Deep Neural Network Based Electro-Mechanical Optimization of Electric Motors

Sami Barmada[1], Mauro Tucci[1], Luca Sani[1], and Cesare Tozzo[2]
[1]DESTEC, University of Pisa, Italy
[2]COMSOL, Brescia, Italy

In this contribution the authors use a Deep Neural Network based approach for the optimization of an electric motor, taking into account both electromagnetic and mechanical constraints, i.e. approaching the problem from the multiphysics point of view. In the design process of high speed electric motors, the mechanical design of the rotor is of noteworthy importance, and in case of reluctance motors it cannot be separated from the electromagnetic design. The multiphysics model is created by using a commercial FEM software, and a multiobjective optimization procedure is run by using the before mentioned software. This is the selected tool for the generation of the training dataset used to train a Deep Neural Network, that is used to refine the sub-optimal solutions previously obtained. The results show that the use of a two-step optimization lead to a better solution.

Index Terms—**deep learning, surrogate model, topology optimization**

I. Introduction

LATELY the use of artificial intelligence based techniques for the solution of direct problems in electromagnetics is becoming a popular research topic. As a matter of fact, a complete substitution of the commonly used numerical solution methods with Neural Netowrks (NN) models is not practicable. However, when properly trained on a specific problem (with the training set being obtained by the before mentioned numerical solutions), NN can significantly reduce the overall computational burden when repeated simulations must be run. For this reason, NN are becoming a popular tool to create a surrogate model of the electromagnetic device during an optimization procedure. [1]. Deep Learning (DL) approaches, when topology optimization is used, are a solution that is often selected: as a matter of fact, implemented in the form of Convolutional Neural Networks (CNN), they are capable of dealing with images as input and output, thus enabling to directly handle the bitmaps describing the input geometry, and give as output the desired EM quantity. [1] and [2] are examples of the application of DNN to an optimization problem related to electromagnetic devices. The authors in [2] proposed a DNN approach for the optimization of a reluctance motor, with a further reduction of the computational burden by the use of an additional NN with the aim of reducing the Finite Element Method (FEM) domain to the only part of the machine to be optimized, while in [3] they introduce regularization procedures to better understand the overall performances of DNN based method. [4] offers an overview of the application of NN based techniqes to computational electromagnetics.

In this contribution the authors introduce the use of multiphysics simulations together with DNN for the optimization of a reluctance motor: in this case not only the flux barriers are shaped in order to obtain an objective (maximum torque), but also mechanical stress due to rotation is taken into account. As a matter of fact the pure optimization of the flux barriers' shape can lead to sub-otpimal solutions from the mechanical point of view, that could lead to critical situations when the rotor nominal rotational speed is high. In these cases the designer introduces additional iron bridges to improve mechanical performances (reducing stress on specific spots); such additional bridges must be placed and designed properly, since while they improve the system from the mechanical point of view, they can create additional flux path inside the rotor reducing the motor's performances. To the author's knowledge in the recent literature there are many attempts to use DNNs as surrogate models to solve electromagnetic problems, but a coupled multi-physic model that involves both electromagnetic and mechanical modelling has not yet been analysed by the use of an advanced Deep Learning approach, and this is one of the main contributions of this paper. Another important contribution is the generation of the learning dataset by using the intermediate steps of a preliminary optimization. The paper is organized as follows: in section II the details of the FEM model are given; in section III the way the surrogate model is obtained, starting from the FEM model implemented in the commercial software, is described; in Section IV the results are shown.

Corresponding author: S. Barmada (email: sami.barmada@unipi.it).

