Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2023

2D Hybrid Analytical Model Based Performance Optimization for Linear Induction Motors

Michael Thamm
University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Part of the Electrical and Computer Engineering Commons

2D Hybrid Analytical Model Based Performance Optimization for Linear Induction Motors

By

Michael Thamm

A Thesis
Submitted to the Faculty of Graduate Studies
through the Department of Electrical & Computer Engineering
in Partial Fulfillment of the Requirements for
the Degree of Master of Applied Science
at the University of Windsor

Windsor, Ontario, Canada

2023

2D Hybrid Analytical Model Based Performance Optimization for Linear Induction
Motors

by

Michael Thamm

APPROVED BY:

_____
J. Urbanic

Department of Mechanical, Automotive & Materials Engineering

_____
M. S. Toulabi

Department of Electrical & Computer Engineering

_____
N. C. Kar, Advisor

Department of Electrical & Computer Engineering

April/11/2023

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

In this thesis the domain of double-layer, single-sided, 3-phase, integral slot winding, linear induction motor (LIM)s is analyzed. Motor meta parameters such as slots and poles are difficult to optimize since they drastically effect the configuration of the motor and require heuristic optimization implementations.

A non-dominated sorting genetic algorithm II (NSGAII) was implemented with the Platypus-Opt Python library. It serves as a robust, yet flexible integration while maximizing thrust and minimizing the mass of each motor iteration. Each iteration was accurately modelled using the hybrid analytical model (HAM), producing the necessary performance parameters for the NSGAII's objective function. Field plotting capability of the processed HAM allowed for the feasibility check on post-processing constraints, increasing the robustness of the optimization.

Validation between the HAM to finite element analysis (FEA) and HAM to the baseline proved the accuracy of the modelling algorithm within the objective function. The optimization concluded that the optimal motor had 36 slots and 4 poles within the domain of 12-54 slots and 4-12 poles, where 9 feasible motors were objectively compared.

The proposed design tool lacks the ability to produce a fully functional optimized motor due to domain and complexity constraints. However, it saved significant time and effort while generating reproducible results within a constrained domain. The entire optimization was completed in 5 minutes, whereas the total time for configuring all motors via FEA within the domain took 4.5 hours, proving its worth.

DEDICATION

I would like to express sincere gratitude towards Dr. Kar, who contributed in many ways to "get the monkey off my back" and complete this MASc journey. Whether I required financial or academic support, Dr. Kar was always optimistic and willing to progress research within the lab. This created a competitive and supportive lab environment within CHARGE Labs that I was lucky enough to take part in. The critical feedback I received from my thesis committee members, Dr. Toulabi and Dr. Urbanic, was constructive in nature and challenged my thesis at its core. I am grateful to have worked with such cooperative and supportive committee members.

Throughout the countless hours of work, I grew as an academic and a teammate among many talented and supportive lab members who have helped to make this journey possible. I would like to thank Shruthi Mukundan, Himavarsha Dhulipati, and Aida Mollaeian for being the first few lab members to drive my initial understanding of electric motor theory and simulation in a masters setting. Lastly, I would like to thank the lab members that were there for me throughout most of my journey Brad Sato, Tim Stachl, and Areej Fatima. Without these supportive people, I would not be publishing this thesis and I am forever grateful for these amazing friends.

ACKNOWLEDGEMENTS

The rollercoaster of emotions and stress that comes with the thesis journey become worthwhile only if shared with the most important people in your life. Solange, Isabelle, Heike, Uwe I love you all so much for the patience and openness you have shown me throughout the long years of my work and MASc career. I am proud to have such an amazing family on my side and I will always return the favour without hesitation.

If you see this, thank you Renée B. Goodman, for taking the time to help me with countless theoretical problems and always being optimistic!

Zoe Zhu, thank you for helping me get through the biggest roadblock of my thesis and being a great collaborator!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

xiii

LIST OF ABBREVIATIONS

Abbreviation	Description
API	Application Programming Interface
EMF	Electromotive force
FEA	Finite Element Analysis
GA	Genetic Algorithm
HAM	Hybrid Analytical Model
HM	Harmonic Model
KCL	Kirchhoff's Current Law
LEM	Linear Electric Motor
LIM	Linear Induction Motor
MEC	Magnetic Equivalent Circuit
MMF	Magnetomotive Force
NSGAII	Non-Dominated Sorting Genetic Algorithm II
PSO	Particle Swarm Optimization
WDT	Winding Distribution Table

NOMENCLATURE

Variable	Description
N_s	Number of slots
N_p	Number of poles
m	Number of phases
v	Synchronous velocity
f	Electrical frequency
I_p	Peak current
J	Current density
N_t	Number of turns per coil
q	Slots per poles per phase
σ	Conductivity
ρ	Resistivity
μ_r	Relative permeability
μ_o	Vacuum permeability
h_s	Slot height
h_y	Yoke height
w_t	Tooth width
w_s	Slot width
λ	Slot pitch
h_g	Airgap
h_{Al}	Aluminum thickness
h_{bi}	Back iron thickness
L_p	Primary motor length
H_p	Primary motor height
D_p	Primary motor depth
τ_{per}	Periodical length of model
n	Space harmonic
N	Number of space harmonics
a_n, b_n	Complex harmonic analysis unknowns for nth space harmonic

x	Spatial position in the x-direction
y	Spatial position in the y-direction
β	Number of harmonic model regions in the model
γ	Number of magnetic equivalent circuit regions in the model
l	y-index of a node in the magnetic equivalent circuit region
k	x-index of a node in the magnetic equivalent circuit region
L	Number of rows in a magnetic equivalent circuit region
K	Number of columns in a magnetic equivalent circuit region
M	Total nodes in a magnetic equivalent circuit region
$\mathcal{F}$	Magnetomotive force
Φ	Magnetic flux
Ψ	Complex scalar potential
F	Primary motor force
B	Magnetic flux density
H	Magnetic field
δ	Skin depth of magnetic field in material
n_c	Number of columns in a WDT table
y_c	Coil pitch of a double-layer winding
y_p	Pole pitch of the motor

CHAPTER 1   Introduction

1.1. Electric Trains - A Green Alternative

In 2009, the European commission for science and environmental policy [1] stated that the world must not exceed the 1 trillion carbon budget to avoid a two-degree Celsius rise in the world's average temperature. Through a combined effort of all countries across the world human civilization has introduced 6.36 trillion tonnes of CO_2 emissions from the late 1800s to 2020, according to Science for Environment Policy [2]. The impact of accelerated rates of CO_2 output has serious implications on the health of the world's ecosystems and is quickly becoming a concern across economic and geopolitical conversations. Since global transportation accounts for 37% of the CO_2 emissions from end-use sectors [3], it is imperative that there be an initiative which can alleviate some of this contribution. According to the government of Canada [4] the efficiency of energy conversion [5] from on-board storage to turning the wheels is nearly five times greater for electricity when compared to gasoline, at approximately 76% and 16%, respectively. If this data were scaled to the number of combustion engines that exist globally, the lost potential and the environmental impact becomes clear.

One of the most efficient ways to travel is via high-speed [6] electric train due to ride sharing and efficiency [7], which could drastically reduce the global carbon footprint of transportation if it were the primary means for transportation. The Siemens Velaro D (DB Class 407) high-speed electric train [8] is designed for operation at 320 km/h with an output power of 8 MW. Since these trains can span hundreds of meters, 16 motors were distributed

across the train cars, each producing 500–600 kW. The class 407 trains were first operational in 2013 and are competitive efficient high-speed electric motors [9].

2013 ICE Siemens Velaro D powered train. [6]

Double sided LIM in its final stage of production at Gebrüder Meier AG in Regensdorf. [10]

In contrast the Swissloop team produced a double-sided linear induction motor (LIM) electric train prototype that can achieve a top speed of 252 km/h at 250 kW of output power [11]. Since rotary electric motor applications require mechanical traction, they experience mechanical losses and complexity [12], which are not applicable to a linear electric motor (LEM). Therefore, it is optimal to select LEMs when a linear force is required [13]-[14] and rotary electric motors when a torque is required due to the minimization of lost energy during energy transfer. LEMs are commonly used in precise, high-acceleration applications like actuators and in high-speed, low-acceleration systems like electric trains. With careful design considerations, a combination of speed, thrust [15], and efficiency can be achieved to meet the application's design objectives.

1.2. Motor Slot and Pole Count - Winding Configurations

The slot and pole count of an LEM is an important electromagnetic relationship that determines the resultant magnetic field waveform found in the airgap of the motor application. Table 1.1 summarizes the effects that slots and poles have on a motor's performance. To achieve greater efficiency, the generated wave in the primary field should

2

approximate a sine wave [16]. This approach is called distributed winding and is achieved through different slot and pole combinations in the primary motor.

TABLE 1.1

SLOT AND POLE TREND DECISION-MAKING

Increase Slots	Advantage	• Better primary field sine wave approximation resulting in improved efficiency • Reduced mass due to the metal core being denser than the combination of copper and insulation in the slots
	Disadvantage	• Complex to manufacture and wind the coils • Localized saturation of the primary core if the primary tooth width is too low
Increase Poles	Advantage	• Improved operating efficiency • Reduced mass
	Disadvantage	• Increased eddy current losses • Reduced maximum speed • Reduced thrust force

The fewer pole pairs a motor has, the less drag the motor experiences and therefore the more thrust it generates [17], [18]. As the slot count of the motor increases, the discretization effect of the resultant magnetic field waveform found in the airgap decreases, approaching a sine wave resemblance. There are many combinations of winding patterns [19] when considering various slot and pole counts. It is important that there be a governing set of rules which all winding configurations abide by to ensure that each implementation is feasible and effective. A winding distribution table (WDT) serves this purpose which is

3

detailed in [20], providing a robust definition of its approach. The general principle is that the WDT balances the slot EMFs over the phases, creating symmetrical windings.

1.3. Literature Survey on Motor Modelling

When choosing a suitable modelling workflow for a motor optimization problem, many constraint considerations must be made [21]. Considering optimization efficiency, robustness, integration complexity [22], and flexibility when choosing the best modelling algorithms within a workflow proves to be challenging. Within Table 1.2, a comparison between finite element analysis (FEA), magnetic equivalent circuit (MEC), and harmonic model (HM) techniques [23], [24] is provided to aid in the decision-making process. Although FEA is often known as the ultimate modelling application due to its accurate modelling ability, it is often not an efficient medium for custom optimization problems. This is due to the lack of access to the back-end code resulting in the user having to conform to the functionality provided by the application itself. When creating custom modelling algorithms, this constraint is relieved and is often preferred when the optimization is in the intermediary development phases. Once the custom algorithm has narrowed the design space via its convergence on an optimal solution [25], then it is beneficial for modelling final designs in FEA [26]. The hybrid analytical model (HAM) is a merger of multiple modelling techniques [27], which utilize the advantages of each individual technique in regions within the model. For example, when merging MEC modelling [28] with HM, the resulting model achieves the advantages of each found in Table 1.2. This allows complex geometries to be accurately modelled using MEC regions and simple geometries with HM regions [29] to achieve greater efficiency and accuracy.

4

TABLE 1.2

MODELLING ALGORITHM COMPARISON

FEA	**Advantage**	• Modular modelling capability which can be extended to many fields of physics like electromagnetics, thermodynamics • Efficient machine code and modelling techniques produces the most accuracy within reasonable time • Accurately models both magnetic and electrical losses under transient conditions
	Disadvantage	• High computation demand for dense-mesh, transient simulations • The freedom in designing and optimizing a model may be limited to the software's capability • Difficult to automate an optimization workflow due to lack of customization
MEC	**Advantage**	• Flexible modelling methodology which can equate a large range of models within a domain • Accurately models complex geometries with a relatively dense mesh
	Disadvantage	• Discretization of the spatial domain requires dense meshing to produce an accurate solution • Errors in the solution occur near abrupt changes in source potential
HM	**Advantage**	• The computation intensity only scales with harmonics, not the number of nodes in the mesh of the region • Since the region does not require discretization, the solution can be calculated at any spatial position within the region rather than at the center of a node
	Disadvantage	• Inaccurately models complex geometries which are common in motor applications such as motor teeth and windings • Requires many harmonics to accurately predict the waveform

1.4. Induction Motor Optimization

Three categories of optimization algorithms are further sub-sectioned in Fig. 1.1 featuring gradient-based, evolutionary, and neural network algorithms. Two elite algorithms within each subset of the categories are highlighted in Fig. 1.1 which serve unique purposes in optimization workflows. The simplest of the three is the gradient-based algorithm which requires function evaluations to determine the rate of improvement towards the objective. This often leads to finding local minima and maxima rather than the global counterparts since the algorithm will return to the local minimum/maximum once the gradient direction becomes negative.
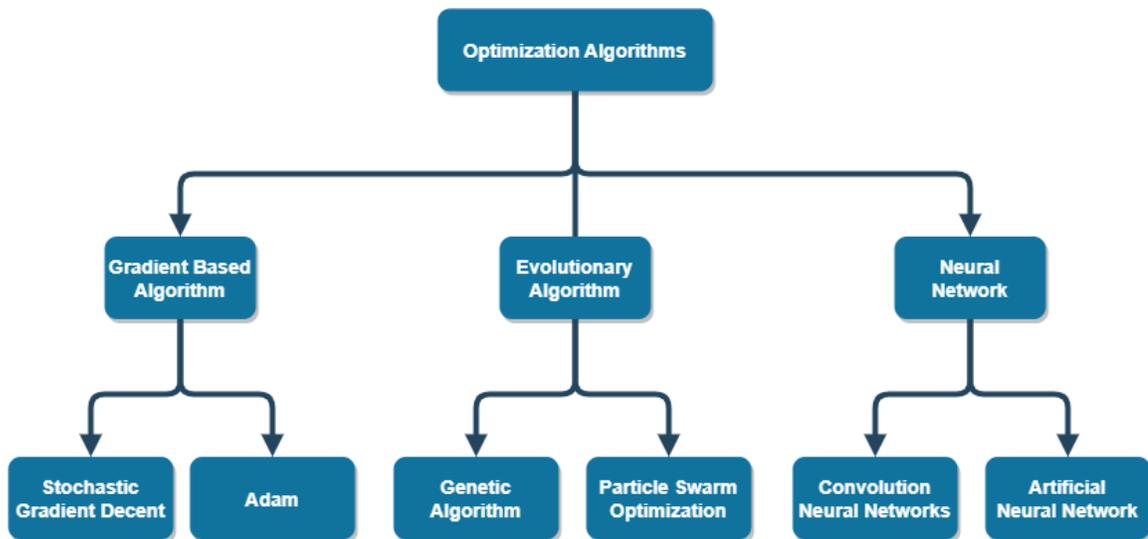
Fig. 1.1. Classification of modern optimization algorithms.

Similarly, evolutionary algorithms are heuristic and operate by generationally optimizing the population. Different methods of variation on the population have a direct correlation to the robustness of the solver's ability to converge on the global minimum/maximum. Alternatively, neural networks are model-based algorithms which compare solutions based on their prediction of the solution's future performance. However, neural networks tend to be computationally intensive, in addition to the difficulty of deciphering how the algorithm

6

determines its solution. Evolutionary algorithms find an equilibrium between the performances of gradient-based algorithms and neural networks, which makes them an attractive prospect for intermediate optimization problems.

In addition to custom implementations of optimization algorithms, there exists modern software applications which can achieve many of the same things. Examples of these are optimetrics optimization [29] and PyAEDT [30] integration within ANSYS Electronics, a modern FEA software useful for electric motor modelling. The benefit of these existing applications is that they act as algorithms wrapped around the ANSYS modelling software. The main problem with optimetrics is that iterative slot-pole combination changes the geometry of the motor model, making it hard to implement such a general application. Alternatively, "PyAEDT is a Python library that interacts directly with the AEDT application programming interface (API) to make scripting simpler for the end user" [31]. This greatly increases the flexibility of the application since Python [32] has many modules for programming [33] applications. Although this seems like the ideal implementation framework, it takes extensive effort to takes time to thoroughly investigate the feasibility of integrating a complex solution with an API like PyAEDT. Additionally, the productivity of the project would be dependent on the availability and robustness of the code base, which is non-existent for a custom implementation like the one in this thesis. Lastly, APIs often have an associated cost to make calls, which makes custom implementation desirable.

1.5. Research Motivations

Many researchers have implemented motor optimization for different design goals, for example: primary weight [34], maximizing the thrust and power to weight ratio [35], optimal winding design of LIM [36], [37]. In other research, efficiency and the power

7

factor were maximized [38], [39], in addition to the imperialist competitive algorithm implemented for SLIM design [40], [41]. The proposed solution in [42] attempts to improve upon the papers referenced above via multi-objective, genetic algorithm optimization. Although these research references improved their designs via narrow optimization, they often do not consider enough of the optimization domain.

To effectively improve motor performance, it is important to optimize motor meta parameters before tuning lower-level parameter optimizations. Meta parameters are the first design considerations when designing a new motor which has a rippling effect on other motor parameters. Motor slot and pole counts are possible meta parameters because when changed, the winding configuration and the motor core geometry need to be revised. From the performance trends found in Table 1.1, it is not intuitive to predict an optimal slot-pole combination due to the advantages and disadvantages carrying similar design weight. It is known that a greater slot count improves the efficiency and reducing the poles improves the thrust force [43]. An intuitive guess would predict that a 36 slot, 2 pole motor produces the best motor performance. Although the solver may trend towards a solution like this, there are disadvantages to a design at this extreme of the motor domain. For this reason, it is important to introduce an iterative evolutionary algorithm which may predict better designs than a manual design workflow can produce. After simulating multiple feasible motor models, the solver shall navigate the motor domain enough to observe correlations between the slot-pole combination and the performance parameters.

1.6. Research Objectives

Optimizing across an entire sub-category of LIMs is achievable using HAM and an optimization workflow described in Chapters 2 and 3, respectively. It is not sufficient to

8

produce a motor that outperforms its alternatives in one performance parameter while sacrificing other performance parameters. Therefore Pareto-optimal solutions that outperform the alternative motors in most performance parameters, if not all, are required. To determine that motors produced via the optimization process are feasible in all aspects, including consequential constraints such as saturation effect [44], robust feasibility checks are added to the solver (detailed in Chapter 4.2). To ensure the accuracy of the model, the results are objectively compared to FEA via transient electromagnetic analysis in the software ANSYS Electronics.

