Parameter Estimation of Induction Motors from Manufacturer Data for Simulation Purposes

by

Vinita Kumari

A thesis submitted to the Faculty of Graduate Studies of
the University of Manitoba
in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering
University of Manitoba
Winnipeg, Canada

Abstract

Induction motors form a major component of power system loads and account for a significant share of electrical energy consumption. Their operating characteristics influence power system stability, making simulation studies essential for analyzing performance under transient and steady-state conditions. To carry out such studies, electromagnetic transient simulators such as EMTDC/PSCAD™ and RTDS/RSCAD™ provide well-developed induction motor models that require equivalent circuit parameters as input. However, manufacturers typically provide only key performance characteristics—such as power factor, efficiency, and current—on nameplates and in catalog data. The required equivalent circuit parameters are not readily available to the end user.

This thesis presents a methodology for estimating the equivalent circuit parameters of three-phase induction motors using readily available manufacturer data. The estimated parameters are intended for use in electromagnetic transient simulations for power system studies. Two distinct non-linear optimization techniques are employed: the Nelder-Mead Simplex method and a variant of Genetic Algorithms. These methods determine equivalent circuit parameters to ensure that the simulated performance characteristics—such as current, efficiency, and power factor—closely match the manufacturer-published data. The proposed approach is applicable to both single-cage and double-cage induction motor models.

The quality of the estimated parameters is assessed by evaluating how closely the simulated characteristics match the published manufacturer data. While the Nelder-Mead Simplex method offers computational efficiency and ease of implementation, the Genetic Algorithm excels in global search capability, reducing the risk of getting trapped in local minima. Overall, the results establish the effectiveness of the proposed methods in estimating reliable equivalent circuit parameters for simulation applications.

A key contribution of this work is the integration of the estimation method as an interactive tool in RSCAD, designed with a flexible data input structure that accommodates both minimal nameplate data and additional catalog information. This adaptability ensures compatibility with variations in motor ratings and manufacturer standards, enabling the tool to be used directly in real-time electromagnetic transient simulations.

Acknowledgements

I would like to express my deepest gratitude to my advisor, Prof. Aniruddha Gole, for his invaluable guidance, encouragement, and insightful discussions throughout this research. His expertise and support have been instrumental in shaping this work.

I am also sincerely grateful to my supervisor at RTDS Technologies, Dr. Ali Dehkordi, for his mentorship and continuous support during my time there. His advice and technical insights have greatly contributed to the development of this research.

I would like to express my heartfelt thanks to Dr. Jhair Acosta for his extensive support and guidance throughout my thesis work. His thoughtful feedback, technical insights, and patient discussions were invaluable in shaping the direction and quality of this research. I truly appreciate the time and effort he invested in helping me overcome challenges and improve my work.

I would also like to thank the professors with whom I completed my coursework, whose teachings and insights have contributed to my academic growth. I am grateful to the department staff, particularly Shrimal, for their support and assistance throughout my studies.

I acknowledge the generous support provided by RTDS Technologies and the Mitacs program, which made this research possible. Additionally, I would like to thank Greg,

Zubaer, Soumya, and others at RTDS Technologies for their assistance and the valuable discussions that enriched my research experience.

A very special thanks goes to Tuhin, not only for his valuable advice and constant support in both academic and non-academic matters, but also for being a wonderful friend throughout this journey. His encouragement and willingness to help—no matter what it was—always made a big difference.

I am also grateful to my lab mates—Kaustav, Hao Xiao, Ajinkya, and others—for the time we spent together.

I would also like to thank my fellow cohorts and friends, especially Shehani, Devin, Gayashan and others, for the good conversations, shared experiences, and support along the way.

Finally, I am deeply grateful to my family and my friends for their unconditional love, patience, and unwavering belief in me. Their constant encouragement has been my source of strength throughout this journey. No matter the distance or the challenges, their support has never wavered, and I wouldn't have been able to reach this point without them.

Dedications

To my beloved parents and my dear brother

List of Symbols

R_s	Stator resistance
X_s	Stator leakage reactance
X_m	Magnetizing reactance
R_r	Rotor resistance (single-cage)
X_r	Rotor leakage reactance (single-cage)
X_{12}	Mutual leakage reactance between the rotor cages (double-cage)
R_1	Outer cage resistance (double-cage)
R_2	Inner cage resistance (double-cage)
X_2	Inner cage leakage reactance (double-cage)
s	Slip
I_{FL}	Rated current
I_{LR}	Locked-rotor current
PF_{FL}	Rated input power factor

P_{out}	Rated mechanical output power
Q_{in}	Rated input reactive power
T_{FL}	Rated torque
T_{LR}	Locked rotor torque
T_{max}	Maximum torque
η_{FL}	Rated efficiency

List of Abbreviations

EMT	Electromagnetic Transients
FL	Full load
GA	Genetic Algorithm
IEC	International Electrotechnical Commission
IM	Induction Motor
LR	Locked Rotor
NEMA	National Electrical Manufacturers Association
PF	Power Factor
pu	Per Unit
RPM	Revolutions per Minute

List of Figures

List of Tables

Table of Contents

Chapter 1

Introduction

1.1 Background

Electrical energy accounts for over 30% of global energy consumption, and typically 60-70% of it is consumed by electrical motors [1, 2]. Amongst electric motors, induction motors are the most widely used motors both for household and industrial applications. High popularity of induction motors is attributed to their low cost and ruggedness. Induction motors, being the most widely used family of motors, are also called the workhorse of the electric power industry [2].

Motors accounting for highest percentage of the total electrical energy consumption, form a major portion of the system load [2]. The characteristics of the load have a significant impact on the stability of the power system. Simulation studies are often required to be carried out to analyze the system performance during transient and steady-state conditions. Studies related to induction motors typically involve start-stop requirements, voltage drop calculations, black-start studies, interaction during a disturbance, etc. Power system simulation tools like

RTDS/RSCAD™ and EMTDC/PSCAD™ have well-developed models for induction motors. These models require machine data in terms of equivalent circuit parameters. However, manufacturers, except for very large motors, do not provide these parameters; rather, they provide some performance characteristics data on the machine nameplate and catalog. Traditionally, these parameters are determined through costly no-load and locked-rotor tests, which require different instruments for different motor sizes.

This thesis aims to develop a robust method for estimating equivalent circuit parameters of induction motors directly from nameplate data. The estimation method uses optimization algorithms to satisfy external performance characteristics and obtains a best-fit parameter which is suitable for simulation purposes.

1.2 Problem Statement

The performance characteristics of an induction motor are analyzed using its equivalent circuit, which is modeled based on specific study objectives. For power system studies, the equivalent circuit is commonly represented using either a single-cage or double-cage model.

This thesis focuses on estimating the equivalent circuit parameters for both single-cage and double-cage representations based solely on manufacturer-specified data. The proposed methodology involves formulating an objective function that ensures the external performance characteristics provided by the manufacturer—such as rated output power, power factor, and efficiency—are met within acceptable tolerances.

The steady-state phasor domain equations are used to define the objective function. Starting from an initial guess of the parameters, a direct search optimization algorithm is applied to minimize the objective function and determine the best-fit parameters.

Two estimation routines are developed, employing distinct optimization techniques: the Nelder-Mead Simplex (Simplex) method and a variant of Genetic Algorithms. The input data consists of readily available motor performance characteristics, including rated output power, starting torque and current, and other nameplate quantities such as power factor, efficiency, and speed at rated load conditions.

The method outputs a set of best-fit parameters that are suitable for use in Electromagnetic Transient (EMT) simulation programs. It is designed to be quick, computationally inexpensive, and efficient for practical applications.

1.3 Literature Review

Parameter estimation of induction motor has been extensively studied in literature. There are several approaches of estimating the equivalent circuit parameters depending upon the available data and the application it is intended for. Reference [3] classifies these methods into five broad categories:

i. **Parameter estimation from motor design data:** Parameters of an induction motor can be estimated from motor design data if the machine's geometry and the properties of the materials used is available in great detail. [4–6] show detailed calculations for these methods. These are the most accurate methods but are also very costly and highly complex in nature. Hence, these methods are mainly used by manufacturers for designing of the machine [3].

ii. **Frequency-domain parameter estimation:** Methods based on frequency domain estimates the parameters from transfer function obtained from stand-still frequency response tests [1, 7, 8]. Some of the methods in this category are Kalman filter [9] and

Lyapunov method [10]. These methods are very accurate, however stand-still tests are not the common industry practice [3].

iii. **Time-domain parameter estimation:** Time domain methods require time domain motor measurements. Parameter estimation is then carried out by adjusting the model parameters to match the measurements [11–13]. In these methods, for accurate estimation of parameters, high precision instrumentation is required.

iv. **Real-time parameter estimation:** Parameter estimation methods which are based on real-time measurements require continuous measurement of certain variables, such as speed, current, voltage, and so on, during machine operation. These methods are mostly applicable for drive applications for continuous tuning of control parameters.

v. **Parameter estimation based on steady-state motor models:** These methods calculate the machine parameters by solving equations derived from steady-state models and given manufacturer data. There are several kinds of optimization techniques (analytical, iterative, or evolutionary) which can be used. This method is most suitable for system studies since sufficient data is usually available to determine a motor model of reasonable accuracy [3].

1.3.1 Parameter Estimation Methods for System Studies

For system studies involving induction motors, parameter estimation methods based on steady-state motor models are the most relevant and inexpensive approach. These methods rely on nameplates data or standardized test results such as no-load and blocked-rotor tests, as outlined in IEEE Standard 112 [14]. The no-load and blocked-rotor tests help determine rotational losses and total reactance, respectively.

However, these tests are quite expensive to conduct, and manufacturers often do not provide the results, making it challenging to rely on these methods for parameter estimation. Additionally, they fall short in estimating the parameters of double-cage motors, where individual rotor resistance and reactance components are critical for meeting performance criteria like starting torque and breakdown torque [15].

There are several other methods of parameter estimation reported in literature which uses the steady-state equations to synthesize the equivalent circuit parameters from manufacturer provided data [3, 15–31].

References [4–6, 15] synthesize parameters of double-cage circuit to meet certain design criteria by using the techniques used by motor designers. These methods assume that stator circuit resistance and reactance and magnetizing reactance is fixed by the choice of stator winding and air gap dimensions. [16] extends the methods proposed in [15] with the emphasis on matching the motor output values rather than meeting certain minimum design. All of these methods require prior knowledge of stator impedance, which is usually not provided by the manufacturer.

Reference [17] uses a per unit approach with some simplified assumptions to estimate the parameters for double-cage equivalent circuit based on NEMA specifications. References [18,19] develop estimation methods based on nameplate and available catalog. Both of these methods require the availability of motor power factor and efficiency data under specific loading conditions other than full-load. This requirement limits the method's applicability, as such detailed loading condition data is not always provided by manufacturers. Reference [20] uses sensitivity analysis to make an initial guess of parameters and then obtains the final parameter values by applying some simplified assumptions. References [21, 23, 24] propose iterative algorithms based on Newton's method to solve a system of non-linear equations,

aiming to satisfy external constraints derived from performance characteristics. Reference [22] applies Genetic Algorithm and Newton's method and establishes that Newton's method being highly sensitive to the initial estimate may converge to entirely wrong solution.

References [29–31] employ swarm intelligence-based algorithms to approximate equivalent circuit parameters. However, these methods require a rigid dataset and have been validated on a limited number of machines, restricting their broader applicability.

1.4 Research Gaps

The current methods for induction motor parameter estimation face several limitations. Most techniques are tailored specifically for either single-cage or double-cage induction motors, and many rely on overly simplified assumptions that compromise their reliability. Additionally, numerous methods are heavily dependent on detailed manufacturer data, which is not always available, thereby reducing their practical applicability. Some methods have only been validated on a small number of motors, making it difficult to rely on their generalizability. While iterative and heuristic methods, such as genetic algorithms and swarm intelligence, show potential, they still require further validation to confirm their reliability and accuracy. Furthermore, there is a need for a universal method that can be adapted for both single-cage and double-cage motors and can operate effectively under various data availability conditions.

1.5 Research Objectives

The primary objective of this research is to develop a robust and efficient method for estimating induction motor equivalent circuit parameters solely from nameplate data, for both single-cage and double-cage model. The goal is to achieve a best-fit parameter set that

accurately represents the motor's behavior under various operating conditions, suitable for power system simulations.

A secondary objective is to develop the methodology as a readily available tool that can operate using different combinations of commonly available motor specifications. Motor nameplate information varies based on factors such as application, rating range, and manufacturer-specific standards. By designing an efficient tool, this research aims to enhance the applicability and usability of the parameter estimation technique across a wide range of motor types and ratings, ensuring practical implementation in real-world scenarios.

1.6 Thesis Organization

The thesis is organized into seven chapters as follows:

Chapter 1: Introduction

This chapter introduces the background and motivation for the research, outlines the problem statement, and presents the research objectives. It also includes a literature review, identifying existing methods for induction motor parameter estimation and highlighting the research gaps. Additionally, it provides an overview of the thesis organization.

Chapter 2: Three-Phase Induction Motor Model

This chapter presents the fundamental principles of induction motor operation and its equivalent circuit models. It delves into the details of single-cage and double-cage models commonly used for power system simulations.

Chapter 3: Optimization Algorithms

Chapter 3 provides an overview of the optimization algorithms employed in this research — Nelder-Mead Simplex and Genetic Algorithm. It discusses the key concepts, advantages,

and limitations of each algorithm. It also explains methods for imposing constraints in direct search algorithms.

Chapter 4: Three-Phase Induction Motor Nameplate Data

This chapter describes the important electrical information present on the nameplate of a typical three-phase induction motor and explains the minimum details consistently present on every nameplate. Additionally, it discusses their relevance to the parameter estimation method.

Chapter 5: Parameter Estimation Methodology and Implementation on External Tool

This chapter outlines the proposed methodology for estimating induction motor parameters. It describes the objective function formulation, the implementation details of the optimization algorithms, and the iterative process used to refine the parameter estimates. It also discusses the integration of the proposed methodology into the RSCAD simulation environment, a power system simulation tool.

Chapter 6: Parameter Estimation from Nameplate and Catalog Data

Chapter 6 demonstrates the application of the parameter estimation methodology by estimating the parameters of single-cage and double-cage equivalent circuits for a set of induction motors. It presents a qualitative analysis of the results, focusing on the methodology's ability to closely match the input parameters.

Chapter 7: Contributions, Conclusions and Recommendations for Future Work

Chapter 7 summarizes the key findings of the research, highlights the contributions made, and discusses the limitations and potential areas for future work. It also provides concluding remarks on the effectiveness of the proposed approach.

Chapter 2

Three-Phase Induction Motor Model

2.1 Introduction

Induction motors form a significant portion of the power system load. Commercial and industrial application induction motor size may vary from fractional small motors to thousands of horse power large motors. The aggregate effect of a large number of small size motors may be comparable to a single very large industrial motor [32]. To effectively analyze and study the performance characteristics of an induction motor as well as its behavior as a power system load, a mathematical model in terms of per phase equivalent circuit is used [33]. This chapter explains the per-phase steady- state equivalent circuit model of induction motor appropriate for system studies.

2.2 Constructional Feature and Principle of Operation

An induction machine operates on the principle of electromagnetic induction. It can operate both as a generator or motor (i.e., load). However, its performance characteristics as

generator are not so satisfactory and hence only used as a generator for special applications such as wind energy generators [34]. This is primarily because an induction generator requires an external source of reactive power to establish its magnetic field, which makes it unsuitable for standalone operation without additional equipment like capacitors or a grid connection. However, the induction machine is widely used as a motor in numerous applications.

2.2.1 Constructional Feature

In a three-phase induction motor, the stator has three-phase symmetrical windings placed in uniform slots distributed across the stator frame. The rotor may be of the wound rotor type or squirrel-cage type. A wound rotor has a complete set of three-phase winding similar to the windings on the stator. A squirrel-cage rotor winding consists of copper, aluminium, or alloy bars inserted into rotor slots and short-circuited by end-rings at both ends. A uniform air gap is maintained between stator and rotor windings. Induction motors operate with the rotor winding terminals in a short-circuited state. For squirrel cage rotors, the windings are short-circuited internally and for wound rotor motors, the windings are connected through slip rings to an external impedance.

2.2.2 Principle of Operation

Both stator and rotor windings of induction motor carry alternating currents. In a three-phase induction motor, stator windings are supplied with a balanced three-phase ac source. A set of balanced three-phase ac currents flowing through three- phase distributed winding produces a rotating magnetic field rotating at constant speed. A changing magnetic field in the stator induces a current in the rotor. Due to the interaction between the rotor currents and the stator produced magnetic field, torque is generated which accelerates the

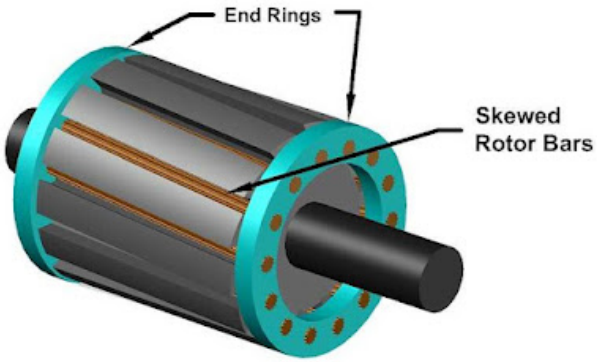

Figure 2.1: Typical rotor structure of a squirrel-cage induction motor [35]

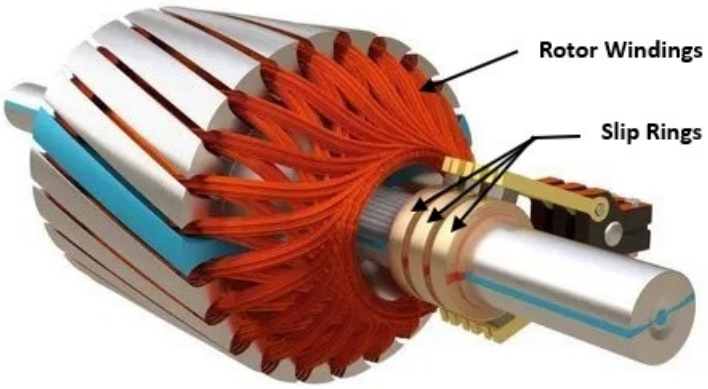

Figure 2.2: Typical rotor structure of a wound-rotor induction motor [36]

rotor in the direction of rotation of the stator magnetic field.

The speed of the rotating magnetic field is called as synchronous speed. Synchronous speed in rpm (revolutions per minute) is calculated as

$$n_{sync} = \frac{120 f_s}{P} \tag{2.1}$$

where:

- f_s: supply frequency in Hz
- P: number of poles
- n_{sync}: synchronous speed in rpm

Under normal steady-state motor operation, the rotor speed of an induction motor always remains below the synchronous speed. If the rotor were to rotate at synchronous speed, its conductors would be stationary relative to the rotating magnetic field, resulting in no induced voltage, no rotor current, and consequently no torque production. This would cause the rotor to decelerate due to friction and load torque. However, during transient events, such as a sudden drop in supply frequency, the rotor may momentarily exceed or match synchronous speed due to inertia before slowing down.

2.2.3 Generation of Rotating Magnetic Field

When three-phase distributed stator windings are excited by a balanced set of three-phase voltages, balanced three-phase current is established in the stator windings. Let the currents in the three phases be

$$\begin{aligned} i_a &= I_m \cos \omega_s t \\ i_b &= I_m \cos(\omega_s t - \frac{2\pi}{3}) \\ i_c &= I_m \cos(\omega_s t + \frac{2\pi}{3}) \end{aligned} \tag{2.2}$$

Each of these phase currents produce a pulsating sinusoidally distributed mmf wave displaced by $\frac{2\pi}{3}$ electrical radian in space from each other. At any instant of time, the peak of the mmf wave lies along the axis of the phase winding and its amplitude is proportional

to the instantaneous value of the phase current. The mmf contribution from each phase at any arbitrary point θ relative to phase a axis can be represented as

$$\begin{aligned} mmf_a(\theta) &= N_s i_a \cos\theta \\ mmf_b(\theta) &= N_s i_b \cos(\theta - \frac{2\pi}{3}) \\ mmf_c(\theta) &= N_s i_c \cos(\theta + \frac{2\pi}{3}) \end{aligned} \tag{2.3}$$

where:

- N_s: Effective number of turns per phase in the stator winding.
- i_a, i_b, i_c: Stator winding phase currents.

The resultant mmf wave in air gap at any point θ relative to axis of phase a winding is

$$mmf(\theta) = mmf_a(\theta) + mmf_b(\theta) + mmf_c(\theta) \tag{2.4}$$

Replacing the value of phase currents from equation (2.2) into equation (2.4),

$$\begin{aligned} mmf(\theta, t) &= N_s I_m \left(\cos\omega_s t \cos\theta + \cos(\omega_s t - \frac{2\pi}{3}) \cos(\theta - \frac{2\pi}{3}) + \cos(\omega_s t + \frac{2\pi}{3}) \cos(\theta + \frac{2\pi}{3}) \right) \\ &= \frac{3}{2} N_s I_m \cos(\omega_s t - \theta) \end{aligned} \tag{2.5}$$

The net mmf in the air gap given by equation (2.5) represents an mmf wave rotating at constant angular velocity ω_s electrical rad/s.

The air gap mmf wave rotating at synchronous speed links through the rotor bars (or windings). Due to the relative motion between magnetic field and the rotor conductors,

an emf is induced in the rotor windings. Since the rotor conductors are short-circuited, current is established in the rotor circuit. Similar to the stator winding, current flowing through rotor windings generates a rotating flux wave in the air gap of the machine. Due to the interaction of these two magnetic fields, torque is produced which accelerates the rotor in the direction of the stator magnetic field.

2.2.4 Concept of Rotor Slip

Under normal operation mode, both the rotor and stator magnetic fields rotate at synchronous speed, while the rotor itself operates at a lower speed. The difference between the synchronous speed i.e., the speed of rotating magnetic field and the rotor speed is commonly referred to as the slip speed of the rotor:

$$n_{slip} = n_{sync} - n_{rotor} \tag{2.6}$$

where:

- n_{slip}: slip speed of machine
- n_{sync}: synchronous speed
- n_{rotor}: mechanical shaft speed of motor

Slip is another important term, defined as the per-unit ratio of slip speed to synchronous speed.

$$s = \frac{n_{sync} - n_{rotor}}{n_{sync}} \tag{2.7}$$

The rotating air gap flux induces voltages in the rotor winding of frequency f_r called as slip frequency.

$$f_r = sf_s \tag{2.8}$$

2.3 Dynamic Model of Induction Motor

The behavior of an induction motor can be modeled by a set of non-linear differential equations that describe its transient and steady-state characteristics. First the basic machine equations in terms of phase variables is written and it is transformed to d-q reference frame. For developing a mathematical model of induction motor suitable for power system studies, following assumptions have been made:

1. Flux in the air gap is sinusoidally distributed and hence there are no space harmonics.
2. The air gap flux is not saturated.
3. The eddy current and hystersis losses are negligible.
4. Stator and rotor resistances are independent of temperature.

These assumptions are necessary to develop a practical model with reduced complexity.

2.3.1 Dynamic Equations in abc Components

These equations apply to the transient model of the IM. They are derived in [2, 37]:

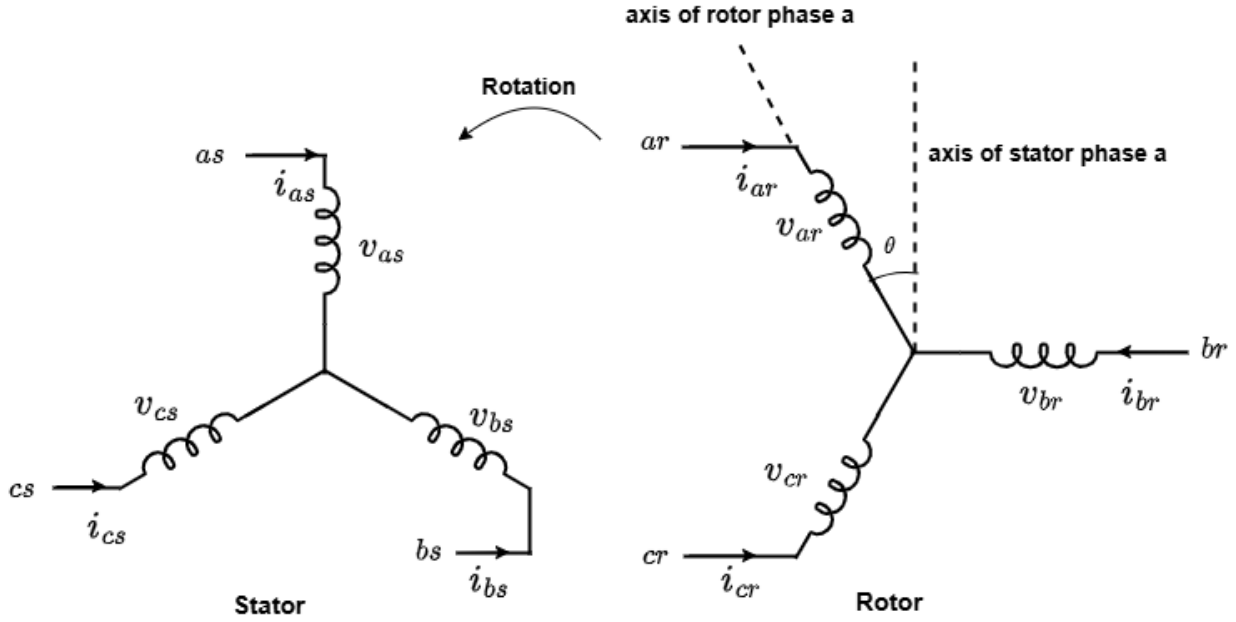

Figure 2.3: Representation of the stator and rotor circuits in a three-phase induction motor

The stator and rotor dynamic equations are derived for the induction machine using the winding current and voltage directions as shown in Figure 2.3.

- Stator voltage equations:

$$\underbrace{\begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix}}_{v_{abcs}} = R_s \underbrace{\begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix}}_{i_{abcs}} + \frac{d}{dt} \underbrace{\begin{bmatrix} \psi_{as} \\ \psi_{bs} \\ \psi_{cs} \end{bmatrix}}_{\psi_{abcs}} \tag{2.9}$$

- Rotor voltage equations:

$$\underbrace{\begin{bmatrix} v_{ar} \\ v_{br} \\ v_{cr} \end{bmatrix}}_{v_{abcr}} = R_r \underbrace{\begin{bmatrix} i_{ar} \\ i_{br} \\ i_{cr} \end{bmatrix}}_{i_{abcr}} + \frac{d}{dt} \underbrace{\begin{bmatrix} \psi_{ar} \\ \psi_{br} \\ \psi_{cr} \end{bmatrix}}_{\psi_{abcr}} \tag{2.10}$$

- Stator flux linkage equations:

$$\underbrace{\begin{bmatrix} \psi_{as} \\ \psi_{bs} \\ \psi_{cs} \end{bmatrix}}_{\psi_{abcs}} = \mathbf{L}_{ss} \underbrace{\begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix}}_{i_{abcs}} + \mathbf{L}_{sr} \underbrace{\begin{bmatrix} i_{ar} \\ i_{br} \\ i_{cr} \end{bmatrix}}_{i_{abcr}} \tag{2.11}$$

- Rotor flux linkage equations:

$$\underbrace{\begin{bmatrix} \psi_{ar} \\ \psi_{br} \\ \psi_{cr} \end{bmatrix}}_{\psi_{abcr}} = (\mathbf{L}_{sr})^T \underbrace{\begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix}}_{i_{abcs}} + \mathbf{L}_{rr} \underbrace{\begin{bmatrix} i_{ar} \\ i_{br} \\ i_{cr} \end{bmatrix}}_{i_{abcr}} \tag{2.12}$$

where

- v_{abcs}, v_{abcr}: stator and rotor voltages
- i_{abcs}, i_{abcr}: stator and rotor currents
- ψ_{abcs}, ψ_{abcr}: stator and rotor flux linkages
- R_s, R_r: stator and rotor resistances

- $\mathbf{L}_{ss}$: stator self inductances
- $\mathbf{L}_{rr}$: rotor self inductances
- $\mathbf{L}_{sr}$: stator-to-rotor mutual inductances

$$\mathbf{L}_{ss} = \begin{bmatrix} L_{ls} + L_s & L_{sm} & L_{sm} \\ L_{sm} & L_{ls} + L_s & L_{sm} \\ L_{sm} & L_{sm} & L_{ls} + L_s \end{bmatrix} \tag{2.13}$$

$$\mathbf{L}_{rr} = \begin{bmatrix} L_{lr} + L_r & L_{rm} & L_{rm} \\ L_{rm} & L_{lr} + L_r & L_{rm} \\ L_{rm} & L_{rm} & L_{lr} + L_r \end{bmatrix} \tag{2.14}$$

$$\mathbf{L}_{sr} = L_{sr} \begin{bmatrix} \cos\theta & \cos(\theta + \frac{2\pi}{3}) & \cos(\theta - \frac{2\pi}{3}) \\ \cos(\theta - \frac{2\pi}{3}) & \cos\theta & \cos(\theta + \frac{2\pi}{3}) \\ \cos(\theta + \frac{2\pi}{3}) & \cos(\theta - \frac{2\pi}{3}) & \cos\theta \end{bmatrix} \tag{2.15}$$

$$\theta = (1 - s)\omega_s t \tag{2.16}$$

where

- θ: angle by which the axis of phase 'a' rotor winding leads the axis of phase 'a' stator winding in the direction of rotation
- s: slip of rotor
- ω_s: angular velocity of stator field in electrical rad/s

Analysis has been done considering one pair of poles and hence all angles are in electrical radians.

