2015 ACEMP - OPTIM - ELECTROMOTION joint conference

Optimization in Design of Electric Machines: Methodology and Workflow

Stjepan Stipetic and Werner Miebach
Motor Engineer s.a.r.l.
Biviers, France
stipetic@motor-engineer.net,
miebach@motor-engineer.net

Damir Zarko
University of Zagreb Faculty of Engineering and Computing
Department of Electric Machines, Drives and Automation
Zagreb, Croatia
damir.zarko@fer.hr

***Abstract*—This paper presents an overview of methodology for usage of mathematical optimization procedures e.g. optimization algorithms to achieve optimal design of an electrical machine. A special care must be taken in order to handle parameter definition, boundary constraints, constraint functions, model feasibility and computationally expensive calculation such as finite element analysis. Variants of workflows using different approaches to optimization are presented.**

***Index Terms*—electric machines, optimization, optimal design, differential evolution**

I. Introduction

Optimization is a very popular term in modern design of electrical machines and devices in general. Due to the everlasting competition in the world markets, increased cost of electrical energy and pressures for its conservation, design optimization of electrical machines becomes more and more interesting and important. In other words, mathematical optimization helps designers to push the existing invisible design boundaries while using available materials and technology. The objective of the optimization process is usually to minimize either the initial cost of the machine or its lifetime cost including the cost of lost energy. Other objectives such as mass minimization or efficiency maximization may be also appropriate in some situations [1].

It is very important to differ the exact mathematical optimization procedure from the mere parameter variation. Many machine designers and scientists will use the word "optimization" without being aware of its true background. One can quite often find papers presented on conferences proclaiming optimal design, but actually describing sensitivity analysis done on a single problem by varying one or few parameters with heavy conclusion drawn at the end. This can be explained and understood through the words of prof. TJE Miller (who is certainly aware of the true optimization) [2]: "To a WISE engineer, *optimal design* means a compromise between conflicting factors, often producing an imperfect result from optimistic aspirations. Who would use a title such as *Compromises in the design of...*? Optimal sounds better.... This paper is written at a basic engineering level and makes no attempt to apply sophisticated optimization theory."

This paper offers an overview of methodology for usage of mathematical optimization procedures (techniques) to achieve optimal design of an electrical machine. A thorough literature overview is given through definition of the terms like parameters (variables), boundary constraints, constraint functions, model feasibility and stopping criterion. Different workflows used in optimization applications are explained.

II. Methodology

Most of the requirements for electrical machine design are in contradiction to each other (reduction in volume or mass, increase in efficiency etc.). Therefore finding a design that will satisfy all of them can be an overwhelming task due to a large number of parameters whose effects on the motor performance and quality of the design are strongly coupled. There is an obvious need for a systematic approach to decision making based on an iterative scheme that would gradually lead to an optimal motor design which satisfies all the constraints imposed upon it and still fulfills its main task to produce torque.

The design of a machine can be described by a vector $\vec{x}$ of D variables stating dimensions, non-dimensional parameters, current densities, types of materials used etc. The design is subject to a set of m constraints which may include specifications arising from international technical standards and electromagnetic, thermal, mechanical or manufacturing constraints. The goal of the design optimization is to make a chosen objective function $f(\vec{x})$ reach its minimum or maximum value while keeping other technical indices within acceptable ranges [1].

The general multi-objective optimization problem can be mathematically defined as:

Find the vector of parameters

$$\vec{x} = [x_1, x_2, \ldots, x_D], \ \vec{x} \in R^D$$

subject to D parameter constraints (boundary constraints)

$$x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, \ldots, D$$

and subject to m inequality constraints (constraint functions)

$$g_j(\vec{x}) \leq 0, \quad j = 1, \ldots, m$$

which will minimize the vector function

$$f(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x})]$$

The result of the single-objective optimization is a single vector whose parameters completely define a single machine design, and the result of the multi-objective optimization is a population of non-dominant solutions which belong to a *Pareto optimal set*. Since none of the vectors dominate, they are all equally good solutions which provide invaluable insight to the decision maker on how to choose the best design to satisfy the performance criteria.

