

Future Trends in Optimal Design of Electromagnetic Devices

Paolo Di Barba¹

²Department of Electrical, Computer, and Biomedical Engineering, University of Pavia, Pavia, I-27100, Italy

In computational electromagnetism there are manyfold advantages when using machine learning methods, because no mathematical formulation is required to solve the direct problem for given input geometry. Moreover, thanks to the inherent bidirectionality of a convolutional neural network, it can be trained to identify the geometry giving rise to the prescribed output field. All this puts the ground for neural meta-modeling of fields, in spite of different levels of cost and accuracy. For the sake of an example, a surrogate model of the field in a small device is shown. In particular, a concept of multi-fidelity model makes it possible to control both prediction accuracy and computational cost. In particular, generative adversarial networks can help multiobjective optimal design.

Index Terms—Deep neural network, generative adversarial network, magnetostatic field, optimal shape synthesis.

I. INTRODUCTION

IN RECENT years, the impressive developments in the area of machine learning and big data [1] have put the ground for pattern recognition in field problems. In this respect, the use of deep neural networks (DNN) might substantially accelerate the solution of field analysis problems. Accordingly, the possibility of applying a DNN models to field analysis and synthesis is here considered: in particular, a bitmap approach, which describes a device geometry as a set of pixels, is proposed. The architecture of the selected network resembles that for image segmentation purposes and it is easily trained by means of a set of finite-element (FE) analyses: this way, the ground information is amenable to electric and magnetic field distributions over the input domains, simulated from traditional FE solvers. Therefore, a physics-informed background makes the neural surrogate reliable in spite of different levels of cost and accuracy.

II. SURROGATE MODELLING WITH CNNs

Recently, deep learning techniques have been used to build surrogate models for field analysis of electromagnetic devices [2],[3]. In this respect, deep learning (DL) approaches are particularly suited for developing surrogate models in electromagnetic analysis. In fact, DL models like deep neural networks (DNN), are capable of dealing with high-resolution images as input and output, therefore enabling to directly handle the bitmaps describing geometry and field in a electromagnetically complex scenario. In particular, Convolutional Neural Networks (CNNs) are neural networks

which allow to evaluate a field quantity (either a value or a vector of values like *e.g.* a field distribution), starting from an image [4]; to this end, transposed 2D deconvolution layers are conveniently used. The training of this class of networks is usually based on numerical analyses and in particular the FE method is normally used as a standard. Once a neural network is trained, it can be used as a surrogate model and placed in an optimization loop. This way, the computational burden is limited to the net training, while the evaluation of the field quantities during the optimization loop is rather inexpensive. Another fascinating and innovative way to use DNNs for solving inverse problems is to invert the network previously trained for solving the associated direct problem: an application in material property identification can be found *e.g.* in [5].

III. PERSPECTIVES IN COMPUTATIONAL ELECTROMAGNETISM

It appears that more and more Machine Learning (ML) approaches are being exploited as a tool for the resolution of electromagnetic (EM) problems. The advantages of using ML are at least twofold:

- i) first, no mathematical formulation is required to implement the problem solver, as Deep Neural Networks (DNN) simply require well suited data sets for their training;
- ii) next, once trained, DNNs can solve an EM problem in an almost negligible time with respect to classical numerical methods: this puts the ground for equation-less, and so FEM-less, models of analysis problems.

An additional advantage of using ML is the possibility to exploit the inherent bidirectionality of this approach. In fact, DNN's can be trained

- either to promptly solve the direct electromagnetic problem for given input geometry and sources,
- or, alternatively, to look for the geometry giving the prescribed output field with assigned sources,
- or, finally, to look for field sources or material properties in an assigned geometry.

This, in turn, puts the ground for an optimization-less approach to the automated synthesis of electromagnetic devices. The whole scenario is tempting and surely deserves serious consideration in the community; at the time being, however, the training cost of a network is still a bottle-neck in

Manuscript received April 1, 2015; revised May 15, 2015 and June 1, 2015; accepted July 1, 2015. Date of publication July 10, 2015; date of current version July 31, 2015. (Dates will be inserted by IEEE; “published” is the date the accepted preprint is posted on IEEE Xplore®; “current version” is the date the typeset version is posted on Xplore®). Corresponding author: F. A. Author (e-mail: f.author@nist.gov). If some authors contributed equally, write here, “F. A. Author and S. B. Author contributed equally.” IEEE TRANSACTIONS ON MAGNETICS discourages courtesy authorship; please use the Acknowledgment section to thank your colleagues for routine contributions.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier (inserted by IEEE).

many EM applications, and future work should be addressed towards techniques mitigating the computational burden.