II. Finite Element Method Modelling

A. Electromagnetic and Mechanical Modelling

The model of the reluctance motor is shown in Fig 1 and it has been implemented on the commercial software COMSOL Multiphysics (named the software from now on). The FEM model is based on a standard vector potential formulation, with a moving mesh (with interface in the air-gap) to address different rotor positions. The EM torque has been accurately calculated by using the Arkkio's method, which guarantees higher accuracy if compared to the standard use of the Maxwell's stress tensor and is expressed in equation (1)

$$T = \frac{\ell_{axial}}{(r_{out} - r_{in})\mu_0} \int_0^{2\pi} \int_{r_{in}}^{r_{out}} B_r B_\theta r dS. \qquad (1)$$

in which r_{out} and r_{in} are the inner and outer radius of the air gap of the rotor.
The mechanical modelling of the rotor has been performed by considering the usual quantities related to the stress evaluation (Young's modulus, Poisson ratio). The high rotating speed of the rotor (18000 rpm) imposes a centripetal force that can be critical in some specific configuration of the flux barriers; in particular, narrower portions of the rotor, resulting from flux barriers shapes can be subject to mechanical stress dangerously reaching the yield strength.
In the multiphysics model, the centripetal force has been modeled by imposing a distributed radial mechanical load on the rotor, equal to

$$f = \rho \Omega r^2, \tag{2}$$

in which Ω is the angular speed of the rotor, ρ is the mass density and r is the radius variable. The optimization procedure aims at finding optimum torque and minimum mechanical stress on the rotor; in particular, as far as the mechanical stress is concerned, different objective functions have been considered,

1) the evaluation average of the Von Mises stress;
2) the evaluation of the maximum Von Mises stress;

As clearly explained in section III, the model implemented by the use of the commercial software is used to generate the dataset for the DNN training. Under this point of view, running different optimization procedures with different, yet meaningful, mechanical objective functions allows the exploration of different design solution. At the same time it is obvious that, between the two objective functions described above, number 2) is the one that has the highest design meaning: the maximum values of the mechanical stress is located at higher radius and close to material discontinuity (flux barriers). These peaks are in limited number and do not affect much the average value; on the contrary, the peak value is significant for the reaching of the yield strength.

B. Topology optimization implementation

The topology optimization performed by the software is based on [5] and considers a material assuming (soft) values represented by a topological control variable θ_c between 0 and 1 discretized on each element. The interpolation is specific to the physics (in this case mechanical or electromagnetic), and it is constructed such that intermediate value of the control variable are suboptimal; in addition, the need of imposing a minimum length scale R_{min} on the domain control variable, is imposed by the Helmholtz filter

$$\theta_f = R_{min}^2 \nabla^2 \theta_f + \theta_c, \tag{3}$$

in which R_{min} is the mesh element size [5]. This is a way to have a good regularization of the optimization problem, but leads to areas in which the filtered variable θ_f is characterized by intermediate values (i.e. non physical meaning). These areas can then be reduced using a projection operation based on the hyperbolic tangent function

$$\theta = \frac{\tanh(\beta(\theta_f - \theta_\beta)) + \tanh(\beta\theta_\beta)}{\tanh(\beta(1-\theta_\beta)) + \tanh(\beta\theta_\beta)}, \tag{4}$$

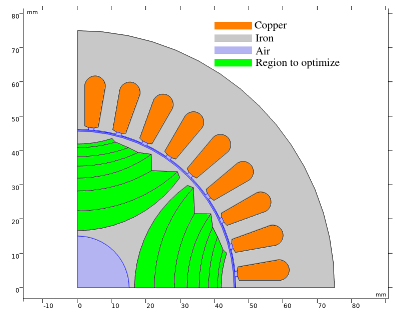

Fig. 1: Motor initial geometry.

in which θ is the output material volume factor, while θ_β and β respectively are the projection slope and point.