It can be concluded that optimization techniques in general require different type of choices, such as the following [3]:

1) type of optimization algorithm;
2) optimization variables (parameters), their type and constraints;
3) constraint functions;
4) objective function(s);
5) parametrized model of the problem to be solved;

A. Optimization algorithm

There is a wide variety of optimization techniques which can be used for motor design. Some of the techniques require providing a feasible starting point for the search process to begin. Finding a feasible starting point that would lead to a global minimum of the objective function is an almost impossible task. The complexity of electric machine design is such that explicit methods of optimization, such as those dependent on making

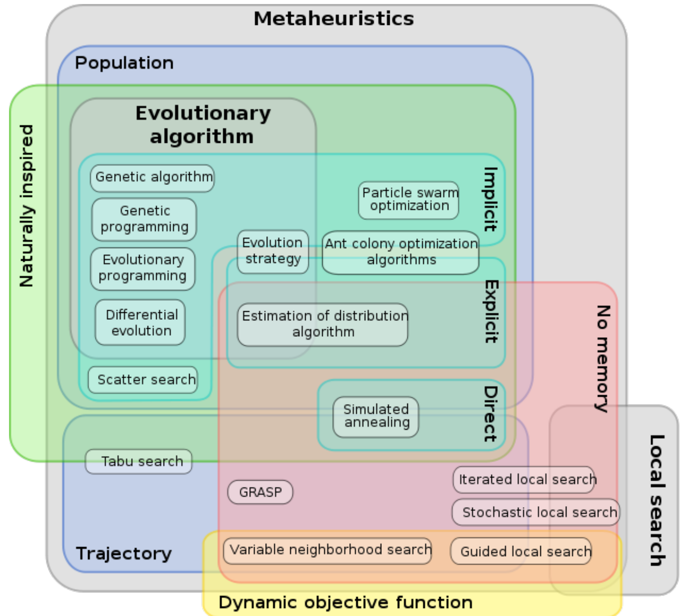

Fig. 1. Overview of metaheuristic algorithms, source: Wikipedia

certain derivatives equal to zero, are not feasible [1]. The optimization techniques which do not require a specific starting point represent a more flexible and attractive approach. Therefore mostly metaheuristic techniques capable of solving global optimization problems subject to non-linear constraint are used (Fig. 1). Metaheuristic algorithms, however, do not strictly mathematically guarantee that the optimal solutions are ever found, but there is a high possibility that a near optimal solution will be determined [4]. From designer and engineering point of view, **it is a global optimum**.

One of the most promising algorithms from the class of evolutionary algorithms widely used in the field of electric machines is Differential Evolution (DE) [5]–[11] first introduced by Price and Storn [12] in 1995. Several authors have tested the algorithm using some well known and difficult numerical test problems [13], [14] and showed that it was capable of outperforming other well known optimization algorithms. The algorithm was later improved and named Generalized Differential Evolution (GDE) (extended DE for constrained multi-objective optimization) by Lampinen [15]–[17].

In short, DE method works on a population (generation) which is a set of NP individuals (members), where each individual presents one machine design. Initial population is randomly initialized inside the boundary constraints. Candidate (trial) population is obtained by crossover and mutation processes from the existing population. Next generation is obtained by comparing the existing and canditate population by choosing members that satisfy boundary functions and/or have better objective function.

Variety of other algorithms is used in electric machine design optimization: Genetic Algorithm (GA) [18]–[21], Particle Swarm Optimization (PSO) [22]–[25], Simulated Annealing (SA) [26] etc. Authors in [26] compared GA, SA and DE on design optimization of permanent magnet motor and authors in [27] compared DE, GA and PSO on design optimization of microstrip antennas. Both groups agree that DE performance is the best. In [28], [29] PSO and GA were compared and PSO was found computationally more effective with slightly better objective function value reached. In [29], it is shown how PSO performs better then GA so some authors decided to use Hybrid GA-PSO method [30].

Any ranking attempt between the different algorithms is not truly appropriate since the performance is problem- and case-dependent and from engineering point of view, satisfying in all cases. Nevertheless, authors mostly agree that DE achieves the best fitness values, i.e. the minimum objective function value, usually with smaller number of evaluations. The second best-performing algorithm is often PSO.

B. *Definition of variables*

The variables of the optimization algorithm that compose the vector $\vec{x}$ are geometrical and other quantities that describe the outlook of the model or are derived from them. The most influential variables on the target functions are usually identified using a sensitivity analysis tool [31], [32]. All the variables are constrained in prescribed intervals, so called *boundary constraints*, which define the *search space* or the *design space*. After reproduction in optimization algorithm, some variables of the newly created candidate vectors may fall out of boundaries. These variables can be "repaired" using random values generated within the feasible range using the scheme proposed in [15].

Some authors [3], [4], [33] used model parameters (stator bore diameter, depth of stator slot etc.) directly as optimization variables while some authors [1], [5], [6], [8], [34], [35] used ratios of model parameters as optimization variables. It is better practice to choose variables which are given as non-dimensional ratios of related geometrical parameters, for example ratio of slot depth to difference between stator outer radius and stator inner radius, ratio of stator inner diameter to stator outer diameter, ratio of tooth (or slot) width to slot pitch, ratio of magnet length to airgap length, magnet pole arc relative to the pole pitch. Some other geometrical parameters can just be considered as optimization variables directly within prescribed interval (outer stator diameter, stack length, slot current density) or can be used as relative parameters (outer stator diameter to maximum outer stator diameter). The usage of dimensionless variables, particularly as a ratio of the pole pitch, enables for example the extension of results from the studied configuration to other motors with different number of poles [35].

