World Electric Vehicle Journal

MDPI

Article

Optimal Design of Interior Permanent Magnet Synchronous Motor Considering Various Sources of Uncertainty

Giacomo Guidotti, Dario Barri , Federico Soresini and Massimiliano Gobbi *

Department of Mechanical Engineering, Politecnico di Milano, Milan 20156, Italy; giacomo.guidotti@polimi.it (G.G.); dario.barri@polimi.it (D.B.)
* Correspondence: massimiliano.gobbi@polimi.it

Abstract: The automotive industry is experiencing a period of transition from traditional internal combustion engine (ICE) vehicles to electric vehicles. Although electric machines have always been used in many applications, they are generally designed neglecting the sources of uncertainty, even such uncertainty can lead to significant deterioration of the motor performance. The aim of this paper is to compare the results obtained from the multi-objective optimization of an interior permanent magnet synchronous motor (IPMSM) using a robust approach versus a deterministic one. Unlike other studies in the literature, this research simultaneously considers different sources of uncertainty, such as geometric parameters, magnet properties, and operating temperature, to assess the variability of electric motor performance. Different designs of a 48 slot–8 pole motor are simulated with finite element analysis, then the outputs are used to train artificial neural networks that are employed to find the optimal design with different approaches. The method incorporates an innovative use of the neural network-based variance estimation (NNVE) technique to efficiently calculate the standard deviation of the objective functions. Finally, the results of the robust optimization are compared with those of the deterministic optimization. Due to the small margin of improvement in robustness, both methods lead to similar results.

Keywords: multi-objective optimization; robust optimization; electric motor; synchronous motor; IPMSM

Academic Editor: Michael Fowler

Received: 29 December 2024
Revised: 28 January 2025
Accepted: 2 February 2025
Published: 5 February 2025

Citation: Guidotti, G.; Barri, D.; Soresini, F.; Gobbi, M. Optimal Design of Interior Permanent Magnet Synchronous Motor Considering Various Sources of Uncertainty. *World Electr. Veh. J.* **2025**, *16*, 79. https://doi.org/10.3390/wevj16020079

1. Introduction

Nowadays, global warming has become an urgent and undeniable problem. The European government has set a target of reducing greenhouse gas emissions to 80% of 2008 emissions by 2030 [1]. This drastic change will influence many different aspects of society, among which is mobility. The transition from traditional internal combustion vehicles to electric vehicles could be a promising solution to this problem [2]. Electric motors are a key component of electric vehicles, and among the various motor configurations, permanent magnet motors show excellent performance characteristics [3]. The design of these highly efficient traction motors has thus become a hot topic, increasingly driven by optimization processes.

Optimization is a complex decision-making process where different objectives have to be taken into account simultaneously. In fact, the goals are often in conflict. The solution to this type of problem is generally a compromise achieved through a multi-objective programming method within a deterministic framework [4]. After the definition of design variables, that uniquely define a design, and of the objective functions, that quantify the quality of a solution, multi-objective programming allows the designer to select a compromise among a set of optimal solutions [5].

The design of electric motors is a well-known application of multi-objective programming and many publications about this topic can be found [4,6–12]. For instance, Gupta et al. [13] applied this method to a conventional induction motor and managed to simultaneously increase the torque and efficiency and to reduce the mass. With a similar approach, Ajie et al. [14] succeeded in enhancing the efficiency, torque, and power of a switched reluctance motor. Sardar et al. [15] optimized five objective functions; they managed to improve, at the same time, the efficiency, torque, torque ripple, magnet mass, and power factor of an interior permanent magnet synchronous motor.

Even if deterministic multi-objective programming quickly finds optimal designs, there could be significant differences between the expected and actual performance due to a variety of causes, such as manufacturing tolerances, aging, variability of operating conditions, and material properties. For this reason, optimal engineering design problems should be solved within a robust design framework to ensure product quality as well as confidence in product reliability [16]. Today, more and more papers about applications of this approach in various fields can be found. For instance, Rathod et al. [17] compare different robust optimization methods used in the context of the structural optimization of a cantilever beam with uncertain dimensions, material properties, and stresses. Another common application of robust design is the design of active suspension controls. For example, thanks to a controller obtained with this approach, Wang et al. [18] managed to improve comfort and to mitigate the influence of the variability of tires and suspension properties on the performance.

1.1. Literature Review

In the literature, many publications concerning the robust optimization of electric motors can be found [19,20]. However, they generally focus on single-objective optimizations or they consider only one source of uncertainty. For instance, Lee et al. [21] considered the impact of the uncertainty of the magnet properties and of the geometric tolerances on the back-EMF. This allowed researchers to increase the mean and to reduce the standard deviation of the back-EMF by means of a Taguchi optimization of the dimensions of the motor. The same sources of uncertainty were also considered by Cheng et al. [22], who, however, optimized the torque and the losses. In their research, robust optimization was able to significantly reduce the sensitivity of the performance of the motor to the skew angle and to the remanence of the permanent magnets. Unlike the majority of studies, which consider a uniform distribution of uncertainties, Y. Yang et al. investigated the effect of non-uniform allocation [23]. They show that considering a uniform distribution of uncertainties leads to an underestimation of the cogging torque, which can be, in the presence of non-uniform uncertainties, even more than twice as high as estimated in the uniform case. Taran et al. [24] investigated the influence of the magnet material on the robustness of the design by comparing the sensitivity of the objective functions to variations in design variables when two different magnet materials are used. The outcome was a much lower sensitivity to uncertainties for the designs with a stronger magnet grade. This is a consequence of the presence of saturated regions in the rotor, whose magnetic flux is not significantly affected by the rotor dimensions. J. Mun et al. [25] also considered the uncertainty in the operating temperature, which influences the properties of the materials. By considering its variability, they managed to decrease by 48% the probability of constraint violation on the cogging torque. A. Aggarwal et al. [26] also took into account the presence of eccentricity in the rotor. They demonstrated that variations in the design can significantly reduce the torque ripple and the vibrations induced by the eccentricity. However, this was only possible by decreasing the torque.

1.2. Contribution

This research, unlike others in the literature, studies a complex robust optimization problem applied to a traction electric motor for automotive applications. Firstly, the problem has multiple objectives and takes into account multiple sources of uncertainty, such as geometric parameters, magnet properties, and operating temperatures. Secondly, the presented method introduces novel and efficient estimations of the standard deviations of the objective and constraint functions using a neural network-based variance estimation (NNVE) approach [27]. Thanks to neural networks, the expected value and the standard deviations of the objective functions are evaluated at the same time. Table 1 highlights the novelty of this publication, showing how it utilizes at the same time methods and sources of uncertainty that have never been adopted simultaneously in this context.

Table 1. Comparison of the contents of this paper with other publications in the literature. An 'X' is present in the boxes of the table when the topic is addressed in the cited publication.

Article	Electric Motor	Manufacturing Tolerances	Temperature Variation	Magnetic Material Variability	Multi-Objective Optimization	NNVE
[26]	X	X				
[25]	X	X	X		X	
[24]	X	X		X		
[21]	X	X		X		
[20]	X	X			X	
[22]	X	X		X	X	
[23]	X	X				
[27]					X	X
This article	X	X	X	X	X	X

The aim of this study is to explore the advantages of robust optimization for the neural network-based multi-objective optimization of an interior permanent magnet synchronous motor (IPMSM) in the presence of different sources of uncertainty.

In Section 2, the proposed method is described. Then, a case study is presented. In Section 4, the results are shown and discussed. In Section 5, final considerations are provided.

2. Method

When addressing design optimization, it is essential to account for uncertainties arising from various sources. In this research, the design of an electric motor was optimized using a robust method that explicitly considers these uncertainties, as opposed to the classical deterministic approach, which entirely disregards them. In the robust optimization framework, uncertainties are embedded within the design variables and subsequently propagated to the objective and constraint functions. This allows for the identification of solutions that are less sensitive to variations and ensure reliable performance under real-world operating conditions.

In this section, the proposed method for conducting robust optimization is presented. It is based on the following steps:

- Definition of a design of experiments (DoE);
- Thermal and electromagnetic MotorCAD finite element analysis simulations of operating points ;

- Selection of a tolerance range for each input, with each assumed to be normally distributed and independent from the others;
- Estimation of the standard deviation of each input from the tolerance range;
- Feedforward neural network training with the outputs of the simulations;
- Estimation of the expected values, the standard deviation of the objective, and constraint functions with the neural network;
- Multi-objective constrained optimizations with a genetic algorithm.

The method is illustrated in Figure 1.

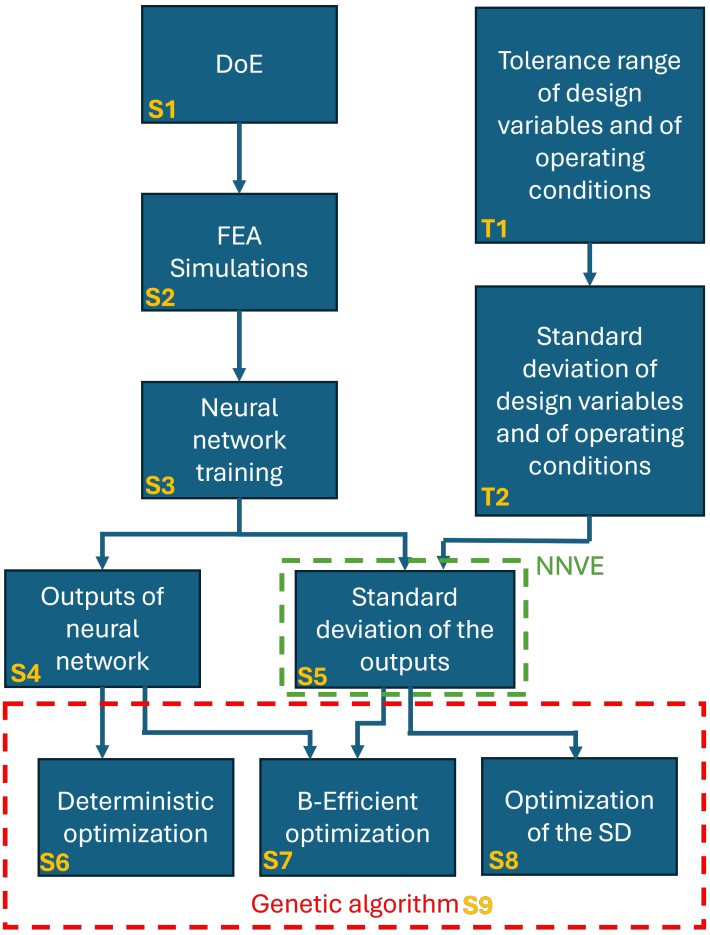

Figure 1. Flow chart of the proposed method.

2.1. DoE

The design of experiments (DoE), S1 in Figure 1, is a method to explore the relationship between design variables and outputs. The idea is to identify the main sources of variation and the main interactions among design variables [28]. The one used in this research is a full-factorial DoE. It is based on the selection of a series of levels for each design variable and then on the simulation of all the possible combinations of levels. This makes it suitable for this project as it accurately represents all the interactions between design variables. The main limitation of this method is the large size of the experiments even if only a few levels per design variable are considered [28]. The experiment plan is defined in the ANSYS MotorCAD v 15.1.2 software.