Dynamic equations of the induction motor are mainly used for analysis of transient behavior of the system [37]. They can also be used to derive the steady-state phasor model of the machine as shown in Section 2.4.

2.3.2 Dynamic Equations in d-q Components

Dynamic equations in phase domain contain inductance terms that vary with θ, which makes the analysis complex. A simpler and more intuitive representation can be achieved by transforming the stator and rotor phase variables into d-q components along a rotating reference frame. For induction machines, an axis rotating at synchronous speed—known as the synchronous reference frame—is the preferred choice [33, 38]. In this analysis, the synchronous reference frame has been adopted to facilitate simplification and better interpretation. As shown in Figure 2.4, the convention with the q-axis at 90° ahead of the d-axis in the direction of rotation is used. Also, d-axis is chosen such that it coincides with the phase a axis of stator at time $t = 0$. At any time t, the displacement of phase d-axis from phase a axis is $\omega_s t$.

The transformation matrices for stator and rotor to convert between abc-domain variables and qdo- domain variables is:

$$\mathbf{T}_s = \frac{2}{3}\begin{bmatrix} \cos(\theta-\theta_r) & \cos(\theta-\theta_r-\frac{2\pi}{3}) & \cos(\theta-\theta_r+\frac{2\pi}{3}) \\ -\sin(\theta-\theta_r) & -\sin(\theta-\theta_r-\frac{2\pi}{3}) & -\sin(\theta-\theta_r+\frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{2.17}$$

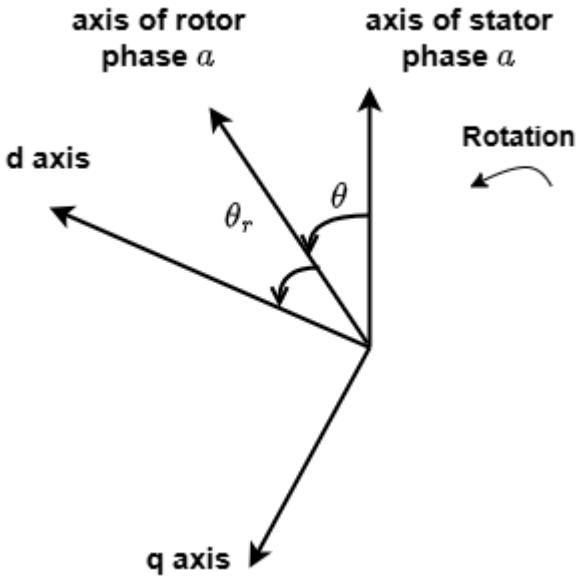

Figure 2.4: Orientation of d- and q-axes relative to the stator and rotor phase axes

$$\mathbf{T}_r = \frac{2}{3}\begin{bmatrix} \cos\theta_r & \cos(\theta_r - \frac{2\pi}{3}) & \cos(\theta_r + \frac{2\pi}{3}) \\ -\sin\theta_r & -\sin(\theta_r - \frac{2\pi}{3}) & -\sin(\theta_r + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{2.18}$$

$$\theta_r = \omega_s t - \theta \tag{2.19}$$

where θ_r is the angle by which d-axis leads phase a axis of rotor.

Applying these transformation matrices to phase domain equations of rotor and stator flux in Equations (2.11) and (2.12) yields the following in d-q domain:

$$\begin{aligned} \psi_{dq0s} &= \mathbf{T}_s\mathbf{L}_{ss}\mathbf{T}_s^{-1}i_{dq0s} + \mathbf{T}_s\mathbf{L}_{sr}\mathbf{T}_r^{-1}i_{dq0r} \\ \psi_{dq0r} &= \mathbf{T}_r\mathbf{L}_{sr}^T\mathbf{T}_s^{-1}i_{dq0s} + \mathbf{T}_r\mathbf{L}_{rr}\mathbf{T}_r^{-1}i_{dq0r} \end{aligned} \tag{2.20}$$

Similar transformations can be applied to voltage Equations in (2.9) and (2.10):

$$\begin{aligned} v_{dq0s} &= R_s i_{dq0s} + \mathbf{T}_s \frac{d}{dt}(\mathbf{T}_s^{-1}\psi_{dq0s}) \\ v_{dq0r} &= R_r i_{dq0r} + \mathbf{T}_r \frac{d}{dt}(\mathbf{T}_r^{-1}\psi_{dq0r}) \end{aligned} \tag{2.21}$$

Let

$$\begin{aligned} L_{ss} &= L_{ls} + L_s - L_{sm} \\ L_{rr} &= L_{lr} + L_r - L_{rm} \end{aligned} \tag{2.22}$$

Upon simplification, stator and rotor quantities obtained in d-q domain are:

Stator and rotor flux linkages:

$$\begin{aligned} \psi_{ds} &= L_{ss} i_{ds} + L_m i_{dr} \\ \psi_{qs} &= L_{ss} i_{qs} + L_m i_{qr} \\ \psi_{dr} &= L_{rr} i_{dr} + L_m i_{ds} \\ \psi_{qr} &= L_{rr} i_{qr} + L_m i_{qs} \end{aligned} \tag{2.23}$$

with $L_m = \frac{3}{2} L_{sr}$

Stator and rotor voltages:

$$\begin{aligned} v_{ds} &= R_s i_{ds} - \omega_s \psi_{qs} + \frac{d\psi_{ds}}{dt} \\ v_{qs} &= R_s i_{qs} + \omega_s \psi_{ds} + \frac{d\psi_{qs}}{dt} \\ v_{dr} &= R_r i_{dr} - s\omega_s \psi_{qr} + \frac{d\psi_{dr}}{dt} \\ v_{qr} &= R_r i_{qr} + s\omega_s \psi_{dr} + \frac{d\psi_{qr}}{dt} \end{aligned} \tag{2.24}$$

With balanced stator and rotor currents, only d- and q-axis components remain in d-q domain.

$$\begin{aligned} i_{as} + i_{bs} + i_{cs} &= 0 \\ i_{ar} + i_{br} + i_{cr} &= 0 \end{aligned} \tag{2.25}$$

2.4 Steady-State Model of Induction Motor

Steady-state operation is the subset of dynamic behavior of the machine. Hence, steady-state model can be derived from the dynamic equations [37]. For the problem considered in this thesis, the available manufacturer data is typically provided under steady-state conditions. Therefore, a steady-state model is considered adequate for the parameter estimation process. This section derives the steady-state model of the induction motor from its dynamic equations.

For balanced steady-state operating conditions, stator currents may be written as:

$$\begin{aligned} i_{as} &= I_m \cos(\omega_s t + \alpha) \\ i_{bs} &= I_m \cos(\omega_s t + \alpha - \frac{2\pi}{3}) \\ i_{cs} &= I_m \cos(\omega_s t + \alpha + \frac{2\pi}{3}) \end{aligned} \tag{2.26}$$

where α is the phase angle of i_a with respect to the time origin.

Applying the d-q transformation yields:

$$\begin{aligned} i_{ds} &= I_m \cos\alpha \\ i_{qs} &= I_m \sin\alpha \end{aligned} \tag{2.27}$$

Thus, for balanced steady-state operation, stator currents appear as direct currents in d-q reference frame. Stator voltage and rotor current can be expressed similarly.

Under steady-state sinusoidal conditions, all the time derivative terms are zero and stator

and rotor phasors are related to d-q components as:

$$\begin{aligned} V_s &= \frac{1}{\sqrt{2}}(v_{ds} + jv_{qs}) \\ I_s &= \frac{1}{\sqrt{2}}(i_{ds} + ji_{qs}) \\ I_r &= \frac{1}{\sqrt{2}}(i_{dr} + ji_{qr}) \end{aligned} \tag{2.28}$$

where V_s, I_s and I_r are the phasor form of stator voltage, stator current, and rotor current respectively.

Substituting the d-q domain variables as phasor variables in Equations (2.23) and (2.24) and combining them yields:

$$\begin{aligned} V_s &= R_s I_s + j\omega_s L_{ss} I_s + j\omega_s L_m I_r \\ &= R_s I_s + j\omega_s (L_{ss} - L_m) I_s + j\omega_s L_m (I_s + I_r) \end{aligned} \tag{2.29}$$

where

$X_s = \omega_s(L_{ss} - L_m)$ = stator leakage reactance

$X_m = \omega_s L_m$ = magnetizing reactance

With rotor short-circuited $v_{dr} = v_{qr} = 0$. Hence, from Equations (2.23) and (2.24),

$$\begin{aligned} v_{dr} = 0 &= R_r i_{dr} - s\omega_s(L_{rr} i_{qr} + L_m i_{qs}) \\ v_{qr} = 0 &= R_r i_{qr} + s\omega_s(L_{rr} i_{dr} + L_m i_{ds}) \end{aligned} \tag{2.30}$$

From Equations,(2.28) and (2.30),

$$
\begin{aligned}
V_r = 0 &= \frac{R_r}{s} I_r + j\omega_s L_{rr} I_r + j\omega_s L_m I_s) \\
&= \frac{R_r}{s} I_r + j X_r I_r + j X_m (I_s + I_r)
\end{aligned}
\tag{2.31}
$$

where: $X_r = \omega_s(L_{rr} - L_m)$ = rotor leakage reactance

2.5 Equivalent Circuit Representation of Induction Motor

The operational characteristics of an induction motor can be represented using an equivalent circuit model. The equivalent circuit model enables the analysis and prediction of the motor's performance with reasonable accuracy.

Equations (2.29) and (2.31) represent the steady-state performance of the induction motor. These equations can be represented by the per-phase equivalent circuit shown in Figure 2.5, where all quantities are referred to the stator side.

This representation is called as the single-cage equivalent circuit representation as the rotor circuit is modeled by a single set of windings. This is mostly used to represent wound rotor and single squirrel-cage rotor designs.

An optimal design of an induction motor requires high torque during startup and high efficiency during operation. High starting torque is achieved with high rotor resistance, but high rotor resistance results in lower efficiency. One way to meet these conflicting requirements is to use wound rotor motors, in which external resistances are inserted during startup to increase starting torque and removed during operation for better efficiency. However, wound rotor motors are very expensive, require high maintenance, and have complex automatic control circuits [34].

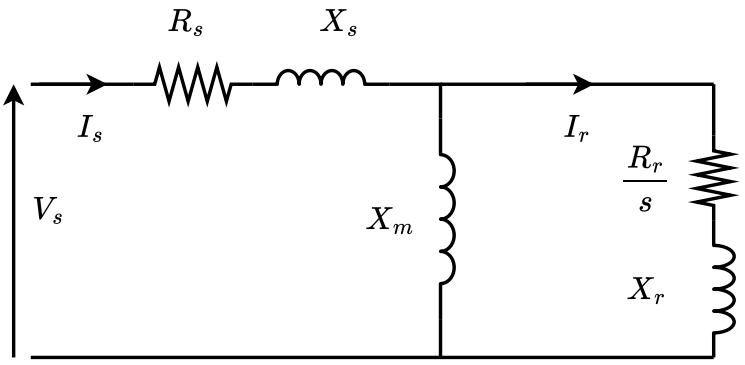

Figure 2.5: Single-cage equivalent circuit representation of induction motor

The elements of the single-cage equivalent circuit shown in Figure 2.5 are described as follows:

V_s : terminal voltage

I_s : stator current

I_r : rotor current

R_s : stator resistance

X_s : stator leakage reactance

X_m : magnetizing reactance

R_r : rotor resistance

X_r : rotor reactance

To address these conflicting requirements during startup and operation, squirrel-cage induction motors have alternative designs such as deep-bar rotors and double squirrel-cage

rotors. These designs offer high resistance during startup, meeting the requirement for high starting torque, and low resistance during steady-state operation, maintaining high efficiency. These motors are best represented by double-cage equivalent circuits, as shown in Figure 2.6 [2].

The double-cage equivalent circuit representation is essentially an extension of the single-cage equivalent circuit representation, with an additional winding to accommodate the unique characteristics of these motors. The details of double-cage and deep-bar type rotors are explained in Section 2.6.

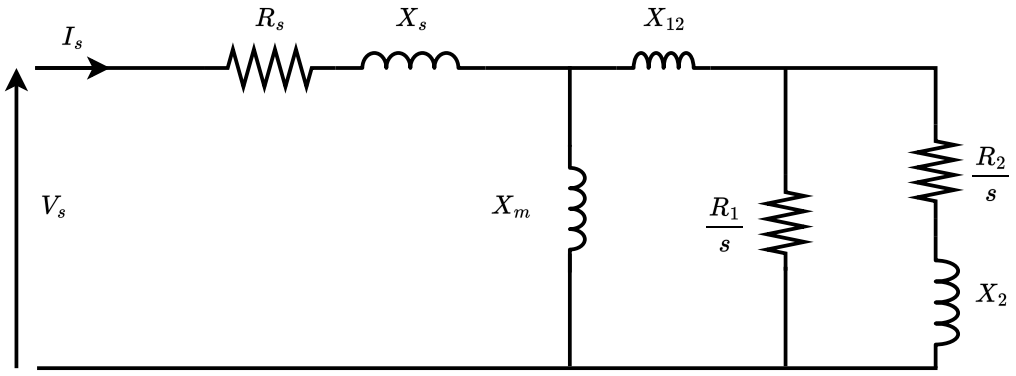

Figure 2.6: Double-cage equivalent circuit representation of induction motor

The elements of the double-cage equivalent circuit shown in Figure 2.6 are described as follows:

V_s : terminal voltage

I_s : stator current

R_s : stator resistance

X_s : stator leakage reactance

X_m : magnetizing reactance

X_{12} : mutual reactance between rotor cages

R_1 : outer-cage rotor resistance

R_2 : inner-cage rotor resistance

X_2 : inner-cage rotor reactance

2.6 Deep-bar and Double-cage Rotor Designs

Special squirrel-cage designs, such as deep and narrow bars or double squirrel-cage rotors, are used to achieve a high effective resistance during starting and a low resistance during full-load operation. The designs of deep and narrow bar rotors and double-cage rotors are shown in Figures 2.7 and 2.8 respectively.

A double squirrel-cage rotor consists of two layers of rotor bars, both short-circuited by end rings. The upper bars, with a small cross-section, have high resistance and are positioned near the rotor surface. This placement ensures the leakage flux encounters high reluctance, resulting in low leakage inductance. The lower bars, on the other hand, have a larger

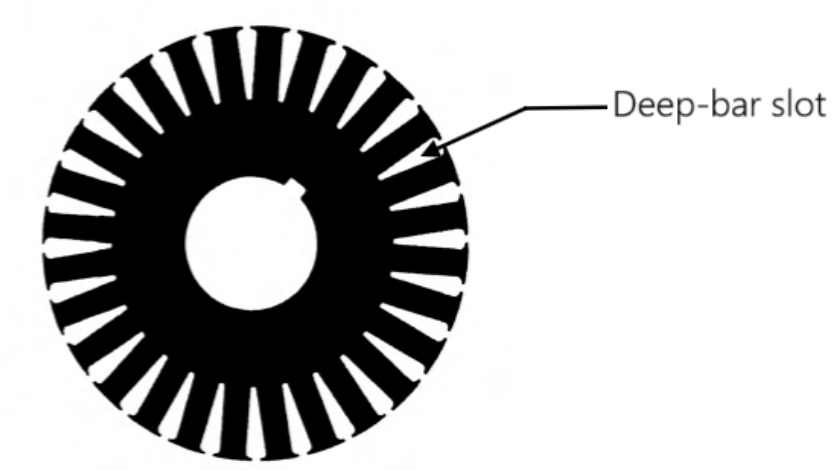

Figure 2.7: Typical slots of a deep-bar induction motor rotor [34]

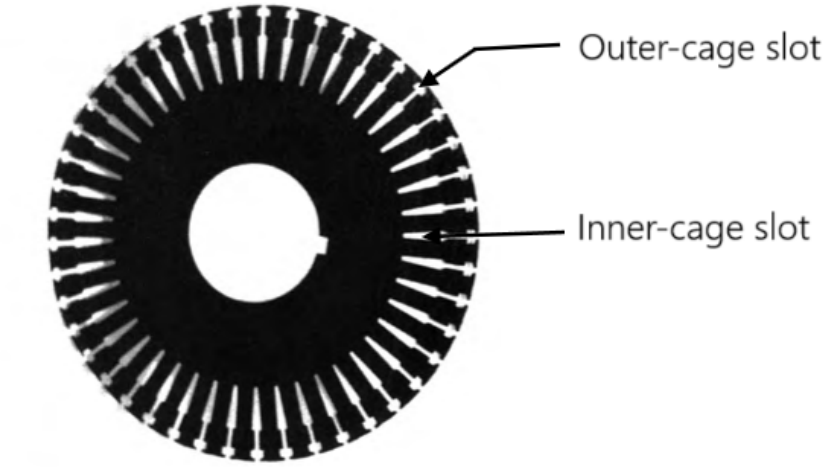

Figure 2.8: Typical slots of a double-cage induction motor rotor [34]

cross-section, lower resistance, and higher leakage inductance. At starting, when the rotor frequency is high, most of the current flows through the high-resistance upper bars, as the lower bars carry minimal current. During normal operation with low slip, leakage reactances become negligible, and the majority of the current flows through the low-resistance lower bars. In this state, the effective rotor resistance is equivalent to the two sets of bars in parallel.

The deep and narrow bar rotor design produces torque-speed characteristics similar to those of a double-cage rotor. Due to the proximity of the air gap near the upper cross-section, leakage reactance is relatively low and increases progressively towards the bottom layer. At starting, with high rotor frequency, current tends to concentrate in the top layers of the rotor bar. As the rotor accelerates and the slip decreases, the current distribution becomes more uniform, resulting in lower effective resistance.

The equivalent circuit of a double-cage induction motor is illustrated in Figure 2.6. Since the deep-bar rotor exhibits characteristics similar to those of a double-cage rotor, it can also be represented using the same equivalent circuit [2].

2.7 Significance of Single-cage and Double-cage Representations for Power System Studies

For power system studies, there are two widely accepted induction machine equivalent circuit models — single-cage and double-cage.

Equivalent circuit representation used for power system studies generally ignore the resistance corresponding to the core loss component. With core loss resistance not taken into account, the equivalent circuit can not be used in energy management type of studies, where efficiency calculation plays an important role [32]. EMT simulation softwares such as PSCAD® and RSCAD® have well developed induction machine models which are used for power system studies, and these models do not include core loss resistance in them. Also, RSCAD induction machine has models up to two cages while PSCAD model has options which can model up to three cages. Reference [4] mentions that double-cage circuit is convenient to represent the performance characteristics of triple squirrel cages or deep bar

squirrel cage rotors.

2.8 Conclusion

This chapter derived the equations of steady-state equivalent circuit of induction motor from its dynamic equations. The steady-state equivalent circuit model of induction motor derived in this chapter is suitable for power system studies. These circuits have been used in the later part of the thesis for developing parameter estimation method.

Parameter estimation method requires the calculation of the primary steady-state performance characteristics- efficiency, power factor, current, starting torque, maximum torque, and related parameters. Detailed equations for these characteristics, applicable to both single-cage and double-cage equivalent circuits, are provided in appendices A and B respectively.

Chapter 3

Optimization Algorithms

The parameter estimation method developed in this thesis is formulated as a constrained minimization problem that requires the use of a non-linear optimization algorithm along with appropriate constraint-handling techniques. Two direct-search optimization algorithms are employed: the Nelder-Mead Simplex method and a variant of Genetic Algorithms. This chapter provides an overview of these two algorithms and their application in solving the parameter estimation problem. Additionally, it explains the method for handling constraints in direct search algorithms employed in this thesis.

3.1 Introduction

Optimization algorithms can be broadly classified into gradient-based methods and direct search techniques, each with distinct advantages and limitations. In this study, two direct search algorithms — the Nelder-Mead Simplex method and the Genetic Algorithm (GA) — have been investigated for parameter estimation due to their ability to optimize complex objective functions without requiring explicit gradient information. The parameter

estimation problem addressed in this thesis involves a highly nonlinear objective function where derivative computation is quite complex. Consequently, direct search methods offer a practical and effective alternative.

The Nelder-Mead Simplex method is a deterministic algorithm that has been widely applied in various optimization contexts due to its ease of implementation and relatively low computational cost. However, its performance tends to degrade as the problem's dimensionality increases. Literature suggests that the method exhibits slow convergence in higher-dimensional spaces due to search directions becoming increasingly orthogonal to the local downhill gradient [39–41]. Some studies indicate that its effectiveness may be compromised in problems with as few as eight variables [42]. However, for the parameter estimation problem in this thesis, which involves seven decision variables for the double-cage induction machine, the Simplex method has demonstrated satisfactory performance.

In contrast, the Genetic Algorithm is a stochastic search technique that employs probabilistic transition rules, enabling broader exploration of the search space. This characteristic helps GA avoid getting trapped in local optima, making it more resistant to premature convergence. As a result, it is better suited for high-dimensional and complex optimization problems. However, this advantage comes at the cost of increased computational expense and the need for careful tuning of hyperparameters such as population size, mutation rate, etc., which are problem-specific. While GA is more computationally demanding than the Simplex method, its ability to escape local minima makes it an attractive alternative for complex search landscapes.

This thesis investigates the Nelder-Mead Simplex method and the Genetic Algorithm, comparing their effectiveness in estimating induction motor parameters. It explores the trade-offs between these methods and evaluates their suitability for the given estimation

problem. The subsequent sections provide a brief overview of the algorithms and the specific parameter settings used in this study.

3.2 Nelder-Mead Simplex Algorithm

The Nelder-Mead Simplex [43] algorithm is a zero-order optimization technique that does not require any gradient information of the objective function. It is highly efficient in optimizing a multi-variable non-linear objective function whose gradient is difficult to calculate. The method was originally developed by Spendley, Hext, and Himsworth in 1962 [44]. The method that is used in present form today, was modified and improved by Nelder and Mead in 1965 and is now popularly known as Nelder-Mead Simplex algorithm.

The method uses the concept of simplex. A simplex is a geometric figure consisting of $n + 1$ vertices in an n-dimensional space. For example:

- For a 2-variable problem, a simplex is a triangle (3 vertices).
- For a 3-variable problem, a simplex is a tetrahedron (4 vertices).

The algorithm refines the simplex iteratively using geometric transformations—reflection, expansion, contraction, and shrinkage—to converge to the optimal solution of the objective function.

3.2.1 Geometric Transformations of Simplex Method

In a minimization problem, the Nelder-Mead Simplex algorithm iteratively modifies a simplex — a geometric figure consisting of $n+1$ vertices in an n-dimensional space — to move away

from the worst solution (the vertex with the highest function value) toward better solutions with lower function values. These modifications are performed using a set of geometric transformations that adapt the simplex shape based on the function values at its vertices.

The transformations involved in the algorithm are illustrated in Figure 3.1 for a two-variable problem. At each iteration, the algorithm evaluates the function values at the simplex vertices and applies one of the following transformations to guide the search toward improved solutions:

- **Reflection:** The worst vertex is reflected across the centroid of the remaining vertices. This attempts to move the simplex toward a lower function value by stepping in the opposite direction of the worst solution, effectively exploring new regions of the search space.
- **Expansion:** If reflection results in a significantly better function value, the algorithm extends the step further in the same direction. This allows the simplex to take larger strides when moving toward promising regions.
- **Contraction:** If reflection fails to produce improvement, the algorithm contracts the simplex by moving the worst vertex closer to the centroid. This prevents unnecessary exploration in unproductive directions and focuses the search on a smaller region.
- **Shrinkage:** If none of the previous transformations yield improvement, all vertices (except the best one) are moved closer to the best vertex. This reduces the simplex size and concentrates the search around the most promising solution found so far.

By iteratively applying these transformations, the simplex adapts to the landscape of the objective function, seeking to converge toward an optimal solution.

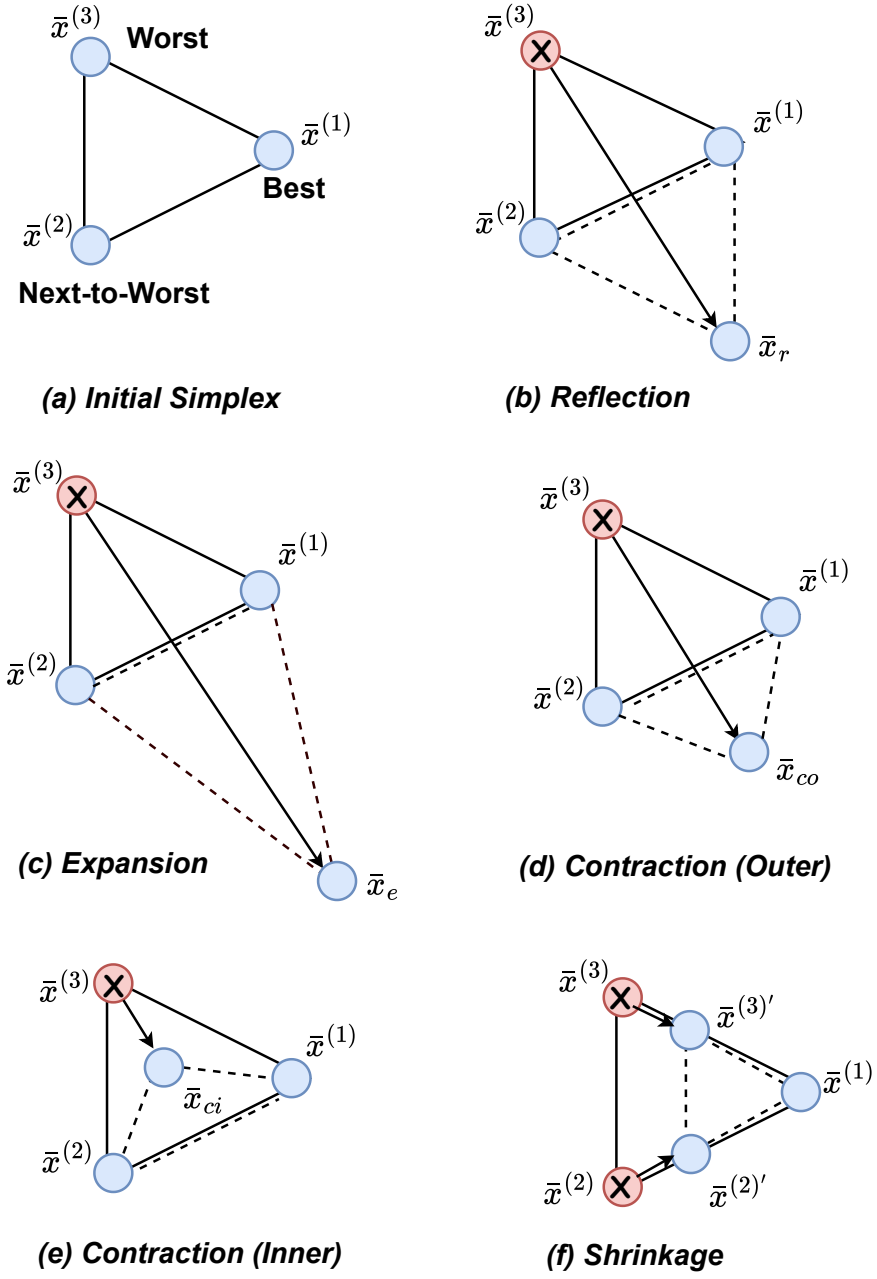

Figure 3.1: Visualization of Nelder-Mead Simplex algorithm transformations for 2-variable minimization problem

3.2.2 Optimization of an n-variable Problem

To minimize an objective function $f(\bar{x})$ characterized by n decision variables, $\bar{x} = [x_1, x_2, x_3, \ldots, x_n]^T$, the steps of the algorithm are:

1. **Initialization:**

   - Define an initial solution $\bar{x}^{(0)} = [x_1^{(0)}, x_2^{(0)}, x_3^{(0)}, \ldots, x_n^{(0)}]^T$ in n-dimensional space.
   - Generate a simplex with $n+1$ vertices around $\bar{x}^{(0)}$. The vertices are denoted as:

   $$\bar{x}^{(1)}, \bar{x}^{(2)}, \ldots, \bar{x}^{(n+1)}.$$

   - Set algorithm parameters: reflection coefficient α, expansion coefficient γ, contraction coefficient β, and shrinkage coefficient δ.
   - Define stopping criteria based on a tolerance or maximum number of iterations.

