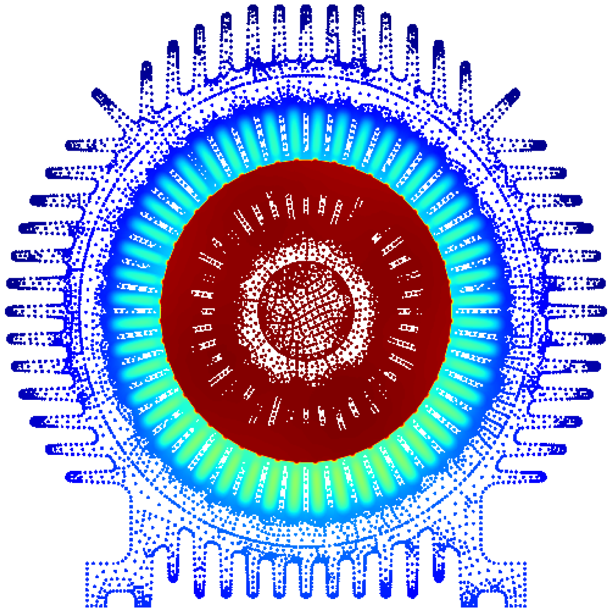

Neural Operator–Based Thermal Modeling for Electric Motor and Power Electronics Systems

A Data-Driven Surrogate for Rapid Electric-Machine Thermal Analysis

Master's thesis in Systems, Control and Mechatronics

JINGHAO LI

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026
www.chalmers.se

Master's thesis 2026

Neural Operator–Based Thermal Modeling for Electric Motor and Power Electronics Systems

A Data-Driven Surrogate for Rapid Electric-Machine Thermal Analysis

Jinghao Li

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
Gothenburg, Sweden 2026

Neural Operator–Based Thermal Modeling for Electric Motor and Power Electronics Systems:
A Data-Driven Surrogate for Rapid Electric-Machine Thermal Analysis

Jinghao Li

Supervisor: Panagiotis Kakosimos, ABB; Nikolce Murgovski, Chalmers University of Technology
Examiner: Nikolce Murgovski, Chalmers University of Technology

Master's Thesis 2026
Department of Electrical Engineering
Division of Systems and Control

Chalmers University of Technology
SE-412 81 Gothenburg
Telephone: +46 31 772 1000

Cover: FEM-reference temperature field of the 15 kW squirrel-cage induction motor at the final simulated time point under the 100% constant-load condition.

Typeset in LaTeX
Printed by Chalmers Digital Printing
Gothenburg, Sweden 2026

Abstract

Transient finite-element method (FEM) simulations provide detailed temperature fields for electric machines but can be computationally expensive when repeated evaluations are required. This thesis develops a load-memory-enhanced Fourier Neural Operator (FNO) as a computationally efficient surrogate for predicting two-dimensional transient temperature fields in a 15 kW squirrel-cage induction motor.

The available FEM data are converted from an irregular mesh to a common 512×512 Cartesian grid. The proposed direct model predicts the temperature rise relative to the initial condition using normalized elapsed time, spatial coordinates, a fixed base heat-source field, the instantaneous load, and three exponentially weighted moving-average load-memory features. These memory features represent short-, medium-, and long-term load exposure without requiring an autoregressive temperature rollout.

The model is evaluated through leave-one-condition-out cross-validation on four constant-load conditions and on an independent 100%–50%–75% varying-load trajectory. Across the cross-validation folds, the mean field root-mean-square error is 4.82 ± 2.22 °C, with better accuracy for intermediate-load interpolation than for boundary extrapolation. For the varying-load case, the root-mean-square errors of the maximum and spatial mean temperatures are 4.61 °C and 3.02 °C, respectively. Complete-trajectory inference requires approximately 2.19 seconds, corresponding to an approximately 74-fold measured wall-clock speedup over the reference FEM execution. The results demonstrate the potential of load-memory-enhanced neural operators for rapid transient thermal-field prediction.

Keywords: electric machines, transient thermal analysis, Fourier Neural Operator, neural operators, surrogate modelling, finite-element method, load memory, exponentially weighted moving average .

Acknowledgements

I would like to express my sincere gratitude to my supervisors, Panagiotis Kakosimos at ABB SECRC and Nikolce Murgovski at Chalmers University of Technology, for their guidance, constructive feedback, and valuable technical discussions throughout the thesis. Their support was particularly valuable during the initial exploration of different modeling approaches and the subsequent development and evaluation of the Fourier Neural Operator method.

I would also like to thank my colleagues and fellow thesis students at ABB for the welcoming and collaborative working environment. I am especially grateful for the cooperation concerning the ANSYS simulation data and for the discussions of alternative temperature-field prediction approaches. These exchanges helped clarify the data requirements, compare modeling choices, and improve the overall quality of the work.

Finally, I would like to thank my family and friends for their patience, encouragement, and support throughout my studies and thesis project.

Jinghao Li, Gothenburg, August 2026

Declaration of Generative AI Use

Generative AI tools, including OpenAI Codex and Microsoft 365 Copilot provided through ABB, were used during the preparation of this thesis to support technical discussions, code-related work, document organization, and the drafting and revision of selected text. The ABB-provided environment was used for company-related work in accordance with the applicable organizational data-handling requirements. All generated suggestions were critically reviewed, verified, and edited by the author. The author assumes full responsibility for the methodology, implementation, experiments, results, interpretations, and final content of the thesis.

Table of contents

List of figures

List of tables

1. Introduction

1.1 Background

Electric machines are central to modern electrification, with applications ranging from electric vehicles and renewable-energy systems to industrial automation. Increasing demands for power density, torque density, and compact machine design make thermal behavior an important design and operating constraint. Excessive winding and core temperatures can increase electrical losses, reduce efficiency, accelerate insulation ageing, and damage temperature-sensitive components [1]. Accurate prediction of the internal temperature distribution is therefore important for cooling design, hotspot identification, reliability assessment, and safe machine operation [2].

Finite-element methods (FEM) are widely used for thermal analysis because they can represent complex geometries, material regions, heat sources, and boundary conditions [3]. In transient analysis, the temperature field is evaluated repeatedly over time, allowing the heating and cooling behavior of different machine regions to be studied. The resulting spatial information is valuable when local temperatures and hotspots are important. However, transient FEM simulation requires repeated numerical solution over many time steps. Its computational cost becomes significant when many load conditions, parameter combinations, or operating sequences must be evaluated [4].

Several approaches have been proposed to reduce the cost of thermal prediction. Lumped-parameter thermal networks provide compact models for temperature estimation and monitoring [5]. More general model-order-reduction techniques can reduce the computational requirements of multiphysics electric-machine models [6]. Hybrid methods have also combined lumped thermal structures with machine learning to retain a compact physical representation while increasing modeling flexibility [7]. These methods are computationally efficient, but they commonly provide a limited number of temperatures rather than a complete two-dimensional field. This can make them less suitable when the spatial distribution or localized hotspots are of interest.

Machine-learning surrogates offer another way to accelerate numerical simulation. Once trained on reference data, a surrogate can approximate the relationship between operating conditions and temperature without repeating the complete FEM calculation. Physics-informed neural networks incorporate governing equations into the learning objective [8], and physics-informed neural operators extend related ideas to operator learning [9]. Data-driven surrogate models have also been investigated for parameterized thermal fields in power-electronic systems [10].

Alternative machine-learning surrogates include convolutional and graph neural networks. CNN-based encoder–decoder models can formulate temperature-field prediction as image-to-image regression on a structured grid, and physics-informed CNNs have been applied to thermal-field prediction from different heat-source layouts [11]. Graph neural networks can instead operate on mesh nodes and connections, potentially avoiding regular-grid interpolation when the original FEM connectivity is available [12]. PINNs were also considered during the exploratory stage of this project. However, they require the governing equations, boundary conditions, and material properties to be represented in the training objective. Since complete FEM temperature fields were available and the objective was fast supervised field-to-field prediction, the final study focused on the FNO. No controlled comparison between these approaches was performed.

Neural operators are designed to approximate mappings between functions, making them suitable for problems in which both inputs and outputs contain spatially distributed quantities [13]. DeepONet is one foundational neural-operator architecture [14]. The Fourier Neural Operator (FNO) instead represents global spatial interactions through learned transformations in Fourier space [15]. An FNO-based model has recently been applied to steady-state spatial temperature monitoring of power modules under varying current and cooling conditions [16]. These developments indicate that neural operators are a promising approach for rapid spatial thermal prediction.

Transient electric-machine temperatures introduce an additional difficulty: the temperature at a given time depends not only on the instantaneous load, but also on the preceding load exposure. Two instances with the same current load can have different thermal states if their load histories differ. A surrogate intended for time-varying operation must therefore receive some representation of thermal history. In this thesis, exponentially weighted moving averages (EWMAs) are used to provide compact short-, medium-, and long-term summaries of the applied load.

The specific application considered in this thesis is transient thermal-field prediction for a 15 kW squirrel-cage induction machine. The available FEM data represent a two-dimensional cross-section of this machine under several constant-load conditions and one independent varying-load sequence.

1.2 Problem Statement

The available FEM data contain complete transient temperature fields, but repeated FEM execution is comparatively expensive. A computationally efficient surrogate is therefore desirable for applications requiring multiple temperature-field evaluations. Such a surrogate should retain the spatial resolution of the reference data, predict the transient rather than only the steady-state response, and account for the effect of previous operating loads.

Three challenges must be addressed. First, the FEM results are defined on an irregular mesh, whereas the implemented FNO operates on structured spatial tensors. The raw temperature and heat-source data must therefore be mapped to a common regular grid without losing their principal spatial structure. Second, the model must predict complete two-dimensional temperature fields at different times and operating conditions. Third, the model must represent the dependence of the current thermal state on preceding load exposure when evaluated under a changing-load sequence.

The problem considered in this thesis is therefore to construct and evaluate a neural-operator surrogate that maps an initial temperature field, elapsed time, spatial information, current operating load, and compact load history to the corresponding transient temperature-rise field. The surrogate is trained using constant-load FEM trajectories and is evaluated both on held-out constant-load conditions and on an independently generated time-varying load profile.

1.3 Aim

The aim of this thesis is to develop and evaluate a computationally efficient surrogate model for predicting two-dimensional transient temperature fields in the machine from FEM data. The proposed approach combines a direct Fourier Neural Operator with exponentially weighted load-memory features. The model is intended to reproduce the principal spatial and temporal behavior of the FEM temperature fields while reducing the time required for the complete-trajectory evaluation.

To achieve this aim, the work addresses the following objectives:

- Transform irregular-mesh FEM outputs into a structured dataset suitable for Fourier neural-operator learning;
- Formulate a direct temperature-rise prediction problem using the initial field, time, spatial information, current load, and load-memory features;
- Train and evaluate the model under held-out constant-load conditions;
- Evaluate the trained surrogate on an independent varying-load trajectory;
- Compare the measured inference time with the corresponding FEM execution time.

1.4 Research Questions

The thesis is guided by the following research questions:

RQ1: How accurately can a Fourier Neural Operator predict complete transient temperature fields for constant-load conditions excluded from model training?

RQ2: Can a direct FNO conditioned on the instantaneous load and EWMA load memories reproduce the principal heating, cooling, and reheating behaviour of an independent varying-load FEM trajectory?

RQ3: What reduction in measured complete-trajectory evaluation time can the trained surrogate provide compared with the reference FEM execution?

1.5 Scope and Delimitations

The available dataset covers one machine and comprises six constant-load trajectories and one independently generated 100%–50%–75% varying-load trajectory. The individual load magnitudes in the varying-load case are represented in the constant-load data, whereas their temporal order and the resulting heating, cooling, and reheating transitions are not included in training. The evaluation therefore examines generalization across the available load conditions and one dynamic sequence, rather than arbitrary operating histories.

