by red, black and blue respectively and each figure is divided into several traces for each different situation. For a speed demand of 101% of rated speed, the mean squared errors obtained by GA are about 64% more than SA. These extra errors are contributed during the transient condition. SA performs slightly better in both steady state and transient apart from a higher overshoot which occurs between 0.01s and 0.4s. The speed responses tuned by GA and SA are shown in Figure 8.30. The error between the estimated and reference speed is shown in Figure 8.31.

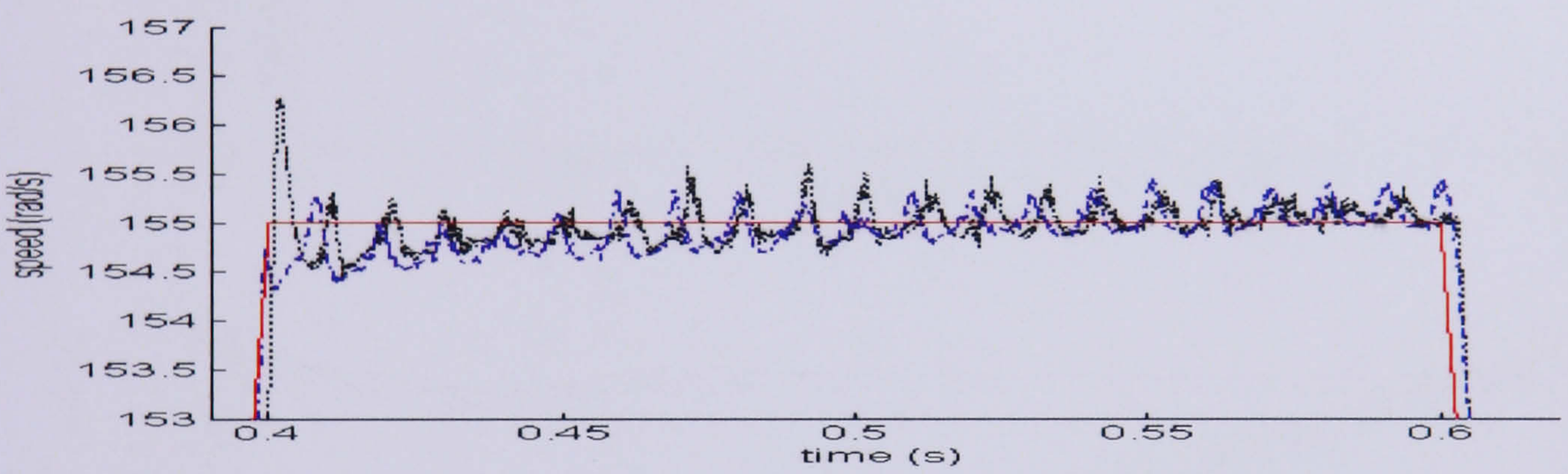
For a speed demand of 70% of rated speed, the speed responses tuned by GA and SA is shown in Figure 8.32. The error between the estimated speed and the reference speed is shown in Figure 8.33. GA performed better in obtaining the mean squared error and is expected to have a better result than SA. GA has a higher overshoot during 0.01s but a good result during steady state and transient. SA exhibits an overshoot before settling to the steady state.



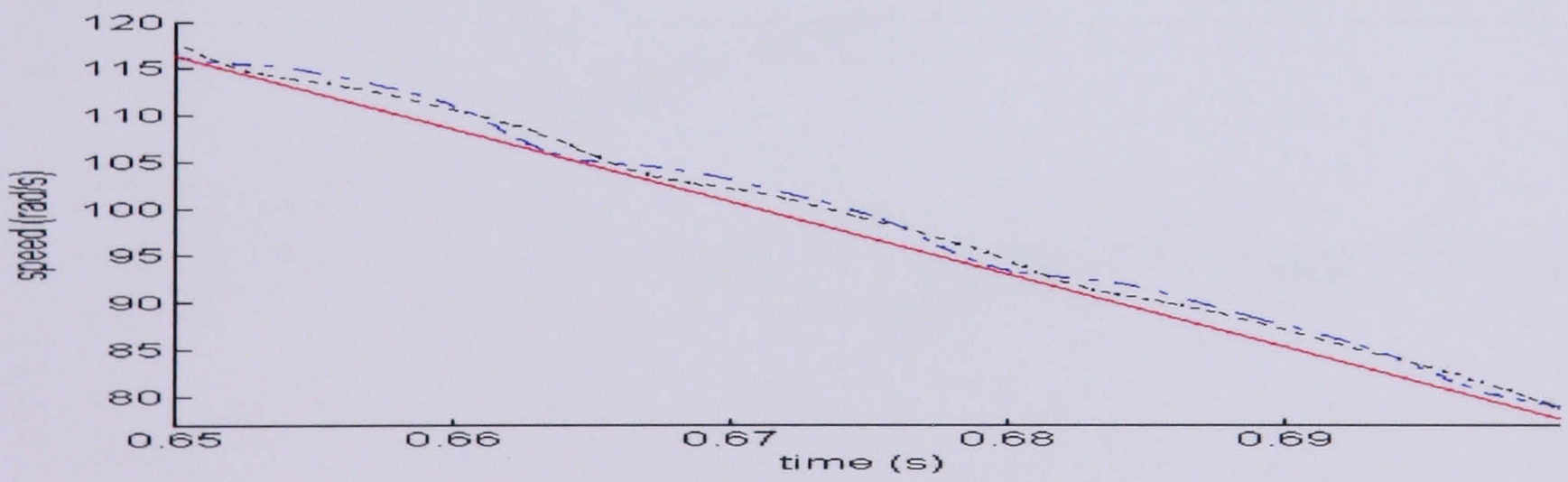
(a)



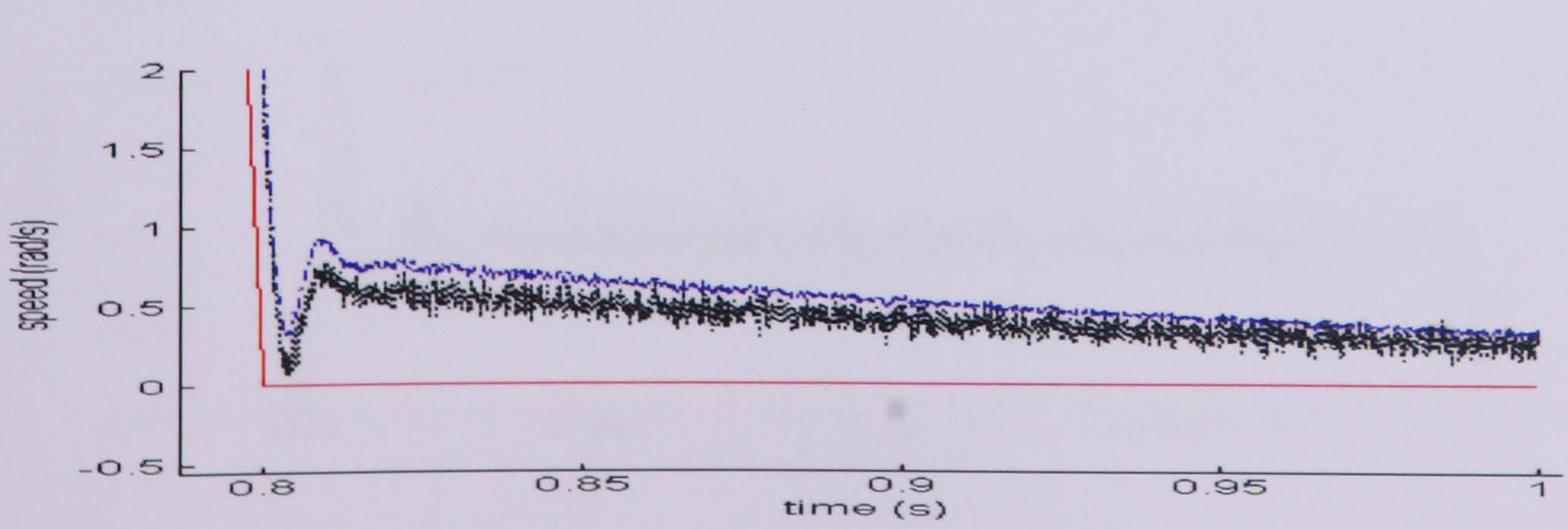
(b)



(c)



(d)



(e)

Figure 8.30: (a-e) Speed responses using 101% of rated speed

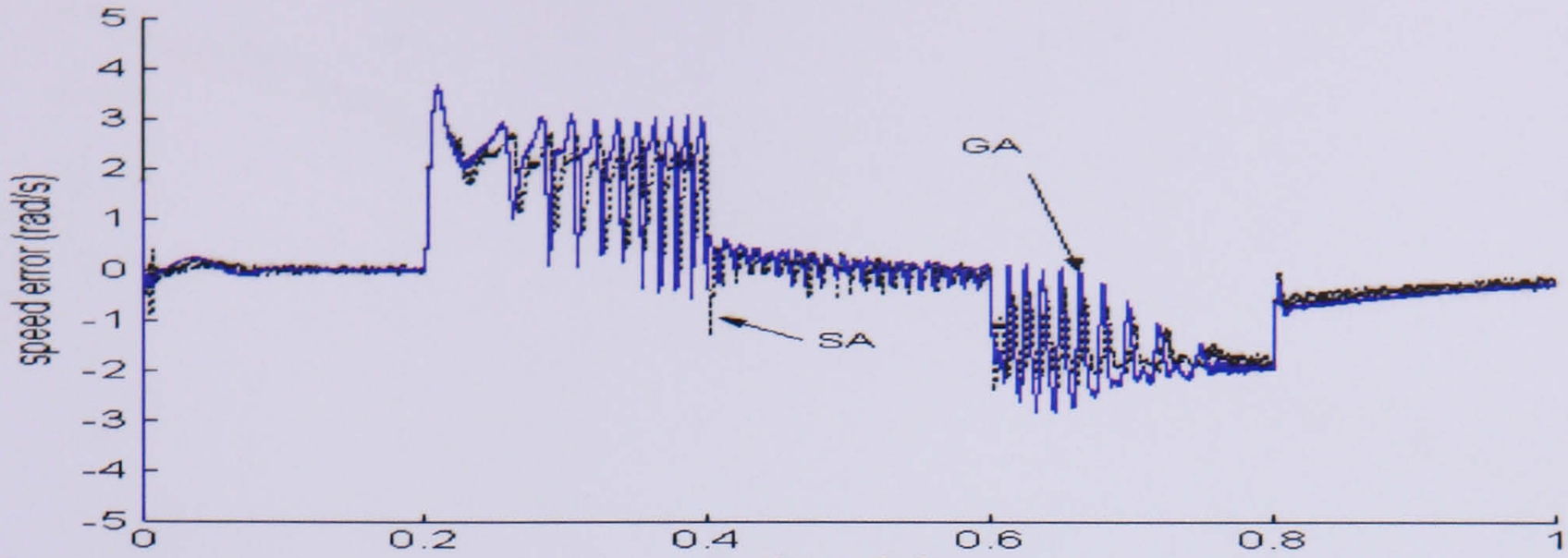
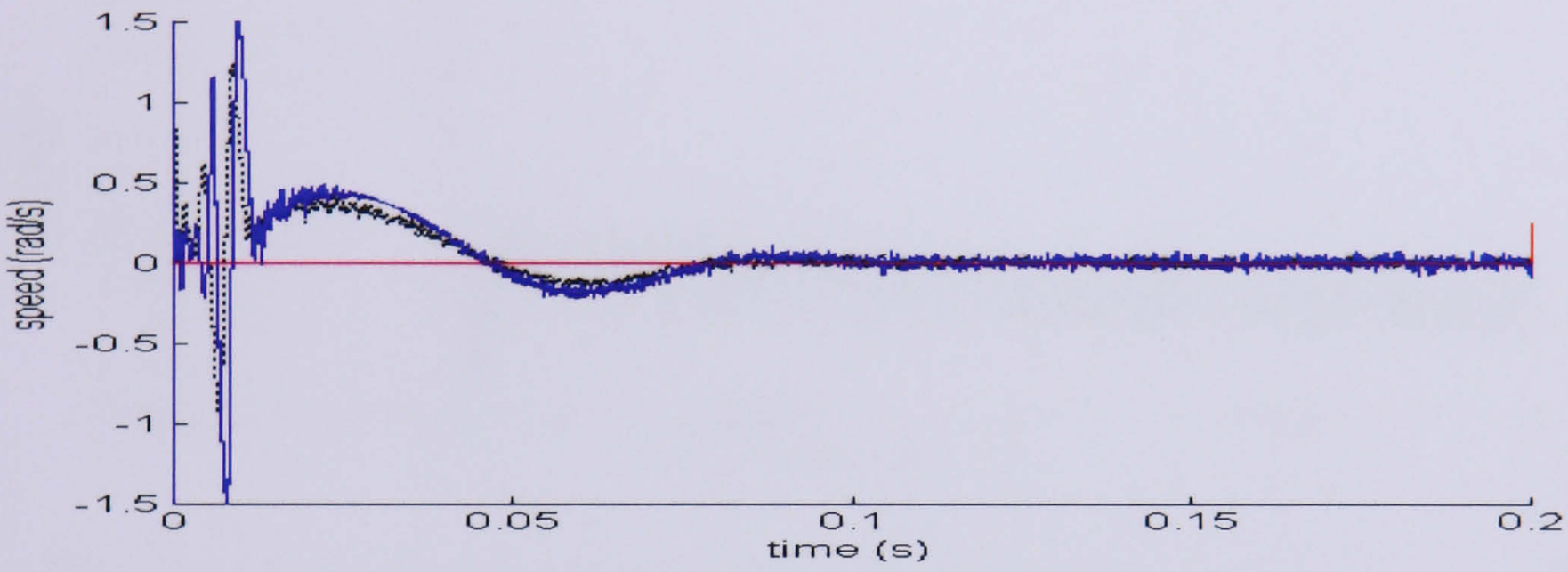
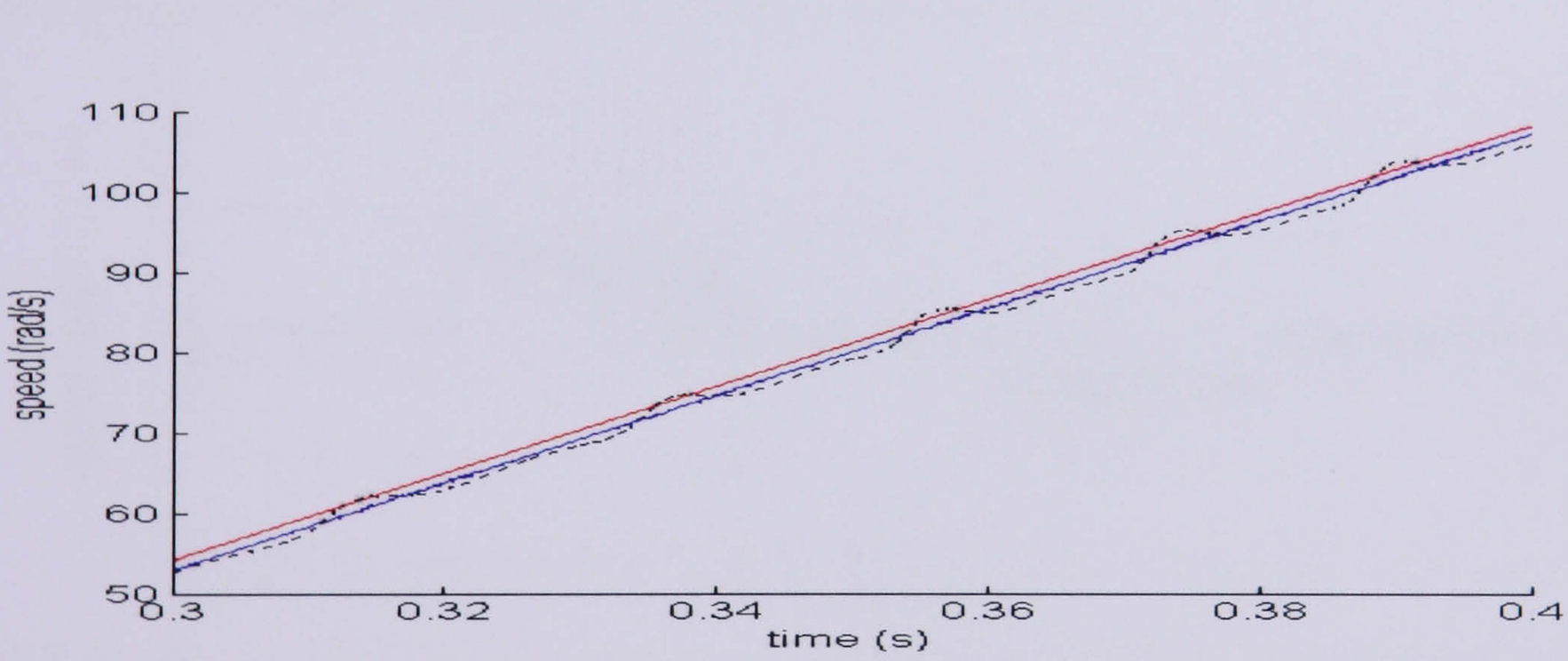


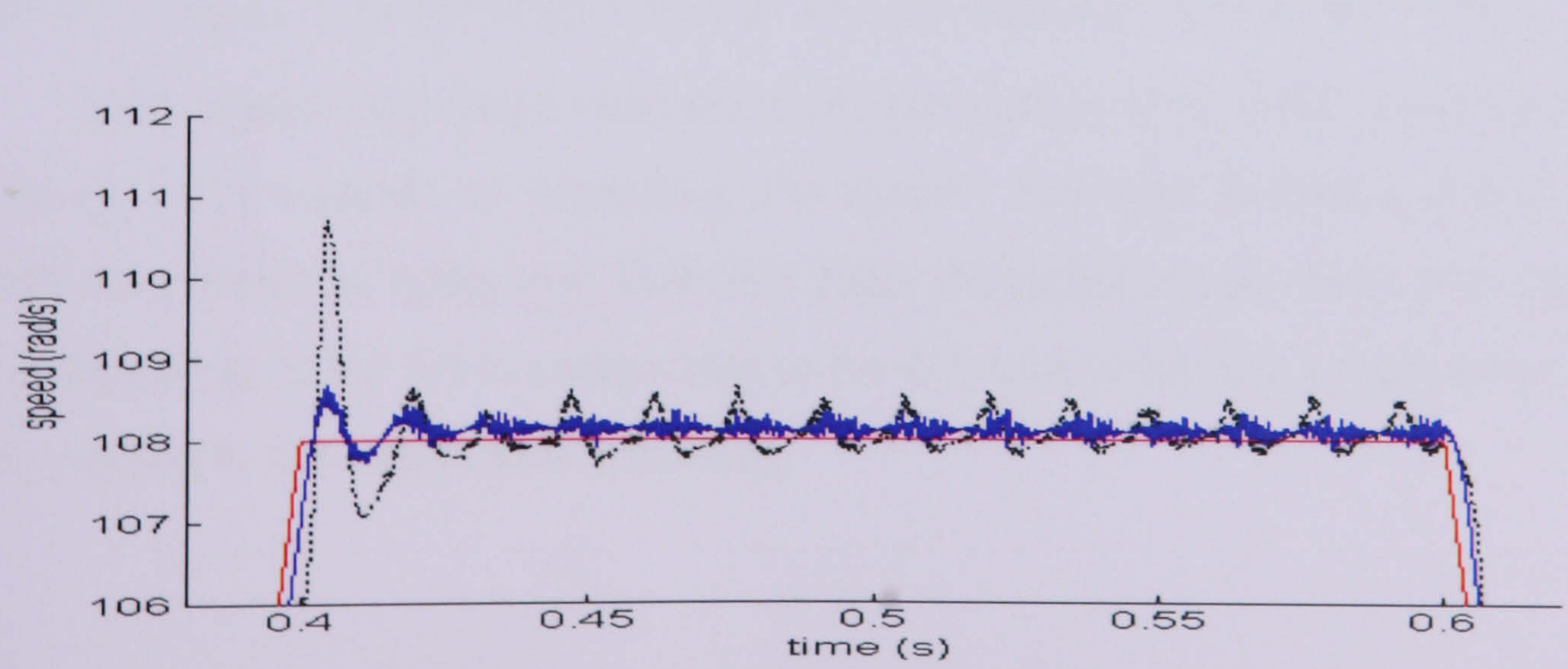
Figure 8.31: Speed error between reference and estimate for 101% of rated speed



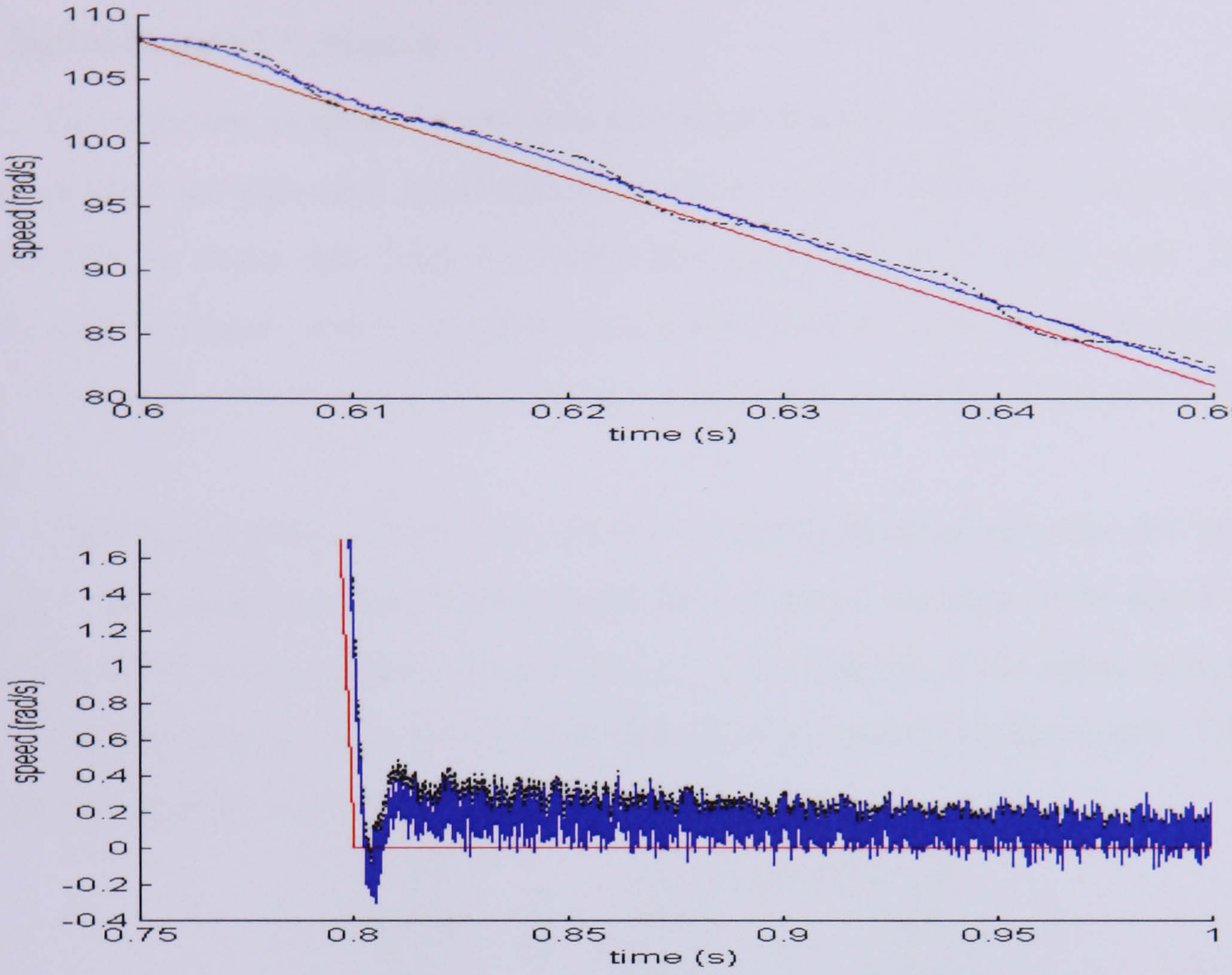
(a)



(b)



(c)



(d)

(e)

Figure 8.32: (a-e) Speed responses using 101% of rated speed

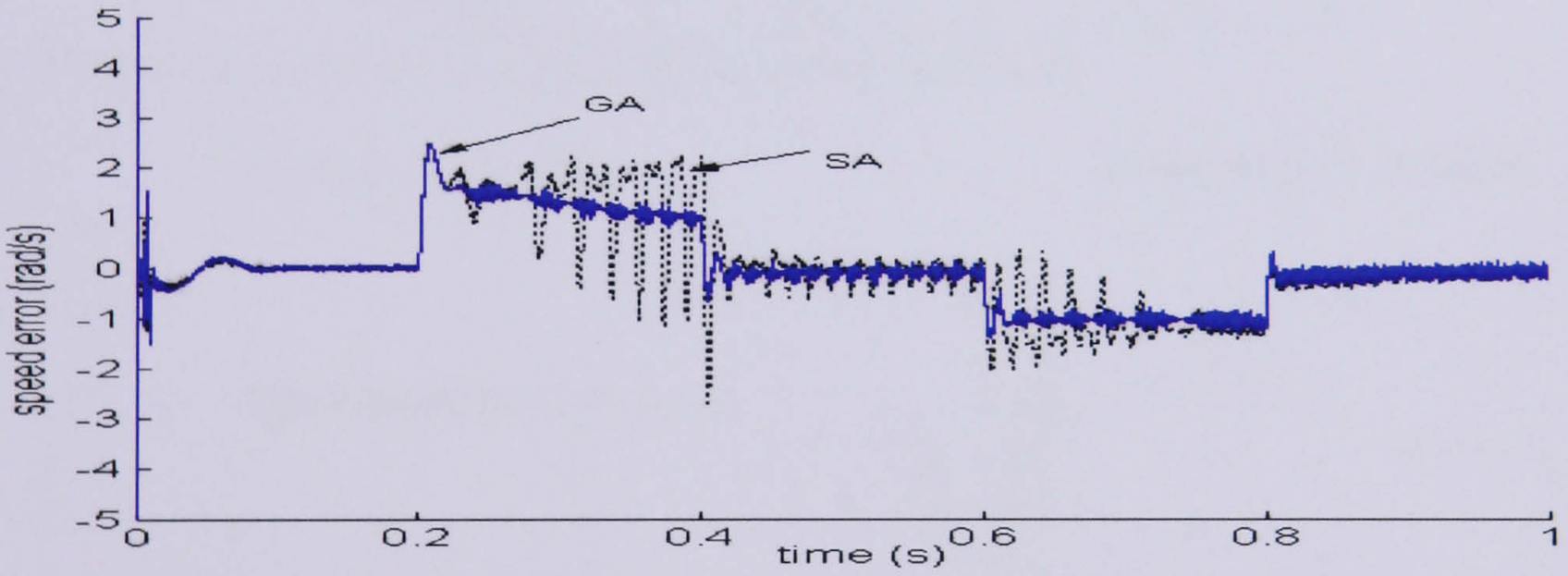


Figure 8.33: Speed error between reference and estimate for101% of rated speed.

In the closed loop speed estimation, the parameters of the EKF tuned by SA and GA are shown to be capable of estimating the speed. For both methods, despite the mean squared error obtained being low, there are some difficulties in the early start-up condition. The performance of the SA is comparable to the GA except that the SA produces overshoots before settling to the steady state condition.

8.7.1 Speed Demand Variation

The response to speed variation is investigated using a speed demand that takes the speed transient through zero speed and to negative speed. There are 2 starting point: The first test lets the motor start from zero speed and accelerates until steady state. In a second test the speed demand starts at zero and stays at zero for 0.1s before accelerating until steady state. For speed variation two different speed demands are used, 101% and 70% of rated speed.

The mean squared errors obtained by the speed demand variation are presented in Table 8.1. The speed demand referred in the first column is the type of the speed demand as explained earlier in this section. From Table 8.1, it is clear that if the speed demand is starts at zero and held at zero speed for a few seconds, both estimators perform well. The detail of each speed response is shown in Figure 8.34 and 8.35.

Speed demand	Mean Squared Errors	
	GA	SA
1(a)	N/A	614.37
1(b)	1097.2	N/A
2(a)	4.35	2.68
2(b)	0.67	1.77

Table 8.1 Mean squared error obtained for speed variation.