2.2. Simulation

The DoE is simulated using the ANSYS MotorCAD v15.1.2 [29] electromagnetic and thermal modules; S2 in Figure 1. ANSYS MotorCAD is an integrated multiphysics analysis

software for electric motor design. The temperature is estimated by means of the MotorCAD concentrated parameters model [30]. The estimated on-load losses are used to calculate the thermal performance of the motor. In particular, a convergence analysis between electromagnetic and thermal solutions is performed. This leads to an improvement in simulation accuracy as losses are evaluated in steady-state conditions.

2.3. Neural Network

The data obtained with the thermal and electromagnetic simulations are used to train neural networks, which are able to quickly estimate the expected value and standard deviations of objective and constraint functions, S3–S5 in Figure 1.

A neural network (NN) is a general purpose non-linear model that can be used when there is poor knowledge about the relationship between inputs and outputs. Neural networks are used in several fields, including adaptive control, predictive modeling, and artificial intelligence [31–33]. They are an abstract simulation of a nervous system, in which many neurons communicate with each other thanks to a network of connections. Each neuron receives inputs from other neurons or external sources and computes an output signal. The way each neuron produces an output is shown in Figure 2. A weighted sum is calculated from the inputs and then a bias is added. The result of this sum is the input for the activation function. This expression can be written as

$$x_o = \varphi((\sum_{i=1}^{n} w_{io} \cdot x_i) + b_o) \quad (1)$$

where x_o is the output of the neuron, x_i is an input, w_{io} is the weight associated with the input x_i, b_o is the bias of the neuron, and φ is the transfer function of the neuron. The weights and biases are computed during the training process in order to fit the data samples, S3 in Figure 1, while the activation function is defined according to the problem studied [28].

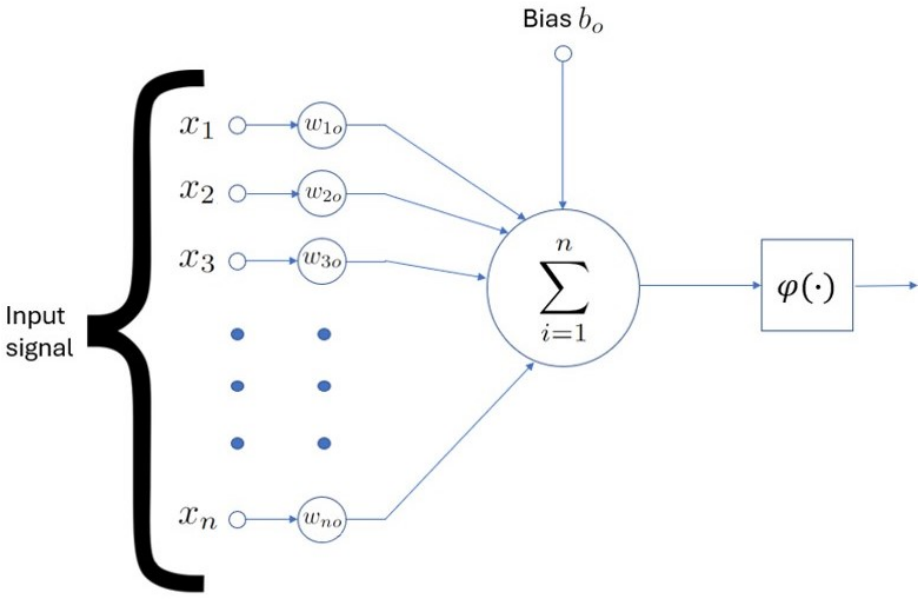

Figure 2. Structure of a neuron in a neural network.

As shown in Figure 3, a neural network is generally made up of several layers of neurons, represented by circles. The first layer is the so-called input layer because it receives the external information directly. The last layer is called the output layer since its output is the result produced by the whole artificial neural network (ANN). The layers placed

between the input and output layers are called hidden layers. Each layer receives as input the output of the previous layer and there are no connections skipping one layer.

The number of layers and the number of neurons belonging to each layer have to be chosen according to the complexity of the problem. An optimal compromise between neural network complexity and precision has to be found. In this study, in particular, a single-hidden-layer structure has been chosen.

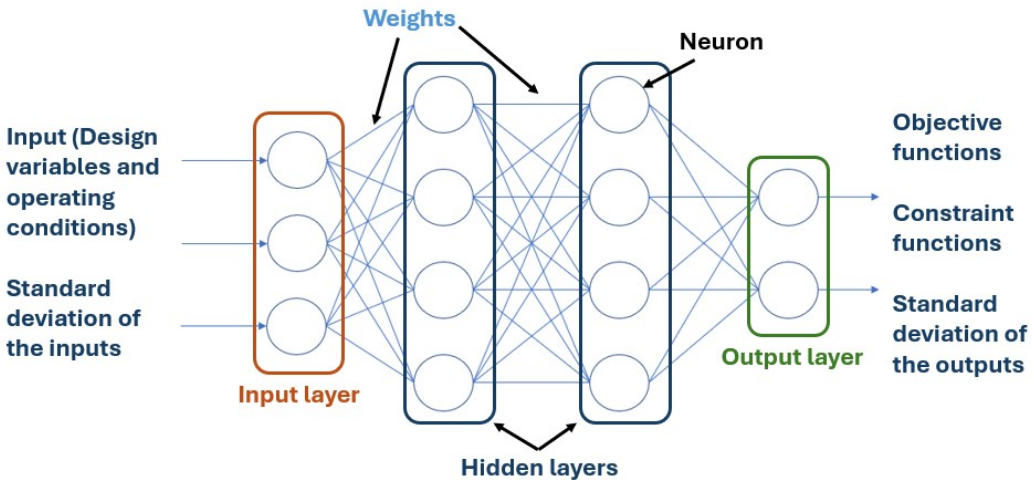

Figure 3. Structure of the feedforward artificial neural networks used in this paper. They receive as input the design variables, the operating conditions, and their standard deviation. The output is constituted by the average value and standard deviation of objective and constraint functions.

As in other studies [34,35], feedforward neural networks are used to estimate the performance indexes and constraint functions from the design variables and operating conditions; S4 in Figure 1. Unlike the previously mentioned papers, in this article, neural networks were also used to estimate the standard deviation of both objective functions and constraint functions [27]; S5 in Figure 1. If the uncertainties are assumed to be independent, normally distributed, non-correlated, and symmetric, for a first-order Taylor series of a generic function $f(x)$, the variance of the output of $f(x)$ can be computed as

$$Var(f(x)) = \sum_{i=1}^{N} \left(\frac{\partial f}{\partial x_i} \right)^2 \sigma_i^2 \tag{2}$$

where $Var(f(x))$ is the variance of the output of the function $f(x)$ for a given point x, N is the number of inputs, x_i is the ith input, and σ_i^2 is the variance of the ith input [36,37]. Thus, to estimate the variance of the outputs, both the partial derivatives of the outputs with respect to the inputs and the variance of the inputs are needed.

The output of a single-layer neural network can be expressed as

$$y = \sum_{k=1}^{m} w_k \varphi_k \left(\left(\sum_{i=1}^{n} w_{io} \cdot x_i \right) + b_o \right) \tag{3}$$

where y is the output of the neural network, m is the number of neurons, w_k is the output weight of each neuron, φ_k is the activation function of each neuron, and x_i is the ith input. So, the derivative of the output of a single-layer neural network with respect to each input x_i can be expressed as

$$\frac{\partial y}{\partial x_i} = \sum_{k=1}^{m} w_k \varphi_k' \left(\left(\sum_{i=1}^{n} w_{io} \cdot x_i \right) + b_o \right) w_{io} \tag{4}$$

where φ' is the derivative of the activation function, which has a unique explicit expression. Since the output and input weights are fixed quantities for a given trained neural network, it is sufficient to evaluate the partial derivatives of the activation functions φ'_k to estimate the partial derivatives of the neural network. These partial derivatives are then used in Equation (2) to estimate the variance of the outputs of the neural network. In addition to the value of the partial derivatives, to compute the variance of the outputs with Equation (2), the value of the variance of the inputs is needed [38,39]. In this study, the standard deviations of the inputs are considered to be equal to a sixth of the extension of the tolerance range; T1 and T2 in Figure 1.

This approach can be referred as neural network-based variance estimation (NNVE). The method relies on the following hypothesis: the distributions of the uncertainties are assumed to be Gaussian and independent. Given these assumptions, NNVE allows for quick estimates of the variance of all the outputs of the neural network when the function tends to behave locally in a linear manner and when there are low standard deviations of the inputs [36]. Since, as shown in paragraph 4.2, these conditions are fulfilled, NNVE is used to estimate the standard deviation.

2.4. Formulation of the Optimization Problem

When carrying out optimization, one generally focuses more on maximizing performance, without taking into account the importance of robustness. This could lead to the determination of optimal solutions that are excessively sensitive to even slight variations in design variables or external conditions. Thus, the actual performance can be significantly worse than expected. Robust optimization is a term used to indicate many optimization methods able to find solutions that perform well even in presence of uncertainties. The aim of robust optimization is to find a trade-off between performance and its variability due to uncertainties [16]. To better explain this concept, let us consider the example in Figure 4. In this image, it is possible to see that the best solution in terms of f(x) is the green one. However, if x is even slightly different from the theoretical value, the value of f(x) can deteriorate significantly. For this reason, if robustness is important, it could be convenient to opt for the light blue solution to achieve a higher robustness in exchange for a slightly worse performance.

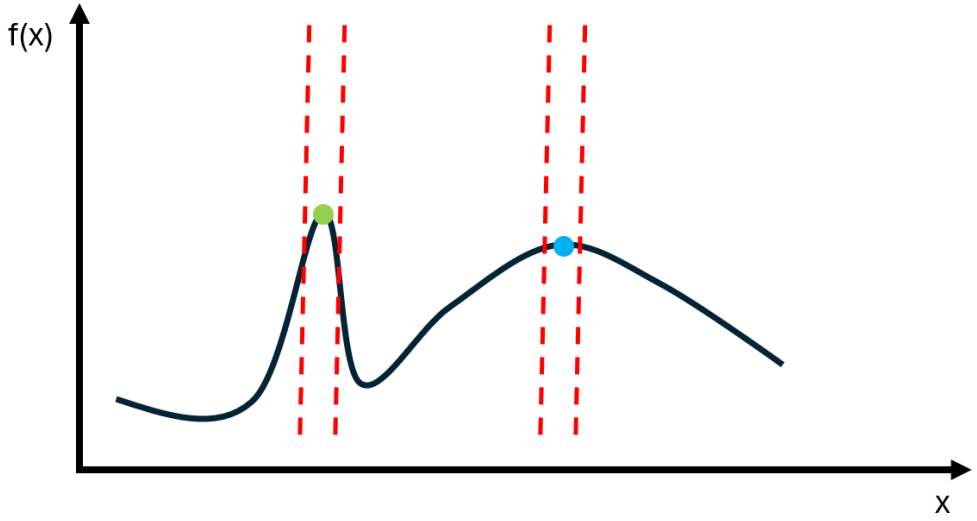

Figure 4. Example of the optimisation of an objective function f(x). The black line represents a generic function f(x) to be optimised, the green and blue points are two local maxima and the dashed red lines are used to highlight the uncertainty interval of the green and blue points.

In the following part, the optimization methods that are compared to evaluate the impact of a robust approach are presented.