The multiphysics model of the reluctance motor is characterized by the above mentioned parametrization of the rotor; the electromagnetic behaviour of the ferromagnetic material is represented by a magnetic permeability expressed as

$$\mu_r = 1 + \theta^{K_1}\left(\frac{B}{H\mu_0} - 1\right), \tag{5}$$

in which B and H are linked by the nonlinear $B-H$ characteristics of the steel, and K_1 is an integer term (in this case assumed as $K_1 = 3$) introduced by the authors with the aim of making the permeability as much as possible close to the physical values. Regarding the physical characteristics of the steel used in the mechanical model, they are expressed as

$$\rho_m = \rho_0 \theta^{K_2} + 1, \tag{6}$$

$$E = E_0 \theta^{K_2} + 1, \tag{7}$$

in which ρ_0, E_0 respectively are the density and the Young modulus of the material, θ is the topological variable introduced before and K_2 is an integer term (assuming the value of $K_2 = 5$ in this case) playing the same role as K_1 in equation (5).

In order to guarantee in any case a physical realization of the rotor, only a subset of it (that is the inner part where usually are located the flux barriers) has been set as object of the optimizaiton, i.e. the green region in Fig. 1.

III. Surrogate DNN Modelling

A. Characteristics of the surrogate model

We aim to develop and investigate a surrogate model based on DNNs, that shall be able to predict the electromagnetic torque and the mechanical stress given a geometry image as input. Attention shall be paid to three main aspects when building a deep learning model: the architecture of the model, the optimization of the hyper-parameters and the generation of the learning data-set. In this work, the latter is the most critical point, for which we propose an innovative approach. In many previous works the training data-set is generated randomly [3], with the disadvantage that most of the samples are expected to be far from the optimal regions. However, previous works [2], [3] indicate that for the network to be able to optimize, the data set must represent in detail the regions where optimal solutions

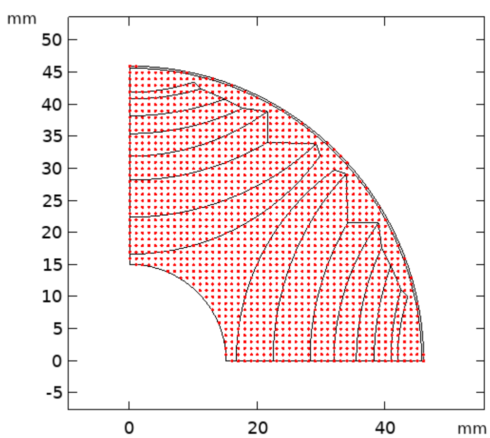

Fig. 2: Discretization of the geometrical rotor domain.

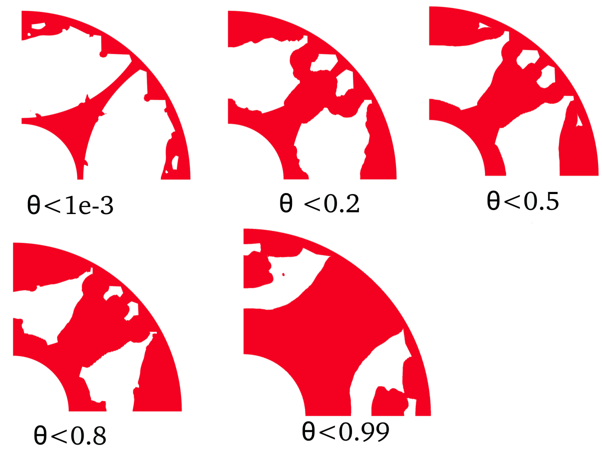

Fig. 3: Geometry as a function of the threshold.

are assumed to be present. A raw, preliminary optimization is then mandatory, and this is also a result confirmed by other authors [1].
In this work we generate the training data-set by collecting the intermediate results of a topology multi-objective optimization performed by the use of the software.

Based on the results obtained in terms of the variables expressed by equations [3], [4], we propose an original approach to augment the training samples by generating and simulating geometries where the material has hard values obtained using a varying threshold. In particular, at the end of the optimization process, each collected soft geometry can be processed in order to obtain a corresponding hard geometry given a threshold, then many solutions can be obtained for different threshold values, as shown in Fig. 3. This approach allows to populate a data-set with thousands of solutions very efficiently and targeted to the optimal solutions. Using such a data-set we trained a CNN, and tested it on a test data-set (20% of the data). The selected CNN architecture comprehends 6 convolutional layers, each one followed by a relu layer and max pooling, with the terminating dense layer followed by a regression layer. The CNN takes the geometry image as input and has two outputs: the electromagnetic torque T and the mechanical stress s.