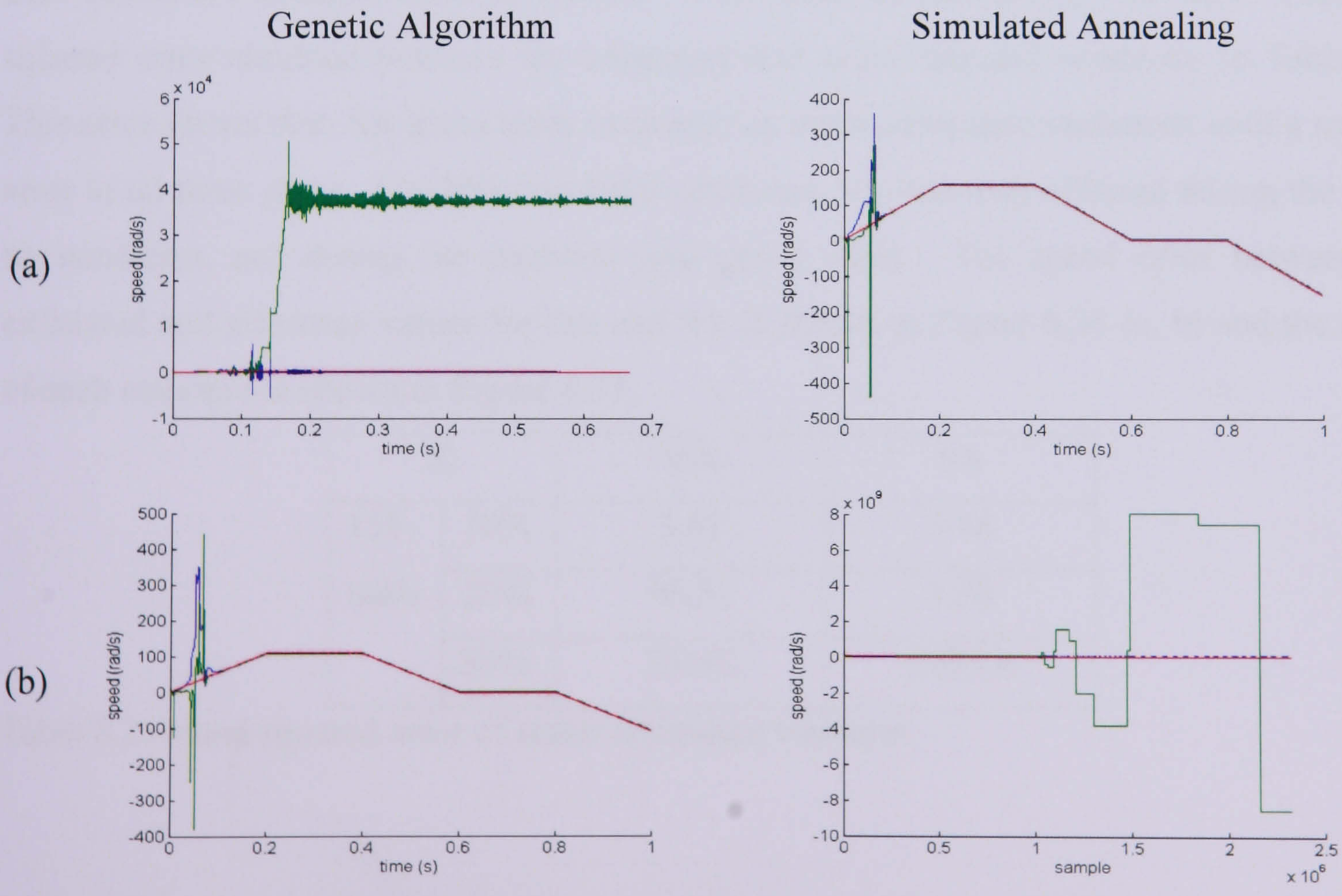


Figure 8.34: Test 1-Speed demand started at zero speed with immediate acceleration (a) 101% case (b) 70% case

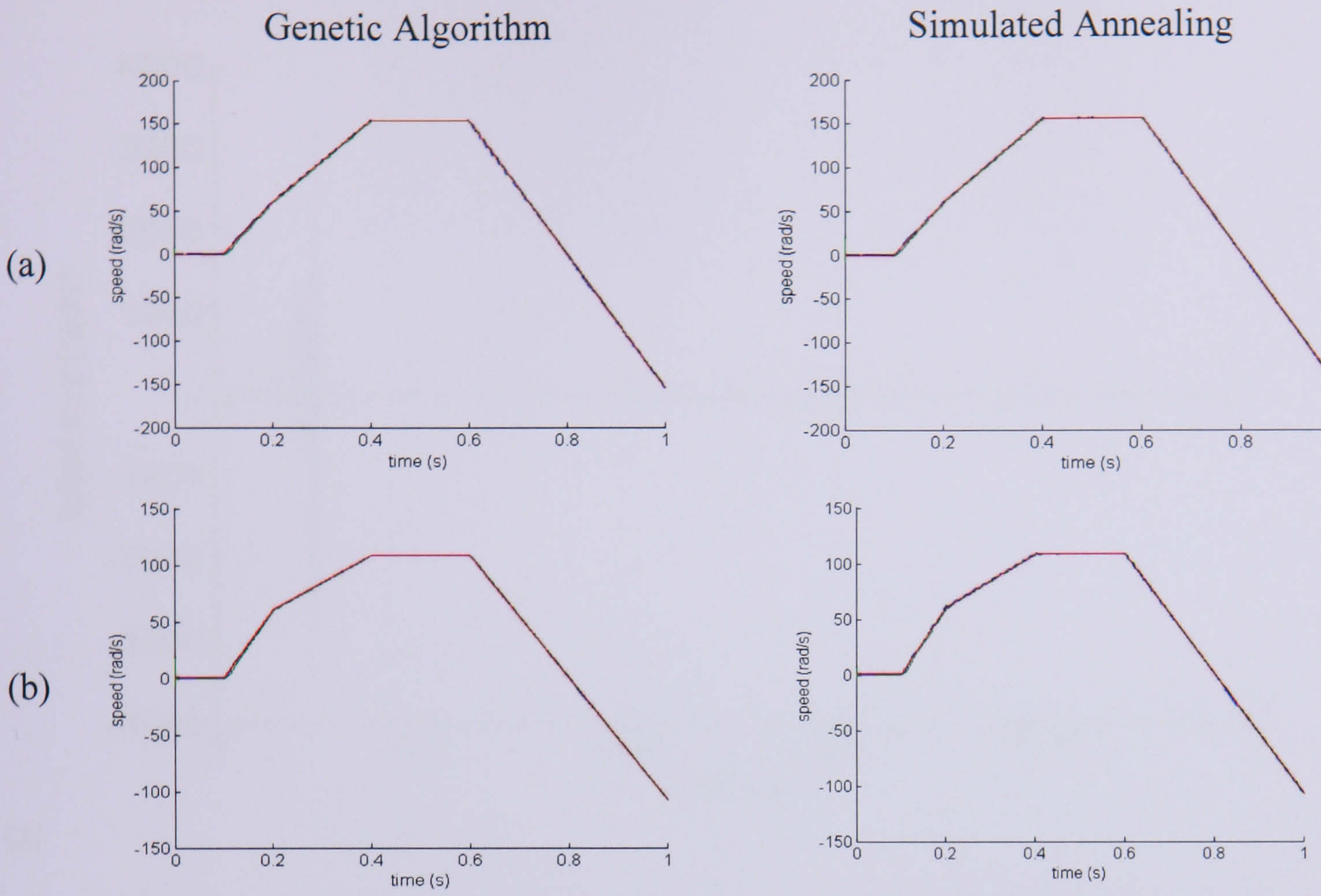


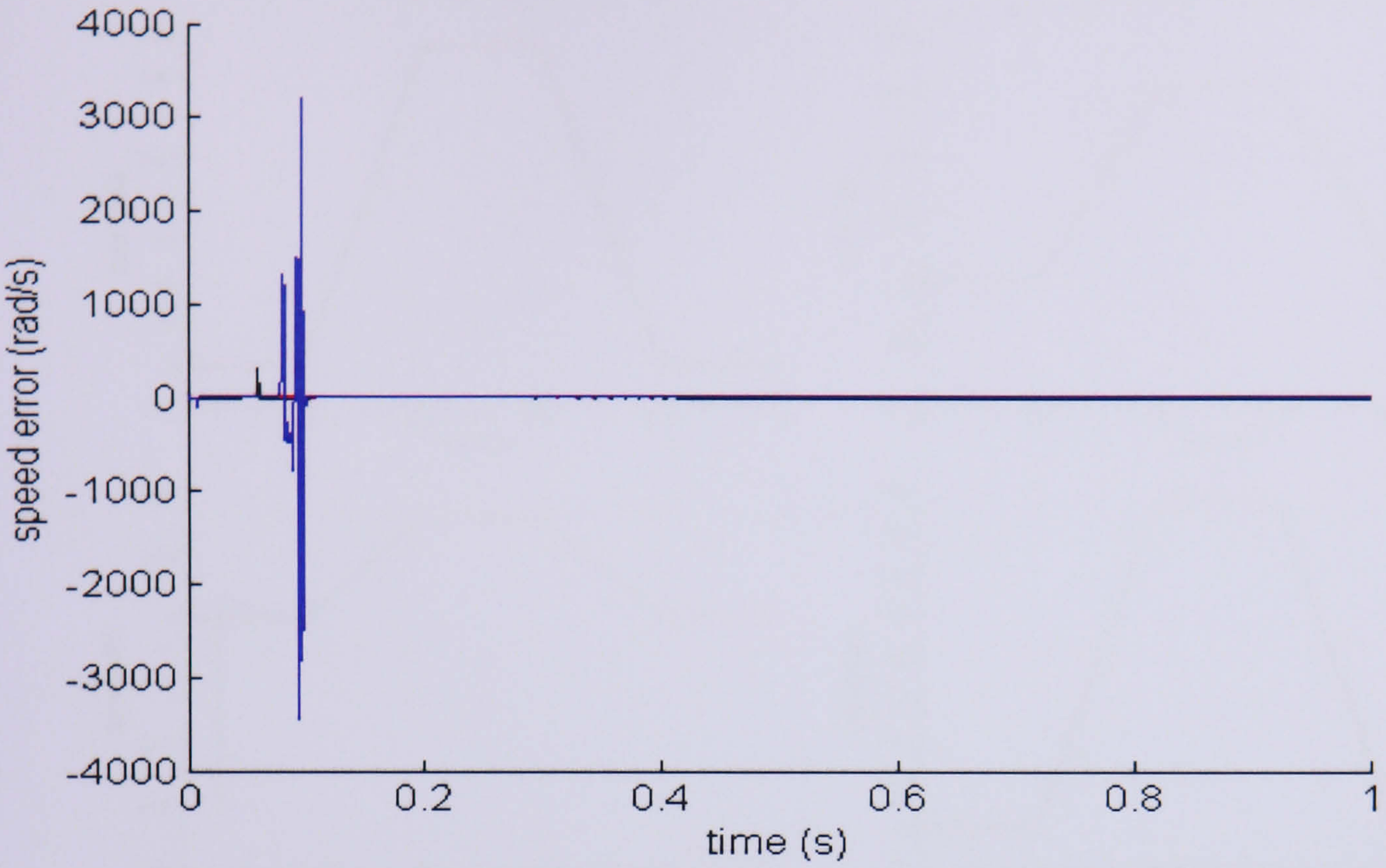
Figure 8.35: Test 2-Speed demand started at zero speed and held constant at zero for 0.1s (a)101% case (b) 70% case

8.7.2 Stator Resistance Variation

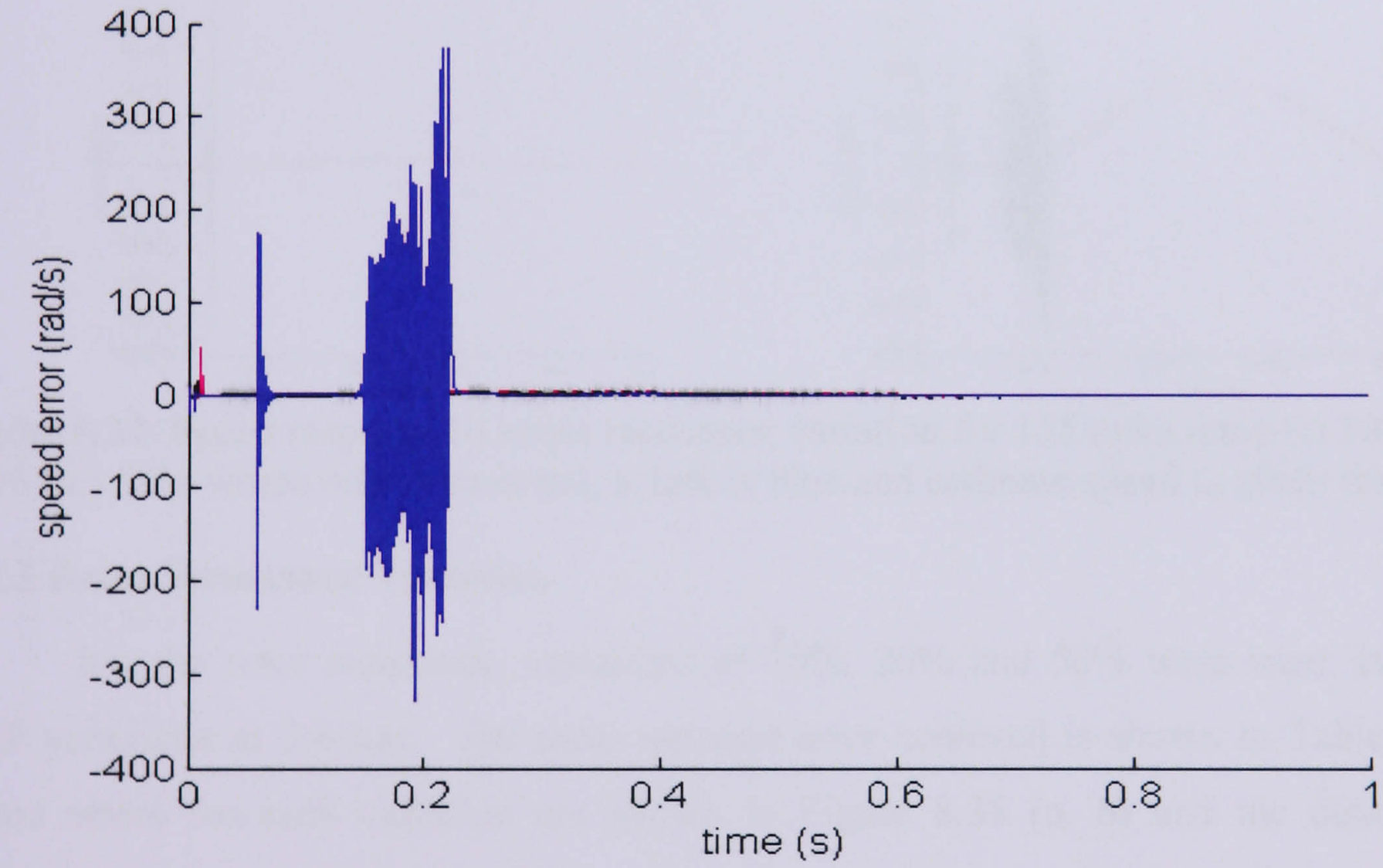
For the stator resistance, a variation of 10, 20 and 50% was used, while keeping the EKF model at a nominal value of 0.6 ohm. The speed demand used is 155 rad/s. The mean squared error obtained between the estimated and speed demand is shown in Table 8.2. This error shows that, SA gives more immunity to stator resistance variations with a smaller error in all three cases. For 10% and 20% variations, SA was only affected during the start-up condition, not during the transient and steady state. The speed error between the estimated and reference values for GA and SA is shown in Figure 8.36 (a, b) and the detail of each response is shown in Figure 8.37.

R_s		GA	SA
155 rad/s	10%	3.61	3.48
	20%	96.51	1.68
	50%	75162	2023.6

Table 8.2: Mean squared error of stator resistance variation



(a)



(b)

Figure 8.36: Error between estimated and reference speed using (a) GA (b) SA

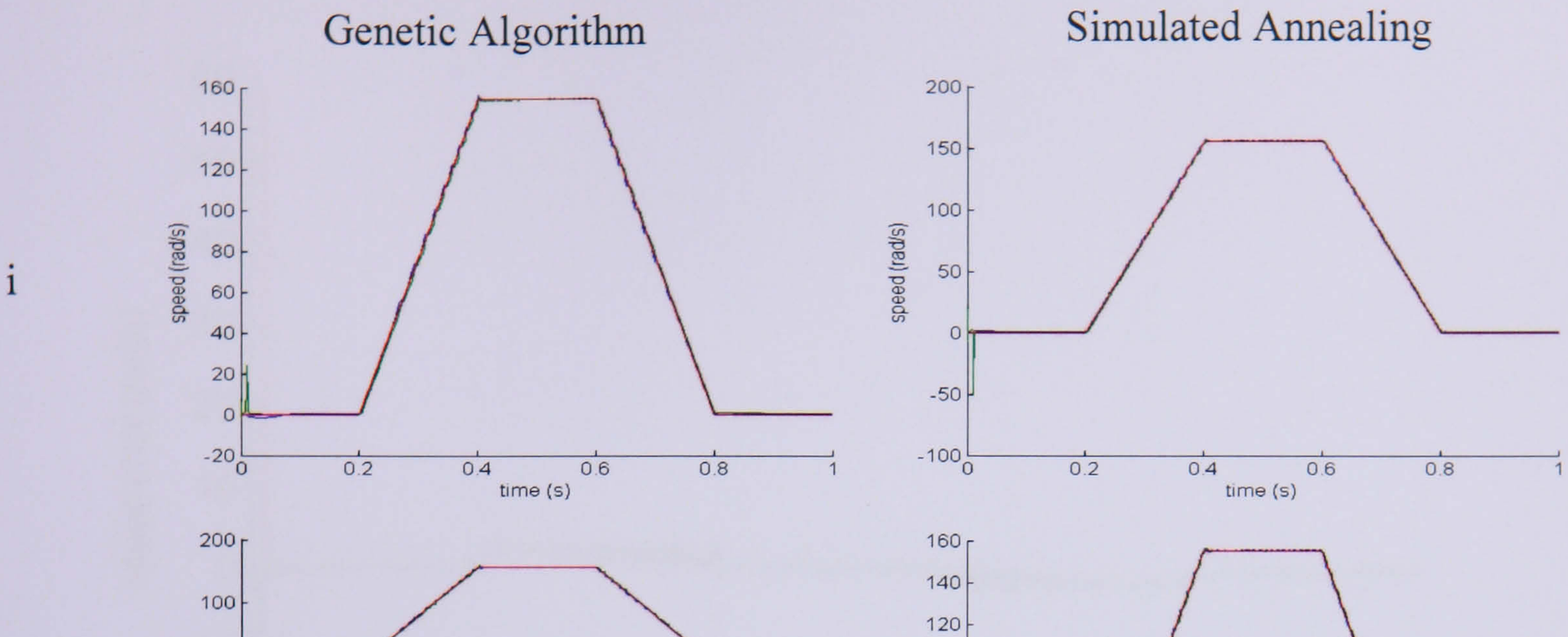


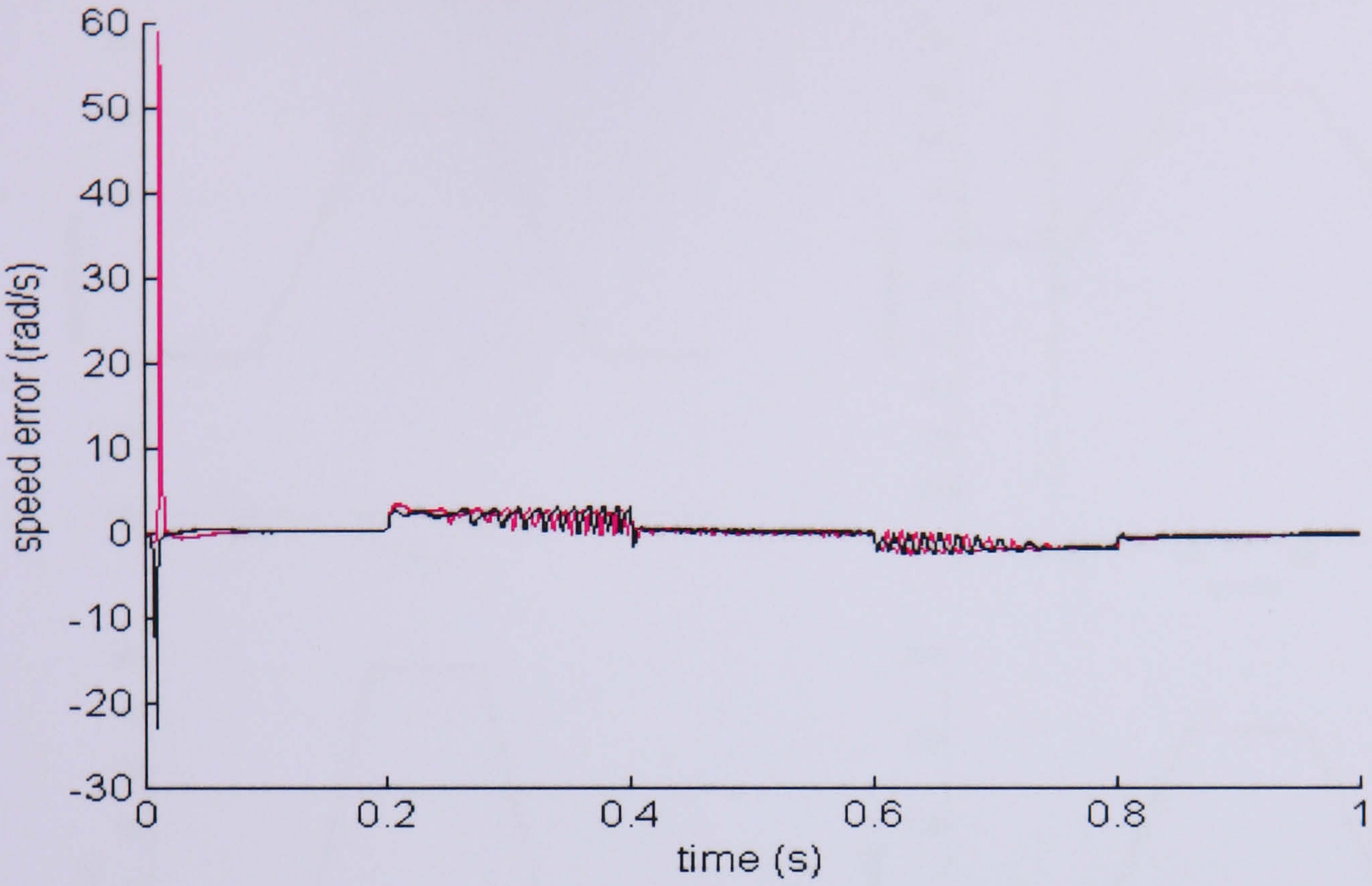
Figure 8.37: Speed response to stator resistance variation for 155 rad/s ramp (i) 10% (ii) 20% (iii) 50% where reference is red, actual is blue and estimate speed is green respectively

8.7.3 Rotor Resistance Variation

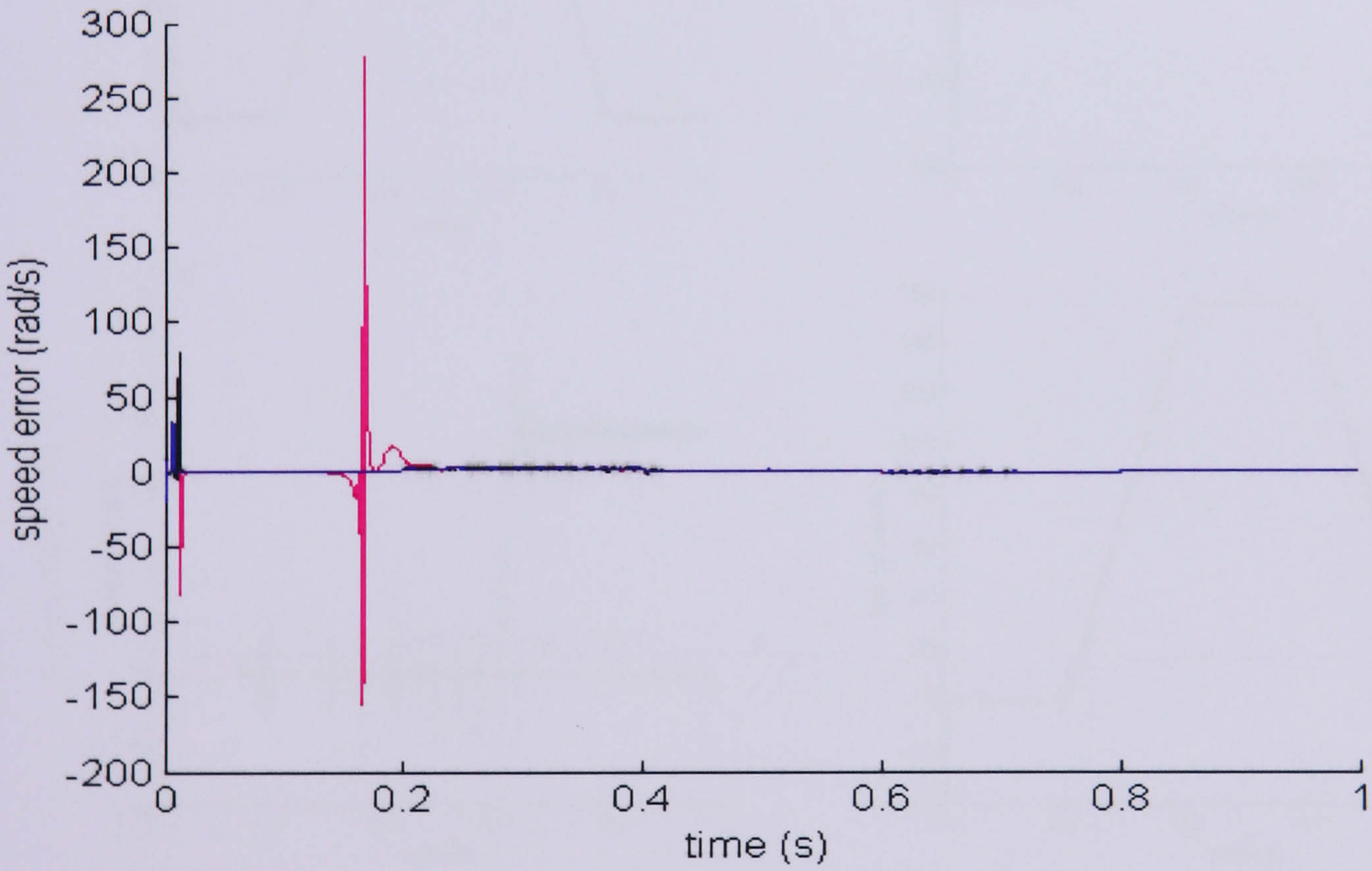
For the rotor resistance, variations of 10%, 20% and 50% were used, keeping the EKF parameter at 0.4ohm. The mean squared error achieved is shown in Table 8.3. The speed errors for each variation are shown in Figure 8.38 (a, b) and the detail of each response is shown in Figure 8.39 (c). The tolerance with SA is shown clearly for variation of the rotor resistance as high as 50%. The GA performs better than SA for 10% and 20% but not when the resistance increases to 50% of the nominal value.