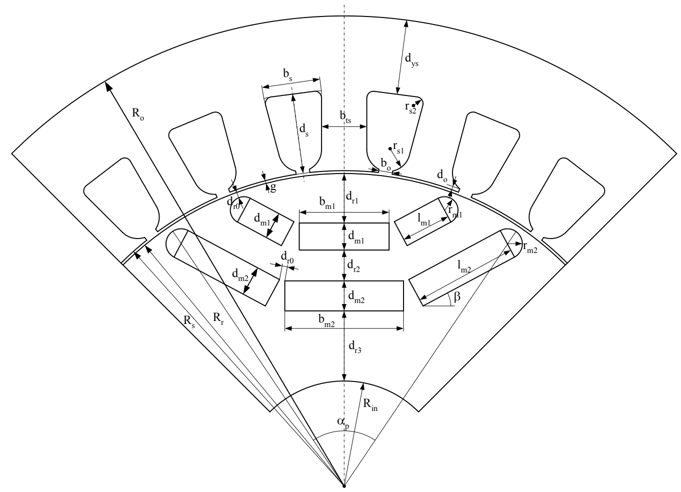

Fig. 2. IPM motor topology with geometric design parameters

An example of parametrized IPM motor topology with geometrical design parameters is given in Fig. 2 and the corresponding optimization variables are listed in Table I.

The Differential Evolution algorithm, for example, assumes that parameters in the population are continuous real numbers. However, in the motor design some of the parameters, e.g. the number of pole pairs, can have only integer values. The example of discrete variables are the standard wire diameters which can be used for the armature winding. The main difference between integer and discrete variables is that although they both have a discrete nature, only discrete variables can assume floating point values. The discrete variables can also be unevenly spaced. The original DE algorithm was modified by Lampinen and Zelinka [15] in order to include mixed-integer-dicrete parameters. Similar approach exists for GA [19].

C. *Feasibility*

The term *feasibility* is usually related to the solution and it denotes that the solution satisfies all the given constraints. In other words, the region enclosed by $g_i(\vec{x}) = 0$

TABLE I
OPTIMIZATION VARIABLES

	Variable	Limits
1.	Ratio of stator inner diameter to outer diameter	$0.45 < D_s / D_o < 0.75$
2.	Ratio of stack length to maximum stack length	$0.6 < l_a / l_{a0} < 1$
3.	Ratio of yoke thickness to difference between stator outer and inner radius	$0.2 < 2d_{ys} / (D_o - D_s) < 0.6$
4.	Permanent magnet data	Table input
5.	Number of pole pairs	$p = 2,3,4,5,6$
6.	Ratio of tooth width to slot pitch at D_S	$0.3 < b_{ts} / \tau_s < 0.7$
7.	Ratio of total cavity to total rotor depth	$0.1 < \lambda_m < 0.5$
8.	Percentage of total cavity depth for inner cavity	$0.25 < \lambda_{h2} < 0.7$
9.	Percentage of total rotor depth for the outermost rotor core section	$0.2 < \lambda_{md1} < 0.6$
10.	Percentage of total rotor depth for middle rotor core section	$0.1 < \lambda_{md2} < 0.4$
11.	Angular span of the inner cavity relative to the pole pitch	$0.6 < \lambda_p < 0.95$
12.	The angle of the slanted magnet	$0.5 < \beta / \beta_0 < 1$

is known as the feasible region. There is another type of feasibility, so called ”geometrical or model feasibility”. Geometrically feasible model is valid for solving: there are no overlapping edges, negative lengths or non-conventional geometric relations that will inevitably produce errors after the start of the solver. In order to avoid drawing and creation of such non-valid model, a procedure to determine the geometrical feasibility can be performed inside the optimization algorithm. Each candidate vector is checked for geometrical feasibility. If the parameters do not pass the feasibility check, the complete set of parameters is randomly initialized again until the geometrical feasibility is achieved.

For example, the parameters describing the stator slot of the electrical machine (tooth width, slot depth, slot corner radius, tooth tip angle, tooth tip length, slot opening) combined with parameters that define stator (stator inner diameter, stator outer diameter) can sometimes result in slots being drawn over the stator outer diameter line. In order to avoid programming of the additional mathematical relations or inclusion of additional parameters, a geometrical feasibility condition to check the stator yoke thickness can be included. Carefully chosen parameters defined as dimensionless ratios, along with the boundaries given on the parameter values, can ensure that feasibility is satisfied in the majority of the cases.

Although geometrical feasibility check sometimes cannot be avoided and is absolutely necessary to avoid solving failure (e.g. to avoid restart of the algorithm), each failed feasibility check will automatically destroy the mutated candidate vector and will reinitialize it randomly.