2.4.1. Multi-Objective Deterministic Optimization

The multi-objective deterministic optimization problem involves solving several objective functions simultaneously, assuming ideal conditions without taking uncertainties into account; S6 in Figure 1. It is formulated as

$$\min \quad \boldsymbol{F}(\boldsymbol{d}) = \begin{Bmatrix} f_1(\boldsymbol{d}) \\ f_2(\boldsymbol{d}) \\ \vdots \\ f_m(\boldsymbol{d}) \end{Bmatrix}$$

$$\text{s.t.} \quad \begin{cases} \boldsymbol{g}(\boldsymbol{d}) = \begin{Bmatrix} g_1(\boldsymbol{d}) \\ g_2(\boldsymbol{d}) \\ \vdots \\ g_n(\boldsymbol{d}) \end{Bmatrix} \leq 0 \\ \\ \boldsymbol{x_l} \leq \boldsymbol{d} \leq \boldsymbol{x_u} \end{cases} \tag{5}$$

where $\boldsymbol{d}$ is the vector that contains the expected values of the design variables, $\boldsymbol{F}(\boldsymbol{d})$ is the vector that contains the expected value of the objective functions $f_i(\boldsymbol{d})$, g($\boldsymbol{d}$) is the vector that contains the expected value of constraint functions $g_i(\boldsymbol{d})$, $\boldsymbol{x_l}$ and $\boldsymbol{x_u}$ are, respectively, the upper and lower bounds of the design variables, m is the number of objective functions, and n is the number of constraint functions [28].

2.4.2. β-Efficient Formulation

The robust formulation used in this study is the β-efficient problem with a normal distribution ($k\alpha$ formulation) [40]; S7 in Figure 1. It can be written as

Given the probabilities $\beta_1, \beta_2, \ldots, \beta_k$

$$\min \begin{Bmatrix} f_1(\boldsymbol{d}) + \alpha_{f_1}\sigma_{f_1}(\boldsymbol{d}) \\ f_2(\boldsymbol{d}) + \alpha_{f_2}\sigma_{f_2}(\boldsymbol{d}) \\ \vdots \\ f_m(\boldsymbol{d}) + \alpha_{f_m}\sigma_{f_m}(\boldsymbol{d}) \end{Bmatrix}$$

$$\text{s.t.} \quad \begin{cases} \begin{Bmatrix} g_1(\boldsymbol{d}) + \alpha_{g_1}\sigma_{g_1}(\boldsymbol{d}) \\ g_2(\boldsymbol{d}) + \alpha_{g_2}\sigma_{g_2}(\boldsymbol{d}) \\ \vdots \\ g_n(\boldsymbol{d}) + \alpha_{g_n}\sigma_{g_n}(\boldsymbol{d}) \end{Bmatrix} \leq 0 \\ \\ \boldsymbol{x_l} \leq \boldsymbol{d} \pm \alpha_d\boldsymbol{\sigma_d} \leq \boldsymbol{x_u} \\ \alpha_{f_i} = \Phi^{-1}(\beta_{f_i}) \text{ for i = 1, 2,...,m} \\ \alpha_{g_i} = \Phi^{-1}(\beta_{g_i}) \text{ for i = 1, 2,...,n} \end{cases} \tag{6}$$

where σ_{f_i} is the standard deviation of the function f_i, $\Phi^{-1}(\cdot)$ is the inverse of the standard normal distribution, β_{g_i} is the probability that the final product violates the constraint $g_i(d)$, and β_{f_i} is the probability that the objective function f_i of the final product is worse than $f_i(d) + \alpha_{f_i}\sigma_{f_i}(d)$. The solutions to these problems are a compromise between performance and robustness. The higher α, the more the trade-off sacrifices performance to maximize the robustness. In the objective function, the sign of the standard deviation is chosen so that a higher standard deviation value leads to a worsening of the objective function. For instance, in the case of a maximization problem of a function f_{max}, its β-efficient formulation can be written as

$$\begin{aligned} &\min(-f_{\max}(\boldsymbol{d}) + \alpha_{f_{\max}}\sigma_{f_{\max}}(\boldsymbol{d})) \\ &\alpha_{f_{\max}} = \Phi^{-1}(\beta_{f_{\max}}). \end{aligned} \tag{7}$$

It should be noted that the standard deviation is not only present inside the objective functions but also inside the constraint functions. Thus, a higher margin of safety is kept with respect to the violation of the constraints. In this paper, the same value of α, which is in one case equal to 2 (β = 95.45%) and in the other to 3 (β = 99.73%), is used for each objective and constraint function.

2.4.3. Multi-Objective Minimization of the Standard Deviation (MOMSD)

In addition to the optimizations presented in the previous paragraphs, a multi-objective minimization of the standard deviation (MOMSD) of the objective functions with deterministic constraints is conducted; S8 in Figure 1. It is formulated as

$$\begin{aligned} &\min \begin{Bmatrix} \sigma_{f_1}(\boldsymbol{d}) \\ \sigma_{f_2}(\boldsymbol{d}) \\ \vdots \\ \sigma_{f_m}(\boldsymbol{d}) \end{Bmatrix} \\ &\text{s.t.} \begin{cases} \begin{Bmatrix} g_1(\boldsymbol{d}) \\ g_2(\boldsymbol{d}) \\ \vdots \\ g_n(\boldsymbol{d}) \end{Bmatrix} \leq 0 \\ \boldsymbol{x_l} \leq \boldsymbol{d} \leq \boldsymbol{x_u}. \end{cases} \end{aligned} \tag{8}$$

By comparing the results of the MOMSD to the deterministic ones, it is possible to estimate the margin of improvement in robustness of the deterministic solutions.

2.5. Determination of the Optimal Solutions

The optimization problem is solved using a non-dominated sorting genetic algorithm [41,42], S9 in Figure 1, specifically designed and implemented by the authors for this research. The genetic algorithm was chosen for its proven efficiency and effectiveness in solving multi-objective optimization problems [7,42]. Its standout features include computational speed, achieved through an improved sorting mechanism, and its ability to compute a set of non-dominated solutions. In the optimization algorithm, the designs are represented by individuals which belong to a population, which is constituted by a group of designs. Individuals can reproduce with each other by generating a child, which is a design obtained by combining the parents' designs. Then, each child could be subject to mutation, which is represented

by a random change in the design. The fitness is assigned according to the dominance depth. The number of individuals is set equal to 10^4; 10 bits per design variable and a mutation probability of 0.5% are used. The algorithm is stopped when all the individuals are non-dominated or after 1000 generations. This configuration of parameters represents an effective balance between the quality of the results and the computation time.

The steps that constitute the developed non-dominated sorting genetic algorithm are reported in Algorithm 1.

Algorithm 1 Non-dominated sorting genetic algorithm

Step 1: Initialize Population
Generate a random population of N individuals i, each represented by a set of decision variables.
Step 2: Evaluate Objective Functions
For each individual i, evaluate all objective functions $f_1(i), f_2(i), f_3(i), \ldots, f_M(i)$, where M is the number of objectives.
Step 3: Assign Fitness (Dominance Depth)
Non-dominated Sorting:
- Divide the population into fronts based on dominance relationships.
- Individuals in the first front have a dominance depth of 1 (non-dominated by any others), those in the second front have a dominance depth of 2, and so on.
- $Fitness = \frac{1}{DominanceDepth}$
if Constraints are not satisfied for an individual i **then**
Penalize the fitness of i (set $Fitness = 10^{-6}$).
end if
while Stopping criteria not satisfied (max generations reached or all individuals have a fitness equal to 1) **do**
Step 4: Select Parents
Select the parents according to their fitness
Step 5: Recombine Genes (Crossover)
Apply crossover operations, where parents are paired to produce offspring using the roulette wheel selection technique. This step allows exploration of new regions in the design variable space.
Step 6: Mutation
Apply mutation operations to the offspring to introduce small random changes. This allows to maintain genetic diversity.
Step 7: Evaluate New Candidates
Evaluate the objective functions for each offspring individual.
Step 8: Select the Next Generation
Perform non-dominated sorting on the combined population (current population + offspring).
Assign Fitness as in step 3
Select the best N individuals for the next generation according to their fitness
end while

3. Case Study

The described method is applied in the context of the optimization of an electric motor design obtained from 2010 Toyota Prius motor characteristics with the approach described in [43]. It is a V-shaped IPMSM; its cross-section is shown in Figure 5. This motor features 48 slots and 8 poles. Its magnets are sintered neodymium–iron–boron (NdFeB) magnets made of N48SH, while the material of the stator and rotor laminations is M235-35A electrical steel. The slots are parallel and with distributed winding. Motor performances are evaluated at 2900 rpm, with a current of 134 Arms and a phase advance angle of 38.4°. The DC bus voltage is set to 650 V. The cooling system is based on a housing with a spiral water jacket. The motor is assumed to be mounted in the horizontal direction and to have

no endcap cooling. The flow of the coolant goes from the front to the rear in a single flow path. The volumetric flow rate of the fluid is 7 L/min and its inlet temperature is set to 65 °C. The selected coolant is water. The initial design of this motor has a rated power, a torque, an efficiency, and a torque ripple which are, respectively, equal to 60 kW, 205 Nm, 96.2%, and 59 Nm.

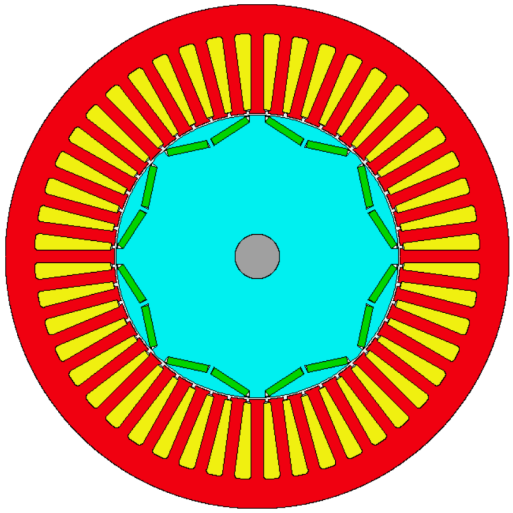

Figure 5. Cross-section of the motor used in the case study. White, green, red, yellow, grey and light blue are respectively the colors associated with air, magnets, stator material, slots, shaft and rotor material.

3.1. Design Variables

The design variables are mainly geometric dimensions, as highlighted in Figure 6. Additional design variables are the number of turns and strands in hand. The nominal value, the tolerance range (T1 in Figure 1), and the maximum and minimum tested values of all these quantities are collected in Table 2. As can be seen from the table, the range of values tested is larger than the range of uncertainty as it is necessary to simulate the entire design domain with MotorCAD in order to represent it correctly with the neural networks.

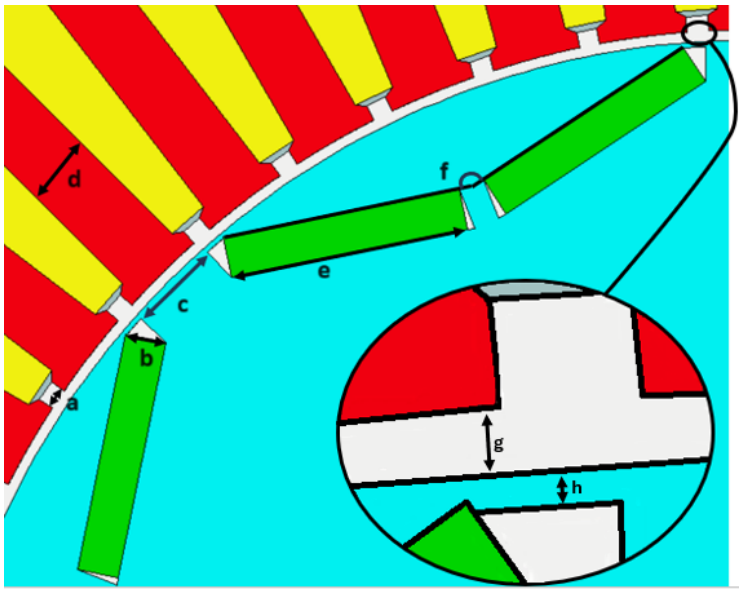

Figure 6. Radial section of the motor in which the following design variables are highlighted: (**a**) slot opening, (**b**) magnet thickness, (**c**) web thickness, (**d**) tooth width, (**e**) magnet width, (**f**) pole V angle, (**g**) airgap, (**h**) bridge thickness. White, green, red, yellow, light blue and grey are respectively the colors associated with air, magnets, stator material, slots, rotor material and slot opening.