R_r		GA	SA
155 rad/s	10%	7.32	151.31
	20%	2.78	5.81
	50%	N/A	1.33

Table 8.3: Mean squared error of rotor resistance variation



(a)



(b)

Figure 8.38: (a) Error between estimated and reference speed using GA for 10% and 20% variation (b) Error between estimated and reference speed using SA for 10%, 20% and 50%

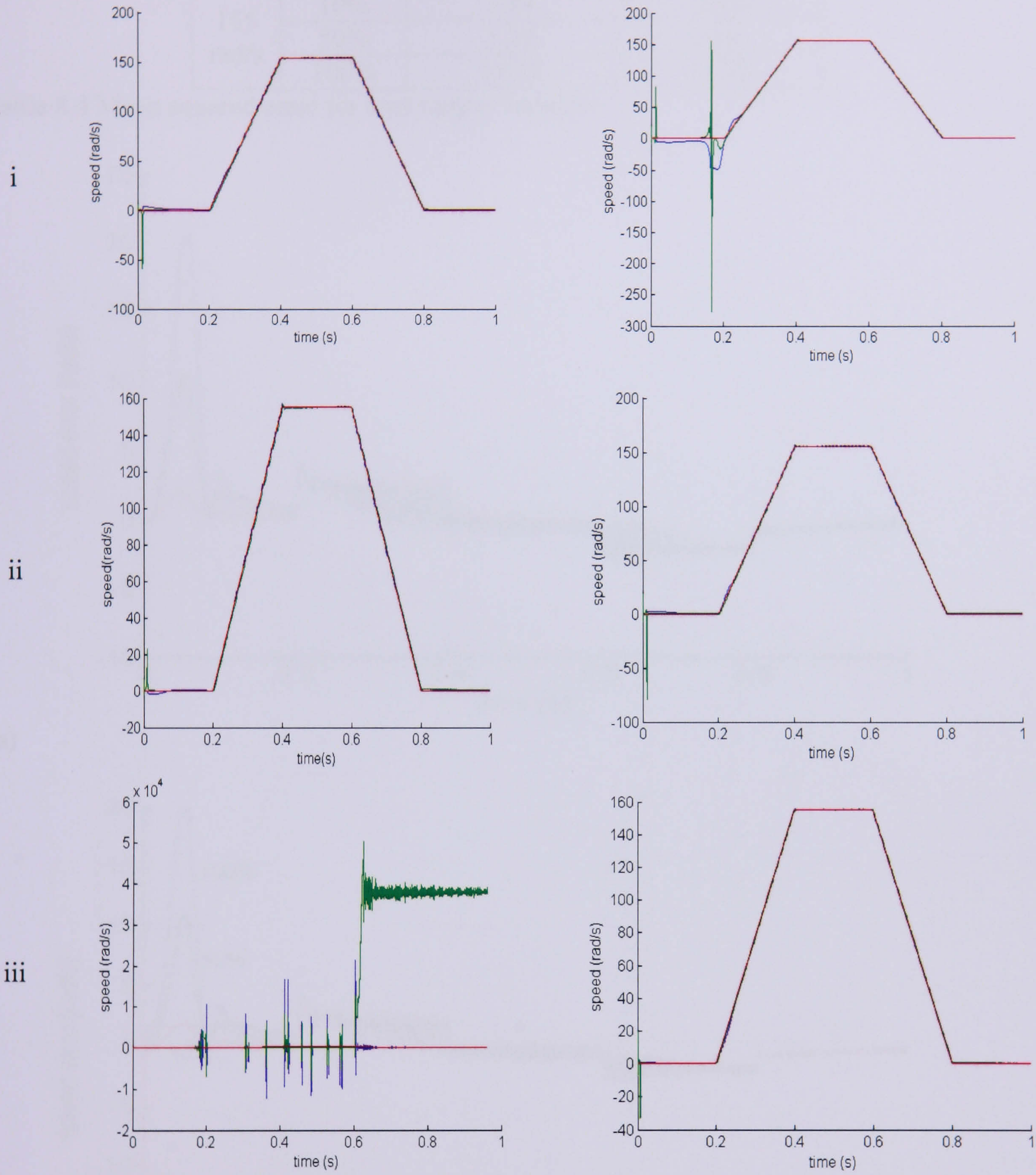


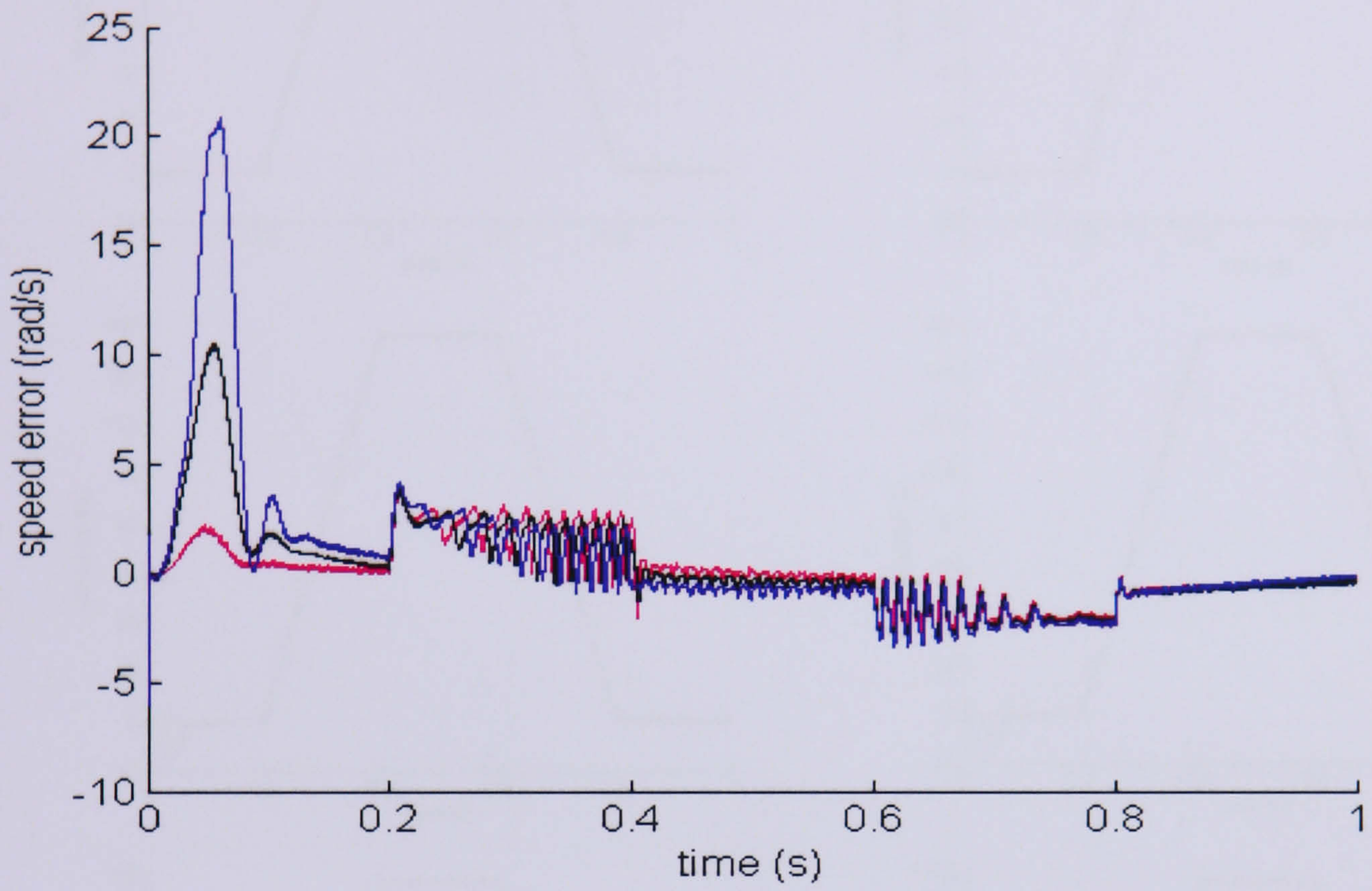
Figure 8.39: Speed response to rotor resistance variation for 155 rad/s ramp where reference is red, actual is blue and estimated is green

8.7.4 Load Torque Variation

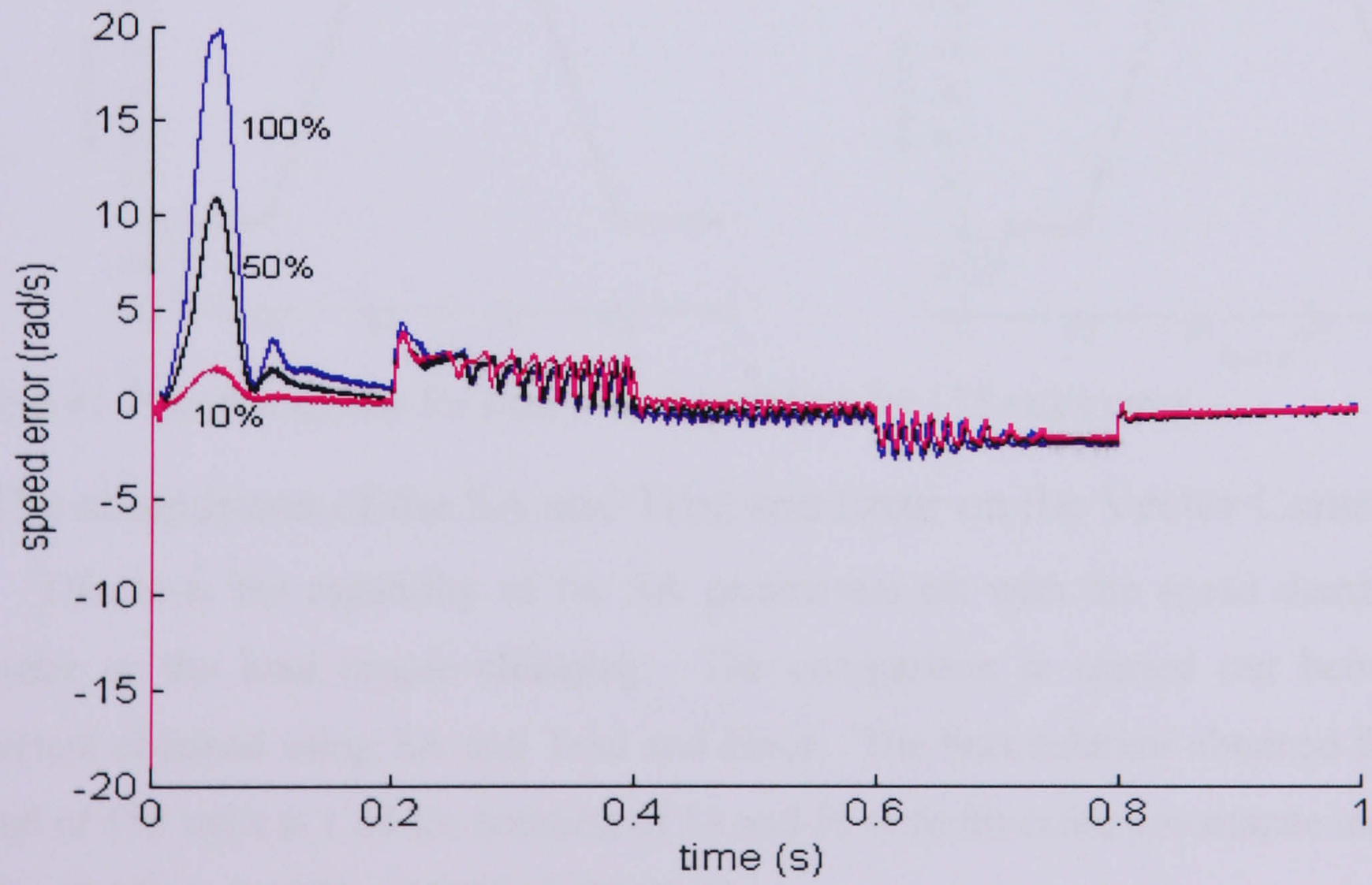
The nominal load torque is 48.84 Nm. Load torque of 10%, 50% and 100% of nominal is applied. The SA parameters performed better than GA as shown in Table 8.4 and Figure 8.40. Each of the responses is shown in Figure 8.41. Both SA and GA are affected by the load torque variation for 0.1s which is before settling to steady state. For all variations SA performs better than GA.

Load Torque		GA	SA
155 rad/s	10%	3.24	1.98
	50%	5.59	4.61
	100%	19.94	13.70

Table 8.4 Mean squared error for load torque variation.



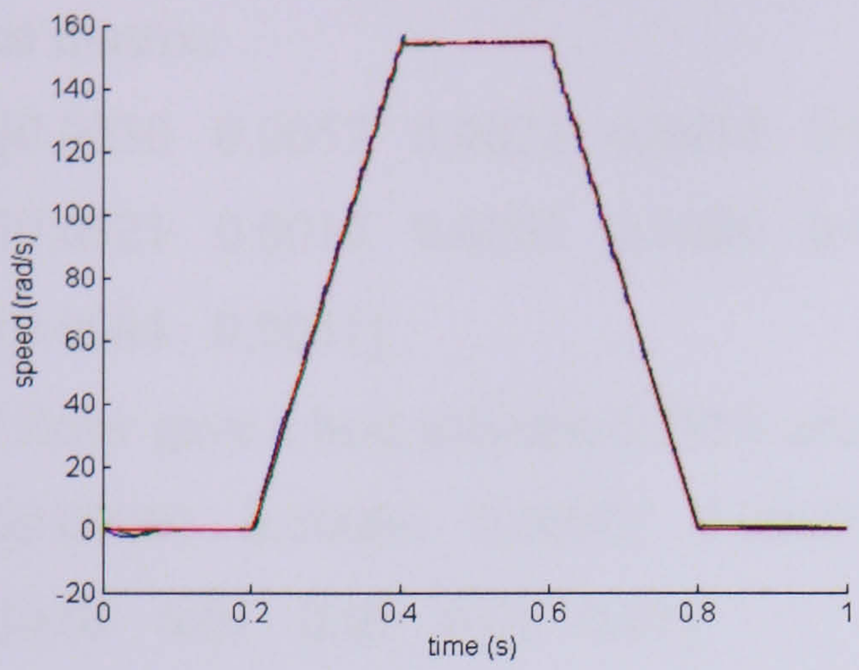
(a)

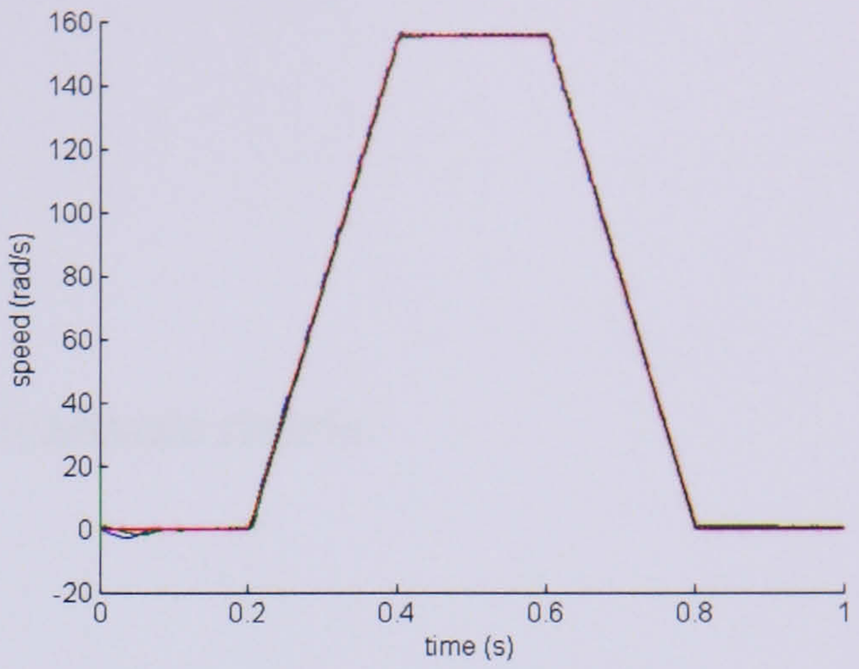


(b)

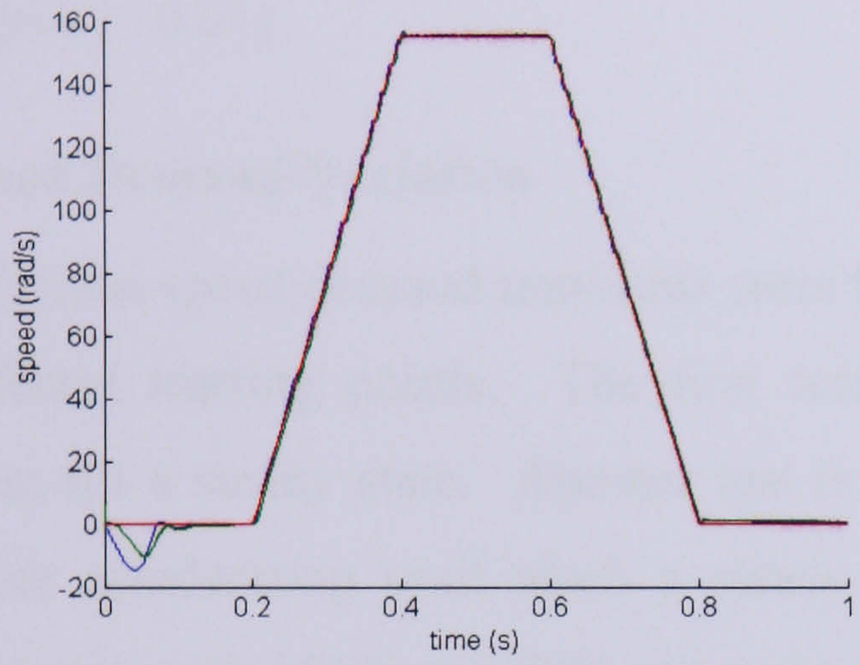
Figure 8.40: Error between estimated and reference speed using (a) GA (b) SA

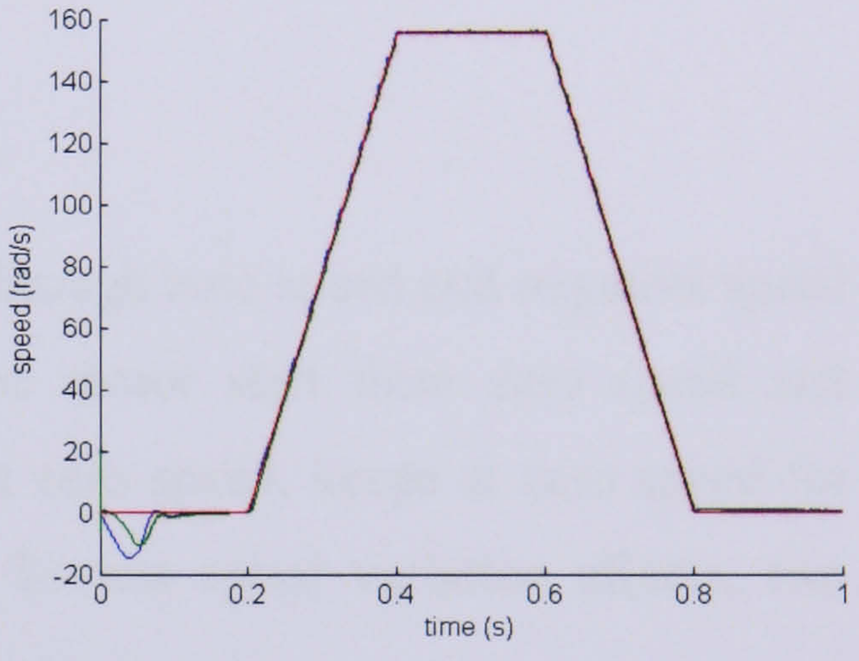
i



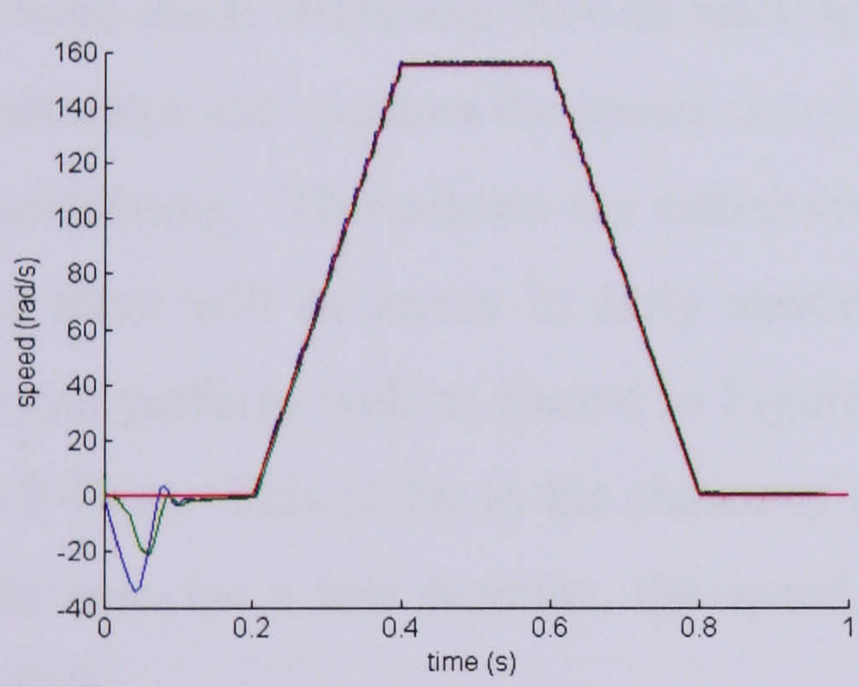


ii





iii



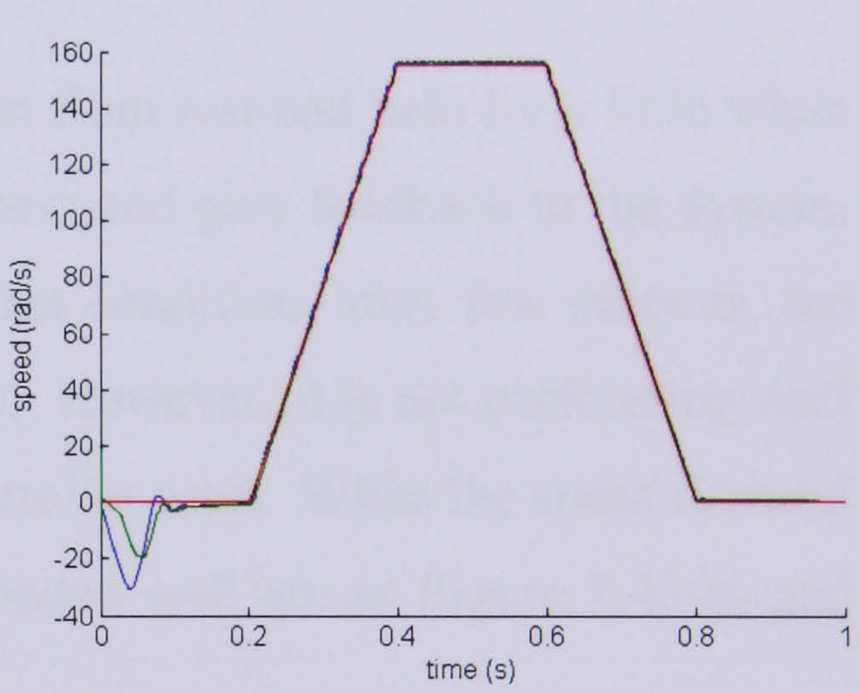


Figure 8.41: Speed response for load torque variation for 155 rad/s ramp

8.8 The comparison of the SA and Trial and Error on the Vector Control of IM

This tests the capability of the SA parameters set with the speed demand, motor parameter or the load torque changing. The comparison is carried out between EKF parameters obtained using SA and Trial and Error. The best solution obtained for a speed demand of 155 rad/s is 1.88 for iteration of 14 and 15 with the noise covariance matrix:

Q =diag [0.0065 0.0075 0.0009 0.0010 1]

G =diag [0.0078 0.0015 0.0011 0.0022 0.0071]

R= diag [0.0034 0.0023]

The best solution obtained for 108 rad/s is 0.9607 with iterations of 246-279 and noise covariance matrix:

Q =diag [0.0030 0.0057 0.0022 0.0014 0.9786]

G =diag [0.0021 0.0014 0.0005 0.0026 0.0076]

R= diag [0.0044 0.0051]

Trial and Error gave a best solution 0.7856 using the covariance matrix:

Q =diag [0.00001 0.00001 0.00001 0.00001 1]

G =diag [0.01 0.01 0.01 0.01 0.01]

R= diag [0.01 0.01]

8.8.1 Speed Demand Variation

Various speed demand transients were tested, through zero speed and negative speed with different starting points. The first test lets the motor start from zero speed and accelerates till a steady state. Another test is starts at zero speed, keeps at zero speed for 0.1s before accelerating until reach a steady state. To test speed variation effects, two demands were used: 101% and 70% of rated speed.

Estimator use requires the speed demand to start from rest and held for a little while before accelerating. This allows the estimator to process and give feedback to the system. Otherwise there will be errors in early starting. In this condition, after few seconds, the estimator can perform well as shown in Figure 8.42(a). However, it is not performing well in Figure 8.43(a). This is due to the choice of gain controller used. When the speed demand is hold for zero for a few seconds, the speed is responded well are as Figure 8.42(b) and Figure 8.43(b).