2. **Evaluate the Objective Function:** Compute the function values at all $n+1$ vertices of the simplex:

   $$f(\bar{x}^{(1)}), f(\bar{x}^{(2)}), \ldots, f(\bar{x}^{(n+1)}).$$

3. **Sort the Vertices:** Sort the vertices based on their function values in ascending order:

   $$f(\bar{x}^{(1)}) \leq f(\bar{x}^{(2)}) \leq \ldots \leq f(\bar{x}^{(n+1)}),$$

   where $\bar{x}^{(1)}$ having the lowest function value is the best point and $\bar{x}^{(n+1)}$ having the highest function value is the worst point.

4. **Compute the Centroid:** Exclude the worst point $\bar{x}^{(n+1)}$ and compute the centroid $\bar{x}_c$ of the remaining n points:

$$\bar{x}_c = \frac{1}{n} \sum_{i=1}^{n} \bar{x}^{(i)}.$$

5. **Reflection:** Reflect the worst point $\bar{x}^{(n+1)}$ across the centroid $\bar{x}_c$ to generate the reflection point $\bar{x}_r$:

$$\bar{x}_r = \bar{x}_c + \alpha(\bar{x}_c - \bar{x}^{(n+1)}).$$

   Evaluate $f(\bar{x}_r)$:

   - If $f(\bar{x}^{(1)}) \leq f(\bar{x}_r) < f(\bar{x}^{(n)})$, accept $\bar{x}_r$ and replace $\bar{x}^{(n+1)}$ with $\bar{x}_r$.

6. **Expansion:** If $f(\bar{x}_r) < f(\bar{x}^{(1)})$, perform an expansion to generate $\bar{x}_e$:

$$\bar{x}_e = \bar{x}_c + \gamma(\bar{x}_r - \bar{x}_c).$$

   Evaluate $f(\bar{x}_e)$:

   - If $f(\bar{x}_e) < f(\bar{x}_r)$, accept $\bar{x}_e$ and replace $\bar{x}^{(n+1)}$ with $\bar{x}_e$.
   - Otherwise, accept $\bar{x}_r$.

7. **Contraction:** If $f(\bar{x}_r) \geq f(\bar{x}^{(n)})$, perform a contraction:

   - For an *outside contraction*, calculate:

$$\bar{x}_{oc} = \bar{x}_c + \beta(\bar{x}_r - \bar{x}_c).$$

- For an *inside contraction*, calculate:

$$\bar{x}_{ic} = \bar{x}_c + \beta(\bar{x}^{(n+1)} - \bar{x}_c).$$

Evaluate $f(\bar{x}_{oc})$ or $f(\bar{x}_{ic})$, and accept the better point if it improves upon $\bar{x}^{(n+1)}$.

8. **Shrink:** If the contraction fails, shrink the simplex towards the best point $\bar{x}^{(1)}$:

$$\bar{x}^{(i)} = \bar{x}^{(1)} + \delta(\bar{x}^{(i)} - \bar{x}^{(1)}), \quad \text{for } i = 2, \ldots, n+1.$$

9. **Check for Convergence:** Terminate the algorithm if the simplex size or the difference in function values across vertices falls below a specified tolerance. Otherwise, repeat the process from Step 2.

The overall process of the Nelder-Mead Simplex algorithm is summarized as a flowchart in Figure 3.2, providing an overview of the key steps involved. The specific parameter setting values used in the implementation of the algorithm for this thesis are listed in Table 3.1. The coefficient values chosen are commonly used in the literature and have been found to work effectively for this parameter estimation problem.

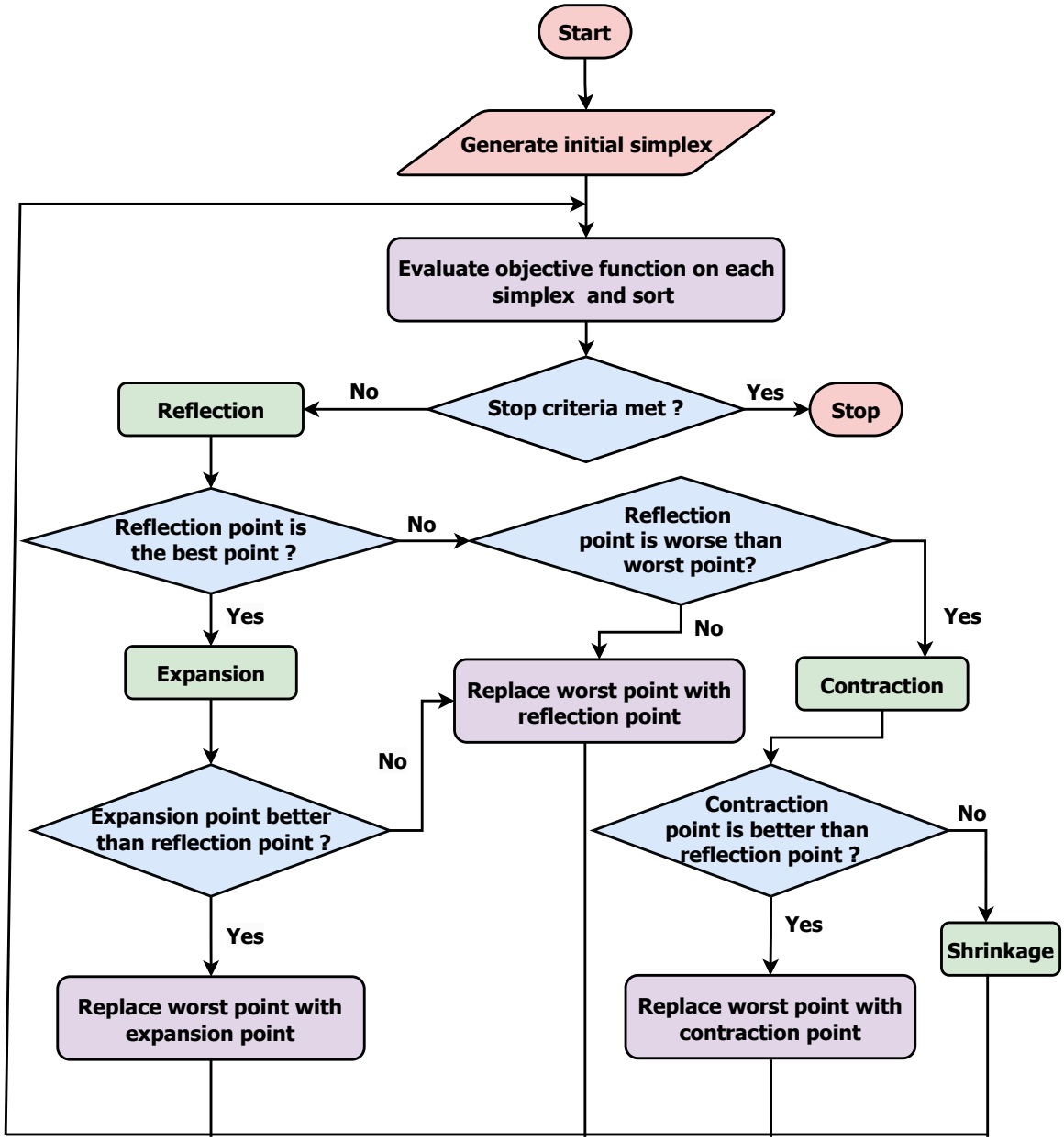

Figure 3.2: Nelder-Mead Simplex algorithm flowchart

Table 3.1: Nelder-Mead Simplex algorithm parameters

Parameter	**Value/Description**
Reflection coefficient (α)	1.0
Expansion coefficient (γ)	2.0
Contraction coefficient (β)	0.5
Shrinkage coefficient (δ)	0.5
Maximum iterations	10000
Fitness convergence value	$\leq 10^{-6}$
Stopping criteria	Fitness convergence/Max. iterations

3.3 Genetic Algorithms

Genetic Algorithms are heuristic optimization techniques inspired by the principles of natural evolution and genetics [45]. They are highly effective for tackling complex optimization problems characterized by large, multidimensional search spaces and the presence of multiple local optima [46]. The algorithm was first introduced by John Holland in the 1970s. It is a population-based approach that mimics the evolutionary processes of reproduction, mutation, recombination, and selection to evolve candidate solutions towards the optimal solution [47]. Each candidate solution, also called a chromosome, represents a point in the search space and encodes a possible solution to the optimization problem. The flowchart shown in Figure 3.3 outlines the steps involved in a typical optimization routine based on Genetic Algorithms.

3.3.1 Overview of Genetic Algorithms

The Genetic Algorithms process begins with the initialization of a population of candidate solutions (individuals), which represent points in the search space. Each individual

corresponds to a possible solution and may be encoded in various forms, such as binary strings, real numbers, or other symbolic representations. In this thesis, real numbers have been used to represent individual solutions, as this provides a more natural representation of the continuous parameter space and eliminates the need for binary-to-real conversion.

Once the initial population is generated, it evolves over multiple generations through the application of genetic operators such as selection, crossover, and mutation. In each generation, the fitness of each individual is evaluated using a predefined fitness function, which quantifies how well a candidate solution satisfies the optimization objective. Based on this evaluation, individuals are selected, modified through crossover and mutation, and passed on to the next generation, driving the optimization process forward.

3.3.2 Genetic Algorithm Operators

Any GA largely relies on three fundamental operators -selection, crossover, and mutation - to guide the search process and refine candidate solutions over successive generations [48]. These operators play a crucial role in directing the optimization process and significantly influence the algorithm's performance. Different variations of these operators exist, and their choice impacts the effectiveness of the GA. The specific genetic operators used in the implementation of GA in this thesis are explained in detail in subsequent sections.

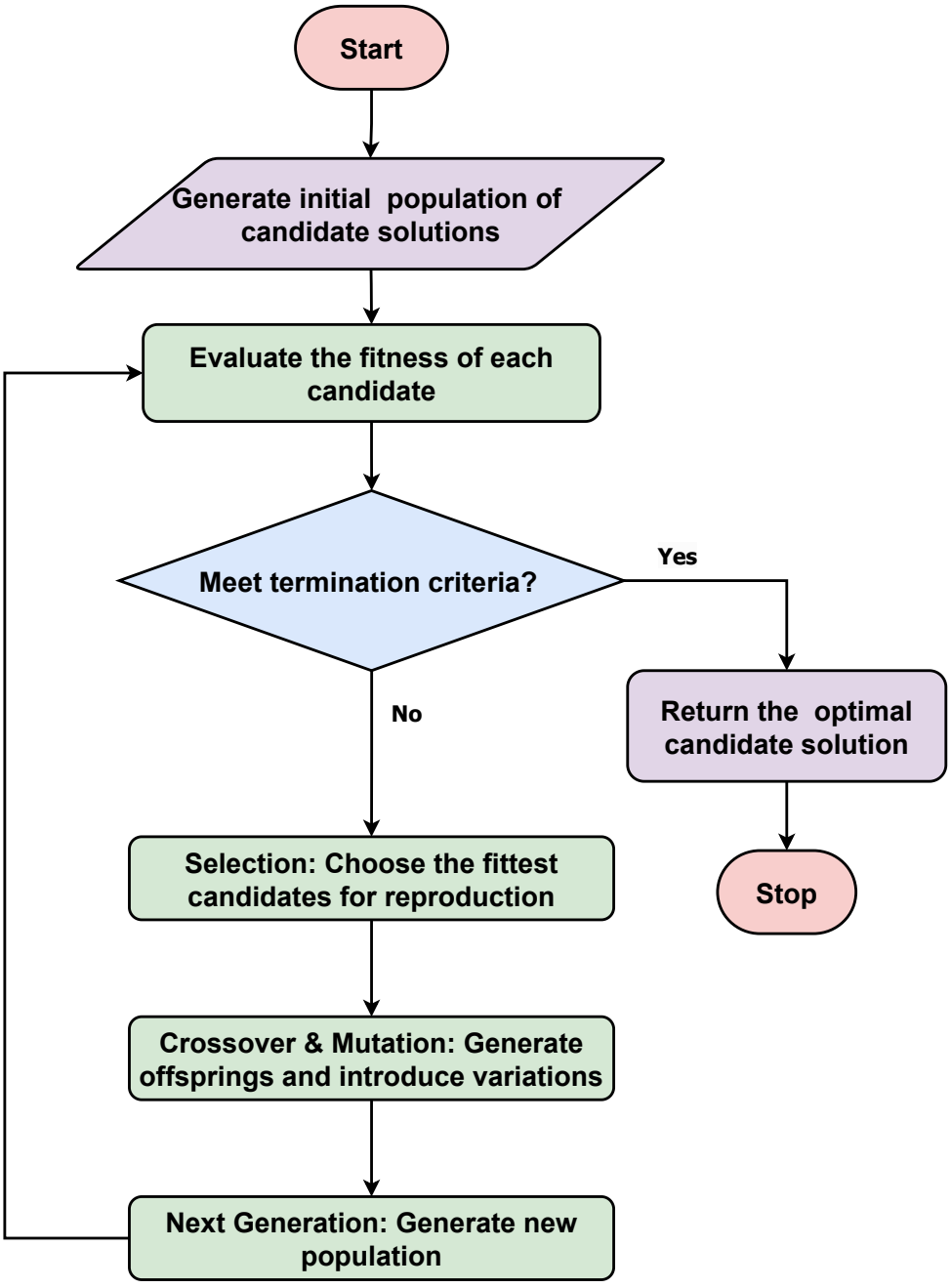

Figure 3.3: Genetic Algorithms flowchart

Selection

Selection is typically the first operator applied to the population. It is the process of choosing individuals from the population for reproduction. It determines which individuals from the current population will contribute to the next generation. It is guided by the principle of survival of the fittest: individuals with higher fitness are more likely to be selected.

Several selection methods exist in the literature, each with unique characteristics. For this thesis, the tournament selection method is used. In this method, each candidate solution competes against a randomly chosen subset of individuals from the population. The fitness of each solution is compared against the subset, and the number of wins is recorded as its score. Candidates are then ranked based on their scores, and the top-performing solutions are selected for reproduction.

Unlike other methods of selection, tournament selection is superior in maintaining a balance between population diversity and selective pressure [48]. It ensures that fitter individuals have a higher chance of reproduction while still allowing less fit individuals a chance to compete, thus preserving diversity within the population.

Crossover

Crossover is another fundamental genetic operator in GA that facilitates the exchange of genetic information between parent solutions to generate new offspring. It mimics the biological recombination process, where traits from two parents combine to produce a new individual. The purpose of crossover is to explore new regions of the parameter space by creating diverse solutions while preserving high-quality genetic traits from the parent population. By recombining different parts of promising solutions, the GA increases the likelihood of finding an optimal or near-optimal solution.

In this thesis, a real-valued arithmetic crossover method has been employed. The specific version of crossover implemented in this thesis follows a process where, after the selection phase, two parent solutions are chosen randomly. For each decision variable, a random weight factor α is generated within the range $(0, 1)$. Using this factor, two offspring solutions are created as follows:

$$\begin{aligned} \text{child}_1^j &= \alpha \cdot \text{parent}_1^j + (1 - \alpha) \cdot \text{parent}_2^j \\ \text{child}_2^j &= \alpha \cdot \text{parent}_2^j + (1 - \alpha) \cdot \text{parent}_1^j \end{aligned} \tag{3.1}$$

where j represents the index of the decision variable, and α is a randomly generated number in $(0, 1)$ for each variable. This approach ensures that the offspring solutions lie within the convex combination of their parents, which helps to maintain a balance between exploration and exploitation of the solution space [48]. The procedure is repeated for all decision variables to produce two offspring solutions per pair of parents.

This method enables a smooth exploration of the search space by allowing partial inheritance from both parents, preventing premature convergence, and promoting diversity in the population.

Mutation

Mutation is a fundamental genetic operator in GA that introduces random variations in the population to maintain diversity and prevent premature convergence. By perturbing selected individuals, mutation ensures that the search process does not get trapped in local optima and continues exploring new regions of the solution space. The mutation operator is typically applied with a low probability to avoid excessive randomness while still enabling occasional modifications to candidate solutions.

The specific mutation strategy implemented in this thesis involves applying a Gaussian perturbation to each decision variable with a certain mutation probability. For each decision variable x_j, a probability p_m is drawn from a uniform distribution. If p_m is below a predefined mutation threshold P_{mutation}, the decision variable is modified as follows:

$$x'_j = x_j + \mathcal{N}(0, \sigma_j) \tag{3.2}$$

where $\mathcal{N}(0, \sigma_j)$ represents a Gaussian random variable with zero mean and standard deviation σ_j, which is computed as:

$$\sigma_j = \frac{|b_j - a_j|}{3} \tag{3.3}$$

Here, a_j and b_j represent the lower and upper bounds of the step size (step size is restricted to a range between 50% and 200% of the initial guess for the variable). This ensures that the mutation step size adapts to the feasible region of the solution space. The mutation process is repeated for all decision variables in an individual. A mutation rate of 5% has been found to work well for the parameter estimation problem in this study and is used in the implementation.

By using a Gaussian distribution, the mutation process ensures that the new solutions are not drastically different from the parent solutions but are instead sampled from a neighborhood of the current solution. This helps maintain solution quality while allowing gradual exploration of the parameter space. Other mutation methods, such as uniform mutation, could be less effective in continuous spaces as they might introduce larger changes that are less meaningful in terms of refining parameter values.

After the mutation process, the algorithm proceeds to the next generation by evaluating the fitness of the updated population. This iterative process continues until one or more predefined termination criteria are met. The termination conditions are designed to stop the optimization process once a satisfactory solution has been found or if further generations no longer contribute significant improvements. The program terminates once one of the following termination criteria is met:

- A maximum number of generations is reached.
- A predefined fitness threshold is achieved.
- The improvement in fitness over successive generations becomes negligible ($\leq 10^{-3}$).

The specific parameter setting values used in the GA implementation in this thesis is summarized in Table 3.2.

Table 3.2: Genetic Algorithm parameters

Parameter	**Value/Description**
Population size	50
Number of generations	2000
Selection method	Tournament selection
Crossover method	Arithmetic crossover
Mutation method	Gaussian
Crossover rate	Random value between (0,1)
Mutation rate	0.05
Fitness threshold value	$\leq 10^{-6}$
Fitness threshold between generations	$\leq 10^{-3}$

3.4 Constrained Optimization

Optimization problems often involve constraints that define the feasible region within which the optimal solution must lie. The optimization of complex, real-world engineering problems often involves one or more of the following types of constraints:

- Variable bounds, including upper and/or lower limits
- Equality Constraints
- Inequality Constraints

The constraints define the feasible region, and the optimization algorithm must ensure that all solutions lie within this region.

3.4.1 Problem Formulation for Constrained Optimization

A general constrained optimization problem can be mathematically expressed as:

$$\min_{\mathbf{x}} \quad f(\mathbf{x}) \tag{3.4}$$

$$\text{subject to} \quad g_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \ldots, m, \tag{3.5}$$

$$h_j(\mathbf{x}) = 0, \quad j = 1, 2, \ldots, n, \tag{3.6}$$

where:

- $f(\mathbf{x})$ is the objective function to be minimized.
- $g_i(\mathbf{x}) \leq 0$ are m inequality constraints.
- $h_j(\mathbf{x}) = 0$ are n equality constraints.

- $\mathbf{x}$ is the vector of decision variables.

Variable bounds and equality constraints can be transformed into inequality constraints to make them easier to handle.

A variable bound $a \leq x \leq b$ can be expressed as two inequality constraints [49]:

$$x - a \geq 0 \quad \text{and} \quad b - x \geq 0.$$

An equality constraint $h(x) = 0$ can be converted into two inequalities [49]:

$$h(x) \geq 0 \quad \text{and} \quad h(x) \leq 0.$$

3.4.2 Method for Handling Constraints

Constrained optimization aims to find the best solution to an objective function while satisfying a given set of constraints. While constraints reduce the search space, they also increase the complexity of the optimization problem.

Various approaches for constraint handling—such as the penalty function method, change of variable method, and others—are available in the optimization literature [50, 51]. The suitability of each method depends on the nature of the problem and the optimization algorithm used.

This thesis employs the "change of variable method" for constraint handling due to its straightforward implementation and its ability to inherently satisfy constraints. A detailed explanation of this method is provided in the following sections.

Change of Variable Method

The change of variables method is an approach in optimization that simplifies the imposition of constraints by transforming the original variables into new variables. New variables are chosen in such a way that transformation ensures that the constraints get inherently satisfied. Objective function is reformulated with the transformed variables, allowing the optimization algorithm to operate in an unconstrained space. This also eliminates the need for explicit constraint-handling within the optimization routine. This is especially advantageous for direct search algorithms, such as the Nelder-Mead Simplex and GA, which operate optimally in unconstrained spaces.

To minimize an objective function $f(x)$ subject to the constraint $a \leq x \leq b$, the change of variables method can be employed to simplify the problem. By redefining the variable x in terms of a new variable z, we can ensure that the constraints are inherently satisfied.

Consider the transformation:

$$x = \frac{b+a}{2} + \frac{b-a}{2}\sin(z),$$

where z is the new variable.

Explanation of the Transformation:

- The sine function $\sin(z)$ always lies within the interval $-1 \leq \sin(z) \leq 1$, regardless of the value of z.
- As a result, x is guaranteed to remain within the bounds $a \leq x \leq b$.

By substituting this transformation into the original objective function $f(x)$, the problem

is reformulated entirely in terms of z. This results in a new, equivalent optimization problem: $\min f(z)$, where z replaces x as the decision variable. This transformation eliminates the need to explicitly enforce the bounds $a \leq x \leq b$, allowing the optimization process to operate in an unconstrained domain for z.

Alternative transformations, such as using absolute value ($x = |z|$) or a squared term ($x = z^2$), can be used in cases where variables must be restricted to non-negative values. The choice of transformation depends on the specific constraint requirements and the smoothness properties needed for the optimization algorithm [52].

3.5 Conclusion

This chapter explained the Nelder-Mead Simplex and Genetic Algorithm optimization techniques, emphasizing the specific parameters and configurations applied in this thesis for parameter estimation. Additionally, it discussed the use of the change-of-variable method for handling constraints in direct search algorithms. These optimization methods and techniques form the foundation of the parameter estimation approach, which is further developed and applied in the following chapters of this thesis.

Chapter 4

Three-Phase Induction Motor Nameplate Data

The primary objective of this thesis is to estimate the parameters of a three-phase induction motor solely from its nameplate data. These estimated parameters should, at the very least, be able to reproduce the performance characteristics listed on the nameplate. The electrical performance characteristics provided on the nameplate can vary significantly depending on the standards followed by manufacturers and the specific application of the motor. This chapter explains the typical electrical information found on a three-phase induction motor's nameplate and identifies the minimum set of data that is consistently present.

4.1 Introduction

The nameplate of a three-phase induction motor provides essential information about its design characteristics and performance specifications. This data is critical for selecting, operating, and maintaining motors in industrial and commercial applications. The specific

information on the nameplate is primarily governed by the standards that manufacturers follow.

There are two major standards for motors followed worldwide: NEMA (National Electrical Manufacturers Association) in North America and IEC (International Electrotechnical Commission) in Europe and most of the world. Additionally, many countries have their own standards, such as the Canadian Standards Association (CSA), the British Standards (BS), and the Japanese Electrotechnical Committee (JEC). However, all these standards are closely derived from NEMA and IEC [53].

NEMA and IEC present similar critical information on motor nameplates but in slightly different ways. For example, NEMA specifies locked-rotor current using a NEMA kVA code, whereas IEC motors convey this information through a different coding system [54]. Additionally, NEMA standards specify a wider range of tolerances, while IEC motors tend to be more precise with less built-in tolerance.

The minimum set of electrical performance characteristics specified on a typical induction motor nameplate (IEC or NEMA) are:

- **Rated output power (HP or kW)**: Rated output power refers to the mechanical power delivered at the motor shaft. It is typically expressed in horsepower (hp) for NEMA motors and in kilowatts (kW) for IEC motors.

- **Rated line voltage (V)**: It specifies the supply voltage for which the motor is designed to operate most efficiently. NEMA motors specify a tolerance of $\pm 10\%$ of the rated value, while IEC motors include a tolerance of only $\pm 5\%$ [55, 56].

- **Rated full-load primary current (A)**: Full-load primary current indicates the current drawn by the motor at rated load and rated voltage. NEMA nameplates allow

for a tolerance of ±10%, while IEC nameplates are more exact and do not account for tolerances at all.

- **Rated full-load speed (RPM)**: The rated full-load speed expressed in revolutions per minute indicates the speed at which motor delivers full-load torque under rated voltage and frequency conditions.
- **Frequency (Hz)**: This is the supply frequency (e.g., 50 Hz or 60 Hz) at which the motor is designed to operate at the rated output power, voltage, and speed.
- **Number of phases**: It specifies the number of supply phases for polyphase motors.
- **Nominal efficiency**: It indicates the motor's efficiency at full-load and rated conditions. Commonly, efficiency is specified as the average of efficiency of a large number of motors of that class.
- **Secondary full-load current (A)**: This is specific to wound rotor motors only. It specifies the secondary (rotor) current at full-load.
- **Secondary open-circuit voltage (V)**: This is also specific to wound rotor motors only. The secondary voltage of wound-rotor motors is the open-circuit voltage at standstill, measured across the slip rings, with rated voltage applied on the primary winding.

Apart from these, two important additional specifications that may be present on a NEMA squirrel cage motor are:

- **Locked-Rotor Current**: Locked-rotor current is the current drawn by motor at full voltage with the rotor at standstill. NEMA motors specify locked-rotor current by a

series of code letters from 'A' to 'V' which indicate a range of the locked-rotor kVA per HP.

- **Design**: NEMA sqirrel cage motors have four design classes that are specified by design letters from 'A' to 'D' which indicates the torque, slip and starting characteristics of the motor.

Detailed explanation of the Locked-rotor current codes and design is provided in subsequent Sections 4.2 and 4.3 respectively.

The nameplates of a typical three-phase NEMA induction motor and a typical three-phase IEC induction motor are shown in Figures 4.1 and 4.2, respectively.