D. Constraint functions

Constraint functions normally arise from different electromagnetic, thermal, mechanical, manufacturing, economic or standard limits such as maximum flux density in stator tooh, maximum PM temperature, maximum stress in the IPM rotor bridge, minimum dimensions of magnet plate, maximum cost of the active material, maximum noise etc.

Traditional approach for handling constraint functions uses penalty functions to penalize the solutions which violate constraints. This principle is implemented in the form of weighted sums which modifies each objective function. Despite the popularity of penalty functions, they have several drawbacks of which the main one is the requirement for careful fine tuning of the penalty factors that accurately estimates the degree of penalization to be applied as to approach efficiently the feasible region. In addition, this method suffers from problems related to poor choice of the weight factors which can affect the convergence.

Widely accepted technique to efficiently handle boundary functions in DE is Lampinen’s criterion [36]. According to Fig. 3, a trial vector is selected for the new generation if [37]

- it satisfies all constraints and has a lower or equal objective function value than the design from the current generation, or
- it satisfies all constraints, while the current vector does not, or
- neither the trial nor the current vector satisfy the constraints, however, the trial vector does not violate any constraint more than the current vector.

The main advantages of this approach are: it forces the selection towards feasible regions where constraints are satisfied thus resulting in faster convergence, it saves time since no evaluation of the objective function occurs if constraints are violated.

Furthermore, if any of the boundary constraints is violated, other boundary constraints are not even calculated at all, which is especially interesting for computationally expensive calculation. For example, if boundary function g_1 contains purely analytical and fast calculation (for example calculation of the linear current density in stator bore), function g_2 which calculates load point with magnetostatic FEA calculation and function g_3 which calculates demagnetization with transient FEA calculation is not run if g_1 does not satisfy, which saves overall optimization time.

As advised in [1], constraint functions may be of widely differing magnitudes. Such differences may make some boundary functions more sensitive than others in the optimization process, possibly leading to failures to converge. For this reason, it is advisable to normalize all functions by choosing suitable base values and expressing all quantities in per unit of those values.

A good base value is in fact the minimum or maximum value of the constraint imposed. For example, for the minimum efficiency boundary the constraint function would be

$$\eta_{con}(\vec{x}) = 1 - \frac{\eta(\vec{x})}{\eta_{min}}$$

and for the maximum tooth flux density

$$B_{st,con}(\vec{x}) = \frac{B_{st}(\vec{x})}{B_{st,max}} - 1$$

where $\eta(\vec{x})$ and $B_{st}(\vec{x})$ are the efficiency and the tooth flux density of the motor design defined by vector $\vec{x}$, η_{min} is the minimum allowed efficiency and $B_{st,max}$ is the maximum allowed tooth flux density.

E. Objective functions

The terms *cost function*, *fitness function* are synonyms for the objective function. Various authors use various objective functions, it all depends on the field of application. Some of the common objective functions in single-objective optimizations are to minimize the material cost [1], [33] or to maximize torque per volume [8], but sometimes conflicting objectives that would normally be suitable for multi-objective optimizatin are reduced to single function, for example to minimize cost/efficiency [32].

The optimization of maximum torque per volume is rather common, but it can be seen that higher torque requires more space, and maximum torque per volume designs are heavier. Therefore, they are not suitable for weight sensitive applications, such as wind power generation, where the system performance is significantly influenced by the mass on the top of tower. In this case, the maximum torque per weight is preferred [38].

When dealing with multi-objective optimization, the use of Pareto-optimality-based algorithms is particularly adapted to the industrial framework: they do not lead to a single and definitive optimal solution, but to a large set of Pareto-optimal solutions, so a degree of freedom is still available at the end of optimization process [19]. A special method for studying Pareto fronts is presented in [39]. Some authors state that a good optimization tool should allow users to prioritize one or more objective functions based on their requirements [40].