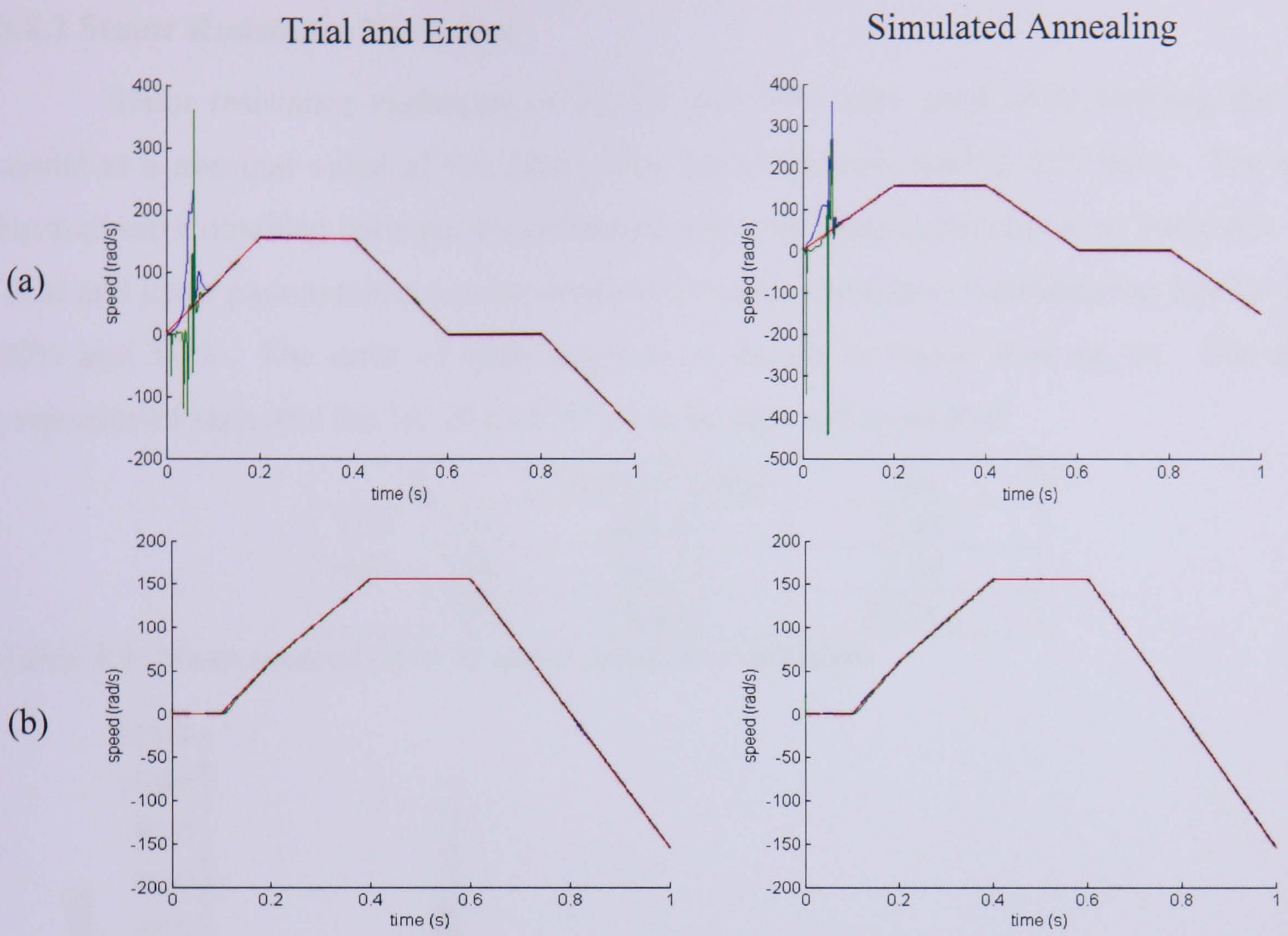


Figure 8.42: Speed demand 155 rad/s using Trial and Error and SA

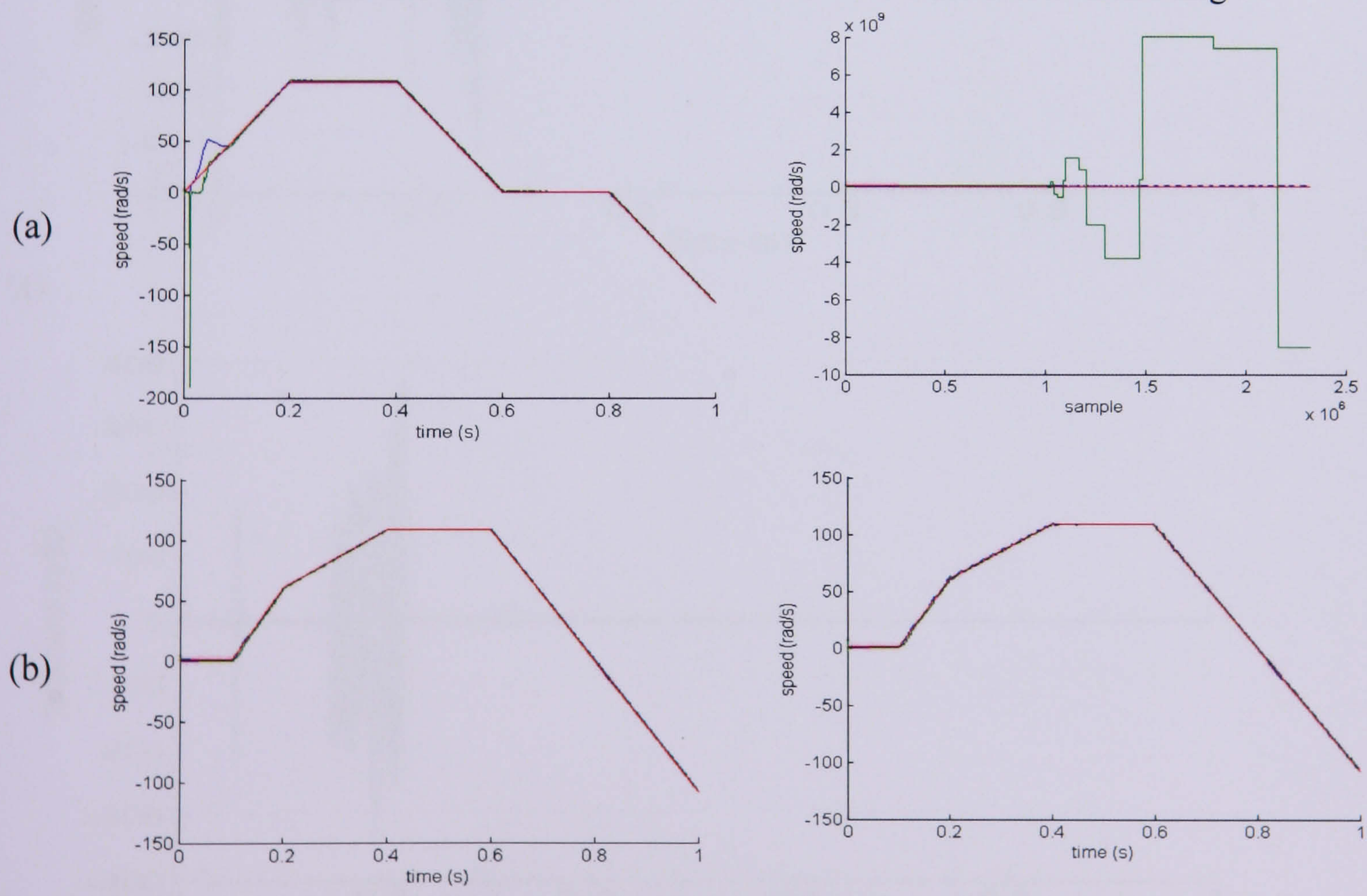


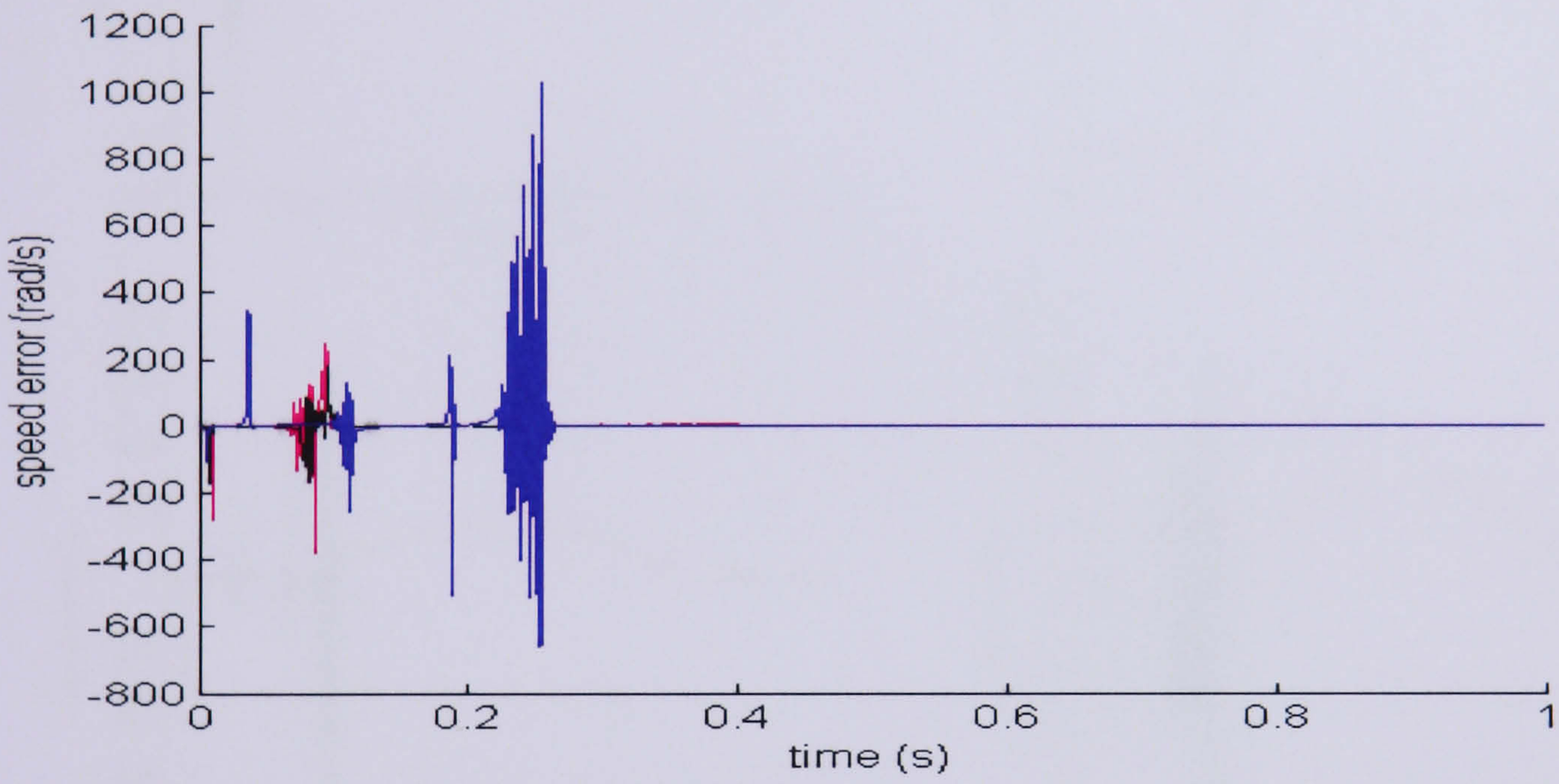
Figure 8.43: Speed demand 108 rad/s using Trial and Error and SA

8.8.2 Stator Resistance Variation

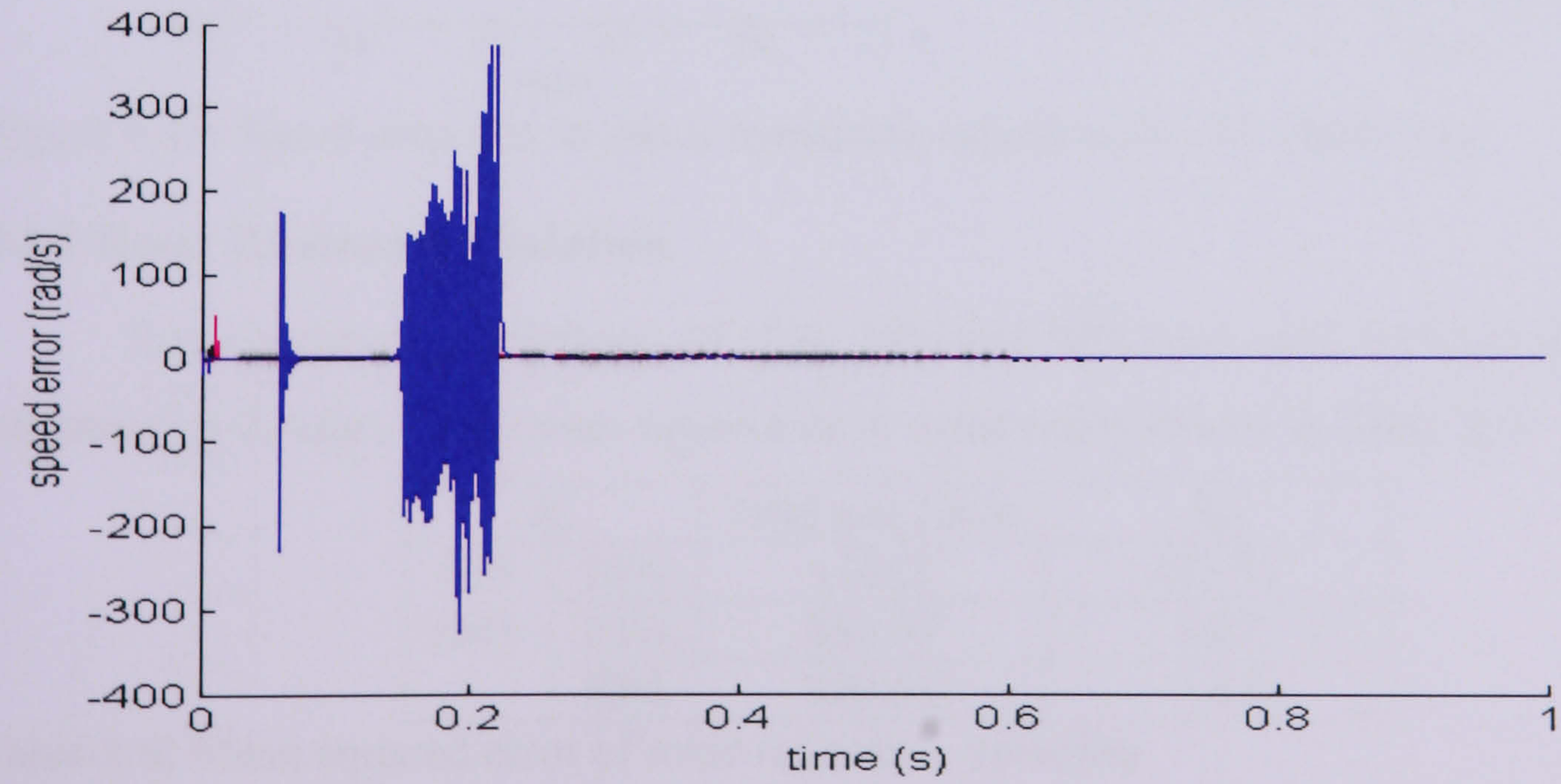
Stator resistance variations of 10, 20 and 50% were used while keeping the EKF model at a nominal value of 0.6 ohm. The speed demand used is 155 rad/s. The mean squared error obtained between the estimated and speed demand is shown in Table 8.5. The Trial and Error parameters are more sensitive to stator resistance variation than SA for 10%, 20% and 50%. The error of each response is shown in Figure 8.44 (a, b). The speed responses of each trial for 10, 20 and 50% can be seen in Figure 8.45.

R_s		Trial and Error	SA
155 rad/s	10%	463.61	3.48
	20%	165.13	1.68
	50%	5025.0	2023.6

Table 8.5: Mean squared error of stator resistance variation



(a)



(b)

Figure 8.44: Error between estimated and reference speed (a) Trial and Error (b) SA.

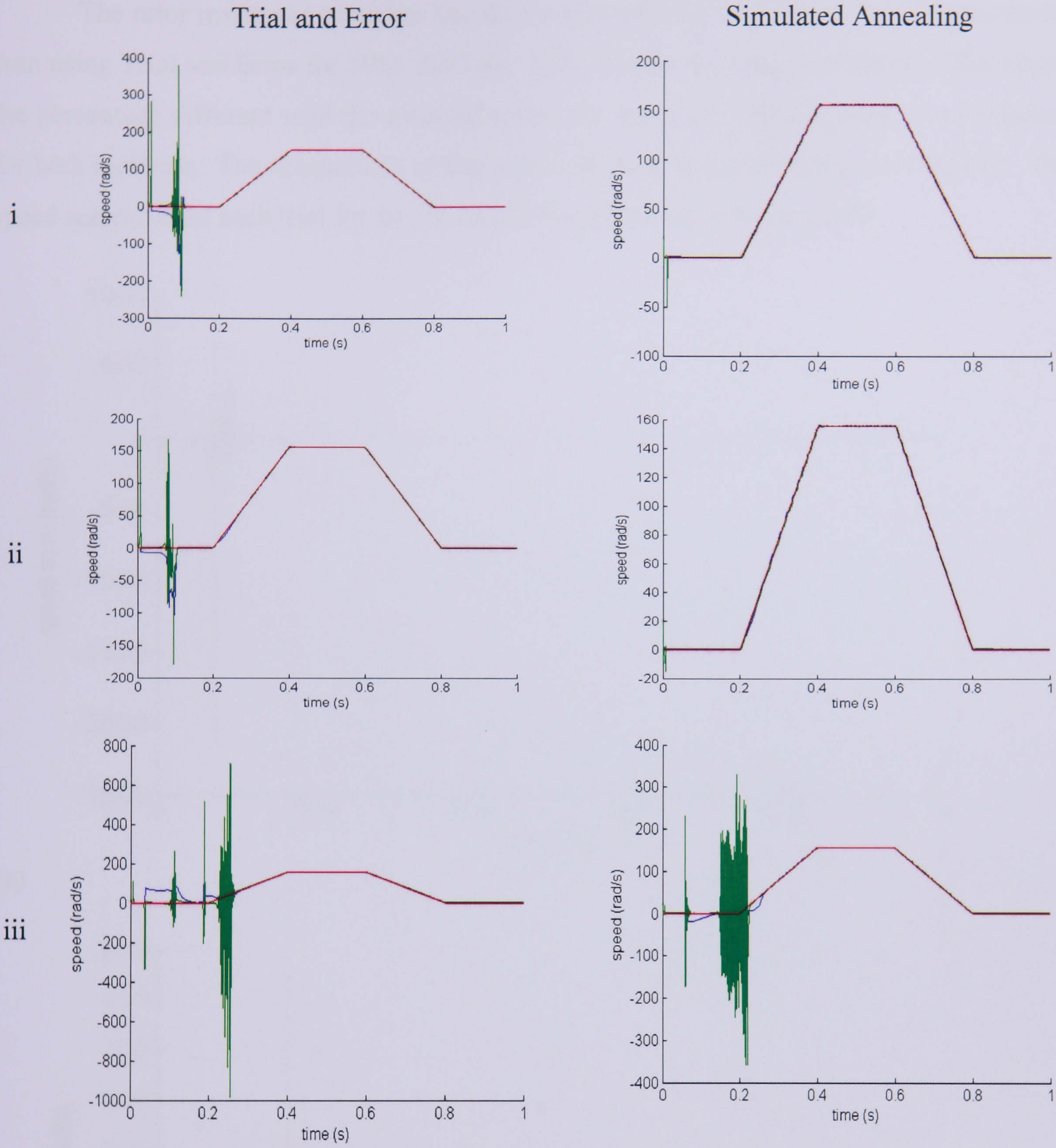


Figure 8.45: Speed response in stator resistance variation for 155 rad/s ramp

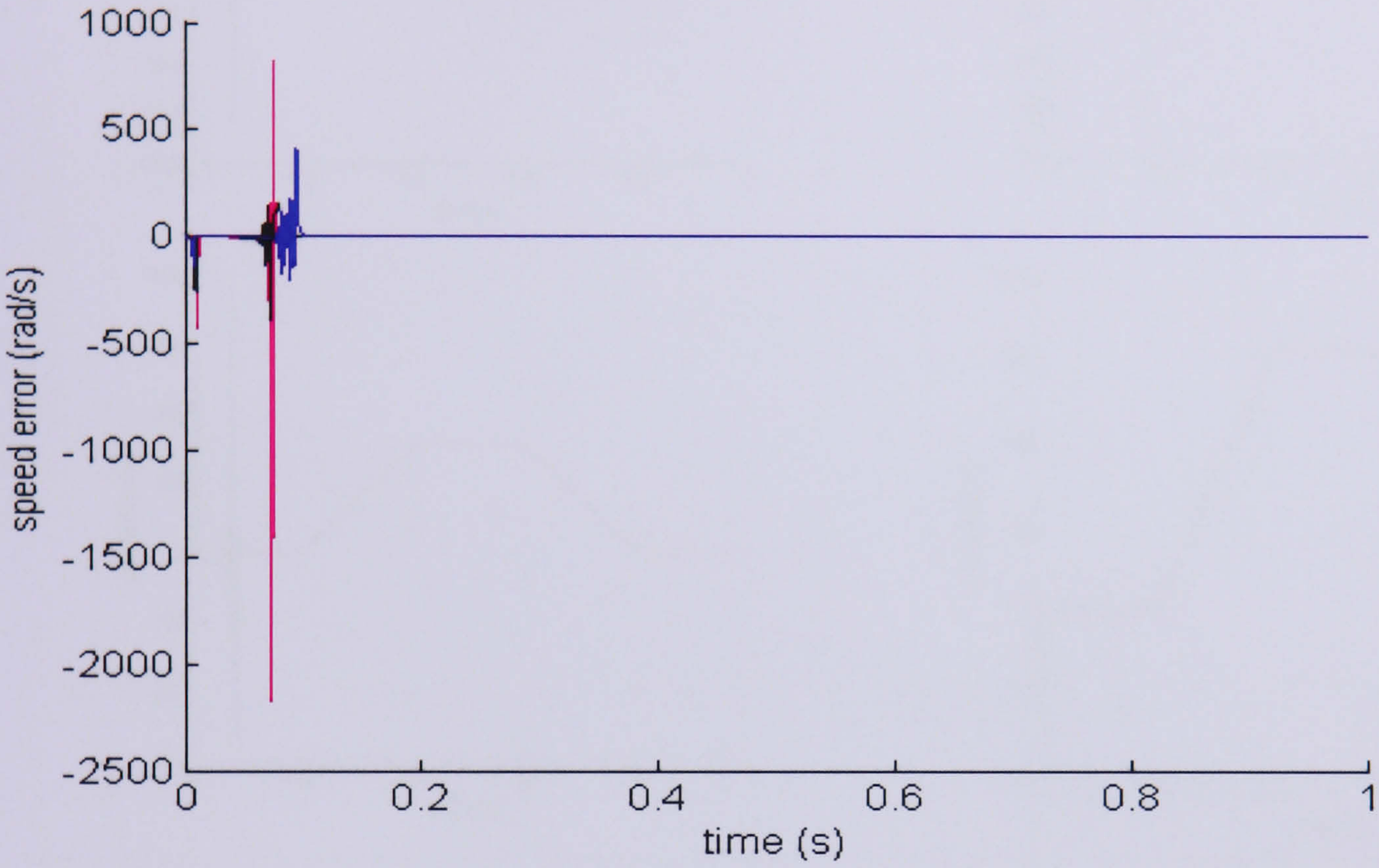
8.8.3 Rotor Resistance Variation

Rotor resistance variations, of 10%, 20% and 50% were used while keeping the EKF parameter at 0.4ohm. The mean squared error achieved is shown in Table 8.6.

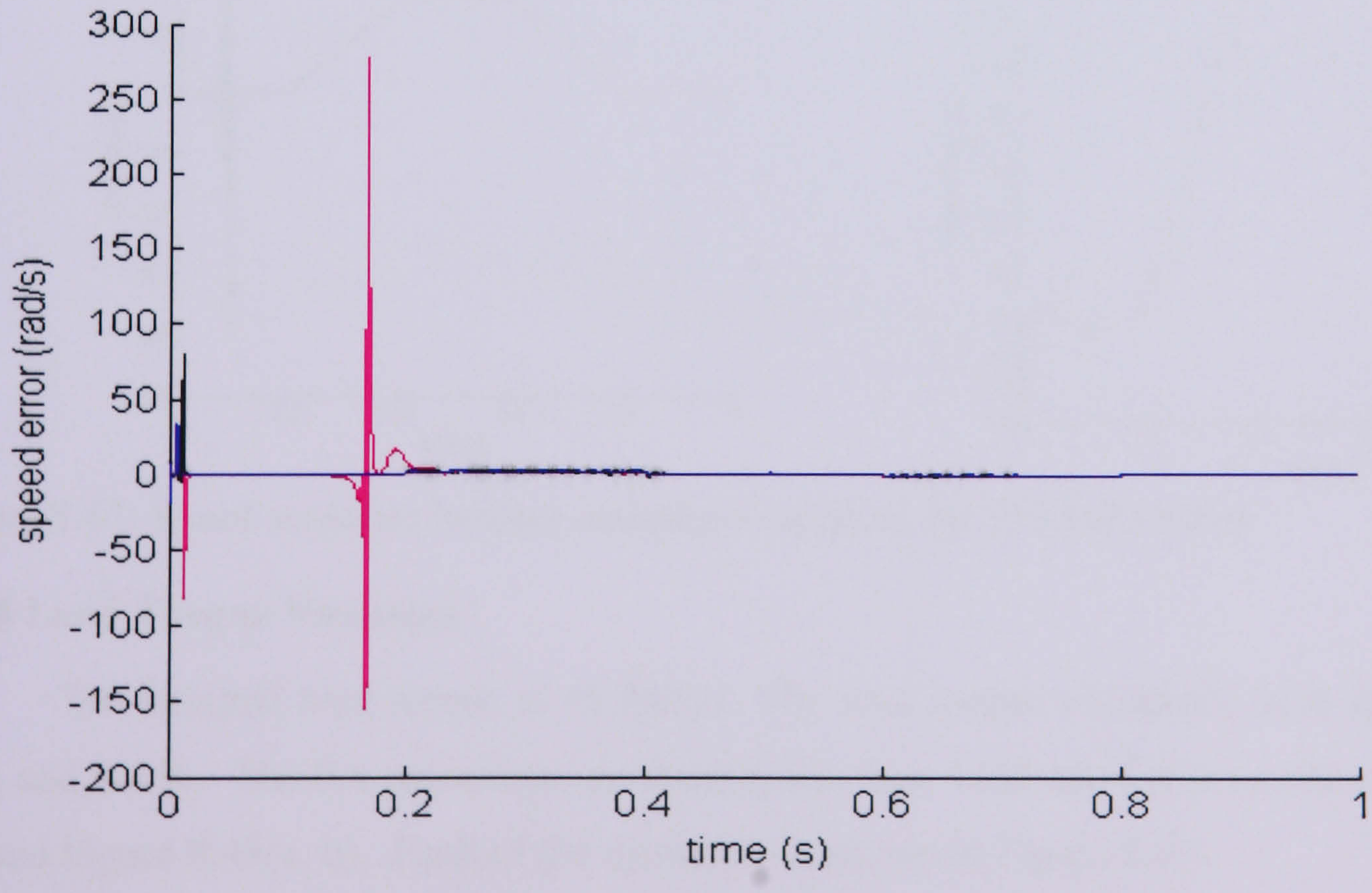
R_r		Trial and Error	SA
155 rad/s	10%	14862	151.31
	20%	404.07	5.81
	50%	350.53	1.34

Table 8.6: Mean squared error of rotor resistance variation

The rotor resistance variation has shown that the EKF tuned by SA performed better than using Trial and Error for 10%, 20% and 50% of rotor resistance variations. The bigger the percentage different with the nominal value, the lesser the mean squared error obtained for both methods. The comparison of the errors obtained is shown in Figure 8.46(a-b). The speed responses of each trial for 10, 20 and 50% can be seen in Figure 8.47.



(a)



(b)

Figure 8.46: Error between estimated and reference speed using (a) Trial and Error (b) SA.

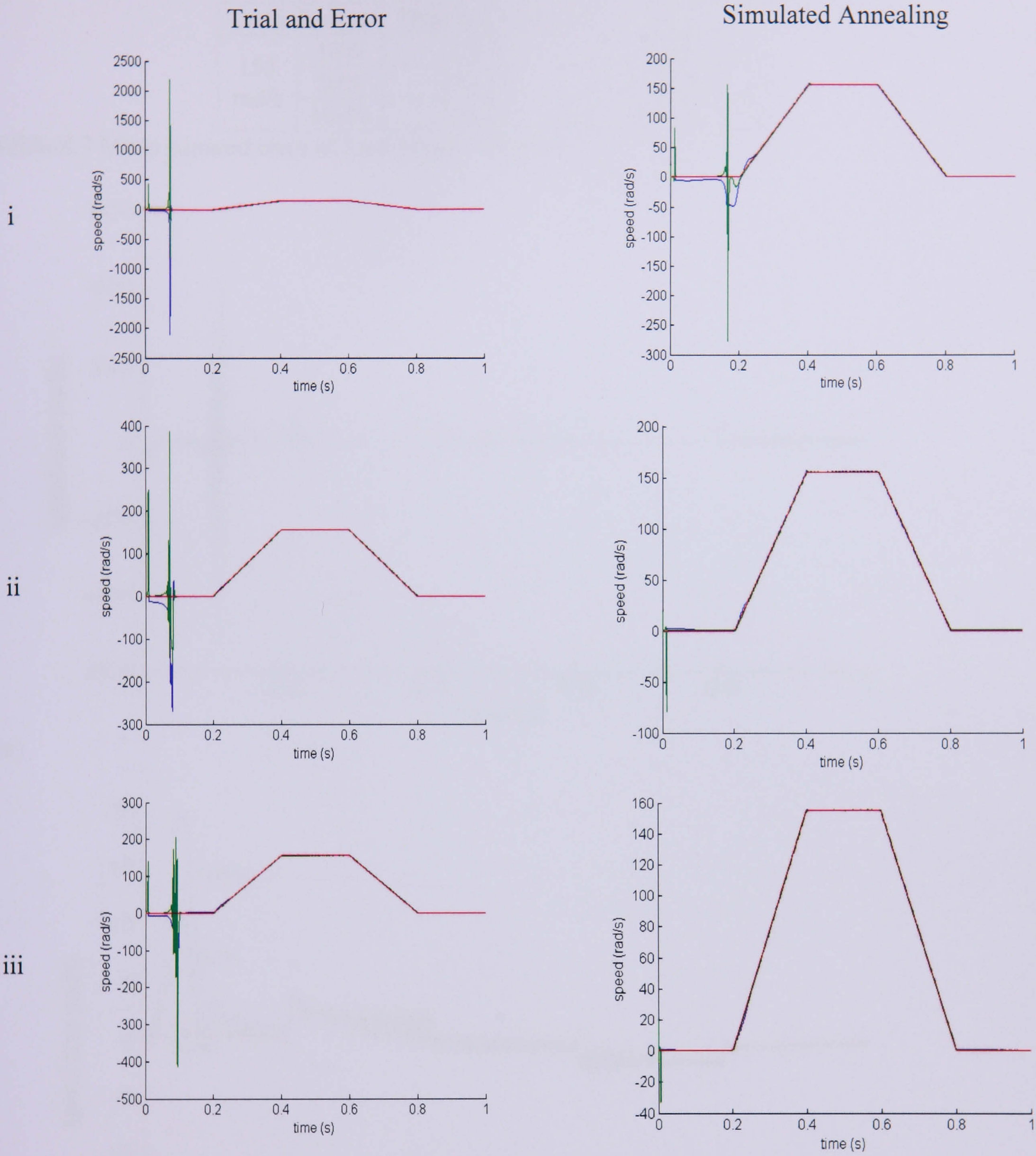


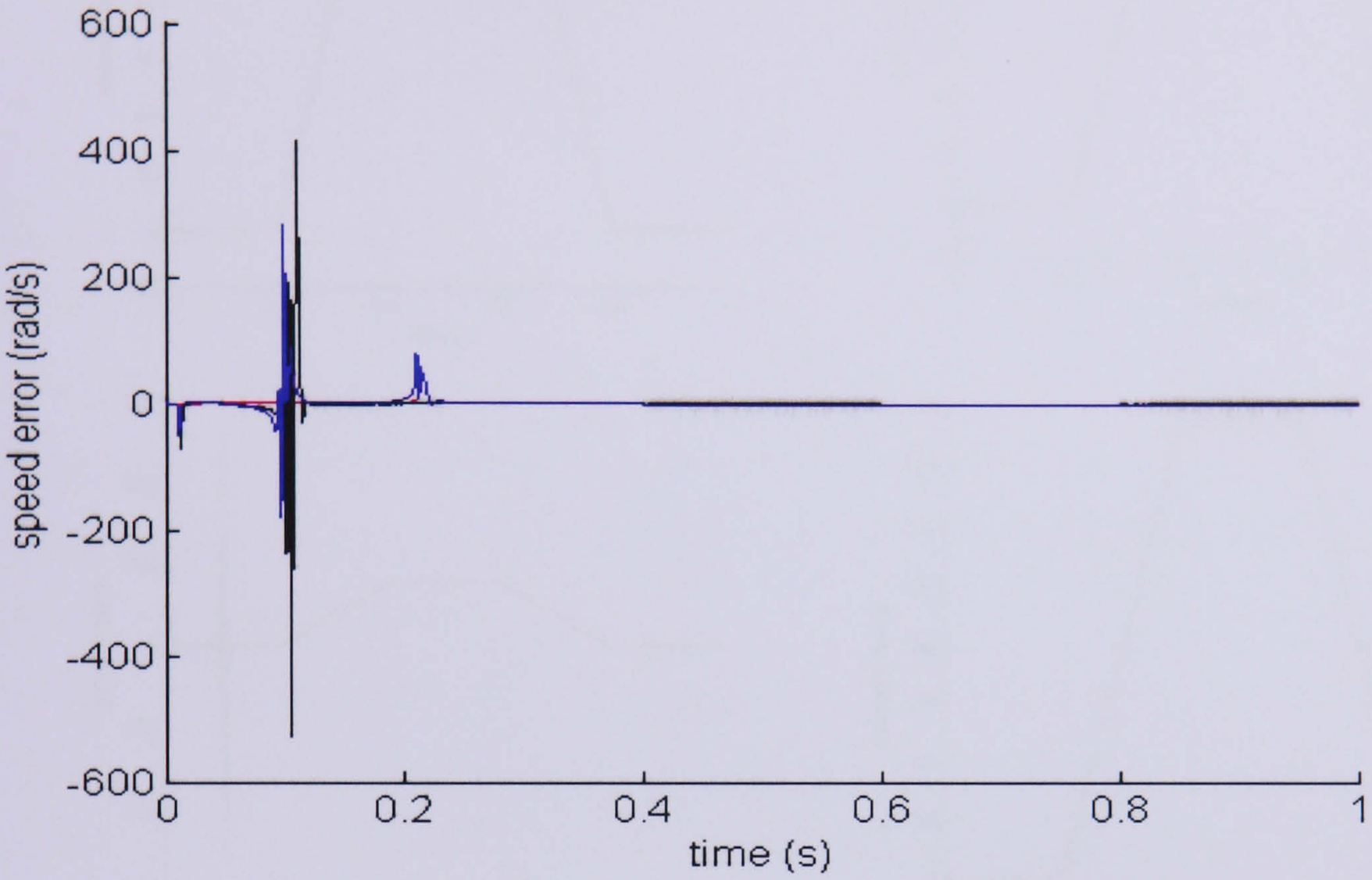
Figure 8.47: Speed response in rotor resistance variation for 155 rad/s ramp

8.8.4 Load Torque Variation

The nominal load torque is 48.84Nm. The load torque variations were again 10%, 50% and 100%. The SA parameters perform better than Trial and Error as shown in Table 8.7 and Figure 8.48(a, b). Each of the responses is shown in Figure 8.49.