Table 2. Design variables: Reference value, tolerance, and minimum and maximum values. The symbol "/" is used to indicate the absence of a tolerance range for the number of turns and strands in hand.

Design Variable	Reference	Tolerance	Min	Max
Airgap	1 mm	±0.025 mm	0.7 mm	1.3 mm
Slot opening	2 mm	±0.06 mm	1.4 mm	2.6 mm
Magnet thickness	4 mm	±0.1 mm	2.8 mm	5.2 mm
Magnet width	23 mm	±0.1 mm	16.1 mm	23 mm
Tooth width	6.76 mm	±0.1 mm	4.73 mm	8.79 mm
Bridge thickness	0.5 mm	±0.025 mm	0.35 mm	0.65 mm
Web thickness	9 mm	±0.1 mm	6.3 mm	11.7 mm
Pole V angle	158°	±0.05°	130°	160°
Number of turns	29	/	20	36
Number of strands in hand	9	/	5	13

3.2. Additional Sources of Uncertainty

In this research, it is not only the uncertainties caused by the manufacturing and geometric tolerances that are considered. During operation, the electric motor can experience different thermal conditions that can affect performance. Therefore, the variability in the operating temperature and magnet properties are also taken into account; T1 and T2 in Figure 1. These quantities are out of the control of the designer and they only contribute to the total uncertainty of the objective and constraint functions. Their nominal value, confidence range, and maximum and minimum values are shown in Table 3. These quantities are tested only inside their range of uncertainty because they do not vary inside a design domain. The confidence range for the residual flux density B_r is similar to [21]. On the other hand, as far as the temperature is concerned, the majority of publications focus on small fluctuations around the steady-state point [25,44]. However, the goal of this research is to investigate a much larger range of temperatures in order to consider both cold starts and prolonged full-load runs. In particular, the extension of the tolerance range is chosen since in many studies [45,46] comparable ranges of temperature are obtained during thermal transients. In the electromagnetic simulations, the magnet temperature $T_{magnets}$ and winding temperature $T_{windings}$ are assumed to be equal to reduce the number of variables and the computational effort.

Table 3. Reference values, and maximum and minimum values tested of the additional sources of uncertainty.

Dimension	Reference	Variation Range	Min	Max
$T_{magnets}$	120 °C	±30 °C	90 °C	150 °C
$T_{windings}$	120 °C	±30 °C	90 °C	150 °C
B_r	1.37 T	±0.01 T	1.36 T	1.38 T

3.3. Constraints

Almost every design variable can vary between 70% and 130% of its nominal value. The only exceptions are the width of the magnet and the pole V angle, since wider intervals of variability would lead to unfeasible or poorly performing designs. This is why reduced intervals are employed for these design variables. The number of turns and strands in hand

can freely vary in the design domain between 20 and 36 and between 5 and 13, respectively. Even in this design domain, many configurations are unfeasible and therefore the feasibility of the design is checked before the optimization. Additional constraints concern the voltage, which is limited by the battery, and the maximum operating temperatures of magnets and windings. The limit values for all these quantities are listed in Table 4.

Table 4. Constraints' limit values.

Parameter	Limit Value
$T_{magnets,max}$	150 °C
$T_{windings,max}$	180 °C
V_{max}	650 V

3.4. Performance Indexes

The ideal electric motor should feature the lowest mass, the highest efficiency, and the highest output torque without too-wide oscillations. In this research, the mass is neglected while the average torque and the efficiency have been chosen as objective functions and have to be maximized. The third objective function is the torque ripple, which is defined as the difference between maximum torque and minimum torque during a complete revolution of the motor. Torque ripple is a consequence of cogging torque, non-ideal back-EMF, and saturation phenomena. It has to be reduced in order to decrease the vibrations, to improve the mechanical reliability and comfort, in particular in standing-start conditions [47,48].

3.5. Neural Networks

The performance indexes and some constraint functions are estimated by means of two neural networks obtained with the Matlab Deep Learning Toolbox [49]. The structures of the two neural networks were chosen after a sensitivity analysis of the number of neurons and of the number of hidden layers. Configurations with up to three layers and a maximum number of neurons per layer of 80 were tested. Specifically, the aim of the sensitivity analysis was to find an optimal balance between the accuracy and the complexity of the neural network. Moreover, other activation functions, such as ReLU, Tanh, and softplus, were tested but did not allow the distribution of performance indexes to be correctly represented, leading to non-physical values such as efficiencies above 100%.

The first neural network is trained with the data coming from the electromagnetic MotorCAD simulations and it is used to estimate the performance indexes and the voltage; S2–S4 in Figure 1. It is trained with a full-factorial design with five levels per variable; S1 in Figure 1. This is a very high number of samples but is necessary to correctly represent the behavior of the torque ripple, which varies significantly even for small variations in design variables.

The structure of this neural network is shown in Figure 7. It is based on a single hidden layer with 34 neurons, whose activation function is the log-sigmoid [50], which is given by

$$logsig(x_h) = \frac{1}{1+e^{-x_h}}. \tag{9}$$

Its derivative, which is used in Equation (10), is equal to

$$\frac{d\ logsig(x_h)}{dx_h} = logsig(x_h)(1 - logsig(x_h)). \tag{10}$$

The inputs are the 10 design variables listed in Table 2, the operating temperature and the residual magnetic flux. Whereas the outputs are torque, torque ripple, efficiency, and voltage.

A second neural network is trained with the simulations of the thermal module and is used to estimate the maximum operating temperatures of the magnets and of the windings; S4 in Figure 1. The motors whose temperature prediction exceeds the limit values are assigned a penalty during the optimization process with the genetic algorithm, so that the probability of survival is very low. The training set is given by a four-level full-factorial DoE; S1–S3 in Figure 1. A lower number of samples is used with respect to the other neural network because the thermal simulations perform a convergence analysis, which is highly time consuming. The ANN structure is shown in Figure 8. As can be noted, the framework is very similar to the previous neural network. However, after a sensitivity analysis, a number of hidden neurons equal to 65 was chosen. Moreover, the number of inputs is 10 since the temperature is no more an input and the impact of the residual flux density is neglected. The outputs are the peak temperature of the magnets and of the windings.

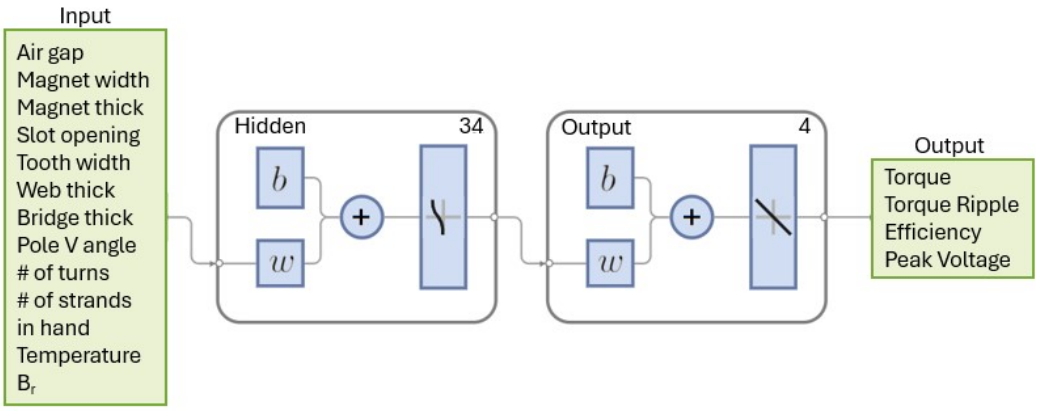

Figure 7. Structure of the neural network used to evaluate the performance indexes and the peak voltage.

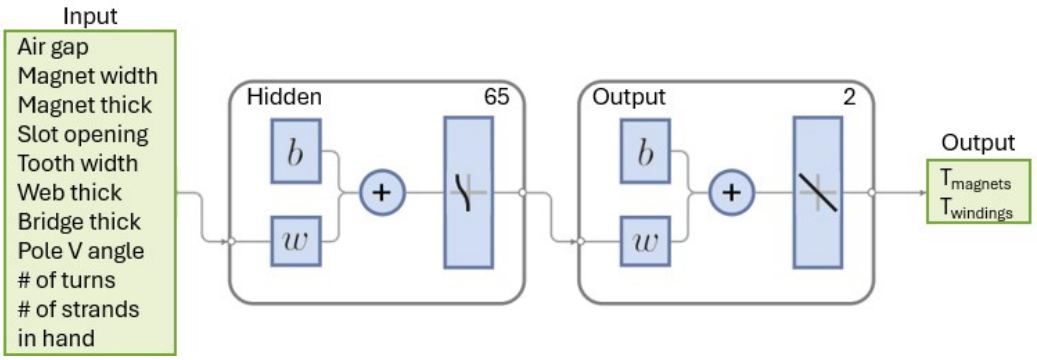

Figure 8. Structure of the neural network used to evaluate the peak temperatures of magnets and windings.

4. Results

In this section, the results for both deterministic and robust approaches are presented. Specifically, the parameters related to the neural network architecture and the genetic algorithm used to obtain these results are detailed in Table 5.

Table 5. Tunable parameters of the proposed method.

Parameter	Value	Parameter	Value
Number of levels per variable (DoE electromagnetic)	5	Number of levels per variable (DoE thermal)	4
Number of hidden layers (ANN electromagnetic)	1	Number of hidden layers (ANN thermal)	1
Number of neurons per hidden layer (ANN electromagnetic)	34	Number of neurons per hidden layer (ANN thermal)	65
Activation function (ANN electromagnetic)	Log-sigmoid	Activation function layer (ANN thermal)	Log-sigmoid
Population size (Genetic algorithm)	10^4	Bits per design variable (Genetic algorithm)	10
Mutation probability (Genetic algorithm)	0.5%	Max number of generations (Genetic algorithm)	1000

4.1. Accuracy of the Estimates of the Expected Values of Objective and Constraint Functions

To evaluate the accuracy of the neural networks, the correlation plots, that are represented in Figures 9 and 10, are implied. In these graphs, the outputs of the neural networks are compared to the ones of the simulations when the same input is used. The ideal behavior is a straight line with a slope equal to 1, since it would mean that the two outputs are exactly the same. In this case, both neural networks are quite close to the ideal response and the Pearson coefficient assumes values which are considered adequate for many applications [51,52]. For these reasons, the level of accuracy is assumed to be sufficient for this case study.