For the sake of an example, let an electrostatic micromotor exhibiting 18 stator electrodes, 6 rotor teeth and 60 μm outer rotor radius, be considered. The stator electrodes are supplied by a three-phase system of square voltages while the rotor potential is floating. A convolutional neural network (CNN) composed of 18 layers is used for implementing the surrogate model [6]; in fact, the architecture of the selected network resembles that of an image decoder/decompressor. In order to train a net, a two-dimensional FE model of the motor is implemented; the driving torque T at no-load condition is computed over a rotation angle of 60 degrees (pole pitch), and its maximum value T_m is considered.

The rotor geometry can vary depending on three parametric variables $x=[x_1, x_2, x_3]=[R_1, \alpha, \beta]$ which are the inner rotor radius and two angles defining the geometry of the rotor tooth, respectively. Two classes of problems have been solved: in terms of forward modelling, given a device geometry, identify the field pattern (CNN-1) or compute the torque profile (CNN-2) as shown in Fig.1. In contrast, in terms of inverse modelling, given the prescribed torque profile, find the relevant device geometry (CNN 3) as depicted in Fig. 2. In particular, CNN-3 is the conjugate network of CNN-2.

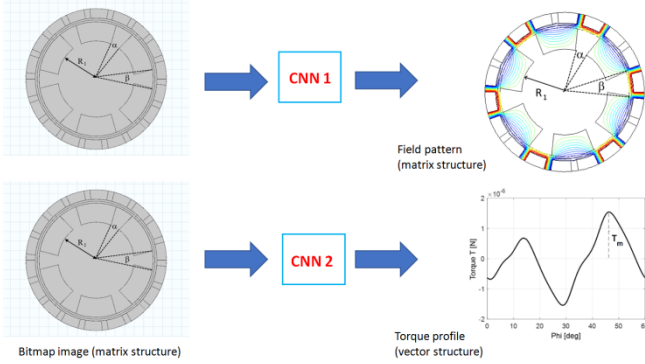


Fig. 1. CNN-based surrogate model of forward problems: given a micromotor geometry, find the relevant field pattern (up) or, alternatively, find the torque profile (down).

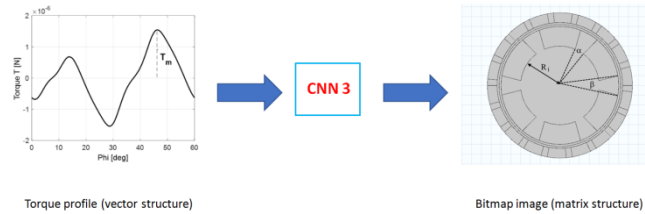


Fig. 2. CNN-based surrogate model of an inverse problem: given a prescribed torque profile, find the relevant micromotor geometry.

A general strategy in order to reduce the computational burden and still keep a good accuracy in the predicted values, could be the use of a multi-fidelity approach: a network is trained first with a low-fidelity model; then, it is treated as a pre-trained network and subsequently specialized with the high-fidelity model. This way, the pre-trained network has a low, or moderate, computational cost because of the low-fidelity model, while the subsequent training with high-fidelity model is not very expensive because a reduced set of samples is used for training the network.

Formally, a multi-fidelity surrogate fuses information from different models based on a space mapping σ which relates the high- (x_H) and the low-fidelity (x_L) variable spaces, namely

$$x_L = \sigma(x_H) \quad (1)$$

such that

$$f_L(\sigma(x_H)) \cong f_H(x_H) \quad (2)$$

where f_L and f_H are the low and the high fidelity models, respectively.

For the sake of an example, again referring to the previous case study, the high-fidelity field model consists of $n_H=101$ FE analyses, where a step of $\Delta_H=0.6$ degrees is considered for the rotation over 60 degrees; in contrast, $n_L=26$ FE analyses are performed for the low-fidelity field model, where a step of $\Delta_L=2.4$ degrees is considered for rotation.

This way, in general, a physics-informed background makes the neural surrogate reliable in spite of different levels of cost and accuracy. The main practical goal of using this kind of surrogate models is their speed which makes them suitable for optimization loops.