The work focuses on data preprocessing, surrogate-model development, and evaluation. Development and validation of the underlying FEM model are not part of the thesis. The original irregular-mesh outputs are interpolated onto a fixed 512×512 Cartesian grid, and all reported models are trained and evaluated at this resolution. Generalization across spatial resolution, machine geometries, material definitions, or cooling configurations is not investigated.

The model predicts a two-dimensional temperature field and does not explicitly represent three-dimensional thermal effects. No additional physical experiments are performed as part of this work. The reported performance is therefore specific to the available FEM dataset and should not be interpreted as validation for other machines or three-dimensional configurations.

The EWMA spans and FNO hyperparameters were selected through preliminary experiments. The reported configuration provided the best performance among the settings that were tested. However, an exhaustive hyperparameter search and a formal ablation of the individual EWMA channels were not performed. The computational comparison reports the measured wall-clock times of the available FEM and neural-network executions. Since their computational resource usage was not controlled to be identical, the resulting speedup is interpreted as a practical measured comparison rather than a hardware-normalized algorithmic speedup.

1.6 Contributions

The main contributions of the thesis are:

- An end-to-end preprocessing workflow that converts the irregular-mesh FEM temperature and heat-source data into a structured HDF5 dataset;

- A direct nine-channel FNO formulation for predicting the temperature-rise field from the initial temperature, elapsed time, spatial coordinates, a base heat-source field, the current load, and three EWMA load-memory features;

- The use of three EWMA features to provide short-, medium-, and long-term information about the previous load history;

- Leave-one-condition-out evaluation across constant-load operating conditions, including both interpolation and boundary-extrapolation cases;

- Evaluation on an independent varying-load trajectory through global temperature indicators, spatial field comparisons, and local temperature traces;

- A measured computational-cost comparison between complete-profile FNO inference and the reference FEM execution.

The varying-load model achieves root-mean-square errors of 4.61 °C for maximum temperature and 3.02 °C for spatial mean temperature. Complete-trajectory inference requires approximately 2.2 s, corresponding to an approximately 74-fold measured wall-clock speedup over the reference FEM execution. These results demonstrate the feasibility of the proposed surrogate for rapid transient thermal-field prediction while also showing reduced robustness at the boundaries of the evaluated load range.

1.7 Ethical and Sustainability Aspects

Electric machines are important in transportation, industrial automation, and other electrified systems. Reliable thermal analysis can support the identification of hotspots and the development of appropriate cooling and operating strategies. A computationally efficient temperature-field surrogate could enable more operating conditions and design alternatives to be evaluated within a limited time. In this way, the proposed approach has the potential to support the development of efficient and reliable electric-machine systems. However, this thesis does not directly optimize the motor design or quantify any resulting reduction in energy consumption.

From a sustainability perspective, the trained surrogate requires considerably less execution time than the reference FEM simulation for repeated temperature-field evaluations. This could reduce the computational resources needed for parameter studies, design iterations, or repeated operating-condition assessment. Nevertheless, the FEM data generation and neural-network training also require computational resources. Since no energy-consumption or life-cycle assessment is performed, the measured execution-time reduction should not be interpreted directly as a quantified environmental benefit.

The main ethical and responsible-use concern is the possibility of relying on surrogate predictions outside the conditions for which the model has been validated. The model is developed using data from one two-dimensional FEM representation and a limited set of operating conditions. It also inherits the assumptions and simplifications of the underlying FEM model. Consequently, it should be used as a supporting engineering tool rather than as a replacement for physical validation or safety-critical thermal analysis. Predictions for other motors or unrepresented load ranges require additional validation and engineering judgement. The study does not involve personal data; therefore privacy and demographic bias are not central ethical concerns in this application.

1.8 Thesis Structure

The remaining thesis is organized as follows. Chapter 2 describes the available FEM dataset and the preprocessing pipeline used to construct the regular-grid HDF5 dataset. Chapter 3 presents the proposed load-memory-enhanced FNO, including the model inputs, EWMA formulation, architecture, and loss function.

Chapter 4 describes the cross-validation design, independent varying-load evaluation, normalization protocol, evaluation metrics, and computational-cost measurement. Chapter 5 presents the constant-load, varying-load, and computational results. Chapter 6 discusses the results, limitations, and directions for future work. Finally, Chapter 7 summarizes the main findings and outlines directions for future work.

2. FEM Dataset and Data Preprocessing

2.1 Methodological Overview

This thesis develops a data-driven surrogate for predicting the transient temperature distribution of a 15 kW squirrel-cage induction machine under constant and time-varying loads. The reference temperature fields are obtained from finite-element method (FEM) data generated in ANSYS. Since the raw results are defined on an irregular mesh, they cannot be supplied directly to the implemented Fourier Neural Operator (FNO), whose Fourier layers operate on structured spatial tensors. A preprocessing pipeline is therefore used to transform the nodal and element-wise FEM data into regular-grid input and target fields.

The methodology consists of four main stages. First, the available FEM files are read and matched through their node identifiers and spatial coordinates. Second, the irregular-mesh temperature and heat-source data are interpolated onto a common two-dimensional Cartesian grid, and a geometry mask is created to separate the machine domain from the background. Third, supervised learning samples are constructed from the initial temperature field, normalized elapsed time, spatial coordinates, base heat-source field, instantaneous load, and load-memory features. The prediction target is the temperature rise relative to the initial condition. Finally, the samples and their metadata are stored in an HDF5 dataset used for model training and evaluation.

The FEM data and preprocessing stages are described in this chapter. The load-memory formulation and FNO architecture are introduced in Chapter 3, while the experimental protocol is presented in Chapter 4.

Figure 2.1 emphasizes the separation between the reference data and the developed surrogate workflow. The FEM results are treated as the reference solution. The methodological contribution begins with reconstructing these results on a regular grid and continues through sample construction, load-memory representation, FNO training, and evaluation.

2.2 FEM Dataset

The dataset consists of transient thermal FEM results generated in ANSYS for a two-dimensional cross-section of the machine. The data describe the evolution of its

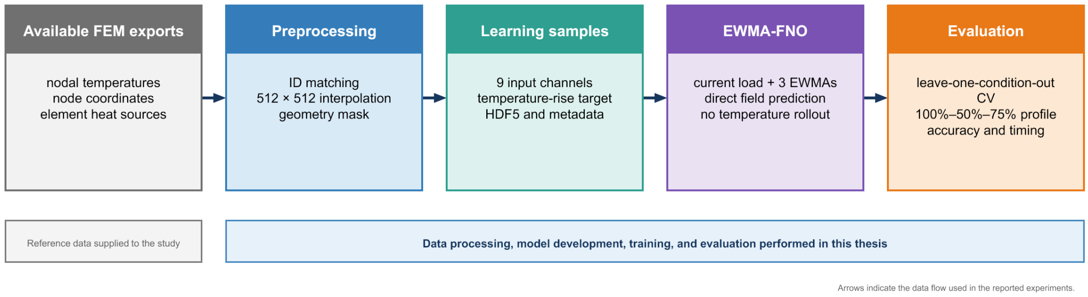

Figure 2.1: Workflow from the available FEM exports to the processed HDF5 dataset, load-memory-enhanced FNO, and model evaluation. The reference FEM data are inputs to the study; the preprocessing, surrogate-model development, training, and evaluation constitute the work reported in this thesis.

temperature distribution under several constant operating loads. For each condition, temperatures are available at multiple time instances and at the nodes of the original irregular FEM mesh.

In addition to the transient nodal temperatures, the dataset contains the spatial coordinates of the FEM nodes and element-centroid heat-source information. These quantities constitute the raw data used in this thesis. They are processed into regular-grid inputs and targets suitable for the FNO.

2.2.1 Operating Conditions

Six constant-load conditions are available:

$$p \in \{0.25, 0.50, 0.625, 0.75, 0.875, 1.00\}, \tag{2.1}$$

corresponding to 25%, 50%, 62.5%, 75%, 87.5%, and 100% load. Each condition contains 100 time samples, resulting in 600 constant-load samples in total. The nominal interval between consecutive FEM output samples is approximately 217.8 s.

The condition order in the HDF5 file is listed in Table 2.1. It is non-monotonic in load but is handled through explicit condition-index arrays rather than by assuming a particular sample order.

The independently generated 100%–50%–75% trajectory is not included in these 600 samples. It is reserved for evaluating the model under a changing load history.

Figure 2.2 visualizes the maximum and active-domain mean temperatures in all six processed constant-load trajectories. Each trajectory starts near the common

Table 2.1: Constant-load conditions in the processed dataset.

Condition	Load fraction	Samples	Dataset indices
25%	0.250	100	0–99
62.5%	0.625	100	100–199
75%	0.750	100	200–299
87.5%	0.875	100	300–399
100%	1.000	100	400–499
50%	0.500	100	500–599
Total	–	600	0–599

initial temperature and approaches a load-dependent thermal level. The increasing separation between the curves shows that the dataset contains distinct thermal responses rather than only different labels for otherwise similar fields.

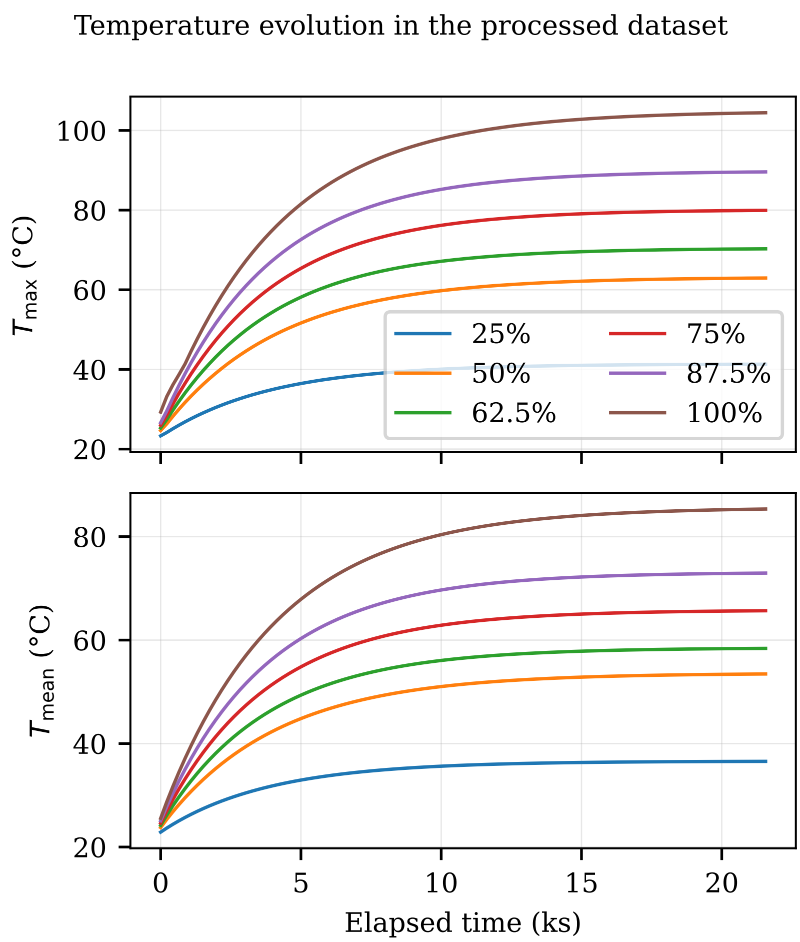

Figure 2.2: Maximum and active-domain mean temperature evolution in the six processed constant-load trajectories. Time is measured relative to the first exported snapshot of each trajectory.

2.2.2 Available Raw Data

The preprocessing pipeline uses three types of information:

- node identifiers and spatial coordinates;
- nodal temperatures at different physical time instances; and
- element-centroid heat-source values.