B. Geometry representation and extraction

The input images to train the CNN are extracted form the intermediate results of the topology multi-objective optimization performed directly in the software. The resolution of the image can be arbitrarily selected after the optimization by creating a grid of points (a 2D cut point) where an expression related to the variable θ can be evaluated for each optimization step. Figure 2 shows the selected 64×64 grid, where each point represents a pixel of the bitmap. A bitmap can be obtained comparing θ against a threshold $\theta > th$ in each point of the grid, associating a value of 1 or 0 to the pixel if the result of the comparison is true or false, respectively. In figure 2 only the points inside of the rotor domain are shown, outer points are set to 0, while inner points in regions not to be optimized are set to 1. Finally, an entire family of bitmaps can be obtained for each optimization step by sweeping the value of the threshold, as shown in figure 3. Following this procedure we generated a dataset with 5000 cases, each one comprehending an input geometry image and two output targets for the electromagnetic torque T and the mechanical stress s .

TABLE I: CNN test errors

Quantity	MAPE
Electromagnetic torque	3.4%
Mechanical stress	5%

C. Architecture of the CNN

In this paper we train a CNN as a surrogate model to predict two magneto-mechanical quantities of interest as a function of the input geometry. As a trade-off between accurate representation of the rotor shape and the necessity to reduce the size of the CNN input a 64×64 image resolution was selected. The architecture of the CNN was selected by comparing the performance on validation data of different models with different number of layers and filters per layer. The best obtained model is shown in table II, consisting in 23 layers, with six convolutional/relu/max pooling blocks. The first three blocks (layers from 2 to 10) maintain the size of the input data and are expected to prepare the input for the successive three blocks (layers from 11 to 19) that reduce the dimension to 64 features. The successive two fully connected layers perform the regression to predict the two outputs. The final regression layer performs the performance loss calculation.

IV. Results

A. CNN training and testing

Calculating the mean absolute percentage error ($MAPE$) on both outputs over the test data we obtain the results shown in table I, which validate the CNN and reveal the good accuracy of the model. $MAPE$ on a generic output variable x is defined as:

$$MAPE = \frac{100}{N_{test}} \sum_{k=1}^{N_{test}} \left| \frac{x_k^{pred} - x_k^{target}}{x_k^{target}} \right| \tag{8}$$

TABLE II: CNN architecture

Layer n°	Layer Type	Layer Parameters	Activations (Width, Height, Depth)
1	Image Input Layer	size = 64 × 64	64, 64, 1
2	Convolution 2d Layer	size = 4, n° filters = 4, padding = same	64, 64, 4
3	Relu Layer		64, 64, 4
4	Max Pooling 2d Layer	size = 2, stride = 2	32, 32, 4
5	Convolution 2d Layer	size = 4, n° filters = 16, padding = same	32, 32, 16
6	Relu Layer		32, 32, 16
7	Max Pooling 2d Layer	size = 2, stride = 2	16, 16, 16
8	Convolution 2d Layer	size = 4, n° filters = 64, padding = same	16, 16, 64
9	Relu Layer		16, 16, 64
10	Max Pooling 2d Layer	size = 2, stride = 2	8, 8, 64
11	Convolution 2d Layer	size = 4, n° filters = 64, padding = same	8, 8, 64
12	Relu Layer		8, 8, 64
13	Max Pooling 2d Layer	size = 2, stride = 2	4, 4, 64
14	Convolution 2d Layer	size = 4, n° filters = 64, padding = same	4, 4, 64
15	Relu Layer		4, 4, 64
16	Max Pooling 2d Layer	size = 2, stride = 2	2, 2, 64
17	Convolution 2d Layer	size = 4, n° filters = 64, padding = same	2, 2, 64
18	Relu Layer		2, 2, 64
19	Max Pooling 2d Layer	size = 2, stride = 2	1, 1, 64
20	Fully Connected Layer	size = 24	1, 1, 24
21	Relu Layer		1, 1, 24
22	Fully Connected Layer	size = 2	1, 1, 2
23	Regression Layer	Response Names = Electromagnetic Torque, Mechanical Stress	1, 1, 2