Load Torque		Trial and Error	SA
155 rad/s	10%	1.067	1.98
	50%	740.55	4.6078
	100%	503.31	13.70

Table 8.7 Mean squared error of load torque variation.



(a)

(b)

Figure 8.48: (a) Error between estimated and reference speed using SA (b) Error between estimated and reference speed using SA for each variation

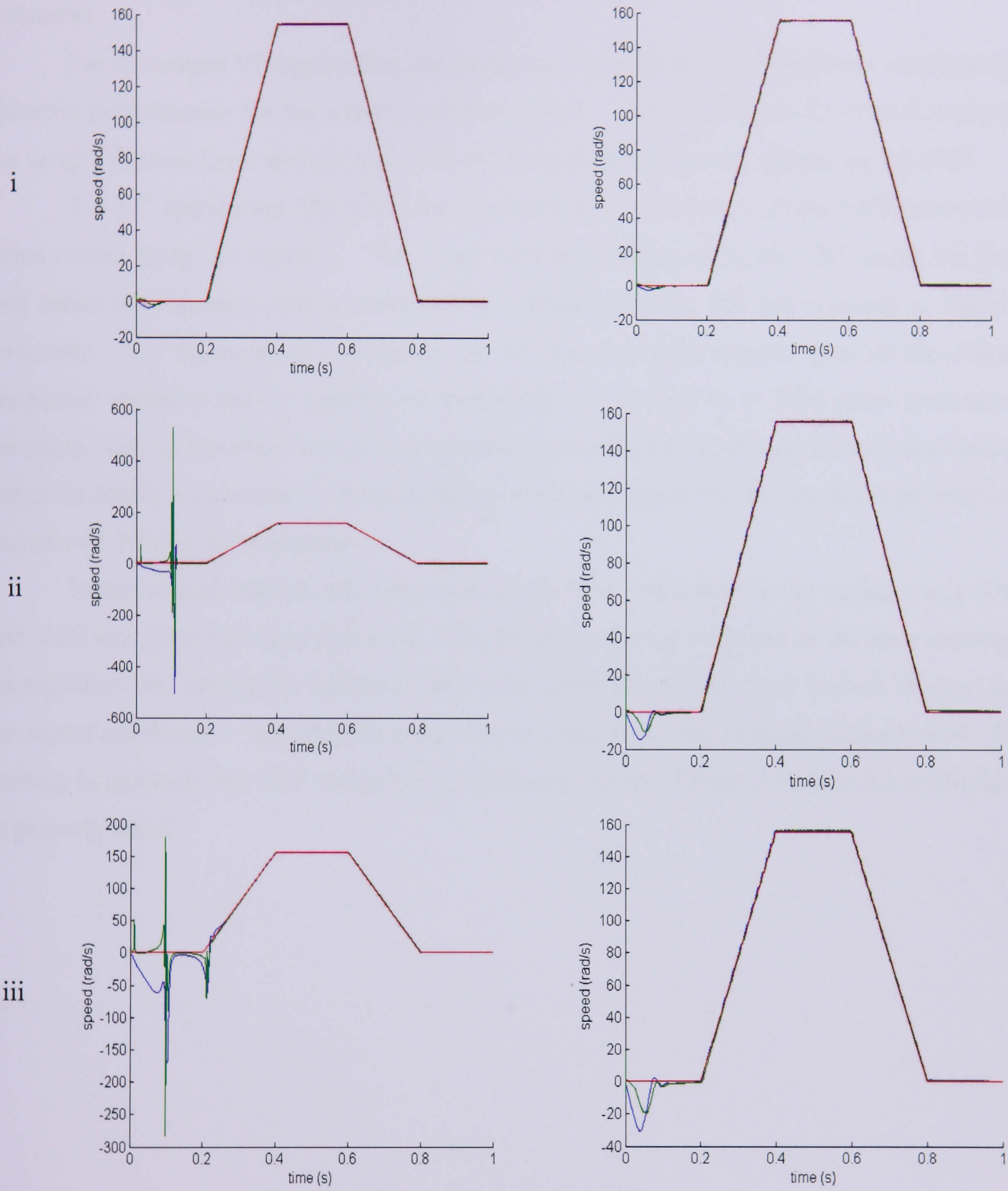


Figure 8.49: Speed response in load torque variation for 155 rad/s ramp

8.9 Conclusions

This chapter gave a comparison of the performance of the proposed method, SA, GA as well as the common Trial and Error method. The application of the GA in tuning the EKF for constant V/f and VC drives was covered. Then the performance of each method in tuning the EKF including robustness was discussed. Estimated speed, speed demand

variations, stator and rotor resistance discrepancy and the change in the load torque were discussed.

For a constant V/f application, the proposed SA worked well but did not insure good dynamic performance for the whole operating region. However, it can be used to replace the usual Trial and Error method and offer an alternative solution for optimising the EKF.

For VC application, the first section presented the robustness of the EKF parameter when tuning using SA and GA. The speed estimation obtained by the EKF using the SA had better performance during transients as compared to the GA but resulted in higher overshoot. For the parameter variation, the SA has a slightly smaller error in the stator resistance variation and is capable of estimating for as high as a 50% rotor resistance variation. GA on the other hand is very good at estimating the speed against rotor resistance variation which is important in the application of the sensorless VC, but cannot cope with an increase of 50% of rotor resistance.

In the second section, the robustness of the EKF parameter when tuning using SA and Trial and Error had been discussed. The errors are mostly obtained in the early starting up condition and during the transition from zero speed to transient, and recover as soon as the motor accelerates. Ignoring the early start up condition, both methods perform well. If starting is included, the EKF using SA performed better in all trials provided the controller is properly tuned.

CHAPTER 9: CONCLUSIONS AND RECOMMENDATIONS

The work in this thesis focuses on the application of the EKF for the speed estimation of the IM. The main objective was to produce an alternative method for tuning the EKF parameters to improve the performance of the estimator including robustness against parameter variation. Continuous and discrete IM models were developed in standard d-q axis form. These models were validated against well tried and tested continuous models from the Research Group's Drives Library.

A speed sensor is often used in an IM drive and the measurement from the sensor feedback to the controller. This device adds extra cost and connections to the motor and may not be installed in a hostile environment. Speed sensorless methods offer great potential advantages. Knowledge of the rotor flux of the motor in an IM drives controller is essential. Therefore a KF was used to estimate the rotor flux of the motor using the measurement of the stator voltage and current. In addition speed estimation was required. To be able to simultaneously estimate the rotor flux and the speed, the EKF was required. The combined flux and speed estimator was tested using Matlab / Simulink on a AMD Athlon ™ XP 2600+ 2.09 GHz, 1.00 GB of RAM computer.

The performance of the EKF was tested using a simple direct sinusoidal sine wave power supply and also using constant V/f. In some applications the drive needs to be run at variable speed therefore the estimator was tested for variable speed and load torque conditions. The robustness of the estimator against parameter variation was also tested. The results show the estimated speed was very close to the measured in a variable speed condition provided the EKF parameters used was properly tuned. Considering the diagonal matrix only for noise covariance, the EKF requires 12 parameters for optimisation, a very demanding task. Tuning was initially done by Trial and Error. Robustness was also tested against stator and rotor resistance variations. This showed robustness for stator resistance variations in steady state operation but was very sensitive with rotor resistance variation. This tuning method cannot guarantee best performance of the speed response because of this parameter variation sensitivity.

Aiming to find an alternative tuning method for the EKF parameters, the SA Algorithm was utilised. The SA algorithm has proved to be a powerful tool in optimisation

in the literature. Based on a crystallisation process where the alloy is heated up and then cooled down slowly to allow its atoms to reach a configuration of a perfectly regular crystal, a very high temperature is required hence a large searching space is needed and long computation time is involved.

The use of SA for tuning the EKF was studied for three different control approaches, a direct sine wave supply, a constant V/f and direct VC. For these control approaches, the SA maintained the same values for its parameters, therefore the computation time of the tuning process appears not affected by the types of the control used but does depend on the time for the complete motor run. The offline tuning done by simulation does give the possibility of a smaller area to be identified which would reduce computation time. Hence results obtained during offline tuning can be used for a set of guides in searching online.

For a direct sinusoidal supply, and then a constant V/f control, the performance of the SA application for tuning the EKF for speed estimation was shown. Chapter 6 demonstrates the improvement in the performance of the speed response as compared to Trial and Error. SA gives much better convergence i.e. a smaller mean squared error could be obtained. SA was also shown to have better performance and be less sensitive for stator and rotor resistance variations.

In high performance IM drives direct and indirect VC is widely applied. In this present work direct VC utilising the rotor flux to determine the field angle was used. The application of the estimator has been tested with and without sensor feedback to the controller. Both open loop and closed loop estimators have been used with a proportional-integral controller for both speed and current controller. It has been shown that the performance of the VC also depends on the settings of the controller. For the controller to performed better, a large proportional and integral gain is required. A very high value of proportional gain can cause execution problems. Therefore two sets of the controller gain were chosen, termed SET A and B. SET A and SET B differ in that SET A can be used in the open loop estimator, but has difficulties in the closed loop estimator. SET B has been successful for both conditions but with lowered performance. The tuning of the controller introduces extra complication into VC IM drives. The performance of the EKF for direct VC drives was demonstrated. In both conditions, with and without sensors, the estimator performs well.

A SA algorithm has been used for the offline optimisation of the EKF parameter with direct VC. The closed loop estimator showed better performance than open loop in

estimating the speed demand however it requires longer computation. The convergence to the best solution was more settled on the closed loop than for the open loop estimator. The bigger the search area for SA, the longer is required for the algorithm to converge to an optimum solution.

In Chapter 8, the application of the GA to optimising the EKF is given. Since GA is one of the powerful tools in optimisation it was chosen for comparison purposes to verify the effectiveness of the proposed method. GA itself was shown to be able to estimate the speed of the motor in both constant V/f and direct VC.

In a constant V/f drive, all three methods are applicable of estimating the speed for different conditions. With stator resistance variation, SA has been shown to be able to estimate correctly even for a 50% increase, while Trial and Error can be used for 25%; while GA was found to be the most sensitive to stator resistance variation. For rotor resistance variation, SA worked well but did not always ensure good dynamic performance in the transient, but was capable of estimating the speed for as high as a 50% increase.

Using direct VC a comparison on a closed loop estimator was carried out. GA and SA show good performance, the changing parameters have an affect on the early start-up condition especially with SA. For stator resistance variation, SA was less affected with a smaller error for all three variations tested. However, with rotor resistance variation, GA performed better at 10% and for the 20% change used in this case but SA was robust for as high as a 50% variation. Trial and Error was very sensitive to stator and resistance variation as compared to SA.

This new work has shown that the proposed method, SA, can optimise the EKF performance by providing the parameters to be used for estimating the speed in both transient and steady state conditions, including start up. The proposed method was shown to perform effectively under conditions of constant and variable speeds during parameter variations. The proposed scheme can be used for replacing the usual Trial and Error method and offers an alternative solution for optimising the EKF.

9.1 Future Works

The continuous time IM model has been compared with the IM model from the University's Drive Library which is a conventional sine wave of inductance, d-q axis, and simple model. It is common practice to design on the basis of such a simple model. A more realistic representation of a real drive, with a more comprehensive model incorporating 2nd

order features such as hysteresis losses, gap flux harmonics, temperature effects on resistance and magnetic saturation, and including power electronic effects, would be more related to experimental conditions. The investment of replacing the Simulink representation of the KF to a C coded s-function would be worthwhile as it is more suited to industrial application and would contribute to much faster computation.

The optimisation of the EKF using the proposed method was simulated on a lower performance drive using constant V/f where the estimated speed was not fed back to perform closed loop control. Further work could close the control loop and progress to verify the performance by experiment. For a high performance drive, VC has been used. Open loop and closed loop estimators were tested by simulation and the performance of the proposed methods fully discussed. To further verify the performance and robustness of the proposed method and conform to real practice, further work involving experimental testing would again be a natural next step. The online tuning and the optimisation process will take place on the motor drive itself and take into account any nonlinearity.

For sensorless VC of an IM knowledge of the rotor flux for the transformation of the states and the speed is required. Although the estimator may be capable of estimating the rotor flux and speed simultaneously, in this thesis only estimated speed was used for closing the loop. The rotor flux was obtained from the stator current. A further study could be the application to both estimated parameters. For the application of VC, the EKF has been implemented on the stationary reference frame where most of the work on VC has been implemented in the synchronous reference frame. The latter form has been claimed to have better performance [49].

The drawbacks of the EKF include the derivation of the Jacobian matrices which can be complex and cause implementation difficulties with the linearization possibly leading to filter instability if the time step intervals are not sufficiently small. The Unscented KF is claimed to be a superior alternative to the EKF for a variety of estimation and control problems and in addressing these drawbacks, and could be investigated.

In the presently implemented form of the SA algorithm there is not a provision for automatically stopping iterations when a suitable chosen error threshold is reached. This procedure was adopted to fully explore the convergence properties of the SA, but an automatic halt would be straightforward. This simple change may well prevent the occasional anomalous results were the error can rise a little at the very end, and where resistance parameter sensitivity occasionally behaved worse at lower changes than higher.

The major drawback of the whole work in this thesis is the computational burden which is contributed both by the estimator and the proposed method. The work so far used a total of 12 parameters, five parameters each from gain weight **G** and process noise source **Q** and two parameters from measurement noise **R**. Despite concerns, the decoupling of **G** and **Q** appears to have no significant impact. Some authors use gain weight and some do not therefore the gain matrix **G** could be omitted. For generality this work assumed two sources of noise on stator currents. In practice two identical sensors are usually used for measuring stator current, so the sensor uncertainties could be assumed to be the same. Therefore measurement noise **R** could be considered as only one noise source. Using these ideas, the total number of parameters needing to be tuned can be reduced to six, a very significant 2:1 reduction. This reduction in the number of parameter will greatly reduce the computation burden.

Another suggestion is by using a reduced order EKF to replace the full order EKF. The states can be reduced from 5 states with dimension of a 5 x 5 matrix into 3 states which is 3 x 3. Some works on reducing the order of EKF has been proposed in the literature and have shown to reduce the computational burden while keeping the estimator performance largely unchanged. A further idea is to use the simplified SA which has been claimed to perform better and to reduce the computational complexity. Therefore, a combination of reduced order EKF with a simplified SA may have potential for overcome this problem. Increased processing power at reduced cost also offers improvement.

PAPERS PUBLISHED

1. Buyamin, S., Finch, J.W.: 'Tuning extended Kalman filters for induction motor drives using simulated annealing', 26th IASTED International Conference on Modelling, Identification, and Control, 12-14 February 2007

2. Buyamin, S., Finch, J.W.: 'Comparative study on optimisation the EKF for speed estimation of an induction motor using simulated annealing and genetic algorithm', International Electric Machines and Drives Conference IEMDC07, 3-5 May 2007

REFERENCES

1 HOLTZ, J.: 'Sensorless control of IM drives', *Proc. of the IEEE*, 2002, **90,** (8), pp. 1359-1394

2 CONSOLI, A., SCARCELLA, G., and TESTA, A.: 'Speed- and current-sensorless field-oriented IM drive operating at low stator frequencies', *Ind. Applications, IEEE Trans. on*, 2004, **40,** (1), pp. 186-193

3 MUNOZ-GARCIA, A., LIPO, T. A., and NOVOTNY, D. W.: 'A new IM V/f control method capable of high-performance regulation at low speeds', *IEEE Trans. Ind. Appl.*, 1998, **34,** (4), pp. 813-821

4 ANDRZEJ M., T., 'The field Orientation Principle in Control of IMs' (Kluwer, 1994)

5 BIMAL, K. B., 'Modern Power Electronics and AC Drives' (Prentice Hall, 2002)

6 CHIN, T.-H., MIYASHITA, I., and KOGA, T.: 'Sensorless IM drive: an innovative component for advanced motion control', *Control Engineering Practice*, 1997, **5,** (12), pp. 1653-1659

7 OUHROUCHE, M. A.: 'VC of an IM with on-line rotor resistance identification', *1999 IEEE Canadian Conference on Electrical and Computer Engineering 'Engineering Solutions for the Next Millennium'*, 1999, **2,** 1121-1125

8 JACOBINA, C. B., FO, J. B., SALVADORI, F., LIMA, A. M. N., and RIBEIRO, L. A. S.: 'Simple indirect field oriented control of induction machines without speed measurement'. 35th IAS Annual Meeting and World Conference on Industrial Applications of Electrical Energy, Oct 08-Oct 12 2000, 2000, pp. 1809-1813

9 REGINATTO, R., and BAZANELLA, A. S.: 'Robustness of global asymptotic stability in indirect field-oriented control of IMs', *IEEE Trans. Automatic Control*, 2003, **48,** (7), pp. 1218-1222

10 GANJI, A., GUILLAUME, P., PINTELON, R., and LATAIRE, P.: 'IM dynamic and static inductance identification using a broadband excitation technique', *IEEE Trans. Energy Conv.*, 1998, **13,** (1), pp. 15-20

11 LIN, F.-J., WAI, R.-J., and SHIEH, H.-J.: 'Robust control of IM drive with rotor time-constant adaptation', *Electric Power Systems Research*, 1998, **47,** (1), pp. 1-9

12 WANG, J. B., and LIAW, C. M.: 'Control of IM drive for improving operating characteristics and dynamic response', *Mechatronics*, 1997, **7,** (7), pp. 641-661

13 INANC, N.: 'A new sliding mode flux and current observer for direct field oriented IM drives', *Electric Power Systems Research*, 2002, **63,** (2), pp. 113-118

14 AKIN, E., KAYA, M., and KARAKOSE, M.: 'A robust integrator algorithm with genetic based fuzzy controller feedback for direct VC', *Computers & Electrical Engineering*, 2003, **29,** (3), pp. 379-394

15 HUANG, K. S., WU, Q. H., and TURNER, D. R.: 'Effective identification of IM parameters based on fewer measurements', *IEEE Trans. Energy Conv.*, 2002, **17,** (1), pp. 55-60

16 DERDIYOK, A., GUVEN, M. K., REHMAN, H.-U., INANC, N., and XU, L.: 'Design and implementation of a new sliding mode observer for speed-sensorless control of induction machine', *IEEE Trans. Ind. Electron.* 2002, **49,** (5), pp. 1177-1182

17 BARAMBONES, O., and GARRIDO, A. J.: 'A sensorless variable structure control of IM drives', *Electric Power Systems Research*, 2004, **72,** (1), pp. 21-32

18 TURSINI, M., PETRELLA, R., and PARASILITI, F.: 'Adaptive sliding-mode observer for speed-sensorless control of IMs', *IEEE Trans. Ind. Appl.*, 2000, **36,** (5), pp. 1380-1387

19 KIM, S.-M., HAN, W.-Y., and KIM, S.-J.: 'Design of a new adaptive sliding mode observer for sensorless IM drive', *Electric Power Systems Research*, 2004, **70,** (1), pp. 16-22

20 LIN, F.-J., WAI, R.-J., and LIN, P.-C.: 'Robust speed sensorless IM drive', *IEEE Trans. Aerospace and Electronic Systems*, 1999, **35,** (2), pp. 566-578

21 KUBOTA, H., SATO, I., TAMURA, Y., MATSUSE, K., OHTA, H., and HORI, Y.: 'Regenerating-mode low-speed operation of sensorless IM drive with adaptive observer', *IEEE Trans. Ind. Appl.*, 2002, **38,** (4), pp. 1081-1086

22 MATSUSE, K., TANIGUCHI, S., YOSHIZUMI, T., and NAMIKI, K.: 'A speed-sensorless VC of IM operating at high efficiency taking core loss into account', *IEEE Trans. Ind. Appl.* 2001, **37,** (2), pp. 548-558

23 MATSUSE, K., KAWAI, H., KOUNO, Y., and OIKAWA, J.: 'Characteristics of Speed Sensorless VC fed Dual IM Drive Connected in Parallel Fed by a Single Inverter', *IEEE Trans. Ind. Appl.*, 2004, **40,** (1), pp. 153-161

24 YAMADA, T., MATSUSE, K., and SASAGAWA, K.: 'Sensorless control of direct-field-oriented IM operating at high efficiency using adaptive rotor flux observer', Proc. of the 1996 IEEE 22nd International Conference on Industrial Electronics, Control, and Instrumentation, IECON, 1996, **2,** 1149-1154

25 WANG, M., and LEVI, E.: 'Evaluation of steady-state and transient behaviour of a MRAS based sensorless rotor flux oriented induction machine in the presence of parameter detuning', *Electric Machines and Power Systems*, 1999, **27,** (11), pp. 1171-1190

26 WANG, M., LEVI, E., and JOVANOVIC, M.: 'Compensation of parameter variation effects in sensorless indirect vector controlled induction machines using model-based approach', *Electric Machines and Power Systems*, 1999, **27,** (9), pp. 1001-1027

27 VASIC, V., VUKOSAVIC, S. N., and LEVI, E.: 'A stator resistance estimation scheme for speed sensorless rotor flux oriented IM drives', *IEEE Trans. Energy Conversion*, 2003, **18,** (4), pp. 476-483

28 HU JUN B.R., D., and VILATHGAMUWA, M.: 'MRAS-based speed sensorless field oriented control of IM with on-line stator resistance tuning', *Proc. of the IEEE International Conference Power Electronics, Drives & Energy Systems for Industrial Growth, PEDES*, 1998, **1,** 38-43

29 PERNG, S.-S., LIU, C.-H., and LAI, Y.-S.: 'Novel sensorless controller based upon model reference adaptive system for an IM drive', *Journal of the Chinese Institute of Electrical Engineering, Trans. of the Chinese Institute of Engineers, Series E/Chung KuoTien Chi Kung Chieng Hsueh K'an*, 2000, **7,** (4), pp. 249-261

30 RASHED, M., and STRONACH, A. F.: 'A stable back-EMF MRAS-based sensorless low-speed IM drive insensitive to stator resistance variation', *IEE Proc. : Electric Power Appl.*, 2004, **151,** (6), pp. 685-693

31 KIM, S.-H., PARK, T.-S., YOO, J.-Y., and PARK, G.-T.: 'Speed-sensorless VC of an IM using neural network speed estimation', *IEEE Trans. Industrial Electron.* 2001, **48,** (3), pp. 609-614

32 LEE, C.-M., and CHEN, C.-L.: 'Speed sensorless VC of IM using Kalman-filter-assisted adaptive observer', *IEEE Trans. Industrial Electron.* 1998, **45,** (2), pp. 359-361

33 YANG, W.-Q., LI, S.-G., and JIA, Z.-C.: 'Speed sensorless VC of IM based on reduced order EKF', *Shanghai Jiaotong Daxue Xuebao/Journal of Shanghai Jiaotong University*, 2003, **37,** (9), pp. 1362-1365

34 UTKIN, V. I.: 'Sliding mode control design principles and applications to electric drives', *IEEE Trans. Industrial Electron.* 1993, **40,** (1), pp. 23-36

35 PARK, C.-W., and KWON, W.-H.: 'Simple and robust speed sensorless VC of IM using stator current based MRAC', *Electric Power Systems Research*, 2004, **71,** (3), pp. 257-266

36 HUANG, S.-J., HUANG, C.-L., and LIN, Y.-S.: 'Sensorless speed identification of vector--controlled induction drives via neural network based estimation', *Electric Power Systems Research*, 1998, **48,** (1), pp. 1-10

37 AKATSU, K., and KAWAMURA, A.: 'Sensorless very low-speed and zero-speed estimations with online rotor resistance estimation of IM without signal injection', *IEEE Trans. Ind. Appl.*, 2000, **36,** (3), pp. 764-771

38 BARUT, M., BOGOSYAN, S., and GOKASAN, M.: 'Speed sensorless direct torque control of IMs with rotor resistance estimation', *Energy Conversion and Management*, 2005, **46,** (3), pp. 335-349

39 CHEN, S., TSUJI, M., and YAMADA, E.: 'On-line identification of stator resistance for sensorless VC system of IMs', *Zhongguo Dianji Gongcheng Xuebao/Proc. Chinese Society of Electrical Engineering*, 2003, **23,** (2), pp. 88-92

40 GUIDI, G., and UMIDA, H.: 'Novel stator resistance estimation method for speed-sensorless IM drives', *IEEE Trans. Ind. Appl.*, 2000, **36,** (6), pp. 1619-1627

41 LEPPANEN, V.-M., and LUOMI, J.: 'Speed-sensorless induction machine control for zero speed and frequency', *IEEE Trans. Ind. Electron.* 2004, **51,** (5), pp. 1041-1047

42 MATSUSE, K., TANIGUCHI, S., YOSHIZUMI, T., and NAMIKI, K.: 'A speed-sensorless VC of IM operating at high efficiency taking core loss into account', *IEEE Trans. Ind. Appl.*, 2001, **37,** (2), pp. 548-558

43 GUZINSKI, J., ABU-RUB, H., and TOLIYAT, H. A.: 'An advanced low-cost sensorless IM drive', *IEEE Trans. Ind. Appl.*, 2003, **39,** (6), pp. 1757-1764

44 XEPAPAS, S., KALETSANOS, A., XEPAPAS, F., and MANIAS, S.: 'Sliding-mode observer for speed-sensorless IM drives', *IEE Proc. Control Theory and Appl.*, 2003, **150,** (6), pp. 611-617

45 CAMPBELL, J., and SUMNER, M.: 'Practical sensorless IM drive employing an artificial neural network for online parameter adaptation', *IEE Proc. Electric Power Appl.*, 2002, **149,** (4), pp. 255-260

46 YAMADA, T., MATSUSE, K., and SASAGAWA, K.: 'Sensorless control of direct-field-oriented IM operating at high efficiency using adaptive rotor flux observer', *IECON Proc. (Industrial Electronics Conference)*, 1996, **2,** 1149-1154

47 KALMAN, R. E.: 'A new approach to linear filtering and prediction problems', *Journal of Basic Engineering*, 1960, **82,** (D), pp. 35-45

48 B. AKIN, U. O., A. ERSAK,M. EHSANI. 'A Comparative Study on Non-Linear State Estimators Applied to Sensorless AC Drives:MRAS and KF', IEEE Industrial Electronics Society, 2004

49 CHEN, F., and DUNNIGAN, M. W.: 'Comparative study of a sliding-mode observer and KFs for full state estimation in an induction machine', *IEE Proc. Electric Power Appl.*, 2002, **149,** (1), pp. 53-64

50 AKIN, B., ORGUNER, U., and ERSAK, A.: 'State estimation of IM using unscented KF', Proc. of 2003 IEEE Conference Control Applications, 2003, pp. 915-919 vol.2

51 LEITE, A. V., ARAUJO, R. E., and FREITAS, D.: 'Full and reduced order EKF for speed estimation in IM drives: A comparative study', IEEE 35th Annual Power Electronics Specialists Conference, PESC04, Jun 20-25 2004, 2004, pp. 2293-2299

52 FINCH, J. W., ATKINSON, D. J., and ACARNLEY, P. P.: 'Full-order estimator for IM states and parameters', *IEE Proc. Electric Power Appl.*, 1998, **145,** (3), pp. 169-179

53 P. MURACA, C. P.: 'A reduced-order EKF algorithm for parameter and state estimation of an IM: P. Muraca, C. Picardi, pp 225-230', *Control Engineering Practice*, 1993, **1,** (1), pp. 217

54 BEGUENANE, R., OUHROUCHE, M. A., and TRZYNADLOWSKI, A. M.: 'A new scheme for sensorless IM control drives operating in low speed region', *Mathematics and Computers in Simulation*, 2006, **71,** (2), pp. 109-120

55 ATKINSON, D. J., ACARNLEY, P. P., and FINCH, J. W.: 'Observers for IM state and parameter estimation', *IEEE Trans. Ind. Appl.*, 1991, **27,** (6), pp. 1119-1127

56 MORA, J. L., TORRALBA, A., and FRANQUELO, L. G.: 'An adaptive speed estimator for IMs based on a KF with low sample time', 2001 IEEE 32nd Annual Power Electronics Specialists Conference, Jun 17-21 2001, 2001, pp. 794-798

57 ATKINSON, D. J., FINCH, J. W., and ACARNLEY, P. P.: 'Estimation of rotor resistance in IMs', *IEE Proc. Electric Power Appl.*, 1996, **143,** (1), pp. 87-94

58 GARCIA SOTO, G., MENDES, E., and RAZEK, A.: 'Reduced-order observers for rotor flux, rotor resistance and speed estimation for VC fed IM drives using the EKF technique', *IEE Proc. Electric Power Appl.*, 1999, **146,** (3), pp. 282-288

59 SHI, K. L., CHAN, T. F., WONG, Y. K., and HO, S. L.: 'Speed estimation of an IM drive using an optimized EKF', *IEEE Trans. Ind. Electron.* 2002, **49,** (1), pp. 124-133

60 CAI, L., ZHANG, Y., ZHANG, Z., LIU, C., and LU, Z.: 'Application of GAs in EKF for speed estimation of an IM', 2003 IEEE 34th Annual Power Electronics Specialists Conference, Jun 15-19 2003, 2003, pp. 345-349