More generally, assessing in a comparative way new emerging methods would be both interesting and helpful in view of e.g. a categorization of strong and weak points as well as limits of validity. To this end, the TEAM Workshop problems repository [10] offers a wide choice of benchmarks, with a historical series of results obtained via different methods in different eras of computational electromagnetics.

IV. A PARADIGMATIC ARCHITECTURE: GENERATIVE ADVERSARIAL NETWORK

A generative adversarial network (GAN) is another extremely interesting paradigm, composed of two networks, i.e. a generator (G) and a discriminator (D). Given a training data set characterized by a probability density function PDF_t , network D is trained in such a way to maximize the probability of correctly classifying real data and artificial data; in turn, network G is trained in such a way to maximize the probability of D to classify artificial data as real data.

The final goal is to synthesize G as a good estimator of PDF_t : as a remarkable result, it can be mathematically proven that the goal is achieved when the probability density function featuring the generator PDF_g is equal to the one of the data set, i.e. $PDF_g = PDF_t$ [8]. At this point in time, G is able to generate an optimal solution in an inexpensive way.

In particular, a GAN can be trained exploiting the data derived from a multi-objective optimization problem solved with NSGA-II algorithm; in other words, a subset of the solutions originated during the optimization procedure form the data set. This way, both dominated and non-dominated solutions are supplied to network D: the former are labelled as *real* while the latter are labelled as *fake*. As far as the G network is concerned, the input is Gaussian white noise sampled from a latent variable space, while the output is an (ideally) non-dominated solution in the design space. This puts the ground for densifying the Pareto optimal set, originally approximated by means of the preliminary optimization procedure, at a zero marginal cost. Eventually, a feedforward neural network (multilayer perceptron) is trained using the

same data set of the GAN, in order to approximate the forward problem solution. This allows to estimate in an inexpensive way the objectives function values corresponding to the non-dominated solutions generated by network G , without the need of additional field analyses. In Fig. 3 the architecture of the implemented GAN is shown.

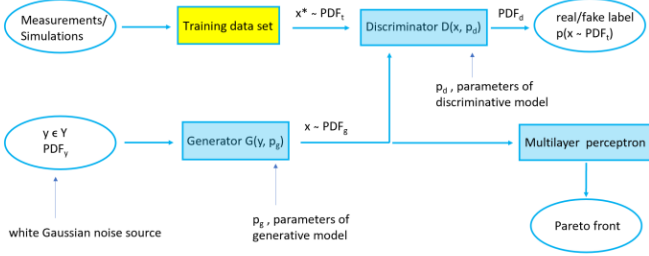


Fig. 3. Architecture of the implemented GAN.

As far as the architecture of the two networks is concerned, the Generator is implemented with a CNN characterized by 13 layers, in which 3 blocks can be highlighted. Each block is composed of a transposed convolutional 2D layer, a batch normalization layer and a Rectified Linear Unit (ReLU) function,

V. TEAM WORKSHOP PROBLEM 35

The multi-objective Pareto optimization of a solenoid winding is considered as the case study according to the TEAM problem 35 [9]. Specifically, the winding is composed of twenty series-connected circular turns, with the width of each turn $w = 1$ mm and the height $h = 1.5$ mm, carrying a current of 3 A (corresponding to a current density of 2 A mm^{-2}). Assuming a cylindrical symmetry, only ten turns are considered in the model (Fig. 4).

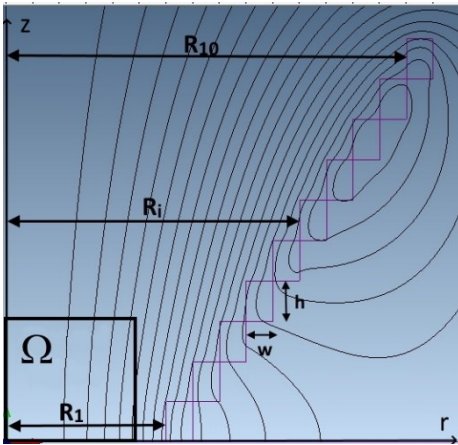


Fig. 4. Geometry of the winding, control region and flux lines.

The underlying magnetic field problem is numerically solved using a finite-element axisymmetric model subject to the following boundary conditions: tangential flux lines at $r=0$ and normal flux lines at $z=0$. The magnetic field pattern corresponding to a given geometry of the winding is shown in Fig. 4.

The optimization problem consists of identifying the distribution of turn radii such that the field uniformity in a controlled subregion Ω around the solenoid axis is maximum and, simultaneously, the power loss in the winding is minimum.