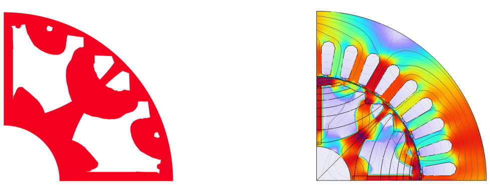

Fig. 4: Optimal geometry after NSGA optimization.

TABLE III: CNN optimization results

Quantity	Optimal value	
	Preliminary	CNN
Electromagnetic torque	12.2 Nm	12.9 Nm
Total elastic strain energy	1.77 J	1.74 J

B. DNN Optimization

The trained CNN was then used to further optimize the solution using a well known multi-objective genetic algorithm, NSGA-II [6]. The surrogate model is very fast then the optimization can exploit a large number of function evaluations. The final result, shown in Fig. 4, was then simulated in the FEM and the optimal solution previously found by the preliminary optimization was improved as shown in Table III (higher torque and lower mechanical stress). The time required to generate the dataset was 5000s, including the time for preliminary optimization. The time for training and optimizing the CNN was 1000s. It is obvious that the results obtained after a topology optimization procedure generally lead to geometries that do not take into account the aspects of cost and ease of fabrication. Further work will be to introduce such smoothing procedure in the DNN based model in order to reach this goal.

V. Conclusion

In this work the authors propose a Deep Neural Network based optimization procedure considering multiphysics modelling of an electric reluctance motor, in which both the torque and the mechanical stress of the rotor are optimized. The topology optimization is efficiently carried out by the DNN, while the training set is obtained by the use of a commercial FEM code in which a topology optimization has been implemented. The use of the FEM code allows an easy obtainment of a large number of data points, and the DNN refines the geometry. The results show the efficiency of the proposed technique.

References

[1] H. Sasaki, Y. Hidaka and H. Igarashi, "Explainable Deep Neural Network for Design of Electric Motors," in IEEE Transactions on Magnetics, vol. 57, no. 6, pp. 1-4, June 2021, Art no. 8203504, doi: 10.1109/TMAG.2021.3063141.

[2] S. Barmada, N. Fontana, A. Formisano, D. Thomopulos and M. Tucci, "A Deep Learning Surrogate Model for Topology Optimization," in IEEE Transactions on Magnetics, vol. 57, no. 6, pp. 1-4, June 2021, Art no. 7200504, doi: 10.1109/TMAG.2021.3063470.

[3] M. Tucci, S. Barmada, A. Formisano, and D. Thomopulos, "A Regularized Procedure to Generate a Deep Learning Model for Topology Optimization of Electromagnetic Devices," Electronics, vol. 10, no. 18, p. 2185, Sep. 2021, doi: 10.3390/electronics10182185.

[4] P. Di Barba, M. E. Mognaschi, S. Wiak, "Neural metamodelling of fields: Towards a new deal in computational electromagnetics" (2022) International Journal of Applied Electromagnetics and Mechanics, 69 (2), pp. 127-137.

[5] B. S. Lazarov and O. Sigmund, "Filters in topology optimization based on Helmholtz-type differential equations," International Journal for Numerical Methods in Engineering, vol. 86, no. 6, pp. 765–781, 2011.

[6] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," in IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182-197, April 2002, doi: 10.1109/4235.996017.