61 BOLOGNANI, S., TUBIANA, L., and ZIGLIOTTO, M.: 'EKF tuning in sensorless PMSM drives', *IEEE Trans. Ind. Appl.*, 2003, **39,** (6), pp. 1741-1747

62 LOEBIS, D., SUTTON, R., CHUDLEY, J., and NAEEM, W.: 'Adaptive tuning of a KF via fuzzy logic for an intelligent AUV navigation system', *Control Engineering Practice*, 2004, **12,** (12 SPEC ISS), pp. 1531-1539

63 VALAPPIL, J., and GEORGAKIS, C.: 'Systematic estimation of state noise statistics for EKFs', *AIChE Journal*, 2000, **46,** (2), pp. 292-308

64 HAJJI O., B. P.: 'Comparing stochastic optimization methods used in electrical engineering', *IEEE SMC*, 2002,

65 CHEN, S., ISTEPANIAN, R. H., WHIDBORNE, J. F., and WU, J.: 'Adaptive SA for designing finite-precision PID controller structures', Optimisation in Control: Methods and Applications (Ref. No. 1998/521), IEE Colloquium on, 1998, pp. 3/1-3/3

66 ALSADIQ, Y. A., 'Tuning of brushless DC drive speed controllers using stochastic search methods' (University of Newcastle, 2005),'PhD thesis'

67 L.INGBER. 'SA : "Practice versus theory', *Mathematical and Computer Modelling*, 1993, **18,** (11), pp. 29-57

68 L. INGBER, B. R.: 'GAs and very fast simulated reannealing: a comparison', *Mathematical and Computer Modelling*, 1992, **16,** (11), pp. 87-100

69 JWO, W.-S., LIU, C.-W., LIU, C.-C., and HSIAO, Y.-T.: 'Hybrid expert system and SA approach to optimal reactive power planning', *IEE Proc. Generation, Transmission and Distribution*, 1995, **142,** (4), pp. 381-385

70 CERANIC, B., FRYER, C., and BAINES, R. W.: 'An application of SA to the optimum design of reinforced concrete retaining structures', *Computers and Structures*, 2001, **79,** (17), pp. 1569-1581

71 ARES, F., RENGARAJAN, S. R., VILLANUEVA, E., SKOCHINSKI, E., and MORENO, E.: 'Application of GAs and SA technique in optimising the aperture distributions of antenna array patterns', *Electronics Letters*, 1996, **32,** (3), pp. 148-149

72 MANTAWY, A. H., and NEGM, M. M.: 'A SA-based optimal controller for a three phase IM', Power System Technology, 2002. Proc. PowerCon 2002, International Conference, 2002, pp. 750-755 vol.2

73 ACARNLEY, P., and AL-SADIQ, Y.: 'Tuning PI speed controllers for electric drives using SA', Industrial Electronics, 2002. ISIE 2002. Proc. of the 2002 IEEE International Symposium on, 2002, pp. 1131-1135 vol.4

74 THOMPSON, M., and FIDLER, J. K.: 'Application of the GA and SA to LC filter tuning', *Circuits, Devices and Systems, IEE Proc. [see also IEE Proc. G- Circuits, Devices and Systems]*, 2001, **148,** (4), pp. 177-182

75 BARRAL, D., PERRIN, J.-P., DOMBRE, E., and LIEGEOIS, A.: 'SA combined with a constructive algorithm for optimising assembly workcell layout',

International Journal of Advanced Manufacturing Technology, 2001, **17,** (8), pp. 593-602

76 ORILLE, A. L., SOWILAM, G. M. A., and VALENCIA, J. A.: 'New simulation of symmetrical three phase IM under transformations of park', *Proc. of the 1998 24th International Conference on Computers and Industrial Engineering*, 1999, **37,** (1-2), pp. 359-362

77 VAS, P., 'Sensorless Vector and Direct Torque Control' (Oxford, 1998)

78 MAYBECK, P. S., 'Stochastic Models, Estimation and Control' (Academic Press, 1979)

79 GIAOURIS, D., 'Open Loop Control and Stability of IM Drives, including Wavelets' (University of Newcastle, 2004), 'PhD Thesis'

80 GREWAL, M. S., ANDREWS, A. P., 'Kalman Filtering : theory and practice using MATLAB' (New York: Wiley, 2001)

81 BOZIC, S. M., 'Digital and Kalman Filtering' (Edward Arnold, 1994)

82 CANDY, J. V., 'Signal processing: The modern approach' (McGraw-Hill, 1988)

83 DU, T., VAS, P., and STRONACH, F.: 'Design and application of extended observers for joint state and parameter estimation in high-performance AC drives', *IEE Proc. Electric Power Appl.*, 1995, **142,** (2), pp. 71-78

84 YANG, S. K., and LIU, T. S.: 'State estimation for predictive maintenance using KF', *Reliability Engineering and System Safety*, 1999, **66,** (1), pp. 29-39

85 ATKINSON, D. J., 'The Application of Estimation Theory to IM Control' (Newcastle University, 1991),'PhD Thesis'

86 DHAOUADI, R., MOHAN, N., and NORUM, L.: 'Design and implementation of an EKF for the state estimation of a permanent magnet synchronous motor', *IEEE Trans. Power Electron.* 1991, **6,** (3), pp. 491-497

87 KIRKPATRICK, S., GELATT, C.D., JR., VECCHI,, and M.P.: 'Optimization by SA', *Science*, 1983, **220,** (4598), pp. 671-680

88 PHAM, D. T. A. K., D., 'Intelligent Optimisation Techniques' (Springer-Verlag London Limited, 2000)

89 METROPOLIS, A., ROSENBLUTH, W.,, and ROSENBLUTH, M. N.: 'Equation of state calculations by fast computing machines', *The Journal of Chemical Physics*, 1953, **21,** (6), pp. 1087-1092

90 KWOK, D. P., and SHENG, F.: 'GA and SA for optimal robot arm PID control', Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proc. of the First IEEE Conference on, 1994, pp. 707-713 vol.2

91 GAMAL, A. E., HEMACHANDRA, L., SHPERLING, I., and WEI, V.: 'Using SA to design good codes', *IEEE Trans. Information Theory*, 1987, **33,** (1), pp. 116-123

92 MANTAWY, A. H., ABDEL-MAGID, Y. L., and SELIM, S. Z.: 'A SA algorithm for unit commitment', *IEEE Trans. Power Systems*, 1998, **13,** (1), pp. 197-204

93 L. NOLLE, D. A. Armstrong, A. A. HOPGOOD AND J. A. WARE. 'SA and GAs applied to finishing mill optimisation for hot rolling of wide steel strip', *International Journal of Knowledge-Based Intelligent Engineering Systems*, 2002, **6,** (2), pp. 104-111

94 YAO, X.: 'New SA algorithm', *International Journal of Computer Mathematics*, 1995, **56,** (3-4), pp. 161-168

95 ESPINOSA-PEREZ, G., CHANG, G. W., ORTEGA, R., and MENDES, E.: 'On field-oriented control of IMs: Tuning of the PI gains for performance enhancement', *Proc. 37th IEEE Conference Decision and Control (CDC), Dec 16-Dec 18*, 1998, **1,** 971-976

96 LI, Y., ANG, K. H., and CHONG, G. C. Y.: 'PID control system analysis and design: Problems, remedies, and future directions', *IEEE Control Systems Magazine*, 2006, **26,** (1), pp. 32-41

97 INANC, N.: 'A robust sliding mode flux and speed observer for speed sensorless control of an indirect field oriented IM drives', *Electric Power Systems Research*, 2007, In Press, Corrected Proof.

98 WADE, S., DUNNIGAN, M. W., and WILLIAMS, B. W.: 'Improving the accuracy of the rotor resistance estimate for vector-controlled induction machines', *IEE Proc. Electric Power Appl.*, 1997, **144,** (5), pp. 285-294

99 HOLLAND, J. H., 'Adaptation in natural and artificial systems' (University of Mitchigan Press, 1975)

100 CHIPPERFIELD, A. J., and FLEMING, P. J.: 'The MATLAB GA toolbox', IEE Colloquium applied control techniques using MATLAB, 1995, pp. 10/1-10/4

101 D.T.PHAM, D. K., 'Intelligent optimisation technique' (Springer-Verlag, London Limited, 2000)

102 WRIGHT, A. H.: 'GAs for real parameter optimization', *In Foundations of GA, J.E. Rawlins (Ed.), Morgan Kaufmann*, 1991, 12

103 GEN, M. A. C., R.:'GAs and engineering design', (John Wiley, Editor. 1997)

104 SILVA, D. G., 'Speed control of electric drives in the presence of load disturbances' (University of Newcastle, 1999), Ph D thesis

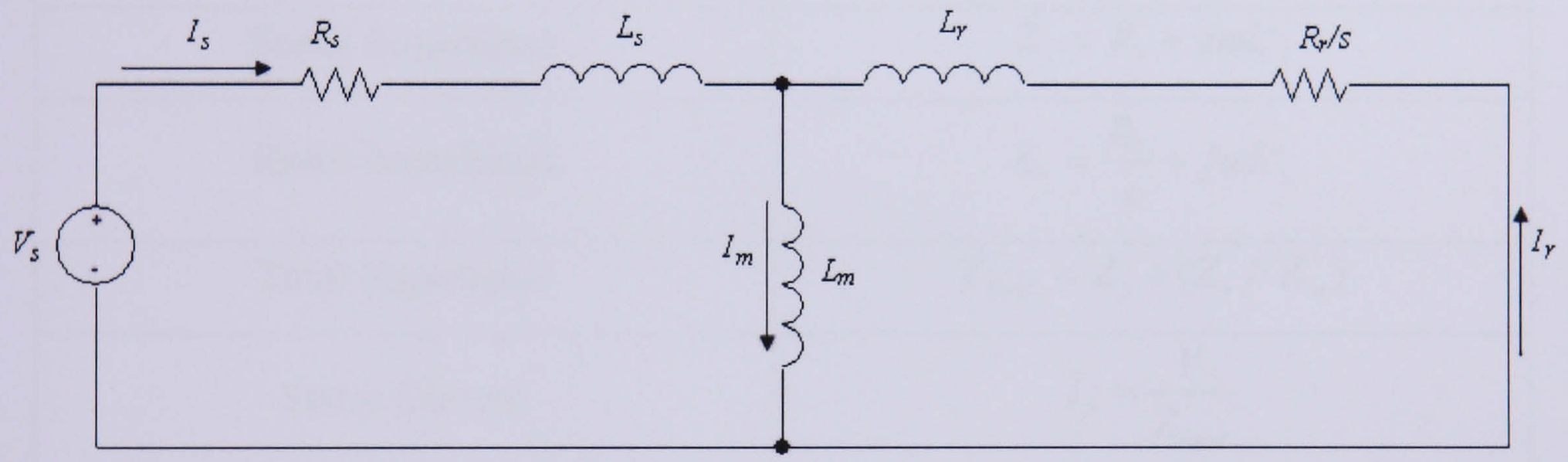
105 A. J. CHIPPERFIELD, P. J. F., 'GA Toolbox for use with MATLAB,' Dept. of Automatic Control and Systems Engineering, University of Sheffield, England, pp 105

APPENDIX A

EQUIVALENT CIRCUIT CALCULATION

A.1 Introduction

The equivalent parameters are most useful for calculating the performances of the IM and its drive system. They can be calculated using the equivalent circuit method which is very simple and convenient for analysis and prediction of the motor performances on a steady state condition. This appendix describes a method of calculating the equivalent parameters based on the equivalent circuit shown in Figure A.1.



Steady State per Phase Equivalent Circuit

A.2 Calculation of Motor Parameter based on a Synchronous and Rated Speed

The equivalent circuit calculation using Microsoft Excel is given. The calculation of the motor parameter and states is presented in real and imaginary coordinates, modulus and angle. The squirrel cage IM characteristics used in this work are given in Chapter 3. The motor is initially assumed to run at synchronous speed, and then running at rated speed. The calculations are shown in Table A.1, while each set is summarised in Table A2 and A3.

One approach to determine the rated speed is by using the slip calculation using the rated power of the machine. Another approach is to use the given rated line current. Since the line current given is the rated value, the rotor speed determined is considered as rated speed. Using the rated line current which is in this case the stator current, the total impedance can be determined and the slip can be resolved and so the rotor speed.

Stator Leakage Inductance	$L'_s = L_s - L_m$
Rotor Leakage Inductance	$L'_r = L_r - L_m$
Stator Electrical Frequency	$\omega = 2\pi f$
Stator Phase Voltage (RMS),	$V = V / \sqrt{3}$
Synchronous Speed,	$\omega_s = 2\pi f \ / \ polepair$
Rotor Speed,	$\omega_r = speed * 2\pi / 60$
Per Unit Slip,	$s = (\omega_s - \omega_r) / \omega_s$
Turns Ratio	1
Magnetizing Impedance	$Z_m = j\omega L_m$
Stator Impedance	$Z_s = R_s + j\omega L'_s$
Rotor Impedance	$Z_r = \frac{R_r}{s} + j\omega L'_r$
Total Impedance	$Z_{Total} = Z_s + (Z_r // Z_m)$
Stator Current	$I_s = \frac{V}{Z_{total}}$
Referred Rotor Current	$I_r = -\frac{(Z_m * I_s)}{(Z_m + Z_r)}$
Magnetizing Current	$I_m = \frac{(Z_r * I_s)}{(Z_m + Z_r)}$
Airgap Voltage	$V_{ai\,rgap} = \frac{Vphase * (Z_m * Z_r)}{(Z_m + Z_r)}$
Airgap Flux Linkage	$\psi_{airgap} = L_m * I_m$
Airgap Power	$P_{airgap} = I_r^{\,2} * \frac{R_r}{s}$
Electrical Torque	$T_e = \frac{3 * P_{airgap}}{\omega_s}$
Rotor Flux Linkage	$\psi_r = (L_m * I_m) + (I_r * L'_r)$

Table A.1: Formula used in Equivalent Circuit Calculation

	D-axis	Q-axis	Mag (RMS)	mag (peak)	rad	angle (°)
Stator Current (A)	0.093	-5.975	5.976	8.451	-1.555	-89.150
Referred Rotor Current (A)	-0.001	0.000	-0.001	-0.001	0.016	0.890
Magnetising Current (A)	0.093	-5.975	5.976	8.451	-1.555	-89.156
Stator Voltage (V RMS)	230.940	0.000	230.940	326.598	0.000	0.000
Air-Gap Voltage (V RMS)	225.253	3.497	225.280	318.594	0.016	0.890
Air Gap Flux Linkage (Wb)	0.011	-0.717	0.717	1.014	-1.555	-89.156
RotorFlux (wb)	0.011	-0.717	0.717	1.014	-1.555	-89.155

Torque (Nm)	0.002
Rotor Speed (RPM)	1500.000

Table A.2: The solution of the steady state, per phase equivalent circuit using synchronous speed

	D-axis	Q-axis	Mag (RMS)	mag (peak)	rad	angle (°)
Stator Current (A)	11.573	-7.610	13.850	19.588	-0.582	-33.345
Referred Rotor Current (A)	-11.741	1.858	-11.887	-16.811	-0.157	-8.999
Magnetising Current (A)	-0.168	-5.751	5.754	8.137	-1.600	-91.722
Stator Voltage (V RMS)	230.940	0.000	230.940	326.598	0.000	0.000
Air-Gap Voltage (V RMS)	216.824	-6.341	216.917	306.767	-0.029	-1.676
Air Gap Flux Linkage (Wb)	-0.020	-0.690	0.690	0.976	-1.600	-91.722
RotorFlux (wb)	-0.107	-0.676	0.685	0.969	-1.728	-99.043

Torque (Nm)	48.844
Rotor Speed (RPM)	1466.851

Table A.3: The Solutions of steady state, per phase equivalent circuit using rated speed

APPENDIX B

DYNAMIC MODELLING OF INDUCTION MOTOR

In a balanced three-phase system, the stator phasor quantities can be written as:

$$\bar{I}_s = \frac{2}{3}(i_{sa} + a i_{sb} + a^2 i_{sc}) \tag{B.1}$$

$$\bar{u}_s = \frac{2}{3}(u_{sa} + a u_{sb} + a^2 u_{sc}) \tag{B.2}$$

$$\bar{\psi}_s = \frac{2}{3}(\psi_{sa} + a \psi_{sb} + a^2 \psi_{sc}) \tag{B.3}$$

Each of the three phases is spaced 120 degrees apart. Manipulation of the stator phasors quantities and employing the Euler's identity:

$$e^{j\omega t} = \cos(\omega t) + j \sin(\omega t) \tag{B.4}$$

Therefore;

$$a = e^{j120} = -\frac{1}{2} + j\frac{\sqrt{3}}{2} \tag{B.5}$$

$$a^2 = e^{j240} = -\frac{1}{2} - j\frac{\sqrt{3}}{2} \tag{B.6}$$

and the expressions for the two axes stator voltage in terms of the three phase voltage are:

$$u_{sD} = \frac{2}{3}(u_a - \frac{1}{2}u_b - \frac{1}{2}u_c) \tag{B.7}$$

$$u_{sQ} = \frac{2}{3}(\frac{\sqrt{3}}{2}u_b - \frac{\sqrt{3}}{2}u_c) \tag{B.8}$$

By utilising the Faraday's Law and Kirchoff's Voltage Law,

$$u = Ri + L\frac{di}{dt} \tag{B.9}$$

$$u = Ri + \frac{dLi}{dt} \tag{B.10}$$

$$u = Ri + \frac{d\Psi}{dt} \tag{B.11}$$

Therefore, the stator and rotor voltage equations in a three single-phase motor are given as follows:

$$u_{sa} = R_s i_{sa} + \frac{d\psi_{sa}}{dt} \tag{B.12}$$

$$u_{sb} = R_s i_{sb} + \frac{d\psi_{sb}}{dt} \tag{B.13}$$

$$u_{sc} = R_s i_{sc} + \frac{d\psi_{sc}}{dt} \tag{B.14}$$

Utilising Eqn (B.1)-(B.3) and Eqn (B.12)-(B.14), the space vector stator voltage differential equation can be derived as follows:

$$\bar{u}_s = \frac{2}{3}(u_{sa} + a u_{sb} + a^2 u_{sc})$$

$$\bar{u}_s = \frac{2}{3}((R_s i_{sa} + \frac{d\psi_{sa}}{dt}) + a(R_s i_{sb} + \frac{d\psi_{sb}}{dt}) + a^2(R_s i_{sc} + \frac{d\psi_{sc}}{dt}))$$

$$\bar{u}_s = \frac{2}{3}((R_s(i_{sa} + a i_{sb} + a^2 i_{sc}) + (\frac{d\psi_{sa}}{dt} + a\frac{d\psi_{sb}}{dt} + a^2\frac{d\psi_{sc}}{dt}))$$

$$\bar{u}_s = R_s \bar{I}_s + \frac{d\bar{\psi}_s}{dt} \tag{B.15}$$

An expression for rotor voltage vectors can be obtained by similar process and yield:

$$\bar{I}_r = \frac{2}{3}(i_{ra} + a i_{rb} + a^2 i_{rc}) \tag{B.16}$$

$$\bar{u}_r = \frac{2}{3}(u_{ra} + a u_{rb} + a^2 u_{rc}) \tag{B.17}$$

$$\bar{\psi}_r = \frac{2}{3}(\psi_{ra} + a\psi_{rb} + a^2\psi_{rc}) \tag{B.18}$$

Where the rotor voltage is given as:

$$u_{ra} = R_r i_{ra} + \frac{d\psi_{ra}}{dt} \tag{B.19}$$

$$u_{rb} = R_r i_{rb} + \frac{d\psi_{rb}}{dt} \tag{B.20}$$

$$u_{rc} = R_r i_{rc} + \frac{d\psi_{rc}}{dt} \tag{B.21}$$

By utilising Eqn (B.16)-(B.18) and (B.19)-(B.21), yields the space vector rotor voltage differential equation:

$$\bar{u}_r = \frac{2}{3}(u_{ra} + a u_{rb} + a^2 u_{rc})$$

$$\bar{u}_r = \frac{2}{3}((R_r i_{ra} + \frac{d\psi_{ra}}{dt}) + a(R_r i_{rb} + \frac{d\psi_{rb}}{dt}) + a^2(R_r i_{rc} + \frac{d\psi_{rc}}{dt}))$$

$$\bar{u}_r = \frac{2}{3}((R_r(i_{ra} + a i_{rb} + a^2 i_{rc}) + (\frac{d\psi_{ra}}{dt} + a\frac{d\psi_{rb}}{dt} + a^2\frac{d\psi_{rc}}{dt}))$$

$$\bar{u}_r = R_r \bar{I}_r + \frac{d\bar{\psi}_r}{dt} \qquad \text{(B.22)}$$

Eqn (B.22) describes the rotor electrical dynamics in a reference frame attached to the rotor. To write the rotor voltage in terms of stator reference frame it is necessary to transform the quantities. The transformation of the stator and rotor quantities can be seen in Figure 4.1. Referring to Figure 4.1 (a), the space phasors of stator currents, voltages and flux linkages in the stationary reference frame fixed to the stator can be written as:

$$\bar{I}_s = |\bar{i}_s| e^{j\alpha_s} \qquad \text{(B.23)}$$

$$\bar{u}_s = |\bar{u}_s| e^{j\alpha_s} \qquad \text{(B.24)}$$

$$\bar{\psi}_s = |\bar{\psi}_s| e^{j\alpha_s} \qquad \text{(B.25)}$$

While the space phasors of rotor currents, voltages and flux linkages in the reference frame fixed to the rotor is shown in Figure 4.1 (b) and is given as follows:

$$\bar{I}_r = |\bar{i}_r| e^{j\alpha_r} \qquad \text{(B.26)}$$

$$\bar{u}_r = |\bar{u}_r| e^{j\alpha_r} \qquad \text{(B.27)}$$

$$\bar{\psi}_r = |\bar{\psi}_r| e^{j\alpha_r} \qquad \text{(B.28)}$$

These stator and rotor phasor quantities can be represented in their real and imaginary axes model as:

Current phasors:

$$\bar{I}_s = i_{sD} + j i_{sQ} \qquad \text{(B.29)}$$

$$\bar{I}_r = i_{r\alpha} + j i_{r\beta} \qquad \text{(B.30)}$$

Voltage phasors:

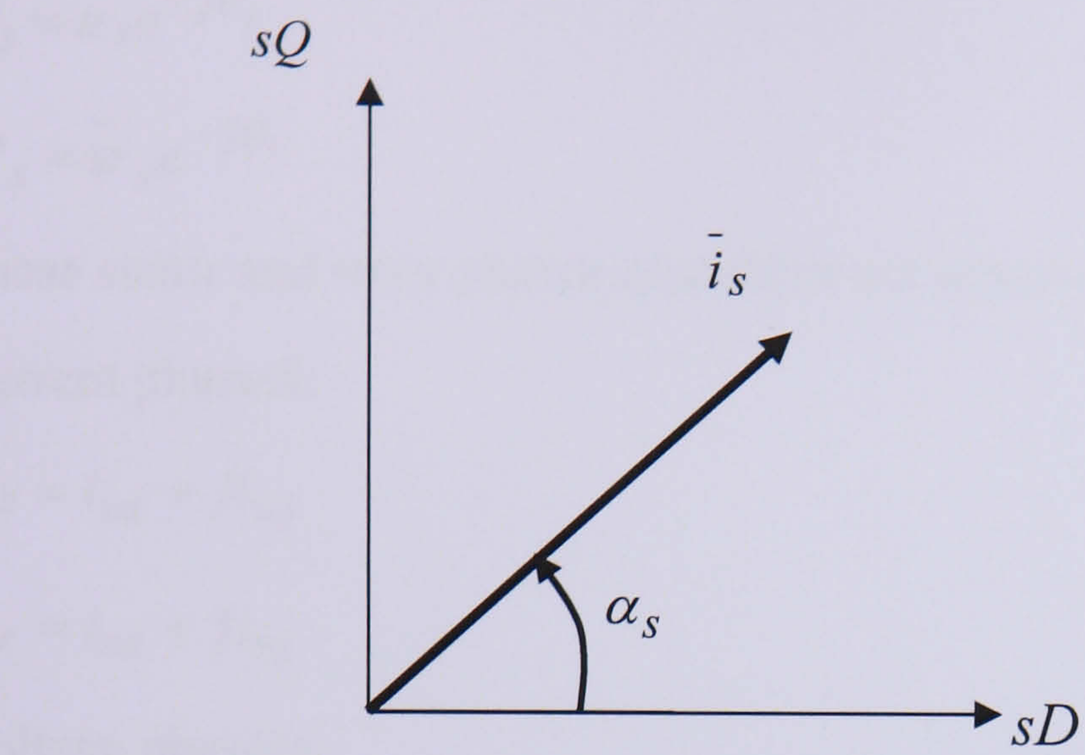
$$\bar{u}_s = u_{sD} + j u_{sQ} \qquad \text{(B.31)}$$

$$\bar{u}_r = u_{r\alpha} + j u_{r\beta} \qquad \text{(B.32)}$$

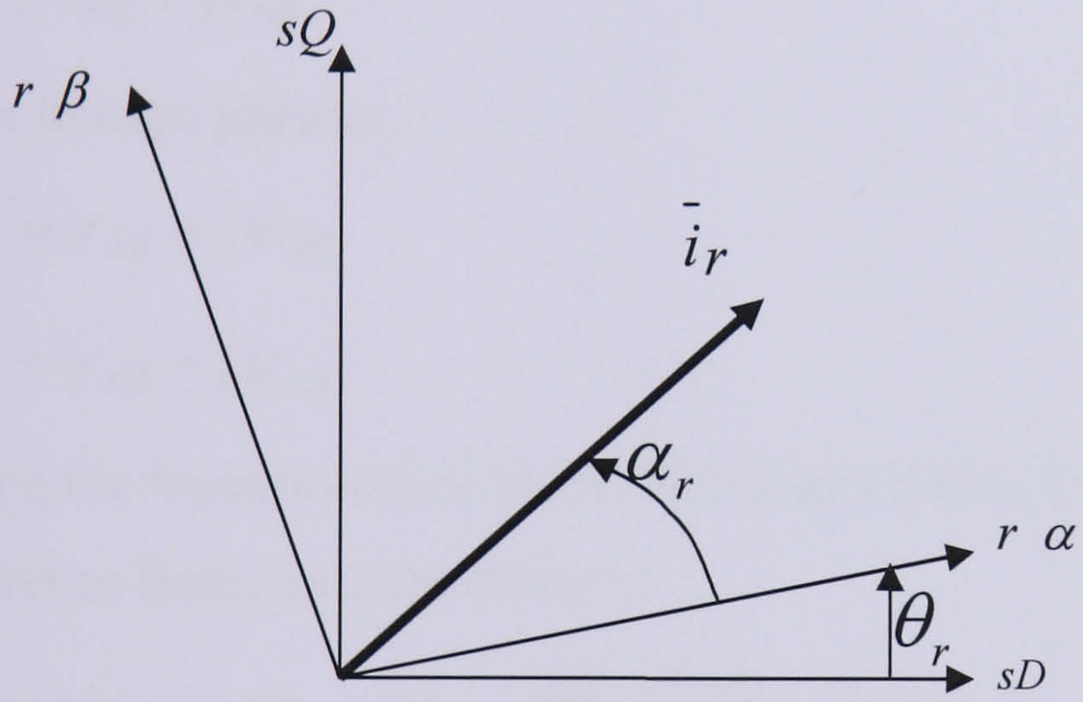
Flux linkage phasors:

$$\bar{\psi}_s = \psi_{sD} + j\psi_{sQ} \tag{B.33}$$

$$\bar{\psi}_r = \psi_{r\alpha} + j\psi_{r\beta} \tag{B.34}$$



(a)



(b)

Figure 4.1: Transformation of the (a) stator quantities (b) rotor quantities [77]

Using Figure 4.1 (a) and (b) , the transformation of the space phasors of rotor and stator currents, voltages and flux linkages in the reference frame fixed to the stator and the rotor can be written as follows:

Reference frame fixed to the stator:

$$\bar{I}'_r = \bar{i}_r e^{j\theta_r} = \left|\bar{i}_r\right| e^{j(\alpha_r + \theta_r)} \tag{B.35}$$

$$\bar{u}'_r = \bar{u}_r e^{j\theta_r} \tag{B.36}$$

$$\bar{\psi}'_r = \bar{\psi}_r e^{j\theta_r} \tag{B.37}$$

Reference frame fixed to the rotor:

$$\bar{I}'_s = \bar{i}_s e^{-j\theta_r} = \left|\bar{i}_s\right| e^{j(\alpha_s - \theta_r)} \tag{B.38}$$

$$\bar{u}'_s = \bar{u}_s e^{-j\theta_r} \tag{B.39}$$

$$\bar{\psi}'_s = \bar{\psi}_s e^{-j\theta_r} \tag{B.40}$$

These stator and rotor phasor quantities are represented using d, q axes model.