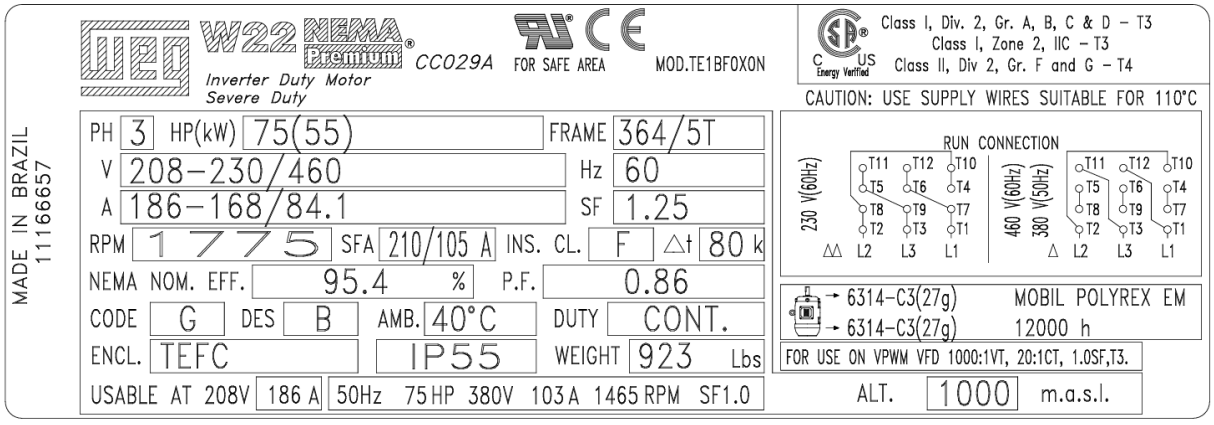

Figure 4.1: Typical nameplate of a three-phase NEMA induction motor

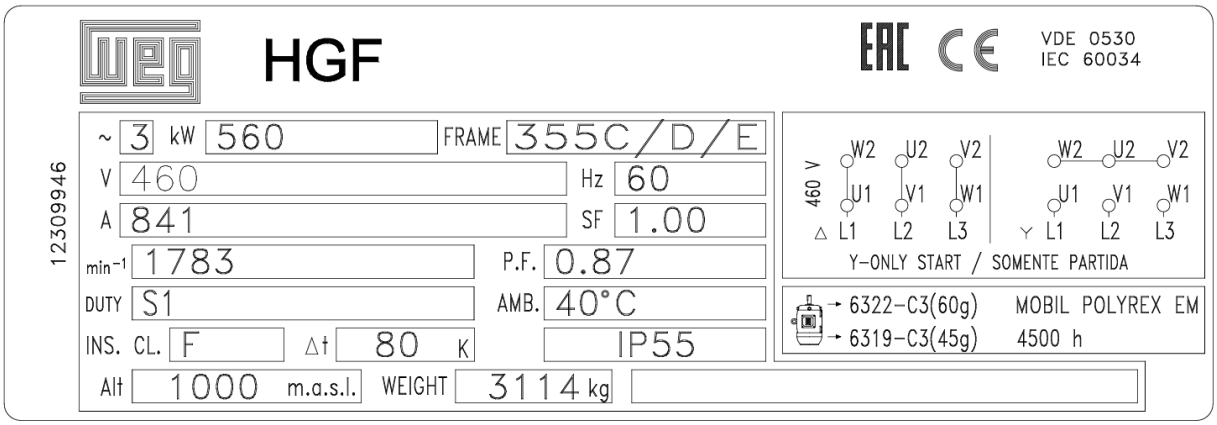

Figure 4.2: Typical nameplate of a three-phase IEC induction motor

4.2 NEMA Locked-Rotor kVA Code Letters

The code letter on the NEMA motor nameplate indicates the locked-rotor kVA per horsepower (kVA/HP) rating. Locked-rotor kVA provides insight into the motor's starting characteristics, specifically the current drawn during start-up at full voltage and rated frequency. The NEMA code classifications for locked-rotor kVA/HP is shown in Table 4.1.

4.3 NEMA Design Letter of Polyphase Squirrel Cage Motors

The National Electrical Manufacturers Association (NEMA) classifies polyphase squirrel-cage induction motors into four design classes (A, B, C, and D) based on their torque-speed characteristics, starting currents, and typical applications. Torque-speed characteristics for these are shown in Figure 4.3. A brief overview of these design classes is

Table 4.1: NEMA code letter classification [55]

NEMA Code Letter	Locked-Rotor kVA/HP
A	0-3.15
B	3.15-3.55
C	3.55-4.0
D	4.0-4.5
E	4.5-5.0
F	5.0-5.6
G	5.6-6.3
H	6.3-7.1
J	7.1-8.0
K	8.0-9.0
L	9.0-10.0
M	10.0-11.2
N	11.2-12.5
P	12.5-14.0
R	14.0-16.0
S	16.0-18.0
T	18.0-20.0
U	20.0-22.4
V	22.4 and above

provided below:

- **Design A:** These motors typically exhibit a maximum slip of 0.05 pu at full-load. Their maximum torque occurs at a slip value below 0.20 pu. They are characterized by high to medium starting currents, ranging from 5- 8 times of rated current. They possess normal locked-rotor torque and breakdown torque, with the latter reaching 2-3 times of full-load torque. Design A motors are well-suited for general-purpose

applications such as fans and pumps.

- **Design B:** These motors also have a maximum full-load slip of 0.05 pu. They often employ deep-bar rotor designs to achieve low starting current while maintaining high locked-rotor torque. The breakdown torque of design B motors is greater than twice of full-load torque but lower than that of design A. Due to their low starting current, design B motors have largely replaced design A in many applications. They are suitable for a wide range of uses, particularly those with normal starting torque requirements, such as HVAC systems, fans, blowers, and pumps.

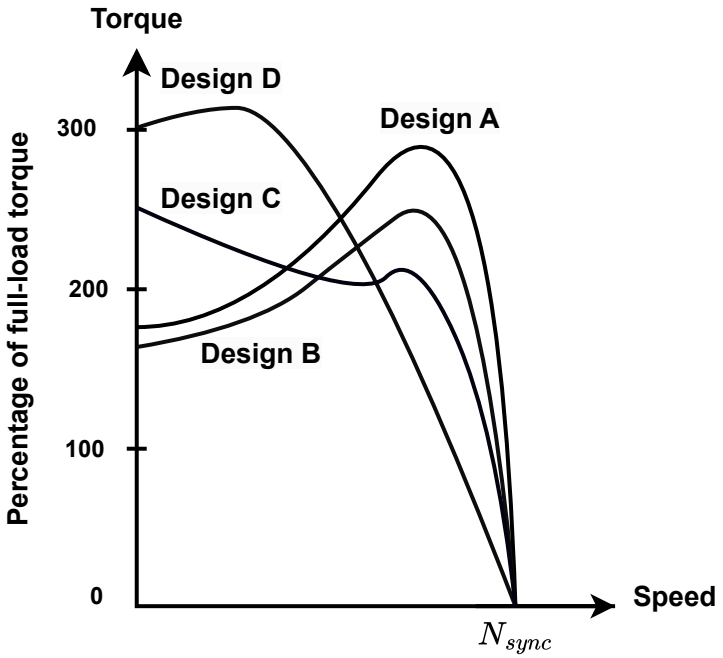

Figure 4.3: Typical torque-speed curves for different NEMA design classes of induction motor [34, 57]

- **Design C:** These motors, like design B, have a maximum full-load slip of 0.05 pu. They typically employ deep-bar or double-cage rotor designs to achieve low starting current and high locked-rotor torque. However, their breakdown torque is slightly lower than that of design A. These motors are primarily used for applications that require high starting torque.

- **Design D:** These motors operate with a higher maximum full-load slip range of 0.05-0.13 pu. They employ rotor bars made of high-resistance materials, which shifts the maximum torque to higher slip values. This design results in very low starting current and exceptionally high locked-rotor torque, exceeding 2.75 times of full-load torque. These motors are ideally suited for applications with extremely high starting inertia, such as cranes, hoists, and flywheels.

Based upon the design class, motors generally tend to have a particular ratio of stator to rotor reactance [34]. The relationship is summarized in the Table 4.2.

Table 4.2: Empirical relation between X_s and X_r based upon design class of IM [34]

DESIGN	**Stator Reactance** (X_s)	**Rotor Reactance** (X_r)
Wound Rotor	0.5	0.5
A	0.5	0.5
B	0.4	0.6
C	0.3	0.7
D	0.5	0.5

4.4 Conclusion

This chapter provided an overview of the essential electrical information published on the nameplate of a three-phase induction motor. It outlined the minimum set of data that is consistently present on every nameplate, regardless of the manufacturer's standard. This published electrical information, provided by the manufacturer, serves as the foundation for developing a parameter estimation methodology in the subsequent chapters of this thesis.

Chapter 5

Parameter Estimation Methodology and Implementation on External Tool

This chapter develops a methodology for estimating the parameters of three-phase induction motors based on their nameplate specifications. The method applies to both single-cage and double-cage equivalent circuit model parameters. Two distinct optimization algorithms are investigated and implemented: the Nelder-Mead Simplex algorithm and the Genetic Algorithm. The method using the Nelder-Mead Simplex optimization is implemented in RSCAD as a custom library component for the RTDS real-time simulator.

5.1 Introduction

Simulating an induction motor in EMT simulation platforms such as RSCAD requires the equivalent circuit parameters such that the simulation results from these parameters closely match the published performance characteristics of the motor. Manufacturers, however, typically do not provide these parameters. Instead, they supply performance characteristics,

such as rated power, speed, torque, current, and more, on the motor's nameplate or in a separate catalog. Translating this data into equivalent circuit parameters is a complex task, as the data provided varies depending on the motor's rating, intended application, and manufacturer-specific standards.

This chapter presents a methodology for estimating these parameters exclusively from the available nameplate or catalog data. The parameter estimation problem is formulated as a non-linear, constrained optimization problem. The objective is to minimize the discrepancy between the simulated performance characteristics, calculated using the estimated parameters, and the actual performance characteristics specified by the manufacturer.

To achieve this, an objective function is derived from the motor's steady-state equations. The steady-state equations used for the objective function formulation is detailed in Appendices A and B. Beginning with an initial parameter estimate, the optimization algorithm iteratively minimizes the objective function.

In this thesis, two distinct optimization algorithms—the Nelder-Mead Simplex algorithm and a variant of the Genetic Algorithm—are applied. Both algorithms are selected for their robustness and their ability to explore the solution space effectively without the need for gradient information. These methods start with initial solutions and iteratively refine them to achieve an optimal or near-optimal parameter set in a finite number of iterations.

5.2 Parameter Estimation Methodology

The proposed parameter estimation method uses performance characteristics such as rated output power, current, and other data from the nameplate and catalog as input. This

input data is used to calculate additional performance characteristics and provide an initial estimate of parameters. An objective function is then formulated, based on the available input data, which aims to minimize the difference between the given and the derived performance characteristics.

This objective function, along with the initial parameter estimates, is minimized using an optimization algorithm. This thesis examines two popular optimization algorithms: the Nelder-Mead Simplex method and a variant of the Genetic Algorithms. Both algorithms can minimize the objective function, outputting the estimated parameters for either single-cage or double-cage induction motors, as specified by the user.

Each step of this process is explained in detail in later subsections. An overview of the key steps in the estimation process is outlined in the flowchart in Figure 5.1.

5.2.1 Step 1: Manufacturer Input Data

The first step in the estimation process involves reading input data from the motor nameplate, as well as any available catalog data. While the specific information provided on the nameplate may vary between different motors, there is a minimum set of data that is consistently present on every nameplate. This essential data is used as input for the method. The required minimum input data that is available on every nameplate includes:

1. Rated output power (P_{out})
2. Rated line-to-line supply voltage (U)
3. Rated current (I_{FL})
4. Rated speed in RPM (RPM)

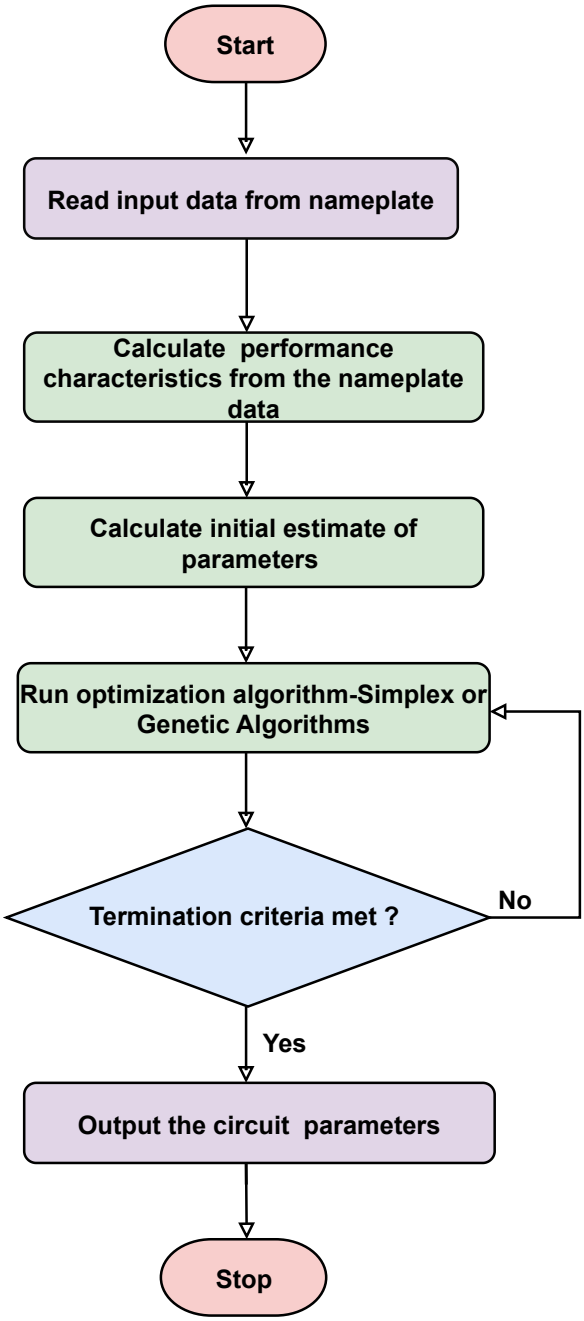

Figure 5.1: Induction motor parameter estimation process flowchart

5. Supply frequency (f)

6. Rated efficiency (η_{FL})

7. Rated power factor (PF_{FL})

Key performance characteristics available on the nameplate, such as output power, efficiency, and input power factor at rated conditions, are used to construct an objective function. The objective function aims to generate a parameter set which is capable of reproducing these performance characteristics.

Additional data, such as locked-rotor kVA, maximum torque and locked-rotor torque may also be available depending on the motor's manufacturer, power rating, and application. If available, this additional data can be incorporated into the objective function to improve the accuracy of the parameter estimation.

If magnetic saturation data is available, the parameter estimation algorithm can be extended to include nonlinear magnetization characteristics. This involves augmenting the equivalent circuit model with a nonlinear inductance representation—such as that described in [2]—to reflect the B-H curve of the magnetic core. The estimation algorithm may then be adapted to minimize the discrepancy between measured and simulated responses while accounting for the nonlinear behavior of the magnetizing branch.

Incorporating saturation effects can enhance the accuracy of the estimated parameters, particularly under high excitation or fault conditions, where linear models may not adequately capture magnetic characteristics. However, this also introduces additional complexity into the estimation process, and such detailed saturation data is typically not provided by the manufacturer. This activity is left for future work.

5.2.2 Step 2: Calculation of Additional Performance Characteristics from Nameplate Data

The minimum set of data available on nameplate is used to calculate some additional performance characteristics, such as rated torque and input reactive power. Along with the performance characteristics available on nameplate, these calculated characteristics are also included in the objective function that aims to minimize the difference between the calculated and the known characteristics.

For the nameplate under consideration, the following additional characteristics are calculated:

- Rated torque:

$$T_{FL} = \frac{P_{out}}{\frac{2\pi}{60} RPM} \tag{5.1}$$

- Rated input reactive power:

$$Q_{in} = \frac{P_{out}\sqrt{1 - PF_{FL}^2}}{\eta_{FL} PF_{FL}} \tag{5.2}$$

If kVA code is available on the nameplate, locked-rotor current can be calculated as :

- Locked-rotor current:

$$I_{LR} = \frac{P_{out} LR_{kVA/HP}}{745.7\sqrt{3}U} \tag{5.3}$$

The locked-rotor kVA per HP ($LR_{kVA/HP}$) is determined using the NEMA code letter specification (see Table 4.1), which provides an upper and lower range of kVA values for a given motor design type. The mean value of the upper and lower limits is taken

to get a rough estimate of the locked-rotor current.

All equations employ SI units, unless stated otherwise.

5.2.3 Step 3: Calculation of Initial Estimate

The parameter estimation process uses an optimization algorithm to minimize the objective function. Alongside the input data, which forms the basis for the objective function, the optimization algorithm also requires an initial estimate of the decision variables (i.e., estimated parameters). Each solution of the objective function is a vector of decision variables.

The decision variables for single-cage and double-cage equivalent circuits have four and seven parameters, respectively.

- **Single-cage model decision variables**: $\vec{x} = \{R_s, X_s, X_m, R_r\}$
  - A single-cage equivalent circuit of an induction motor (see Figure 2.5) has five circuit parameters. However, as discussed in Chapter 4, motors of a given design class generally exhibit a characteristic ratio of stator to rotor reactance [34]. To account for this, a relationship between X_s and X_r is imposed, as summarized in Table 4.2. This reduces the number of independent decision variables to four.
- **Double-cage model decision variables**: $\vec{x} = \{R_s, X_s, X_m, X_{12}, R_1, R_2, X_2\}$
  - Unlike the single-cage equivalent circuit, the literature does not specify any equality relationship among the parameters of the double-cage equivalent circuit. Consequently, the number of decision variables for double-cage parameter estimation corresponds to the number of circuit parameters in the model.

– However, inequality constraints—such as $R_1 > R_2$ and $X_2 > X_{12}$, which are characteristic of double-cage circuits—are imposed in the estimation process of double-cage parameters.

The quality of the estimated parameters significantly depends on the initial estimates provided to the optimization algorithms. A well-chosen initial estimate can greatly enhance the convergence speed and accuracy of the results. Based on certain simplified assumptions, the initial estimates for the equivalent circuit parameters of single-cage and double-cage models are calculated as follows [23]:

Initial Estimate for Single-Cage Model

- Magnetizing Reactance:

$$X_m \approx \frac{U^2}{Q_{in}} \tag{5.4}$$

  where Q_{in} is the rated input reactive power. This equation assumes that most of the reactive power is consumed by the magnetizing reactance.

- Stator Reactance:

$$X_s \approx 0.07 X_m \tag{5.5}$$

  This assumes that the stator reactance is typically 5-10% of the magnetizing reactance.

- Rotor Resistance:

$$R_r \approx \frac{U^2 \cdot s_{FL}}{P_{out}} \tag{5.6}$$

  where s_{FL} is the rated slip. This equation assumes that the rotor resistance is primarily responsible for power dissipation at full-load.

- Stator Resistance:

$$R_s \approx R_r \tag{5.7}$$

For initial estimation, stator and rotor resistances are assumed to be equal.

Initial Estimate for Double-Cage Model

The initial estimates for the double-cage circuit model are derived from the single-cage estimates [23]. The stator parameters R_s, X_s, and X_m are assumed to remain the same for both models. The additional rotor parameters for the double-cage model are initialized as follows:

- Rotor Resistance (Inner Cage):

$$R_2 \approx R_r \tag{5.8}$$

- Rotor Resistance (Outer Cage):

$$R_1 \approx 2R_r \tag{5.9}$$

- Rotor Reactance (Inner Cage):

$$X_2 \approx 2X_s \tag{5.10}$$

- Mutual Rotor Reactance:

$$X_{12} \approx X_s \tag{5.11}$$

One set of initial estimate is calculated using the Equations (5.4) to (5.11). Two distinct optimization algorithms are utilized, each requiring multiple initial estimates for the design variables.

For an n-variable optimization, the Nelder-Mead Simplex algorithm requires $n + 1$ initial estimates, while the Genetic Algorithm requires an initial population equal to the predetermined population size. In this work, a population size of 50 was found to be adequate. Therefore, to generate the additional initial estimates needed for each algorithm, the one set of calculated estimates are varied randomly between 50% and 200% of the values of each design variable. These initial estimates provide a reliable starting point for the optimization algorithms employed in this study.

5.2.4 Step 4: Formulation of Objective Function

The objective function is designed to minimize the difference between the manufacturer-provided performance characteristics and those calculated from the equivalent circuit model. The performance characteristics included in objective function include efficiency, power factor, output power, torque, input reactive power at rated conditions. Additionally, it includes current and torque under locked-rotor conditions, as well as the maximum torque if available.

The objective function is defined by Equations (5.12) to (5.14) and is expressed as follows:

$$fitness(\vec{x}) = \min\left(\frac{\sum_{n=1}^{N} w_n f_n^2(\vec{x})}{N}\right) \tag{5.12}$$

where f_n, for $n = 1, \ldots, N$ with $N = 9$, represents the normalized error for each characteristic (i.e., sub-objective), defined as:

$$
\begin{aligned}
f_1 &= \frac{I_{\mathrm{FL,c}} - I_{\mathrm{FL}}}{I_{\mathrm{FL,c}}}, && \text{(Rated current error)} \\
f_2 &= \frac{T_{\mathrm{FL,c}} - T_{\mathrm{FL}}}{T_{\mathrm{FL,c}}}, && \text{(Rated torque error)} \\
f_3 &= \frac{P_{\mathrm{out,c}} - P_{\mathrm{out}}}{P_{\mathrm{out,c}}}, && \text{(Rated mechanical output power error)} \\
f_4 &= \frac{PF_{\mathrm{FL,c}} - PF_{\mathrm{FL}}}{PF_{\mathrm{FL,c}}}, && \text{(Rated input power factor error)} \\
f_5 &= \frac{\eta_{\mathrm{FL,c}} - \eta_{\mathrm{FL}}}{\eta_{\mathrm{FL,c}}}, && \text{(Rated efficiency error)} \\
f_6 &= \frac{Q_{\mathrm{in,c}} - Q_{\mathrm{in}}}{Q_{\mathrm{FL,c}}}, && \text{(Reactive input power error)} \\
f_7 &= \frac{I_{\mathrm{LR,c}} - I_{\mathrm{LR}}}{I_{\mathrm{LR,c}}}, && \text{(Locked-rotor current error)} \\
f_8 &= \frac{T_{\mathrm{LR,c}} - T_{\mathrm{LR}}}{T_{\mathrm{LR,c}}}, && \text{(Locked-rotor torque error)} \\
f_9 &= \frac{T_{\mathrm{max,c}} - T_{\mathrm{max}}}{T_{\mathrm{max,c}}}, && \text{(Maximum torque error)}
\end{aligned}
\tag{5.13}
$$

where the subscript "c" denotes the calculated values from the equivalent circuit model. The expressions for $P_{\mathrm{out,c}}$, $Q_{\mathrm{in,c}}$ etc. in the sub-objective functions f_1 through f_9 are obtained from the steady-state machine equations explained in Appendices A and B.

The objective function is multiplied by a weight vector, which ensures that only the performance characteristics available to the user are included.

$$
\vec{w}_n = \{1, 1, 1, 1, 1, 1, 1, 1, 1\} \tag{5.14}
$$

Typically, the weights corresponding to unavailable performance characteristics are set to zero. For example, if the locked-rotor current (I_{LR}), the locked-rotor torque (T_{LR}), and the maximum torque (T_{max}) are not specified on the nameplate , the weight vector will be $\vec{w}_n = \{1, 1, 1, 1, 1, 1, 0, 0, 0\}$. It includes only the first six performance characteristics f_1

through f_6 in the objective function , while the weights for the remaining f_7, f_8, and f_9 are set to zero. It is not necessary for the weights to be just 1 or 0. If some sub-objectives are deemed to be more critical than the others, the weight vector can be altered accordingly by giving them proportionately larger non-zero values.

Constraints: To ensure that circuit parameter values are always positive, inequality constraints are imposed so that the objective function depends on the absolute value of the decision variables. Additionally, to obtain meaningful parameters, a constraint is imposed requiring that all parameters exceed a minimum value of 0.01 ohms. For the double-cage circuit, additional inequality constraints $R_1 > R_2$ and $X_2 > X_{12}$ are imposed. Constraints are applied using the change of variable method, as explained in Section 3.4.

Applying these constraints not only ensures physically meaningful parameter values but also reduces the variability of the estimated parameters. For the same problem, multiple solutions can exist, leading to variations in parameter values across different initializations. By imposing constraints, the search space is restricted, which in turn limits the variability of the parameters. With these constraints in place, the estimated parameters become more consistent across different initializations, improving the reliability of the optimization results.

5.2.5 Step 5: Estimated Parameters as Output

The optimization algorithm minimizes the objective function and outputs the estimated parameters corresponding to the minimum value of the objective function. This thesis investigated two different and widely used optimization algorithms:

- Nelder-Mead Simplex algorithm [43]
- Genetic Algorithm [58]

Either algorithm may be employed to minimize the objective function. Starting from an initial set of estimates, the optimization process iteratively refines the solution until a convergence criterion—such as reaching a specified tolerance or the maximum number of iterations—is met. Once the process terminates, the resulting estimated parameters are obtained and can be used to simulate the motor in any EMT simulation tool.

This method is applicable to both single-cage and double-cage models. Depending on the choice of decision variables and the objective function equations, the parameters obtained will correspond to either the single-cage or double-cage model.

Both methods have distinct advantages and disadvantages. The choice between them involves balancing computational efficiency and robustness. While the Nelder-Mead method is computationally less expensive, it is more prone to local minima. In contrast, the GA offers a more comprehensive search of the solution space but demands higher computational effort and careful tuning of hyperparameters such as population size and mutation rate.

5.3 Implementation in an External Tool

The parameter estimation process detailed in previous sections is implemented as a custom library component in RSCAD.

The motivation for this work stemmed from a practical challenge faced by users of the RTDS real-time simulator. Many users lacked a straightforward method to incorporate nameplate parameters into the machine model, which required specifying a large number of parameters. To address this need, the methodology for estimating single-cage and double-cage induction motor parameters, as detailed in previous sections, was initially developed as a Python script. However, integrating this functionality directly into the

RTDS simulator was deemed beneficial to enhance accessibility and streamline the parameter specification process.

Hence, to improve practical usability and provide a convenient solution for a broad user base, the developed methodology was integrated into the RSCAD environment—the software interface of RTDS—as a custom library component. RTDS is a widely used real-time digital simulator for power system studies, and this implementation allows users to perform parameter estimation seamlessly within RSCAD. The process involved developing a custom component with a user-friendly interface, enabling efficient and direct parameter estimation. An overview of the implementation process is provided below.

5.3.1 Implementation of the Parameter Estimation Method in RSCAD

The software interface RSCAD of the RTDS includes a "Component Builder" module that enables users to design custom components. Using this module, the parameter estimation routine is implemented as a custom user component. The component estimates the equivalent circuit parameters of three-phase single-cage and double-cage induction motors based on nameplate data and additional catalog information, if available.

5.3.2 Overview of the Implementation Process

The "Component Builder" module in RSCAD provides tools to construct new components, which consist of:

- A component definition file: Defines the graphical user interface (GUI) and input-output structure.

- An associated algorithm file (.c file): Implements the component's functional behavior.

Within the RSCAD interface, a component definition file was created to design an interactive GUI icon. This icon defines the inputs, outputs, and parameters. Features such as data selection options and the choice between single-cage and double-cage models were implemented. Additionally, error checks (e.g., verifying that synchronous speed does not exceed the rated RPM) were integrated to ensure input validity. Figure 5.2 shows the graphical icon of the developed component.