Table 2.2 summarizes how each exported quantity is used in the preprocessing pipeline. The node coordinates define the common spatial reference, the transient nodal temperatures provide the initial fields and targets, and the element-centroid values provide the spatial heat-source descriptor.

Table 2.2: Available FEM exports and their use in this thesis.

Export	Main contents	Use in preprocessing
Node coordinates	Node identifier and in-plane coordinates	Grid definition, node matching, interpolation, and mask construction
Transient temperatures	Physical time, node identifier, and nodal temperature	Initial field, temperature snapshots, and temperature-rise targets
Element heat sources	Element-centroid coordinates and volumetric source values	Construction of the regular-grid base heat-source field

The temperature records are matched to the coordinate data through their node identifiers. This avoids relying on the row order of the exported files. The loader verifies that the coordinate identifiers are unique, that temperature nodes can be matched to their coordinates, and that the resulting node–time temperature matrices do not contain missing entries.

2.3 Data Preprocessing

All operating conditions are processed using the same grid, geometry mask, and interpolation procedure. This produces samples with identical spatial dimensions and allows the conditions to be combined in a single dataset.

2.3.1 Regular-Grid Construction

The minimum and maximum in-plane node coordinates are first determined from the FEM mesh. A margin equal to 2% of the coordinate range is added on each side of the domain. For the horizontal coordinate, for example,

$$x \in \left[x_{\min} - 0.02(x_{\max} - x_{\min}), x_{\max} + 0.02(x_{\max} - x_{\min})\right], \tag{2.2}$$

with an equivalent definition for the vertical coordinate. Both axes are sampled uniformly using 512 points, resulting in a 512×512 Cartesian grid. The same grid is used for every temperature snapshot and heat-source field.

Two dimensionless spatial-coordinate channels are also constructed:

$$x_{\mathrm{norm},j} = \frac{j}{W-1}, \qquad y_{\mathrm{norm},k} = \frac{k}{H-1}, \tag{2.3}$$

where $H = W = 512$. These channels provide the neural operator with explicit spatial-position information.

2.3.2 Geometry-Mask Generation

The rectangular grid includes both the machine and the surrounding empty area. A binary mask is therefore constructed using the distance from each grid point to its nearest FEM node. Let $\mathbf{g}_{jk}$ denote a grid point and $\mathbf{r}_n$ an FEM-node position. The nearest-node distance is

$$d_{jk} = \min_n \left\| \mathbf{g}_{jk} - \mathbf{r}_n \right\|_2 . \tag{2.4}$$

The geometry mask is then defined as

$$M_{jk} = \begin{cases} 1, & d_{jk} < 1.5 \max(\Delta x, \Delta y), \\ 0, & \text{otherwise}, \end{cases} \tag{2.5}$$

where Δx and Δy are the regular-grid spacings. The final mask contains 113,659 active pixels, corresponding to approximately 43.36% of the complete grid. The mask is used during preprocessing, training, and evaluation so that background pixels do not affect the reported errors.

2.3.3 Temperature-Field Interpolation

At each time instance, nodal temperatures are interpolated from the irregular FEM coordinates to the regular Cartesian grid. Linear scattered-data interpolation is applied first. Grid points for which the linear interpolation is undefined are filled using nearest-neighbour interpolation. The resulting field is multiplied by the geometry mask.

This procedure is repeated for all 100 time instances of each condition. As a result, every physical temperature snapshot is represented by an array with shape 512×512.

2.3.4 Base Heat-Source Field

The available heat-source values are defined at element centroids. They are interpolated onto the temperature grid using the same combination of linear interpolation and nearest-neighbour filling. The geometry mask is then applied to obtain the base heat-source field $Q_{\text{base}}(\mathbf{x})$.

In the final learning representation, the same base heat-source field is used as a spatial input for all samples, while the instantaneous load fraction p_i is provided as a separate channel. The base field describes the spatial distribution of heat-generating regions, whereas p_i identifies the current operating condition. Before being supplied to the model, the base field is standardized using preprocessing statistics stored in the HDF5 dataset. Its physical, unnormalized form is retained for traceability and visualization.

2.3.5 Initial Field, Time, and Target Construction

For each constant-load trajectory, the first interpolated temperature field is used as the initial field:

$$u_0(\mathbf{x}) = T(\mathbf{x}, t_0). \tag{2.6}$$

The initial field is standardized using its preprocessing mean and standard deviation over active pixels. Physical time is represented as normalized elapsed time:

$$t_{\text{norm},i} = \frac{t_i - t_0}{t_{\text{end}} - t_0}. \tag{2.7}$$

The target is the temperature rise relative to the initial field:

$$u(\mathbf{x}, t_i) = T(\mathbf{x}, t_i) - u_0(\mathbf{x}). \tag{2.8}$$

The corresponding absolute prediction is later reconstructed as

$$\widehat{T}(\mathbf{x}, t_i) = u_0(\mathbf{x}) + \widehat{u}(\mathbf{x}, t_i). \tag{2.9}$$

For storage compatibility, the HDF5 targets are standardized using global dataset statistics. During model training, these targets are converted to a normalization calculated only from the selected training samples and active pixels. This training-only normalization is described in Section 4.2.1.

2.3.6 HDF5 Dataset Organization

Each model input contains nine channels:

$$\begin{aligned}\mathbf{a}(\mathbf{x}, t_i) = [&\widetilde{u}_0(\mathbf{x}), t_{\mathrm{norm},i}, x_{\mathrm{norm}}, y_{\mathrm{norm}}, \widetilde{Q}_{\mathrm{base}}(\mathbf{x}), \\ &p_i, m_i^{(3)}, m_i^{(10)}, m_i^{(30)}].\end{aligned} \tag{2.10}$$

The time, load, and EWMA values are scalars broadcast over the spatial grid. The final input and target arrays have the shapes

$$\texttt{X\_all} \in \mathbb{R}^{600\times 9\times 512\times 512}, \qquad \texttt{Y\_all} \in \mathbb{R}^{600\times 1\times 512\times 512}. \tag{2.11}$$

The HDF5 file additionally stores the geometry mask, physical coordinate vectors, physical base heat-source field, load fractions, normalized and physical elapsed times, condition indices, preprocessing statistics, and EWMA metadata. The arrays are stored using 32-bit floating-point precision and gzip compression.

2.3.7 Preprocessing Validation

The direct raw-data preprocessing pipeline was compared with the earlier two-stage workflow. Using absolute and relative tolerances of 10^{-5}, agreement was confirmed for all nine input channels, the active-domain temperature-rise target, the geometry mask, the spatial grids, the condition indices, and the EWMA features. Inactive target pixels are explicitly stored as zero and do not contribute to training or evaluation because the geometry mask is applied throughout.

Visual checks were also performed for the initial-temperature field, base heat-source field, coordinate channels, mask, representative temperature snapshots, and condition-level temperature curves.

Figure 2.3 shows representative fields from the 100% trajectory. The first row contains absolute temperature, the second row contains the corresponding temperature-rise target, and the third row shows the spatial heat-source input. The target is zero at the initial snapshot by construction and develops the same principal spatial pattern as the absolute temperature as heating progresses. The heat-source descriptor remains fixed in time, while the elapsed-time and load-history channels provide the changing temporal information.

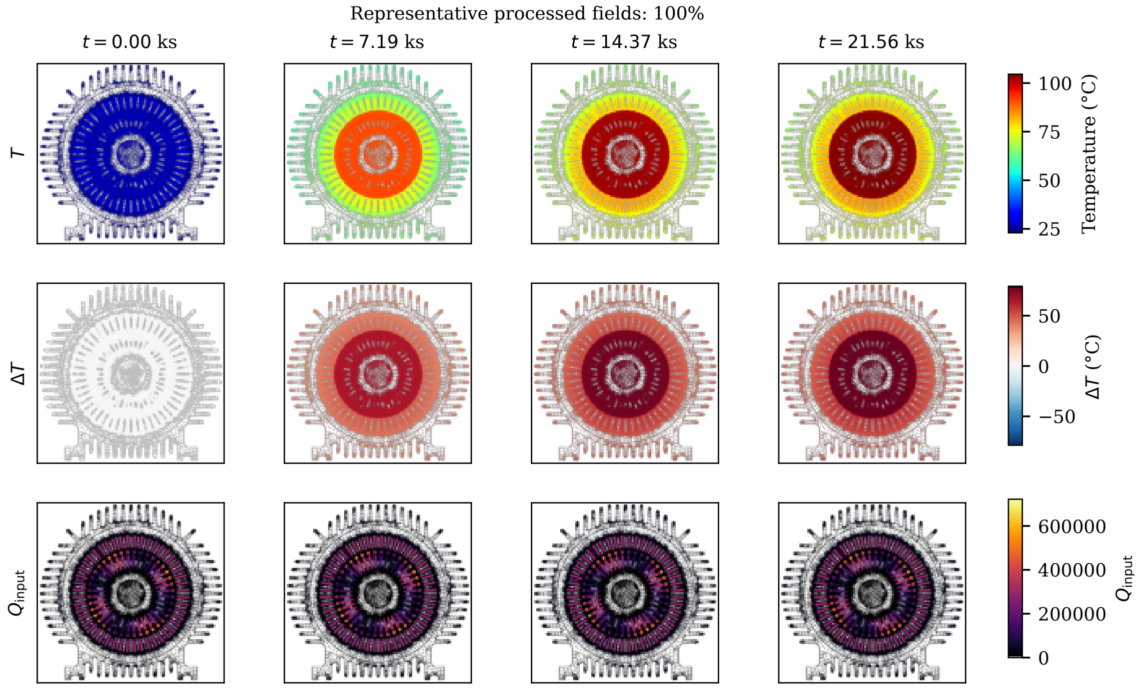

Figure 2.3: Representative processed fields for the 100% constant-load trajectory. From top to bottom: absolute temperature, temperature rise relative to the initial field, and the spatial heat-source input.

2.4 Chapter Summary

This chapter described how the available irregular-mesh FEM data were converted into a structured dataset for neural-operator learning. Six constant-load trajectories were interpolated onto a common 512×512 grid, and a geometry mask was used to isolate the machine domain. The resulting HDF5 dataset contains 600 nine-channel inputs and their corresponding temperature-rise targets. The next chapter describes the load-memory representation and the FNO used to learn the transient temperature-field mapping.

3. Load-Memory-Enhanced Fourier Neural Operator

3.1 Problem Formulation

The objective is to approximate the mapping from the processed input fields and operating history to the transient temperature-rise field:

$$\widehat{u}(\mathbf{x}, t_i) = \mathcal{G}_\theta \left(\mathbf{a}(\mathbf{x}, t_i) \right), \tag{3.1}$$

where $\mathcal{G}_\theta$ denotes the neural operator with trainable parameters θ. The model is direct: each queried time instance is predicted from the initial field and conditioning channels, without feeding previous temperature predictions back into the model. This avoids the accumulation of autoregressive rollout errors.

The instantaneous load alone does not uniquely determine the thermal state. For example, two samples at the same current load can have different temperatures if they were preceded by different load histories. Compact load-memory features are therefore added to the FNO input.

3.2 EWMA Load-Memory Features

Three exponentially weighted moving averages (EWMAs) summarize the recent load history at different temporal scales. An EWMA assigns exponentially decreasing weights to older observations and can therefore retain a compact summary without storing the complete sequence [17]. For a span s, the smoothing factor is

$$\alpha_s = \frac{2}{s+1}. \tag{3.2}$$

The memory is initialized to zero,

$$m_0^{(s)} = 0, \tag{3.3}$$

and uses previous alignment:

$$m_i^{(s)} = \alpha_s p_{i-1} + (1-\alpha_s) m_{i-1}^{(s)}, \qquad i \geq 1. \tag{3.4}$$

Thus, p_i represents the current load, whereas $m_i^{(s)}$ contains only information from preceding samples. At time t_i, the model predicts using p_i and the existing memory state. The memory is then updated using p_i before proceeding to the next sample.