Current phasors:

$$\bar{I}'_s = i_{sd} + j i_{sq} \tag{B.41}$$

$$\bar{I}'_r = i_{rd} + j i_{rq} \tag{B.42}$$

Voltage phasors:

$$\bar{u}'_s = u_{sd} + j u_{sq} \tag{B.43}$$

$$\bar{u}'_r = u_{rd} + j u_{rq} \tag{B.44}$$

Flux linkage phasors:

$$\bar{\psi}'_s = \psi_{sd} + j\psi_{sq} \tag{B.45}$$

$$\bar{\psi}'_r = \psi_{rd} + j\psi_{rq} \tag{B.46}$$

Using the transformation theory discussed above, Eqn (B.22) can be written in stator reference frame as shown below:

$$\bar{u}'_r e^{-j\theta_r} = R_r \bar{I}'_r e^{-j\theta_r} + \frac{d\bar{\psi}'_r e^{-j\theta_r}}{dt}$$

$$\bar{u}'_r e^{-j\theta_r} = R_r \bar{I}'_r e^{-j\theta_r} + e^{-j\theta_r} \frac{d\bar{\psi}'_r}{dt} + \bar{\psi}'_r \frac{de^{-j\theta_r}}{dt}$$

$$\bar{u}'_r e^{-j\theta_r} = R_r \bar{I}'_r e^{-j\theta_r} + e^{-j\theta_r} \frac{d\bar{\psi}'_r}{dt} - j e^{-j\theta_r} \bar{\psi}'_r \frac{d\theta_r}{dt}$$

Eliminate the term $e^{-j\theta_r}$, and let $\frac{d\theta_r}{dt} = \omega_r$ which is the angular rotor speed

$$\bar{u}'_r = R_r \bar{I}'_r + \frac{d\bar{\psi}'_r}{dt} - j\omega_r \bar{\psi}'_r$$

Using differential operator $p = \frac{d}{dt}$ and rearranging the above equations results in

$$\bar{u}'_r = R_r \bar{I}'_r + \bar{\psi}'_r (p - j\omega_r) \tag{B.47}$$

The flux linkage vectors can be written in matrix form as:

$$\begin{bmatrix} \bar{\psi}_s \\ \bar{\psi}'_r \end{bmatrix} = \begin{bmatrix} L_s & L_m \\ L_m & L_r \end{bmatrix} \begin{bmatrix} \bar{I}_s \\ \bar{I}'_r \end{bmatrix} \tag{B.48}$$

$$L_s = L_m + L_{ls}$$

$$L_r = L_m + L_{lr}$$

Where L_m is the mutual inductance and L_s, L_r are the self inductance of the stator and rotor. L_{ls}, L_{lr} are the stator and rotor leakage inductance respectively.

Substituting flux linkage vectors in (B.15) and with (B.48) result in the following equations:

Stator voltage vectors:

$$\bar{u}_s = R_s \bar{I}_s + p(L_s \bar{I}_s + L_m \bar{I}'_r)$$

$$\bar{u}_s = (R_s + pL_s)\bar{I}_s + pL_m \bar{I}'_r \tag{B.49}$$

Rotor voltage vectors:

$$\bar{u}'_r = R_r \bar{I}'_r + (L_r \bar{I}'_r + L_m \bar{I}_s)(p - j\omega_r)$$

$$\bar{u}'_r = R_r \bar{I}'_r + p(L_r \bar{I}'_r + L_m \bar{I}_s) - j\omega_r (L_r \bar{I}'_r + L_m \bar{I}_s)$$

$$\bar{u}'_r = (R_r + (p - j\omega_r)L_r)\bar{I}'_r + (p - j\omega_r)L_m \bar{I}_s \tag{B.50}$$

In matrix form:

$$\begin{bmatrix} \bar{u}_s \\ \bar{u}'_r \end{bmatrix} = \begin{bmatrix} R_s + pL_s & pL_m \\ (p - j\omega_r)L_m & R_r + (p - j\omega_r)L_r \end{bmatrix} \begin{bmatrix} \bar{I}_s \\ \bar{I}'_r \end{bmatrix} \tag{B.51}$$

Substituting all the vectors quantities with their associated dq-components resulted in the following equation.

$$\bar{u}_s = (R_s + pL_s)\bar{I}_s + pL_m \bar{I}'_r$$

$$u_{sD} + ju_{sQ} = (R_s + pL_s)(i_{sD} + ji_{sQ}) + pL_m(i_{rd} + ji_{rq})$$

Collecting the direct and quadrature terms, yields

$$u_{sD} = R_s i_{sD} + pL_s i_{sD} + pL_m i_{rd} \tag{B.52}$$

$$u_{sQ} = R_s i_{sQ} + pL_s i_{sQ} + pL_m i_{rq} \tag{B.53}$$

The same procedure can be carried out for the rotor flux linkage equation. From Eqn (B.48), the rotor flux linkage vector can be written as:

$$\bar{\psi}'_r = L_r \bar{I}'_r + L_m \bar{I}_s$$

$$\psi_{rd} + j\psi_{rq} = L_r(i_{rd} + ji_{rq}) + L_m(i_{sD} + i_{sQ})$$

Collecting the real and imaginary parts together yields:

$$\psi_{rd} = L_r i_{rd} + L_m i_{sD} \tag{B.54}$$

$$\psi_{rq} = L_r i_{rq} + L_m i_{sQ} \tag{B.55}$$

Solve for i_{rd} and i_{rq}:

$$i_{rd} = (\psi_{rd} - L_m i_{sD}) / L_r \tag{B.56}$$

$$i_{rq} = (\psi_{rq} - L_m i_{sQ}) / L_r \tag{B.57}$$

Substitute rotor currents in (B.52) and (B.53) with (B.56) and (B.57) respectively:

$$u_{sD} = (R_s + pL_s)i_{sD} + pL_m(\psi_{rd} - L_m i_{sD}) / L_r$$

$$u_{sD} = (R_s + pL_s)i_{sD} + p\frac{L_m}{L_r}\psi_{rd} - p\frac{L_m^2}{L_r}i_{sD} \tag{B.58}$$

$$u_{sQ} = (R_s + pL_s)i_{sQ} + pL_m(\psi_{rq} - L_m i_{sQ}) / L_r$$

$$u_{sQ} = (R_s + pL_s)i_{sQ} + p\frac{L_m}{L_r}\psi_{rq} - p\frac{L_m^2}{L_r}i_{sQ} \tag{B.59}$$

Rearranging (B.58) and (B.59) and solve for rotor flux linkage vector yields:

$$p\psi_{rd} = (u_{sD} - (R_s + pL_s)i_{sD} + p\frac{L_m^2}{L_r}i_{sD})\frac{Lr}{L_m} \tag{B.60}$$

$$p\psi_{rq} = (u_{sQ} - (R_s + pL_s)i_{sQ} + p\frac{L_m^2}{L_r}i_{sQ})\frac{L_r}{L_m} \tag{B.61}$$

Recall the rotor voltage vector:

$$\overline{u'}_r = R_r\overline{I'}_r + \overline{\psi'}_r(p - j\omega_r)$$

Substitute voltage, rotor and stator currents with its dq-components

$$u_{rd} + ju_{rq} = R_r(i_{rd} + ji_{rq}) + (p - j\omega_r)(\psi_{rd} + j\psi_{rq})$$

Collect its dq components together, yields:

$$u_{rd} = R_r i_{rd} + p\psi_{rd} + \omega_r\psi_{rq} \tag{B.62}$$

$$u_{rq} = R_r i_{rq} + p\psi_{rq} - \omega_r\psi_{rd} \tag{B.63}$$

Substitute rotor currents in (B.62) with (B.56)

$$u_{rd} = R_r\frac{(\psi_{rd} - L_m i_{sD})}{L_r} + p\psi_{rd} + \omega_r\psi_{rq}$$

$$u_{rd} = \frac{R_r}{L_r}\psi_{rd} - \frac{R_r}{L_r}L_m i_{sD} + p\psi_{rd} + \omega_r \psi_{rq}$$

Rearranging and solving for d-axis rotor flux linkage yields:

$$p\psi_{rd} = u_{rd} + \frac{R_r}{L_r}(L_m i_{sD} - \psi_{rd}) - \omega_r \psi_{rq} \tag{B.64}$$

Similar substitution for rotor current in (B.63) with (B.57) Rearrange and solve for q-axis rotor flux linkage yields:

$$p\psi_{rq} = u_{rq} + \frac{R_r}{L_r}(L_m i_{sQ} - \psi_{rq}) + \omega_r \psi_{rd} \tag{B.65}$$

For a squirrel cage machine, the rotor is short circuited therefore the dq voltage components are zero and (B.64) and (B.65) can be simplified as:

$$p\psi_{rd} = \frac{R_r}{L_r}(L_m i_{sD} - \psi_{rd}) - \omega_r \psi_{rq} \tag{B.66}$$

$$p\psi_{rq} = \frac{R_r}{L_r}(L_m i_{sQ} - \psi_{rq}) + \omega_r \psi_{rd} \tag{B.67}$$

$$\text{Let } T_r = \frac{L_r}{R_r}; \tag{B.68}$$

The rotor speed equation is:

$$\omega_r = (-\frac{d(\psi_{rd})}{dt} - \frac{\psi_{rd}}{T_r} + \frac{L_m}{T_r}) / \psi_{rq} \tag{B.69}$$

The rotor flux linkage space vector can be expressed in terms of the stator flux linkage space vector. Utilizing

$$L'_s = L_s - \frac{L_m{}^2}{L_r} \tag{B.70}$$

Where

$$L'_s = \alpha L_s \text{ and } \alpha = 1 - \frac{L_m{}^2}{L_s L_r}$$

and Eqn (B.48) results in:

$$\overline{\psi_r} = \frac{L_r}{L_m}(\overline{\psi}_s - L_s \overline{i_s}) + L_m \overline{i_s}$$

$$\overline{\psi_r} = \frac{L_r}{L_m}\overline{\psi}_s - \frac{L_r}{L_m}L_s \overline{i_s} + \frac{L_r}{L_r}\frac{L_m{}^2}{L_m}\overline{i_s}$$

$$\overline{\psi_r} = \frac{L_r}{L_m}\overline{\psi}_s - \frac{L_r}{L_m}\overline{i_s} + (L_s - \frac{{L_m}^2}{L_r})$$

Yield to:

$$\overline{\psi_r} = \frac{L_r}{L_m}(\overline{\psi}_s - L'_s\,\overline{i_s}) \tag{B.71}$$

The derivative of Eqn (B.71) gives the real and imaginary parts of $\overline{\psi}$ as:

$$\frac{d\psi_{rd}}{dt} = \frac{L_r}{L_m}(\frac{d\psi_{sD}}{dt} - L'_s\,\frac{di_{sD}}{dt}) \tag{B.72}$$

$$\frac{d\psi_{rq}}{dt} = \frac{L_r}{L_m}(\frac{d\psi_{sQ}}{dt} - L'_s\,\frac{di_{sQ}}{dt}) \tag{B.73}$$

Eqn (B.72) and (B.73) contain stator flux linkage which can be obtained from Eqn (B.11) by using:

$$\frac{d\psi_{sD}}{dt} = u_{sD} - R_s i_{sQ} \tag{B.74}$$

$$\frac{d\psi_{sQ}}{dt} = u_{sQ} - R_s i_{sQ} \tag{B.75}$$

Substituting Eqn (B.74) and (B.75) into Eqn (B.72) and (B.73) gives:

$$\frac{d\psi_{rd}}{dt} = \frac{L_r}{L_m}(u_{sD} - R_s i_{sD} - L'_s\,\frac{di_{sD}}{dt}) \tag{B.76}$$

$$\frac{d\psi_{rq}}{dt} = \frac{L_r}{L_m}(u_{sQ} - R_s i_{sQ} - L'_s\,\frac{di_{sQ}}{dt}) \tag{B.77}$$

Rearranging Eqn (B.58) and (B.59) and substituting Eqn (B.66) and (B.67):

$$u_{sD} = (R_s + pL_s)i_{sD} + p\frac{L_m}{L_r}\psi_{rd} - p\frac{{L_m}^2}{L_r}i_{sD}$$

$$u_{sD} = R_s i_{sD} + (L_s - \frac{{L_m}^2}{L_r})pi_{sD} + \frac{L_m}{L_r}p\psi_{rd}$$

$$u_{sD} = R_s i_{sD} + (L_s - \frac{{L_m}^2}{L_r})pi_{sD} + \frac{L_m}{L_r}(\frac{R_r}{L_r}(L_m i_{sD} - \psi_{rd}) - \omega_r\psi_{rq})$$

$$pi_{sD} = [u_{sD} - (R_s + R_r\frac{{L_m}^2}{{L_r}^2})i_{sD} + \frac{L_m}{{L_r}^2}R_r\psi_{rd} + \frac{L_m}{L_r}\omega_r\psi_{rq}]/(L_s - \frac{{L_m}^2}{L_r}) \tag{B.78}$$

for pi_{sQ},

$$u_{sQ} = (R_s + pL_s)i_{sQ} + p\frac{L_m}{L_r}\psi_{rq} - p\frac{{L_m}^2}{L_r}i_{sQ}$$

$$u_{sQ} = (R_s + pL_s)i_{sQ} + \frac{L_m}{L_r}(\frac{R_r}{L_r}(L_m i_{sQ} - \psi_{rq}) + \omega_r\psi_{rd}) - p\frac{{L_m}^2}{L_r}i_{sQ}$$

$$u_{sQ} = (R_s + \frac{{L_m}^2}{{L_r}^2}R_r)i_{sQ} - \frac{L_m}{{L_r}^2}R_r\psi_{rq} + \frac{L_m}{L_r}\omega_r\psi_{rd}) + (L_s - \frac{{L_m}^2}{L_r})pi_{sQ}$$

$$pi_{sQ} = [u_{sQ} - (R_s + \frac{{L_m}^2}{{L_r}^2}R_r)i_{sQ} + \frac{L_m}{{L_r}^2}R_r\psi_{rq} - \frac{L_m}{L_r}\omega_r\psi_{rd}]/(L_s - \frac{{L_m}^2}{L_r}) \quad \text{(B.79)}$$

To summarise:

$$pi_{sD} = [u_{sD} - (R_s + R_r\frac{{L_m}^2}{{L_r}^2})i_{sD} + \frac{L_m}{{L_r}^2}R_r\psi_{rd} + \frac{L_m}{L_r}\omega_r\psi_{rq}]/(L_s - \frac{{L_m}^2}{L_r}) \quad \text{(B.80)}$$

$$pi_{sQ} = [u_{sQ} - (R_s + \frac{{L_m}^2}{{L_r}^2}R_r)i_{sQ} + \frac{L_m}{{L_r}^2}R_r\psi_{rq} - \frac{L_m}{L_r}\omega_r\psi_{rd}]/(L_s - \frac{{L_m}^2}{L_r}) \quad \text{(B.81)}$$

$$p\psi_{rd} = \frac{R_r}{L_r}(L_m i_{sD} - \psi_{rd}) - \omega_r\psi_{rq} \quad \text{(B.82)}$$

$$p\psi_{rq} = \frac{R_r}{L_r}(L_m i_{sQ} - \psi_{rq}) + \omega_r\psi_{rd} \quad \text{(B.83)}$$

APPENDIX C

KALMAN FILTER ALGORITHM

C.1 Kalman Filtering Derivation

One of the well known methods often used for stochastic estimation is the KF. It is based on mathematical equations which implement a predictor and corrector type estimator. It is an optimal estimator in the sense of minimizing the estimated error covariance. Based on the original formulation of KF [47] and with the help of [79, 80], the discrete-time recursive equations of KF has been derived in this thesis. However, some slight changes in the notation have been used for easy understanding.

A stochastic system in Figure C.1 can be described in discrete time as:

$$\mathbf{X}_{k+1} = \mathbf{\Phi}_k \mathbf{X}_k + \mathbf{\Gamma}_k \mathbf{U}_k + \mathbf{W}_k \tag{C.1}$$

$$\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{V}_k \tag{C.2}$$

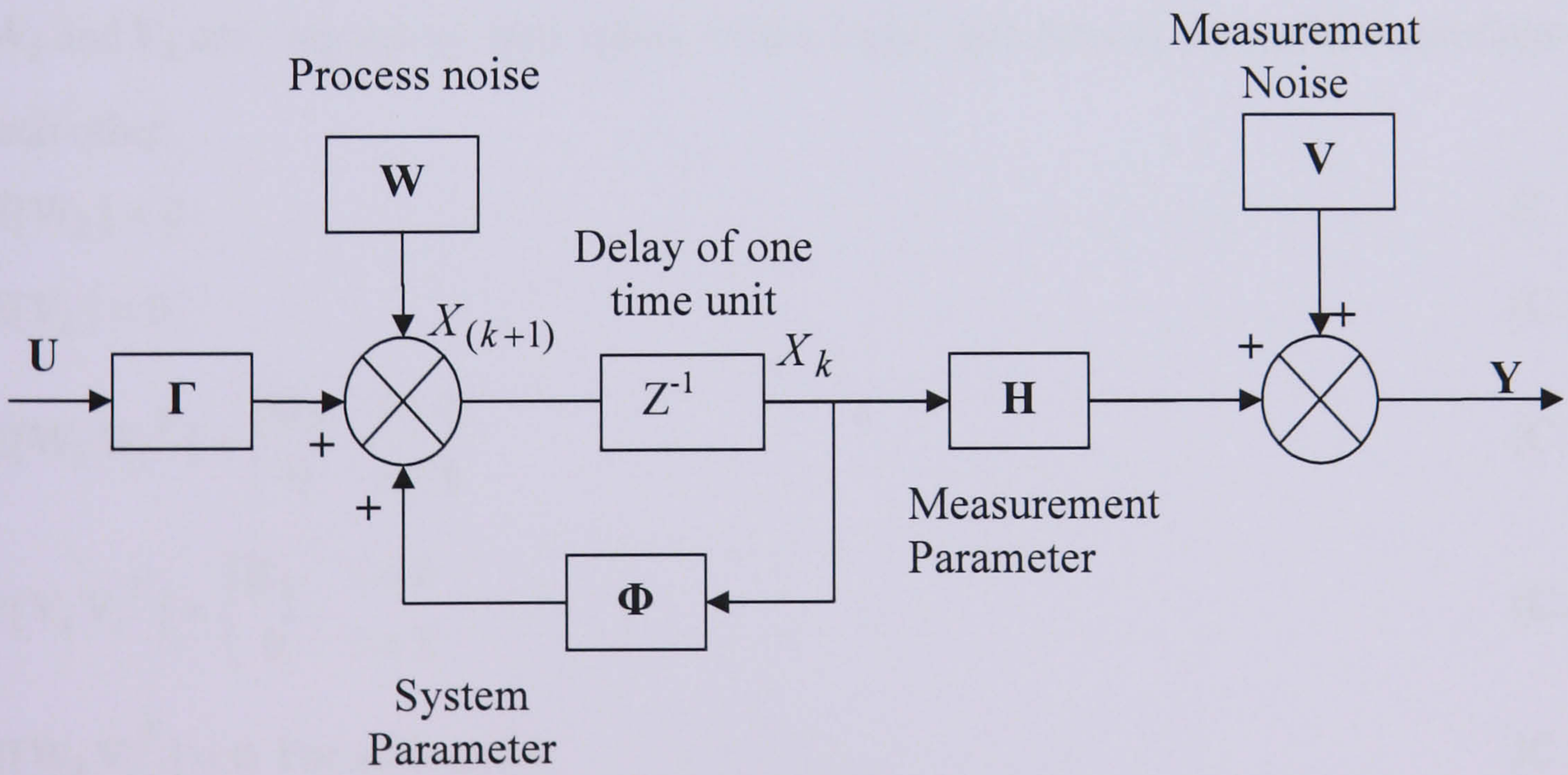


Figure C.1: Stochastic State Space model in discrete time system

The KF addressed the problem of trying to estimate the state of the controlled process Eqn (C.1) with the measurement Eqn (C.2). It estimates the process state at certain time and then obtains feedback in the form of noisy measurement. This is shown in Figure C.2.

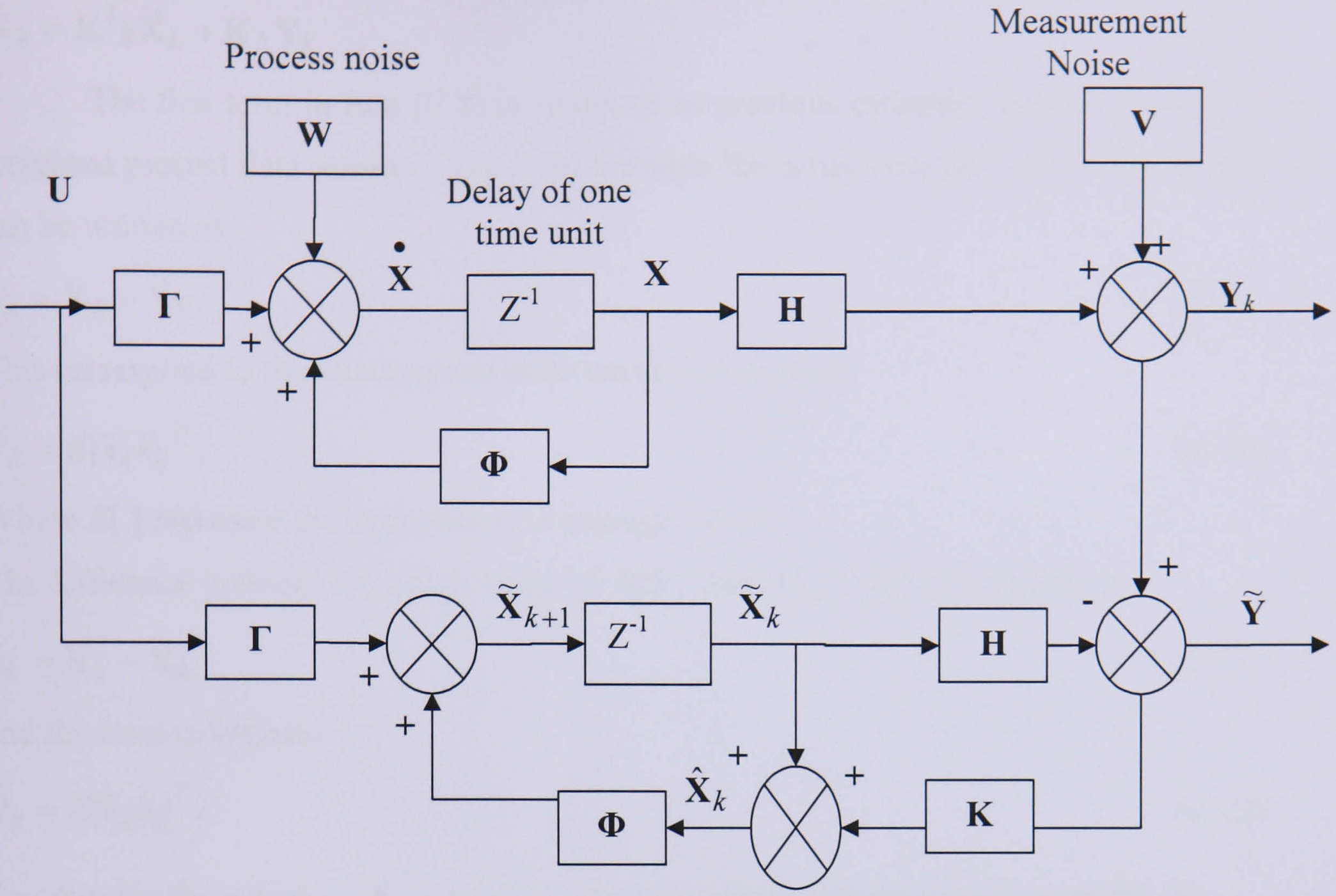


Figure C.2: KF observer

In the following derivation, several assumptions are required. The noisy signals $\mathbf{W}_k$ and $\mathbf{V}_k$ are regard as zero mean, white noise and having zero cross correlation with each other.

$$E[\mathbf{W}_k] = 0 \tag{C.3}$$

$$E[\mathbf{V}_k] = 0 \tag{C.4}$$

$$E[\mathbf{W}_k \mathbf{W}_i^T] = \begin{cases} \mathbf{Q}_k & i = k \\ 0 & i \neq k \end{cases} \tag{C.5}$$

$$E[\mathbf{V}_k \mathbf{V}_i^T] = \begin{cases} \mathbf{R}_k & i = k \\ 0 & i \neq k \end{cases} \tag{C.6}$$

$E[\mathbf{W}_k \mathbf{V}_i^T] = 0$ For all k and i (C.7)

In many cases the estimation is begin with no prior measurement. With the assumption of prior estimation, the measurement output $\mathbf{Y}_k$ is now used to improve the priori estimate. This new estimate is called the posteriori (updated) estimate, $\hat{\mathbf{X}}_k$. The posteriori estimate equation is a linear function of the priori estimate and the measurement $\mathbf{Y}_k$ output and written as:

$$\hat{\mathbf{X}}_k = \mathbf{K}^1{}_k \tilde{\mathbf{X}}_k + \mathbf{K}_k \mathbf{Y}_k \tag{C.8}$$

The first term in Eqn (C.8) is weighted as previous estimate, and the second term is weighted present data sample. The error between the actual process and the priori estimate can be written as

$$\tilde{\mathbf{e}}_k = \mathbf{X}_k - \tilde{\mathbf{X}}_k \tag{C.9}$$

This correspond to the mean square error (error covariance)

$$\tilde{\mathbf{P}}_k = E[\tilde{\mathbf{e}}_k \tilde{\mathbf{e}}_k{}^T] \tag{C.10}$$

Where $E[\ ]$ represent the expectation or average value.

The difference between the actual value of $\mathbf{X}(k)$ and the posteriori estimate is

$$\hat{\mathbf{e}}_k = \mathbf{X}_k - \hat{\mathbf{X}}_k \tag{C.11}$$

and the error covariance is

$$\hat{\mathbf{P}}_k = E[\hat{\mathbf{e}}_k \hat{\mathbf{e}}_k{}^T] \tag{C.12}$$

To optimize the values of $\mathbf{K}^1{}_k$ and $\mathbf{K}_k$, the a posteriori estimates need to satisfy the orthogonality principle which can be written in the form of:

$$E[[\mathbf{X}_k - \hat{\mathbf{X}}_k]\mathbf{Z}_i{}^T] = 0 \text{ Where } i = 1,2,3,\ldots,k-1 \tag{C.13}$$

$$E[[\mathbf{X}_k - \hat{\mathbf{X}}_k]\mathbf{Z}_k{}^T] = 0 \tag{C.14}$$

To solve this optimization problem, $\mathbf{X}_k$, $\mathbf{Y}_k$ and $\hat{\mathbf{X}}_k$ are substituted into Eqn (C.13). By considering the random sequences $\mathbf{W}_k$ and $\mathbf{V}_k$ are uncorrelated, therefore it follows that

$$E\mathbf{W}_k \mathbf{Z}_i{}^T = 0 \text{ and } E\mathbf{V}_k \mathbf{Z}_i{}^T = 0$$

$$E[[\mathbf{\Phi}_{k-1}\mathbf{X}_{k-1} + \mathbf{W}_{k-1} - (\mathbf{K}^1{}_k \tilde{\mathbf{X}}_k + \mathbf{K}_k(\mathbf{H}_k \mathbf{X}_k + \mathbf{V}_k)]\mathbf{Z}_i{}^T] = 0$$

$$E[[\mathbf{\Phi}_{k-1}\mathbf{X}_{k-1} - (\mathbf{K}^1{}_k \tilde{\mathbf{X}}_k + \mathbf{K}_k \mathbf{H}_k \mathbf{X}_k)\mathbf{Z}_i{}^T]] = 0$$

$$E[\mathbf{\Phi}_{k-1}\mathbf{X}_{k-1} - \mathbf{K}^1{}_k \tilde{\mathbf{X}}_k - \mathbf{K}_k \mathbf{H}_k \mathbf{X}_k)\mathbf{Z}_i{}^T] = 0$$

$$E[\mathbf{X}_k - \mathbf{K}^1{}_k \tilde{\mathbf{X}}_k - \mathbf{K}_k \mathbf{H}_k \mathbf{X}_k)\mathbf{Z}_i{}^T] = 0$$

$$E[\mathbf{X}_k(1 - \mathbf{K}_k \mathbf{H}_k) - \mathbf{K}^1{}_k \mathbf{X}_k - \mathbf{K}^1{}_k(\tilde{\mathbf{X}}_k - \mathbf{X}_k)\mathbf{Z}_i{}^T] = 0$$

$$E[\mathbf{X}_k(1 - \mathbf{K}_k \mathbf{H}_k) - \mathbf{K}^1{}_k \mathbf{X}_k + \mathbf{K}^1{}_k(\mathbf{X}_k - \tilde{\mathbf{X}}_k)\mathbf{Z}_i{}^T] = 0$$

$$E[\mathbf{X}_k(1 - \mathbf{K}_k \mathbf{H}_k - \mathbf{K}^1{}_k)\mathbf{Z}_i{}^T] = 0$$

$$\mathbf{X}_k(1-\mathbf{K}_k\mathbf{H}_k-\mathbf{K}^1{}_k)E\mathbf{Z}_i{}^T=0 \tag{C.15}$$

Eqn (C.15) can be satisfied if

$$\mathbf{K}^1{}_k=1-\mathbf{K}_k\mathbf{H}_k \tag{C.16}$$

The choice of $\mathbf{K}_k$ in Eqn (C.16) can satisfy Eqn (C.8).