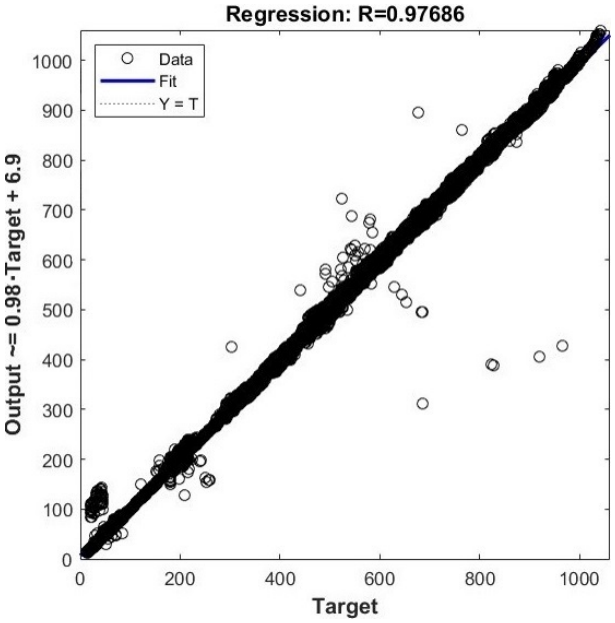

Figure 9. Correlation plot of the neural network trained with the mechanical simulations.

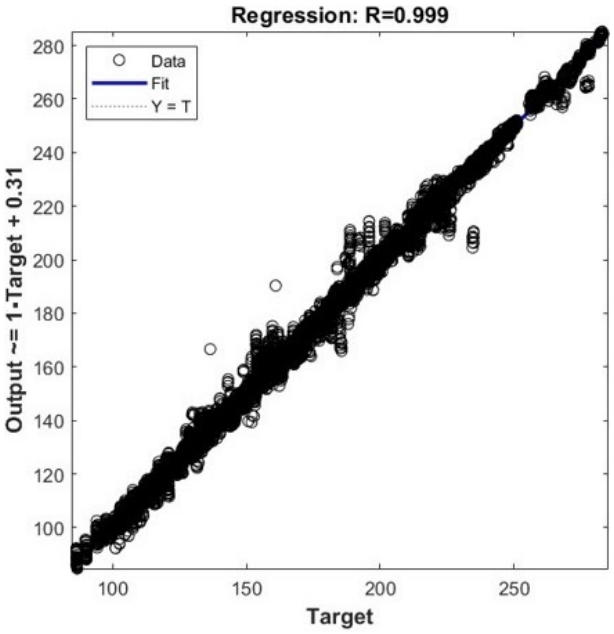

Figure 10. Correlation plot of the neural network trained with the thermal simulations.

4.2. Accuracy of the Estimates of the Standard Deviation of Objective and Constraint Functions

In this section, the output of NNVE, S5 in Figure 1, is compared to that of a Monte Carlo method, described in [53], to verify the accuracy and robustness of NNVE in this context. This comparison is essential because, as presented in Section 2.3, NNVE relies on several assumptions. To apply the Monte Carlo method in this study, the next steps are followed. Each source of uncertainty is assumed to have a Gaussian distribution with a standard deviation reported in Section 3. Then, 10^6 possible combinations of uncertainties are found from these distributions. After that, the design is tested with the neural network in the presence of each combination of uncertainties. Finally, the standard deviations of the objective and constraint functions are estimated from the distributions of the output.

The average absolute differences between the outcomes of the Monte Carlo method and of NNVE are computed as

$$Diff_i = \frac{1}{N}\sum_{i=1}^{N}\frac{|\sigma_{MCi} - \sigma_{NNVEi}|}{\sigma_{Ref}} \tag{11}$$

where σ_{MCi} and σ_{NNVEi} are the standard deviations estimated, respectively, with the Monte Carlo and NNVE methods and σ_{Ref} is the standard deviation of the reference design. The average absolute differences obtained simulating 10 different Monte Carlo experiment plans are collected in Table 6. NNVE gives almost the same results but with great savings in computational time. Actually, the comparison between the two methods was carried out considering different initial populations, as shown in Figure 11. The elapsed time for NNVE remains unchanged by increasing the number of combinations of uncertainties because it needs to evaluate only the partial derivatives of the expected values of the design variables. On the other hand, it increases for the Monte Carlo method because the neural network has to simulate more and more experiments, and the computational time to determine the standard deviation of the output also increases.

Table 6. Average absolute differences between Monte Carlo method and neural network-based method.

Average absolute differences between Monte Carlo method and NNVE			
Diff_{Torque}	0.02%	Diff_{Ripple}	0.08%
$\text{Diff}_{Efficiency}$	0.7%	$\text{Diff}_{Voltage}$	1%
$\text{Diff}_{T_{mag}}$	0.04%	$\text{Diff}_{T_{win}}$	0.06%

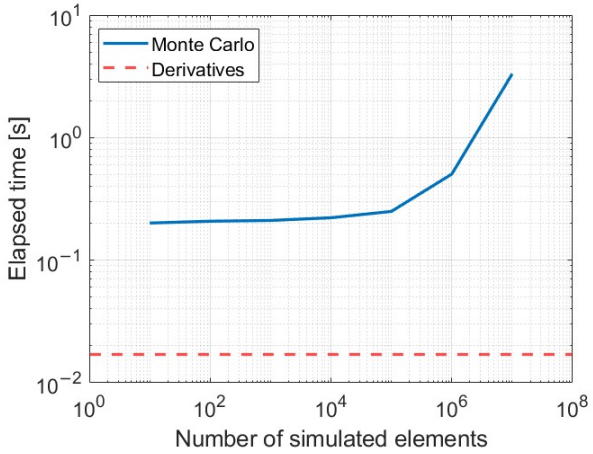

Figure 11. Comparison of the computational effort of the Monte Carlo method and the proposed approach.

4.3. Optimization

In this section, the results of the optimizations described in Section 3 are shown and discussed. As shown in Figure 12, the genetic algorithm, S9 in Figure 1, converges after a few generations in approximately one minute when a population of 10^4 individuals is considered.

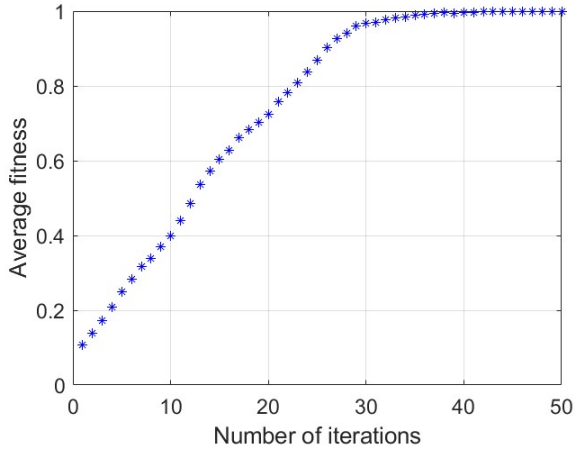

Figure 12. Average fitness of the population during the optimization using non-dominated sorting genetic algorithm with a population of 10^4 individuals.

In the following subsection, the outcome of the deterministic approach is compared with that of MOMSD; S6 and S8, respectively, in Figure 1. Then, the β-optimal designs (both β = 95% and β = 99%), S7 in Figure 1, are shown and compared to the deterministic ones.

4.3.1. Comparison Between MOMSD Solutions and Deterministic Optimal Solutions

To compare MOMSD and the deterministic results, at the end of the two optimizations, the standard deviations of the deterministic optimal designs are computed with NNVE

and the expected values of the MOMSD optimal designs are computed with the neural networks. The results of MOMSD are shown and compared to those of the deterministic approach in Figure 13. As can be seen, the performance indexes are all conflicting and the expected value of torque of the MOMSD optimal solutions tends to be much lower compared with the deterministic solutions. This is obvious since the expected values of the performance indexes are neglected during the optimization process, while their standard deviations are minimized. At the same time, as can be seen from the sizes of the crosses in the figures, the margin of improvement of the standard deviations of the performance indexes of the deterministic solutions is not very high if compared to those of MOMSD in terms of the expected value. Deterministic optimization therefore leads to quite low standard deviations, even though it does not consider any uncertainty.

To quantify these differences, the average values of the performance indexes of the optimal deterministic and MOMSD designs are collected in Figure 14. As can be seen, the deterministic optimal solutions are obviously better in terms of expected value. However, on average, MOMSD's optimal solutions are not so far away in terms of expected values of torque ripple and efficiency. Even if these differences are quite low, those in terms of standard deviations are even lower. This can be seen in the standard deviation of the efficiency, which is almost the same for the two approaches. On the other hand, the average standard deviations of torque ripple and torque differ by tenths of an Nm, which is much lower than the difference in their expected values.

In Figures 15 and 16, the utopia and nadir points [28] of the deterministic and MOMSD optimal designs are represented. As can be seen, the differences between the utopia and the nadir points of the MOMSD are lower. This means that the range of the performance indexes obtained with MOMSD is shorter. Even in this case, there are important differences between the expected values of torque and torque ripple for the two utopia points while their standard deviations are much closer. It is noteworthy that the ranges of the expected value and standard deviation of efficiency are very similar for both approaches.

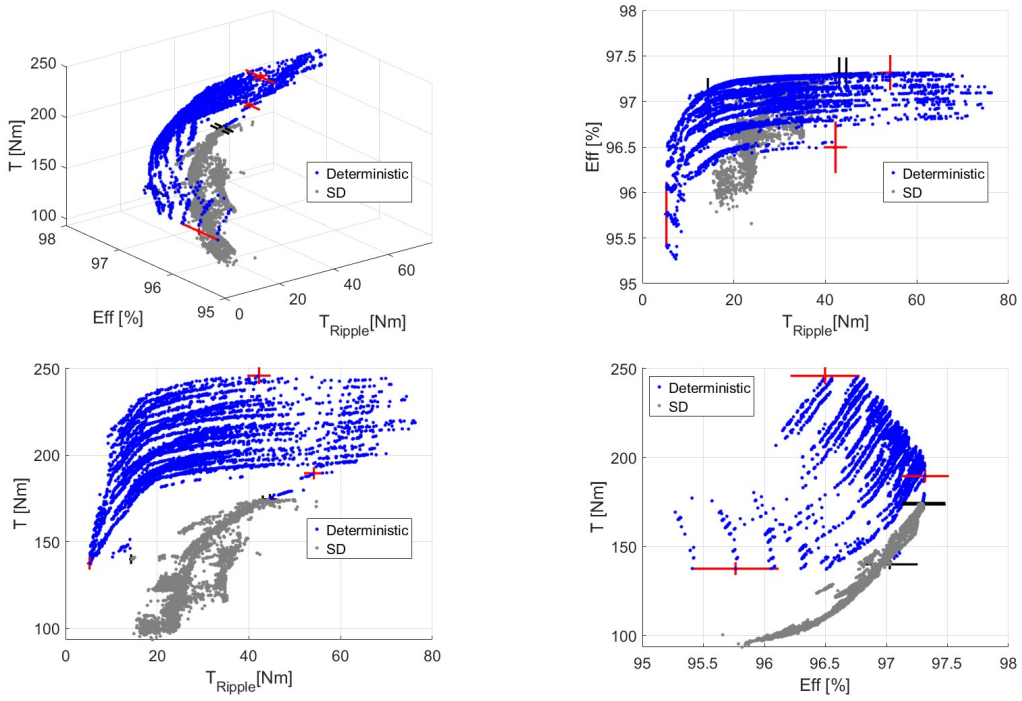

Figure 13. Comparison of the deterministic and MOMSD optimal solutions in terms of expected values. Red and black lines represent the uncertainty range for deterministic and MOMSD solutions, respectively. These lines are centered on three distinct points, each representing the best performance for torque, torque ripple, or efficiency, with an extension of 6σ in each direction.