The selected spans are $s \in \{3, 10, 30\}$, with smoothing factors 0.5, 2/11, and 2/31, respectively. They provide short-, medium-, and long-term summaries of the load history. Given the nominal sampling interval of 217.8 s, the span values correspond to nominal scales of approximately 10.9, 36.3, and 108.9 min. These are descriptive scales rather than finite averaging windows because the EWMA weights decay exponentially and remain non-zero for older samples.

Figure 3.1 illustrates the current-load channel and the three memory features for the independent varying-load profile. At the two switching times, the current load changes immediately while the memory features move gradually toward the new level. The span-3 feature responds most rapidly, whereas the span-30 feature retains the longest influence from the preceding operating stage. Consequently, the model can distinguish a sample that has been at 50% load for a long period from a sample that has only just changed from 100% to 50%, even though both have the same current load.

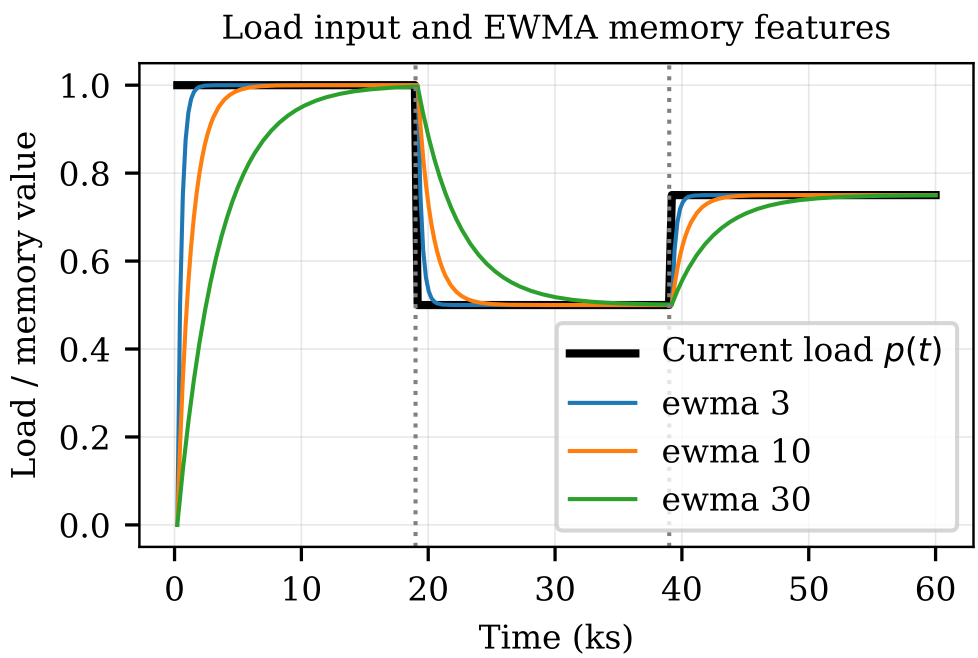

Figure 3.1: Current load and previous-aligned EWMA memory features for the independent 100%–50%–75% trajectory. The dotted vertical lines mark the load changes at 19 ks and 39 ks.

3.3 Fourier Neural Operator

Neural operators learn mappings between functions rather than only between fixed-dimensional vectors [13]. The Fourier Neural Operator parameterizes its global spatial interaction in Fourier space [15]. This makes it suitable for learning mappings between spatially distributed inputs and temperature fields.

The nine input channels are first lifted to a latent representation $\mathbf{v}_0$ with 32 channels. A Fourier layer can be written in the form

$$\mathbf{v}_{\ell+1} = \sigma\left(W_\ell \mathbf{v}_\ell + \mathcal{F}^{-1}\left[R_\ell \mathcal{F}(\mathbf{v}_\ell)\right]\right), \tag{3.5}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the forward and inverse Fourier transforms, R_ℓ contains learned weights for the retained Fourier modes, W_ℓ is a local linear transformation, and σ is the activation function. The spectral and local terms allow the layer to combine global spatial interactions with local feature transformations.

The implemented model contains four Fourier layers with 16 retained modes in each spatial direction. GELU activations are used, and a projection network maps the final latent representation to one normalized temperature-rise channel. The complete model contains 603,489 trainable parameters.

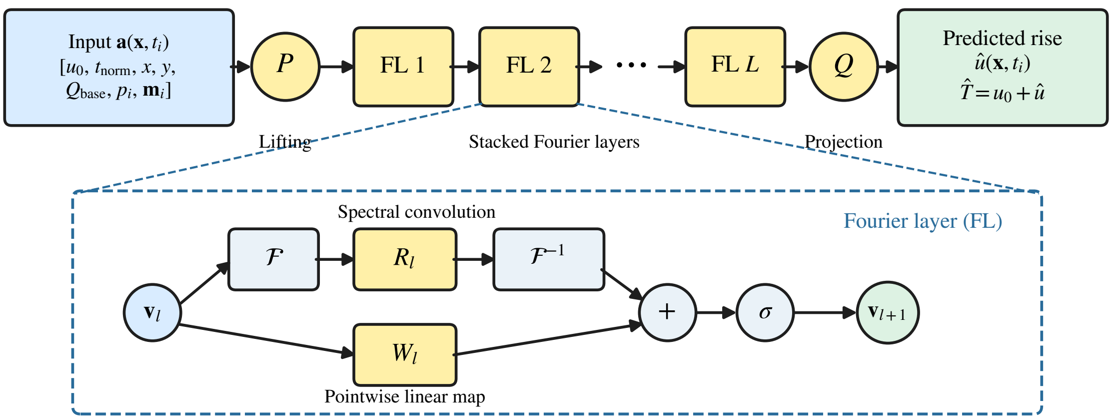

Figure 3.2: Architecture of the direct load-memory-enhanced FNO. Spatial fields, the current query, and the three EWMA features are assembled into a nine-channel tensor. A lifting layer maps the input to 32 latent channels, four Fourier layers process the latent field, and the projection produces one normalized temperature-rise channel.

Figure 3.2 also shows why the formulation is direct rather than autoregressive. The input contains the initial field and the load history but no previously predicted temperature field. Each requested time instance is therefore evaluated independently once its time, current load, and EWMA states have been constructed.

The network output is converted from training-normalized space to physical temperature rise using

$$\widehat{u}(\mathbf{x}, t_i) = \sigma_{u,\mathrm{train}} \widehat{\widetilde{u}}(\mathbf{x}, t_i) + \mu_{u,\mathrm{train}}. \tag{3.6}$$

The absolute temperature field is then reconstructed as

$$\widehat{T}(\mathbf{x}, t_i) = u_0(\mathbf{x}) + \widehat{u}(\mathbf{x}, t_i). \tag{3.7}$$

3.4 Loss Function and Training Configuration

Training minimizes the mean-squared error over active motor pixels:

$$\mathcal{L} = \frac{\sum_{j=1}^{N} \sum_{\mathbf{x}} M(\mathbf{x}) \left[\widehat{\widetilde{u}}_j(\mathbf{x}) - \widetilde{u}_j(\mathbf{x})\right]^2}{N \sum_{\mathbf{x}} M(\mathbf{x})}, \tag{3.8}$$

where N is the batch size and M is the geometry mask. Pixels outside the machine do not influence the optimization.

The main model and training parameters are listed in Table 3.1.

The configuration in Table 3.1 was selected through preliminary experiments and gave the best performance among the settings that were tested. The reported runs use a fixed 100-epoch schedule; the cosine scheduler reduces the learning rate from 10^{-3} to 10^{-5}. An exhaustive architecture search and a formal ablation of the three EWMA channels were not performed. The resulting values should therefore be interpreted as the selected experimental configuration, not as globally optimal hyperparameters.

3.5 Chapter Summary

The proposed model combines a direct FNO with three previous-aligned EWMA load-memory variables. The current load describes the operating condition at the queried time, while the EWMA channels summarize earlier load exposure. The FNO maps these conditioning variables and the spatial input fields to a complete temperature-rise field in a single forward evaluation.

Table 3.1: FNO architecture and training hyperparameters.

Parameter	Value
Input channels	9
Output channels	1
Fourier layers	4
Latent width	32
Retained modes	16×16
Activation	GELU
Optimizer	AdamW
Initial learning rate	10^{-3}
Weight decay	10^{-4}
Batch size	8
Maximum epochs	100
Scheduler	Cosine annealing
Gradient clipping	1.0
Loss	Masked MSE
Random seed	42

4. Experimental Methodology

4.1 Evaluation Design

Two experiments are used to assess the model. The first measures generalization to held-out constant-load conditions through leave-one-condition-out cross-validation. The second evaluates a final model on an independent varying-load trajectory containing heating, cooling, and reheating stages.

Table 4.1 summarizes the data used in the two experiments. Complete trajectories, rather than randomly selected snapshots, are assigned to the test sets. This prevents time-adjacent samples from the same FEM run from appearing in both training and testing.

Table 4.1: Training, validation, and test data used in the two evaluation protocols.

Experiment	Training data	Validation data	Test data
Constant-load LOCO	First 81 samples from each of three non-test trajectories	Final 19 samples from each of the same three trajectories	Complete held-out trajectory with 100 samples
Independent varying load	All 500 samples from the 50%, 62.5%, 75%, 87.5%, and 100% trajectories	None	Separate 100%–50%–75% trajectory with 276 time points

4.2 Leave-One-Condition-Out Cross-Validation

The cross-validation experiment uses the 25%, 50%, 75%, and 100% conditions. In each fold, one complete trajectory containing 100 samples is reserved for testing. For each of the three remaining conditions, the first 81 samples are used for training and the final 19 samples are used for validation. Each fold therefore contains 243 training samples, 57 validation samples, and 100 held-out test samples.

The models are trained for a maximum of 100 epochs. The checkpoint with the lowest validation loss is selected, while the held-out test trajectory is not evaluated during training or used for checkpoint selection. After model selection, the selected checkpoint is evaluated once on the complete held-out condition. This separation prevents information from the test trajectory from influencing model fitting or model selection.

4.2.1 Training-Only Target Normalization

The HDF5 file stores temperature-rise targets using global normalization for compatibility with the preprocessing workflow. Before training each cross-validation model, the physical temperature-rise values are reconstructed and new normalization statistics are calculated using only the active-domain pixels of the 243 training samples.

The normalized temperature-rise target is defined as

$$\widetilde{\Delta T} = \frac{\Delta T - \mu_{\Delta T,\mathrm{train}}}{\sigma_{\Delta T,\mathrm{train}}}, \tag{4.1}$$

where $\mu_{\Delta T,\mathrm{train}}$ and $\sigma_{\Delta T,\mathrm{train}}$ are the mean and standard deviation calculated from the training samples. The same training-derived parameters are applied unchanged to the validation and held-out test samples. Consequently, neither the validation samples nor the held-out condition contributes to the normalization statistics.

The predicted normalized temperature rise is transformed back to degrees Celsius according to

$$\widehat{\Delta T} = \widehat{\widetilde{\Delta T}}\, \sigma_{\Delta T,\mathrm{train}} + \mu_{\Delta T,\mathrm{train}}. \tag{4.2}$$

The normalization parameters are stored together with the configuration of each trained model and are reused during evaluation.