To this end, the radii $R_1, \dots, R_i, \dots, R_{10}$ of the ten turns (Fig. 4) are assumed to be the design variables, while the two objective functions can be cast as follows

$$f_1 = \sup_{j=1, n_p} \left| B_j - B_0 \right| \quad (3)$$

with $n_p = 66$ sample points uniformly spaced inside Ω while $B_0 = 2 \text{ mT}$ is the target field, and

$$f_2 = \sum_{i=1}^{n_t} R_i \quad \text{with } n_t=10. \quad (4)$$

In (3), B_j is the z component of B field at the j -th sample point, while B_0 is the relevant prescribed value. In turn, function f_2 is related to the total ohmic resistance - and so power losses - of the winding. Both objective functions must be simultaneously minimised.

The following remark can be put forward: function f_2 is inexpensive to calculate because it can be analytically computed based on the shape of the turn; in contrast, function f_1 needs a field analysis for the given winding geometry. Hence, in order to evaluate the function f_1 in a fast way, a classical feed-forward fitting neural network (FFNN) is implemented. It is composed of 6 layers with [20, 15, 5, 2, 2] neurons, respectively, and one output layer. The activation function used for all layers is the logistic sigmoid activation function, which sets the output in the range [0,1]. The whole set of solutions obtained with NSGA-II method (black dots in Fig. 5), based on FE analyses, is used for training the net, considering the 70% of samples for the training set, 15% for the validation set and 15% for the test set. For training the network the Levenberg-Marquardt method is used. For evaluating the neural network results, true and predicted points are plotted in Fig. 5.

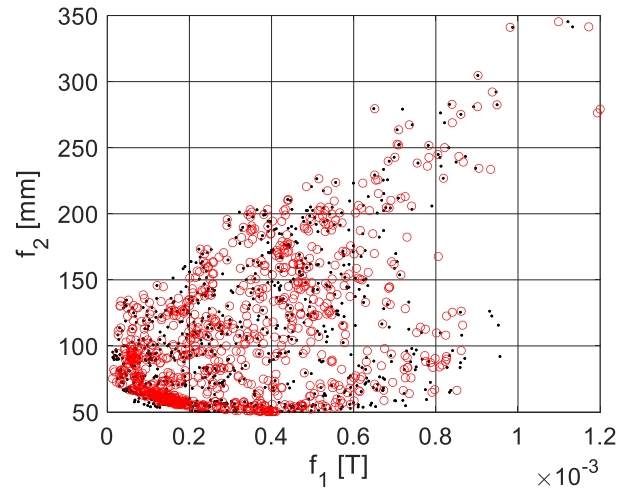


Fig. 5. True points (NSGA-II results): black dots, predicted points (FFNN): red circles.

In turn, in order to train the GAN, a subset of the solutions originated during the optimization process based on NSGA-II algorithm are used, i.e. the Pareto front solutions along with the neighboring solutions.

In this respect, because the number of solutions in the training set substantially affects the convergence of the GAN, a set of solutions larger than the Pareto front subset is used for training the GAN. In particular, a set of solutions close to the Pareto front has been selected. Specifically, considering a single Pareto optimal solution as the reference, a set of the k solutions closest to the reference one, are taken; subsequently, this procedure is repeated for each Pareto optimal solution. These solutions x^* are considered to be close to the Pareto optimal solutions based on the Minkowski's distance metric d_m of order $p = 2$, namely

$$d_m = \left(\sum_{i=1}^n \left| \frac{f_i}{f_{\max_i}} - \frac{\bar{f}_i}{f_{\max_i}} \right|^p \right)^{1/p} \quad (5)$$

with $n = 2$ number of objectives and $p = 2$. According to (5), taking $k = 400$ and discarding the duplicate solutions, a cloud of 624 points is obtained. As can be noted in (5), each component (f_1 and f_2 , respectively) of the distance between two points in the objective function space is normalized by the maximum value f_{\max_i} of the relevant objective function.

Accordingly, the GAN was trained for 5,000 epochs with 624 samples. both Generator and Discriminator explored different configurations - in particular this happens in the first half of the training history - and eventually a trend towards convergence is observed. In particular, the Discriminator score is around 0.5 and it is a good value, because it means that the Discriminator is no longer able to detect the difference between real and generated vectors.