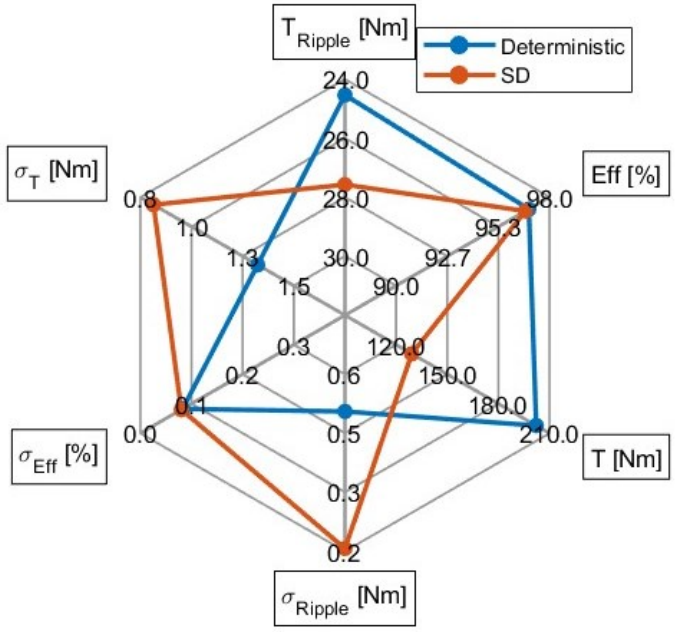

Figure 14. Average performance of the optimal deterministic and MOMSD designs.

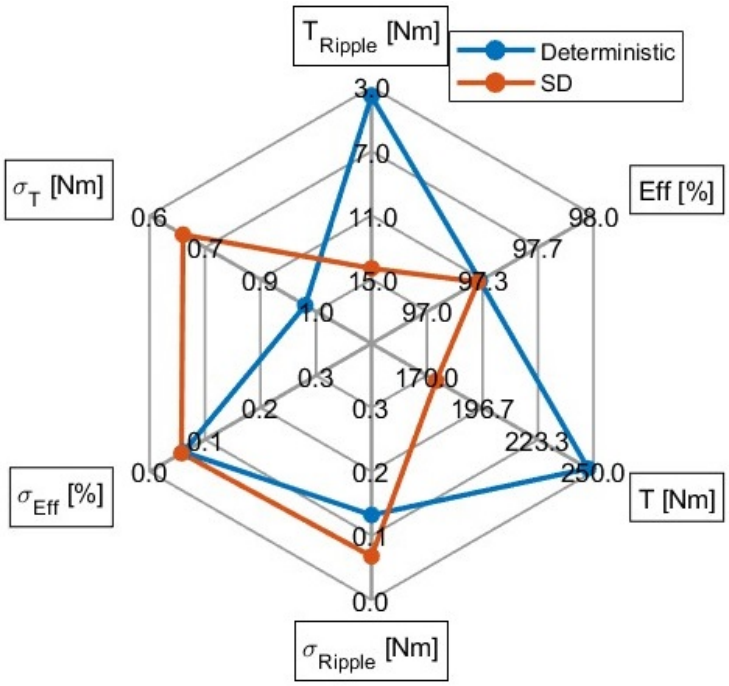

Figure 15. Utopia points of the deterministic and MOMSD optimal solutions.

4.3.2. Comparison Between β-Efficient Approach and Deterministic Optimal Solutions

To compare the results of the different formulations of the optimization problem, the expected value and the standard deviation of each objective and constraint function for each optimal design are implied. These values are computed by means of the neural networks and NNVE at the end of each optimization.

The results of the β-efficient approach are shown and compared to the deterministic optimal solutions in Figures 17 and 18. In these images, the results are almost overlapped. The deterministic method can therefore already be considered robust in this case study. For this reason, the β-efficient method tends to behave almost as a deterministic approach. In fact, this robust optimization approach takes into account the expected value of the

performance. The results are so close that, given a point from the deterministic solutions, on average, there exists a point belonging to the β-efficient optimal designs, whose values of the performance indexes differ by less than 1% from the values of the deterministic point. This gap is even slightly lower for the Pareto obtained with $\beta = 95\%$.

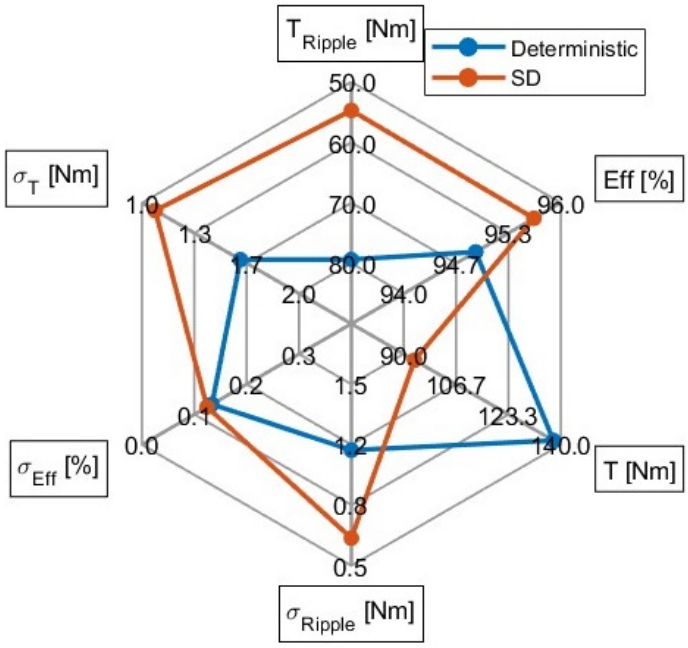

Figure 16. Nadir points of the deterministic and MOMSD optimal solutions.

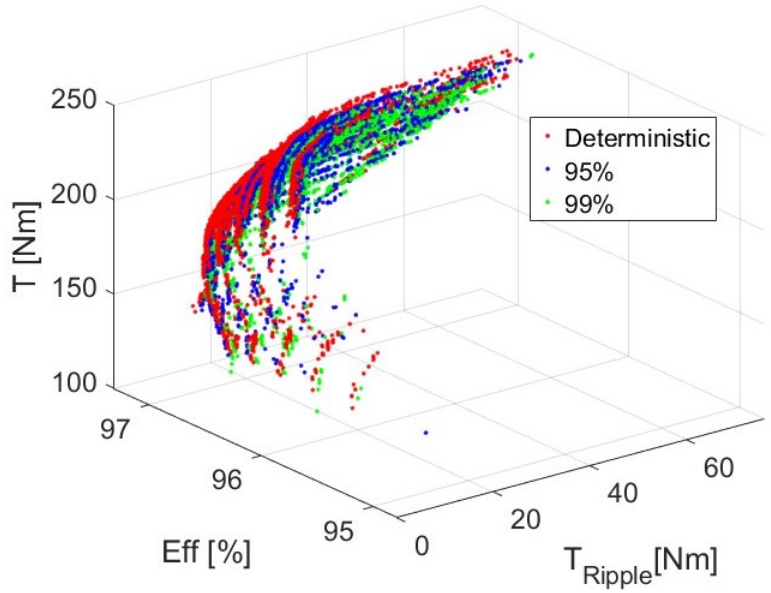

Figure 17. Comparison between the expected values of the objective functions obtained with the deterministic approach and with the β-efficient approach (β = 95% and β = 99%).

This can also be seen in Figure 19, where the average performance of the deterministic and β-efficient optimal designs is shown. As can be seen, all the performance indexes are very close. The only one in which there are evident differences is the torque, which differs by roughly 2 Nm.

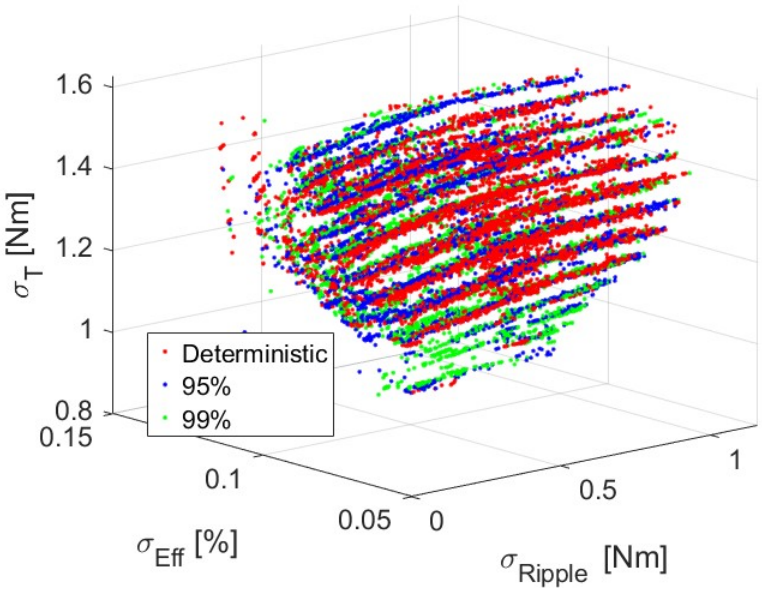

Figure 18. Comparison between the standard deviations of the objective functions obtained with the deterministic approach and with the β-efficient approach (β = 95% and β = 99%).

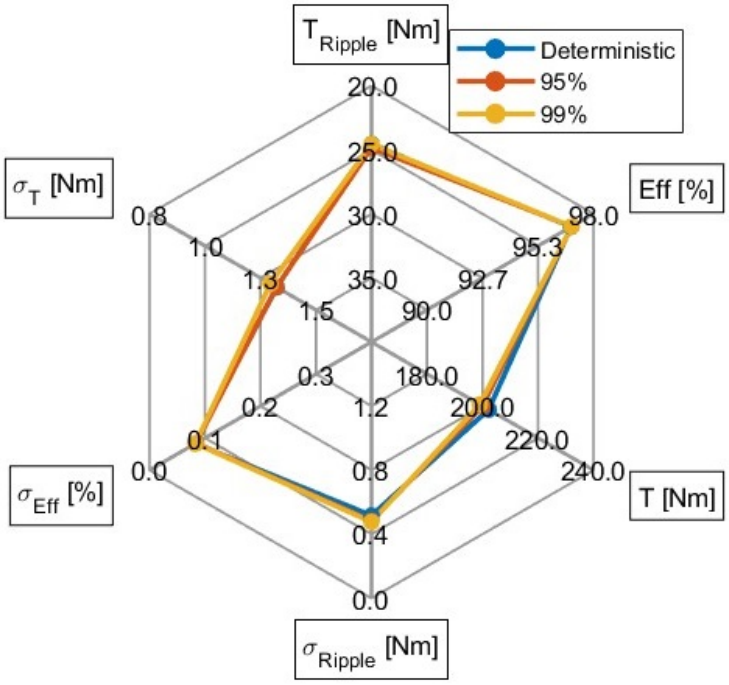

Figure 19. Average performance of the optimal deterministic and β-efficient designs.

Although all these distributions of optimal solutions are very similar, the deterministic designs do not consider robustness and so feature slightly better in terms of expected value.

As shown in Figures 20 and 21, the values of the nadir and utopia points for the three approaches are quite close, too. This means that the ranges of the results of the three approaches are also very similar. The only objective function that assumes slightly different values at the nadir and utopia points obtained with the different approaches is the torque ripple, which can differ by slightly more than 1 Nm.