1.7. Research Contribution and Deliverables

From previous LIM optimization research highlighted in Section 1.5, it is feasible to produce significant improvement in one performance parameter, such as weight, thrust, efficiency, and power factor. However, when conducting narrow optimization, there is unrealized potential. **In this thesis a novel holistic model is proposed which includes flexible motor modelling, modified constraints, multi-objective optimization, and field plotting to serve as a design tool to automate producing optimal motor designs within their constrained domain**. The expected deliverables include:

- A custom optimization workflow that is configurable and expandable for future optimization studies.

- Produce the optimal motor within the subclass of double-layer, single-sided, 3-phase, integral slot winding, LIM.

- Less than 5% HAM modelling error compared to ANSYS Electronics FEA simulation results.

9

The modelling technique for HAM is complex to integrate and program while achieving efficient execution times of the program written to optimize this motor problem. When using Python 3, the extensive libraries related to optimization and data collection make it a prime medium for tying all pieces of the optimization workflow together. Using SciPy for the system of linear equations, Platypus-opt for the optimization algorithm, and the graphical framework Tkinter for data visualization allows for field plots and transient responses to be visualized within the application.

1.8. Organization of Thesis

The major sections of this thesis are as follows:

1) Chapter 1 provides an overview of linear electric motors and the use of optimization algorithms with induction machines, demonstrating the motivations, challenges and objectives associated with the proposed method from a vehicle level to the motor level and the incorporation of the algorithm level.

2) The baseline double-layer single-sided linear induction motor considered for optimization is introduced in Chapter 2, outlining its performance parameters and constraint considerations. The modelling methodology to be implemented on the motor is discussed in detail within this chapter.

3) This chapter includes a case study on the proposed optimization algorithms within the class of evolutionary algorithm, conducted to determine the optimal multi-objective algorithm for implementation on the modelling algorithm described in Chapter 2.

4) Chapter 4 serves to elaborate on the integration of the chosen optimization algorithm from Chapter 3 with the modelling algorithm from Chapter 2. The efficiency and robustness of the solver as well as solver configuration constraints will be discussed.

5) Chapter 5 summarizes the results generated through the proposed method and identifies the future scope of the proposed research and developed method in IMs and algorithm-based IM optimization.

CHAPTER 2   Hybrid Analytical Model

The concept of modelling domains to predict the behaviour of materials and waveforms has drastically improved one's ability to rapid prototype designs with a significant reduction in cost. To achieve this efficiency in the design phase of any project, a modelling algorithm is required that can accurately and timely predict the domain through a system of equations. Finite element analysis (FEA) is generally a good application of modelling and has been implemented across topics like fluid dynamics, wave propagation, thermal transport, and generally anything that can be governed by a system of mathematical equations. Although FEA is the standard for accuracy, it can be time and computationally intensive, leaving researchers with optimization strategies needing more specific solutions. To achieve this capability, it is important to understand the fundamentals of the boundaries that constrain the domain and the equations that govern the space, defined by the problem. This section will define the baseline motor used as a reference to compare future motor solutions against and elaborate on the HAM functionalities [45] required to achieve a processed solution after solving the system of linear equations defined by the pre-processed model.

2.1. Base Model

The base model found in [43] is a double-layer single-sided LIM and acts as ground truth for the HAM model reproduced in this thesis. Since the reference also used the HAM, and its results were within 1.7% compared to FEA, it is an acceptable model for the application of this thesis. Additionally, the base model was manufactured and physically tested for

operating conditions and error in HAM results. The specific spatial, electric/material and model properties of the baseline motor are tabulated in Tables 2.1-2.3.

TABLE 2.1

BASELINE SPATIAL MOTOR PARAMETERS

Pole pitch (mm)	y_p	45
Tooth width (mm)	w_t	6
Slot width (mm)	w_s	10

To account for transverse end-effects in the motor, the aluminum plate conductivity was reduced accordingly in Table 2.2.

TABLE 2.2

BASELINE ELECTRICAL AND MATERIAL MOTOR PARAMETERS

Number of slots	N_s	16
Number of poles	N_p	6
Number of turns per coil	N_t	57

The node elements contained in L rows and K columns presented in Table 2.3 accurately models the magnetic field in the primary of the motor and in the surrounding air.

TABLE 2.3

BASELINE MODEL PARAMETERS

Number of rows in a MEC region	L	53
Number of columns in a MEC region	K	576

13

To validate a model's accuracy, the resulting magnetic flux density in the airgap will be compared against the steady state [43] FEA solution in Fig. 2.1 and Fig. 2.2. This will be done in ANSYS Electronics using the same configuration found in Tables 2.1-2.3.
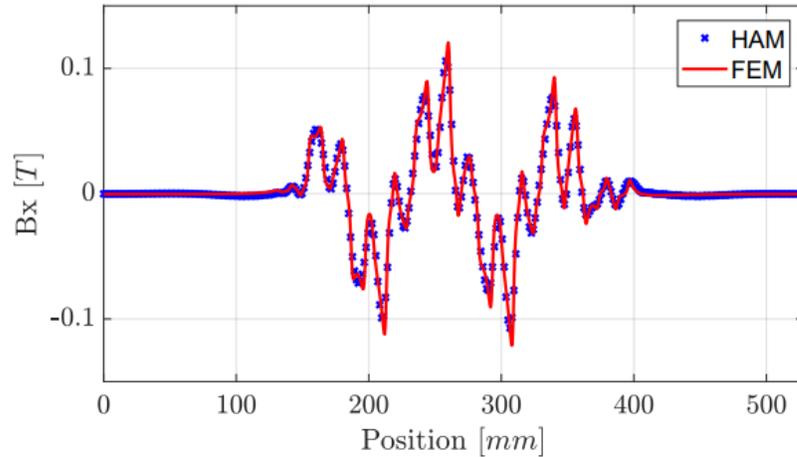
Fig. 2.1. Magnetic flux density in the middle of the airgap in the normal direction comparing the proposed model results against FEA using ANSYS Electronics (I_p = 10 A, v = 0 m/s, f = 100 Hz) [43].
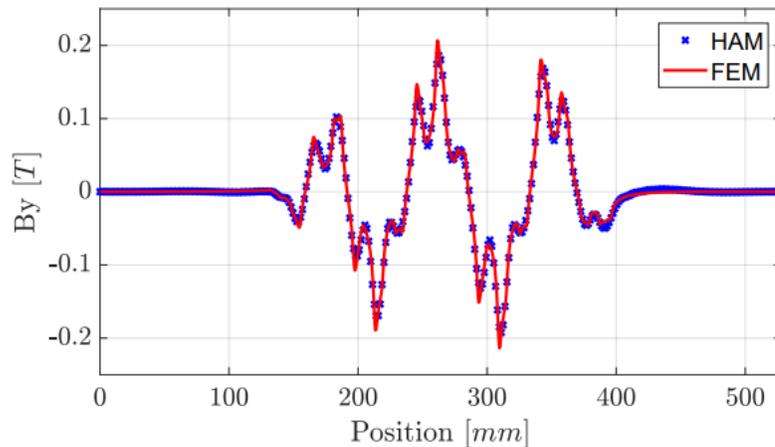
Fig. 2.2. Magnetic flux density in the middle of the airgap in the tangential direction comparing the proposed model results against FEA using ANSYS Electronics (I_p = 10 A, v = 0 m/s, f = 100 Hz) [43].

2.2. Model Relationships

There are many important relationships between the motor parameters, which can be utilized to assign ratios between variables. Creating relationships between variables allows for more flexibility in the model, which tends to produce feasible motor designs. This is an

14

important step which constrains the complexity of the optimization space while improving its robustness. A summation of the individual slot and tooth lengths produces (2.1) to calculate the length of the LIM primary. Since the length of the motor primary is a constant design parameter, it is useful to determine the slot and tooth widths from a given length input with a varying slot input.

$$L_p = \lambda \, (N_s - 1) + w_s + 2w_{t_{end}} \tag{2.1}$$

$$\lambda = w_s + w_t \tag{2.2}$$

One important relationship is between the slot and tooth width of the primary core. Since saturation degrades the motor performance in teeth that do not provide enough volume for the flux, the tooth width must not be too small in a motor design. Alternatively, the slot width should not be made too small, producing unrealized potential. A relationship where the tooth width is 60% the width of a slot produces a ratio that will work for a large range of motors [52] that vary in their slot and pole combination.

$$w_t = \frac{3}{5} w_s \tag{2.3}$$

Increasing the width of the end teeth helps alleviate some of the end-effects [46], [47] by capturing more of the magnetic field. Consequently, the overall thrust produced increases due to the addition of more active surface area to the end teeth, which tend to saturate faster than internal motor teeth. Using the relationship in (2.4), the end tooth width is equal to the slot width.

$$w_{t_{end}} = w_s \tag{2.4}$$

The substitution of (2.2)-(2.4) into (2.1) results in only one unknown variable, w_s. After factoring and isolation, the slot width is solved in (2.5), which can then be substituted back into (2.2)-(2.4) to solve the relationships for tooth width, slot pitch, and end tooth width.

15

$$w_s = \frac{L_p}{\frac{8}{5}(N_s - 1) + 1 + 2} \tag{2.5}$$

Through this approach, the modelling of the motor maintains a constant primary length while varying the motor configuration for varying slot counts. Although these relationships are necessary for the model implementation, they are not optimal in the final motor design. Further low-level motor optimization must take place to fine tune the LIM primary geometry, which will maximize the flux in the core without producing saturation. Following the robust relationship between slot count and motor geometry, the relationship between magnetic poles and the motor performance is defined in (2.6). For a constant frequency the velocity is proportional to the pole pitch which is approximated in (2.7) for a given motor primary length. Since the motor application demands a high operational velocity the pole pitch shall be maximized.

$$y_p = \frac{L_p}{N_p} \tag{2.6}$$

$$v = 2fy_p \tag{2.7}$$

All motors windings considered in this thesis are double-layer, demanding a flexible relationship for the coil pitch. In most cases, the coil pitch is defined by the relationship to the pole pitch defined in (2.8) for a shortened pitch employed for the limitation of both the 5th and 7th harmonics).

$$y_c = \frac{5}{6}y_p \tag{2.8}$$

The number of turns per coil is defined in (2.9) related to the current density, slot area, and peak current.

$$N_t = \frac{0.6Jw_sh_s}{2I_p} \tag{2.9}$$

16

The slot fill factor in the numerator was set to be a constant value of 0.6 [52] to ensure that the ratio of the copper wire and insulation is balanced. The number of winding layers in the denominator was set to be a constant value of 2.

2.3. Winding Distribution Table

Upon every iteration of the optimization loop, a slot-pole combination will be provided to the objective function which is subject to the model relationships within Section 2.2 and then assigned to the HAM to solve for performance parameters. To handle a wide variety of integral slot winding patterns and maintain geometric integrity, the WDT [19] is implemented as the formulation for the winding configuration. To implement a motor winding on the generic formulation described in Table 2.4, a few more variables must be defined for a double-layer, 3-phase, integral slot winding pattern.

TABLE 2.4

ORDER OF THE WDT ELEMENTS [19]

	Col 1	Col. 2	…	Col. n_c
Row 1	1	2	…	n_c
Row 2	$n_c + 1$	$n_c + 2$	…	$2n_c$
Row 3	$2n_c + 1$	$2n_c + 2$	…	$3n_c$
⋮	⋮	⋮	⋱	⋮
Row m	$(m-1)n_c + 1$	$(m-1)n_c + 2$	…	mn_c

17

The number of WDT columns of Table 2.4 are defined in (2.10).

$$n_c = \frac{N_s}{m} \tag{2.10}$$

In summary, the WDT is composed of rows equal to the number of phases and columns equal to the number of slots divided by phases. In case the winding has multiple layers, each cell in the table has their opposite negative (or positive) coil terminal defined by the coil pitch y_c. Since this optimization problem is constrained to non-reduced motors, the rows on the right half of Table 2.5 are shifted down.

TABLE 2.5

WDT OF SAMPLE MOTOR

Coil Direction	Upper Layer				Lower Layer			
	+	+	-	-	-	-	+	+
Phase A	1	7	2	8	4	10	5	11
Phase B	3	9	4	10	6	12	7	1
Phase C	5	11	6	12	8	2	9	3

Rotary motor winding patterns rarely require empty slots due to the continuity of the motor core along the radial direction. This is not true for linear motors due to the linear core. When a linear motor is wound like a rotary motor, there is often a coil which has coil terminals on each end of the motor that produces an imbalance in the per-phase winding resistance which must be avoided. The WDT allows for empty slots to be included in the

formulation of the winding pattern but is omitted in this thesis to reduce the complexity required to produce winding patterns for an entire domain of motors.

2.4. Hybrid Analytical Model Structure

Linear motors that have a flat primary core are naturally formulated in rectangular coordinate systems due to their rectangular-like shape. Before a motor model can be solved, a mesh of rectangular nodes is initialized for the motor geometry by discretizing the model and prioritizing the motor core geometry. Since the slot and coil geometries are generally the most complex, the mesh density in the x and y direction is proportional to the complexity of the core shape. The HAM is an optimal application for this mesh complexity as it merges the benefits of both MEC and harmonic modelling. Within Fig. 2.3, the division of the model into unique regions through continuous and non-continuous boundaries is realized. For the MEC region, variable k will be used for the node index in the x-direction, while l is the node index in the y-direction. The finite index limits for these two index vectors are defined as K and L, where their product results in the total number of node indexes M. The lengths of a node in the x and y direction defines the dimensions of the rectangle nodes throughout the mesh, i.e., the length in the MEC region is constant along the l-direction with a constant k and vice versa. To maintain periodicity in the x-direction, the nodes on the x-boundaries where $k = 1$ and $k = L$ are coupled, which is elaborated in Table 2.6. A single MEC node element is visualized in Fig. 2.4 to aid in the fundamental function of MEC mesh of nodes.
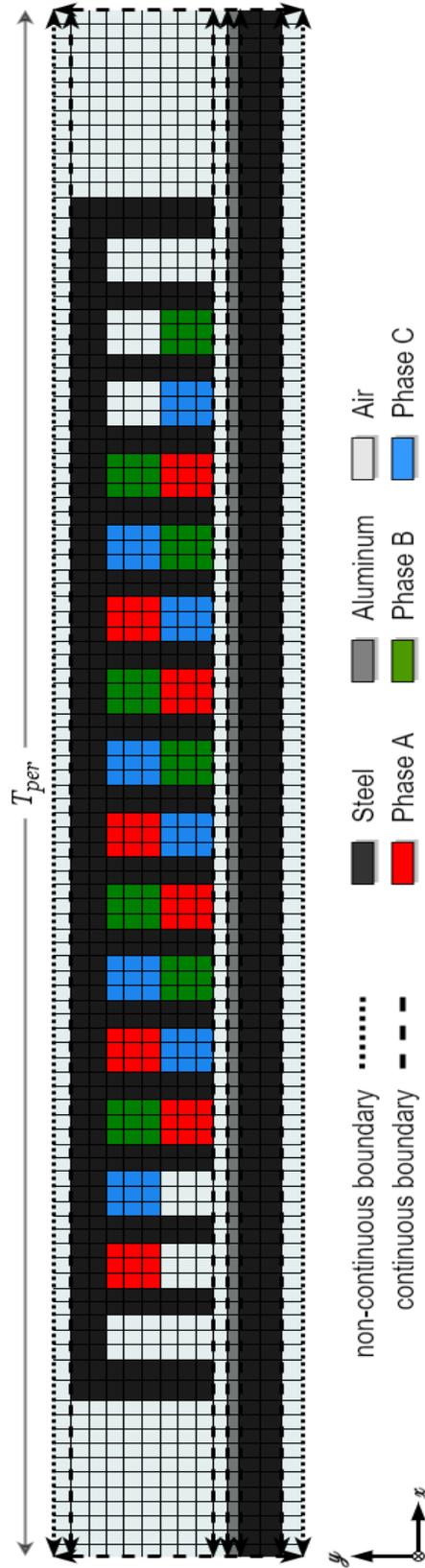
Fig. 2.3. Meshed motor model containing boundary conditions.

TABLE 2.6

MEC NODE INDEX CONTINUITY

Current Node x Index	Left Node x Index	Right Node x Index
$k = 1$	$k = L$	$k = 2$
$k = L$	$k = L - 1$	$k = 1$
$1 < k < L$	$k = k - 1$	$k = k + 1$
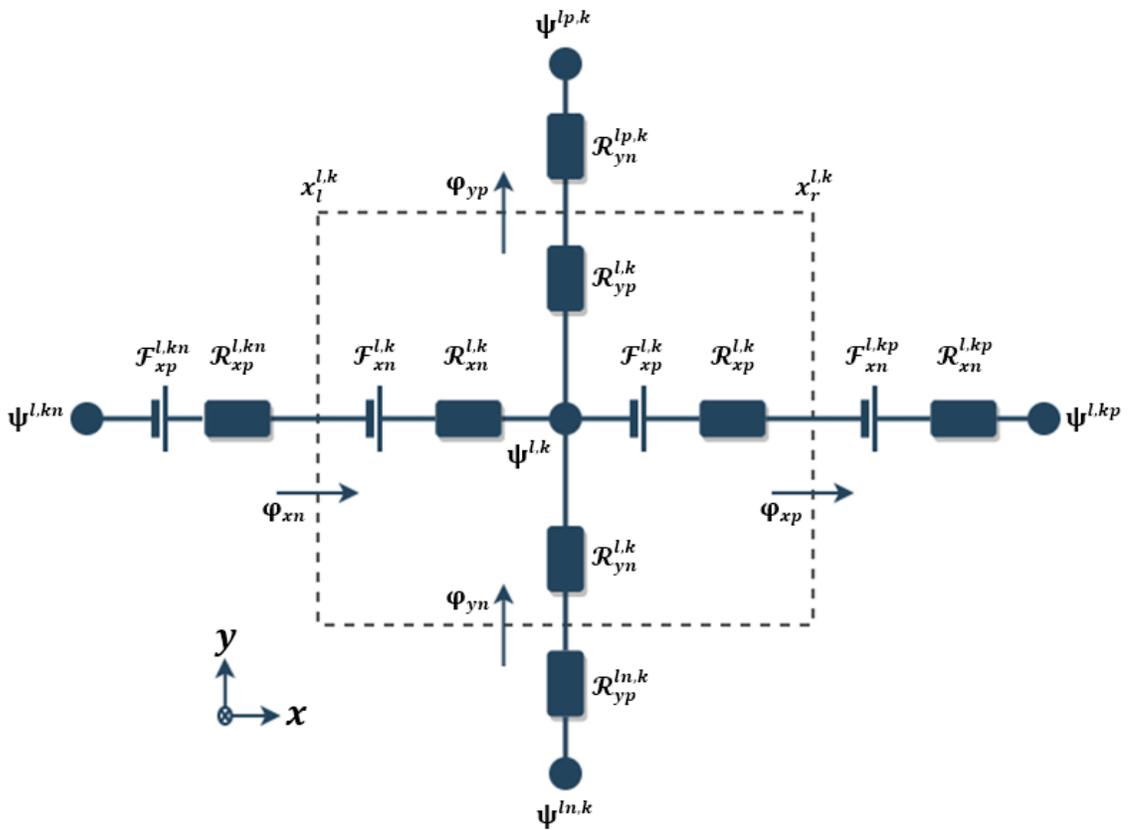
Fig. 2.4. Single MEC node element in a mesh.