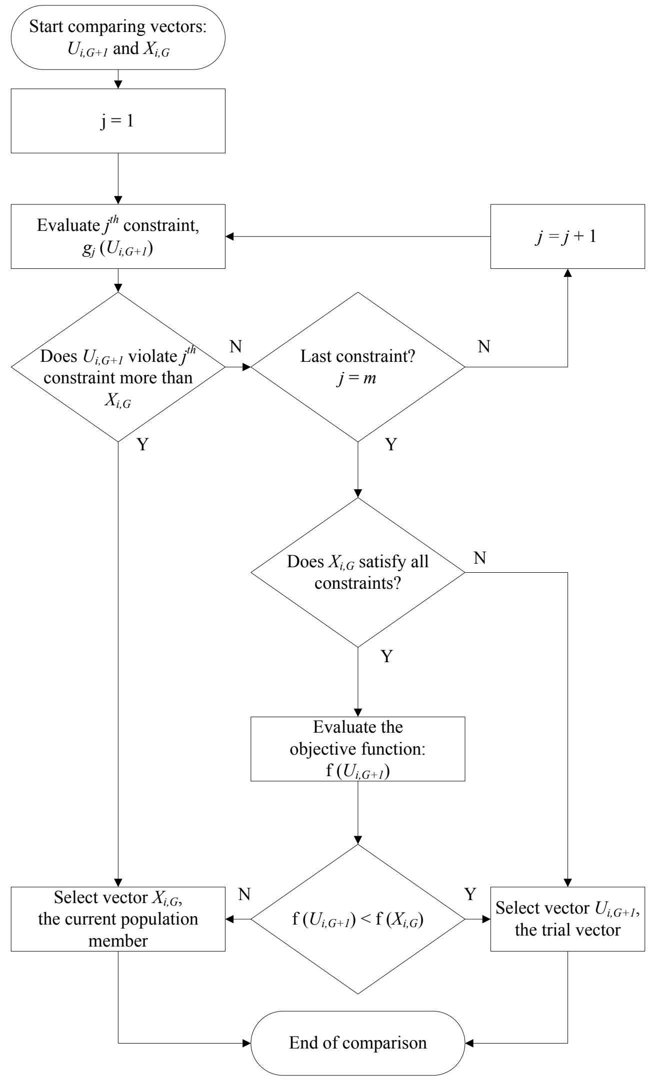

Fig. 3. Lampinen's criterion for trial vector [36]

F. Stopping criterion

Unlike in a deterministic optimization, in stochastic optimization next generation does not always improve in the terms the best objective value (although average objective value of the generation may be better). It is very important to decide when to terminate the search process. A straightforward practical solution is to terminate the search when the number of generations has reached a given maximum value. However, as it may be difficult to establish this value through an exact mathematical process, an experience based or trial-and-error approach is needed [35]. A better approach to terminate the algorithm is an adaptive stopping criterion, especially with computationally intensive computation [41].

III. Workflow for FEA based optimization

Advanced users mostly tend to write their own optimization application source code using available existing optimization algorithms and custom in-house models. By doing this, the total control over the process is available. Typical workflow inside the application is shown in Fig. 4. Program starts with problem definition (boundaries, objectives, model type) and a preset of constant model parameters (slots, poles, winding, etc.). After entering the optimization loop, the following steps are performed iteratively:

1) optimization algorithm generates variables (parameters)
2) the variables are converted to model parameters
3) model is setup (drawn)
4) model is solved
5) model performance is extracted (post-processing)
6) constraint functions and objective function values are calculated
7) constraints and objectives are passed back to optimization algorithm

Eventually, an optimal solution is obtained.

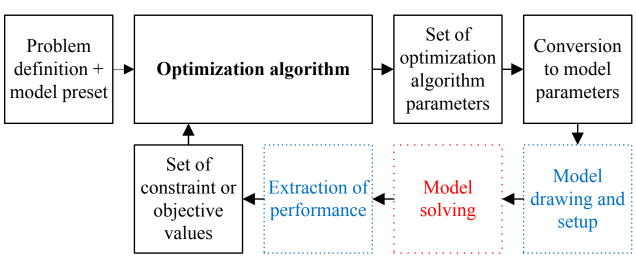

Fig. 4. Workflow for FEA based optimization

Some Universities or research institutions have their own modelling infrastructure (FEA source code). A source code typically exists for all the boxes in Fig. 4, typical platform is Matlab, Pytohn or C. The performance is surely the fastest in this case.

Very often the FEA source code is not available in which case freeware or commercial package must be used. The same workflow is valid, but the red ”model solving box” is realized by linking the source code with the FEA software via built-in links (for example Matlab - COMSOL Multiphysics) or via certain scripting interface (for example Visual Basic - Infolytica MagNET or Python - Infolytica MagNET via *ActiveX*). Since most of the FEA software is not specially dedicated to electric machines, a model is created (drawn or adjusted) through the scripting interface. This procedure can be quite tedious and complicated because in most general sense it is based on calculation of the coordinates of vertices and drawing the lines and arcs, defining regions, boundary conditions. Preset parametrized model can sometimes be used if available. Also, most of the motor performance extraction (efficiency calculation, inductances etc.) is done outside of the FEA package, i.e. within the optimization application, even if post-processing is available.

Furthermore, special software dedicated to the design of electrical machines (such as MotorCAD or SPEED) can be used, not just as ”model solver”, but also ”model drawer” and ”performance extractor” - two blue squares in Fig. 4. This software is also accessed and controlled via scripting interface from the optimization application. Since model design is template based, a preset model with fixed parameters, which is supplied with variable optimization parameters, can be used. In this case, most of the performance extraction is done inside the solving software.