4.3 Independent Varying-Load Evaluation

The final model is trained using the 50%, 62.5%, 75%, 87.5%, and 100% constant-load conditions. It is evaluated on a separate FEM trajectory with the load profile

$$p(t) = \begin{cases} 1.00, & t < 19{,}000 \text{ s}, \\ 0.50, & 19{,}000 \le t < 39{,}000 \text{ s}, \\ 0.75, & t \ge 39{,}000 \text{ s}. \end{cases} \tag{4.3}$$

The complete trajectory extends to approximately 60,000 s and contains 276 evaluated time points. The individual load magnitudes are represented in the constant-load training data, but their temporal ordering and the associated heating, cooling, and reheating transitions are not.

The additional 62.5% and 87.5% conditions increase the coverage between 50% and 100%, which is the load range traversed by the independent trajectory. The 25% trajectory lies outside that range and is not used when training this final model. All 500 samples from the five selected constant-load conditions are used for the fixed 100-epoch training run.

At every evaluated time instance, the existing EWMA state is supplied to the model together with the current load. The temperature field is predicted first, after which the EWMA state is updated using the current load. This ensures that varying-load inference uses the same previous-alignment semantics as dataset construction.

4.4 Evaluation Metrics

All reported temperature errors are calculated after transforming the model output and reference target back to degrees Celsius. For one temperature field, the active-domain RMSE is

$$\mathrm{RMSE}_{\mathrm{field}} = \sqrt{\frac{\sum_{\mathbf{x}} M(\mathbf{x}) \left[\widehat{T}(\mathbf{x}) - T(\mathbf{x})\right]^2}{\sum_{\mathbf{x}} M(\mathbf{x})}}. \tag{4.4}$$

The corresponding MAE is calculated using the absolute error. Field metrics are averaged over the evaluated time instances. Two global temperature quantities are also extracted from every field:

$$T_{\mathrm{max}}(t_i) = \max_{\mathbf{x}:M(\mathbf{x})=1} T(\mathbf{x}, t_i), \tag{4.5}$$

and

$$T_{\mathrm{mean}}(t_i) = \frac{\sum_{\mathbf{x}} M(\mathbf{x}) T(\mathbf{x}, t_i)}{\sum_{\mathbf{x}} M(\mathbf{x})}. \tag{4.6}$$

RMSE and MAE are calculated for both time series. Selected spatial points are additionally used to inspect local temperature dynamics under the varying-load profile. Six points are distributed across cooler outer regions and hotter inner regions of the machine. These traces complement the global T_{max} and T_{mean} metrics by showing whether local heating, cooling, and reheating rates are reproduced at physically different locations.

4.5 Computational-Cost Measurement

The FEM execution time is measured over five repeated runs of the complete varying-load trajectory. Neural-network inference is measured after one untimed warm-up. CUDA is synchronized before and after the timed region. The inference measurement includes feature construction, model forward evaluations, output denormalization, and temperature-summary extraction for the complete profile. It excludes model loading, FEM-data rasterization, metric calculation, plotting, and file output.

The exact FEM CPU allocation was not recorded. The comparison is therefore reported as a measured wall-clock comparison for the available executions, not as a hardware-normalized or algorithmic speedup.

4.6 Implementation Environment

The data-processing, training, and evaluation workflows are implemented in Python using NumPy, SciPy, HDF5, PyTorch, and the NeuralOperator library. The FNO experiments were executed on an NVIDIA A100 80 GB PCIe GPU. The software versions and principal execution commands are provided in Appendix A.

4.7 Chapter Summary

The experimental design separates constant-load condition generalization from dynamic varying-load evaluation. Cross-validation holds out entire load conditions, while the independent trajectory tests a load ordering absent from training. Training-only target normalization and validation-based checkpoint selection prevent the held-out test conditions from influencing model fitting or model selection.

5. Results

5.1 Generalization Across Constant Loads

Table 5.1 presents the mean and population standard deviation across the four leave-one-condition-out folds. The mean field RMSE is 4.82 °C, while the mean field MAE is 4.68 °C. The global maximum-temperature RMSE is 5.68 °C, and the spatial-mean-temperature RMSE is 5.13 °C.

Table 5.1: Mean and population standard deviation across the four leave-one-condition-out folds.

Metric	Mean [°C]	Std. dev. [°C]
Field RMSE	4.82	2.22
Field MAE	4.68	2.19
T_{max} RMSE	5.68	3.20
T_{max} MAE	5.17	2.95
T_{mean} RMSE	5.13	2.37
T_{mean} MAE	4.65	2.20

The fold-level results are shown in Table 5.2. The 50% held-out condition gives the lowest errors, with a field RMSE of 2.71 °C and a T_{max} RMSE of 0.96 °C. The 75% condition also produces relatively low field and mean-temperature errors. By contrast, the 25% condition gives the largest T_{max} error, and the 100% condition produces the largest field and spatial-mean errors.

Table 5.2: Fold-level cross-validation RMSE values.

Held-out load	Field [°C]	T_{max} [°C]	T_{mean} [°C]
25%	5.58	9.99	6.08
50%	2.71	0.96	2.87
75%	2.87	5.85	3.00
100%	8.11	5.93	8.59

The intermediate-load folds are interpolation cases because the training sets contain conditions below and above the held-out load. Their lower errors indicate that the model more reliably interpolates between represented operating conditions than extrapolates beyond the training range. The 25% and 100% folds instead require boundary extrapolation.

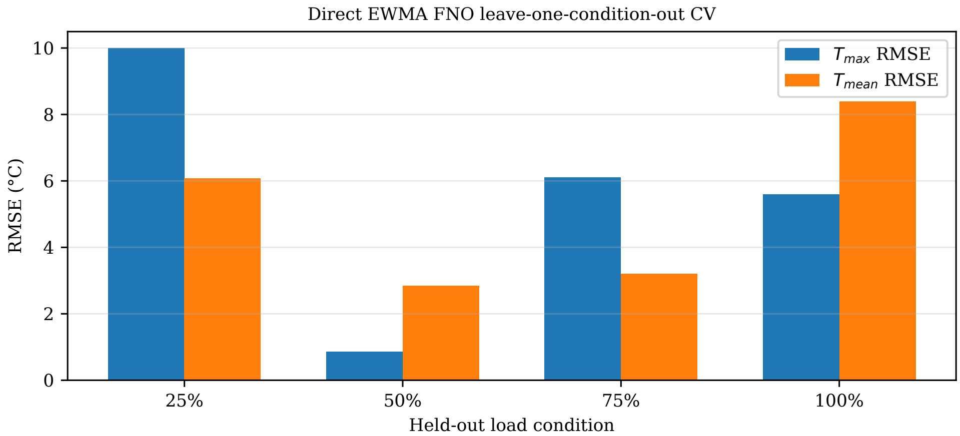

Figure 5.1: Maximum- and spatial-mean-temperature RMSE for each leave-one-condition-out fold. The intermediate 50% and 75% cases produce lower spatial-mean errors than the two boundary cases.

Figure 5.1 makes the difference between the folds visible. The 50% case has the lowest $T_{\max}$ RMSE, while the 75% case retains a low T_{mean} error but has a larger hotspot error. The 25% fold is most difficult for maximum temperature, whereas the 100% fold is most difficult for the spatial mean. The two metrics therefore expose different failure modes: a prediction may reproduce the global temperature level while missing a localized hotspot, or it may preserve the hotspot level while shifting a broader part of the field.

Figure 5.2 presents the complete spatial prediction at six times for the best-performing 50% fold. The predicted field preserves the main radial temperature gradients and hotspot region. The signed-error row shows that the error becomes increasingly negative over the inner machine region as the trajectory approaches its final thermal level. This explains why the field error grows with elapsed time even though the overall spatial pattern remains recognizable.

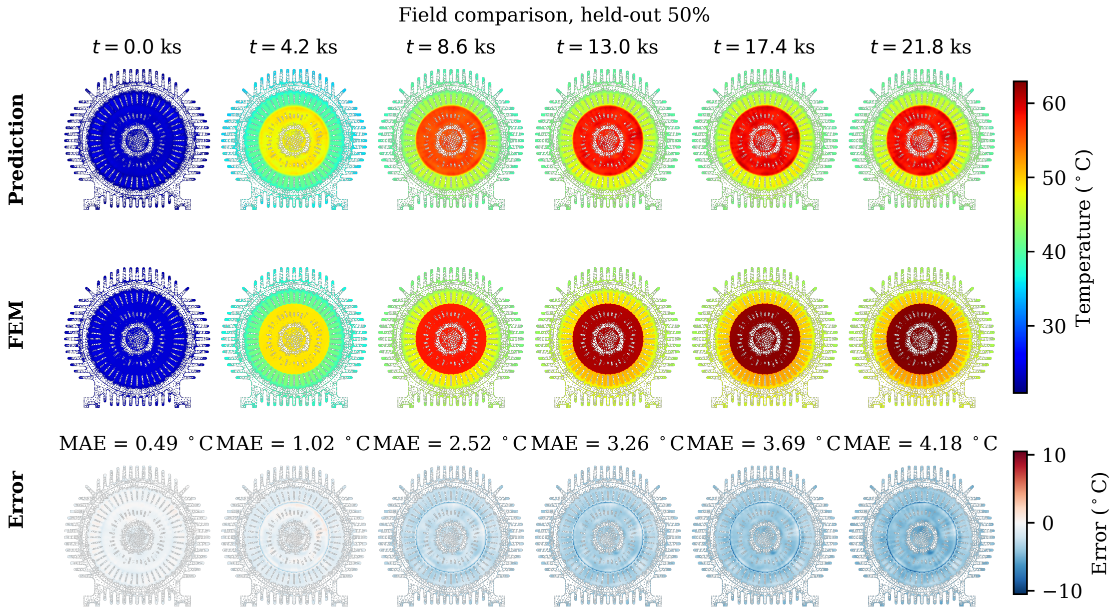

Figure 5.2: Temperature-field comparison for the held-out 50% condition at six elapsed times. The rows show the FNO prediction, FEM reference, and signed error $\widehat{T} - T$, respectively. The value above each error field is its active-domain MAE.

5.2 Varying-Load Prediction

The final model is evaluated on the independent 100%–50%–75% load profile. Table 5.3 summarizes the global time-series errors. The model achieves a T_{max} RMSE of 4.61 °C and a T_{mean} RMSE of 3.02 °C. The corresponding MAE values are 4.27 and 2.62 °C, respectively.

Table 5.3: Prediction errors for the independent varying-load profile.

Metric	Error [°C]
T_{max} RMSE	4.61
T_{mean} RMSE	3.02
T_{max} MAE	4.27
T_{mean} MAE	2.62

The predicted maximum and spatial mean temperatures follow the principal heating, cooling, and reheating stages of the FEM trajectory. The lower error for T_{mean} indicates that the global thermal evolution is captured more accurately than the localized maximum temperature. The largest changes in prediction error occur near the load-switching times, where the thermal response transitions between operating conditions.