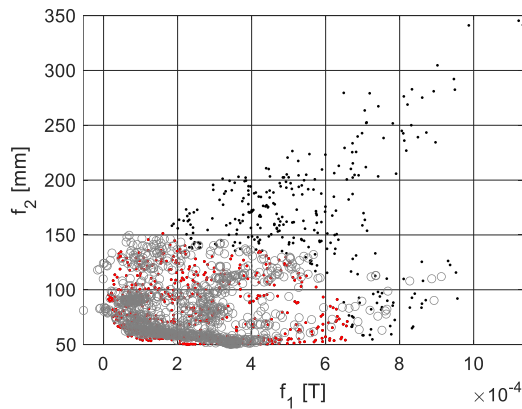


Fig. 6. True points (NSGA-II results): black dots, predicted points (FFNN): red circles.

After the GAN converged, the trained Generator is used by itself for generating 1,000 vectors, starting from 1,000 vectors of random numbers. Each generated vector belongs to the design space and contains the set of 10 radii defining the solenoid geometry. These vectors are then evaluated in the objective space with the FFNN previously trained for the computation of the f_1 function, while f_2 function is analytically computed. The results are shown in Fig. 6; in particular the

grey circles are the points generated by the Generator, mapped into the objective space through the FFNN. The nice feature is that solutions corresponding to the grey circles are inexpensively generated; this way, thousands of points can be generated with no extra cost for discovering new solutions. Consequently, the objective space and in particular the region of interest (red dots) is enriched of points and arbitrarily densified. This makes it possible to obtain thousands of solutions, i.e. different “good” geometries, which all belong to the area of interest at zero marginal cost, a feature that might help the decision making process [7].

VI. CONCLUSION

According to the recent literature, CNNs exhibit good capabilities to work as surrogate models in electromagnetics, mimicking both field analysis and field synthesis by means of equation-less models. Apparently, there are two schools of thought: the one that says that the real world is represented through models based on physical laws, *i.e.* on equations from the theories of Newton, Maxwell, Laplace, Schrödinger, just to mention a few, and the one according to which everything can be reconstructed through data analysis, the so-called big data. This is the data science that is driving tremendous advances in machine learning and neural networks. These are two approaches that are only seemingly alternative: in fact, mathematical models use data and can generate solutions which in turn can represent new data useful for training neural networks. The latter, in turn, can provide substantial help in enriching mathematical models, for example in describing the behavior of biological processes which are experimentally observed at the microscopic and nanoscopic domain. These are just a few examples of how synergy can be created between these two vistas. More likely than not, in the next few years we will see a substantial progress towards algorithms for a new world in computational electromagnetism.

REFERENCES

- [1] I. Goodfellow, Y. Bengio, A. Courville: Deep Learning. The MIT Press, 2016.
- [2] Khan A., Ghorbanian V., Lowther D., “Deep Learning for Magnetic Field Estimation”, IEEE Trans. on Magnetics. 2019;55(6):7202304.
- [3] Khan A., Mohammadi M.H., Ghorbanian V., Lowther D., “Efficiency Map Prediction of Motor Drives Using Deep Learning”, IEEE Trans. on Magnetics. 2020;56(3):7511504.
- [4] S. Barmada, N. Fontana, A. Formisano, D. Thomopoulos, M. Tucci, “A Machine Learning Surrogate Model for Topology Optimization”, Proc. CEFC-2020, November 16-18, 2020.
- [5] M. Baldan, P. Di Barba, B. Nacke, “Magnetic properties identification by using a bi-objective optimal multi-fidelity neural network”. IEEE Transactions on Magnetics, vol.57, no.6, 2021.
- [6] P. Di Barba, M.E. Mognaschi, S. Wiak, “CNN-based Surrogate Model of the Electrostatic Field for MEMS: a Multi-fidelity Approach”, Proc. ISEF-2021, September 20-23, 2021.
- [7] M. Baldan, P. Di Barba, “Discovering Pareto-optimal magnetic-design solutions via GANs”, Proc. COMPUMAG-2021, January 16-20, 2022.
- [8] I. Goodfellow, et al., “Generative Adversarial Networks”, in: Advances in Neural Information Processing Systems, 2014.
- [9] Di Barba, P., Mognaschi, M.E., Lowther, D.A., Sykulski, J.K., “A benchmark TEAM problem for multi-objective Pareto optimization of electromagnetic devices”, IEEE Trans. on Magnetics, 54 (3), 2018.
- [10] <https://www.compumag.org/wp/team>