Applying this to Eqn (C.8), yield to posteriori estimator:

$$\hat{\mathbf{X}}_k=(1-\mathbf{K}_k\mathbf{H}_k)\tilde{\mathbf{X}}_k+\mathbf{K}_k\mathbf{Y}_k \text{ or } \hat{\mathbf{X}}_k=\tilde{\mathbf{X}}_k+\mathbf{K}_k(\mathbf{Y}_k-\mathbf{H}_k\tilde{\mathbf{X}}_k) \tag{C.17}$$

Eqn (C.17) can be written as

$$\hat{\mathbf{X}}_{k+1}=(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\tilde{\mathbf{X}}_{k+1}+\mathbf{K}_{k+1}\mathbf{Y}_{k+1} \tag{C.18}$$

Where $\tilde{\mathbf{X}}_{k+1}=\mathbf{\Phi}_k\hat{\mathbf{X}}_k+\mathbf{\Gamma}_k\mathbf{U}_k$ (C.19)

Substitute Eqn (C.19) into (C.18) for $\tilde{\mathbf{X}}_{k+1}$ and Eqn (C.2) for $\mathbf{Y}_{k+1}$,

$$\hat{\mathbf{X}}_{k+1}=(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})(\mathbf{\Phi}_k\hat{\mathbf{X}}_k+\mathbf{\Gamma}_k\mathbf{U}_k)+\mathbf{K}_{k+1}(\mathbf{H}_{k+1}\mathbf{X}_{k+1}+\mathbf{V}_{k+1})$$

$$\hat{\mathbf{X}}_{k+1}=(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})(\mathbf{\Phi}_k\hat{\mathbf{X}}_k+\mathbf{\Gamma}_k\mathbf{U}_k)+\mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{X}_{k+1}+\mathbf{K}_{k+1}\mathbf{V}_{k+1}$$

$$\hat{\mathbf{X}}_{k+1}=(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})(\mathbf{\Phi}_k\hat{\mathbf{X}}_k+\mathbf{\Gamma}_k\mathbf{U}_k)+\mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{X}_{k+1}+\mathbf{K}_{k+1}\mathbf{V}_{k+1}$$

In this derivation, the notation of priori estimate

$$\tilde{\mathbf{X}}_k=\hat{\mathbf{X}}_{k/k-1} \tag{C.20}$$

can be read as "the estimate of *x* at time *k* based on the information from time *k-1*", or in other words as the estimate based upon the past values of the output. The notation of a posteriori estimate

$$\hat{\mathbf{X}}_k=\hat{\mathbf{X}}_{k/k} \tag{C.21}$$

can be read as "the estimate of *x* at time *k* based on the information from time *k*", or in other words as the estimate based upon the past and current values of the output.

The latter equation can be written as

$$\mathbf{X}_{k+1/k+1}=(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})(\mathbf{\Phi}_k\mathbf{X}_{k/k}+\mathbf{\Gamma}_k\mathbf{U}_k)+\mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{X}_{k+1}+\mathbf{K}_{k+1}\mathbf{V}_{k+1} \tag{C.22}$$

Recall the errors of priori and posteriori estimate with their error covariance:

$$\tilde{\mathbf{e}}_k=\mathbf{X}_k-\tilde{\mathbf{X}}_k$$

$$\hat{\mathbf{e}}_k=\mathbf{X}_k-\hat{\mathbf{X}}_k$$

and using either (C.20) or (C.21), the errors can be written as

$$\bar{\mathbf{e}}_{k+1/k}=\mathbf{X}_{k+1}-\bar{\mathbf{X}}_{k+1/k} \tag{C.23}$$

$$\bar{\mathbf{e}}_{k+1/k+1} = \mathbf{X}_{k+1} - \hat{\mathbf{X}}_{k+1/k+1} \tag{C.24}$$

Substitute Eqn (C.1) and (C.22) into Eqn (C.24)

$$\bar{\mathbf{e}}_{k+1/k+1} = \mathbf{\Phi}_k \mathbf{X}_k + \mathbf{\Gamma}_k \mathbf{U}_k + \mathbf{W}_k - (1 - \mathbf{K}_{k+1}\mathbf{H}_{k+1})(\mathbf{\Phi}_k \hat{\mathbf{X}}_{k/k} + \mathbf{\Gamma}_k \mathbf{U}_k)$$

$$- \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{X}_{k+1} - \mathbf{K}_{k+1}\mathbf{V}_{k+1}$$

To eliminate the term $\mathbf{X}_{k+1}$, Eqn (C.1) is used.

$$\bar{\mathbf{e}}_{k+1/k+1} = \mathbf{\Phi}_k \mathbf{X}_k + \mathbf{\Gamma}_k \mathbf{U}_k + \mathbf{W}_k - (1 - \mathbf{K}_{k+1}\mathbf{H}_{k+1})(\mathbf{\Phi}_k \hat{\mathbf{X}}_{k/k} + \mathbf{\Gamma}_k \mathbf{U}_k)$$

$$- \mathbf{K}_{k+1}\mathbf{H}_{k+1}(\mathbf{\Phi}_k \mathbf{X}_k + \mathbf{\Gamma}_k \mathbf{U}_k + \mathbf{W}_k) - \mathbf{K}_{k+1}\mathbf{V}_{k+1}$$

After multiplication and self cancelled of $\mathbf{\Gamma}_k \mathbf{U}_k$:

$$\bar{\mathbf{e}}_{k+1/k+1} = \mathbf{\Phi}_k \mathbf{X}_k + \mathbf{W}_k - \mathbf{\Phi}_k \hat{\mathbf{X}}_{k/k} + \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{\Phi}_k \hat{\mathbf{X}}_{k/k}$$

$$- \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{\Phi}_k \mathbf{X}_k - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{W}_k - \mathbf{K}_{k+1}\mathbf{V}_{k+1}$$

With the sample time *k*, the of posteriori error can be written as $\bar{\mathbf{e}}_{k/k} = \mathbf{X}_k - \hat{\mathbf{X}}_{k/k}$ where

$\hat{\mathbf{X}}_{k/k} = \mathbf{X}_k - \bar{\mathbf{e}}_{k/k}$ and substitute into latter equation

$$\bar{\mathbf{e}}_{k+1/k+1} = \mathbf{\Phi}_k \mathbf{X}_k + \mathbf{W}_k - \mathbf{\Phi}_k (\mathbf{X}_k - \bar{\mathbf{e}}_{k/k}) + \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{\Phi}_k (\mathbf{X}_k - \bar{\mathbf{e}}_{k/k})$$

$$- \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{\Phi}_k \mathbf{X}_k - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{W}_k - \mathbf{K}_{k+1}\mathbf{V}_{k+1}$$

$$\bar{\mathbf{e}}_{k+1/k+1} = \mathbf{W}_k + \mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k} - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k} - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{W}_k - \mathbf{K}_{k+1}\mathbf{V}_{k+1}$$

$$= (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k} + (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\mathbf{W}_k - \mathbf{K}_{k+1}\mathbf{V}_{k+1}$$

By substituting $\mathbf{F}_{k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})$

$$\bar{\mathbf{e}}_{k+1/k+1} = \mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k} + \mathbf{F}_{k+1}\mathbf{W}_k - \mathbf{K}_{k+1}\mathbf{V}_{k+1} \tag{C.25}$$

Recall the error covariance:

$$\bar{\mathbf{P}}_{k+1/k+1} = E[\bar{\mathbf{e}}_{k+1/k+1}\bar{\mathbf{e}}_{k+1/k+1}{}^T]$$

Eqn (C.25) is substituted into error covariance Eqn above and simplified:

$$\bar{\mathbf{P}}_{k+1/k+1} =$$

$$E(\mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k} + \mathbf{F}_{k+1}\mathbf{W}_k - \mathbf{K}_{k+1}\mathbf{V}_{k+1})(\mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k} + \mathbf{F}_{k+1}\mathbf{W}_k - \mathbf{K}_{k+1}\mathbf{V}_{k+1})^T$$

$$\bar{\mathbf{P}}_{k+1/k+1} =$$

$$E[(\mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k}(\mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k})^T + \mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k}(\mathbf{F}_{k+1}\mathbf{W}_k)^T - \mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k}(\mathbf{K}_{k+1}\mathbf{V}_{k+1})^T]$$

$$+ E[\mathbf{F}_{k+1}\mathbf{W}_k(\mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k})^T + \mathbf{F}_{k+1}\mathbf{W}_k(\mathbf{F}_{k+1}\mathbf{W}_k)^T - \mathbf{F}_{k+1}\mathbf{W}_k(\mathbf{K}_{k+1}\mathbf{V}_{k+1})^T]$$

$$- E[\mathbf{K}_{k+1}\mathbf{V}_{k+1}(\mathbf{F}_{k+1}\mathbf{\Phi}_k \bar{\mathbf{e}}_{k/k})^T + \mathbf{K}_{k+1}\mathbf{V}_{k+1}(\mathbf{F}_{k+1}\mathbf{W}_k)^T - \mathbf{K}_{k+1}\mathbf{V}_{k+1}(\mathbf{K}_{k+1}\mathbf{V}_{k+1})^T]$$

Rearrange:

$$\overline{\mathbf{P}}_{k+1/k+1} =$$

$$\mathbf{F}_{k+1}\mathbf{\Phi}_k(\mathbf{F}_{k+1}\mathbf{\Phi}_k)^T E[\overline{\mathbf{e}}_{k/k}\overline{\mathbf{e}}_{k/k}{}^T] + \mathbf{F}_{k+1}\mathbf{\Phi}_k(\mathbf{F}_{k+1})^T E[\overline{\mathbf{e}}_{k/k}\mathbf{W}_k{}^T]$$

$$- \mathbf{F}_{k+1}\mathbf{\Phi}_k(\mathbf{K}_{k+1})^T E[\overline{\mathbf{e}}_{k/k}\mathbf{V}_{k+1}{}^T] + \mathbf{F}_{k+1}\mathbf{W}_k(\mathbf{F}_{k+1}\mathbf{\Phi}_k)^T E[\mathbf{W}_k\overline{\mathbf{e}}_{k/k})^T]$$
$$+ \mathbf{F}_{k+1}(\mathbf{F}_{k+1})^T E[\mathbf{W}_k\mathbf{W}_k{}^T] - \mathbf{F}_{k+1}(\mathbf{K}_{k+1})^T E[\mathbf{W}_k\mathbf{V}_{k+1}{}^T]$$
$$- \mathbf{K}_{k+1}\mathbf{F}_{k+1}\mathbf{\Phi}_k E[\mathbf{V}_{k+1}\overline{\mathbf{e}}_{k/k}{}^T] - \mathbf{K}_{k+1}\mathbf{F}_{k+1}E[\mathbf{V}_{k+1}\mathbf{W}_k{}^T] + \mathbf{K}_{k+1}\mathbf{K}_{k+1}{}^T E[\mathbf{V}_{k+1}\mathbf{V}_{k+1}{}^T]$$

Consider the uncorrelated condition

$$E[\mathbf{W}_k\mathbf{V}_{k+1}{}^T] = 0$$

$$E[\overline{\mathbf{e}}_{k/k}\mathbf{W}_k{}^T] = 0$$

$$E[\overline{\mathbf{e}}_{k/k}\mathbf{V}_{k+1}{}^T] = 0$$

$$E[\mathbf{V}_{k+1}\mathbf{W}_k{}^T] = 0$$

and covariance structure

$$E[\mathbf{W}_k\mathbf{W}_k{}^T] = \mathbf{Q}_k$$

$$E[\mathbf{V}_{k+1}\mathbf{V}_{k+1}{}^T] = \mathbf{R}_{k+1}$$

and let $E[\overline{\mathbf{e}}_{k/k}\overline{\mathbf{e}}_{k/k}{}^T]$ written as posteriori covariance,

$$\overline{\mathbf{P}}_{k+1/k+1} = \mathbf{F}_{k+1}\mathbf{\Phi}_k(\mathbf{F}_{k+1}\mathbf{\Phi}_k)^T\overline{\mathbf{P}}_{k/k} + \mathbf{F}_{k+1}(\mathbf{F}_{k+1})^T\mathbf{Q}_k + \mathbf{K}_{k+1}\mathbf{K}_{k+1}{}^T\mathbf{R}_{k+1}$$

$$= \mathbf{F}_{k+1}\mathbf{F}_{k+1}{}^T\mathbf{\Phi}_k\mathbf{\Phi}_k{}^T\overline{\mathbf{P}}_{k/k} + \mathbf{F}_{k+1}\mathbf{F}_{k+1}{}^T\mathbf{Q}_k + \mathbf{K}_{k+1}\mathbf{K}_{k+1}{}^T\mathbf{R}_{k+1}$$

$$= \mathbf{F}_{k+1}(\mathbf{\Phi}_k\overline{\mathbf{P}}_{k/k}\mathbf{\Phi}_k{}^T + \mathbf{Q}_k)\mathbf{F}_{k+1}{}^T + \mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}_{k+1}{}^T \qquad \text{(C.26)}$$

To simplified equation above, recall the error covariance from Eqn (C.23)

$$\overline{\mathbf{P}}_{k+1/k} = E[\overline{\mathbf{e}}_{k+1/k}\overline{\mathbf{e}}_{k+1/k}{}^T]$$

$$= E[(\mathbf{X}_{k+1} - \hat{\mathbf{X}}_{k+1/k})(\mathbf{X}_{k+1} - \hat{\mathbf{X}}_{k+1/k})^T] \qquad \text{(C.27)}$$

Posteriori equation $\tilde{\mathbf{X}}_k = \mathbf{\Phi}_k\mathbf{X}_{k-1} + \mathbf{\Gamma}_k\mathbf{U}_k$ can be written in the form of

$$\hat{\mathbf{X}}_{k+1/k} = \mathbf{\Phi}_k\mathbf{X}_{k/k} + \mathbf{\Gamma}_k\mathbf{U}_k .$$

Substitute this equation and Eqn (C.1) into Eqn (C.27),

$$\overline{\mathbf{P}}_{k+1/k} = E[(\mathbf{\Phi}_k\mathbf{X}_k + \mathbf{\Gamma}_k\mathbf{U}_k + \mathbf{W}_k - (\mathbf{\Phi}_k\mathbf{X}_{k/k} + \mathbf{\Gamma}_k\mathbf{U}_k))$$

$$(\mathbf{\Phi}_k\mathbf{X}_k + \mathbf{\Gamma}_k\mathbf{U}_k + \mathbf{W}_k - (\mathbf{\Phi}_k\mathbf{X}_{k/k} + \mathbf{\Gamma}_k\mathbf{U}_k)^T]$$

$$= E[(\mathbf{\Phi}_k\mathbf{X}_k + \mathbf{W}_k - (\mathbf{\Phi}_k\mathbf{X}_{k/k}))(\mathbf{\Phi}_k\mathbf{X}_k)^T + \mathbf{W}_k{}^T - (\mathbf{\Phi}_k\mathbf{X}_{k/k})^T)]$$

$$= E[(\mathbf{\Phi}_k\mathbf{X}_k(\mathbf{\Phi}_k\mathbf{X}_k)^T + \mathbf{\Phi}_k\mathbf{X}_k\mathbf{W}_k{}^T - \mathbf{\Phi}_k\mathbf{X}_k(\mathbf{\Phi}_k\mathbf{X}_{k/k})^T]$$

$$+ E[\mathbf{W}_k(\mathbf{\Phi}_k\mathbf{X}_k)^T + \mathbf{W}_k\mathbf{W}_k{}^T - \mathbf{W}_k(\mathbf{\Phi}_k\mathbf{X}_{k/k})^T]$$

$$- E[(\mathbf{\Phi}_k\mathbf{X}_{k/k}(\mathbf{\Phi}_k\mathbf{X}_k)^T + \mathbf{\Phi}_k\mathbf{X}_{k/k}\mathbf{W}_k{}^T - \mathbf{\Phi}_k\mathbf{X}_{k/k}(\mathbf{\Phi}_k\mathbf{X}_{k/k})^T]$$

Consider the uncorrelated condition,

$$\overline{\mathbf{P}}_{k+1/k} = E[(\mathbf{\Phi}_k\mathbf{\Phi}_k{}^T\mathbf{X}_k(\mathbf{X}_k - \mathbf{X}_{k/k})^T] - E[(\mathbf{\Phi}_k\mathbf{\Phi}_k{}^T\mathbf{X}_{k/k}(\mathbf{X}_k - \mathbf{X}_{k/k})^T + E[\mathbf{W}_k\mathbf{W}_k{}^T]$$

$$= \mathbf{\Phi}_k E[(\mathbf{X}_k - \mathbf{X}_{k/k})(\mathbf{X}_k - \mathbf{X}_{k/k})^T]\mathbf{\Phi}_k{}^T + E[\mathbf{W}_k\mathbf{W}_k{}^T]$$

$$= \mathbf{\Phi}_k E[(\mathbf{X}_k - \mathbf{X}_{k/k})(\mathbf{X}_k - \mathbf{X}_{k/k})^T]\mathbf{\Phi}_k{}^T + E[\mathbf{W}_k\mathbf{W}_k{}^T]$$

Therefore the a priori covariance error

$$\overline{\mathbf{P}}_{k+1/k} = \mathbf{\Phi}_k\overline{\mathbf{P}}_{k/k}\mathbf{\Phi}_k{}^T + \mathbf{Q}_k \tag{C.28}$$

Eqn (C.28) is used to replace the term $\overline{\mathbf{P}}_{k+1/k}$ in Eqn (C.26). This result to

$$\overline{\mathbf{P}}_{k+1/k+1} = \mathbf{F}_{k+1}\overline{\mathbf{P}}_{k+1/k}\mathbf{F}_{k+1}{}^T + \mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}_{k+1}{}^T$$

By replacing again $\mathbf{F}_{k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})$

$$\overline{\mathbf{P}}_{k+1/k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\overline{\mathbf{P}}_{k+1/k}(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1})^T + \mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}_{k+1}{}^T \tag{C.29}$$

The latter Eqn is known as Joseph Form of the covariance update equation.

To minimize the above Eqn we need to solve the $\mathbf{K}_{k+1}$

Using the concept introduced by [80], the measurement error can be written as

$$\mathbf{Ye}_{k+1} = \hat{\mathbf{Y}}_{k+1/k} - \mathbf{Y}_{k+1} = \mathbf{H}_{k+1}\hat{\mathbf{X}}_{k+1/k} - \mathbf{Y}_{k+1} \tag{C.30}$$

and Eqn (C.14) can be written such that

$$E[[\mathbf{X}_{k+1} - \hat{\mathbf{X}}_{k+1/k+1}]\hat{\mathbf{Y}}_{k+1/k}{}^T] = 0 \tag{C.31}$$

and by subtracting Eqn (C.14) from Eqn (C.31) yield to:

$$E[(\mathbf{X}_{k+1} - \hat{\mathbf{X}}_{k+1/k+1})\hat{\mathbf{Y}}_{k+1/k}{}^T] - E[(\mathbf{X}_{k+1} - \hat{\mathbf{X}}_{k+1/k+1})\mathbf{Y}_{k+1}{}^T] = 0$$

$$E[(\mathbf{X}_{k+1} - \hat{\mathbf{X}}_{k+1/k+1})\mathbf{Ye}_{k+1}{}^T] = 0 \text{ From (C.18)}$$

$$\hat{\mathbf{X}}_{k+1/k+1} = (1 - \mathbf{K}_{k+1}\mathbf{H}_{k+1})\hat{\mathbf{X}}_{k+1/k} + \mathbf{K}_{k+1}\mathbf{Y}_{k+1}$$

Substitute for $\mathbf{X}_{k+1}$, $\hat{\mathbf{X}}_{k+1/k+1}$ and $\mathbf{Ye}_{k+1}$ into Eqn (C.32) above.

$$E[(\mathbf{\Phi}_k\mathbf{X}_k+\mathbf{\Gamma}_k\mathbf{U}_k+\mathbf{W}_k-(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\hat{\mathbf{X}}_{k+1/k}-\mathbf{K}_{k+1}\mathbf{Y}_{k+1})(\mathbf{H}_{k+1}\hat{\mathbf{X}}_{k+1/k}-\mathbf{Y}_{k+1})^T]=0$$

$$E[(\mathbf{\Phi}_k\mathbf{X}_k+\mathbf{\Gamma}_k\mathbf{U}_k+\mathbf{W}_k-(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\hat{\mathbf{X}}_{k+1/k}-\mathbf{K}_{k+1}\mathbf{Y}_{k+1})(\mathbf{H}_{k+1}\hat{\mathbf{X}}_{k+1/k}-\mathbf{Y}_{k+1})^T]=0$$

The measurement data $\mathbf{Y}_k$ do not involve the noise term $\mathbf{W}_k$, therefore

$$E\mathbf{W}_k\mathbf{Z}_k{}^T=0 \text{ where } E\mathbf{W}_k\mathbf{Z}_k{}^T=E\mathbf{W}_k(\mathbf{H}_k\mathbf{X}_k+\mathbf{V}_k)^T$$

and lead to $E\mathbf{W}_k\mathbf{Z}_k{}^T=E\mathbf{W}_k\hat{\mathbf{X}}_k=0$

Using the uncorrelated condition as above to the latter equation yields to

$$E[(\mathbf{\Phi}_k\mathbf{X}_k+\mathbf{\Gamma}_k\mathbf{U}_k-(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\hat{\mathbf{X}}_{k+1/k}-\mathbf{K}_{k+1}\mathbf{Y}_{k+1})(\mathbf{H}_{k+1}\hat{\mathbf{X}}_{k+1/k}-\mathbf{Y}_{k+1})^T]=0$$

Substituting Eqn $\mathbf{X}_{k+1}$ and $\mathbf{Y}_{k+1}$ and using the uncorrelated condition $E\tilde{\mathbf{e}}_k\mathbf{V}_k{}^T=0$ result in Kalman gain equations.

$$E[(\mathbf{X}_{k+1}-(1-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\hat{\mathbf{X}}_{k+1/k}-\mathbf{K}_{k+1}(\mathbf{H}_{k+1}\mathbf{X}_{k+1}+\mathbf{V}_{k+1})$$

$$(\mathbf{H}_{k+1}\hat{\mathbf{X}}_{k+1/k}-(\mathbf{H}_{k+1}\mathbf{X}_{k+1}+\mathbf{V}_{k+1}))^T]=0$$

$$E[(\mathbf{X}_{k+1}-\hat{\mathbf{X}}_{k+1/k}+\hat{\mathbf{X}}_{k+1/k}\mathbf{K}_{k+1}\mathbf{H}_{k+1}-\mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{X}_{k+1}-\mathbf{K}_{k+1}\mathbf{V}_{k+1})$$

$$(\mathbf{H}_{k+1}\hat{\mathbf{X}}_{k+1/k}-\mathbf{H}_{k+1}\mathbf{X}_{k+1}-\mathbf{V}_{k+1}))^T]=0$$

$$E[(\mathbf{X}_{k+1}-\hat{\mathbf{X}}_{k+1/k}-\mathbf{K}_{k+1}\mathbf{H}_{k+1}(\mathbf{X}_{k+1}-\hat{\mathbf{X}}_{k+1/k})-\mathbf{K}_{k+1}\mathbf{V}_{k+1})$$

$$(-\mathbf{H}_{k+1}(\mathbf{X}_{k+1}-\hat{\mathbf{X}}_{k+1/k})-\mathbf{V}_{k+1}))^T]=0$$

$$E[(\bar{\mathbf{e}}_{k+1/k}-\mathbf{K}_{k+1}\mathbf{H}_{k+1}\bar{\mathbf{e}}_{k+1/k}-\mathbf{K}_{k+1}\mathbf{V}_{k+1})(-\mathbf{H}_{k+1}\bar{\mathbf{e}}_{k+1/k}-\mathbf{V}_{k+1}))^T]=0$$

$$E[(\mathbf{I}-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\bar{\mathbf{e}}_{k+1/k}-\mathbf{K}_{k+1}\mathbf{V}_{k+1})(-\mathbf{H}_{k+1}\bar{\mathbf{e}}_{k+1/k}-\mathbf{V}_{k+1}))^T]=0$$

$$(\mathbf{K}_{k+1}\mathbf{H}_{k+1}-\mathbf{I})\mathbf{H}_{k+1}{}^T E[\bar{\mathbf{e}}_{k+1/k}(\bar{\mathbf{e}}_{k+1/k})^T]+\mathbf{K}_{k+1}E[\mathbf{V}_{k+1})(\mathbf{V}_{k+1}))^T]=0$$

$$(\mathbf{K}_{k+1}\mathbf{H}_{k+1}-\mathbf{I})\mathbf{H}_{k+1}{}^T\bar{\mathbf{P}}_{k+1/k}+\mathbf{K}_{k+1}\mathbf{R}_{k+1}=0$$

$$(\mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{H}_{k+1}{}^T\bar{\mathbf{P}}_{k+1/k}-\mathbf{H}_{k+1}{}^T\bar{\mathbf{P}}_{k+1/k}+\mathbf{K}_{k+1}\mathbf{R}_{k+1}=0$$

$$(\mathbf{K}_{k+1}(\mathbf{H}_{k+1}\mathbf{H}_{k+1}{}^T\bar{\mathbf{P}}_{k+1/k}+\mathbf{R}_{k+1})-\mathbf{H}_{k+1}{}^T\bar{\mathbf{P}}_{k+1/k}=0$$

$$\mathbf{K}_{k+1}=\mathbf{H}_{k+1}{}^T\bar{\mathbf{P}}_{k+1/k}(\mathbf{H}_{k+1}\bar{\mathbf{P}}_{k+1/k}\mathbf{H}_{k+1}{}^T+\mathbf{R}_{k+1})^{-1} \quad \text{(C.33)}$$

Rearrange the Joseph Form

$$\bar{\mathbf{P}}_{k+1/k+1}=(\bar{\mathbf{P}}_{k+1/k}-\bar{\mathbf{P}}_{k+1/k}\mathbf{K}_{k+1}\mathbf{H}_{k+1})(\mathbf{I}-\mathbf{K}_{k+1}{}^T\mathbf{H}_{k+1}{}^T)+\mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}_{k+1}{}^T$$

$$\bar{\mathbf{P}}_{k+1/k+1}=\bar{\mathbf{P}}_{k+1/k}-\bar{\mathbf{P}}_{k+1/k}\mathbf{K}_{k+1}{}^T\mathbf{H}_{k+1}{}^T-\bar{\mathbf{P}}_{k+1/k}\mathbf{K}_{k+1}\mathbf{H}_{k+1}$$

$$+\overline{\mathbf{P}}_{k+1/k}\mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbf{K}_{k+1}{}^{T}\mathbf{H}_{k+1}{}^{T}+\mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}_{k+1}{}^{T}$$

$$\overline{\mathbf{P}}_{k+1/k+1}=(\mathbf{I}-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\overline{\mathbf{P}}_{k+1/k}-\overline{\mathbf{P}}_{k+1/k}\mathbf{K}_{k+1}{}^{T}\mathbf{H}_{k+1}{}^{T}$$

$$+\mathbf{K}_{k+1}(\mathbf{H}_{k+1}\overline{\mathbf{P}}_{k+1/k}\mathbf{H}_{k+1}{}^{T}+\mathbf{R}_{k+1})\mathbf{K}_{k+1}{}^{T} \qquad \text{(C.34)}$$

and by substituting Eqn (C.33) into the third terms of Eqn (C.34) to cancelled out

$$\mathbf{H}_{k+1}\overline{\mathbf{P}}_{k+1/k}\mathbf{H}_{k+1}{}^{T}+\mathbf{R}_{k+1}$$

This results to error covariance matrix equation:

$$\overline{\mathbf{P}}_{k+1/k+1}=(\mathbf{I}-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\overline{\mathbf{P}}_{k+1/k}-\overline{\mathbf{P}}_{k+1/k}\mathbf{K}_{k+1}{}^{T}\mathbf{H}_{k+1}{}^{T}+\mathbf{H}_{k+1}{}^{T}\overline{\mathbf{P}}_{k+1/k}\mathbf{K}_{k+1}{}^{T}$$

Therefore,

$$\overline{\mathbf{P}}_{k+1/k+1}=(\mathbf{I}-\mathbf{K}_{k+1}\mathbf{H}_{k+1})\overline{\mathbf{P}}_{k+1/k} \qquad \text{(C.35)}$$

For a time varying system, a matrix **G** is introduced. As a time varying process noise coupling matrix added to the system, the stochastic difference equation can be written as

$$\mathbf{X}_{k+1}=\mathbf{\Phi}_k\mathbf{X}_k+\mathbf{\Gamma}_k\mathbf{U}_k+\mathbf{G}_k\mathbf{W}_k$$

$$\mathbf{Y}_k=\mathbf{H}_k\mathbf{X}_k+\mathbf{V}_k$$

By utilising the matrix **G**, all of the KF algorithms are remain the same except for the priori error covariance matrix. The error covariance matrix used in the next chapter will be based on this equation:

$$\overline{\mathbf{P}}_{k+1/k}=\mathbf{\Phi}_k\overline{\mathbf{P}}_{k/k}\mathbf{\Phi}_k{}^{T}+\mathbf{G}_k\mathbf{Q}_k\mathbf{G}_k{}^{T} \qquad \text{(C.36)}$$

The original work on the derivation is based on [79]. Using the notation such as $\widetilde{\mathbf{X}}_i=\widehat{\mathbf{X}}_{i/j}$ makes the whole algorithm much easier to understand. However, some of the equations presented which are (C.11), (C.34), (C.37) and (C.38) has been shown directly and without proof. By combining the original work and ideas from [80], the equation has been derived in detail and gives the same results shown in Eqn (C.17), (C.28), (C.33) and (C.35).