Some commercial FEA packages have optimization add-ons to perform optimization tasks. The add-on is in fast graphical user interface (GUI) containing optimization algorithm and all additional code and mathematics required to control parametrized model solver. A good example for this is Infolytica OptiNET as optimization add-on for Infolytica MagNET FEA package. Some other examples include dedicated and powerful optimizing packages such as HEEDS (can run Solidworks Simulation, Star CCM+, SPEED software) or OPTIY (can run various FEA packages such as JMAG, ANSYS, Infolytica). A typical workflow is shown in Fig. 5.

In some special cases, user interface that utilizes parametrized model solver and a separate optimization package, which acts like a black box, are used. A workflow is shown in Fig. 6.

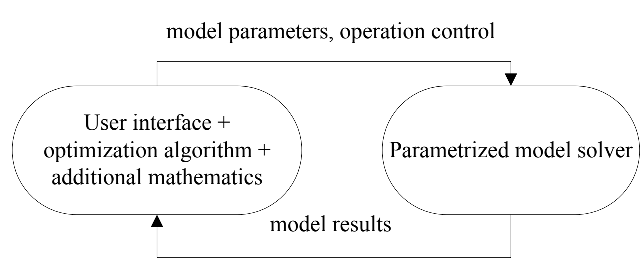

Fig. 5. Workflow for optimization add-ons

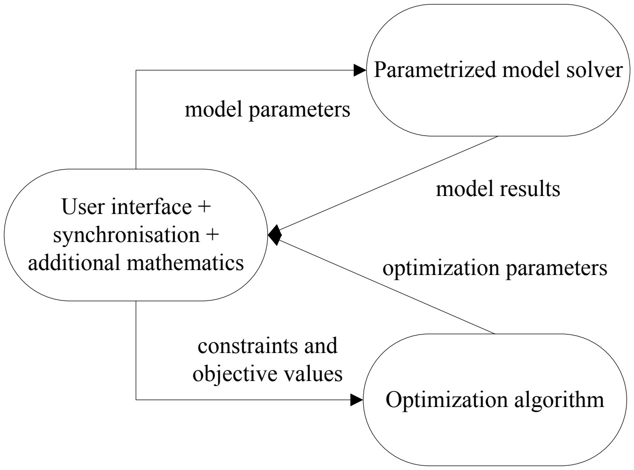

Fig. 6. Workflow for separate optimization algorithm

IV. Conclusion

Optimization slowly becomes important and unavoidable part of the modern electrical machines design process. Very often design engineers primarily rely on their experience to obtain a machine design suited for some particular purpose. This ”classical” approach guarantees only that a fully functional design will be accomplished, but it does not ensure that this design will be accomplished with minimum amount of material used or that it will consume a minimum amount of energy in its exploitation or that its initial cost will be the smallest possible. At the same time these are very important factors that need to be considered to make a machine more competitive on the market.

If properly utilized, the optimization will lead to the design that satisfies all imposed requirements, but is also optimal in a certain sense, depending on the feature on which machine designer puts the emphasis (mass, volume, efficiency, cost or their combination).

Further research in the field of electric machine optimization will include pre-optimization sensitivity analysis, and improvements in existing algorithms, non-linear surrogate models and robustness of the solutions.

Acknowledgment

This paper is part of the research in ADvanced Electric Powertrain Technology (ADEPT) project which is an EU funded Marie Curie ITN project, grant number 607361. Stjepan Stipetic is currently Marie Curie PostDoc Fellow at Motor Engineer s.a.r.l, Biviers, France.

References

[1] X. Liu and G. Slemon, “An Improved Method of Optimization for Electrical Machines,” *IEEE Transactions on Energy Conversion*, vol. 6, no. 3, pp. 492–496, Sep 1991.

[2] T. Miller, “Optimal Design of Switched Reluctance Motors,” *IEEE Transactions on Industrial Electronics*, vol. 49, no. 1, pp. 15–27, Feb 2002.

[3] J. Legranger, G. Friedrich, S. Vivier, and J. Mipo, “Combination of Finite-Element and Analytical Models in the Optimal Multidomain Design of Machines: Application to an Interior Permanent-Magnet Starter Generator,” *IEEE Transactions on Industry Applications*, vol. 46, no. 1, pp. 232–239, Jan 2010.

[4] S. B. Andersen and I. F. Santos, “Evolution strategies and multi-objective optimization of permanent magnet motor,” *Applied Soft Computing*, vol. 12, no. 2, pp. 778–792, 2012.

[5] D. Zarko, D. Ban, and T. Lipo, “Design Optimization of Interior Permanent Magnet (IPM) Motors with Maximized Torque Output in the Entire Speed Range,” in *European Conference on Power Electronics and Applications*, 2005, pp. 10 pp.–P.10.