A well-known advantage of β-efficient optimization is that it allows for greater safety margins against constraint violation [25]. This aspect is confirmed by the maximum voltages of all the optimal designs obtained with the proposed approaches, which are, respectively,

649.9 V for the deterministic approach, 646.6 V for the β-efficient approach with $\beta = 95\%$, and 645.6 V for the β-efficient approach with $\beta = 99\%$. When the β-efficient formulation is employed, the neural networks are asked to output the standard deviation of the objective and constraint functions in addition to the expected value, but the network evaluation number remains the same. Therefore, there is no significant difference in the speed of the method.

In order to describe the differences between the approaches more precisely, Table 7 shows the design variables, the performance indices, and the constraint functions for designs whose performance is close to the average of the results obtained with each approach. Obviously, the similar performance of deterministic and β-efficient approaches is due to very similar designs. On the contrary, as can be seen from variables such as the airgap or the number of turns, the MOMSD finds optimal designs in completely different regions of the design domain. As far as performance is concerned, Table 7 repeats the trends seen above with β-efficient beta solutions close to the deterministic and the MOMSD designs far from the deterministic solutions in terms of expected value and closer in terms of standard deviation.

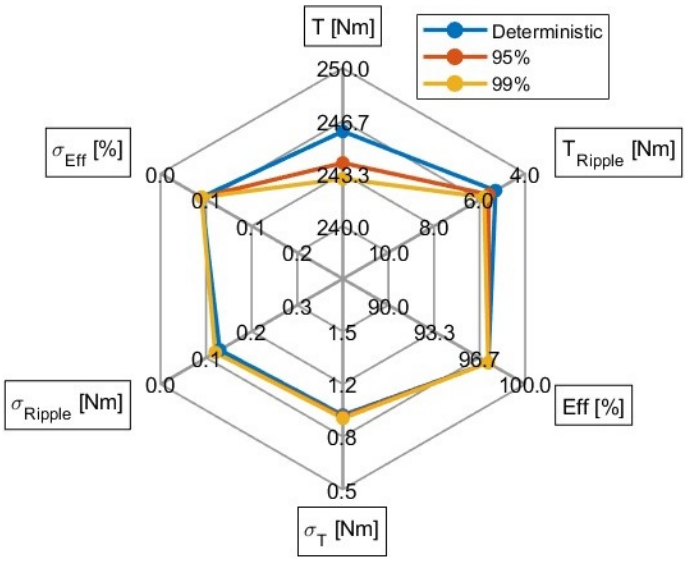

Figure 20. Utopia points of the deterministic and β-efficient ($\beta = 95\%$ and $\beta = 99\%$) optimal solutions.

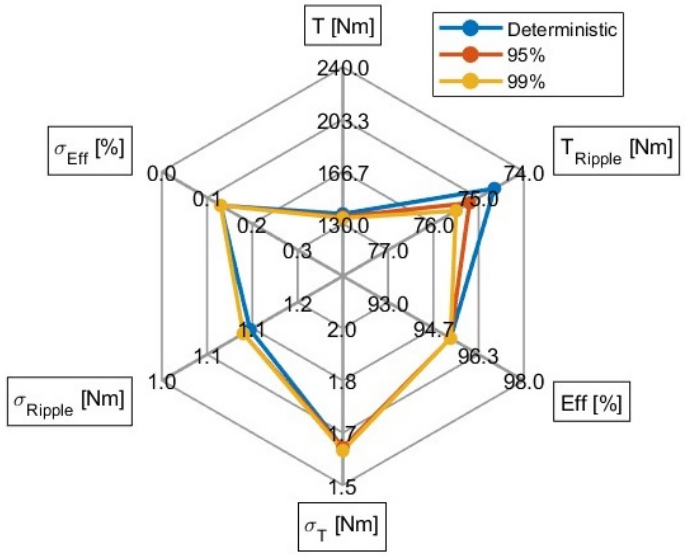

Figure 21. Nadir points of the deterministic and β-efficient ($\beta = 95\%$ and $\beta = 99\%$) optimal solutions.

Table 7. Design variables, performance indices, and constraint functions for designs whose performance is close to average for those found with each approach.

	Deterministic	**β-Eff (95%)**	**β-Eff (99%)**	**MOMSD**
Airgap	0.716 mm	0.717 mm	0.75 mm	1.23 mm
Slot opening	1.84 mm	1.56 mm	1.44 mm	1.45 mm
Magnet thickness	5.2 mm	5.11 mm	5.09 mm	5.17 mm
Magnet width	21.52 mm	21.63 mm	21.84 mm	19.4 mm
Tooth width	6.51 mm	6.45 mm	6.18 mm	7.73 mm
Bridge thickness	0.46 mm	0.46 mm	0.61 mm	0.57 mm
Web thickness	11.66 mm	11.55 mm	11.55 mm	7.52 mm
Pole V angle	153.4°	148.6°	150.1°	130.6°
Number of turns	27	27	28	20
Number of strands in hand	12	12	12	13
T_{Ripple}	25.1 Nm	25.5 Nm	24.5 Nm	27.15 Nm
Eff	97.1%	97.1%	97%	97%
T	205 Nm	205 Nm	205 Nm	142 Nm
V_{max}	507 V	512 V	526 V	349.8 V
$T_{mag,max}$	139 °C	137 °C	144 °C	99 °C
$T_{win,max}$	153 °C	152 °C	159 °C	105 °C
σ_{Ripple}	0.55 Nm	0.51 Nm	0.57 Nm	0.19 Nm
σ_{Eff}	0.008%	0.008%	0.008%	0.007%
σ_{T}	1.32 Nm	1.33 Nm	1.37 Nm	0.93 Nm

5. Conclusions

In this work, a robust multi-objective optimization method, based on different sources of uncertainty, is presented and compared to a deterministic one, in the context of the optimization of an IPMSM. At the beginning, a surrogate model of the motor, based on two feedforward neural networks trained with data coming from MotorCAD simulations, is realized. These neural networks are used, at the same time, to estimate the expected values and the standard deviation, which is used to quantify the variability of the objective functions and constraints. Then, the model is optimized by different methods: a deterministic approach, a multi-objective minimization of the standard deviation (MOMSD), and a β-efficient formulation with two levels of β (95% and 99%). Torque, torque ripple, and efficiency are chosen as performance indexes, while the design parameters are relevant geometric dimensions of the electric motor and parameters concerning the winding configuration. In particular, the uncertainties in the magnet properties, operating temperature, manufacturing tolerances, and geometric dimensions are considered when calculating the standard deviation. Constraints concerning the value of the dimensions, the feasibility of the geometry, the operating voltage, and the peak operating temperature of the magnets and windings are also taken into account. The proposed robust optimizations leverage an innovative use of neural network-based variance evaluation to efficiently calculate the standard deviations of the objective and constraint functions. All the optimization problems are solved by means of the non-dominated sorting genetic algorithm developed. The method is based on three key assumptions: the local linearity of the objective functions, the Gaussian distribution of uncertainties, and the independence of uncertainties. As a result, it can be generalized and applied to any type of electric motor. Once the uncertainty

tolerance range and design constraints are defined, and an electromagnetic model of the motor is available, the method can be implemented. This generalization is facilitated by the versatility of neural networks, the NNVE approach, the β-efficient method, and the non-dominated sorting genetic algorithm.

The results show that the usage of a deterministic formulation leads to designs which are already close to the optimal robustness. On the other hand, when the robustness is maximized, the expected value of the performance indexes worsens. Finally, the use of a β-efficient formulation leads to solutions close to the deterministic ones. In fact, they represent good compromises between performance and robustness. However, at the same time, these results are achieved with a higher margin of safety with respect to the constraints. As far as computational time is concerned, there is practically no difference between the deterministic and the robust method because the number of evaluations of the neural networks remains the same. The only difference is that in one case the neural networks are only asked to calculate the expected value and in the other also to calculate the standard deviation.

In the end, it is also important to consider that these results are obtained for a specific motor, which could have a lower sensitivity to uncertainties with respect to other ones. Future work could involve applying this method to a broader range of motor types to study the sensitivity to uncertainties in different motor architectures. Furthermore, this research utilized β-efficient and deterministic formulation of the optimization problem. It would be interesting to compare these results with those obtained using other formulations (e.g., worst case optimization and Taguchi methods).

Author Contributions: Conceptualization, M.G. and G.G.; methodology, M.G.; software, G.G.; validation, G.G., D.B., and F.S.; formal analysis, D.B. and F.S.; investigation, G.G.; resources, M.G.; data curation, G.G.; writing—original draft preparation, G.G., D.B. and F.S.; writing—review and editing, M.G.; supervision, M.G.; funding acquisition, M.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received received partial funding from the European Union Next-Generation EU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR)—MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4—D.D. 1033 17/06/2022, CN00000023).

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Coppola, P.; Arsenio, E. Driving Societal Changes Towards an Electromobility Future. *Eur. Transp. Res. Rev.* **2015**, *7*, 37. [CrossRef]
2. Dong, H.; Hu, Q.; Li, D.; Li, Z.; Song, Z. Predictive battery thermal and energy management for connected and automated electric vehicles. *IEEE Trans. Intell. Transp. Syst.* 2024, *early access*. [CrossRef]
3. Soresini, F.; Barri, D.; Ballo, F.; Gobbi, M.; Mastinu, G. Noise and vibration modelling of Permanent Magnet Synchronous Motors—A review. *IEEE Trans. Transp. Electrif.* **2024**, *10*, 8728–8745. [CrossRef]
4. Baranyai, M.; Mosavi, A.; Vajda, I.; Varkonyi-Koczy, A.R. Optimal design of electrical machines: State of the art survey. In *Recent Advances in Technology Research and Education, Proceedings of the 16th International Conference on Global Research and Education Inter-Academia 2017, Iași, Romania, 25–28 September 2017*; Springer: Cham, Switzerland, 2018; pp. 209–216. [CrossRef]
5. Levi, F.; Gobbi, M.; Mastinu, G. An application of multi-objective stochastic optimisation to structural design. *Struct. Multidiscip. Optim.* **2005**, *29*, 272–284. [CrossRef]
6. Sardar, M.U.; Manfeng, D.; Saleem, U.; Nawaz, M.K.; Hassan, M. A scholarly review of metods for design optimization of IPM synchronous motors used in electric vehicles. In Proceedings of the 2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE), Lahore, Pakistan, 2–4 December 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–6. [CrossRef]