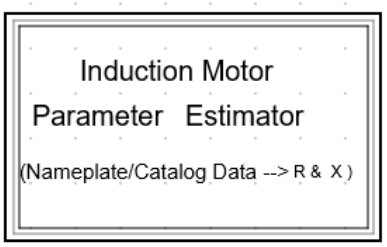

Figure 5.2: Component icon for induction motor parameter estimator in RSCAD

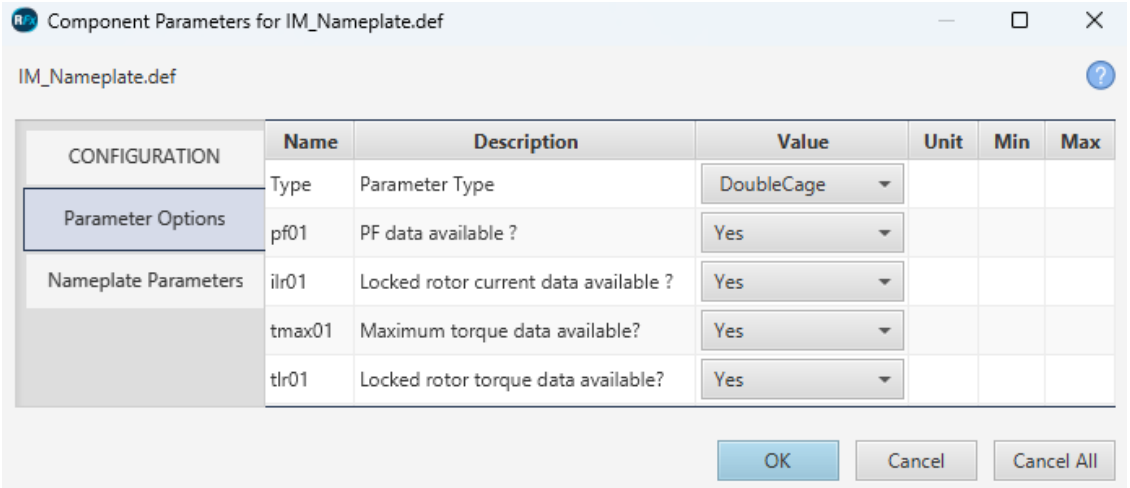

Figure 5.3: Configuration menu of the parameter estimator component in RSCAD

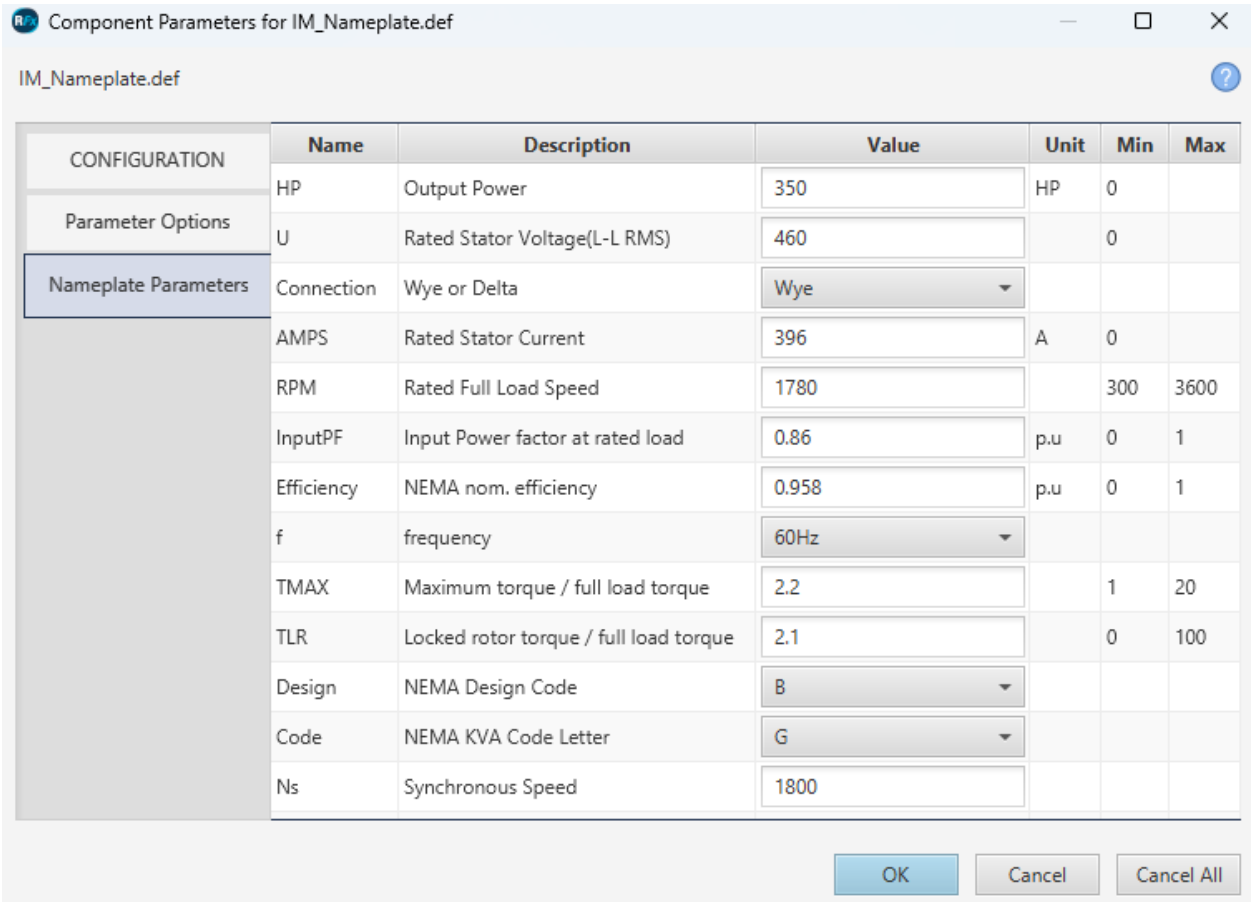

Figure 5.4: Nameplate data input menu for the parameter estimator component in RSCAD

The associated behavior of the component was implemented as a C script, which uses the Nelder-Mead Simplex optimization algorithm due to its ease of programming and minimal computational requirements. After writing the script, it was compiled and linked to the component icon within the RSCAD environment. Upon successful compilation, the component was added to the user library, making it accessible from the main RSCAD menu for direct use.

The developed component takes motor nameplate and catalog data as input, calculates

intermediate performance characteristics, formulates an objective function, and performs optimization to estimate the circuit parameters. The outputs include the parameters of either the single-cage or double-cage model, depending on the selected configuration.

Figures 5.2, 5.3, and 5.4 illustrate the component icon, parameter options menu, and nameplate parameters menu, respectively. The component's functionality was validated by estimating parameters for multiple nameplate scenarios, as discussed in the next chapter.

5.4 Conclusion

This chapter outlined the methodology for estimating induction motor parameters from nameplate data. The proposed approach leverages optimization algorithms to determine parameter values that best match the motor's performance characteristics. The methodology was implemented as a custom library component in RSCAD.

The custom library component is designed to allow users to input motor data directly within RSCAD. It utilizes the C script to perform calculations and optimization, and then outputs the estimated parameters. The component is developed to be flexible and adaptable, accommodating various motor types and input data structures.

Chapter 6

Parameter Estimation from Nameplate and Catalog Data

This chapter validates the parameter estimation methodology presented in Chapter 5 by demonstrating its ability to determine the machine parameters for induction motors based on nameplate and catalog data. The method is applied under two scenarios: when only nameplate data is available, and when additional catalog data is also available.

6.1 Introduction

In Chapter 5, parameter estimation method is developed using two optimization algorithms—the Nelder–Mead Simplex method and a version of Genetic Algorithm. The method is applicable to both single-cage and double-cage equivalent circuit models. This chapter illustrates the application of the methodology by estimating parameters under the following scenarios:

1. **From only nameplate data (single-cage model)**: The method is demonstrated by synthesizing the parameters of a single-cage equivalent circuit using only nameplate specifications.

2. **From nameplate data and additional catalog data (double-cage model)**: A validation is first performed using a machine for which actual double-cage equivalent circuit parameters are known. This step serves as a benchmark to assess the accuracy of the estimation methodology. The validated method is then applied to estimate the parameters for a dataset of 110 machines with power ratings ranging from 0.5 HP to 750 HP, for which actual equivalent circuit parameters are not available.

The parameters obtained from two optimization algorithms—the Nelder-Mead Simplex method and the Genetic Algorithm — are compared to establish their accuracy in getting parameters that match the given data.

6.2 Parameter Estimation from Nameplate Data

In this section, the step-by-step estimation process is illustrated for a motor for which only nameplate data is available. Figure 6.1 shows the nameplate of a design A motor, which—based on its design type—is modeled as a single-cage induction motor.

6.2.1 Step 1: Input Dataset Selection

The input dataset consists of the electrical parameters listed on the motor's nameplate shown in Figure 6.1. The required dataset which will be used as input is tabulated in Table 6.1.

For this nameplate, parameters will be estimated for single-cage equivalent circuit. For any motor, the choice between a single-cage and a double-cage model is determined by the

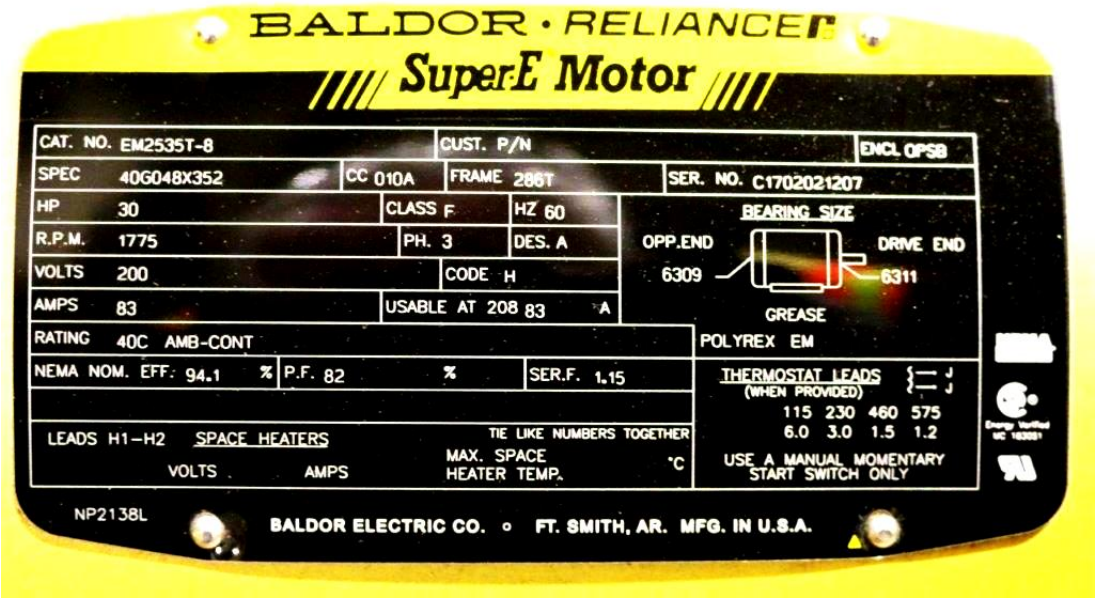

Figure 6.1: Design A (single cage) induction motor nameplate

Table 6.1: Induction motor nameplate data for parameter estimation

Parameter	**Value**	**Unit**
Rated output power (P_{out})	30 (22.37)	$HP(kW)$
Rated line-to-line supply voltage (U)	200	V
Rated speed (RPM)	1775	RPM
Rated current (I_{FL})	83	A
Supply frequency (f)	60	Hz
Rated efficiency (η_{FL})	94.1	%
Rated power factor (PF_{FL})	0.82	pu
NEMA design class of the motor ($DESIGN$)	A	
Locked-rotor kVA per HP code letter ($CODE$)	H	

design type specified on the nameplate. Typically, motors with design types A and D are modeled as single-cage, while types B and C (manufactured as deep-bar or double-cage rotors) are modeled with a double-cage equivalent circuit [34]. If the data for design type is not mentioned then it is recommended to fit it to double-cage, since most of the general purpose motors for industrial applications are of design type B [34].

In this example case, the motor is of design type A and is hence modeled as a single-cage induction motor.

6.2.2 Step 2: Calculation of Additional Performance Characteristics

For the machine with the nameplate in Figure 6.1 and with data as in Table 6.1, additional performance characteristics are calculated. These include the rated torque, rated input reactive power, and locked-rotor current. The calculations use the Equations (5.1) to (5.3) explained in Chapter 5:

- **Rated torque:**

$$T_{FL} = \frac{P_{out}}{\frac{2\pi}{60}\,RPM} \approx 120.33\,\mathrm{Nm}$$

- **Rated input reactive power:**

$$Q_{in} = \frac{P_{out}\sqrt{1 - PF_{FL}^2}}{\eta_{FL}\,PF_{FL}} \approx 16.59\,\mathrm{kVA}$$

- **Locked-rotor current:**

$$I_{LR} = \frac{P_{out} \times LR_{kVA/HP}}{745.7\sqrt{3}\,U} \approx 580.24\,\mathrm{A}$$

In the above, the locked-rotor kVA per HP ($LR_{kVA/HP}$) is determined from the NEMA code letter 'H'. From the Table 4.1, the range for locked-rotor kVA per HP ($LR_{kVA/HP}$) for design 'H' is 6.3–7.1. Hence, the mean value of 6.7 is used to calculate locked-rotor current.

6.2.3 Step 3: Initial Estimate of Parameters

A single-cage equivalent circuit of an induction motor has five circuit parameters, namely R_s, X_s, X_m, R_r, and X_r(see Figure 2.5). For parameter estimation, a relationship between X_s and X_r is imposed as per Table 4.2. The motor nameplate used in this case corresponds to design type A, for which the relation $X_s = X_r$ applies and has been adopted. With this imposed relation, the number of independent decision variables for parameter estimation reduces to four: R_s, X_s, X_m, and R_r.

A set of initial estimate of these decision variables is calculated using simplified assumptions and equations (Equations (5.4) to (5.7) in Chapter 5). The calculated initial estimates, along with the equations used, are summarized in Table 6.2.

Table 6.2: Initial parameter estimates for the single-cage model parameter estimation

Circuit Parameter	**Equation for Initial Estimate**	**Estimated Value (Ω)**
Magnetizing Reactance (X_m)	$X_m \approx \frac{U^2}{Q_{in}}$	2.41
Stator Reactance (X_s)	$X_s \approx 0.07\, X_m$	0.169
Rotor Resistance (R_r)	$R_r \approx \frac{U^2\, s_{FL}}{P_{out}}$	0.025
Stator Resistance (R_s)	$R_s \approx R_r$	0.025

Both optimization algorithms employed in this study require multiple sets of initial

estimates. To generate these additional sets, the single calculated estimate is randomly varied between 50% and 200% of its original value, as explained in Section 5.2. These varied estimates provide diverse starting points for the optimization process, enhancing the robustness of the parameter estimation.

6.2.4 Step 4: Formulation of the Objective Function

An objective function is formulated to minimize the difference between the performance characteristics calculated from the estimated equivalent circuit and the original nameplate data. The function uses the decision variables (for the single-cage model, $\vec{x} = \{R_s, X_s, X_m, R_r\}$) and includes performance characteristics such as power factor, efficiency, rated output power, rated current, rated torque, rated input reactive power, and locked-rotor current.

The objective function is expressed as (see Equations (5.12)–(5.14) in Chapter 5):

$$\text{fitness}(\vec{x}) = \min\left(\frac{1}{N}\sum_{n=1}^{N} w_n f_n^2(\vec{x})\right)$$

Here, each f_n represents the normalized error in a performance characteristic, and the weight vector $\vec{w}_n$ is chosen so that only available data are included (weights corresponding to locked-rotor torque and maximum torque are set to zero since that data is not available for this case).

6.2.5 Step 5: Optimization and Obtaining the Estimated Parameters

Starting from the initial estimates and the formulated objective function, both the Nelder–Mead Simplex and Genetic Algorithm optimizations are run. Table 6.3 lists the parameters obtained from both methods along with their fitness values:

Table 6.3: Single-cage estimated parameters from nameplate data†

Method	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{R}_r$ **(Ω)**	$\mathbf{X}_r$ **(Ω)**	**Fitness**
GA	0.042	0.096	2.660	0.021	0.096	$1.36 \cdot 10^{-5}$
Simplex	0.054	0.094	2.793	0.021	0.094	$5.82 \cdot 10^{-5}$

†Per-unit equivalents provided in Appendix F, Table F.1.

From the Table 6.3, it can be seen that both methods produce comparable values with the objective function fitness value of the order of 10^{-5}. The very small fitness values of the objective function indicates that the simulated performance is able to closely reproduce the original nameplate characteristics. The quality of the estimated parameters is further validated by comparing the simulated performance characteristics in RSCAD with the original nameplate data.

6.2.6 Validation of the Quality of Estimated Parameters

To validate the quality of the estimated parameters, simulations are performed in RSCAD, an EMT tool used for modeling and simulating power systems.

The estimated equivalent circuit parameters obtained from both the methods are input in

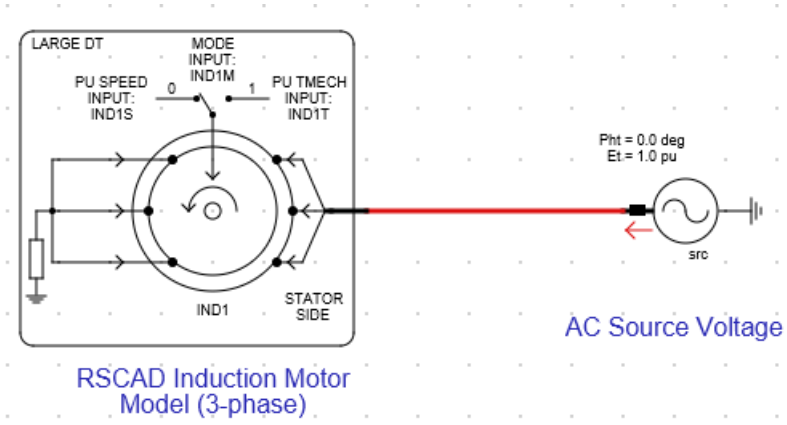

Figure 6.2: Induction motor model set-up in RSCAD for validation of estimated parameters

an induction motor model set-up in RSCAD as shown in Figure 6.2. Simulations are carried out over a speed range from 0 to 1 pu, and the steady-state performance characteristics are recorded at rated speed and under locked- rotor conditions.

The quality of the estimated parameters is then evaluated by measuring how closely the simulated performance characteristics reproduce the manufacturer-specified values. A key metric used for this evaluation is the *absolute percentage error*, which quantifies the absolute difference between the input performance characteristics and those obtained from the estimated parameters, as given by

$$\text{Absolute Percentage Error} = \left| \frac{X_{\text{est}} - X_{\text{manuf}}}{X_{\text{manuf}}} \right| \times 100\% \tag{6.1}$$

where X_{est} represents the performance characteristics obtained from estimated parameters, and X_{manuf} denotes the corresponding manufacturer-specified value.

Table 6.4 shows a comparison between the nameplate data and the simulation outputs along with the absolute percentage error values for both the optimization methods employed.

Table 6.4: Comparison of performance characteristics – nameplate data vs. simulation outputs

Performance Metric	**Unit**	**Actual (Nameplate)**	**Simulated (GA)**	**Simulated (Simplex)**	**Absolute % Error (GA)**	**Absolute % Error (Simplex)**
Rated output power (P_{out})	kW	22.37	22.75	22.53	1.70	0.72
Rated current (I_{FL})	A	83	84.27	82.82	1.53	0.22
Rated efficiency (η_{FL})	%	94.1	94.9	94.0	0.85	0.11
Power factor (PF_{FL})	pu	0.82	0.82	0.83	0.00	1.22
Locked-rotor current (I_{LR})	A	580.24	581.45	579.67	0.21	0.10
Rated torque (T_{FL})	Nm	120.33	122.38	121.24	1.70	0.76
Input reactive power (Q_{in})	kVA	16.59	16.67	15.77	0.48	4.94

From Table 6.4, it can be seen that the simulated performance characteristics from the estimated parameters closely match the original available performance characteristics. This indicates that the estimated parameters can reproduce the original performance characteristics very well. This is true for both methods. GA shows a slightly better performance with absolute percentage errors ranging from 0 - 1.70% as opposed to 0.10 - 4.94% for Simplex. As these parameters reflect the important performance characteristics specified by the manufacturer, they are sufficient for modeling the machine in an EMT program like RSCAD to simulate the motor.

6.3 Parameter Estimation from Nameplate and Catalog Data

In the previous section, the parameter estimation process using only nameplate data was described in detail through an example case. However, some manufacturers provide additional performance data—such as locked-rotor torque and maximum torque—either on the nameplate or in accompanying catalogs. In this section, the previously described methodology is extended to incorporate these additional characteristics, thereby improving the accuracy of parameter estimation, particularly for double-cage equivalent circuit models.

To validate the effectiveness of this extended approach, the method is first applied to a motor for which the actual double-cage equivalent circuit parameters are known. This validation serves as a benchmark for assessing the accuracy of the estimated parameters. Following this, the method is employed to estimate parameters for a catalog of 110 machines for which such equivalent circuit data is not available.

6.3.1 Estimation of Double-Cage Parameters for a Motor with Known Equivalent Circuit Parameters

In this section, the double-cage equivalent circuit parameters are estimated for an induction motor whose equivalent circuit parameters are available. The quality of the estimated parameters is then evaluated by comparing the estimated parameters with the available ones and also with its ability to generate the motor's available performance characteristics.

The step-by-step estimation process is detailed in Chapter 5 and is illustrated in the previous section using a single-cage example.

Input Dataset

The estimation process starts with identifying the input dataset. The available performance characteristic data for the motor, taken from [2] and used as input for parameter synthesis, is listed in Table 6.5. The original data includes both unsaturated and saturated values of the leakage impedances, but here only the unsaturated case has been considered.

Table 6.5: Induction motor catalog data for double-cage parameter estimation

Parameter	**Value**	**Unit**
Rated output power (P_{out})	102.7	kW
Rated line-to-line supply voltage (U)	400	V
Rated speed (RPM)	1770	RPM
Rated current (I_{FL})	180	A
Supply frequency (f)	60	Hz
Rated efficiency (η_{FL})	94	%
Rated input reactive power (Q_{in})	59.6	$kVAR$
Rated power factor (PF_{FL})	0.88	pu
Locked-rotor current (I_{LR})	1021	A
Rated torque (T_{FL})	553.8	Nm
Locked-rotor torque (T_{LR})	681.2	Nm
Maximum torque (T_{max})	1451	Nm

Objective Function

After selecting the input dataset, the objective function defined by Equations (5.12) - (5.14) is modified to incorporate the additional available performance characteristics. For the motor,

all nine performance characteristics are available and hence are included in the objective function by assigning a weight of 1 to each metric in Equation (5.14). The additional metrics, such as locked-rotor torque and maximum torque, are thereby integrated into the optimization process, ensuring that the overall fitness measure reflects a more comprehensive set of performance criteria.

Initial Estimate of Parameters

For the double-cage model, the decision variable vector is defined as

$$\vec{x} = \{R_s,\, X_s,\, X_m,\, X_{12},\, R_1,\, R_2,\, X_2\}$$

Similar to the single-cage case presented earlier, the initial estimates for these parameters are calculated using the approximate relationships in (5.4) to (5.11) and is tabulated as Table 6.6.

Once the initial estimates are obtained, the optimization routine is run using both the Nelder–Mead Simplex and the Genetic Algorithm methods. The optimization process iteratively adjusts the initial guesses so that the simulated performance characteristics—across all nine measures—most closely match the manufacturer and catalog data. The estimated parameters are obtained as output when the predefined exit criteria are satisfied. The estimated parameters using both methods along with the original available parameters are listed in Table 6.7.

Table 6.6: Initial parameter estimates for double-cage model parameter estimation

Circuit Parameter	**Equation for Initial Estimate**	**Estimated Value (Ω)**
Magnetizing Reactance (X_m)	$X_m \approx \frac{U^2}{Q_{in}}$	2.685
Stator Reactance (X_s)	$X_s \approx 0.07\, X_m$	0.188
Inner-Cage Rotor Resistance (R_2)	$R_2 \approx \frac{U^2\, s_{FL}}{P_{out}}$	0.026
Stator Resistance (R_s)	$R_s \approx R_2$	0.026
Outer-Cage Rotor Resistance (R_1)	$R_1 \approx 2R_2$	0.052
Inner-Cage Rotor Reactance (X_2)	$X_2 \approx 2X_s$	0.376
Mutual Rotor Reactance (X_{12})	$X_{12} \approx X_s$	0.188

Table 6.7: Estimated parameter comparison of double cage machine[†]

Parameters	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
Actual	0.050	0.102	3.474	0.077	0.086	0.029	0.086	
GA	0.040	0.104	3.607	0.092	0.044	0.044	0.180	$1.95 \cdot 10^{-4}$
Simplex	0.050	0.121	3.452	0.059	0.080	0.029	0.084	$1.41 \cdot 10^{-8}$

[†]Per-unit equivalents in Appendix F, Table F.2

Evaluation of the Quality of the Estimated Parameters

The quality of the estimated double-cage parameters is evaluated by comparing the estimated parameters with the available machine parameters, as well as how closely the simulated performance characteristics reproduce the manufacturer-specified values. Similar to the

single-cage estimation, *absolute percentage error* metric, defined by Equation (6.1) is used. Table 6.7 shows the estimated parameters from both algorithms along with the actual available machine parameters. It can be seen that both GA and Simplex estimated parameters are very close to the actual parameters.

Table 6.8: Comparison of performance characteristics –actual vs. simulation outputs

Performance Metric	**Unit**	**Actual (Catalog)**	**Simulated (GA)**	**Simulated (Simplex)**	**Absolute % Error (GA)**	**Absolute % Error (Simplex)**
Rated current (I_{FL})	A	180.00	178.43	181.28	0.87	0.71
Rated torque (T_{FL})	Nm	553.80	554.39	559.37	0.11	1.01
Rated output power (P_{out})	kW	102.70	102.76	103.69	0.06	0.96
Power factor (PF_{FL})	pu	0.88	0.88	0.88	0.21	0.08
Rated efficiency (η_{FL})	%	0.94	0.95	0.94	0.96	0.03
Input reactive power (Q_{in})	kVA	59.60	59.57	59.94	0.05	0.57
Locked-rotor current (I_{LR})	A	1021.00	1065.58	1022.34	4.37	0.13
Locked-rotor torque (T_{LR})	Nm	681.20	679.88	682.86	0.19	0.24
Maximum torque (T_{max})	Nm	1451.00	1444.16	1450.39	0.47	0.04

Also, Table 6.8 shows a comparison between the nameplate data and the simulation outputs along with the absolute percentage error values for both optimization methods. From Table 6.8, it can be seen that the simulated performance characteristics from the estimated parameters closely align with the manufacturer and catalog data. This indicates that the estimated parameters can reproduce the original performance characteristics very well. This is true for both methods. The Simplex method shows slightly better performance, with

absolute percentage errors ranging from 0.03% to 1.01%, compared to 0.05% to 4.37% for GA. Even with GA, the errors for most quantities remain below 1%, with the only exception being the locked-rotor current, which shows an error of 4.37%. The applicability of the method is further validated in Subsection 6.3.2 by estimating the double-cage equivalent circuit parameters for a set of 110 motors taken from [59] whose circuit parameters are not known.

6.3.2 Estimation of Double-Cage Parameters for a Catalog of 110 Induction Motors

To evaluate the scalability and practical applicability of the proposed estimation methodology, it is applied to a larger and diverse dataset comprising machines of various ratings. This extensive collection provides a realistic test environment and reflects real-world scenarios, where detailed equivalent circuit parameters are typically unavailable.

Unlike the example case in the previous subsection, where the actual circuit parameters were known and served as a benchmark for validation, the true parameters for these motors are unknown. Therefore, the effectiveness of the estimation is assessed solely based on how well the simulated performance characteristics match those provided in the manufacturer catalog.

Input Dataset for the Catalog Data

For this extended estimation, a catalog dataset consisting of 110 induction motors sourced from [59] is utilized. These motors span a wide power rating range from 0.5 HP (0.37 kW) to 750 HP (552 kW). The detailed data for these motors is provided in Table C.1 of Appendix C. Figure 6.3 displays the distribution of these machines across various power rating ranges.

The dataset covers a broad spectrum of motor sizes and includes a significant number of motors in each size range. A diverse representation of motor ratings provides a robust test for parameter estimation.

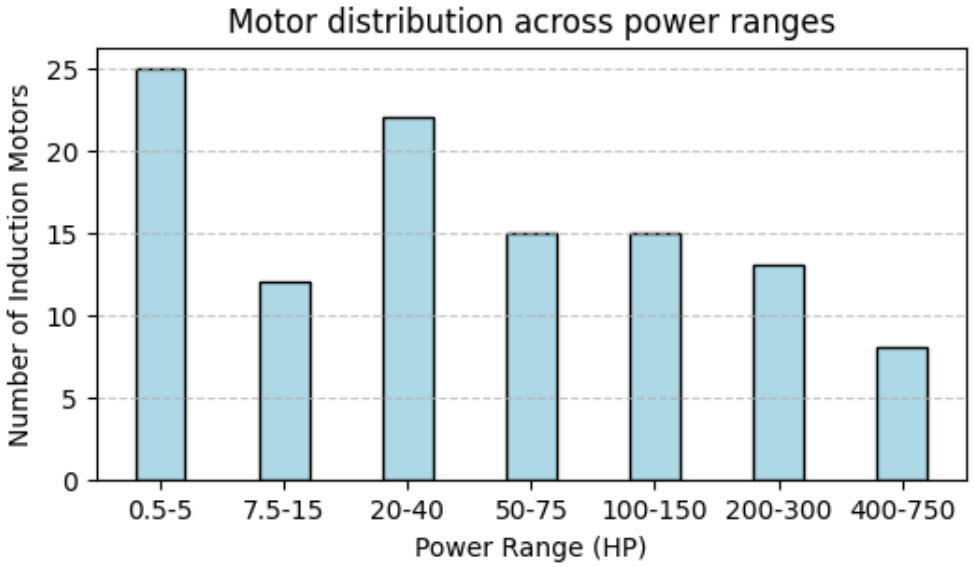

Figure 6.3: Distribution of analyzed motors as a function of rated power

In addition to the standard nameplate data (which includes parameters like rated power, voltage, speed, rated current, etc.), the catalog provides the following additional performance characteristics:

- Locked-rotor current (I_{LR})
- Locked-rotor torque (T_{LR})
- Maximum torque (T_{max})

Since the catalog does not specify the design type, the double-cage equivalent circuit model is adopted. This choice is based on the fact that most general-purpose industrial motors are manufactured with double-cage or deep-bar rotors [34].

Objective Function

The estimation problem is formulated as the objective function defined by Equations (5.12) - (5.14). The catalog dataset contains machine performance characteristics corresponding to all the nine sub-objectives of the objective function. Hence, all nine performance metrics are included in the objective function by assigning a weight of 1 to each metric in Equation (5.14).

Initial Estimate Calculation and Optimization Run

Similar to the double-cage example case presented earlier, the initial estimates for these parameters are calculated using the the approximate relationships in (5.4) to (5.11). This calculation is performed for each of the 110 motors in the dataset.

Once the initial estimates are obtained, the optimization routine is run using both the Nelder–Mead Simplex and the Genetic Algorithm methods and the estimated parameters are obtained as output when the predefined exit criteria are satisfied.

Evaluation of the Quality of the Estimated Parameters

The quality of the estimated double-cage parameters is evaluated by comparing the simulated performance characteristics with the catalog values. As in the earlier case study, the absolute percentage error—defined in Equation (6.1)—is used as the primary evaluation metric.

The optimized best-fit equivalent circuit parameter values for each of the 110 motors is given in Appendix D (obtained with GA) and Appendix E (obtained with Simplex). Figures 6.4 and 6.5 display the plot of absolute percentage errors in various input performance characteristics(i.e., objective function sub-objectives) obtained from the estimated parameters using both the GA and the Simplex method across 110 motors.

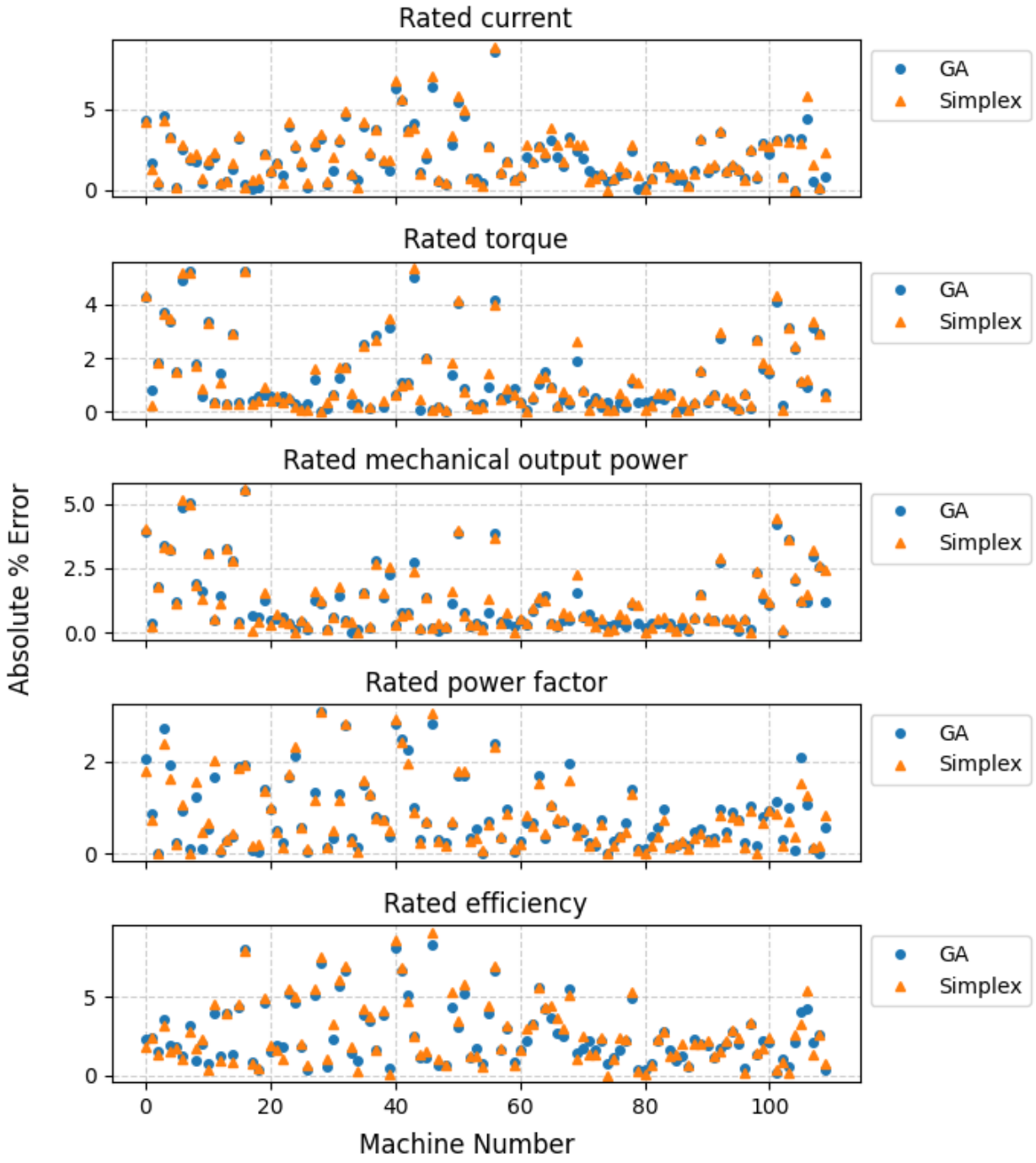

Figure 6.4: Comparison of percentage error in input performance characteristics (f1–f5) from estimated parameters across 110 motors

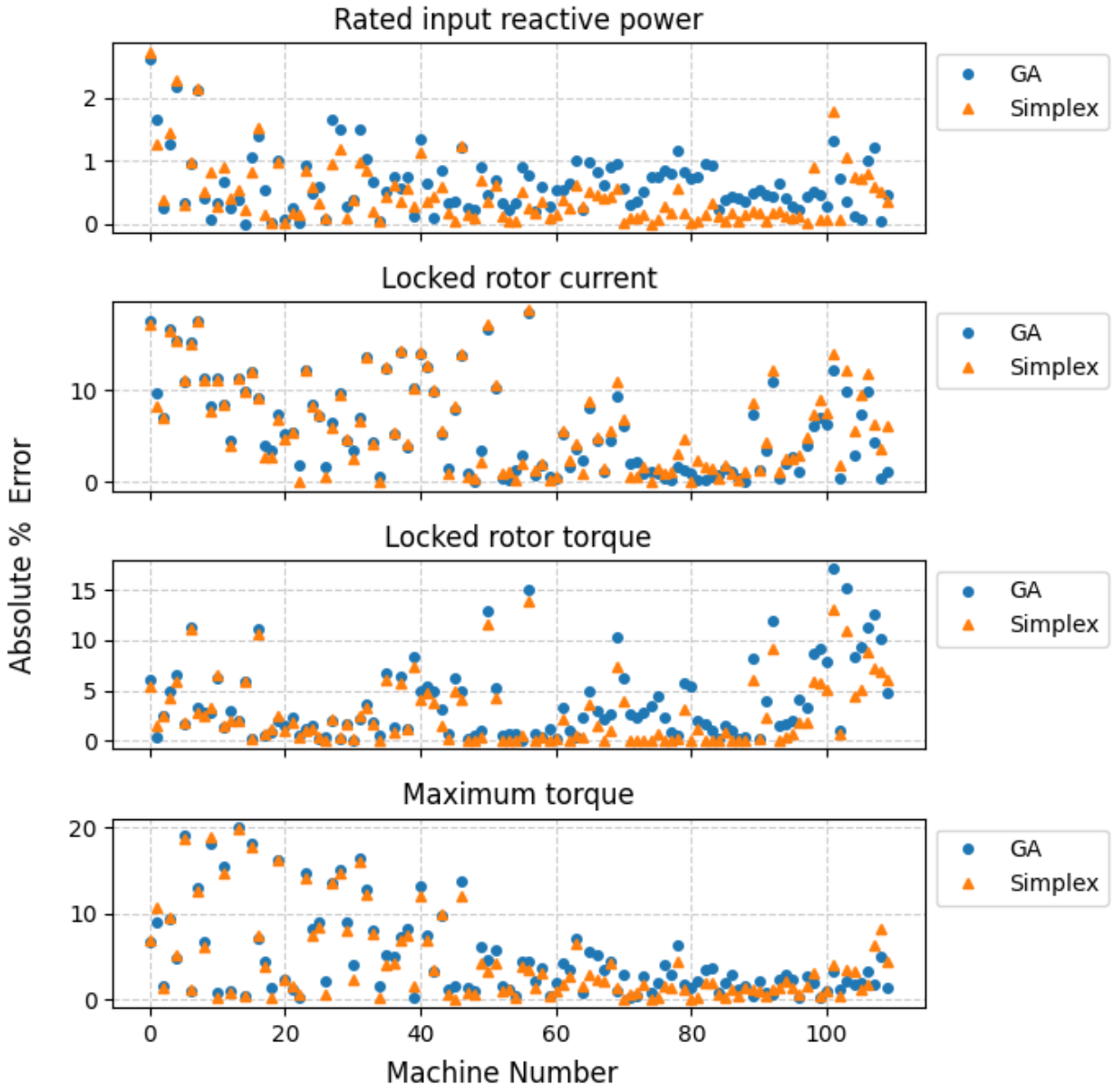

Figure 6.5: Comparison of percentage error in input performance characteristics (f6–f9) from estimated parameters across 110 motors

From these figures, it is observed that the absolute percentage error between the estimated and the input performance characteristics remains within 10% for rated quantities

such as rated current, rated torque, and output power. However, for performance characteristics under conditions other than rated operation, such as locked-rotor torque, locked-rotor current, and maximum torque, the absolute percentage error is within 20%. This higher deviation can be attributed to the fact that manufacturers specify non-rated performance characteristics with a higher tolerance, often providing values as an average across similar machines rather than exact specifications for that particular machine.

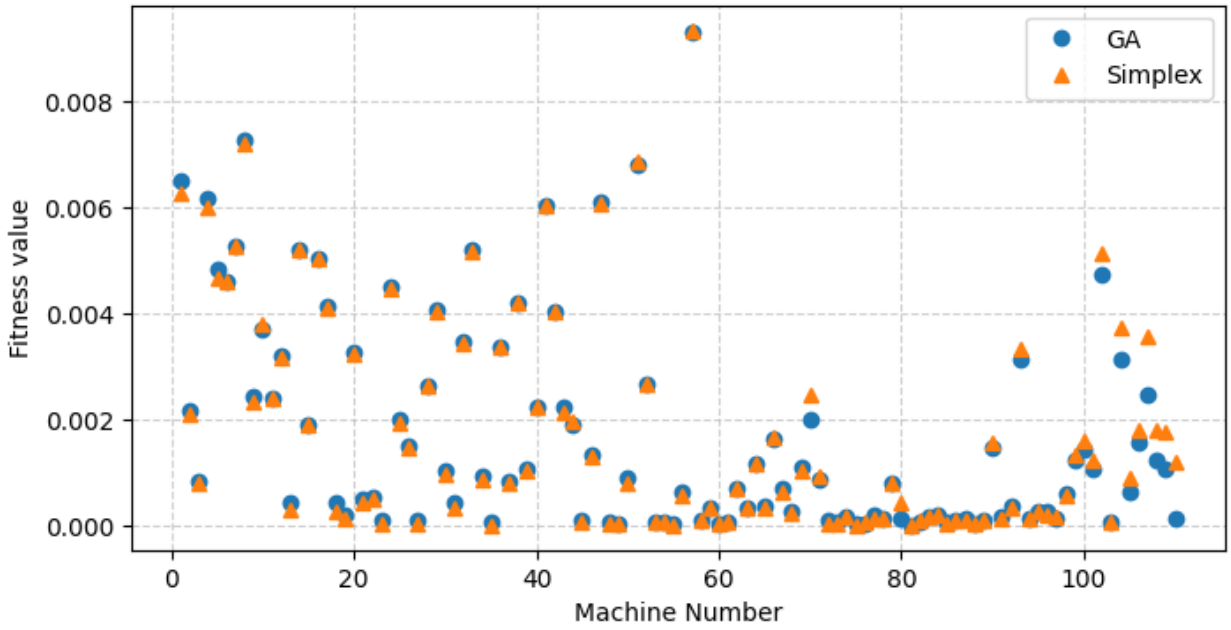

Figure 6.6: Comparison of fitness values of estimated parameters across 110 motors

The overall quality of the estimated parameters is further assessed using the *fitness value* of the objective function, defined as the normalized mean square error between the manufacturer-specified and estimated performance characteristics and given by Equation (5.12). A lower fitness value indicates a closer match between the actual and simulated machine performance and hence higher quality of the estimated parameters. Figure 6.6 presents the fitness values obtained for both GA and the Nelder–Mead Simplex method

across all 110 motors.

As seen in Figure 6.6, the fitness values are of the order of 10^{-3} and also the GA method has comparatively lesser fitness value indicating better quality of estimated parameters. This trend is consistent with the absolute percentage error plots in Figures 6.4 and 6.5. Also, the order of fitness value can be deemed acceptable as the estimated parameters are able to reproduce the original machine characteristics within reasonable tolerance.

Overall, while both methods yield results within acceptable limits, the GA method is somewhat superior as it provides a slightly better match to the manufacturer-published performance characteristics. These findings suggest that the estimated parameters obtained by both methods are reliable and suitable for EMT simulations.

Validation through Transient Simulation on the RTDS simulator

The previous optimization was based on comparing steady-state values. As the aim is to determine the parameters for the full transient model, EMT simulations are performed using the estimated parameters in the RSCAD induction motor model (see Figure 6.2). The simulation outputs are compared against the manufacturer-specified performance characteristics. A closer match between the simulated and specified values indicates better quality of estimated parameters.

For illustration, Figures 6.7 and 6.8 present the torque–speed and current–speed curves for two representative cases: the best-fitting machine and the worst-fitting machine. These plots also include the manufacturer-specified data points - rated, locked rotor, and maximum torque, as well as rated and locked-rotor current. Also, corresponding estimated parameters are presented in Tables 6.9 and 6.10, respectively.

For the best- and worst- fit plots in Figures 6.7 and 6.8, an initial transient is observed in

the torque characteristics due to oscillations caused by the DC component during the starting phase. However, the average locked-rotor torque closely matches the manufacturer-specified value. The rated torque and current values exhibit an excellent match, nearly identical to those specified by the manufacturer, and align well with the estimated parameter plots for both the best- and worst-fitting machines.

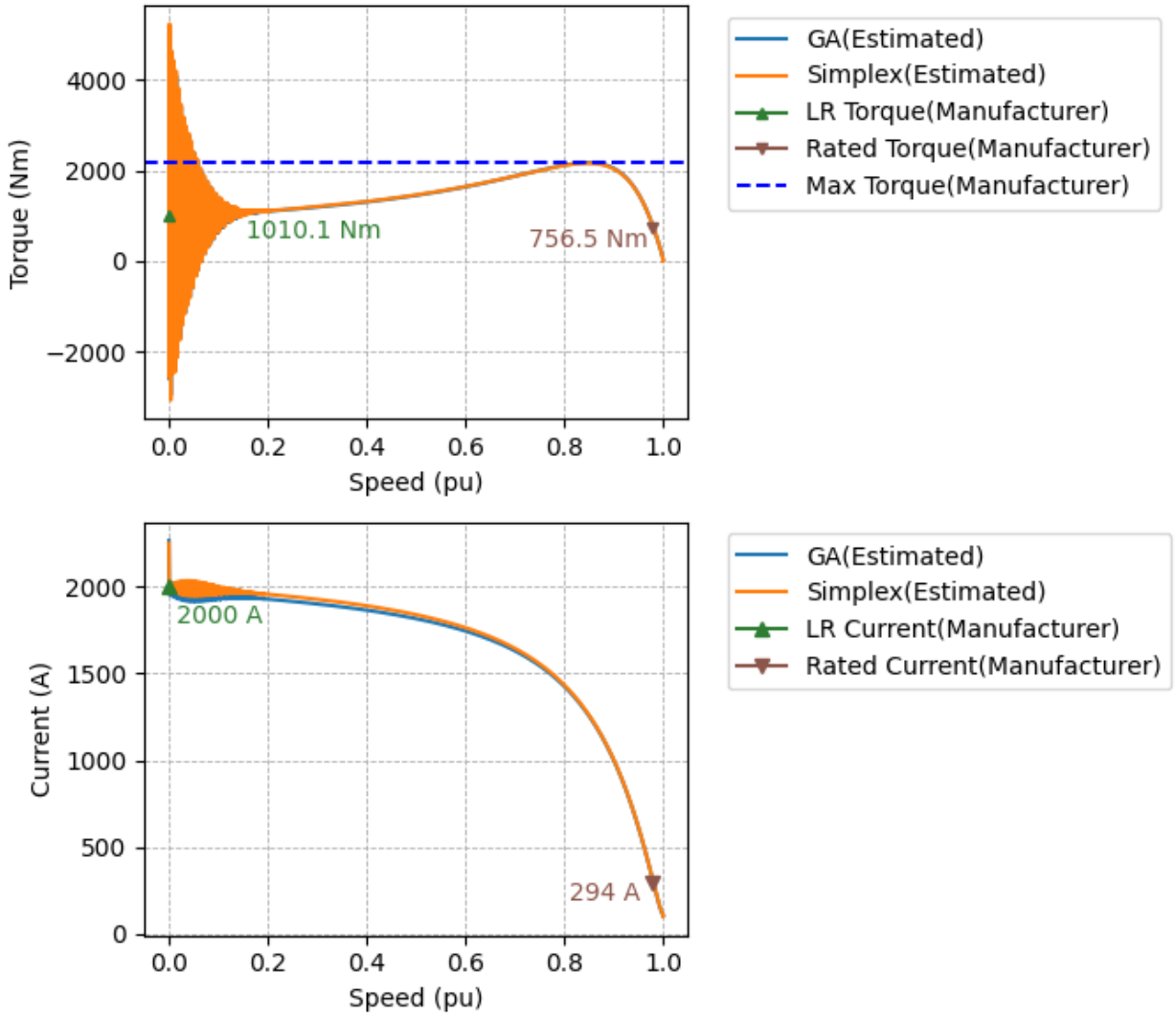

Figure 6.7: Comparison of torque and current characteristics for the best-fitting machine

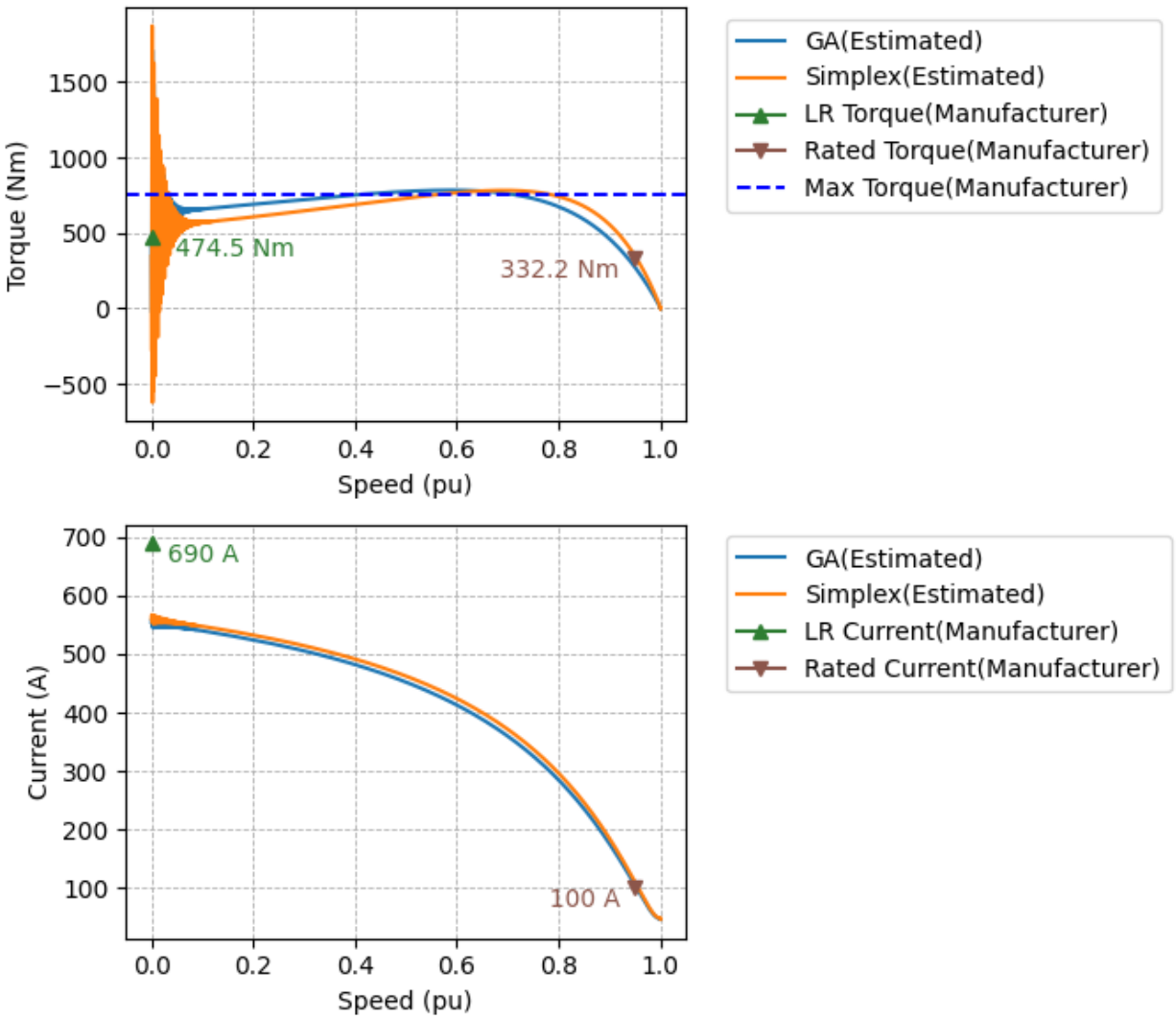

Figure 6.8: Comparison of torque and current characteristics for the worst-fitting machine

Table 6.9: Double-cage estimated parameters for the best-fitting machine†

Method	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
GA	0.026	0.034	1.269	0.014	0.021	0.015	0.018	$2.79 \cdot 10^{-5}$
Simplex	0.028	0.035	1.234	0.010	0.028	0.013	0.014	$1.86 \cdot 10^{-7}$

†Per-unit equivalents in Appendix F, Table F.3

Table 6.10: Double-cage estimated parameters for the worst-fitting machine†

Method	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
GA	0.130	0.118	2.641	0.010	0.157	0.084	0.010	$9.41 \cdot 10^{-3}$
Simplex	0.135	0.104	2.580	0.010	0.839	0.058	0.010	$9.34 \cdot 10^{-3}$

†Per-unit equivalents in Appendix F, Table F.4

For the worst-fitting machine, slight discrepancies are observed in the maximum torque and locked-rotor current values. This mismatch is likely due to data variability, as manufacturers typically provide averaged values for a given machine type rather than exact specifications for an individual unit. Furthermore, according to IEC and NEMA standards, these values are subject to a tolerance of up to ±15% [55, 56].

6.4 Summary of Observations and Comparative Analysis

This section summarizes the observations and comparative analysis of the parameter estimation methodology as demonstrated for both single-cage and double-cage models using

two optimization algorithms: the Nelder–Mead Simplex method and a version of the Genetic Algorithm.

It was observed that both algorithms perform well under the two scenarios—using only nameplate data and incorporating additional catalog data. The estimated parameters reproduced the manufacturer-specified performance characteristics within acceptable tolerance levels. In particular:

- In most cases, the method using Genetic Algorithm achieved slightly better fitness values overall. Its probabilistic nature allowed it to explore a wider search space, thereby yielding parameter estimates that more closely matched the target performance metrics.

- However, the GA is computationally more expensive due to its inherent stochastic processes and the requirement for a larger number of function evaluations. This increased computational demand should be considered when selecting the optimization method for real-time or large-scale applications.

- In contrast, the Nelder–Mead Simplex method is easier to program and is less computationally intensive. Although it may be more susceptible to becoming trapped in local minima, its efficiency makes it attractive for scenarios where computational resources are limited or when a faster implementation is desired.

- The closeness of the parameter sets estimated by both methods serves as a validation of the robustness of the proposed methodology. The reproducibility of the original performance characteristics from the estimated parameters supports their suitability for both steady-state and transient studies in systems involving induction motors.

Overall, while both algorithms yield comparable results, the choice between them may depend on the specific application requirements. GA offers marginally higher accuracy at the cost of increased computational effort, while the Simplex method provides a simpler and more efficient alternative.

6.5 Conclusion

This chapter demonstrated the practical application of the proposed parameter estimation methodology which can use either of the two optimization algorithms-Nelder-Mead Simplex and a version of GA, under two scenarios: using only nameplate data and incorporating additional catalog data. The methodology was validated using an actual nameplate example for the single-cage model, and a motor with known equivalent circuit parameters along with a dataset of 110 induction motors for the double-cage model. The estimated parameters using two optimization methods were compared and analyzed for their usefulness.

The results showed that the estimated parameters from both algorithms reproduced the manufacturer-specified performance characteristics of the motors within acceptable tolerance. The closeness of the parameter set estimated by two different optimization algorithms, justifies the robustness of the method. The reproducibility of the original performance characteristics within acceptable limits from estimated parameters establishes that the parameter sets are suitable for use in both steady-state and transient studies of systems involving induction motors.

Chapter 7

Contributions, Conclusions, and Recommendations for Future Work

This chapter outlines the contributions of this thesis, the conclusions drawn from the results, and the recommendations for future research directions.

7.1 Contributions

The primary contributions of this thesis are as follows:

- Development of a parameter estimation approach for induction motors (applicable to both single-cage and double-cage equivalent circuit models) using only manufacturer-published data. The approach utilizes two optimization algorithms—the Nelder-Mead Simplex method and the Genetic Algorithm—to determine best-fit parameters suitable for power system simulations, effectively addressing the requirements of such studies.

- Integration of the method as a handy tool in RSCAD, designed with a flexible data input structure that accommodates both minimal nameplate data and additional catalog information. This adaptability ensures compatibility with varying motor ratings and manufacturer standards.

7.2 Conclusions

The findings of this study lead to the following conclusions:

- **Feasibility of Parameter Estimation:** By leveraging optimization techniques, the estimation of best-fit induction motor parameters for simulation purposes is feasible using only the performance data published by the manufacturer. This approach is quick and inexpensive as it avoids additional physical testing; thereby making it highly practical for applications where only nameplate and catalog data are available.

- **Performance Comparison of Optimization Methods:** Both the Nelder-Mead Simplex method and the Genetic Algorithm have proven effective for estimating induction motor parameters. The Simplex method is characterized by its simplicity and computational efficiency, often converging quickly when provided with a reasonable initial guess. In contrast, the GA - owing to its stochastic nature—is more robust when handling the inherent inconsistencies in manufacturer data. Such data typically include tolerances and ranges for quantities like locked-rotor current and maximum torque, meaning that the input values are not exact for a given machine. When faced with noisy data, the GA's superior global search capability allows it to better accommodate these uncertainties and match the manufacturer-published data more closely.

The results indicate that both methods yield estimated parameters that align with manufacturer data within an approximate error margin of 20%, though this margin varies depending on the quality and type of input data. Notably, when high-tolerance data—such as locked-rotor current, locked-rotor torque, and maximum torque—is excluded from the matching process and only characteristics at rated conditions are considered, the simulated performance characteristics match the manufacturer data with an error margin of within 10%.

- **Multiplicity of Parameter Sets:** Multiple sets of parameters can yield nearly identical performance characteristics. Imposing constraints, such as maintaining expected relationships between stator and rotor reactances in single-cage estimation and imposing inequality constraints in rotor resistances and reactances in double-cage models, helps in reducing variability and improving the consistency of estimated parameters.

- **Applicability of the Estimated Parameter Sets:** Manufacturer data, including efficiency, power factor, and locked-rotor performance characteristics, often incorporate tolerances or represent average values for a given motor type rather than precise, tested values for an individual machine. As a result, the estimated parameters should not be used in applications requiring very high precision, such as control system design, which involves continuous parameter tuning to account for variations caused by temperature changes or other factors. However, they are sufficiently accurate for power system simulations and general performance analysis, where approximate parameter values are acceptable for steady-state and transient studies.

7.3 Publications Resulting from This Research

The key findings and methodology developed in this thesis have been disseminated through the following technical paper:

- V. Kumari, J. S. Acosta, A. M. Gole, and A. B. Dehkordi, "*Using Optimization Algorithms to Estimate Model Parameters of Induction Motors Based on Manufacturer Data for Simulation Purposes*" presented at the CIGRE Canada Conference & Exhibition, Winnipeg, Canada, October 28-31, 2024.

This paper presents the core concepts and results of the thesis to the broader power systems community, highlighting the practical applicability of the proposed parameter estimation technique in modeling and simulation contexts.

7.4 Recommendations for Future Work

Future research can further build upon this study by addressing the following directions:

- Investigating the inclusion of magnetic saturation effects in the estimation process, particularly when saturation-related data (e.g., no-load current vs. voltage characteristics) is available, to enhance the accuracy of the estimated parameters.
- Extending the study to other types of motors, such as synchronous machines, to assess the broader applicability of the developed methods.

In conclusion, this study develops a robust method for estimating induction motor parameters solely from manufacturer data, enabling the simulation of motors with

unavailable parameters. Future research should focus on addressing current limitations and extending the methodology to other machine types and broader applications.

References

[1] I. Boldea, *Induction Machines Handbook: Steady State Modeling and Performance*, ser. Electric power engineering. CRC Press, 2020, vol. 1.

[2] P. Kundur, *Power System Stability and Control*. New York: McGraw-Hill, 1994.

[3] D. Lindenmeyer, H. W. Dommel, A. Moshref, and P. Kundur, "An induction motor parameter estimation method," *Electric Power Systems Research*, vol. 23, pp. 251–262, 2001.

[4] P. L. Alger and J. H. Wray, "Double and triple squirrel cages for polyphase induction motors," *AIEE Trans*, vol. 72, no. 4, pp. 637–645, 1953.

[5] C. H. Lee, "A design method for double squirrel-cage induction motors," *AIEE Trans, Part 3, Power apparatus and systems*, vol. 72, no. 4, pp. 630–636, 1953.

[6] J. Goodman, "A design method for double squirrel-cage induction motors," *AIEE Trans, Part 3, Power apparatus and systems*, vol. 72, no. 4, pp. 645–650, 1953.

[7] M. O. Sonnaillon, G. Bisheimer, C. D. Angelo, and G. O. García, "Automatic induction machine parameters measurement using standstill frequency-domain tests," *IET Electric Power Applications*, vol. 1, no. 5, pp. 833–838, 2007.

[8] L. Monjo, H. Kojooyan-Jafari, F. Corcoles, and J. Pedra, "Squirrel-cage induction motor parameter estimation using a variable frequency test," *IEEE transactions on energy conversion*, vol. 30, no. 2, pp. 550–557, 2015.

[9] A. Lalami, R. Wamkeue, I. Kamwa, M. Saad, and J. Beaudoin, "Unscented kalman filter for non-linear estimation of induction machine parameters," *IET Electr. Power Appl.*, vol. 6, pp. 611–620, 2012.

[10] P. Castaldi, W. Geri, M. Montanari, and A. Tilli, "A new adaptive approach for on-line parameter and state estimation of induction motors," *Control Eng. Pract.*, vol. 13, pp. 81–94, 2005.

[11] S.-I. Moon and A. Keyhani, "Estimation of induction machine parameters from standstill time-domain data," *IEEE Transactions on Industry Applications*, vol. 30, no. 6, pp. 1609–1615, 1994.

[12] P. Ju, E. Handschin, Z. Wei, and U. Schluecking, "Sequential parameter estimation of a simplified induction motor load model," *IEEE Transactions on Energy Conversion*, vol. 11, no. 1, pp. 319–324, 1996.

[13] J. de Kock, F. van der Merwe, and H. Vermeulen, "Induction machine parameter estimation through an output error technique," *IEEE Transactions on Energy Conversion*, vol. 9, no. 1, pp. 69–76, 1994.

[14] *IEEE Standard Test Procedure for Polyphase Induction Motors and Generators*, Institute of Electrical and Electronics Engineers (IEEE) Std., 2017.

[15] B. J. Chalmers and A. S. Mulki, "Design synthesis of double-cage induction motors," *Proceedings of the IEE*, vol. 117, no. 7, pp. 519–527, 1970.

[16] S. Waters and R. Willoughby, "Modelling induction motors for system studies," *IEEE Transactions on Industrial Applications*, vol. IA-19, no. 5, pp. 875–878, 1983.

[17] G. J. Rogers and D. Shirmohammadi, "Induction machine modelling for electromagnetic transient program," *IEEE Transactions on Energy Conversion*, vol. EC-2, no. 4, pp. 622–628, 1987.

[18] M. H. Haque, "Determination of NEMA design induction motor parameters from manufacturer data," *IEEE Transactions on Energy Conversion*, vol. 23, no. 4, pp. 997–1004, 2008.

[19] R. Natarajan and V. K. Misra, "Parameter estimation of induction motors using a spreadsheet program on a personal computer," *Electric Power Systems Research*, vol. 16, pp. 157–164, 1989.

[20] S. Ansuj, F. Shokooh, and R. Schinzinger, "Parameter estimation for induction machines based on sensitivity analysis," *IEEE Transactions on Industry Applications*, vol. 25, no. 6, pp. 1035–1040, 1989.

[21] B. Johnson and J. Willis, "Tailoring induction motor analytical models to fit known motor performance characteristics and satisfy particular study needs," *IEEE Transactions on Power Systems*, vol. 6, no. 3, pp. 965–969, 1991.

[22] P. Pillay, R. Nolan, and T. Haque, "Application of genetic algorithms to motor parameter determination for transient torque calculations," *IEEE transactions on industry applications*, vol. 33, no. 5, pp. 1273–1282, 1997.

[23] J. Pedra and F. Córcoles, "Estimation of induction motor double-cage model parameters from manufacturer data," *IEEE Transactions on Energy Conversion*, vol. 19, no. 2, pp. 310–317, jun 2004.

[24] J. Pedra, "On the determination of induction motor parameters from manufacturer data for electromagnetic transient programs," *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1709–1718, 2008.

[25] M. Gomez-Gonzalez, F. Jurado, and I. Pérez, "Shuffled frog-leaping algorithm for parameter estimation of a double-cage asynchronous machine," *IET Electr. Power Appl.*, vol. 6, pp. 484–490, 2012.

[26] K. Lee, S. Frank, P. Sen, L. Polese, M. Alahmad, and C. Waters, "Estimation of induction motor equivalent circuit parameters from nameplate data," in *Proceedings of the 2012 North American Power Symposium (NAPS)*, Champaign, IL, USA, 2012, pp. 1–6.

[27] J. Guimaraes, J. Bernardes, A. Hermeto, and E. Bortoni, "Parameter determination of asynchronous machines from manufacturer data sheet," *IEEE Trans. Energy Convers.*, vol. 29, pp. 689–697, 2014.

[28] S. A. Al-Jufout, W. H. Al-Rousan, and C. Wang, "Optimization of induction motor equivalent circuit parameter estimation based on manufacturer's data," *Energies*, vol. 11, no. 7, 2018.

[29] M. Ćalasan, M. Micev, Z. M. Ali, A. F. Zobaa, and S. H. Aleem, "Parameter estimation of induction machine single-cage and double-cage models using a hybrid simulated annealing-evaporation rate water cycle algorithm," *Mathematics*, vol. 8, no. 6, jun 2020.

[30] I. Perez, M. Gomez-Gonzalez, and F. Jurado, "Estimation of induction motor parameters using shuffled frog-leaping algorithm," *Electr. Eng.*, vol. 95, pp. 267–275, 2013.

[31] Z. Danin, A. Sharma, M. Averbukh, and A. Meher, "Improved Moth Flame Optimization Approach for Parameter Estimation of Induction Motor," *Energies*, vol. 15, no. 23, dec 2022.

[32] G. J. Rogers, J. DiMantio, and R. T. H. Allen, "An aggregate induction motor model for industrial plants," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-103, no. 4, pp. 683–690, April 1984.

[33] A. Fitzgerald, C. Kingsley, and S. Umans, *Electric Machinery*, ser. Electrical Engineering Series. McGraw-Hill Companies,Incorporated, 2003.

[34] S. Chapman, *Electric Machinery Fundamentals*. McGraw-Hill Companies, 2005.

[35] "Induction motor basics," 2024. [Online]. Available: https://www.mangoengineer.in/2024/12/induction-motor-basics

[36] "Construction of three-phase induction motor." [Online]. Available: https://www.tutorialspoint.com/electrical_machines/electrical_machines_construction_of_threephase_induction_motor

[37] S. Filizadeh, *Electric Machines and Drives: Principles, Control, Modeling, and Simulation*, 1st ed. CRC Press, 2013.

[38] D. S. Brereton, D. G. Lewis, and C. C. Young, "Representation of induction-motor loads during power-system stability studies," *AIEE Trans*, vol. 76, no. 3, pp. 451–460, 1957.

[39] V. Torczon, "Multi-directional search: A direct search algorithm for parallel machines," Ph.D. dissertation, Rice University, Houston, TX, 1989.

[40] M. Wright, "Direct search methods: once scorned, now respectable," in *Numerical Analysis 1995*, ser. Pitman Research Notes in Mathematics, D. Griffiths and G. Watson, Eds., vol. 344. Boca Raton, FL: CRC Press, 1996, pp. 191–208.

[41] L. Han and M. Neumann, "Effect of dimensionality on the nelder–mead simplex method," *Optim. Methods Softw.*, vol. 21, no. 1, pp. 1–16, 2006.

[42] I. Fajfar, Bűrmen, and J. Puhan, "The nelder–mead simplex algorithm with perturbed centroid for high-dimensional function optimization," *Optimization letters*, vol. 13, no. 5, pp. 1011–1025, 2019.

[43] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[44] W. Spendley, G. Hext, and F. Himsworth, "Sequential application of simplex designs in optimisation and evolutionary operation," *Technometrics*, vol. 4, no. 4, pp. 441–, 1962.

[45] J. Del Ser, E. Osaba, D. Molina, X. S. Yang, S. Salcedo-Sanz, D. Camacho, S. Das, P. N. Suganthan, C. A. Coello Coello, and F. Herrera, "Bio-inspired computation: Where we stand and what's next," *Swarm and Evolutionary Computation*, vol. 48, no. December 2018, pp. 220–250, 2019.

[46] P. A. Vikhar, "Evolutionary algorithms: A critical review and its future prospects," *Proc. - Int. Conf. Glob. Trends Signal Process. Inf. Comput. Commun. ICGTSPICC 2016*, pp. 261–265, 2017.

[47] J. H. Holland, "Adaptation of natural and artificial systems," Ph.D. dissertation, University of Michigan, 1975.

[48] S. Rajashekaran and G. A. Vijayalksmi, *Neural Networks, Fuzzy Logic and Genetic Algorithms*. Prentice-Hall of India Pvt.Ltd, 2004.

[49] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge, UK: Cambridge University Press, 2004.

[50] R. Fletcher, *Practical Methods of Optimization*, 2nd ed. New York: Wiley, 2000.

[51] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*. Academic Press, 1981.

[52] A. Ravindran, K. Ragsdell, and G. Reklaitis, *Engineering optimization: methods and applications*. John Wiley & Sons, 2006.

[53] A. Bhatia, "Understanding motor nameplate information: Nema vs. iec standards," 2022. [Online]. Available: https://www.cedengineering.ca/courses/understanding-motor-nameplate-information-nema-vs-iec-standards-r1

[54] "Nema motor standards vs iec motor standards." [Online]. Available: https://www.nema.org/docs/motor-and-generator-guides-and-resources-library

[55] *NEMA MG 1-2021: Motors and Generators*, National Electrical Manufacturers Association (NEMA) Std., 2021.

[56] *Rotating electrical machines - Part 1: Rating and performance*, International Electrotechnical Commission (IEC) Std., 2010.

[57] P. Sen, *Principles of Electric Machines and Power Electronics*. Wiley, 2013.

[58] J. H. Holland, "Genetic Algorithms," *Scientific American*, vol. 267, no. 1, pp. 66–73, 1992.

[59] J. Kuhlmann, *Design of Electrical Apparatus*. J. Wiley & Sons, Incorporated, 1940.

Appendix A

Single Cage Equivalent Circuit Equations

The equations are applicable for the single-cage equivalent circuit in Figure 2.5.

Steady-state equivalent circuit equations as a function of slip(s):

- Stator Impedance:
$$Z_s = R_s + jX_s$$
- Rotor Impedance:
$$Z_r(s) = \frac{R_r}{s} + jX_r$$
- Stator Current:
$$I_s(s) = \frac{V_s}{Z_s + Z_r(s)||jX_m}$$
where V_s is the phase voltage(star equivalent circuit is considered).
- Rotor Current:
$$I_r(s) = I_s(s)\frac{jX_m}{Z_r(s) + jX_m}$$
- Torque:
$$T(s) = \frac{3}{\omega_{sync}}|I_r|^2\frac{R_r}{s}$$
where ω_{sync} is the synchronous speed in rad/s.

- Slip at maximum torque:

$$s_{max} = \frac{R_r}{\sqrt{R_{th}^2 + (X_{th} + X_r)^2}}$$

where $Z_{th} = R_{th} + jX_{th} = Z_s||jX_m$

Equations used in the objective function (5.13) for single-cage parameter estimation:

- Rated current:

$$I_{\text{FL,c}} = |I_s(s)|\Big|_{s=s_{FL}} \tag{A.1}$$

where s_{FL} is the slip at rated load.

- Rated torque:

$$T_{\text{FL,c}} = T_s(s)\Big|_{s=s_{FL}} \tag{A.2}$$

- Rated mechanical power output:

$$P_{\text{out,c}} = \omega_r T_{\text{FL,c}} \tag{A.3}$$

where ω_r is the rotor speed in rad/s at rated load.

- Input power factor at rated load:

$$PF_{\text{FL,c}} = \cos \angle I_s(s)\Big|_{s=s_{FL}} \tag{A.4}$$

- Efficiency at rated load:

$$\eta_{\text{FL,c}} = \frac{P_{\text{out,c}}}{P_{\text{in,c}}} \tag{A.5}$$

where $P_{\text{in,c}} = 3V_s I_{\text{FL,c}} PF_{\text{FL,c}}$

- Input reactive power at rated load:

$$Q_{\text{in,c}} = 3V_s I_{\text{FL,c}} \sqrt{1 - PF_{\text{FL,c}}^2} \tag{A.6}$$

- Locked rotor current:

$$I_{\text{LR,c}} = |I_s(s)|\Big|_{s=1} \tag{A.7}$$

- Locked rotor torque:

$$T_{\text{LR,c}} = T(s)\Big|_{s=1} \tag{A.8}$$

- Maximum torque:

$$T_{\text{max,c}} = T(s)\Big|_{s=s_{max}} \tag{A.9}$$

Appendix B

Double Cage Equivalent Circuit Equations

The equations are applicable for the double-cage equivalent circuit in Figure 2.6.

Steady-state equivalent circuit equations as a function of slip(s):

- Stator impedance:
$$Z_s = R_s + jX_s$$
- Outer-cage rotor impedance:
$$Z_{r1}(s) = \frac{R_1}{s}$$
- Inner-cage rotor impedance:
$$Z_{r2}(s) = \frac{R_2}{s} + jX_2$$
- Total rotor impedance:
$$Z_r(s) = (Z_{r1}(s)||Z_{r2}(s)) + jX_{12}$$
- Stator current:
$$I_s(s) = \frac{V_s}{Z_s + (Z_r(s)||jX_m)}$$
where V_s is the phase voltage(star equivalent circuit is considered).
- Total rotor current:
$$I_r(s) = I_s(s)\frac{jX_m}{Z_r(s) + jX_m}$$

- Outer-cage rotor current:

$$I_{r1}(s) = I_r(s)\frac{Z_{r2}(s)}{Z_{r1}(s) + Z_{r2}(s)}$$

- Inner-cage rotor current:

$$I_{r2}(s) = I_r(s)\frac{Z_{r1}(s)}{Z_{r1}(s) + Z_{r2}(s)}$$

- Torque:

$$T(s) = \frac{3}{\omega_{sync}}(|I_{r1}|^2\frac{R_1}{s} + |I_{r2}|^2\frac{R_2}{s})$$

  where ω_{sync} is the synchronous speed in rad/s.

Equations used in the objective function (5.13) for double-cage parameter estimation:

- Rated current:

$$I_{\text{FL,c}} = |I_s(s)|\Big|_{s=s_{FL}} \tag{B.1}$$

  where s_{FL} is the slip at rated load.

- Rated torque:

$$T_{\text{FL,c}} = T_s(s)\Big|_{s=s_{FL}} \tag{B.2}$$

- Rated mechanical power output:

$$P_{\text{out,c}} = \omega_r T_{\text{FL,c}} \tag{B.3}$$

  where ω_r is the rotor speed in rad/s at rated load.

- Input power factor at rated load:

$$PF_{\text{FL,c}} = \cos\angle I_s(s)\Big|_{s=s_{FL}} \tag{B.4}$$

- Efficiency at rated load:

$$\eta_{\text{FL,c}} = \frac{P_{\text{out,c}}}{P_{\text{in,c}}} \tag{B.5}$$

  where $P_{\text{in,c}} = 3V_s I_{\text{FL,c}} PF_{\text{FL,c}}$

- Input reactive power at rated load:

$$Q_{\text{in,c}} = 3V_s I_{\text{FL,c}} \sqrt{1 - PF_{\text{FL,c}}^2} \tag{B.6}$$

- Locked rotor current:

$$I_{\text{LR,c}} = |I_s(s)|\Big|_{s=1} \tag{B.7}$$

- Locked rotor torque:

$$T_{\text{LR,c}} = T(s)\Big|_{s=1} \tag{B.8}$$

- Maximum torque:

$$T_{\text{max,c}} = \max_{s \in (0,1)} T(s) \tag{B.9}$$

Appendix C

Catalog Data of Induction Motors used for Double Cage Parameter Estimation [59]

Table C.1: Operating characteristics of 3-phase, 60-cycle constant speed induction motors

HP	Speed (RPM)		Eff. (%)	Power Factor	Current (A) at 220 V		Torque (lb-ft)		
	Syn.	Full-Load			Locked Rotor	Full-Load	Full-Load	Locked Rotor	Max.
0.5	1200	1100	73	65	2.07	8.1	2.38	3.9	4.9
	900	835	69	49	2.9	8	3.13	4.7	6.2
0.75	1800	1735	80	79	2.33	14	2.27	4.2	6.2
	1200	1100	76	72	2.69	11.5	3.57	5.9	7
	900	855	75	59	3.33	14	4.6	6.9	11
1	3600	3490	82	87	2.75	20	1.5	2.8	3.5
	1800	1710	82	80	3	19	3.07	5.8	8.7
	1200	1130	70	75	3.31	15.5	4.64	7.7	9.4
	900	855	77	70	3.64	17	6.15	10	14
1.5	3600	3500	82	86	4.17	30	2.3	4.2	5.4
	1800	1735	85	85	4.07	30	4.53	8.8	13.5

Table C.1 - Continued

HP	Speed (RPM)		Eff. (%)	Power Factor	Current (A) at 220 V		Torque (lb-ft)		
	Syn.	Full-Load			Locked Rotor	Full-Load	Full-Load	Locked Rotor	Max.
	1200	1150	81	77	4.72	25	6.84	11.3	14.2
	900	865	79	68	5.48	26	9.11	15	25
2	3600	3510	82	86	5.56	46	3.1	6.2	7.9
	1800	1740	85	86	5.37	40	6.03	11.2	18
	1200	1155	84	82	5.7	36	9.1	15	19
	900	860	81	70	6.93	35	12.25	19	39
3	3600	3510	85	88	7.87	52	4.5	8.4	11.5
	1800	1730	86	88	7.76	50	9.11	16.9	25
	1200	1160	85	83	8.35	52	13.5	22.5	29
	900	860	82	73	9.85	47	18.3	28	44
5	3600	3460	86	90	12.67	86	7.6	14.1	20
	1800	1740	87	89	12.7	86	15.1	28	45
	1200	1150	86	86	13.25	86	22.8	37.7	48
	900	865	85	77	14.9	84	30.3	47	70
7.5	3600	3475	86	89	19.2	140	11.4	22.5	29
	1800	1740	88	90	18.6	120	22.6	42	62
	1200	1165	88	87	19.25	130	33.8	56	75
	900	865	86	77	22.2	126	45	70	95
10	3600	3510	87	92	24.5	187	15	29	39
	1800	1740	89	91	24.2	175	30.2	56	82
	1200	1160	88	88	25.3	170	45.2	74.8	96
	900	865	87	81	27.8	176	60	90	130
15	3600	3510	88	91	36.7	282	22.5	45	60
	1800	1755	90	91	36	282	44.8	83.2	147
	1200	1165	89	86	38.4	300	67	105	180
	900	875	88	83	40	290	90	160	250
20	3600	3500	88	91	49	450	30	56	85

Table C.1 - Continued

HP	Speed (RPM)		Eff. (%)	Power Factor	Current (A) at 220 V		Torque (lb-ft)		
	Syn.	Full-Load			Locked Rotor	Full-Load	Full-Load	Locked Rotor	Max.
	1800	1760	89.5	88	50	400	59	100	160
	1200	1170	89.5	88	50	430	89	150	300
	900	865	89.5	84	52	340	121	165	250
	720	680	86	83	55	280	154	185	290
	600	560	84.5	78	59.4	250	187	215	350
25	3600	3550	91	91	59.2	575	40	65	110
	1800	1755	90.5	90	60	440	75	112	220
	1200	1165	89.5	87	63	500	112	200	350
	900	865	89.5	84	65	410	152	190	300
	720	690	88	82	68	350	190	280	475
	600	575	88	80	70	350	228	375	580
30	1800	1755	91	91.5	71	540	90	135	230
	1200	1165	89.5	87	76	690	135	235	400
	900	860	89	86	77	500	183	300	440
	720	685	87	85	80	350	230	280	475
	600	570	87	82	82	350	276	375	580
40	1800	1735	89	92	96	630	121	250	360
	1200	1150	89.5	91	96	610	183	325	450
	900	855	89	88	100	690	245	350	560
	720	685	88.5	87	102	490	307	470	680
	600	570	88	83	107	510	369	490	800
50	1800	1740	89.5	92	119	780	150	245	440
	1200	1155	90	91	120	770	227	400	625
	900	870	89.5	88	124	900	303	540	850
	720	685	89	88	125	600	382	525	800
	600	570	87.5	83	135	620	460	550	875
60	1800	1740	90	93	140	900	181	290	600

Table C.1 - Continued

HP	Speed (RPM)		Eff. (%)	Power Factor	Current (A) at 220 V		Torque (lb-ft)		
	Syn.	Full-Load			Locked Rotor	Full-Load	Full-Load	Locked Rotor	Max.
	1200	1165	91	89	145	1230	269	525	800
	900	870	90	89	147	1070	362	650	1000
	720	700	90.5	85	153	970	450	675	1200
	600	575	89.5	82	160	840	547	675	1200
75	1800	1760	91.5	92	176	1820	223	445	800
	1200	1150	89.5	91	180	1100	342	475	840
	900	875	90	89	184	1200	452	900	1400
	720	695	90	88	186	990	566	785	1500
	600	580	89.5	83	198	950	678	780	1475
100	1800	1755	90.5	94	230	1450	299	450	800
	1200	1170	91.5	91	236	1650	448	600	1300
	900	875	90.5	90	240	1600	600	875	1600
	720	695	91	86.5	248	1360	755	910	1850
	600	575	90.5	85	256	1350	914	1200	2000
125	1800	1755	90.5	93.5	290	1900	374	400	1000
	1200	1175	91	91.5	294	2000	558	745	1600
	900	875	91	89	302	2060	750	1100	2200
	720	680	89.5	89	308	1320	967	1160	1900
	600	580	91	84	320	1900	1133	1700	3000
150	1800	1760	91	93	173	1145	446	490	1300
	1200	1170	93	92	172	1150	673	770	1900
	900	870	90.5	91	178	1125	904	1500	2400
	720	695	91.5	89	180	1025	1132	1500	3000
	600	575	90	88	185	930	1367	1700	3100
200	1800	1750	92	93	229	1530	600	600	1550
	1200	1175	93.5	92	228	1510	893	845	2400
	900	870	91	91	234	1500	1205	1600	3250

Table C.1 - Continued

HP	Speed (RPM)		Eff. (%)	Power Factor	Current (A) at 220 V		Torque (lb-ft)		
	Syn.	Full-Load			Locked Rotor	Full-Load	Full-Load	Locked Rotor	Max.
	720	700	91	89	242	2150	1500	2650	5000
	600	580	91	89	243	1400	1810	2500	4500
	450	430	90.5	82	265	1200	2440	2500	5200
	360	345	90.5	82	264	1375	3040	4000	7500
250	600	580	91.5	88	305	1800	2260	2800	6200
	450	430	91	85	317	1500	3060	3000	6400
	360	349	90.5	80	338	1650	3750	2850	9300
300	600	580	92	89	359	2400	2710	3500	7500
	450	436	91	85.5	380	2100	3600	3200	8500
	360	349	91	78	413	2100	4520	3000	11000
400	600	585	92	90.5	470	3060	3600	4200	10000
	450	437	92	85.5	500	2800	4830	3500	12000
	360	349	91.5	81.5	526	2300	6000	3800	14000
450	300	292	92	77	625	3100	8100	5400	18000
500	600	585	92	91	590	4700	4500	5600	12000
	450	437	92.5	97	602	3400	6000	4700	16000
	360	349	91.5	81.5	658	2750	7500	4500	18000
750	360	352	92	84	955	4100	11400	5000	23000

Appendix D

Double Cage Estimated Parameters using Genetic Algorithm

Table D.1: Double-cage estimated parameters using GA

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
1	8.672	11.469	61.644	0.450	16.190	8.100	0.889	$6.35 \cdot 10^{-3}$
2	5.797	10.103	37.785	2.637	13.718	6.864	5.265	$2.36 \cdot 10^{-3}$
3	6.701	3.758	73.896	0.081	6.515	3.263	0.154	$8.09 \cdot 10^{-4}$
4	6.657	7.074	54.870	0.359	11.723	5.867	0.708	$6.08 \cdot 10^{-3}$
5	4.960	7.424	35.770	0.167	6.807	3.409	0.323	$4.73 \cdot 10^{-3}$
6	5.541	1.016	76.279	0.014	4.421	2.215	0.020	$4.61 \cdot 10^{-3}$
7	4.205	4.512	62.522	0.031	6.340	3.175	0.073	$5.26 \cdot 10^{-3}$
8	7.432	4.227	49.027	0.129	8.384	4.197	0.249	$7.24 \cdot 10^{-3}$
9	4.995	4.318	38.710	0.164	5.847	2.928	0.317	$2.37 \cdot 10^{-3}$
10	3.664	0.462	49.003	0.011	2.736	1.373	0.082	$3.71 \cdot 10^{-3}$
11	2.927	2.430	53.915	0.030	3.369	1.690	0.051	$2.42 \cdot 10^{-3}$
12	3.905	1.995	33.875	0.025	3.608	1.809	0.040	$3.17 \cdot 10^{-3}$
13	2.684	3.071	26.192	0.171	3.213	1.611	0.332	$3.26 \cdot 10^{-4}$
14	2.466	0.334	37.413	0.011	1.856	0.933	0.013	$5.22 \cdot 10^{-3}$
15	2.162	1.864	42.390	0.024	2.332	1.171	0.039	$1.91 \cdot 10^{-3}$
16	2.938	1.145	31.967	0.047	2.545	1.277	0.084	$5.04 \cdot 10^{-3}$