[6] W. Ouyang, D. Zarko, and T. Lipo, “Permanent Magnet Machine Design Practice and Optimization,” in *Conference Record of the 2006 IEEE Industry Applications Conference*, vol. 4, Oct 2006, pp. 1905–1911.

[7] G. Sizov, D. Ionel, and N. Demerdash, “Multi-objective optimization of PM AC machines using computationally efficient - FEA and differential evolution,” in *IEEE International Electric Machines Drives Conference (IEMDC)*, May 2011, pp. 1528–1533.

[8] D. Zarko and S. Stipetic, “Criteria For Optimal Design Of Interior Permanent Magnet Motor Series,” in *XXth International Conference On Electrical Machines (ICEM), 2012*, Sept 2012, pp. 1242–1249.

[9] Y. Duan and D. Ionel, “A Review of Recent Developments in Electrical Machine Design Optimization Methods With a Permanent-Magnet Synchronous Motor Benchmark Study,” *IEEE Transactions on Industry Applications*, vol. 49, no. 3, pp. 1268–1275, May 2013.

[10] P. Zhang, D. Ionel, and N. Demerdash, “Saliency Ratio and Power Factor of IPM Motors Optimally Designed for High Efficiency and Low Cost Objectives,” in *IEEE Energy Conversion Congress and Exposition (ECCE)*, Sept 2014, pp. 3541–3547.

[11] P. Zhang, G. Sizov, D. Ionel, and N. Demerdash, “Establishing the Relative Merits of Interior and Spoke-Type Permanent Magnet Machines with Ferrite or NdFeB Through Systematic Design Optimization,” *IEEE Transactions on Industry Applications*, vol. PP, no. 99, pp. 1–1, 2015.

[12] R. Storn and K. Price, “Differential Evolution - a Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces,” *Technical Report TR-95-012, ICSI*, March 1995.

[13] J. Lampinen, “Multi-Constrained Nonlinear Optimization by the Differential Evolution Algorithm,” *6th On-line World Conference on Soft Computing in Industrial Applications (WSC6)*, September 2001.

[14] H. A. Abbass, R. Sarker, and C. Newton, “PDE: A Pareto-Frontier Differential Evolution Approach for Mmulti-Objective Optimization Problems,” *Proceedings of the 2001 Congress on Evolutionary Computation*, vol. 2, pp. 971–978, May 2001.

[15] J. Lampinen and I. Zelinka, “Mixed Integer-Discrete-Continuous Optimization By Differential Evolution - Part 1: The Optimization Method,” in *5th International Mendel Conference On Soft Computing*, 1999, pp. 71–76.

[16] S. Kukkonen and J. Lampinen, "Constrained Real-Parameter Optimization with Generalized Differential Evolution," in *IEEE Congress on Evolutionary Computation, 2006.*, 2006, pp. 207–214.

[17] ——, "Performance Assessment of Generalized Differential Evolution 3 With a Given Set of Constrained Multi-Objective Test Problems," in *IEEE Congress on Evolutionary Computation, 2009. CEC '09*, May 2009, pp. 1943–1950.

[18] M. Lukaniszyn, M. JagieLa, and R. Wrobel, "Optimization of Permanent Magnet Shape for Minimum Cogging Torque Using a Genetic Algorithm," *IEEE Transactions on Magnetics*, vol. 40, no. 2, pp. 1228–1231, March 2004.

[19] J. Le Besnerais, V. Lanfranchi, M. Hecquet, and P. Brochet, "Multiobjective Optimization of Induction Machines Including Mixed Variables and Noise Minimization," *IEEE Transactions on Magnetics*, vol. 44, no. 6, pp. 1102–1105, June 2008.

[20] G. Pellegrino and F. Cupertino, "FEA-based Multi-Objective Optimization of IPM Motor Design Including Rotor Losses," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, Sept 2010, pp. 3659–3666.

[21] N. Bianchi, D. Durello, and E. Fornasiero, "Multi-Objective Optimization of a PM Assisted Synchronous Reluctance Machine, Including Torque and Sensorless Detection Capability," in *6th IET International Conference on Power Electronics, Machines and Drives (PEMD 2012)*, March 2012, pp. 1–6.

[22] Y. Duan, R. Harley, and T. Habetler, "Multi-objective Design Optimization of Surface Mount Permanent Magnet Machine With Particle Swarm Intelligence," in *IEEE Swarm Intelligence Symposium*, Sept 2008, pp. 1–5.

[23] R. Wrobel and P. Mellor, "Particle Swarm Optimisation for the Design of Brushless Permanent Magnet Machines," in *Conference Record of the 2006 IEEE Industry Applications Conference*, vol. 4, Oct 2006, pp. 1891–1897.