7. Barri, D.; Soresini, F.; Gobbi, M.; Mastinu, G. Comparison of multi-objective optimisation methods for the design of electric motors. In Proceedings of the International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, St. Louis, MO, USA, 14–17 August 2022; American Society of Mechanical Engineers: New York, NY, USA, 2022; Volume 86205, p. V001T01A012. [CrossRef]
8. Ma, S.; Chen, K.; Zhang, Q. Analysis of Multi-Objective Optimization Design of Interior Double Radial and Tangential Combined Magnetic Pole Permanent Magnet Drive Motor for Electric Vehicles. *World Electr. Veh. J.* **2024**, *15*, 142. [CrossRef]
9. Yan, S.; Zhang, X.; Gao, Z.; Wang, A.; Zhang, Y.; Xu, M.; Hua, S. Design optimization of a new hybrid excitation drive motor for new energy vehicles. *World Electr. Veh. J.* **2022**, *14*, 4. [CrossRef]
10. Yuan, Y.; Meng, W.; Sun, X.; Zhang, L. Design optimization and analysis of an outer-rotor direct-drive permanent-magnet motor for medium-speed electric vehicle. *World Electr. Veh. J.* **2019**, *10*, 16. [CrossRef]
11. Guo, L.; Yu, H.; Wang, H. Design and Optimization of External Rotor Consequent Pole Permanent Magnet Motor with Low Iron Loss and Low Torque Ripple. *World Electr. Veh. J.* **2024**, *15*, 232. [CrossRef]
12. Barri, D.; Soresini, F.; Gobbi, M.; Di Gerlando, A.; Mastinu, G. Optimal design of Traction Electric Motors by a new Adaptive Pareto Algorithm. *IEEE Trans. Veh. Technol.* 2025, *early access*. [CrossRef]
13. Gupta, A.; Machavaram, R.; Kshatriya, T.; Ranjan, S. Multi-objective design optimization of a three phase squirrel cage induction motor for electric propulsion system using genetic algorithm. In Proceedings of the 2020 IEEE First International Conference on Smart Technologies for Power, Energy and Control (STPEC), Nagpur, India, 25–26 September 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–6. [CrossRef]
14. Ajie, I.K.; Furqani, J.; Fadillah, A. Optimization of Switched Reluctance Motor for Two Wheels Electric Vehicle. In Proceedings of the 2023 International Conference on Electrical Engineering and Informatics (ICEEI), Bandung, Indonesia, 10–11 October 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–6. [CrossRef]
15. Sardar, M.U.; Manfeng, D.; Saleem, U.; Hassan, M.; Nawaz, M.K. State-of-the-Art Design Optimization of an IPM Synchronous Motor for Electric Vehicle Applications. In Proceedings of the 2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE), Lahore, Pakistan, 2–4 December 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–6. [CrossRef]
16. Kang, B.; Choi, K.; Kim, D.H. An efficient serial-loop strategy for reliability-based robust optimization of electromagnetic design problems. *IEEE Trans. Magn.* **2017**, *54*, 7000904. [CrossRef]
17. Rathod, V.; Yadav, O.P.; Rathore, A.; Jain, R. Reliability-based robust design optimization: A comparative study. In Proceedings of the 2011 IEEE International Conference on Industrial Engineering and Engineering Management, Singapore, 6–9 December 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1558–1563. [CrossRef]
18. Wang, Z.; Yin, G.; Jin, X. Design of robust controllers for active suspension using the robust H∞ optimal control. In Proceedings of the 2016 35th Chinese Control Conference (CCC), Chengdu, China, 27–29 July 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 8818–8823. . [CrossRef]
19. Orosz, T.; Rassõlkin, A.; Kallaste, A.; Arsénio, P.; Pánek, D.; Kaska, J.; Karban, P. Robust design optimization and emerging technologies for electrical machines: Challenges and open problems. *Appl. Sci.* **2020**, *10*, 6653. [CrossRef]
20. Sardar, M.U.; Manfeng, D.; Hassan, M.; Saleem, U. Robust Design Optimization of an IPM Synchronous Motor for Electric Vehicle Applications. In Proceedings of the 2023 IEEE International Conference on Emerging Trends in Engineering, Sciences and Technology (ICES&T), Bahawalpur, Pakistan, 9–11 January 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–6. [CrossRef]
21. Lee, S.; Kim, K.; Cho, S.; Jang, J.; Lee, T.; Hong, J. Optimal design of interior permanent magnet synchronous motor considering the manufacturing tolerances using Taguchi robust design. *IET Electr. Power Appl.* **2014**, *8*, 23–28. [CrossRef]
22. Cheng, Y.; Ding, L.; Zhao, T.; Cui, S. Design and Optimization of Electric Vehicle Traction Motor Considering Rotor Topology and Manufacturing Uncertainty. *IEEE Trans. Ind. Electron.* **2023**, *71*, 5034–5044. [CrossRef]
23. Yang, Y.; Bianchi, N.; Zhang, C.; Zhu, X.; Liu, H.; Zhang, S. A method for evaluating the worst-case cogging torque under manufacturing uncertainties. *IEEE Trans. Energy Convers.* **2020**, *35*, 1837–1848. [CrossRef]
24. Taran, N.; Rallabandi, V.; Ionel, D.M.; Zhou, P.; Thiele, M.; Heins, G. A systematic study on the effects of dimensional and materials tolerances on permanent magnet synchronous machines based on the IEEE Std 1812. *IEEE Trans. Ind. Appl.* **2018**, *55*, 1360–1371. [CrossRef]
25. Mun, J.; Lim, J.; Kwak, Y.; Kang, B.; Choi, K.; Kim, D.H. Reliability-based design optimization of a permanent magnet motor under manufacturing tolerance and temperature fluctuation. *IEEE Trans. Magn.* **2021**, *57*, 8203304. [CrossRef]
26. Aggarwal, A.; Strangas, E.G.; Agapiou, J. Analysis of unbalanced magnetic pull in pmsm due to static eccentricity. In Proceedings of the 2019 IEEE Energy Conversion Congress and Exposition (ECCE), Baltimore, MD, USA, 29 September–3 October 2019; IEEE:Piscataway, NJ, USA, 2019; pp. 4507–4514. [CrossRef]

27. Gobbi, M. Optimal and robust design of ground vehicle systems. In Proceedings of the Engineering Systems Design and Analysis, Torino, Italy, 4–7 July 2006; Volume 42495, pp. 225–234. [CrossRef]
28. Mastinu, G.; Gobbi, M.; Miano, C. *Optimal Design of Complex Mechanical Systems: With Applications to Vehicle Engineering*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2007. [CrossRef]
29. Ansys. *Motor-CAD v15.1 Manual*; Ansys: Canonsburg, PA, USA, 2021.
30. Cavazzuti, M.; Gaspari, G.; Pasquale, S.; Stalio, E. Thermal management of a Formula E electric motor: Analysis and optimization. *Appl. Therm. Eng.* **2019**, *157*, 113733. [CrossRef]
31. Agbemenou, A.K.H.; Motamed, R.; Talaei-Khoei, A. A predictive analytics model for designing deep underground foundations using artificial neural networks. *Decis. Anal. J.* **2023**, *7*, 100220. [CrossRef]
32. Cheng, L.; Wang, Z.; Jiang, F.; Li, J. Adaptive neural network control of nonlinear systems with unknown dynamics. *Adv. Space Res.* **2021**, *67*, 1114–1123. [CrossRef]
33. Zhang, C.; Lu, Y. Study on artificial intelligence: The state of the art and future prospects. *J. Ind. Inf. Integr.* **2021**, *23*, 100224. [CrossRef]
34. Im, D.H.; Park, S.C.; Park, D.J. Optimum design of single-sided linear induction motor using the neural networks and finite element method. In Proceedings of the 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan), Nagoya, Japan, 25–29 October 1993; IEEE: Piscataway, NJ, USA, 1993; Volume 3, pp. 2811–2814. [CrossRef]
35. Hadjout, L.; Takorabet, N.; Ibtiouen, R.; Mezani, S. Optimization of instantaneous torque shape of PM motors using artificial neural networks based on FE results. *IEEE Trans. Magn.* **2006**, *42*, 1283–1286. [CrossRef]
36. Beyer, H.G.; Sendhoff, B. Robust optimization—A comprehensive survey. *Comput. Methods Appl. Mech. Eng.* **2007**, *196*, 3190–3218. [CrossRef]
37. Grotti, E.; Santana, P.B.; Filho, J.G.P.; Gomes, H.M. Multiobjective robust optimization framework based on first and second order Taylor expansion applied to a vehicle suspension design. *Optim. Eng.* **2024**, *25*, 637–668. [CrossRef]
38. Ma, B.; Lei, G.; Zhu, J.; Guo, Y.; Liu, C. Application-oriented robust design optimization method for batch production of permanent-magnet motors. *IEEE Trans. Ind. Electron.* **2017**, *65*, 1728–1739. [CrossRef]
39. Lei, G.; Zhu, J.; Guo, Y.; Shao, K.; Xu, W. Multiobjective sequential design optimization of PM-SMC motors for six sigma quality manufacturing. *IEEE Trans. Magn.* **2014**, *50*, 717–720. [CrossRef]
40. Caballero, R.; Cerdá, E.; Munoz, M.; Rey, L.; Stancu-Minasian, I. Efficient solution concepts and their relations in stochastic multiobjective programming. *J. Optim. Theory Appl.* **2001**, *110*, 53–74. [CrossRef]
41. Gobbi, M. A k, k-ε optimality selection based multi objective genetic algorithm with applications to vehicle engineering. *Optim. Eng.* **2013**, *14*, 345–360. [CrossRef]
42. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
43. Di Gerlando, A.; Ricca, C. Design Modeling and Sizing Equations of V-shape IPM Motors. In Proceedings of the 2022 International Conference on Electrical Machines (ICEM), Valencia, Spain, 5–8 September 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1061–1067. [CrossRef]
44. Mun, J.; Choi, K.; Kim, D.H. System-Level Robust Design Optimization of a Permanent Magnet Motor Under Design Parameter Uncertainties. *IEEE Trans. Magn.* **2023**, *59*, 8201104. [CrossRef]
45. Sun, X.; Wan, B.; Lei, G.; Tian, X.; Guo, Y.; Zhu, J. Multiobjective and multiphysics design optimization of a switched reluctance motor for electric vehicle applications. *IEEE Trans. Energy Convers.* **2021**, *36*, 3294–3304. [CrossRef]
46. Kim, K.S.; Lee, B.H. Taguchi's robust design optimisation of water-cooled ISG motors considering manufacturing tolerances. *IET Electr. Power Appl.* **2020**, *14*, 865–871. [CrossRef]
47. Suriano-Sánchez, S.I.; Ponce-Silva, M.; Olivares-Peregrino, V.H.; De León-Aldaco, S.E. A Review of Torque Ripple Reduction Design Methods for Radial Flux PM Motors. *Eng* **2022**, *3*, 646–661. [CrossRef]
48. Lee, T.Y.; Seo, M.K.; Kim, Y.J.; Jung, S.Y. Cogging torque of surface-mounted permanent magnet synchronous motor according to segmented-stator core effect. In Proceedings of the 2016 XXII International Conference on Electrical Machines (ICEM), Lausanne, Switzerland, 4–7 September 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 200–206. [CrossRef]
49. MathWorks. *MATLAB Deep Learning Toolbox*; MathWorks: Natick, MA, USA, 2023.
50. Fathi, E.; Maleki Shoja, B. Chapter 9—Deep Neural Networks for Natural Language Processing. In *Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications*; Gudivada, V.N., Rao, C., Eds.; Handbook of Statistics; Elsevier: Amsterdam, The Netherlands, 2018; Volume 38, pp. 229–316. [CrossRef]
51. Li, G.; Zhang, A.; Zhang, Q.; Wu, D.; Zhan, C. Pearson correlation coefficient-based performance enhancement of broad learning system for stock price prediction. *IEEE Trans. Circuits Syst. II Express Briefs* **2022**, *69*, 2413–2417. [CrossRef]

52. Ištoka Otković, I.; Varevac, D.; Šraml, M. Analysis of neural network responses in calibration of microsimulation traffic model. *Electron. J. Fac. Civ. Eng. Osijek-E* **2015**, *6*, 67–76. [CrossRef]
53. Niederreiter, H. *Random Number Generation and Quasi-Monte Carlo Methods*; SIAM: Philadelphia, PA, USA, 1992. [CrossRef]