Table D.1 - Continued

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
17	1.130	3.294	23.215	0.048	2.673	1.341	0.087	$4.14 \cdot 10^{-3}$
18	1.803	0.144	30.133	0.388	1.251	0.630	0.957	$3.67 \cdot 10^{-4}$
19	1.503	1.316	33.110	0.112	1.862	0.936	0.213	$1.65 \cdot 10^{-4}$
20	2.000	0.406	22.441	0.120	1.594	0.802	0.230	$3.27 \cdot 10^{-3}$
21	1.492	1.705	16.181	0.093	1.835	0.923	0.176	$4.46 \cdot 10^{-4}$
22	0.993	0.747	21.670	0.012	1.108	0.559	0.014	$5.67 \cdot 10^{-4}$
23	0.854	0.727	21.388	0.089	0.975	0.492	0.170	$9.56 \cdot 10^{-5}$
24	1.187	0.585	16.003	0.010	1.130	0.570	0.010	$4.51 \cdot 10^{-3}$
25	1.022	0.886	11.541	0.029	1.029	0.519	0.048	$1.95 \cdot 10^{-3}$
26	0.723	0.209	12.451	0.013	0.673	0.341	0.015	$1.50 \cdot 10^{-3}$
27	0.549	0.508	16.246	0.070	0.657	0.333	0.312	$5.56 \cdot 10^{-5}$
28	0.788	0.091	11.421	0.099	0.572	0.291	0.223	$2.65 \cdot 10^{-3}$
29	0.739	0.500	7.589	0.027	0.685	0.348	0.045	$4.06 \cdot 10^{-3}$
30	0.544	0.029	11.598	0.044	0.379	0.194	0.204	$1.01 \cdot 10^{-3}$
31	0.494	0.318	12.002	0.019	0.490	0.250	0.029	$3.45 \cdot 10^{-4}$
32	0.615	0.138	8.844	0.012	0.483	0.246	0.053	$3.45 \cdot 10^{-3}$
33	0.562	0.375	6.718	0.013	0.512	0.261	0.017	$5.20 \cdot 10^{-3}$
34	0.350	0.062	7.449	0.010	0.250	0.130	0.143	$9.12 \cdot 10^{-4}$
35	0.224	0.283	8.978	0.013	0.253	0.131	0.065	$2.71 \cdot 10^{-5}$
36	0.327	0.227	5.898	0.010	0.271	0.141	0.010	$3.39 \cdot 10^{-3}$
37	0.328	0.184	5.023	0.010	0.267	0.139	0.010	$8.07 \cdot 10^{-4}$
38	0.243	0.090	5.594	0.010	0.197	0.104	0.010	$4.29 \cdot 10^{-3}$
39	0.260	0.068	4.651	0.010	0.164	0.087	0.021	$1.11 \cdot 10^{-3}$
40	0.177	0.209	5.307	0.010	0.178	0.094	0.010	$2.25 \cdot 10^{-3}$
41	0.286	0.211	4.020	0.010	0.253	0.132	0.010	$6.05 \cdot 10^{-3}$
42	0.280	0.320	3.868	0.011	0.328	0.169	0.011	$4.03 \cdot 10^{-3}$
43	0.230	0.413	3.296	0.013	0.374	0.192	0.016	$2.15 \cdot 10^{-3}$
44	0.192	0.016	4.592	0.010	0.080	0.045	0.057	$1.91 \cdot 10^{-3}$
45	0.154	0.186	5.058	0.015	0.146	0.078	0.021	$7.52 \cdot 10^{-5}$
46	0.155	0.162	3.951	0.010	0.165	0.087	0.010	$1.33 \cdot 10^{-3}$

Table D.1 - Continued

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
47	0.231	0.186	3.260	0.010	0.198	0.104	0.010	$6.13 \cdot 10^{-3}$
48	0.145	0.225	3.297	0.035	0.226	0.118	0.060	$4.39 \cdot 10^{-5}$
49	0.130	0.178	3.094	0.065	0.231	0.120	0.120	$2.66 \cdot 10^{-5}$
50	0.169	0.099	4.185	0.010	0.118	0.064	0.017	$8.86 \cdot 10^{-4}$
51	0.147	0.093	3.102	0.010	0.131	0.070	0.010	$6.94 \cdot 10^{-3}$
52	0.164	0.146	3.058	0.012	0.196	0.103	0.013	$2.68 \cdot 10^{-3}$
53	0.137	0.209	3.340	0.045	0.210	0.110	0.092	$7.35 \cdot 10^{-5}$
54	0.124	0.186	2.993	0.067	0.226	0.118	0.130	$6.65 \cdot 10^{-5}$
55	0.088	0.083	3.803	0.033	0.133	0.072	0.066	$1.07 \cdot 10^{-5}$
56	0.118	0.089	3.348	0.026	0.148	0.079	0.041	$6.22 \cdot 10^{-4}$
57	0.130	0.118	2.641	0.010	0.157	0.084	0.010	$9.41 \cdot 10^{-3}$
58	0.092	0.114	2.967	0.056	0.169	0.090	0.102	$1.23 \cdot 10^{-4}$
59	0.097	0.162	2.260	0.026	0.164	0.087	0.043	$3.61 \cdot 10^{-4}$
60	0.067	0.106	3.186	0.015	0.097	0.053	0.019	$3.06 \cdot 10^{-5}$
61	0.075	0.088	2.832	0.020	0.107	0.059	0.030	$6.05 \cdot 10^{-5}$
62	0.086	0.077	2.105	0.010	0.092	0.051	0.010	$7.00 \cdot 10^{-4}$
63	0.084	0.109	2.526	0.035	0.133	0.071	0.066	$3.46 \cdot 10^{-4}$
64	0.105	0.125	1.677	0.015	0.123	0.067	0.029	$1.19 \cdot 10^{-3}$
65	0.027	0.107	3.351	0.014	0.083	0.047	0.017	$3.84 \cdot 10^{-4}$
66	0.075	0.041	1.816	0.010	0.068	0.039	0.010	$1.74 \cdot 10^{-3}$
67	0.074	0.059	1.855	0.010	0.076	0.043	0.010	$6.83 \cdot 10^{-4}$
68	0.069	0.068	1.562	0.014	0.062	0.036	0.029	$2.43 \cdot 10^{-4}$
69	0.074	0.106	1.382	0.010	0.087	0.048	0.011	$1.04 \cdot 10^{-3}$
70	0.050	0.028	1.816	0.010	0.040	0.025	0.010	$2.52 \cdot 10^{-3}$
71	0.055	0.079	1.976	0.010	0.074	0.042	0.010	$9.22 \cdot 10^{-4}$
72	0.036	0.051	1.716	0.020	0.052	0.031	0.046	$6.41 \cdot 10^{-5}$
73	0.035	0.076	1.739	0.022	0.064	0.037	0.035	$6.30 \cdot 10^{-5}$
74	0.053	0.077	1.243	0.021	0.056	0.033	0.040	$1.81 \cdot 10^{-4}$
75	0.035	0.043	2.157	0.017	0.033	0.022	0.034	$3.49 \cdot 10^{-6}$
76	0.033	0.050	1.485	0.010	0.033	0.021	0.010	$1.02 \cdot 10^{-4}$

Table D.1 - Continued

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
77	0.041	0.041	1.299	0.012	0.036	0.023	0.015	$1.62 \cdot 10^{-4}$
78	0.035	0.064	1.153	0.011	0.045	0.027	0.012	$1.25 \cdot 10^{-4}$
79	0.042	0.055	0.999	0.013	0.052	0.031	0.017	$7.98 \cdot 10^{-4}$
80	0.028	0.045	1.612	0.010	0.024	0.017	0.010	$4.62 \cdot 10^{-4}$
81	0.026	0.034	1.269	0.014	0.021	0.015	0.018	$2.79 \cdot 10^{-5}$
82	0.026	0.037	1.019	0.010	0.028	0.019	0.010	$1.26 \cdot 10^{-4}$
83	0.028	0.052	1.269	0.020	0.057	0.033	0.031	$1.86 \cdot 10^{-4}$
84	0.028	0.039	0.765	0.011	0.034	0.022	0.013	$2.06 \cdot 10^{-4}$
85	0.077	0.173	5.145	0.014	0.088	0.049	0.019	$4.60 \cdot 10^{-5}$
86	0.080	0.174	4.716	0.012	0.097	0.054	0.015	$1.06 \cdot 10^{-4}$
87	0.107	0.116	3.851	0.024	0.130	0.070	0.061	$9.53 \cdot 10^{-5}$
88	0.073	0.187	3.761	0.014	0.135	0.072	0.039	$1.06 \cdot 10^{-5}$
89	0.094	0.176	3.488	0.034	0.156	0.083	0.057	$1.08 \cdot 10^{-4}$
90	0.072	0.137	3.986	0.010	0.077	0.043	0.010	$1.58 \cdot 10^{-3}$
91	0.066	0.132	3.532	0.011	0.058	0.034	0.011	$1.28 \cdot 10^{-4}$
92	0.070	0.130	3.144	0.011	0.095	0.053	0.011	$3.68 \cdot 10^{-4}$
93	0.069	0.079	2.358	0.010	0.077	0.043	0.010	$3.34 \cdot 10^{-3}$
94	0.075	0.100	2.637	0.028	0.093	0.051	0.046	$1.29 \cdot 10^{-4}$
95	0.061	0.171	1.876	0.014	0.113	0.061	0.017	$2.74 \cdot 10^{-4}$
96	0.056	0.147	1.811	0.012	0.111	0.061	0.015	$2.47 \cdot 10^{-4}$
97	0.042	0.112	2.068	0.011	0.073	0.042	0.011	$1.76 \cdot 10^{-4}$
98	0.058	0.138	1.779	0.010	0.089	0.050	0.010	$6.02 \cdot 10^{-4}$
99	0.034	0.142	1.393	0.010	0.060	0.035	0.010	$1.44 \cdot 10^{-3}$
100	0.044	0.080	1.779	0.010	0.059	0.035	0.010	$1.64 \cdot 10^{-3}$
101	0.047	0.098	1.444	0.010	0.051	0.030	0.010	$1.30 \cdot 10^{-3}$
102	0.030	0.118	1.073	0.010	0.047	0.028	0.010	$5.36 \cdot 10^{-3}$
103	0.031	0.058	1.512	0.010	0.033	0.021	0.010	$6.34 \cdot 10^{-5}$
104	0.024	0.082	1.202	0.010	0.034	0.022	0.010	$4.01 \cdot 10^{-3}$
105	0.014	0.099	1.031	0.010	0.036	0.023	0.010	$9.87 \cdot 10^{-4}$
106	0.027	0.070	0.660	0.010	0.025	0.018	0.010	$2.35 \cdot 10^{-3}$

Table D.1 - Continued

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
107	0.036	0.027	1.087	0.010	0.023	0.017	0.010	$3.65 \cdot 10^{-3}$
108	0.010	0.057	6.148	0.010	0.025	0.018	0.010	$2.22 \cdot 10^{-3}$
109	0.010	0.080	0.851	0.010	0.028	0.019	0.010	$1.76 \cdot 10^{-3}$
110	0.012	0.052	0.657	0.010	0.011	0.011	0.010	$1.17 \cdot 10^{-3}$

Appendix E

Double Cage Estimated Parameters using Simplex Algorithm

Table E.1: Double-cage estimated parameters using Simplex

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
1	8.464	12.536	60.701	0.010	57.172	5.769	0.010	$6.28 \cdot 10^{-3}$
2	5.856	14.025	33.760	0.010	173.624	3.792	0.017	$2.08 \cdot 10^{-3}$
3	6.774	3.715	73.215	0.010	19.439	2.434	0.010	$7.99 \cdot 10^{-4}$
4	6.543	7.880	54.095	0.010	45.632	4.159	0.011	$6.02 \cdot 10^{-3}$
5	4.909	7.778	35.231	0.010	49.402	2.329	0.011	$4.69 \cdot 10^{-3}$
6	5.577	0.821	75.788	0.010	9.396	1.760	0.019	$4.60 \cdot 10^{-3}$
7	4.238	4.509	62.225	0.010	10.267	2.655	0.010	$5.26 \cdot 10^{-3}$
8	7.504	4.332	48.619	0.010	59.526	2.902	0.010	$7.22 \cdot 10^{-3}$
9	4.918	4.716	38.475	0.010	9.744	2.386	0.010	$2.34 \cdot 10^{-3}$
10	3.689	0.282	48.644	0.010	3.042	1.318	0.181	$3.70 \cdot 10^{-3}$
11	3.023	2.302	53.273	0.010	5.062	1.444	0.010	$2.40 \cdot 10^{-3}$
12	3.979	1.842	33.527	0.010	3.668	1.802	0.011	$3.16 \cdot 10^{-3}$
13	2.659	3.389	25.915	0.010	9.651	1.175	0.010	$2.78 \cdot 10^{-4}$
14	2.477	0.290	37.265	0.010	7.497	0.680	0.011	$5.22 \cdot 10^{-3}$
15	2.262	1.715	41.632	0.010	6.260	0.888	0.010	$1.88 \cdot 10^{-3}$
16	2.986	1.078	31.563	0.010	3.947	1.082	0.010	$5.02 \cdot 10^{-3}$

Table E.1 - Continued

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
17	1.073	3.408	23.242	0.010	10.300	0.972	0.010	$4.11 \cdot 10^{-3}$
18	1.908	0.038	28.474	0.010	1.529	0.577	0.834	$3.01 \cdot 10^{-4}$
19	1.572	1.394	32.337	0.010	13.835	0.645	0.010	$1.12 \cdot 10^{-4}$
20	2.040	0.338	22.012	0.042	3.518	0.631	0.089	$3.25 \cdot 10^{-3}$
21	1.477	1.871	15.979	0.011	2.277	0.820	0.012	$4.12 \cdot 10^{-4}$
22	1.066	0.588	20.713	0.010	3.069	0.421	0.010	$4.96 \cdot 10^{-4}$
23	0.808	0.918	21.555	0.010	2.059	0.386	0.010	$2.37 \cdot 10^{-5}$
24	1.223	0.485	15.737	0.010	2.281	0.457	0.010	$4.48 \cdot 10^{-3}$
25	1.052	0.868	11.411	0.010	3.863	0.376	0.010	$1.92 \cdot 10^{-3}$
26	0.738	0.143	12.185	0.010	3.086	0.246	0.011	$1.48 \cdot 10^{-3}$
27	0.592	0.429	15.744	0.090	0.657	0.333	0.296	$1.70 \cdot 10^{-5}$
28	0.811	0.010	11.107	0.044	0.917	0.248	0.196	$2.61 \cdot 10^{-3}$
29	0.756	0.480	7.511	0.010	13.055	0.235	0.024	$4.04 \cdot 10^{-3}$
30	0.544	0.010	11.545	0.010	0.771	0.155	0.194	$1.02 \cdot 10^{-3}$
31	0.513	0.288	11.708	0.010	1.722	0.183	0.012	$3.24 \cdot 10^{-4}$
32	0.627	0.067	8.688	0.027	0.488	0.249	0.052	$3.42 \cdot 10^{-3}$
33	0.571	0.364	6.641	0.010	0.544	0.252	0.010	$5.18 \cdot 10^{-3}$
34	0.350	0.010	7.430	0.032	0.476	0.107	0.129	$9.36 \cdot 10^{-4}$
35	0.238	0.139	8.904	0.039	4.609	0.090	0.136	$1.97 \cdot 10^{-6}$
36	0.339	0.201	5.810	0.010	1.187	0.100	0.010	$3.36 \cdot 10^{-3}$
37	0.338	0.156	4.981	0.010	6.607	0.093	0.012	$7.91 \cdot 10^{-4}$
38	0.251	0.060	5.435	0.010	0.521	0.079	0.010	$4.23 \cdot 10^{-3}$
39	0.270	0.020	4.482	0.010	0.315	0.070	0.012	$1.04 \cdot 10^{-3}$
40	0.179	0.202	5.289	0.010	0.781	0.067	0.011	$2.23 \cdot 10^{-3}$
41	0.295	0.191	3.966	0.010	1.162	0.094	0.010	$6.03 \cdot 10^{-3}$
42	0.281	0.315	3.862	0.010	3.480	0.115	0.011	$4.02 \cdot 10^{-3}$
43	0.225	0.419	3.299	0.010	1.725	0.137	0.010	$2.13 \cdot 10^{-3}$
44	0.189	0.010	4.674	0.010	0.233	0.033	0.056	$2.00 \cdot 10^{-3}$
45	0.159	0.185	4.993	0.010	0.308	0.061	0.010	$5.61 \cdot 10^{-5}$
46	0.158	0.157	3.938	0.010	0.272	0.072	0.010	$1.33 \cdot 10^{-3}$

Table E.1 - Continued

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
47	0.241	0.165	3.199	0.010	1.570	0.072	0.010	$6.08 \cdot 10^{-3}$
48	0.140	0.247	3.303	0.010	0.342	0.099	0.054	$3.46 \cdot 10^{-5}$
49	0.128	0.176	3.089	0.061	0.299	0.108	0.111	$2.15 \cdot 10^{-5}$
50	0.183	0.062	3.938	0.010	0.802	0.044	0.010	$7.92 \cdot 10^{-4}$
51	0.152	0.078	3.061	0.010	0.425	0.051	0.010	$6.89 \cdot 10^{-3}$
52	0.166	0.144	3.041	0.010	0.382	0.082	0.010	$2.66 \cdot 10^{-3}$
53	0.130	0.209	3.393	0.051	0.215	0.111	0.093	$5.49 \cdot 10^{-5}$
54	0.124	0.236	2.944	0.016	0.219	0.114	0.125	$6.64 \cdot 10^{-5}$
55	0.087	0.054	3.838	0.040	0.350	0.055	0.069	$5.61 \cdot 10^{-6}$
56	0.127	0.091	3.180	0.011	0.217	0.067	0.023	$5.63 \cdot 10^{-4}$
57	0.135	0.104	2.580	0.010	0.839	0.058	0.010	$9.34 \cdot 10^{-3}$
58	0.095	0.154	2.890	0.010	0.182	0.083	0.094	$1.25 \cdot 10^{-4}$
59	0.101	0.174	2.210	0.010	0.357	0.066	0.021	$3.28 \cdot 10^{-4}$
60	0.067	0.107	3.168	0.010	0.385	0.038	0.011	$1.78 \cdot 10^{-5}$
61	0.076	0.097	2.808	0.010	0.115	0.056	0.027	$5.88 \cdot 10^{-5}$
62	0.090	0.069	2.076	0.010	0.186	0.040	0.010	$6.85 \cdot 10^{-4}$
63	0.086	0.132	2.488	0.010	0.130	0.070	0.064	$3.45 \cdot 10^{-4}$
64	0.101	0.125	1.700	0.019	0.124	0.067	0.029	$1.18 \cdot 10^{-3}$
65	0.025	0.111	3.375	0.010	0.243	0.034	0.010	$3.34 \cdot 10^{-4}$
66	0.082	0.023	1.729	0.010	0.096	0.033	0.010	$1.77 \cdot 10^{-3}$
67	0.077	0.050	1.817	0.010	0.185	0.032	0.010	$6.56 \cdot 10^{-4}$
68	0.068	0.071	1.560	0.010	0.077	0.032	0.027	$2.41 \cdot 10^{-4}$
69	0.073	0.102	1.384	0.010	0.552	0.033	0.010	$1.01 \cdot 10^{-3}$
70	0.051	0.021	1.808	0.010	0.382	0.016	0.011	$2.22 \cdot 10^{-3}$
71	0.056	0.074	1.965	0.010	0.236	0.030	0.010	$9.03 \cdot 10^{-4}$
72	0.039	0.010	1.722	0.027	0.300	0.022	0.057	$3.01 \cdot 10^{-5}$
73	0.033	0.058	1.761	0.020	0.282	0.026	0.042	$3.89 \cdot 10^{-5}$
74	0.052	0.086	1.237	0.013	0.056	0.033	0.039	$1.78 \cdot 10^{-4}$
75	0.037	0.038	2.096	0.010	0.088	0.015	0.029	$5.56 \cdot 10^{-8}$
76	0.034	0.044	1.470	0.010	0.145	0.014	0.010	$6.89 \cdot 10^{-5}$

Table E.1 - Continued

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
77	0.043	0.038	1.274	0.012	0.038	0.022	0.014	$1.42 \cdot 10^{-4}$
78	0.035	0.066	1.151	0.010	0.047	0.027	0.010	$1.16 \cdot 10^{-4}$
79	0.043	0.056	0.987	0.011	0.052	0.031	0.015	$7.93 \cdot 10^{-4}$
80	0.028	0.032	6.255	0.025	0.010	0.010	2.075	$2.45 \cdot 10^{-4}$
81	0.028	0.035	1.234	0.010	0.028	0.013	0.014	$1.86 \cdot 10^{-7}$
82	0.025	0.030	1.028	0.011	0.244	0.012	0.012	$1.00 \cdot 10^{-4}$
83	0.027	0.060	1.267	0.013	0.057	0.033	0.029	$1.77 \cdot 10^{-4}$
84	0.028	0.036	0.762	0.011	0.049	0.018	0.012	$2.06 \cdot 10^{-4}$
85	0.079	0.171	5.080	0.010	0.252	0.036	0.015	$3.64 \cdot 10^{-5}$
86	0.079	0.173	4.697	0.011	0.453	0.038	0.011	$7.91 \cdot 10^{-5}$
87	0.107	0.130	3.845	0.011	0.129	0.069	0.061	$9.50 \cdot 10^{-5}$
88	0.074	0.191	3.751	0.010	0.140	0.071	0.036	$9.48 \cdot 10^{-6}$
89	0.096	0.188	3.449	0.020	0.155	0.082	0.053	$1.03 \cdot 10^{-4}$
90	0.073	0.130	3.964	0.011	0.323	0.030	0.012	$1.57 \cdot 10^{-3}$
91	0.067	0.126	3.503	0.010	0.312	0.023	0.010	$1.15 \cdot 10^{-4}$
92	0.069	0.127	3.148	0.010	0.401	0.037	0.010	$3.40 \cdot 10^{-4}$
93	0.073	0.070	2.326	0.010	0.324	0.030	0.010	$3.28 \cdot 10^{-3}$
94	0.074	0.112	2.606	0.010	0.140	0.043	0.041	$1.27 \cdot 10^{-4}$
95	0.062	0.171	1.867	0.010	0.444	0.044	0.011	$2.55 \cdot 10^{-4}$
96	0.061	0.143	1.788	0.010	0.321	0.044	0.010	$2.06 \cdot 10^{-4}$
97	0.041	0.109	2.074	0.010	0.241	0.030	0.011	$1.55 \cdot 10^{-4}$
98	0.054	0.137	1.806	0.010	0.304	0.036	0.010	$5.66 \cdot 10^{-4}$
99	0.029	0.140	1.417	0.010	0.458	0.024	0.010	$1.33 \cdot 10^{-3}$
100	0.041	0.082	1.804	0.010	0.084	0.030	0.010	$1.65 \cdot 10^{-3}$
101	0.045	0.099	1.453	0.010	0.062	0.028	0.010	$1.32 \cdot 10^{-3}$
102	0.027	0.116	1.087	0.010	0.202	0.020	0.010	$5.19 \cdot 10^{-3}$
103	0.032	0.051	1.497	0.010	0.367	0.014	0.010	$2.80 \cdot 10^{-5}$
104	0.022	0.080	1.218	0.010	0.115	0.016	0.010	$3.94 \cdot 10^{-3}$
105	0.012	0.096	1.038	0.010	0.146	0.016	0.010	$9.05 \cdot 10^{-4}$
106	0.026	0.067	0.667	0.010	0.075	0.012	0.010	$2.48 \cdot 10^{-3}$

Table E.1 - Continued

Machine	$\mathbf{R}_s$ **(Ω)**	$\mathbf{X}_s$ **(Ω)**	$\mathbf{X}_m$ **(Ω)**	$\mathbf{X}_{12}$ **(Ω)**	$\mathbf{R}_1$ **(Ω)**	$\mathbf{R}_2$ **(Ω)**	$\mathbf{X}_2$ **(Ω)**	**Fitness**
107	0.038	0.019	1.061	0.010	0.118	0.011	0.010	$3.38 \cdot 10^{-3}$
108	0.010	0.053	7.550	0.011	0.065	0.012	0.011	$2.77 \cdot 10^{-3}$
109	0.010	0.076	0.858	0.010	0.089	0.013	0.010	$1.75 \cdot 10^{-3}$
110	0.014	0.051	0.656	0.010	0.011	0.010	0.010	$1.45 \cdot 10^{-3}$

Appendix F

Per-unit Values of Estimated Parameters for Tables 6.3, 6.7, 6.9, and 6.10

Note: The per-unit values are based on each machine's rated output power and voltage.

Table F.1: Per-unit single-cage estimated parameters from nameplate data[1]

Method	$\mathbf{R}_s$ (pu)	$\mathbf{X}_s$ (pu)	$\mathbf{X}_m$ (pu)	$\mathbf{R}_r$ (pu)	$\mathbf{X}_r$ (pu)	**Fitness**
GA	0.023	0.054	1.488	0.012	0.054	$1.36 \cdot 10^{-5}$
Simplex	0.030	0.053	1.562	0.012	0.053	$5.82 \cdot 10^{-5}$

Table F.2: Per-unit double-cage estimated parameters from catalog data[2]

Method	$\mathbf{R}_s$ (pu)	$\mathbf{X}_s$ (pu)	$\mathbf{X}_m$ (pu)	$\mathbf{X}_{12}$ (pu)	$\mathbf{R}_1$ (pu)	$\mathbf{R}_2$ (pu)	$\mathbf{X}_2$ (pu)	**Fitness**
Actual	0.032	0.065	2.230	0.049	0.055	0.019	0.055	
GA	0.026	0.067	2.315	0.059	0.028	0.028	0.116	$1.95 \cdot 10^{-4}$
Simplex	0.032	0.078	2.216	0.038	0.052	0.019	0.054	$1.41 \cdot 10^{-8}$

[1] Corresponds to Table 6.3.
[2] Corresponds to Table 6.7.

Table F.3: Per-unit double-cage estimated parameters for the best-fitting machine[3]

Method	$\mathbf{R}_s$ (pu)	$\mathbf{X}_s$ (pu)	$\mathbf{X}_m$ (pu)	$\mathbf{X}_{12}$ (pu)	$\mathbf{R}_1$ (pu)	$\mathbf{R}_2$ (pu)	$\mathbf{X}_2$ (pu)	**Fitness**
GA	0.050	0.065	2.444	0.027	0.040	0.029	0.035	$2.79 \cdot 10^{-5}$
Simplex	0.054	0.067	2.377	0.019	0.054	0.025	0.027	$1.86 \cdot 10^{-7}$

Table F.4: Per-unit double-cage estimated parameters for the worst-fitting machine[4]

Method	$\mathbf{R}_s$ (pu)	$\mathbf{X}_s$ (pu)	$\mathbf{X}_m$ (pu)	$\mathbf{X}_{12}$ (pu)	$\mathbf{R}_1$ (pu)	$\mathbf{R}_2$ (pu)	$\mathbf{X}_2$ (pu)	**Fitness**
GA	0.080	0.073	1.628	0.006	0.097	0.052	0.006	$9.41 \cdot 10^{-3}$
Simplex	0.083	0.064	1.590	0.006	0.517	0.036	0.006	$9.34 \cdot 10^{-3}$

[3]Corresponds to Table 6.9.
[4]Corresponds to Table 6.10.