[24] D. Zhang, C. S. Park, and C. S. Koh, "A New Optimal Design Method of Rotor Slot of Three-Phase Squirrel Cage Induction Motor for NEMA Class D Speed-Torque Characteristic Using Multi-Objective Optimization Algorithm," *IEEE Transactions on Magnetics*, vol. 48, no. 2, pp. 879–882, Feb 2012.

[25] C. Ma and L. Qu, "Multiobjective Optimization of Switched Reluctance Motors Based on Design of Experiments and Particle Swarm Optimization," *IEEE Transactions on Energy Conversion*, vol. PP, no. 99, pp. 1–10, 2015.

[26] M. Mutluer and O. Bilgin, "Comparison of Stochastic Optimization Methods for Design Optimization of Permanent Magnet Synchronous Motor," *Neural Computing and Applications*, vol. 21, no. 8, pp. 2049–2056, 2012.

[27] A. Deb, B. Gupta, and J. Roy, "Performance comparison of Differential Evolution, Genetic Algorithm and Particle Swarm Optimization in Impedance Matching of Aperture Coupled Microstrip Antennas," in *11th Mediterranean Microwave Symposium (MMS)*, Sept 2011, pp. 17–20.

[28] Y. Duan, R. Harley, and T. Habetler, "Comparison of Particle Swarm Optimization and Genetic Algorithm in the Design of Permanent Magnet Motors," in *IEEE 6th International Power Electronics and Motion Control Conference, IPEMC '09.*, May 2009, pp. 822–825.

[29] M. Mutluer and O. Bilgin, "Design Optimization of PMSM by Particle Swarm Optimization and Genetic Algorithm," in *International Symposium on Innovations in Intelligent Systems and Applications (INISTA), 2012*, July 2012, pp. 1–4.

[30] A. Sarikhani and O. Mohammed, "Hybrid GA-PSO Multi-Objective Design Optimization of Coupled PM Synchronous Motor-Drive Using Physics-Based Modeling Approach," in *14th Biennial IEEE Conference on Electromagnetic Field Computation (CEFC)*, May 2010, pp. 1–1.

[31] P. Zhang, G. Sizov, D. Ionel, and N. Demerdash, "Design Optimization of Spoke-Type Ferrite Magnet Machines by Combined design of Experiments and Differential Evolution Algorithms," in *IEEE International Electric Machines Drives Conference (IEMDC)*, May 2013, pp. 892–898.

[32] Y. Duan and D. Ionel, "Nonlinear Scaling Rules For Brushless PM Synchronous Machines Based On Optimal Design Studies For A Wide Range Of Power Ratings," *IEEE Transactions On Industry Applications*, vol. 50, no. 2, pp. 1044–1052, March 2014.

[33] R. Ramarathnam, B. Desai, and V. Rao, "A Comparative Study of Minimization Techniques for Optimization of Induction Motor Design," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-92, no. 5, pp. 1448–1454, Sept 1973.

[34] N. Bianchi and S. Bolognani, "Brushless DC Motor Design: an Optimisation Procedure Based on Genetic Algorithms," in *Eighth International Conference on Electrical Machines and Drives*, Sep 1997, pp. 16–20.

[35] Y. Duan and D. Ionel, "Nonlinear Scaling Rules for Brushless PM Synchronous Machines Based on Optimal Design Studies for a Wide Range of Power Ratings," *IEEE Transactions on Industry Applications*, vol. 50, no. 2, pp. 1044–1052, March 2014.

[36] J. Lampinen, "Multi-Constrained Nonlinear Optimization By The Differential Evolution Algorithm," in *6th On-Line World Conference On Soft Computing In Industrial Applications (Wsc6)*, 2001, pp. 1–19.

[37] W. Gong and Z. Cai, "A Multiobjective Differential Evolution Algorithm for Constrained Optimization," in *IEEE Congress on Evolutionary Computation, CEC*, June 2008, pp. 181–188.

[38] W. Chu, Z. Zhu, and J. Chen, "Simplified Analytical Optimization and Comparison of Torque Densities Between Electrically Excited and Permanent-Magnet Machines," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 9, pp. 5000–5011, Sept 2014.

[39] Y. Duan, Q. Sun, and D. Ionel, "Methods for studying the pareto-fronts in multi-objective design optimization problems of electrical machines," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, Sept 2013, pp. 5013–5018.

[40] S. Vun and M. McCulloch, "Optimal design method for large-scale yasa machines," *IEEE Transactions on Energy Conversion*, vol. PP, no. 99, pp. 1–8, 2015.

[41] K. Zielinski and R. Laur, "Stopping Criteria for a Constrained Single-Objective Particle Swarm Optimization Algorithm," *Informatica (Slovenia)*, vol. 31, no. 1, pp. 51–59, 2007.