Now that the size and density of the mesh has been defined, it is important to define the

properties of each individual node within the mesh. Each node has a reluctance, flux, and

MMF component, which is defined by the material the node encloses. To avoid cluttering the image of Fig. 2.5, the index annotations (seen in Fig. 2.4) were omitted.
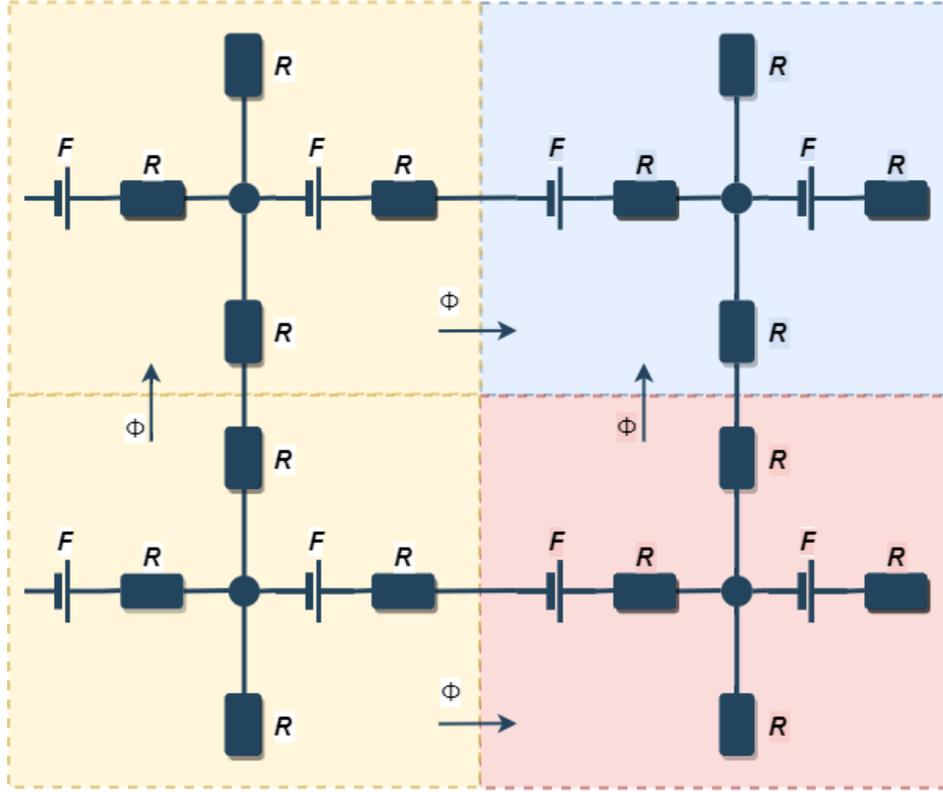
Fig. 2.5. MEC modelling of 4 arbitrary neighbouring nodes within 3 different arbitrary materials (yellow, blue, red).

The conservation of flux is maintained in (2.11), stating that all flux entering one potential node $\psi(l, k, t)$ should be equal to the magnetic flux leaving the node [43].

$$\Phi_{xn}(l, k, t) + \varphi_{yn}(l, k, t) = \varphi_{xp}(l, k, t) + \varphi_{yp}(l, k, t) \tag{2.11}$$

An array of the unknown potential $\psi(l, k)$ produces the ψ section within matrix X of Table 2.7. The source terms producing the flux are the MMFs generated by the coils defined as $\mathcal{F}(l, k, t)$ contained in a node. An array of these constants produces the B_{KCL} section within matrix B of Table 2.7, whereas an array of coefficients produces the A_{KCL} section within matrix A of Table 2.7.

Due to the merger of MEC and HM, the unknown variable for the potential of the node arises in the flux. It is an unknown variable that requires a system of linear equations to solve, which is discussed in detail within Section 2.5. To calculate the HM regions, the equations for magnetic flux density and magnetic field strength materialize in the form of a complex Fourier series. The equation parameters change from l, k, t in the MEC equations to x, y, t since the HM does not require discretized points and is solvable for any x, y coordinate in the model. Since the MEC model determines the mesh density, the HM model follows suite and is calculated at the center of a node for a processed solution. To couple to MEC regions, the HM region has unknown variables a_n and b_n, which are solved in a system of linear equations like the unknown MEC term $\psi(l, k)$.

These equations produce the solution of the magnetic flux density for one periodical length τ_{per} and N spatial harmonics, where one space harmonic is defined as n. Since the HM region contains the same material throughout the region, the value μ_r, and σ are independent of a node index within the region. The relative velocity between the primary and secondary is defined as v.

2.5. System of Linear Equations

Now that the required mesh parameters have been defined, the construction of the system of linear equations relating to the unknown variables can begin. The boundary condition between two neighbouring regions can be between two HM regions, between MEC and HM regions, or it can be non-continuous. This classification defines which unknown variables are included in the equation. Since sources cannot be infinite in magnitude and the air surrounding the model theoretically extends to infinity, the Dirichlet boundary

23

condition applies, forcing all the field components to vanish at the boundary. This equation applies to regions I, VI. For continuous boundaries, defined in (2.12)-(2.14), the normal and tangential components of each neighbouring region must be conserved. This is true for HM-HM boundaries as well as HM-MEC boundaries. Where i is the lower region index at the boundary positioned at y_{BC}.

$$B_{yi} = B_{y(i+1)}|_{y=y_{BC}} \tag{2.12}$$

$$H_{xi} = H_{x(i+1)}|_{y=y_{BC}} \tag{2.13}$$

$$\frac{1}{\mu_r}B_{xi} = \frac{1}{\mu_r}B_{x(i+1)}|_{y=y_{BC}} \tag{2.14}$$

The HM-MEC boundary must be expanded upon to couple the Fourier and MEC solutions. Unlike the MEC region, the HM regions do not produce a source. This means that the transfer of energy into the HM region is conserved at the HM-MEC boundary. Both sides of the equation are in the form of a complex Fourier series with N harmonics. Discretizing the coils into nodes of a mesh creates a staircase shaped waveform, indicating that the Fourier series needs to be modified for a piece-wise continuous function value. Some of the variables that help solve for the function value depend on the position of the node at the index (l, k). To produce a processed mesh model [45], the equations for each boundary condition are separated into a matrix of coefficients A, a matrix of unknown variables X, and a matrix of constants B. Table 2.7 expands on the matrix equation $AX = B$:

TABLE 2.7

SYSTEM OF LINEAR EQUATIONS SOLVING FOR UNKNOWN VARIABLES

$$
\left|
\begin{matrix}
A_b^1 \\
A_t^1 \\
A_b^2 \\
\cdot \\
\cdot \\
A_b^{i-1} \\
A_{MEC}^{i-1} \\
A_{KCL} \\
A_{MEC}^{i+1} \\
A_t^{i+1} \\
A_b^{i+2} \\
\cdot \\
\cdot \\
A_t^L
\end{matrix}
\right|
\left|
\begin{matrix}
X^1 \\
X^2 \\
\cdot \\
\cdot \\
\cdot \\
X^{i-2} \\
X^{i-1} \\
\psi \\
X^{i+1} \\
X^{i+2} \\
\cdot \\
\cdot \\
\cdot \\
X^L
\end{matrix}
\right|
=
\left|
\begin{matrix}
B_b^1 \\
B_t^1 \\
B_b^2 \\
\cdot \\
\cdot \\
B_b^{i-1} \\
B_{MEC}^{i-1} \\
B_{KCL} \\
B_{MEC}^{i+1} \\
B_t^{i+1} \\
B_b^{i+2} \\
\cdot \\
\cdot \\
B_t^L
\end{matrix}
\right|
$$

where the dimensions of the square matrix A are $\beta(2N) + \gamma(M)$, where β is the total number of HM regions in the model and γ is the number of MEC regions in the model. M is defined as the number of nodes in the MEC region, and N is defined as the number of harmonics n in the waveform approximation. The dimensions of the column vectors X and B are $\beta(2N) + \gamma(M)$. To optimize the system of linear equations, the equations and coefficients that are solvable in the pre-processing stage can be removed. In the Dirichlet equations, an infinite position drives the unknown coefficients to ∞ and 0 for a_n and b_n, respectively. These equations can be removed from the equation set along with the b_n^1 and a_n^L unknown variables. The removal of $2N$ equations and $2N$ unknown variables maintains a square matrix A, which has the new dimensions of $(\beta - 1)(2N) + \gamma(M)$. The system of

linear equations is then solved using lower-upper-decomposition to produce the unknown variables of the HM and MEC regions.

2.6. Processed Model

Now that the unknown variables for the HM and MEC regions are solved, their values can be substituted into (2.11) to solve for processed mesh parameters. An important performance parameter used in the genetic algorithm (GA) objective function is the thrust of the motor. The force on the primary of the motor has a tangential (2.15) and normal (2.16) component which can be calculated with the equations below:

$$F_x = \frac{-L_s}{\mu_o} \int_0^{\tau_{per}} \left[B_{xn}(x,y,t) B_{yn}^*(x,y,t) \right] dx \tag{2.15}$$

$$F_y = \frac{-L_s}{2\mu_o} \int_0^{\tau_{per}} \left[B_{xn}(x,y,t) B_{xn}^*(x,y,t) - B_{yn}(x,y,t) B_{yn}^*(x,y,t) \right] dx \tag{2.16}$$

These equations were derived from the Maxwell stress tensor [48], [49] in the airgap where the complex conjugate of a complex variable is denoted with a * in the superscript. These post processing parameters are useful to create field plots on the mesh of the motor to highlight the performance at various operating conditions. For example, the saturation effect can be determined by plotting the magnetic flux density on the motor core and determining if the magnetic flux density has exceeded the capability of the motor core material. The equations described in this chapter depend on the number of harmonics considered in the Complex Fourier Series approximation MEC discrete waveform at the boundary conditions. To highlight this effect, Fig. 2.6 shows the discrete plot of the magnetic flux density in the air gap in black.
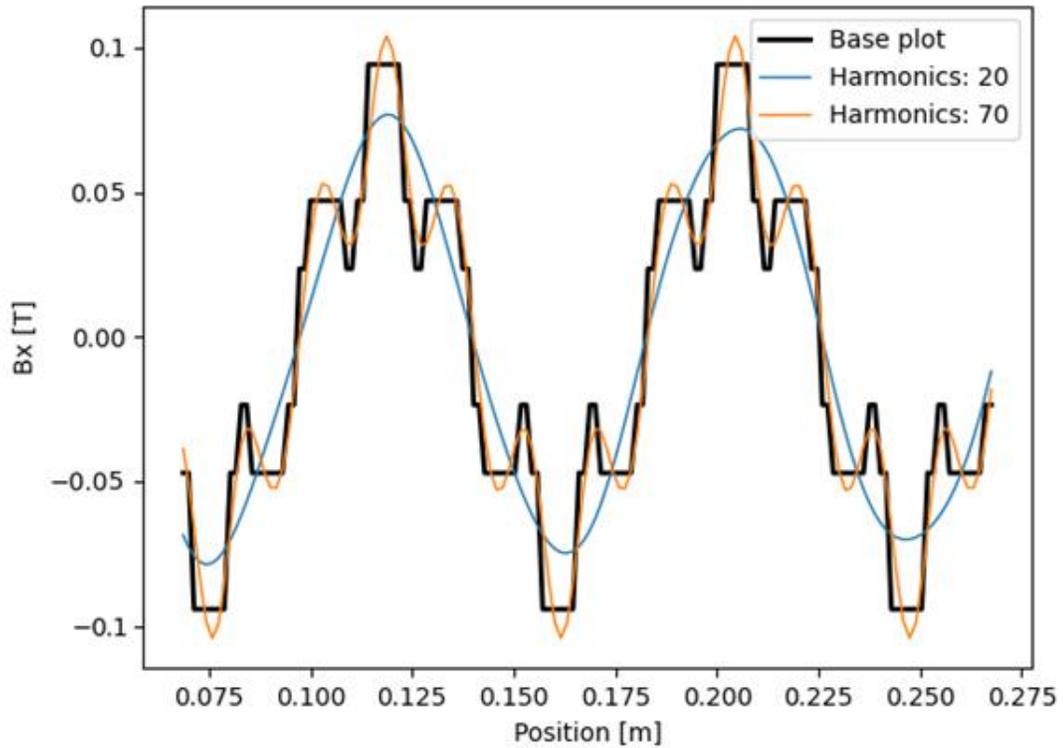
Fig. 2.6. Complex Fourier Series approximation of the discrete magnetic flux density at the center of the airgap for various harmonics.

Two Complex Fourier Series approximations are overlayed on the base plot to show the improvement in approximation accuracy when more harmonics are considered [50]. Visually the orange plot that contains 70 harmonics in the series can capture the peaks of the waveform much better than that of the blue plot that contains only 20 harmonics. Although an increase in harmonics correlates with approximation accuracy, it dramatically increases the computation complexity and should, therefore be appropriately chosen.

CHAPTER 3   Optimization Algorithm

Within the scope of evolutionary algorithms, the GA and particle swarm optimization (PSO) are the dominant algorithms when the problem demands robustness and performance. With the overarching objective of integrating the optimization algorithm with the HAM, the comparison between PSO and GA must be carefully considered to ensure that the chosen solver can meet the unique demand of having the HAM as its objective function. In this section, the core functionality of each algorithm will be discussed and then compared against one another in a case study to statistically determine the optimal solver for the problem.

3.1. Genetic Algorithm

The GA is a kind of evolutionary algorithm that mimics the general concept of evolution. Natural selection is often mentioned in the context of evolution since it is the strong individuals that survive in each environment. Being the strongest is a generalization that is defined by the objective function applied to the optimization problem. The structure of a population subject to the GA encapsulates a fixed number of chromosomes, which themselves encapsulate genes, visualized in Fig. 3.1.
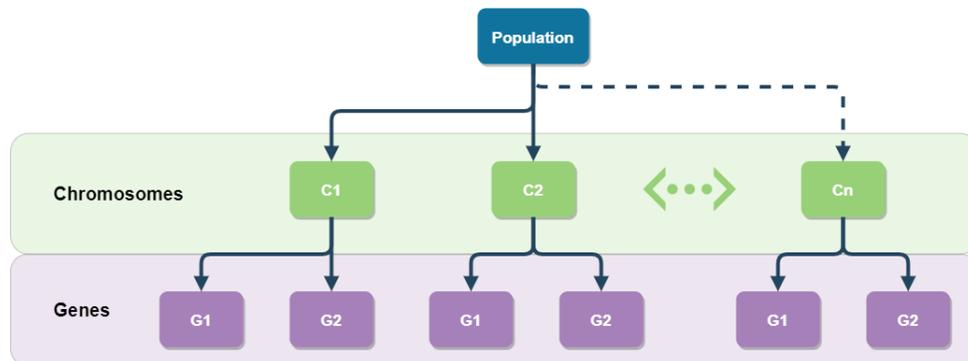
Fig. 3.1. Layout of a genetic algorithm with an arbitrary number of chromosomes and genes per population.

To understand the function of a gene, the application of the algorithm must be defined since the genes are merely input variables to the model that requires solving. If the optimization problem were a 2-dimensional surface plot minimization, the inputs to the model would be an arbitrary 2-dimension coordinate. Each dimension of this coordinate is considered a gene using the nomenclature of the GA.
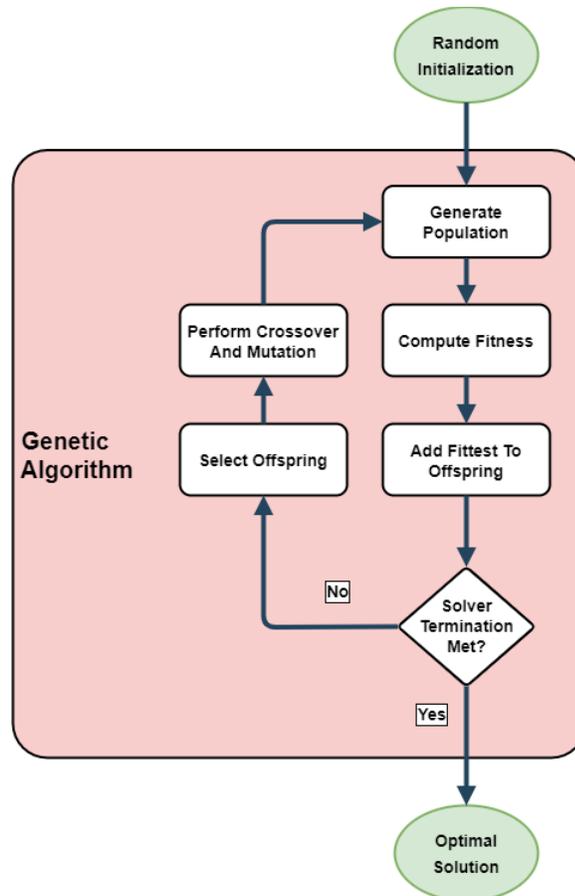
Fig. 3.2. Layout of a genetic algorithm execution loop.

Throughout each iteration of the solver, a new population is produced via selection, crossover, and mutation. This iterative loop ensures that the algorithm favors the desirable solutions while maintaining robustness through some degree of randomized search throughout the optimization domain.

3.2. Particle Swarm Optimization

Like GA, the PSO mimics the natural phenomenon of the power of a collective. Instead of the population, chromosomes, genes, and offspring nomenclature, the PSO uses swarm size, particles, and leaders. The optimization loop of the PSO operates by updating velocities and positions per particle in the swarm as elaborated in (3.1) and (3.2).

$$
\begin{aligned}
v_{ij}(t+1) = wv_{ij}(t) + r_1 c_1 \left(x_{min_{ij}}(t) - x_{ij}(t) \right) \\
+ r_2 c_2 \left(G_{min_j}(t) - x_{ij}(t) \right)
\end{aligned}
\tag{3.1}
$$

$$
x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t)
\tag{3.2}
$$

The current and successive iterations are denoted as t and $t+1$ respectively, where the local $x_{min_{ij}}$ and global G_{min_j} best solutions are determined prior to updating positions x_{ij} and velocities v_{ij}. The inertial weight coefficient w, local weight coefficients c_1 and r_1, and global weight coefficients c_2 and r_2 are integral in determining the relative influence the swarm has on the particle and vice versa. The ranges for these coefficients are defined in Table 3.1 to provide transparency in the objective comparison of algorithms in Section 3.3.

TABLE 3.1

PSO VELOCITY AND POSITION COEFFICIENTS

Constant	Range
R	$\in [0.0, 1.0]$
C	$\in [1.5, 2.0]$
W	$\in [0.1, 0.5]$

Referring to the optimization loop, the final step before calculating the objective function on the updated particles is to subject each particle to a mutation algorithm with a designated probability that the mutation executes. This varies the swarm and increases the robustness of the solver to avoid convergence on local minima and maxima.

3.3. Schwefel Function Minimization Case Study

A case study was conducted to determine the optimal optimization algorithm among the subset of evolutionary algorithms through the Schwefel test function. A test function is used to test the ability of an optimization algorithm to converge on a solution that is the global maximum or minimum rather than the function's local maxima or minima. The Schwefel function was chosen since it has a plethora of local maxima and minima, which can stall solvers prior to converging on the solution. Fig. 3.3 contains a 3-dimensional Surface plot of the Schwefel function.
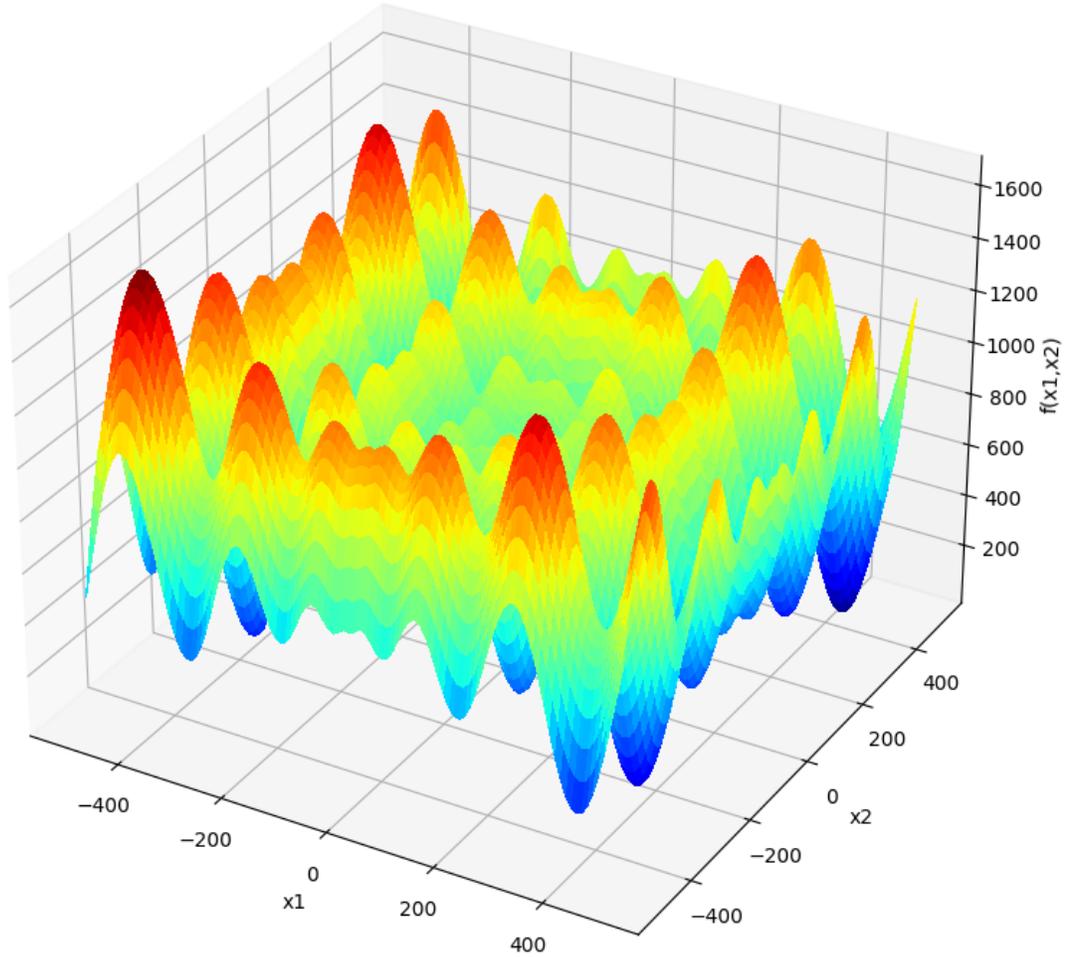
Fig. 3.3. Surface plot of the Schwefel function on the $x_{1,2} \in [-500,500]$ input range.

The function for Fig. 3.3 is defined in (3.3)

$$f(x) = 418.9829d - \sum_{i=1}^{d} x_i sin\left(\sqrt{|x_i|}\right) \qquad (3.3)$$

In (3.3), variable d is the number of input dimensions and x_i is the function input per dimension i. The global minimum $f(x^*) = 0$ is located at $x^* = (420.9687, ..., 420.9687)$ inside of the hypercube $x_i \in [-500,500]$ for all $i = 1, ..., d$. Creating a contour plot of (3.3) produces Fig. 3.4 and will be used to visualize the solvers exploring the search space later in this section.

32
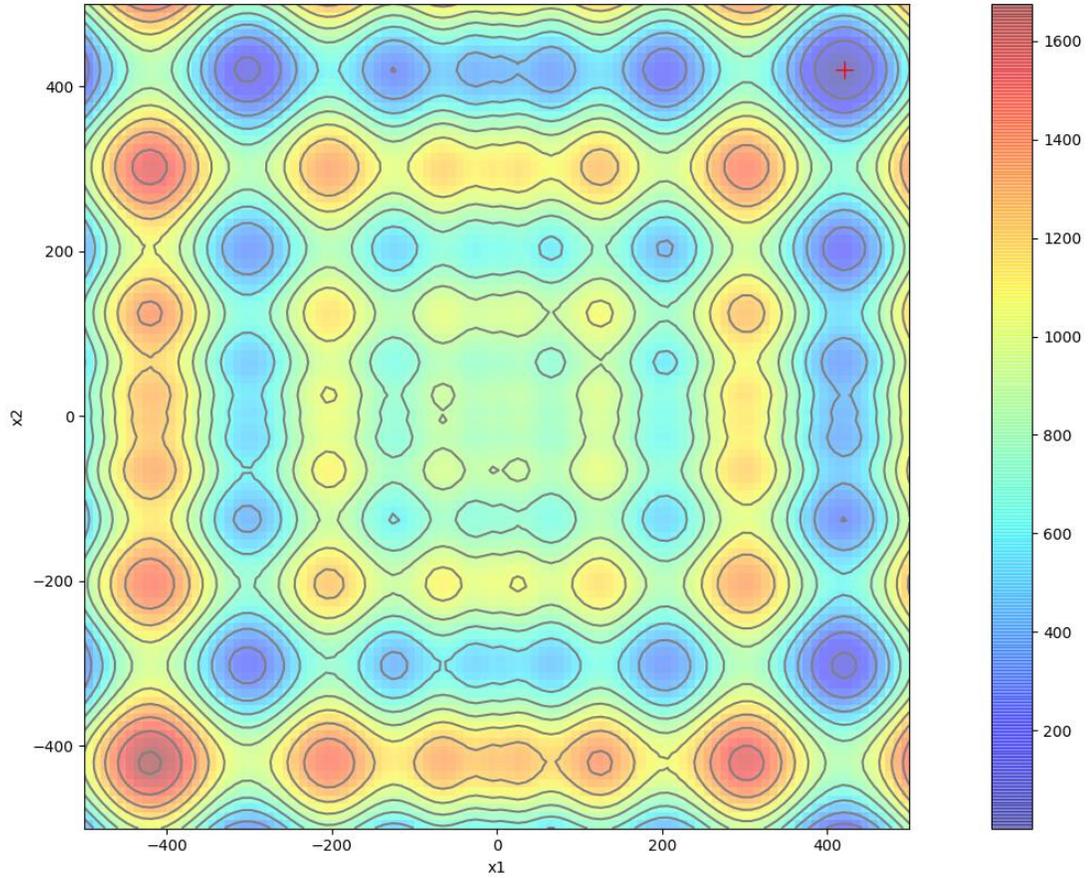
Fig. 3.4. Contour plot of the Schwefel function on the $x_{1,2} \in [-500,500]$ input range highlighting the global minimum with a red cross.

To couple a solver to this test function, a new input x^* is generated by the solver per iteration. These inputs are used to calculate and minimize the objective value through the Schwefel function until convergence on a solution. To ensure that each optimization algorithm is fairly compared in this case study, common solver parameters are used to configure each algorithm which can be found in Table 3.2. Every algorithm will iterate over its population or swarm with the only solver termination criteria being the max number of stall iterations reached. Other solver termination criteria, like reaching objective tolerance, timeout, and maximum iterations, were omitted in this case study to isolate each solver through a consistent test domain. Additionally, the optimization process is

conducted 5 times per algorithm to determine the average performance to ensure that an outlier does not significantly impact the decision making.

TABLE 3.2

OPTIMIZATION ALGORITHM CONFIGURATION

PSO		GA	
Population/Swarm Size	200	**Population Size**	200
Max Leader Size	100	**Offspring Size**	100
Comparator Key	Objective Value	**Crossover Percentage**	30%
Mutation Percentage	10%	**Mutation Percentage**	10%
Algorithm Stall Iterations	25	**Algorithm Stall Iterations**	25
Global Upper Bound	[500, 500]	**Global Upper Bound**	[500, 500]
Global Lower Bound	[-500, -500]	**Global Lower Bound**	[-500, -500]

Table 3.3 compares the evolutionary algorithms: PSO, and GA through performance parameters like execution time and error. The solver robustness is the principal performance parameter, while the solver time holds less value as a performance parameter.

34

TABLE 3.3

AVERAGE OPTIMIZATION ALGORITHM RESULTS

Algorithm	PSO	GA
Time (s)	1.5246	1.4546
Objective Function Executions	10760	4171
Solver Iterations	57	116
Value of X1 Solution	420.9728	420.9522
Error of X1 Solution (%)	0.5053	1.6543
Value of X2 Solution	420.9669	420.9729
Error of X2 Solution (%)	0.1810	0.4172
Value of Final Objective	0.0001	0.0002
Error of Final Objective (%)	0.0052	0.0195

From the data found in Table 3.3, it is evident that both the GA and PSO converge on the global minimum across 5 trials. The error in the final coordinate and the error in the resulting objective value at the coordinate were considerably low, although GA was not able to search the peaks as well as PSO. The characteristics of the algorithm's ability to search the space can be visualized by plotting the swarm or population for a given solver

35

iteration. A comparison between GA and PSO searching the space on the contour plot shown in Fig. 3.4 is achieved by selecting the early iterations of each solver. This comparison is found in Table 3.4, which highlights the meta differences between GA and PSO. The GA tends to cluster in the minima that it finds after the first iteration and spawn offspring that allows it to search those minima further. This continues until the population produces enough generations at better minima, reducing the number of offspring centered around the local minima. Contrasting this with PSO, the swarm finds the global optimal solution after the first iteration and begins to orient the velocities towards the swarm's global minimum, which is different from the domain's global minimum. When particles find other local minima, they will slightly affect the swarm's orientation unless it is the swarm's new global minimum. In this case, the swarm begins to reorient towards this point with much greater influence.