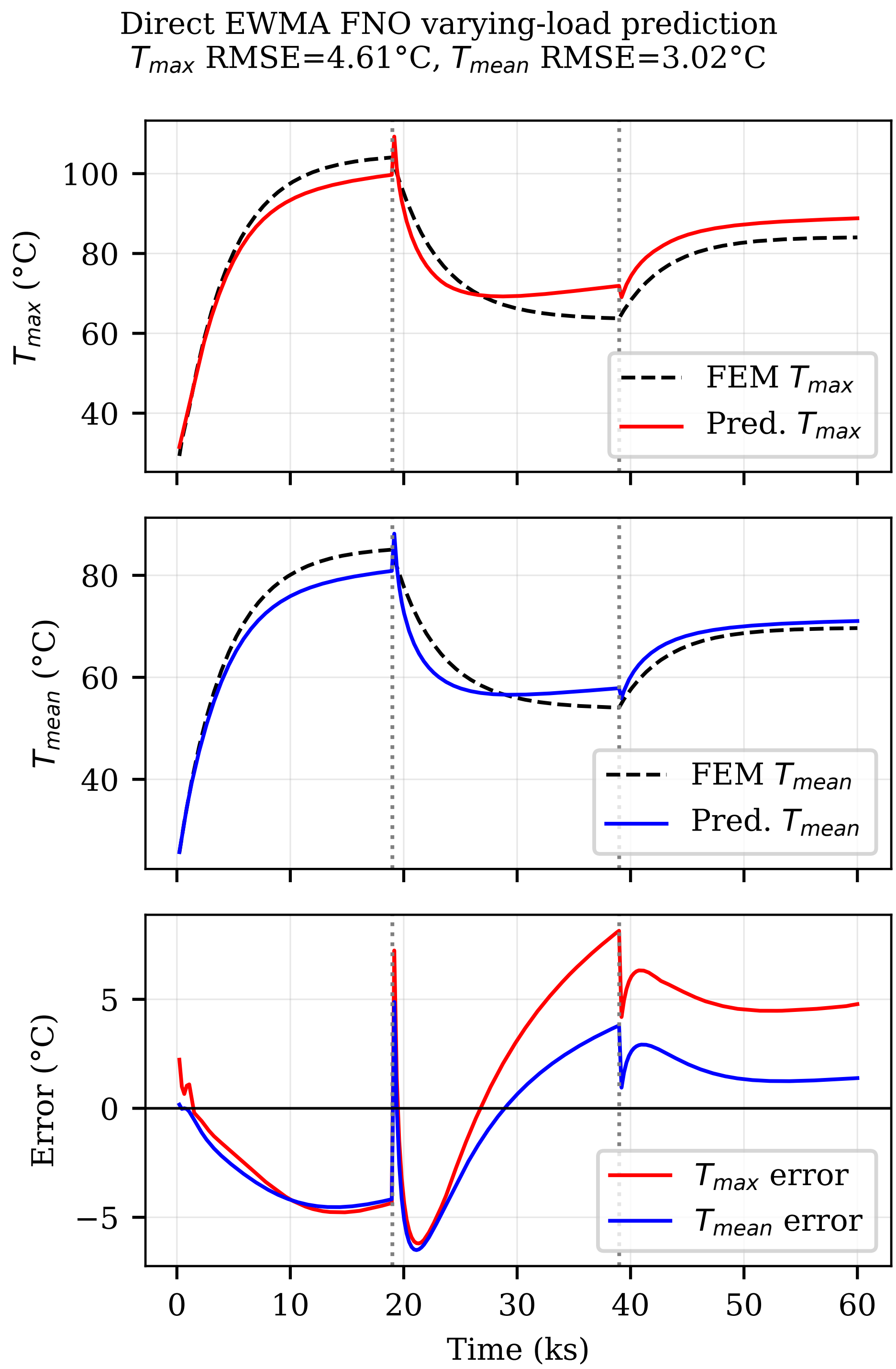

Figure 5.3: Predicted and FEM-reference maximum and spatial-mean temperatures for the independent varying-load trajectory. The bottom panel shows the signed prediction errors, and the dotted lines mark the load changes at 19 ks and 39 ks.

As shown in Fig. 5.3, the model follows the initial heating stage but gradually underpredicts both global quantities before the first load change. Immediately after the reduction from 100% to 50%, the model reproduces the cooling trend, although the signed errors temporarily become more negative. During the later part of the 50% stage, the errors change sign. Following the increase to 75%, the model captures the renewed heating but slightly overpredicts the final temperature level. The error is therefore structured in time rather than behaving as uncorrelated noise.

Representative field comparisons show that the model reproduces the main spatial temperature distribution and hotspot region throughout the varying profile. Pointwise traces at selected machine locations likewise follow the overall transient response, although the local errors depend on position and operating stage.

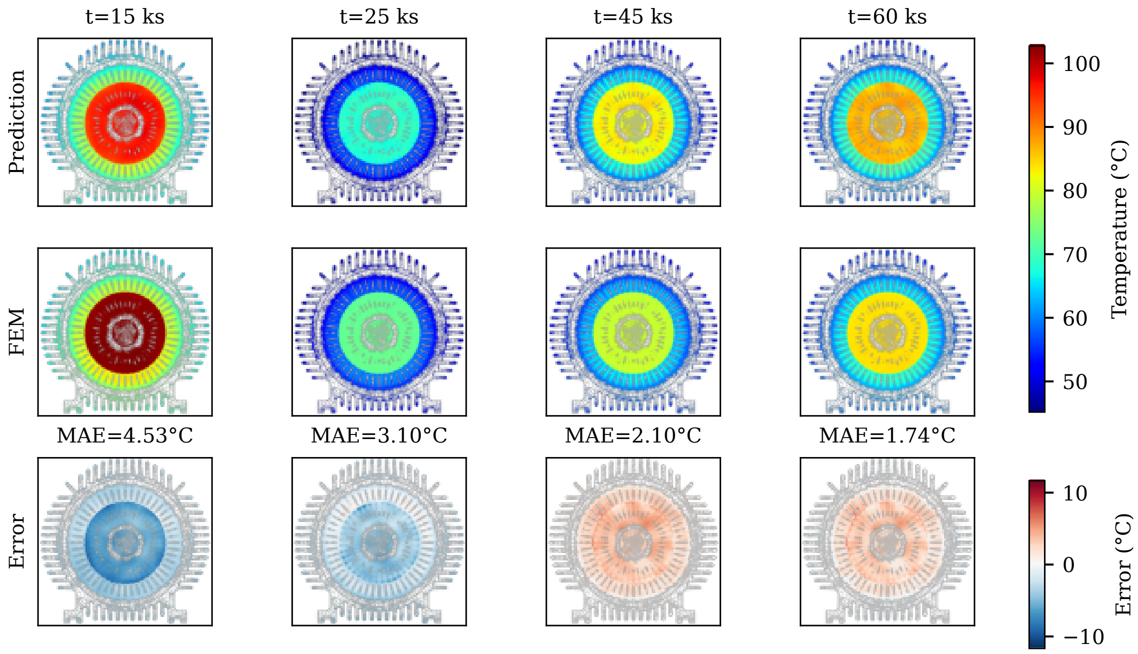

Figure 5.4: Spatial temperature-field comparison for the independent varying-load trajectory. The columns correspond to selected times in the heating, cooling, and reheating stages. The rows show the FNO prediction, FEM reference, and signed error $\widehat{T} - T$.

The field comparisons in Fig. 5.4 have active-domain MAE values of 4.53, 3.10, 2.10, and 1.74 °C at 15, 25, 45, and 60 ks, respectively. At 15 and 25 ks, the error is predominantly negative over the warmer inner region. At 45 and 60 ks, the sign changes and the model slightly overpredicts much of that region. Nevertheless, the location and overall shape of the high-temperature region are maintained through all three operating stages.

The pointwise RMSE values are summarized in Table 5.4. The two cooler outer points, P1 and P2, give the lowest RMSE values. The four points closer to the hotter inner region have larger errors, with P4 and P5 reaching approximately 4.1 °C. Figure 5.6 nevertheless shows that all six traces reproduce the ordering and principal slopes of the heating, cooling, and reheating stages.

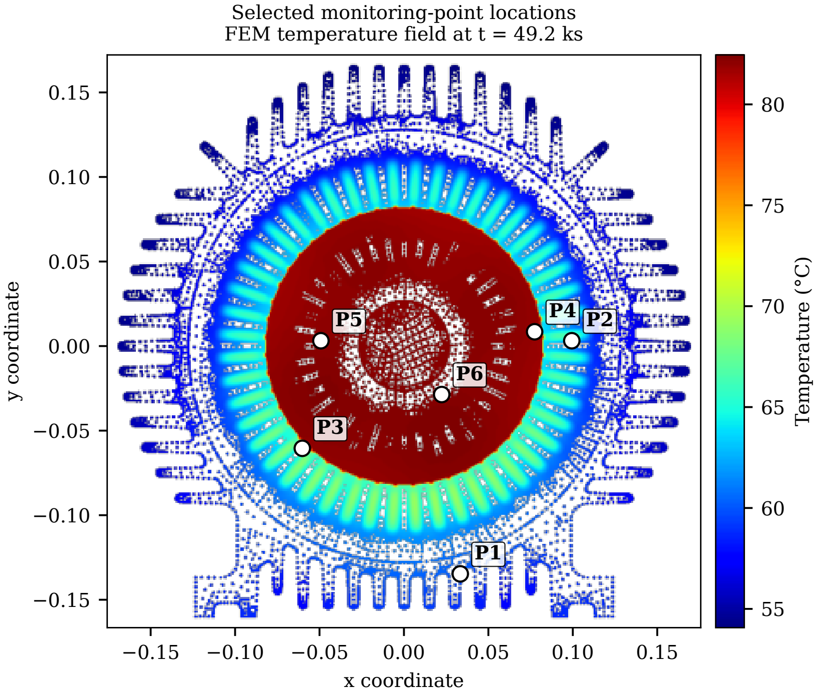

Figure 5.5: Six monitoring points selected across cooler outer and hotter inner regions. The background is the FEM temperature field at approximately 49.2 ks.

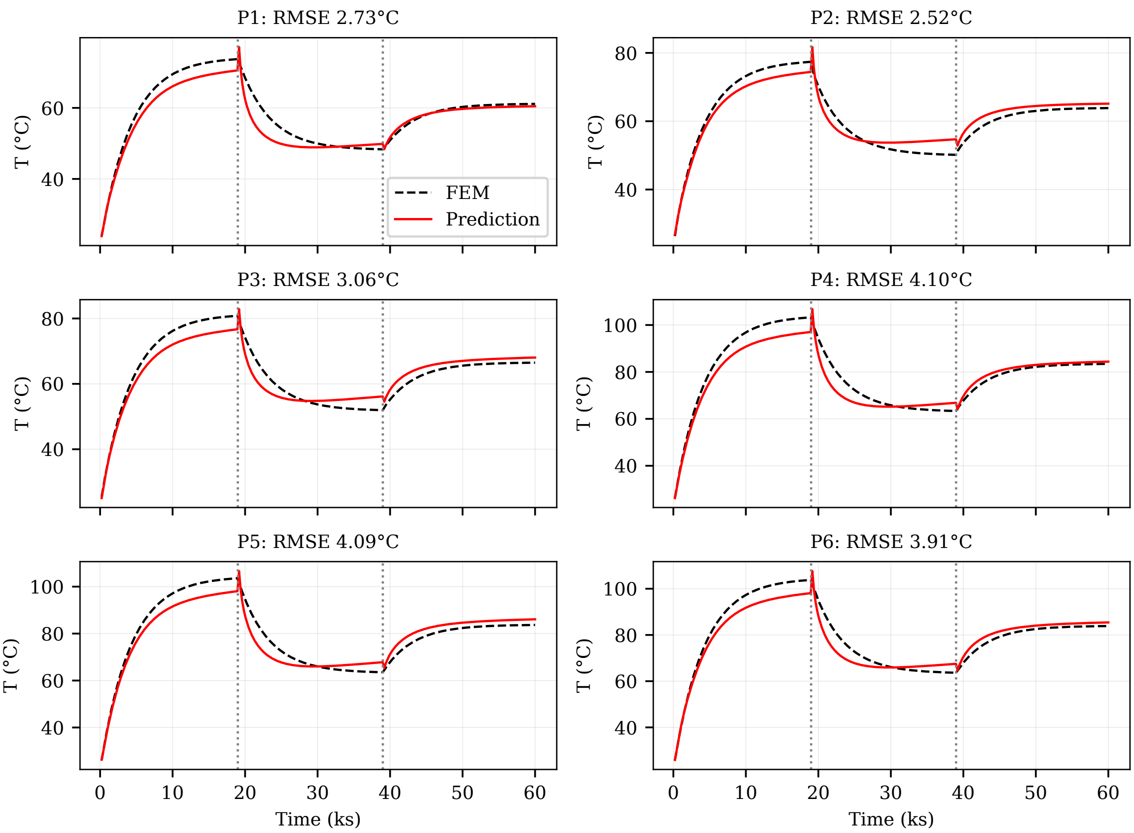

Figure 5.6: FEM-reference and predicted temperature traces at the six monitoring points in Fig. 5.5. The dotted lines indicate the two load changes.

Table 5.4: Pointwise errors for the independent varying-load trajectory.

Point	RMSE [°C]	MAE [°C]	Max. absolute error [°C]
P1	2.73	2.00	7.64
P2	2.52	2.28	6.46
P3	3.06	2.70	6.33
P4	4.10	3.24	9.34
P5	4.09	3.58	9.00
P6	3.91	3.24	8.90

5.3 Computational Performance

Five FEM timing runs give a mean execution time of 161.12 s, a median of 160.27 s, and a standard deviation of 5.64 s. A representative three-condition cross-validation model requires approximately 378 s to train for 100 epochs. The final five-condition model requires 624.71 s, or approximately 10.4 min. Both training times are one-time offline costs and are not included in the inference speedup.

The complete 276-point varying-load trajectory is evaluated by the trained model in 2.19 s. The mean inference time per requested time point is 0.00792 s. Relative to the mean FEM time, the measured wall-clock speedup is

$$S = \frac{161.12}{2.187249} \approx 73.67. \tag{5.1}$$

This corresponds to an execution-time reduction of approximately 98.6%. The computational results are summarized in Table 5.5.

Table 5.5: Measured computational cost.

Operation	Time [s]	Relative speedup
FEM trajectory	161.12 ± 5.64	1×
FNO training, three conditions	378.00	Not applicable
FNO training, five conditions	624.71	Not applicable
FNO inference per point	0.00792	–
FNO complete trajectory	2.19	73.67×

The reported value is a measured wall-clock comparison rather than a hardware-normalized speedup. Nevertheless, it demonstrates the intended advantage of the surrogate: after the one-time training stage, complete temperature-field trajectories can be generated substantially faster than the corresponding FEM execution.

5.4 Results Summary

The cross-validation experiment shows that prediction accuracy depends on the position of the test condition relative to the training range. Intermediate conditions are predicted more accurately than boundary conditions requiring extrapolation. On the independent varying-load trajectory, the model reproduces the main heating, cooling, and reheating dynamics with RMSE values of 4.61 °C for maximum temperature and 3.02 °C for spatial mean temperature. Complete-trajectory inference requires approximately 2.2 s, corresponding to an approximately 74-fold measured wall-clock speedup over FEM.

The implications, limitations, and directions for future work associated with these results are discussed in the following chapter.

6. Discussion

6.1 Overview of the Main Findings

This thesis investigated whether a load-memory-enhanced Fourier Neural Operator can act as a fast surrogate for transient thermal-field prediction in the studied 15 kW squirrel-cage induction motor. The results support this possibility within the range covered by the available data. The model reproduces the main spatial temperature pattern for held-out constant-load conditions and follows the heating, cooling, and reheating behavior of the independent varying-load trajectory. However, its accuracy is not uniform. Predictions are more reliable for loads located inside the training range than for boundary conditions requiring extrapolation, and the global thermal level is reproduced more accurately than the localized maximum temperature.

The results also demonstrate the computational advantage of the surrogate. Once training has been completed, the full 276-point varying-load trajectory is evaluated in approximately 2.19 s, compared with a mean FEM execution time of 161.12 s. This speed comes with a restricted domain of validity: the model learns from the supplied FEM trajectories and should not be expected to replace the reference model for motor geometries, cooling configurations, or operating conditions that are not represented by the training data.

6.2 Generalization Across Constant Loads

The leave-one-condition-out experiment evaluates whether the FNO can predict an entire constant-load trajectory that was excluded from training. Across the four folds, the mean field RMSE is 4.82 °C. This aggregate value shows that the model can approximate unseen constant-load responses, but the fold-level results are more informative than the average alone.

The held-out 50% and 75% conditions give field RMSE values of 2.71 and 2.87 °C, respectively. In both cases, training conditions exist below and above the test load. The model can therefore interpolate between thermal responses represented during training. The 25% and 100% conditions instead lie at the boundaries of the evaluated load range and give field RMSE values of 5.58 and 8.11 °C. Their higher errors are consistent with the greater difficulty of extrapolation. The available results therefore suggest that coverage of the intended operating range is more important than simply increasing the number of samples along already represented trajectories.

The signed errors in Fig. 5.2 also become more negative in the warmer inner region at later times. The model retains the main spatial pattern while underestimating its magnitude. This distinction is important for application: identifying the approximate hotspot location is not equivalent to predicting the hotspot temperature with the same accuracy.

6.3 Prediction Under a Varying Load

The independent 100%–50%–75% trajectory provides a more demanding test than evaluating additional snapshots from the constant-load simulations. Its three load magnitudes are represented in the training set, but their ordering and the associated transitions are not. The resulting evaluation therefore tests generalization to an unseen operating sequence within a represented load range; it does not test extrapolation to previously unseen load magnitudes.

The model obtains RMSE values of 4.61 °C for T_{max} and 3.02 °C for T_{mean}. It follows the principal temperature slopes during initial heating, cooling after the first load reduction, and reheating after the second load change. This shows that a direct prediction formulation can reproduce the dominant transient response without using previously predicted temperatures as inputs.

The lower T_{mean} error indicates that the model captures the overall thermal level more accurately than the most temperature-sensitive local region. This interpretation is supported by the pointwise results. The cooler outer points P1 and P2 have RMSE values between 2.52 and 2.73 °C, whereas points in the warmer inner region reach approximately 4.1 °C. Nevertheless, all six traces retain the correct ordering and principal heating and cooling trends. The model therefore provides a useful approximation of the complete field, while localized hotspot estimates remain the more demanding output.

The varying-load errors are structured rather than random. The model first underpredicts the thermal level, the error changes following the transition to 50% load, and the final 75% stage is slightly overpredicted. A likely interpretation is that the learned mapping approximates the main dynamic response but does not reproduce all time constants and transition behavior with equal accuracy. Additional load profiles with different switching times and stage durations would be required to determine whether this pattern is specific to the evaluated trajectory or represents a more general model bias.

6.4 Computational Performance and Practical Use

The measured complete-trajectory inference time of 2.19 s corresponds to a 73.67-fold speedup and a 98.6% execution-time reduction relative to the mean FEM time. The final model requires approximately 10.4 min to train, but this is an offline cost. The computational advantage therefore becomes most relevant when a trained model is reused for repeated evaluations, rapid screening of operating sequences, or other tasks in which full spatial temperature fields are requested many times.

The comparison should be interpreted as a practical wall-clock measurement, not as a hardware-independent comparison of the two methods. The exact FEM CPU allocation was not recorded, and the FNO was evaluated on an NVIDIA A100 GPU. In addition, the inference timing assumes that the trained model and the required regular-grid inputs are available; model loading, preprocessing of raw FEM data, metric calculation, plotting, and file output are excluded. These choices are appropriate for measuring repeated surrogate inference, but they limit broader claims about algorithmic efficiency.

Within these constraints, the result confirms the central computational motivation of the study. The surrogate does not reproduce the FEM reference exactly, but it produces complete temperature-field trajectories in a small fraction of the measured execution time. Whether this accuracy–speed trade-off is acceptable depends on the intended use and the permitted error near temperature-critical regions.

6.5 Limitations

The study is limited by the amount and scope of the available data. The dataset describes one motor through a two-dimensional FEM representation and contains six constant-load trajectories and one varying-load trajectory. The independent sequence is valuable because its load ordering is absent from training, but one sequence cannot demonstrate generalization to arbitrary duty cycles. Rapid load fluctuations, longer operating sequences, loads outside the represented range, and different initial thermal states are not evaluated. In addition, the model is evaluated against FEM results rather than physical measurements. Agreement with the reference data therefore does not constitute experimental validation of the physical motor, and any uncertainty in the FEM model is inherited by the surrogate.

The FEM outputs are interpolated from an irregular mesh to one fixed 512×512 grid. Although preprocessing checks confirm consistency with the earlier workflow, interpolation can smooth local gradients and produces a representation that differs from the original mesh. Generalization across grid resolutions is not investigated. The model also uses one base heat-source field together with a scalar load channel, so operating changes that cannot be described by these inputs are outside the learned formulation.

The FNO hyperparameters and EWMA spans were selected through preliminary experiments rather than an exhaustive search. No formal comparison was made with an EWMA-free FNO, an alternative history representation, or a non-neural reduced-order baseline. The reported results therefore establish the performance of the selected complete model, but not the individual benefit of each design choice. The experiments also use a fixed random seed, without repeated training runs. The cross-validation standard deviations describe variation among the four held-out load conditions rather than variation caused by random initialization or stochastic training.

Finally, the computational comparison uses the available wall-clock measurements. The exact FEM CPU allocation was not recorded, and the compared methods do not use identical computational resources. The reported speedup is therefore specific to the measured environment and timing scope described in Section 6.4.

6.6 Future Work

The most important next step is to expand the dataset. Additional constant-load conditions near and beyond the current boundaries would test and potentially improve extrapolation. Multiple varying-load trajectories with different load orders, switching intervals, and initial temperatures would provide stronger evidence of dynamic generalization. Comparison with physical temperature measurements would additionally determine how errors from the FEM reference and the learned surrogate combine in practice.

A controlled ablation study should compare the complete model with an otherwise identical FNO using only the current load, as well as variants with individual EWMA spans or alternative history representations. This would quantify whether the memory channels improve varying-load prediction and whether all three temporal scales are necessary. Repeated training runs and a more systematic hyperparameter study would help separate architectural effects from random variation.

Further improvements could target the warmer inner region directly. Possible approaches include region-weighted or hotspot-aware loss functions, multi-resolution field representations, and increased spectral capacity. Extension to three-dimensional data and to multiple motor geometries, materials, and cooling configurations would be required before making claims about transfer beyond the present case. Finally, recording complete hardware and software configurations for both FEM and FNO execution would enable a more controlled computational comparison

6.7 Chapter Summary

The results show that the proposed FNO provides useful transient temperature-field predictions and a substantial measured reduction in evaluation time. Generalization is strongest for interpolation within the training range, while boundary extrapolation and localized hotspot temperature remain more challenging. The independent varying-load experiment supports the feasibility of the complete load-memory-enhanced model, but the limited data and absence of an EWMA ablation restrict the conclusions that can be drawn about arbitrary duty cycles and the individual contribution of the memory features.

7. Conclusions

This thesis developed a direct load-memory-enhanced Fourier Neural Operator for predicting two-dimensional transient temperature fields in a 15 kW squirrel-cage induction motor. The work included transforming irregular-mesh FEM outputs into a structured HDF5 dataset, representing recent load exposure with three previous-aligned EWMAs, and evaluating the resulting surrogate on held-out constant-load conditions and an independent varying-load sequence.

The leave-one-condition-out experiment produced a mean field RMSE of 4.82 °C across four constant-load conditions. Prediction accuracy was higher for the intermediate 50% and 75% loads than for the boundary conditions, indicating that the model interpolates more reliably within the represented operating range than it extrapolates beyond it.

For the independent 100%–50%–75% trajectory, the model reproduced the principal heating, cooling, and reheating behaviour. The RMSE values were 4.61 °C for maximum temperature and 3.02 °C for spatial mean temperature. The overall thermal evolution was captured more accurately than the localized hotspot response. Although the complete model uses EWMA load-memory features, their individual contribution cannot be determined without an ablation study.