TABLE 3.4

ALGORITHM CONVERGENCE VISUALIZATION

Solver	Best Objective	Iteration 1
GA	Value: 166.0198 Coordinate: (-284.4012, 428.5994)	
PSO	Value: 128.4705 Coordinate: (-296.7353, 427.7533)	
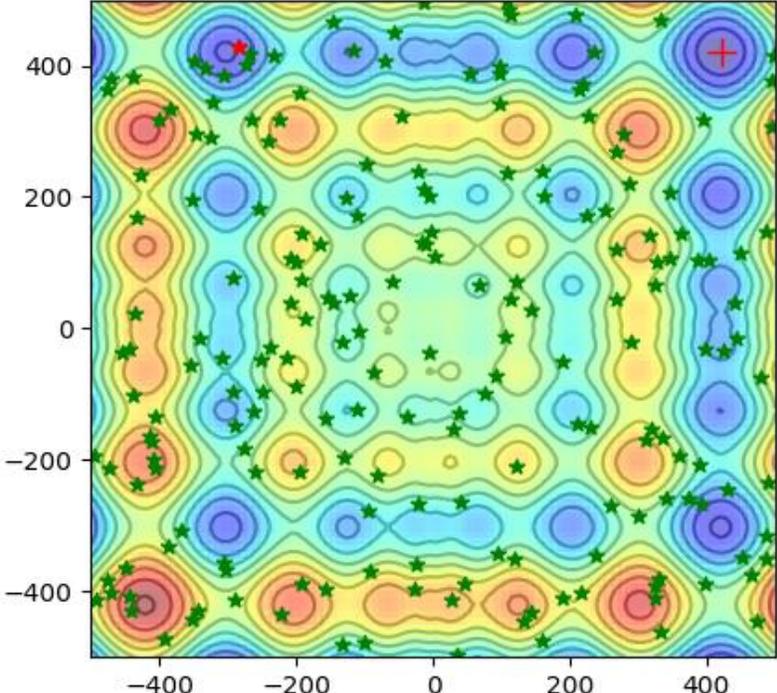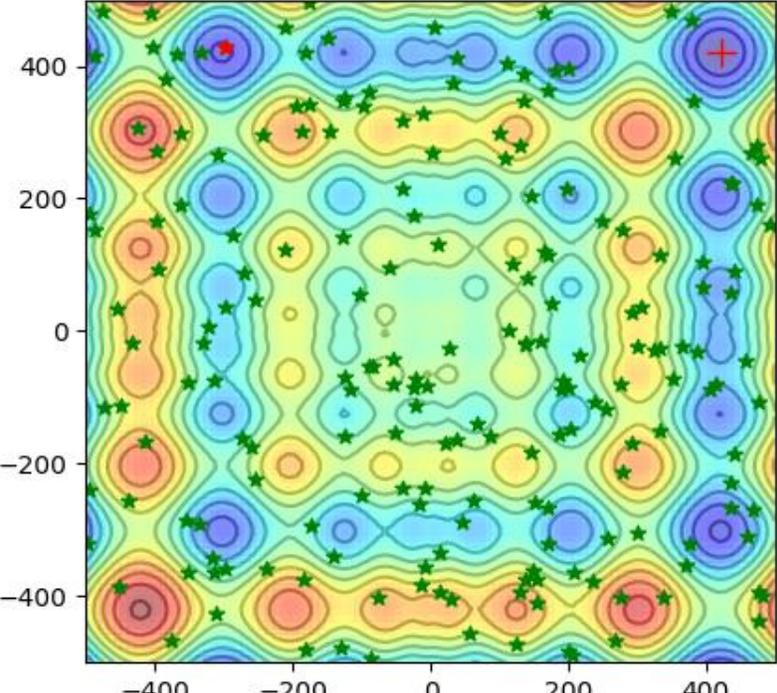
Solver	Best Objective	Iteration 3
GA	Value: 102.4799 Coordinate: (395.6167, 406.8422)	
PSO	Value: 119.4965 Coordinate: (-303.3086, 423.7552)	
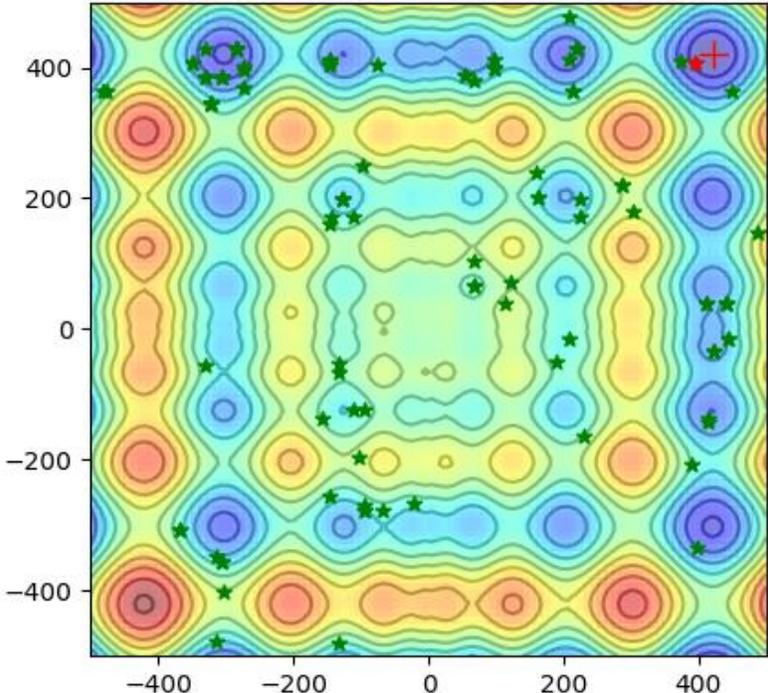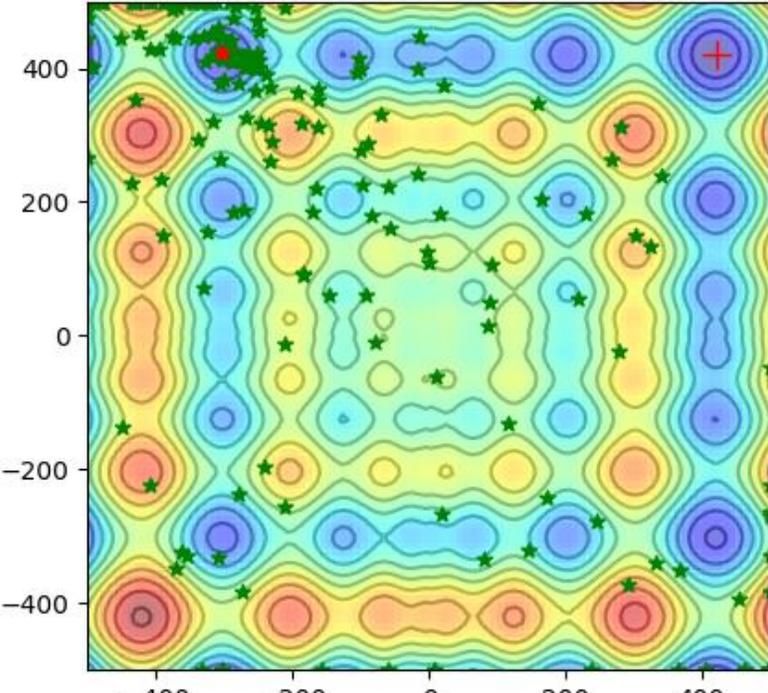
Solver	Best Objective	Iteration 5
GA	Value: 102.4799 Coordinate: (395.6167, 406.8422)	
PSO	Value: 119.1884 Coordinate: (-303.1026, 423.3363)	
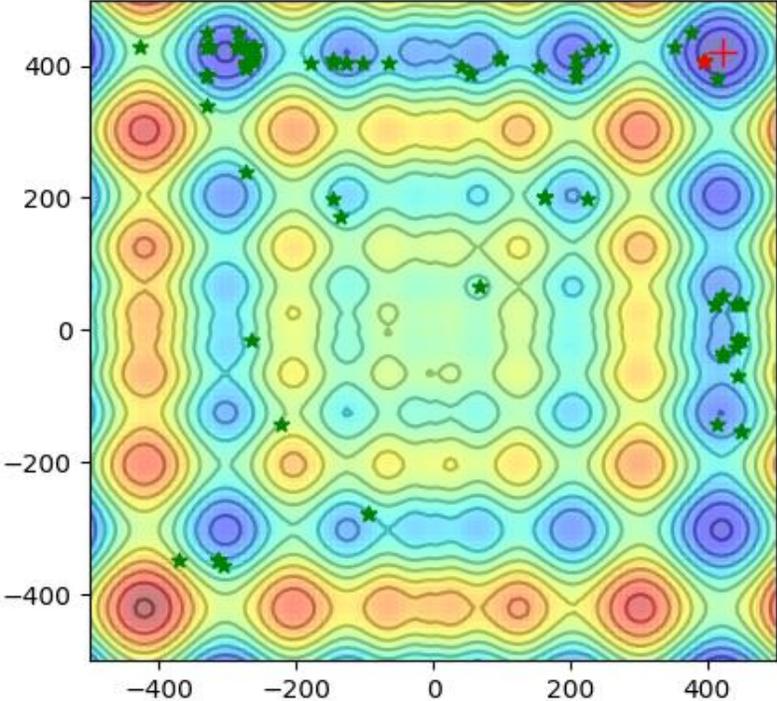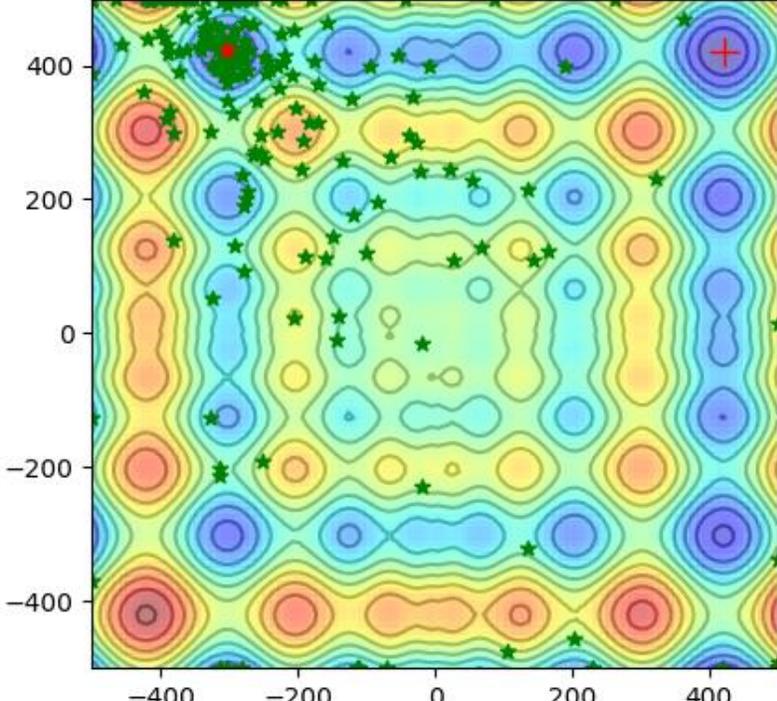
The time of termination, after 25 stall iterations were reached, for each algorithm was approximately the same due to the lack of computation intensity this optimization problem requires. However, the number of solver iterations, i.e., the number of new swarms or populations produced, were much greater in the GA, although this is not a concern. The method in which the swarms and populations are produced are time efficient and only significantly hinder computation time when the swarm or population are significant in size. When comparing the objective function executions required to converge on a solution, this is where there is a clear difference between the GA and PSO [51]. The number of function executions is more than double that of the GA, which is not intuitively a problem. The visualization of the problem is introduced in Fig. 3.5, showing the divergence of the GA and PSO solver times when the objective function execution time increases.
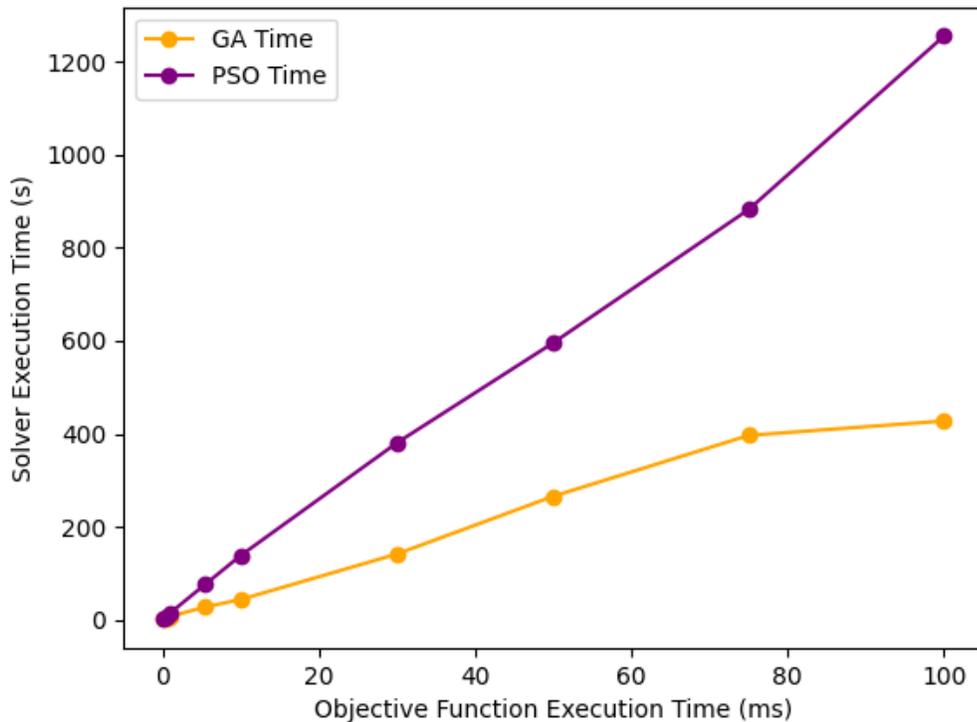
Fig. 3.5. Comparison of the average solver execution time between GA and PSO until 25 stall iterations are achieved using the Schwefel test function at different artificial objective function execution times.

The Schwefel function was artificially slowed down from the original 0.0086 to 100 ms in steps shown in the plot. Solving the Schwefel test function at each of these steps and logging the time it takes each algorithm to converge proves that the GA is much more efficient for slower objective functions. This is a very important decision variable when choosing between GA and PSO since the HAM will need to be solved multiple times per iteration of the motor optimization problem. If the data in the plot were extrapolated to seconds or even minutes in duration, then the difference in solver execution time between GA and PSO would be much more apparent. In summary, the GA is chosen as the optimal optimization algorithm for the Schwefel test function, which will act as the foundation for the solver in the motor optimization problem.

3.4. NSGAII Configuration

Without modification, the GA in Fig. 3.2 cannot optimize multi-objective problems and requires a modified implementation that produces non-dominated solutions. The non-dominated sorting genetic algorithm II (NSGAII) is a modified implementation of the GA, which will be implemented for the motor optimization problem. There are many core functionalities that are required for the NSGAII to successfully navigate a problem's constrained space and optimize towards a solution. This is no simple task, and a misconfiguration of just one core function can result in an unstable solver. The classification of NSGAII's core functionality can be segregated into the selection of dominant parents and variation for searching the domain in a robust manner. These functionalities will be discussed in detail within the following sub-sections.

3.4.1. Solver Selection

Selection is a core solver function that identifies the strongest parents among the population through comparison of fitness function results. There are many robust selection algorithms that will find the highest performing parents, such as Roulette Wheel and Rank selection, although in this thesis, the focus will be on Tournament selection. The likelihood of a parent being selected is dependent on the selection pressure, which is a probabilistic measure of a candidate's likelihood of participation in a tournament. This parameter is an indicator of a solver's ability to converge since higher selection pressure relates to a higher convergence rate. The structure of Tournament selection is highlighted in Fig. 3.6 for an arbitrary minimization problem.
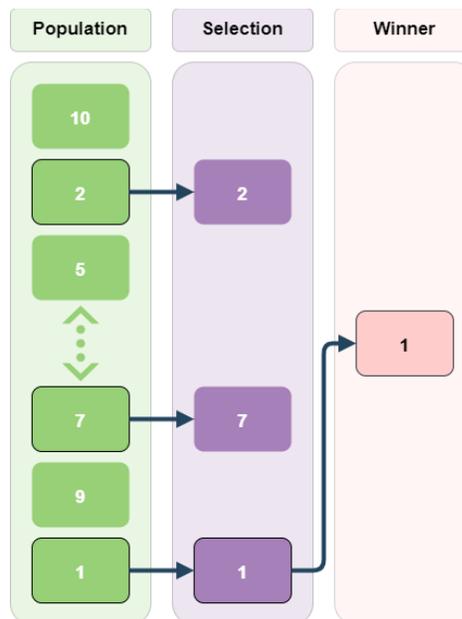
Fig. 3.6. Layout of a Tournament selection algorithm using arbitrary objective values to highlight the winning decision based on a minimization problem.

During the case study, optimal parameter ranges for Tournament selection were determined and will be used in the HAM optimization problem. Since tuning the selection

configuration is specific to the optimization problem, these parameters will have to be slightly modified, serving as a benchmark for optimization effectiveness.

3.4.2. Solver Variation

The NSGAII crossover core function allows parents to exchange their qualities and produce children while the remaining qualities are subject to some form of randomized initialization. The number of variables that are subject to be overwritten is defined by a crossover point. The crossover point highlighted in Fig. 3.7 determines the percentage of variables shared among parents. It is important not to choose too small or large of a ratio due to solver robustness. If a small percentage of variables from the parents were crossed over, then the solver may become stuck in local minima or maxima rather than the desired global alternative.
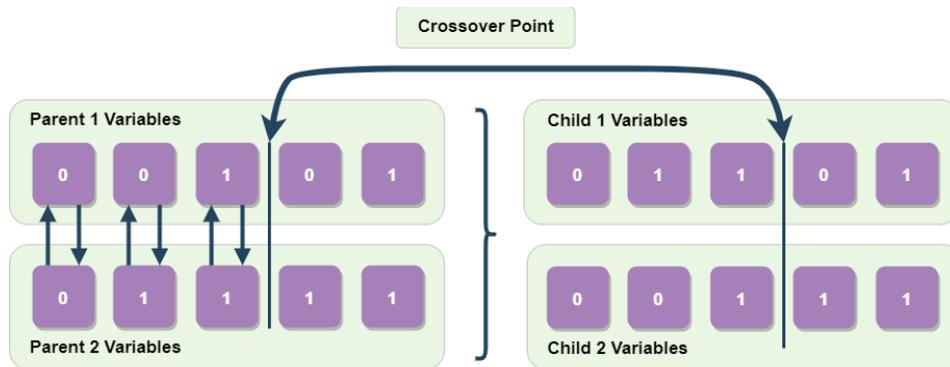
Fig. 3.7. Visualization of crossover between two parent variables to produce two child variables governed by the crossover point.

Alternatively, a large percentage of variables crossed over between parents will have large variations in the solution and can cause instability in the solver. The frequency that the crossover is applied is also an important configuration consideration. This is defined as the probability that crossover will occur between parents and is integral in the solver's robustness. Like the crossover point, if the probability of crossover is set too high, then the

parents will often share variables when producing children, which is susceptible to finding local minima or maxima rather than the desired global alternative. Contrasting this with a low probability of crossover between parents, the solver may become unstable. This is due to the children's variable initialization relying on randomized initialization, which will resist solver convergence.
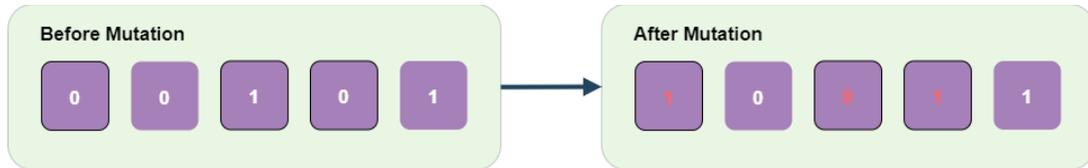
Fig. 3.8. Layout of a genetic algorithm with an arbitrary number of chromosomes and genes per population.

Mutation, highlighted in Fig. 3.8 is another important function of evolutionary algorithms which is responsible for manipulating the values of randomly selected variables within a parent. The probability for mutating a parent's variables shall remain low to maintain solver robustness rather than introducing instability.

44

CHAPTER 4   Model Optimization Integration

Due to the size and complexity required to build a HAM, it is important to simplify the model into smaller procedures. Fig. 4.1 highlights the state transitions made by the model to produce a pre-processed motor, solve the system of linear equations, and produce a processed motor model. The motor's performance parameters are then used to compute the GA objective function value and compare it to a desired solver tolerance.
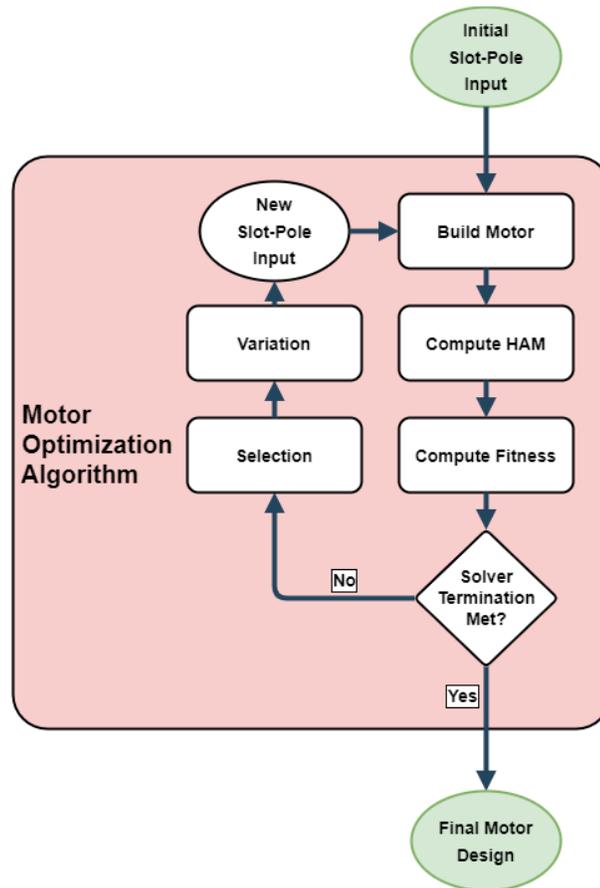
Fig. 4.1. Motor optimization algorithm state chart for hybrid analytical modelling.

The Build Motor and Compute HAM states were defined in Chapter 2. The Variation, Selection, Solver Termination, and Compute Fitness state structures were discussed in Chapter 3 using the Schwefel test function for visualization. In this chapter, the integration of these states with HAM will be discussed in more detail.

45

4.1. Optimization Constants

The baseline motor parameters were detailed in Chapter 2. Tables 4.1-4.3 in this section detail the model and motor parameters which were kept constant during the optimization process. The LIM secondary was kept the same as the baseline since the focus of the optimization was on the primary motor, and the parameters are mutually exclusive.

TABLE 4.1

MODEL PARAMETERS

Description	Variable	Value
Number of space harmonics	N	100
Number of HM regions in the model	β	5
Number of MEC regions in the model	γ	1

TABLE 4.2

MODEL MATERIAL PROPERTIES

Cast aluminum conductivity (S/m)	σ_{Al}	$17x10^6$
Iron conductivity (S/m)	σ_{Al}	$4.5x10^6$
Relative permeability of iron	μ_r	1000

TABLE 4.3

CONSTANT MOTOR PARAMETERS

Description	Variable	Value
Primary slot height (mm)	h_s	20
Primary yoke height (mm)	h_y	6.5
Airgap (mm)	h_g	2.7
Aluminum thickness (mm)	h_{Al}	2
Back iron thickness (mm)	h_{bi}	8
Primary length (mm)	L_p	270
Primary height (mm)	H_p	26.5
Primary depth (mm)	D_p	50
Periodical length of model (mm)	τ_{per}	525
Number of phases	m	3
Synchronous velocity (m/s)	v	0
Electrical frequency (Hz)	f	100
Peak current (A)	I_p	10

4.2. Motor Feasibility

To ensure that all motors produced by the optimization algorithm are feasible, the model abides by the rules in Table 4.4. If every solver iteration produces a feasible design, then there is no wasted computation.

TABLE 4.4

MOTOR FEASIBILITY CONSTRAINTS

Rule	Explanation
$N_p \% 2 = 0$	Monopoles cannot exist
$N_p > 4$	Rotating/moving fields require minimum 2 pole pairs.
$N_s \% m = 0$ and $q \% 1 = 0$	All motor slots must be filled with a coil and only integral slot windings are considered.
$B < 1.7$ T	The primary core material saturates at 1.7 T, which must be avoided by limiting the current density, J of a coil terminal.
$J <= 6.0x10^6 \frac{A}{m^2}$	The limited current density avoids saturation.
$\frac{L_p}{w_t} <= 150$	Motor core teeth produced past this threshold are considered mechanically fragile.
$\delta < d$	The required frequency at high-speed operation cannot produce a skin depth, deeper than the thickness of the aluminum.

Alternatively, even if a small percentage of the produced motors are infeasible, the resulting computation intensity can be costly in a complex optimization problem such as the one in this thesis. It is important to consider the skin effect [52], [53] if the motor application demands high speeds since the mechanical speed is directly proportional to the primary electrical frequency f from (4.1). The skin depth is calculated assuming low frequencies [54] as:

$$\delta = \sqrt{\frac{2\rho}{2\pi f \mu_r \mu_o}} \tag{4.1}$$

where ρ and μ are the resistivity and permeability of the aluminum plate, respectively. When plotting the skin depth across frequencies, the feasible frequency range is realized in Fig. 4.2.
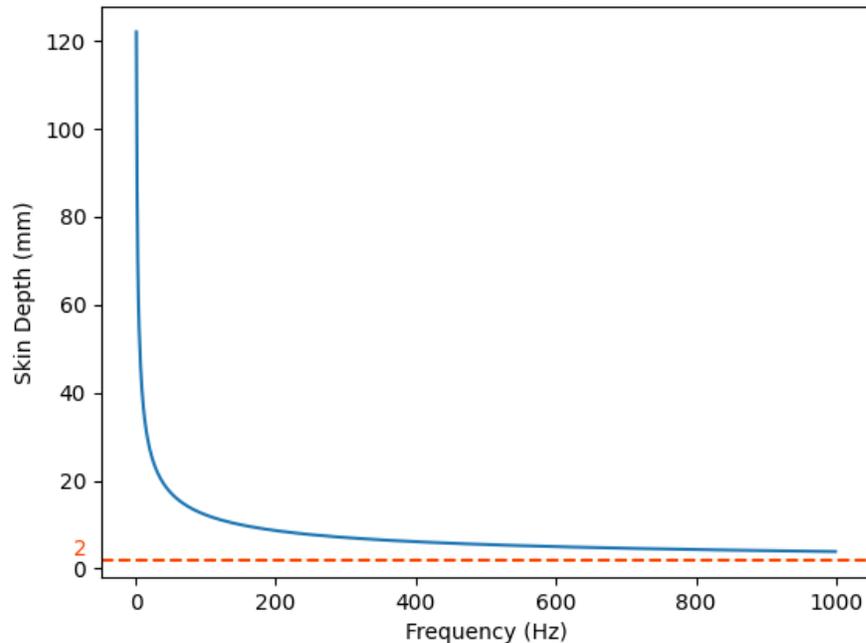

Fig. 4.2. Skin depth (blue) in the secondary aluminum plate, with increased resistivity to account for the transverse end-effects, including the plate thickness (orange).

The plate thickness and the skin depth curve do not intersect within the 1 kHz electrical frequency range, meaning that the secondary back iron is always coupled [55] with the

primary. This is an important check for motors used by the optimization algorithm otherwise, their operating frequency may be constrained by the secondary design. If the dimensions or material properties of the aluminum sheet [56] were different from the values considered in this thesis, this constraint may be applicable.

From the maximum frequency of operation in Fig. 4.2, the theoretical top speed of the motor can be solved using (2.7). This is a theoretical top speed due to the required input voltage to overcome the equivalent impedance and resistance of the motor to continue driving enough current to produce sufficient force to overcome mechanical losses to accelerate towards this speed. The frequency of operation and the slip are contributing factors to the equivalent impedance and resistance, which can help predict the required voltage at rated operating conditions. Once the supply voltage to produce this top speed is deemed feasible, the true top speed of the motor is determined.

4.3. Objective Function

After solving the Compute HAM state in Fig. 4.1, the performance parameters can be gathered and then maximized, minimized, or trended towards a bias. The performance parameters chosen for this optimization problem are defined in Fig. 4.3, which are the outputs of each motor produced by the HAM. By optimizing for thrust and mass with equal weight, the optimal motors will have a larger thrust-weight ratio. Theoretically, more performance parameters can be added to the multi objective optimization, although adding more objectives results in less non-dominated solutions being produced. This results in more HAM executions without producing an improved motor which is undesirable, especially when computation considerations are a key focus of this optimization problem.
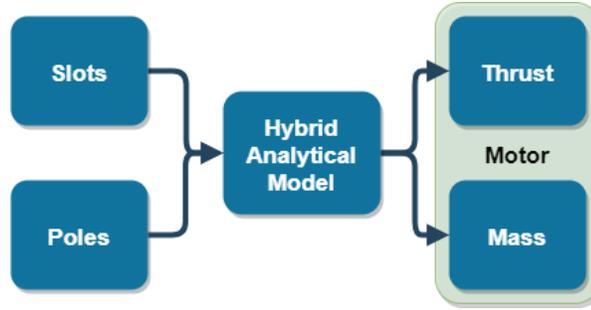
Fig. 4.3. Layout of the motor optimization algorithm inputs and the resultant multi-objectives.

Since all objectives are relatively important performance parameters, it is important that the solver produces pareto-optimal solutions, meaning the solution equally satisfies the fitness function criteria. A solution that is not pareto-optimal will still optimize every multi-objective variable but with an unequal emphasis.

The thrust calculation of the motor is described in (2.15) and is a product of the HAM equations. To calculate the mass of the motor, only the primary is considered since the secondary is a fixed design constraint. The primary mass is a summation of the core, winding, and insulation masses defined by their respective volumes times material density which are found in Table 4.5 for each material.

TABLE 4.5

MOTOR MASS EQUATIONS

Region	Material	Volume Equation	Density $(\frac{g}{cm^3})$
Core	Iron	$DLH - V_{slots}$	7.8
Winding	Copper	$2(y_p + D + w_s)Cu_{diam}N_tN_s$	8.96
Insulation	Plastic	$(1 - fillfactor)w_sh_sDN_s$	1.4

Section 4.2 discusses the criteria for feasible motor designs, which must be reflected in the fitness function. If a motor is considered not feasible, then the solver discards this iteration to save computation effort and intrinsically avoid future motor combinations like it. This is done by setting a constraint condition within the NSGAII problem definition to only allow solutions returning "True" to the feasibility in question.

4.4. Baseline Validation

4.4.1. Air Gap Magnetic Flux Density

Prior to executing an optimization workflow, the HAM model proposed in this thesis must be validated so that future resultant motors can be trusted throughout the optimization loop. The definition of the baseline motor was discussed in Section 2.1, which will serve as ground truth data since it includes FEM, experimental, and HAM data validation. The HAM model produced in this thesis is a reproduction of the one within the reference paper, so it is natural to use its data for validation. The validation was conducted for B_x in Fig. 4.4 and Fig. 4.5, and for B_y Fig. 4.6 and Fig. 4.7 allowing for the discussion of model accuracy and efficiency in this section. Within the upper-left text box in Fig. 4.8 and 4.9, the mean absolute error between curves is highlighted and is measured in tesla. The mean absolute error between two curves is calculated in (4.2).

$$\overline{error} = \frac{\sum |y_2(x) - y_1(x)|}{length(y)} \tag{4.2}$$

where y_1 and y_2 are the two curves which are dependent on the independent coordinate x. The mean absolute error for both the normal and tangential magnetic flux density plots of the air gap is low which supports the validity of the HAM produced in this thesis.

Fig. 4.4. Tangential magnetic flux density in the center of the air gap comparison between Ansys Electronics FEA (green) and the HAM produced in this thesis (blue).

Fig. 4.5. Tangential magnetic flux density in the center of the air gap comparison between the reference paper results (green) and the HAM produced in this thesis (blue).
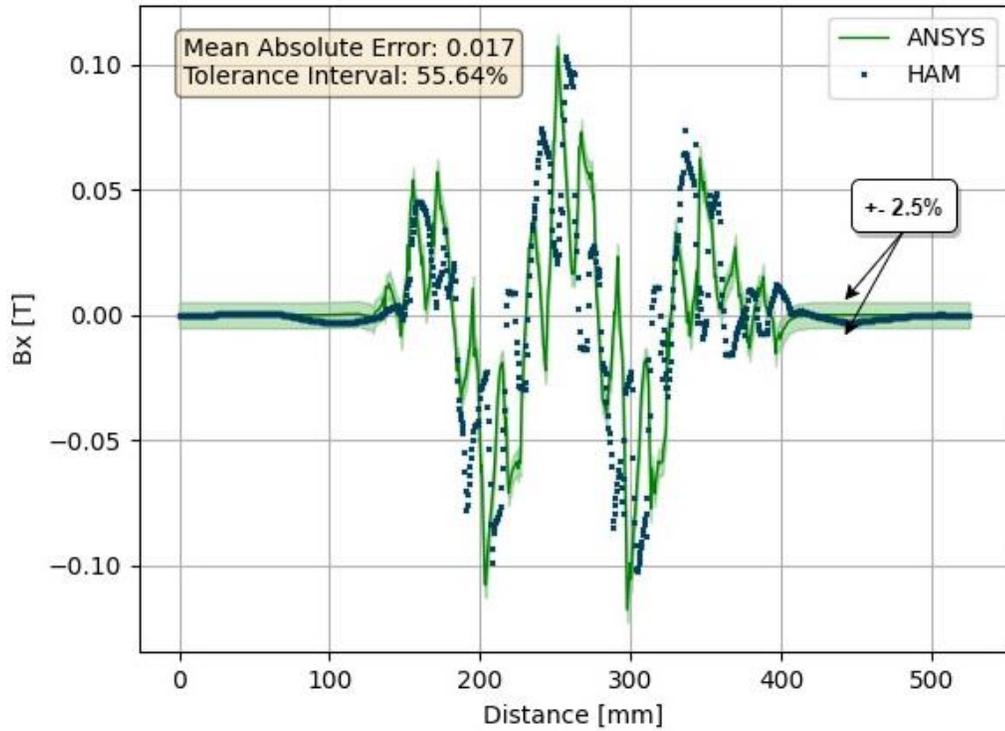
Fig. 4.6. Normal magnetic flux density in the center of the air gap comparison between Ansys Electronics FEA (blue) and the HAM produced in this thesis (blue).
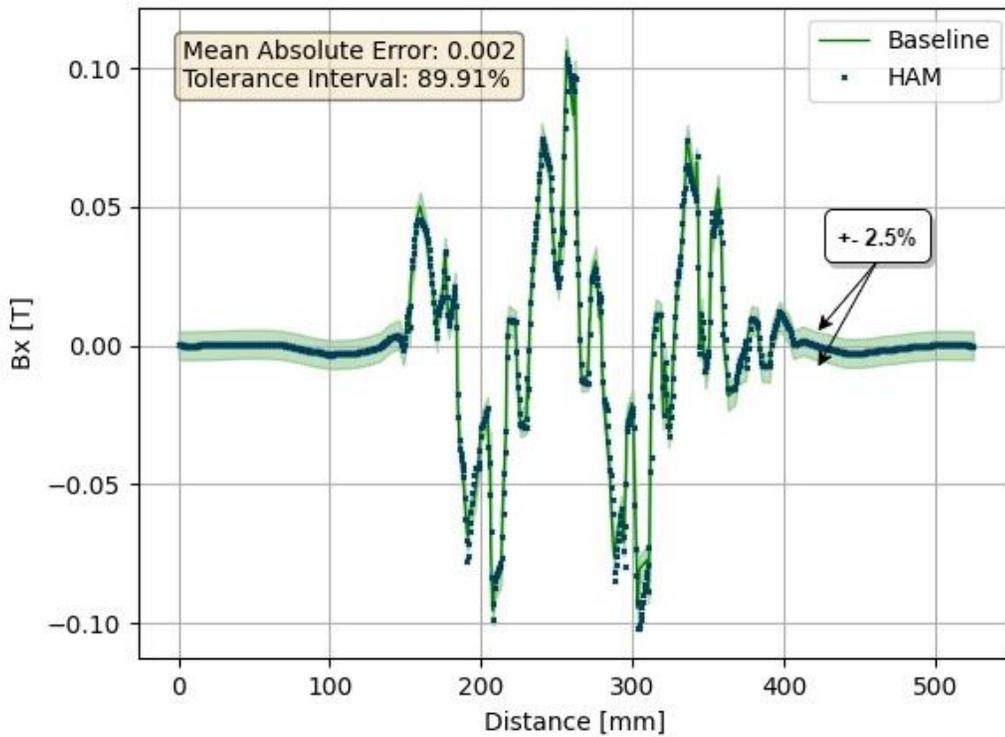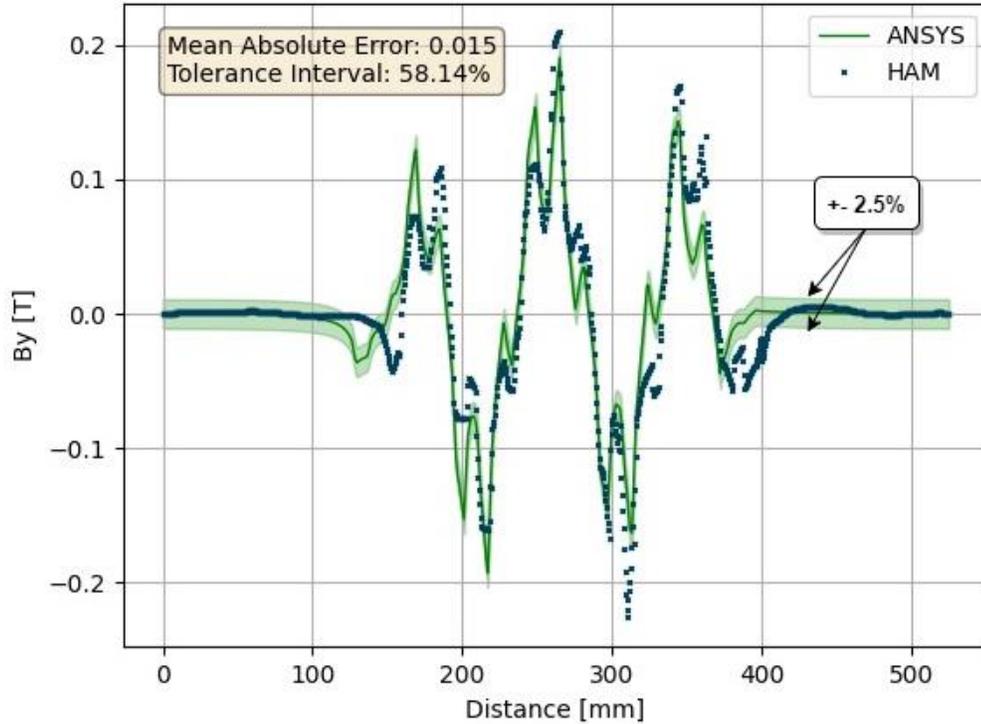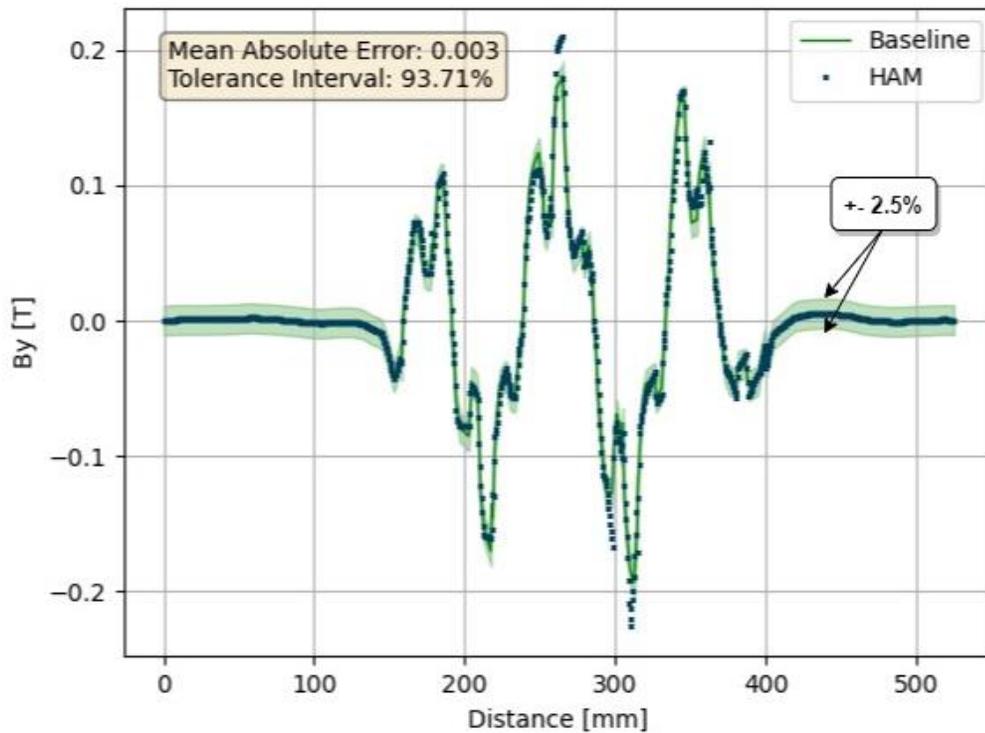

Fig. 4.7. Normal magnetic flux density in the center of the air gap comparison between the reference paper results (green) and the HAM produced in this thesis (blue).

To conduct further error analysis, a tolerance interval was provided. This is accomplished by offsetting the ANSYS or baseline plots by $\pm2.5\%$ of the peak-to-peak magnitude. Then the percentage of HAM data points that fall within this range, shaded in green in Fig. 4.4-4.7, is determined as the tolerance interval. The range of $\pm2.5\%$ was chosen due to the 5% modelling accuracy objective defined in Section 1.7. The tolerance intervals are high for the baseline-HAM comparison, whereas the ANSYS-HAM has more inaccuracies. The small misalignment and error between the Ansys Electronics results comes from the difficulty in choosing the correct steady state time step to match the reference paper since this was not provided. The air gap plots were simulated at 60 ms in Ansys Electronics which was enough time for the simulation to reach steady state and had the best correlation to the baseline results. This time step could be slightly modified to align with the baseline results better but the mean absolute error between the Ansys Electronics, baseline, and HAM curves is low and can be considered accurate. The same can be determined for the satisfactory tolerance intervals.

The histograms provided in Fig. 4.8 and Fig. 4.9 show the distribution of the difference (error) in the result. Red, dotted lines show the tolerance interval range and are provided as a reference. The distributions are approximately centered around a magnetic flux density of 0 T highlighting that the result is not biased when compared to the reference. A relatively large peak in the center of both histograms solidifies the tolerance interval and highlights the extent of the error provided by outliers. Sources of error for ANSYS results include:

- The meshing density of HAM is less than that of ANSYS in this example (polygons vs rectangle mesh).
- The time slice chosen for the simulation in Fig. 4.4-4.7 is not the same.
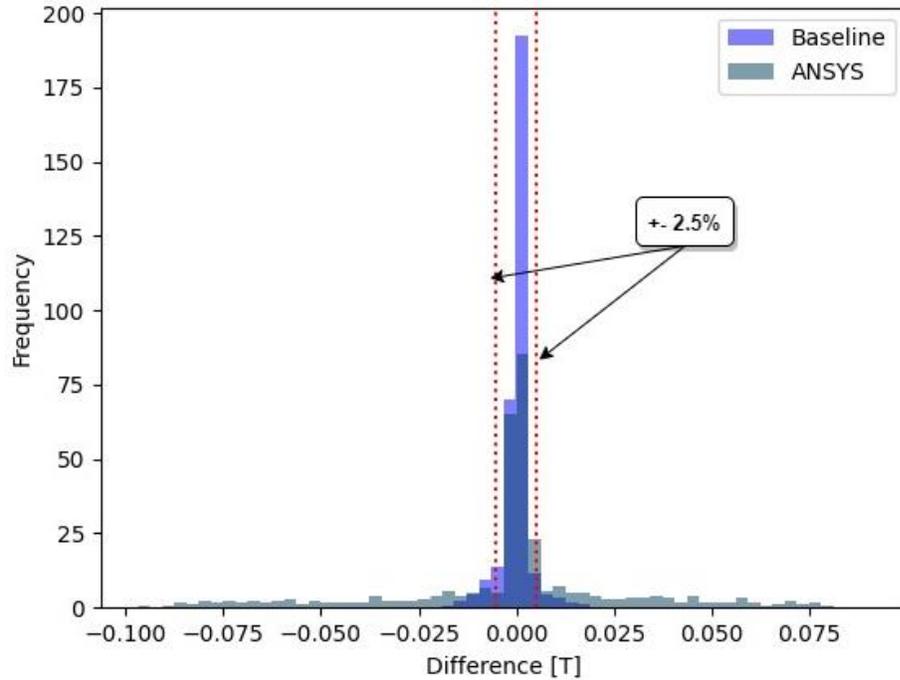
Fig. 4.8. Tangential magnetic flux density in the center of the air gap error distribution comparison of Ansys Electronics FEA (grey) and the reference paper results (purple) against the HAM produced in this thesis.
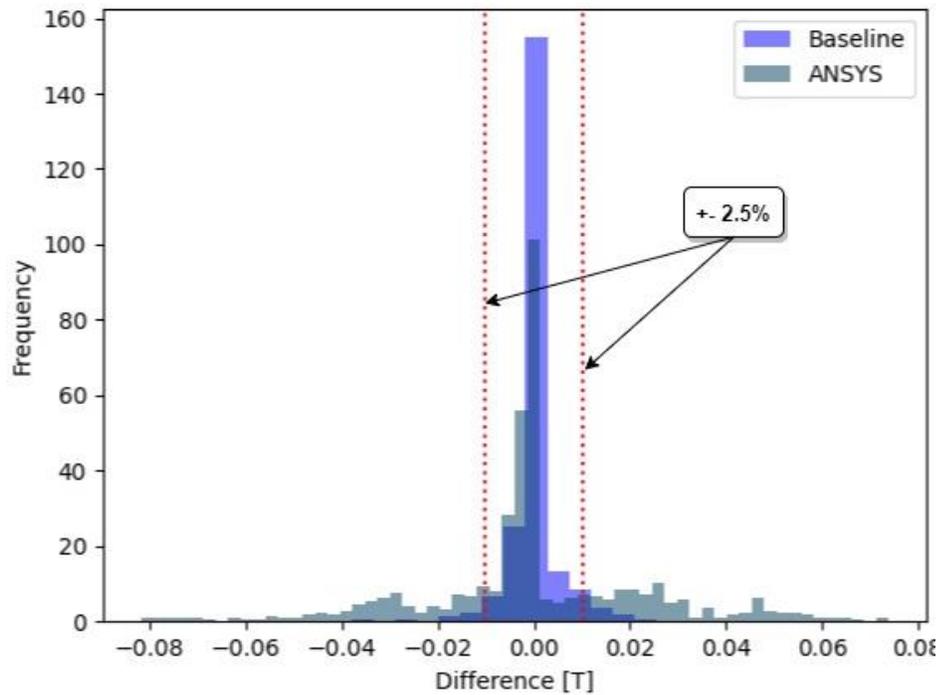
Fig. 4.9. Normal magnetic flux density in the center of the air gap error distribution comparison of Ansys Electronics FEA (grey) and the reference paper results (purple) against the HAM produced in this thesis.

56

4.4.2. Motor Core Magnetic Flux Density Plot Validation

In Section 4.2, Table 4.4 addressed the saturation constraint which must exist for all motor solutions to avoid degrading the motor core material and performance. The HAM is powerful enough to produce field plots for any post-processing model parameters such as magnetic flux density. The field plot needs to be filtered to only contain magnetic flux density data within the geometry of the motor core. A comparison between the field plotting results of the HAM proposed in this thesis and Ansys Electronics motor simulation software are visualized in Fig. 4.10 and Fig. 4.11, respectively. The filtering functionality in the HAM model is in effect on the right side of the image, which has the magnitude of the magnetic flux density per node mapped to the color bar on the left. The transparent, light blue overlay is helpful in visualizing the region where the filter is applied. Alternatively, on the left side of the figure, the original model is visible without the filter applied to highlight the flexibility of the filtering implementation.
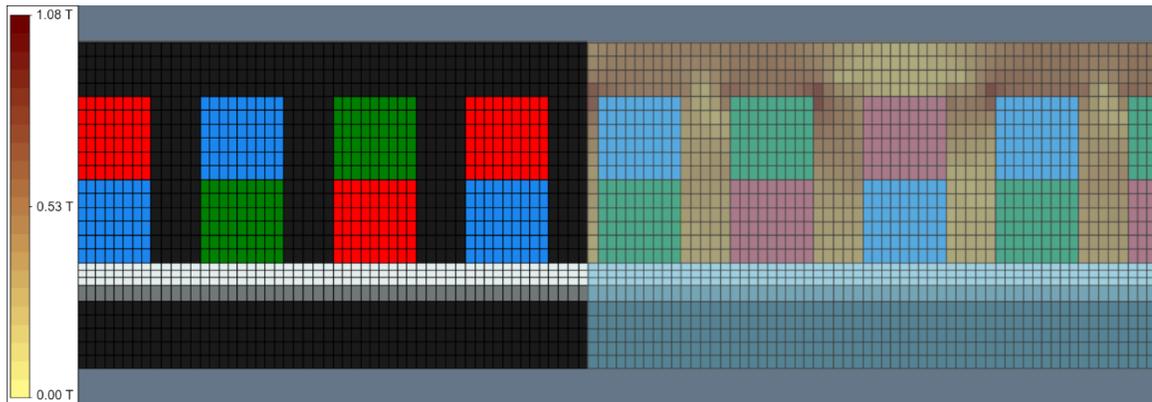
Fig. 4.10. Magnetic flux density (B) field plot in the motor core of the HAM simulation for the baseline motor to validate accuracy.
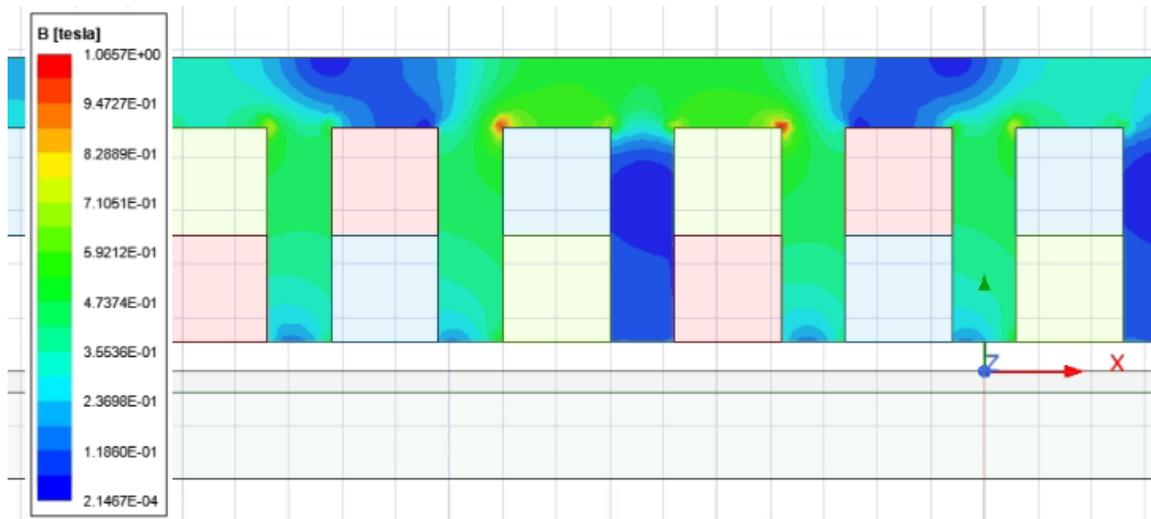
Fig. 4.11. Magnetic flux density (*B*) field plot in the motor core of the Ansys Electronics simulation for the baseline motor to validate accuracy.

Visually the results between Ansys and HAM are quite similar but vary slightly due to the difficulty in choosing the perfect time slice for the plotting to match without error. More importantly, the bounds of the color bar for magnetic flux density within the core of the motor for both plots have a percentage error of 1.3%, which is due to these factors:

- The meshing density of HAM is less than that of ANSYS in this example (polygons vs rectangle mesh).

- The time slice chosen for the simulation between Fig. 4.10 and Fig. 4.11 is not the same.

Given that the error is relatively low, this proves the accuracy of the HAM modelling algorithm and provides a robust method for creating feasible motors that do not saturate [57].

4.5. Solver Configuration

After converging on a final motor solution via the proposed optimization discussed in this thesis, the solver optimization configuration was tabulated in Table 4.6. Within the legend

58

of Fig. 4.12, there are 9 motors, each with a series for normal and tangential output force. The upper cluster of series are the tangential (thrust) forces, and the lower cluster of series are the normal forces.

TABLE 4.6

OPTIMIZATION CONFIGURATION

Configuration	Value
Iterations	27
Optimization time (s)	52.17
Motors modelled	9
Ns (min, max)	(12, 54)
Np (min, max)	(4, 12)

A trend can also be visualized in Fig. 4.14 which highlights that the data for each colored q group has a layered effect starting from the top left (orange) to the bottom right (blue). This data suggests that the higher q value motors tend to outperform the motors with lower q values.
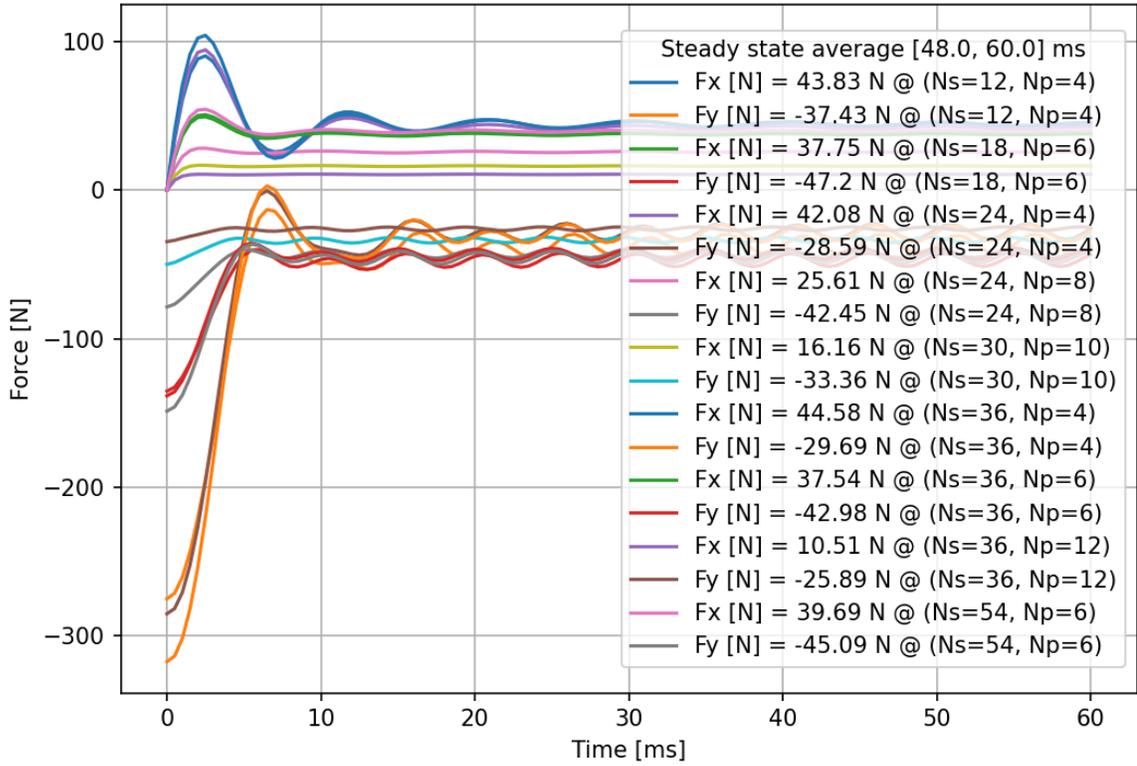
Fig. 4.12. Transient tangential (F_x) and normal (F_y) force plot of all motors evaluated during the optimization process calculated as a steady state average between the time interval of 48 to 60 ms.

60
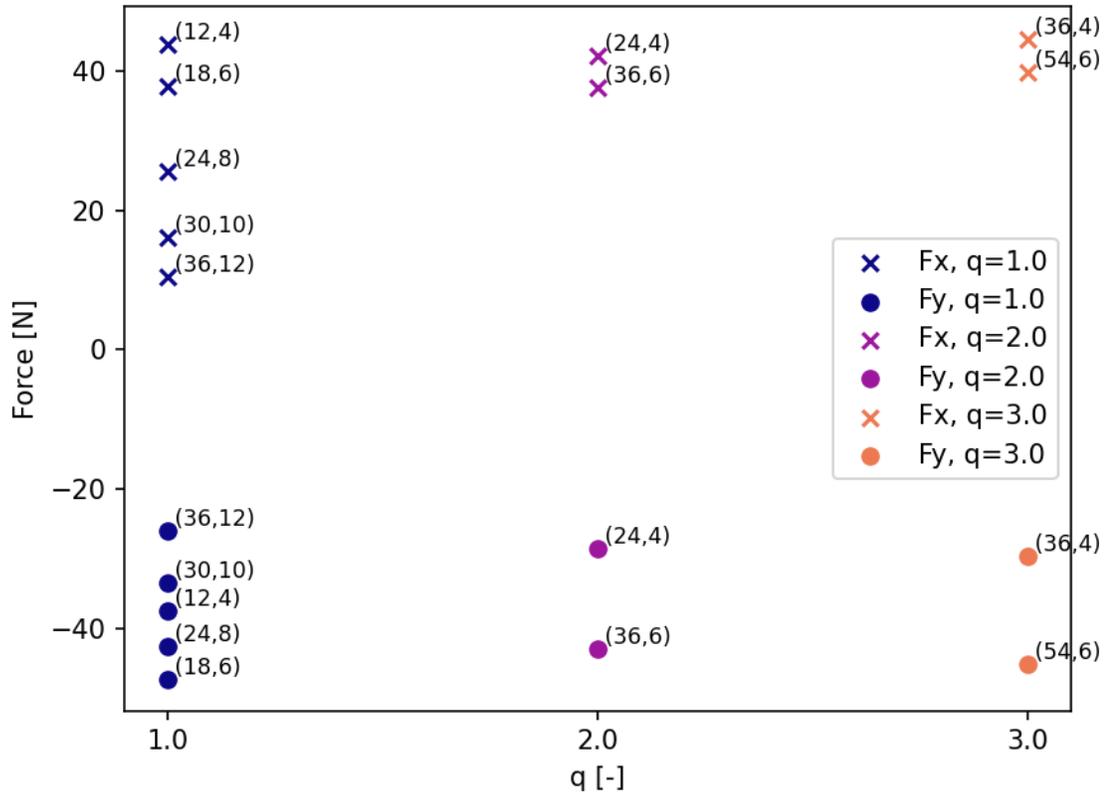
Fig. 4.14. Pareto plot of the objectives: motor mass (*x*-axis) and average steady state thrust (*y*-axis), with data point annotations in the format: (N_s, N_p).

Pareto plots, like the one in Fig. 4.14, are ideal for analyzing a multi-objective optimization algorithm's resulting solution. In most cases, the algorithm will choose the optimal solution for the application unless the application does not equally consider the objectives. For example, the two best-performing solutions presented by the algorithm are (N_s=36, N_p=4) and (N_s=54, N_p=6), which are compared in Table 4.7.

TABLE 4.7

PARETO FRONT ANALYSIS

Motor (N_s, N_p)	Mass [Kg], F_x [N]	Δ Objectives Motor 1 to 2
1: (36, 4)	1.80, 44.58	Mass: (+8%)
2: (54, 6)	1.66, 39.69	F_x: (+12%)

When the objectives are considered without bias, the best motor is $(N_s=36, N_p=4)$. However, if the application held a bias towards lower mass, then the objectives between both motors are close enough to argue that $(N_s=54, N_p=6)$ is the optimal motor.

4.6. Optimal Motor

Like the tables in Chapter 2, Tables 4.8 and 4.9 highlight the final values determined for the parameters that were variable throughout the optimization process. Since the mesh density of the MEC region heavily determines the model's accuracy when modelling the motor, the resultant values were provided in Table 4.8.

TABLE 4.8

OPTIMAL MODEL PARAMETERS

Number of rows in a magnetic equivalent circuit region	L	15
Number of columns in a magnetic equivalent circuit region	K	458

The variables contained within Table 4.9 are provided to compare against the tabulated values of the baseline motor in Section 2.1 which are not constant. These values are determined to be optimal within the constrained domain provided to the algorithm's solver.

TABLE 4.9

OPTIMAL MOTOR PARAMETERS

Pole pitch (mm)	y_p	67.5
Tooth width (mm)	w_t	2.7
Slot width (mm)	w_s	4.6
Number of slots	N_s	36
Number of poles	N_p	4
Number of turns per coil	N_t	14
Coil pitch	y_c	8

CHAPTER 5   Research Summary, Conclusion, and Future Research

5.1. Research Summary

The implementation of the procedures in previous chapters creates the foundation for producing a novel optimization algorithm within the subclass of double-layer, single-sided, 3-phase, integral slot winding, linear induction motors. This is a very constrained optimization space required to address the general optimization problem of the slot-pole combination trends in motor performance parameters. A key theme throughout this thesis is flexibility which is evident within each chapter. The WDT provides flexibility for extremely complex winding patterns. The HAM allows for complex motor models to be meshed and processed using MEC and HM modelling techniques to maintain computation efficiency. The NSGAII multi-objective optimization algorithm allows for custom objective functions that can be implemented to optimize virtually any aspect of the HAM. This is valuable considering that modern motor modelling software lacks the capability to perform general optimizations such as slot-pole combinations without some serious back-end coding implementation. To support such claims, it is important that the results produced within this thesis have fundamental value and be thoroughly validated.

5.2. Summary and Conclusion

A tolerance interval with a range of $\pm 2.5\%$ was analyzed in Section 4.4. The tolerance intervals for Baseline-HAM had an average of 56.89% between B_x and B_y plots, compared to 91.81% for ANSYS-HAM. Section 4.4 also provided histograms to show the distribution of the error in the B_x and B_y plots compared to ground truth data. The distributions proved minimal bias in the error and provided further insight into the accuracy of the HAM. The

64

observed percentage error of the magnetic flux density in the motor core between ANSYS and HAM was 1.3%. The source for all these errors were discussed in Section 4.4, and the observed errors are within the acceptable tolerance, satisfying the modelling accuracy objective defined in Section 1.7.

Modern optimization applications provide convenience via effective solutions while reducing the time complexity for the end user. Ironically, this is not the case when training and validating such an application. The optimization concluded in Section 4.6 that the optimal motor had 36 slots and 4 poles within the domain of 12-54 slots and 4-12 poles, where 9 feasible motors were objectively compared. The optimization is completed in 5 minutes, whereas the average time for configuring and simulating a motor in ANSYS takes 30 minutes. This is a conservative estimation since a template ANSYS model reduced the configuration time significantly.

The true potential of the HAM optimization application is realized once onboarding an optimization task to a new lab member or employee who might lack the knowledge of optimizing within a target motor domain. This would prompt them to begin building their own template from scratch using their motor modelling software of choice. Evidently, a task such as this is extremely time intensive and is exaggerated when multiple people are onboarded with this task. By producing a configurable motor optimization tool, the end user only needs to be onboarded on the configuration of the tool rather than all the other aspects of the motor optimization process.

5.3. Future Research

All software projects originate from an idea to solve a problem, regardless of their scale, and will endure many revisions prior to completion. Each revision serves to build on the

65

previous layer until a final product is produced. It is often the case that the original vision was too general to consider the intricacies of the final product, introducing inefficiencies in the project structure due to the time sensitivity of the target deliverables. This sub-section will investigate these considerations and highlight the potential for future revisions in their implementations.

5.3.1. Extending Motor Specifications

Determining all operating conditions and specifications was out of scope for this thesis and requires additional constraints to constrain the newly expanded scope. To maintain focus on robust optimization of the primary slot and pole meta-parameters, specifications such as efficiency, rated conditions, drive cycle performance, and other detailed motor specifications were omitted. This is a possible feature to add in future work and would further simplify the design process at the cost of complexity and invested effort.

5.3.2. Constrained Optimization Domain

This thesis discussed many processes to ensure the production of feasible motors. However, there are many more processes that can be implemented. For example, the skin depth calculation could be extended to the primary motor's winding wire in the case of high frequencies of operation. There are many necessary constraints implemented which reduce the complexity and narrow the focus of motor optimization within its domain. The constants are highlighted in Section 4.1 and partially limit the ability to generally search for an optimal motor within a relatively unconstrained domain. To truly find the optimal motor in all aspects of the application, many layers of complex optimization are required. A wide range of operating conditions need to be considered. For example, a motor may

have an operational frequency range from 50 Hz to 500 Hz, which requires simulations at steps within this range. This thesis optimizes at a constant frequency and the optimal motor may differ at different operational frequencies. AI or a layered approach to the optimization problem, keeping the top percentile of motors from each optimization layer until one remains, would solve this problem while drastically increasing the complexity.

5.3.3. Winding Feasibility Using WDT

The feasibility of linear motor winding patterns was discussed in Section 2.3, which highlights the limitation of producing balanced windings. Due to the time and effort, it would take to implement this feature. It was omitted to reduce the complexity required to produce winding patterns for an entire domain of motors. Future work should improve upon this downfall and utilize the capability of the WDT to include empty slots, improving the effectiveness of the optimization. The current WDT implementation still produces accurate trends in motor performance, and the final motor design would require a slight adjustment in the number of empty or partially empty slots to achieve a feasible motor.

5.3.4. ANSYS Implementations

Using a custom modelling algorithm added necessary flexibility to the optimization within this thesis. However, using a custom model also introduces an error that would be minimized if the optimization was coupled directly with ANSYS motor model results. This is theoretically possible using PyAEDT to interact with the AEDT API and pipe those results into a custom optimization algorithm all within a single python application. Further investigation on the feasibility and effort to implement this solution would be required but serves as a viable alternative to the implementation conducted in this thesis.

5.3.1. Rotary Induction Motor Adaptation

This thesis focused on the domain of linear induction motors although the HAM could be modified to solve for rotary induction motors if the coordinate system were switched to cylindrical or polar. The optimization workflow would require very few, if any, modifications since it is generic to the type of motor. The feasibility checks and the motor relationships would require some rework as well as any LIM specific configurations to switch to the rotary induction motor domain. Although it would be a lot of work, it would greatly improve the usefulness for more motor designers.

REFERENCES

[1]   "Total greenhouse gas emissions," Our World in Data. [Online]. Available: https://ourworldindata.org/grapher/total-ghg-emissions. [Accessed: 20-Jan-2023].

[2]   "Science for Environment Policy," Environment. [Online]. Available: https://environment.ec.europa.eu/research-and-innovation/science-environment-policy_en. [Accessed: 20-Jan-2023].

[3]   Iea, "Transport – topics," IEA. [Online]. Available: https://www.iea.org/topics/transport. [Accessed: 20-Jan-2023].

[4]   N. R. Canada, "Government of Canada," Natural Resources Canada, 20-Oct-2022. [Online]. Available: https://www.nrcan.gc.ca/energy-efficiency/transportation-alternative-fuels/personal-vehicles/choosing-right-vehicle/buying-electric-vehicle/21034. [Accessed: 20-Jan-2023].

[5]   E. D. Santos, J. Camacho, A. Paula and C. H. Salerno "The Linear Induction Motor (LIM) Power Factor, Efficiency and Finite Element Considerations," 2002.

[6]   J. H. Xue and F. P. Banko, "Pioneering the Application of High Speed Rail Express Trainsets in the United States," Parsons Brinckerhoff, 2012.

[7]   D. Hu, W. Xu, and R. Qu, "Electromagnetic design optimization of single-sided linear induction motor for improved drive performance based on linear metro application," in Proc. Australas. Univ. Power Eng. Conf. (AUPEC), pp. 1–6, 2014.

[8]   "Ice 3," Wikipedia, 14-Jan-2023. [Online]. Available: https://en.wikipedia.org/wiki/ICE_3. [Accessed: 20-Jan-2023].

[9]   Yuichiro Nozaki, Terufumi Yamaguchi and Takafumi Koseki "Equivalent Circuit Model of Linear Induction Motor with Parameters Depending on Secondary Speed for Urban Transportation System," 2015

[10]   "Claude Nicollier," Swissloop. [Online]. Available: https://swissloop.ch/prototypes/claude-nicollier/. [Accessed: 20-Jan-2023].

[11]   J. Gieras, Linear Induction Drives (Monographs in Electrical and Electronic Engineering). Oxford, U.K.: Clarendon, 1994.

[12]   S. Nonaka and T. Higuchi, "Design of single-sided linear induction motors for urban transit," IEEE Transactions on Vehicle Technology, vol. TVT-37, no. 3, pp. 167–173, Aug. 1988.

[13]   A. Fatima, F. Khan, M. A. Khan, and L. U. Rahman, "Performance comparison of partitioned primary hybrid excited linear flux switching machine," Mechanics Based Design of Structures and Machines, vol. 51, no. 1, pp. 359–380, 2020.

[14]   S. Wang, D. M. Miao and J. X. Shen, "Optimal design of a linear induction motor for woodworking machine application," 2014 17th International Conference on Electrical Machines and Systems (ICEMS), Hangzhou, China, 2014, pp. 3633-3637, doi: 10.1109/ICEMS.2014.7014121.

[15]   B. Hague, "The principles of electromagnetism, applied to electrical machines," in Electromagnetic Problems in Electrical Engineering, 1st ed. New York, NY, USA: Dover, 1962.

[16]   J. Pyrhönen, T. Jokinen and V. Hrabovcová, "Design of Rotating Electrical Machines," 2008.

[17]   S. Yamamura, Theory of Linear Induction Motors, 1st ed., Wiley, 1972.

[18]   S. Wang, Z. Zhu, A. Pride, J. Shi, R. Deodhar, and C. Umemura, "Comparison of different winding configurations for dual three-phase interior PM machines in electric vehicles," World Electric Vehicle Journal, vol. 13, no. 3, p. 51, 2022.

[19]   M. Caruso, A. Di Tommaso, F. Marignetti, R. Miceli, and G. Ricco Galluzzo, "A general mathematical formulation for winding layout arrangement of Electrical Machines," Energies, vol. 11, no. 2, p. 446, 2018.

[20]   M. Amrhein and P. T. Krein, "3-D magnetic equivalent circuit framework for modeling electromechanical devices," IEEE Transactions on Energy Conversion, vol. 24, no. 2, pp. 397–405, Jun. 2009.

[21]   C. H. H. M. Custers, T. T. Overboom, J.W. Jansen, and E. A. Lomonova, "2-D semianalytical modeling of eddy currents in segmented structures," IEEE Transactions on Magnetics, vol. 51, no. 11, pp. 1–4, Nov. 2015.

[22]   B. L. Gysen, K. J. Meessen, J. J. Paulides, and E. A. Lomonova, "General formulation of the electromagnetic field distribution in machines and devices using Fourier analysis," IEEE Transactions on Magnetics, vol. 46, no. 1, pp. 39–52, 2010.

[23]   K. Boughrara, F. Dubas, and R. Ibtiouen, "2-D analytical prediction of eddy currents, circuit model parameters, and steady-state performances in solid rotor induction motors," IEEE Transactions on Magnetics, vol. 50, no. 12, pp. 1–14, 2014.

[24]   S.-B. Yoon, J. Hur, and D.-S. Hyun, "A method of optimal design of single-sided linear induction motor for transit," IEEE Transactions on Magnetics., vol. 33, no. 5, pp. 4215–4217, 1997.

[25]   T. T. Overboom, J. Smeets, J. W. Jansen and E. A. Lomonova, "Semi-analytical modeling of a linear induction motor including primary slotting". In Proceedings of the 15th International Symposium on Electromagnetic Fields in Mechatronics (ISEF), Funchal, Portugal, 1–3 September 2011; pp. 1–8.

[26]   S. Ouagued, A. Aden Diriye, Y. Amara, and G. Barakat, "A General Framework Based on a Hybrid Analytical Model for the Analysis and Design of Permanent Magnet Machines," IEEE Transactions on Magnetics, vol. 51, no. 11, pp. 1–4, Nov. 2015, doi: 10.1109/tmag.2015.2442214.

[27]   D. C. Horvath, S. D. Pekarek, and S. D. Sudhoff, "A scaled mesh/nodal formulation of magnetic equivalent circuits with motion," IEEE Transactions on Energy Conversion, vol. 34, no. 1, pp. 58–69, 2019.

[28] B. L. Gysen, E. Ilhan, K. J. Meessen, J. J. Paulides, and E. A. Lomonova, "Modeling of flux switching permanent magnet machines with Fourier analysis," IEEE Transactions on Magnetics, vol. 46, no. 6, pp. 1499–1502, 2010.

[29] T. M. Masuku, R. -J. Wang, M. C. Botha and S. Gerber, "Design Strategy of Traction Induction Motors," 2019 Southern African Universities Power Engineering Conference/Robotics and Mechatronics/Pattern Recognition Association of South Africa (SAUPEC/RobMech/PRASA), 2019, pp. 316-321, doi: 10.1109/RoboMech.2019.8704761.

[30] "PYAEDT documentation 0.6.43#," PyAEDT documentation 0.6.43 - PyAEDT. [Online]. Available: https://aedt.docs.pyansys.com/release/0.6/. [Accessed: 20-Jan-2023].

[31] Verez, Guillaume. (2022). Finite Elements coding with python: Electromagnetics. 10.13140/RG.2.2.15505.51040.

[32] Bhamidi, Sarveswra Prasad. "Design of a single sided linear induction motor (SLIM) using a user interactive computer program," 2005.

[33] S. Osawa, M. Wada, M. Karita, D. Ebihara, and T. Yokoi, "Light-weight type linear induction motor(lim) and its characteristics," 1992. Digests of Intermag. International Magnetics Conference, 1992.

[34] M. Kitamura, N. Hino, H. Nihei, and M. Ito, "A direct search shape optimization based on complex expressions of 2-dimensional magnetic fields and forces," IEEE Transactions on Magnetics, vol. 34, no. 5, pp. 2845–2848, 1998, doi: 10.1109/20.717662.

[35] B. Laporte, N. Takorabet, and G. Vinsard, "An approach to optimize winding design in linear induction motors," IEEE Transactions on Magnetics, vol. 33, no. 2, pp. 1844–1847, Mar. 1997, doi: 10.1109/20.582640.

[36] T. Mishima, M. Hiraoka and T. Nomura, "A study of the optimum stator winding arrangement of LIM in maglev systems," IEEE International Conference on Electric Machines and Drives, 2005., San Antonio, TX, USA, 2005, pp. 1238-1242, doi: 10.1109/IEMDC.2005.195880.

[37] A. H. Isfahani, H. Lesani and B. M. Ebrahimi, "Design Optimization of Linear Induction Motor for Improved Efficiency and Power Factor," 2007 IEEE International Electric Machines & Drives Conference, Antalya, Turkey, 2007, pp. 988-991, doi: 10.1109/IEMDC.2007.382810.

[38] A. H. Isfahani, B. M. Ebrahimi, and H. Lesani, "Design Optimization of a Low-Speed Single-Sided Linear Induction Motor for Improved Efficiency and Power Factor," IEEE Transactions on Magnetics, vol. 44, no. 2, pp. 266–272, Feb. 2008, doi: 10.1109/tmag.2007.912646.

[39] C. Lucas, Z. Nasiri-Gheidari, and F. Tootoonchian, "Application of an imperialist competitive algorithm to the design of a linear induction motor," Energy Conversion and Management, vol. 51, no. 7, pp. 1407–1411, Jul. 2010, doi: 10.1016/j.enconman.2010.01.014.

71

[40] A. Shiri and A. Shoulaie, "Design optimization and analysis of singlesided linear induction motor, considering all phenomena," IEEE Transaction on Energy Conversion, vol. 27, no. 2, pp. 516–525, Jun. 2012.

[41] A. Shiri and A. Shoulaie, "Multi-objective optimal design of low-speed linear induction motor using genetic algorithm," 2012.

[42] A. Nekoubin "Using Finite Element Method for Determination of Poles Number in Optimal Design of Linear Motor," 2011.

[43] S. R. Aleksandrov, T. T. Overboom, and E. A. Lomonova, "2D hybrid steady-state magnetic field model for linear induction motors," Mathematical and Computational Applications, vol. 24, no. 3, p. 74, 2019.

[44] V. Ostoviˊc, Dynamics of Saturated Electric Machines. New York, NY, USA: Springer-Verlag, 1989.

[45] K. J. Pluk, J. W. Jansen, and E. A. Lomonova, "Hybrid analytical modeling: Fourier modeling combined with mesh-based magnetic equivalent circuits," IEEE Transactions on Magnetics, vol. 51, no. 8, pp. 1–12, 2015.

[46] K. Woronowicz and A. Safaee, "A novel linear induction motor equivalent-circuit with optimized end-effect model including partially-filled end slots," 2014 IEEE Transportation Electrification Conference and Expo (ITEC), 2014.

[47] J. Lu and W. Ma, "Research on end effect of linear induction machine for high-speed industrial transportation," IEEE Transactions on Plasma Science, vol. 39, no. 1, pp. 116–120, 2011.

[48] M. Amrhein and P. T. Krein, "Force calculation in 3-D magnetic equivalent circuit networks with a Maxwell stress tensor," IEEE Transactions on Energy Conversion, vol. 24, no. 3, pp. 587–593, 2009.

[49] J. F. Gieras, A. R. Eastham, and G. E. Dawson, "Performance calculation for single-sided linear induction motors with a solid steel reaction plate under constant current excitation," IEE Proceedings B Electric Power Applications, vol. 132, no. 4, p. 185, 1985.

[50] J. W. Gibbs, "Fourier series," Nature, vol. 59, p. 200, 1898.

[51] F. G. Lobo, D. E. Goldberg, and M. Pelikan, "Time complexity of genetic algorithms on exponentially scaled problems". In Darrell Whitley et al., editors, Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000), pages 151-158, Morgan Kaufmann, 2000.

[52] C. Timperio "Linear Induction Motor (LIM) for Hyperloop Pod Prototypes," Published by ETH Zurich, Institute of Electromagnetic Fields (IEF), 2019.

[53] I. Boldea and M. Babescu, "Multilayer approach to the analysis of single-sided linear induction motors," Proceedings of the Institution of Electrical Engineers, vol. 125, no. 4, p. 283, 1978, doi: 10.1049/piee.1978.0072.

[54] W. Hayt and J. Buck, Engineering Electromagnetics, New York: McGraw-Hill, 2012.

[55] T. T. Overboom, J. P. C. Smeets, J. W. Jansen, and E. Lomonova, "Decoupled control of thrust and normal force in a double-layer single-sided linear induction motor," Mechatronics, vol. 23, no. 2, pp. 213–221, 2013.

[56] H. Bolton, "Transverse edge effect in sheet-rotor induction motors," Proceedings of the Institution of Electrical Engineers, vol. 116, no. 5, p. 725, 1969.

[57] J. Bao, B. L. Gysen, and E. A. Lomonova, "Hybrid analytical modeling of saturated linear and rotary electrical machines: Integration of Fourier modeling and magnetic equivalent circuits," IEEE Transactions on Magnetics, vol. 54, no. 11, pp. 1–5, 2018.

VITA AUCTORIS

NAME:	**Michael Thamm**
PLACE OF BIRTH:	Dresden, GER
YEAR OF BIRTH:	1997
EDUCATION:	St.Anne High School, Belle River, ON, 2015
	University of Windsor, B.A.Sc., Windsor, ON, 2019