Complete-trajectory inference required approximately 2.19 s, compared with a mean FEM execution time of 161.12 s. This corresponds to an approximately 74× measured wall-clock speedup. The result demonstrates the practical computational advantage of the trained surrogate, although the comparison is specific to the available computing environment.

Overall, the proposed FNO provides a fast approximation of the transient temperature distribution while preserving the main spatial pattern and dynamic response. Its current applicability is limited to the studied motor and the operating range represented by the training data. Further validation should include additional load histories, improved coverage near the load boundaries, ablation of the memory features, and comparison with physical measurements.

References

[1] Z.-Q. Zhu and D. Liang, "Perspective of thermal analysis and management for permanent magnet machines, with particular reference to hotspot temperatures," *Energies*, vol. 15, no. 21, p. 8189, 2022. DOI: 10.3390/en15218189.

[2] E. Gundabattini, A. Mystkowski, A. Idzkowski, R. Singh R., and D. G. Solomon, "Thermal mapping of a high-speed electric motor used for traction applications and analysis of various cooling methods—a review," *Energies*, vol. 14, no. 5, p. 1472, 2021. DOI: 10.3390/en14051472.

[3] D. A. Staton and A. Cavagnino, "Convection heat transfer and flow calculations suitable for electric machines thermal models," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 10, pp. 3509–3516, Oct. 2008. DOI: 10.1109/TIE.2008.922604.

[4] X. Huang, A. Ali, P. Boulanger, U. Nasir, K. Floros, and J. Chandrappan, "Enhanced thermal impedance analysis with reduced-order modelling: A fast and efficient computational approach," in *Proceedings of ECCE Europe*, 2025, pp. 1–5. DOI: 10.1109/ECCE-Europe62795.2025.11238941.

[5] M. Jimenez, J. Tapia, P. Lindh, L. Aarniovuori, A. Anttila, and J. Pyrhonen, "Alternative procedures for thermal design on electric machine," in *Proceedings of the IEEE International Electric Machines and Drives Conference (IEMDC)*, 2021, pp. 1–6. DOI: 10.1109/IEMDC47953.2021.9449555.

[6] K. L. Kiss and T. Orosz, "Model order reduction methods for rotating electrical machines: A review," *Energies*, vol. 17, no. 20, p. 5145, 2024. DOI: 10.3390/en17205145.

[7] W. Kirchgassner, O. Wallscheid, and J. Bocker, "Thermal neural networks: Lumped-parameter thermal modeling with state-space machine learning," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105 537, 2023. DOI: 10.1016/j.engappai.2022.105537.

[8] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019. DOI: 10.1016/j.jcp.2018.10.045.

[9] Z. Li et al., "Physics-informed neural operator for learning partial differential equations," *ACM/IMS Journal of Data Science*, 2024. DOI: 10.1145/3648506.

[10] Y. Yang et al., "Surrogate modeling of parameterized SiC power module thermal fields with physics-informed neural networks," in *Proceedings of the IEEE Energy Conversion Congress and Exposition (ECCE)*, 2025, pp. 1–6. DOI: 10.1109/ECCE58356.2025.11260085.

[11] X. Zhao, Z. Gong, Y. Zhang, W. Yao, and X. Chen, *Physics-informed convolutional neural networks for temperature field prediction of heat source layout without labeled data*, 2021. arXiv: `2109.12482 [cs.LG]`. [Online]. Available: `https://arxiv.org/abs/2109.12482`.

[12] T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, and P. Battaglia, "Learning mesh-based simulation with graph networks," in *International Conference on Learning Representations*, 2021. [Online]. Available: `https://openreview.net/forum?id=roNqYL0_XP`.

[13] N. Kovachki et al., "Neural operator: Learning maps between function spaces with applications to PDEs," *Journal of Machine Learning Research*, vol. 24, no. 89, pp. 1–97, 2023.

[14] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators," *Nature Machine Intelligence*, vol. 3, pp. 218–229, 2021. DOI: `10.1038/s42256-021-00302-5`.

[15] Z. Li et al., "Fourier neural operator for parametric partial differential equations," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[16] Y. Tang, C. Zhan, L. Zhu, Z. Zhao, K. Zhao, and S. Ji, "Advanced spatial temperature monitoring of power modules via fourier neural operator-based thermal model," *IEEE Transactions on Power Electronics*, vol. 41, no. 3, pp. 3154–3160, Mar. 2026. DOI: `10.1109/TPEL.2025.3611030`.

[17] E. Luxenberg and S. Boyd, "Exponentially weighted moving models," *arXiv preprint arXiv:2404.08136*, 2024. DOI: `10.48550/arXiv.2404.08136`.

A. Implementation and Reproducibility Details

This appendix summarizes the software environment, dataset configuration, model hyperparameters, and principal commands used to reproduce the reported experiments. The complete dependency specifications are retained in the project files `requirements.txt` and `environment.yml`.

A.1 Software and Computing Environment

The data preprocessing, model training, evaluation, and visualization were implemented in Python. The principal software versions recorded in the project environment are listed in Table A.1.

Table A.1: Principal software used in the implementation.

Software	Version
Python	3.11.14
PyTorch	2.5.1 with CUDA 12.1 support
NeuralOperator	2.0.0
NumPy	2.4.2
SciPy	1.17.0
h5py	3.15.1
Matplotlib	3.10.8

The FNO training and inference experiments were executed on an NVIDIA A100 80 GB PCIe GPU. The exact CPU configuration used for the reference FEM executions was not recorded. Consequently, the reported computational comparison represents the measured wall-clock times in the available computing environments rather than a hardware-normalized algorithmic comparison.

A.2 Dataset Configuration

Table A.2 summarizes the principal configuration used to generate the structured HDF5 dataset.

The main arrays stored in the HDF5 dataset are:

Table A.2: Dataset and preprocessing configuration.

Setting	Value
Constant-load conditions	25%, 50%, 62.5%, 75%, 87.5%, and 100%
Samples per condition	100
Total constant-load samples	600
Structured-grid resolution	512×512
Primary interpolation method	Linear interpolation
Extrapolation and missing-value treatment	Nearest-neighbour filling
Active geometry pixels	113,659, corresponding to approximately 43.36% of the grid
Heat-source input	Fixed base heat-source field, with current load supplied separately
EWMA spans	3, 10, and 30 samples
EWMA alignment	Previous-aligned; the stored memory state excludes the current load
EWMA initial state	Zero
Prediction target	Temperature rise, $\Delta T(\mathbf{x}, t) = T(\mathbf{x}, t) - T(\mathbf{x}, 0)$
Input-array shape	$600 \times 9 \times 512 \times 512$
Target-array shape	$600 \times 1 \times 512 \times 512$
Storage precision	32-bit floating point
HDF5 compression	Gzip, compression level 4

- `X_all`: nine-channel model inputs containing the initial temperature, normalized elapsed time, spatial coordinates, base heat-source field, current load, and three EWMA load-memory features;
- `Y_all`: normalized temperature-rise targets;
- `mask`: binary geometry mask;
- `x_grid` and `y_grid`: physical grid coordinates;
- `power_fraction_all`: instantaneous load fractions;
- `time_norm_all`: normalized elapsed times;
- `elapsed_time_all`: physical elapsed times; and
- condition-specific sample indices.

The three EWMA features are stored directly as channels 7–9 of `X_all`, rather than as separate top-level HDF5 arrays. Since the current load and EWMA values are scalar quantities at each time point, each value is broadcast over the complete 512×512 spatial grid.

A.3 Model and Training Configuration

The FNO and optimization settings used in the reported experiments are summarized in Table A.3.

For each cross-validation fold, the temperature-rise normalization parameters were calculated using the active-domain pixels of the training samples only. The resulting statistics were then applied unchanged to the validation and held-out test data. Each trained model stored its configuration and normalization parameters in a corresponding `config.json` file.

A.4 Evaluation Configuration

The leave-one-condition-out evaluation used the 25%, 50%, 75%, and 100% trajectories. In each fold, one complete trajectory was held out for testing. The remaining conditions were divided into training and validation samples, and the held-out condition was evaluated only after model selection.

The model used for the independent varying-load evaluation was trained on the 50%, 62.5%, 75%, 87.5%, and 100% constant-load trajectories. The 25% condition was excluded because it lies outside the load range traversed by the evaluated 100%–50%–75% sequence.

Table A.3: FNO architecture and training configuration.

Setting	Value
Input channels	9
Output channels	1
Hidden-channel width	32
Fourier layers	4
Retained Fourier modes	16×16
Trainable parameters	603,489
Loss function	Geometry-masked mean-squared error
Optimizer	AdamW
Initial learning rate	1×10^{-3}
Weight decay	1×10^{-4}
Batch size	8
Maximum epochs	100
Learning-rate scheduler	Cosine annealing
Gradient-norm clipping	1.0
Random seed	42

A.5 Execution Commands

The following commands document the principal workflow. File paths may need to be adapted to the local execution environment.

A.5.1 Dataset generation

```
python train/direct_ewma/make_direct_ewma_dataset.py \
    --data_dir data/raw \
    --output data/processed/fno_6cond_element_512_direct_ewma
       .h5 \
    --resolution 512 \
    --interp_method linear \
    --spans 3 10 30 \
    --heat_source_mode base \
    --ewma_alignment previous \
    --target_storage global_normalized \
    --require_all_conditions
```

A.5.2 Cross-validation training

The following example holds out the 25% condition:

```
python train/direct_ewma/train_direct_ewma.py \
    --scenario loco_val \
    --dataset data/processed/
       fno_6cond_element_512_direct_ewma.h5 \
    --output_dir results/direct_ewma_model/test_25 \
```

```
    --include_conditions 25,half,75,full \
    --test_condition 25 \
    --epochs 100 \
    --batch_size 8 \
    --device cuda:1
```

Equivalent runs were performed with the 50%, 75%, and 100% conditions held out.

A.5.3 Cross-validation evaluation

```
python train/direct_ewma/eval_direct_ewma_cv.py \
    --cv_root results/direct_ewma_model \
    --dataset data/processed/
       fno_6cond_element_512_direct_ewma.h5 \
    --model_file best_model.pth \
    --batch_size 8 \
    --device cuda:1
```

A.5.4 Final-model training

```
python train/direct_ewma/train_direct_ewma.py \
    --scenario final_all \
    --dataset data/processed/
       fno_6cond_element_512_direct_ewma.h5 \
    --output_dir results/direct_ewma_model/final_all \
    --include_conditions half,75,full,62.5,87.5 \
    --epochs 100 \
    --batch_size 8 \
    --device cuda:1
```

A.5.5 Independent varying-load evaluation

```
python train/direct_ewma/infer_direct_ewma_varying.py \
    --model_dir results/direct_ewma_model/final_all \
    --dataset data/processed/
       fno_6cond_element_512_direct_ewma.h5 \
    --val_csv data/raw/
       NodeTemperature_100_50_75_correct_sampling_time.csv \
    --coord_file data/raw/coordinate_exp_test.txt \
    --output_dir results/direct_ewma_model/final_all/
       visualization \
    --device cuda:1 \
    --t_end 60000
```

The device identifier `cuda:1` records the GPU index used in the available environment and should be changed when reproducing the workflow on another system.

A.6 Runtime Measurement Scope

The reported full-trajectory FNO inference time includes feature construction, model forward passes, output denormalization, and extraction of the temperature summaries for all 276 requested time points. It excludes model loading, raw FEM-data loading, FEM-field interpolation, metric calculation, plotting, and file output.

The FEM time was measured over five complete runs of the varying-load trajectory. Because the FEM and neural-network calculations were not performed under a hardware-normalized setup, the reported speedup applies only to the measured execution environments.

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY