J. Automation & Systems Engineering 13-2 (2019): 29-43

Regular Paper

Multiobjective optimal design of an in-wheel permanent magnet synchronous motor for an electric vehicle

JASE
Journal of Automation & Systems Engineering

Lassaad Zaaraoui
CES Laboratory, Engineering School
Of Sfax, Tunisia
University of Sfax
lassaad.zaaraoui@gmail.com

Ali Mansouri
Higher School of Applied Sciences
Technology of Gafsa, Tunisia
University of Gafsa
ali.mansouri@isetgf.rnu.tn

Abstract- ***Abstract-This paper presents an optimal design of the geometric dimensions of a synchronous radial flux permanent magnets machine with an external rotor and concentrated windings, designated for an in-wheel electric vehicle motor. To achieve this goal, we applied five state-of-the-art optimization algorithms which are: Cellular Genetic Algorithm for Multi-Objective problems (MOCell), Nondominated Sorting Genetic Algorithm II (NSGA-II), Nondominated Sorting Genetic Algorithm III (NSGA-III), Speed-constrained Multi-Objective PSO (SMPSO), and Multi-Objective Particle Swarm Optimizer (OMOPSO). The magnetic and electrical properties of the machine's analytical model are presented. According to this model, the multiobjective problem is formulated. The sizing optimization is done according to three objectives which are the maximization of the machine efficiency, the minimization of its mass, and the minimization of its torque ripple. It is also characterized by certain specifications in the form of mechanical and magnetic constraints to properly model the machine. Furthermore, this paper deals with an experimental comparison between the used algorithms according to certain criteria to fully characterize the qualities of each of them in the optimization of our real problem.***

Keywords: Electric vehicle, In-wheel motor, optimal sizing, objective functions, constraints, MOCell, NSGA-II, NSGA-III, SMPSO, OMOPSO, Performance metric.

1. INTRODUCTION

Due to recent advances in the field of high quality magnetic materials, Permanent Magnet Synchronous Machines (PMSMs) have become increasingly used in industrial applications, particularly in the automotive field. Indeed, these machines have several advantages over traditional machines, namely: the high power density, the good power factor, the best efficiency, the light weight and the compact structure [1] [2].

The purpose of this work is to design an electromagnetic architecture that will provide a high torque and a high power density while respecting the constraints of weight, simplicity of construction and manufacture costs for an in-wheel motor, which corresponds to the direct integration of a PMSM into the electric vehicle wheels. Indeed, this design matches with the optimization of the machine geometry while respecting certain magnetic and mechanical constraints and with regard to three objective functions which are the maximization of the motor efficiency, the minimization of its mass and the minimization of its ripple torque. To achieve this goal, we use an analytical model as a model for optimization sizing. It provides a set of parameterized equations that describe, as closely as

possible, the operation of the machine. Then, starting from an initial geometry, we use five state-of the-art multiobjective optimization algorithms which three of them are evolutionary algorithms, and the other two algorithms belong to the family of particle swarm optimization.

This work also deals with a comparison between the performances of the used algorithms, from the point of view of Pareto dominance, execution time, diversity of solutions and ability to solve this problem while ensuring a good compromise between those three objectives.

2. MACHINE TOPOLOGY

Several criteria make it possible to distinguish the different topologies of the PMSMs [3], namely: the direction of the magnetic flux (radial, axial or transverse), the rotor position rotor (internal or external), the location of magnets (surface mounted or inserted in the rotor), and the distribution of windings (distributed or concentrated).

The subject topology of the optimal design that will be studied in this work is a three-phase synchronous motor having 18 permanent magnets mounted on the inner surface of an outer rotor and 27 slots housing concentrated windings [4].

This selection is made according to several criteria in order to choose the most suitable structure for an in-wheel motor application. Indeed, it has been shown in [5] that the radial flux machine (RFM) performs slightly better than the single-sided axial flux machine (AFM) with higher torque and lower inductance. In addition, the manufacture of a RFM for an in-wheel motor can be less complicated and less expensive than a double-sided AFM, while the two topologies are very competitive in terms of efficiency and active mass of the machine [6]. Furthermore, the magnetic losses of a transversal flux machine (TFM) are larger than a RFM; add to its complexity of structure and its high manufacturing cost [7]. Therefore, the RFM should be a viable option as an in-wheel motor.

In a RFM, the manufacture of an external rotor machine makes it possible to increase the torque density, minimize the total mass of the machine and make the system more compact [1] [4]. Moreover, it has been demonstrated in [8] that this topology is able to generate good results in terms of increasing the efficiency and minimizing the ripple of the couple compared to an internal rotor machine. Further, it has been found that the performance of the surface mounted magnet machine is slightly better than the interior magnet machine [8].

For the winding arrangement, the double-layer concentrated windings have better properties than the distributed windings. Indeed, the use of this type of winding reduces copper losses, iron losses and the weight of the machine [9]. In addition, this configuration is characterized by its high power density, low cost, simplicity of manufacture and good fault tolerance capability [3] [10].

Finally, the stator of the machine is made with 27 slots to reduce the complexity of the optimization problem. In this case, the number of corresponding poles (18 poles) is chosen so as to maximize the fundamental winding factor [3] [4].

3. ANALYTICAL MODEL OF THE MACHINE

After defining the appropriate machine topology for our application, the next step in the optimization process is the development of the analytical model. This model is a set of parameterized equations that describe, as closely as possible, the operation of the machine

to be studied. Indeed, it determines all the geometric dimensions of the machine, as well as the magnetic and electrical properties.

The major important machine geometric dimensions are illustrated in the Figure 1 [1]. In this work, we give a brief presentation of the model that will be used for the design of the in-wheel motor.

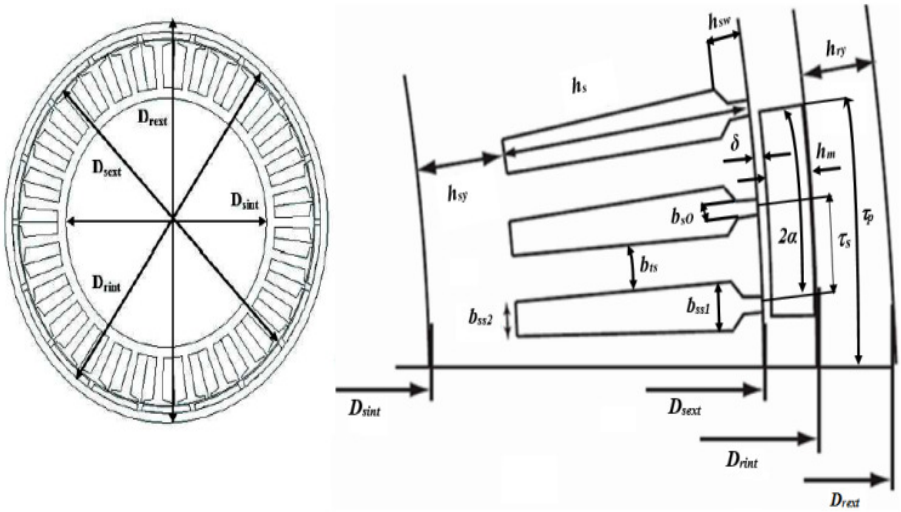

Figure 1 A cross section of the studied machine

3.1 Geometric parameters of the machine

There are 10 important geometrical parameters for the design of the machine. They are given in Table 1 with their ranges of values [4] (the exterior diameter of the rotor D_{rext} is considered equal to 302 mm). Based on these parameters, we can define the relation between the sizing equations of the machine.

Table 1: Major geometrical parameters of the studied machine

Main specification	Symbols	Ranges of values
Interior rotor diameter	D_{rint} (mm)	[282;295]
Air gap length	δ (mm)	[1;3]
Slot wedge height	h_{sw} (mm)	[2;4]
Magnet height	h_m (mm)	[2;6]
Stator slot height	h_s (mm)	[30;50]
Stator teeth width	b_{ts} (mm)	[25;40]
Active machine length	l (mm)	[50;70]
Opening slot factor	k_{open}	[0.2;0.6]
interior Stator diameter	D_{sint} (mm)	[170;195]
Half pole angle	α (°)	[20;22]

3.2 Modeling of magnetic fields

In order to guarantee an optimal dimensioning, we are interested in a magnetic modeling of the studied machine. Indeed, this model makes it possible to evaluate flux and induction values in all the elements of the magnetic circuit such as the air gap and the iron parts.

3.2.1 Air gap induction

The slotted stator causes a non-homogeneous distribution of the air gap magnetic induction. In fact, it decreases at the positions located in opposition to the slot opening. The knowledge of the air gap induction makes it possible to determine the dimensions of the permanent magnets. Indeed, it can be expressed by a relation between the mean air gap induction produced by the magnets, (B_m), and remanent induction of the magnets, (B_r) [4] [11]:

$$B_m = \frac{h_m}{h_m + \left(\mu_r k_c \delta\right)} B_r \tag{1}$$

Where μ_r is the relative permeability of the magnet, and k_c is the Carter factor.

All the same, the amplitude of the fundamental component of the air gap induction (B_g) must be expressed, because it controls the permanent magnet dimensions. Indeed, it is given by the following expression [4]:

$$B_g = \frac{4}{\pi} B_m \sin\alpha \tag{2}$$

3.2.2 Inductions in iron parts

In this section, we are interested in the calculation of the magnetic inductions created by permanent magnets in the different ferromagnetic parts. We successively give the inductions in the stator yoke, $B_{sy,}$ the stator teeth, $B_{st,}$ and in the rotor yoke, B_{ry} [11]. In order to perform this calculation, we assume that the different ferromagnetic parts are not saturated and that the leakage magnetic flux between them is negligible.

$$B_{sy} = \frac{\tau_p B_g}{2\, h_{sy}\, k_{fer}} \tag{3}$$

$$B_{st} = \frac{\left(l + 2\delta\right)\tau_s B_m}{k_{fer}\, l\, b_{ts}} \tag{4}$$

$$B_{ry} = \frac{\tau_m B_m}{2\, h_{ry}\, k_{fer}} \tag{5}$$

Where τ_p is the pole pitch, τ_m is the magnet pitch, h_{sy} is the stator yoke height, k_{fer} is the Stator-core space factor, and h_{ry} is the rotor yoke height.

3.3 Electric modelling

In this section, we present first the induced electromotive force (EMF). Next, the electromagnetic power is outlined. Then, we outline the electromagnetic torque. Finally, the Joule losses and the iron losses are addressed.

3.3.1 Induced electromotive force EMF

The induced electromotive force (EMF) is a voltage in the stator windings, induced by a rotating field produced by the permanent magnets when the rotor rotates at a speed ω_r. The fundamental EMF in phase a ($\hat{E}_{a,1}$), is given by the following equation [11]:

$$\widehat{E}_{a,1} = \pi \sqrt{2}\, f\, N_c\, B_g\, k_{w1}\, \frac{D_{sext}\left(l + 2\,\delta\right)}{p} \tag{6}$$

Where f is the electrical frequency, N_c is the number of windings turns of a phase, and k_{w1} is the fundamental winding factor.

3.3.2 Electromagnetic power

The mean electromagnetic power, P_e, of a PMSM is expressed by the following equation [11]:

$$P_e = m\, \widehat{E}_{a,1}\, \widehat{I}_s \sin\beta \tag{7}$$

Where m is the number of phase, $\hat{I}_s$ is the amplitude of the fundamental stator current of a phase, and β is the electrical angle between the fields produced by the stator current and the magnet.

3.3.3 Electromagnetic torque

The mean electromagnetic torque (T_e) is calculated from the ratio of the mean electromagnetic power to the rotor speed as following [11]:

$$T_e = \frac{P_m}{\omega_m} = \frac{m}{\sqrt{2}} N_c\, B_g\, k_{w1}\, D_{sext}\left(l + 2\,\delta\right)\widehat{I}_s \sin\beta \tag{8}$$

3.3.4 Copper loss

Copper loss (P_{co}) is due to current flowing through the copper windings. It is proportional to the resistance of the copper windings and the square of the current. It can be expressed using the following equation [10]:

$$P_{co} = 3\, I_{ph}^2\, R_{ph} \tag{9}$$

Where I_{ph} is the phase current and R_{ph} is the winding resistance of a phase, which is expressed by the following equation [10]:

$$R_{ph(T)} = R_{ph(20)}\left[1 + \alpha_{co}\left(T - 20\right)\right] \tag{10}$$

Where $R_{ph(T)}$ is the stator resistance at temperature T, $R_{ph(20)}$ is the stator resistance at temperature of 20°, and α_{co} is the temperature coefficient of copper resistance.

3.3.5 Iron losses

Any magnetic material subjected to a variable external field of excitation is the seat of induced currents causing energy losses dissipated in the form of heat called iron losses. These losses will considerably degrade the performance of the machine as well as the efficiency of the magnetic materials used for the construction of the magnetic circuit. It is therefore important to consider these losses during the design process. In our PMSM, we have three iron losses: the first is in the stator yoke, P_{sy}, the second is in the stator teeth, P_{st}, and the last is in the rotor yoke, P_{ry}. Their equations are expressed as follows [11]:

$$P_{sy} = \left(k_h\, f\, B_{sy}^{\beta} + k_e\, f^2\, B_{sy}^2 + 8.67\, k_c\, f^{1.5}\, B_{sy}^{1.5}\right) V_{sy} \tag{11}$$

$$P_{st} = \left(k_h\, f\, B_{st}^{\beta} + k_e\, f^2\, B_{st}^2 + 8.67\, k_c\, f^{1.5}\, B_{st}^{1.5}\right) V_{st} \tag{12}$$

$$P_{ry} = \left(k_h\, f\, B_{ry}^{\beta} + k_e\, f^2\, B_{ry}^2 + 8.67\, k_c\, f^{1.5}\, B_{ry}^{1.5}\right) V_{ry} \tag{13}$$

Where k_h is the hysteresis coefficient which depends on the material, the exponent β is the Steinmetz constant, k_c is the excess losses factor, V_{sy} is the stator yoke volume, V_{st} the stator teeth volume, V_{ry} is the rotor yoke volume, and k_e is the eddy current coefficient, which is given by [10]:

$$k_e = \frac{\pi^2\, \sigma\, d^2}{6} \tag{14}$$

Where σ is the electric conductivity of the iron lamination and d is the thickness of the iron lamination.

3.3.6 Torque ripple

The stator slotting effect influences the distribution of the magnetic flux in the air gap, which gives rise to a torque ripple. It is given by the following equation [4]:

$$T_{rip} = 2\frac{\sqrt{\left(\hat{E}_7 - \hat{E}_5\right)^2 - \left(\hat{E}_{13} - \hat{E}_{11}\right)^2 - \left(\hat{E}_{19} - \hat{E}_{17}\right)^2 - \left(\hat{E}_{25} - \hat{E}_{23}\right)^2}}{\hat{E}_1} \tag{15}$$

3.4 Machine mass

The total mass of the machine is given by the sum of all the components of the machine. It is given by the following equation [1]:

$$mass = m_{st} + m_{ma} + m_{co} \tag{16}$$

Where m_{st}, m_{ma} and m_{co} are respectively the steel mass, the magnets mass and the copper mass.

These masses are calculated respectively by the following equations [1]:

$$m_{st} = \rho_{st}\left(V_{sy} + V_{st} + V_{ry}\right) \tag{17}$$

$$m_{ma} = \rho_{ma} \times V_{ma} \tag{18}$$

$$m_{co} = \rho_{co} \times V_{co} \tag{19}$$

Where ρ_{st}, ρ_{ma} and ρ_{co} are respectively the densities of steel, magnets and copper. V_{sy}, V_{st}, V_{ry}, V_{ma} and V_{co} are respectively the volumes of stator yoke, stator teeth, rotor yoke, permanents magnets and copper.

4. OPTIMIZATION ALGORITHMS

This section aims to present five state-of-the-art metaheuristics used to optimize our multiobjective problem. The first two algorithms belong to the particle swarm optimization family, and the others are evolutionary algorithms. They are respectively: Multi-Objective Particle Swarm Optimizer (OMOPSO) [12], Speed-constrained Multi-Objective PSO (SMPSO) [13], Nondominated Sorting Genetic Algorithm II (NSGA-II) [14], Nondominated Sorting Genetic Algorithm III (NSGA-III) [15] and Cellular Genetic Algorithm for Multi-Objective problems (MOCell) [16].

4.1 OMOPSO approach

M. Reyes and C. Coello [12] proposed an improved Multi-Objective PSO approach based on Pareto dominance. This algorithm is characterized by three main mechanisms: the first one is the use of the crowding distance [17] as a second criterion (additional to Pareto dominance) for the selection of the leaders. In addition, this criterion is also adopted to filter the list of leaders over generations whenever the maximum limit imposed on an archive containing nondominated solutions is exceeded. The second mechanism is the use of a combination of two mutation operators to promote diversity in the swarm and to accelerate the convergence of solutions. In fact, the swarm is subdivided into three sub-swarms of equal size. Two of them apply respectively a uniform mutation and a non-uniform mutation on the decision variables of the particles with a certain probability, while the third sub-swarm does not apply any mutation operator. Thus, these two operators allow exploration (uniform mutation) and exploitation (non-uniform mutation) of the search space along the search process. Finally, the OMOPSO algorithm makes use of the concept of ε-dominance [18] so as to limit the number of solutions produced and to control the size of the external archive.

Various experimental comparisons based on different performance metrics [12] [19], between OMOPSO [12], NSGA-II [14] and SPEA2 [20], show that OMOPSO is highly competitive with respect to the other two algorithms.

4.2 SMPSO approach

Based on the OMOPSO algorithm as a starting point, Nebro et al. [13] proposed in 2009 the SMPSO algorithm. It is characterized by the use of a strategy to control the particle's velocity when it becomes too high. This strategy also avoids the explosion of the system. Indeed, the speed constriction is done by multiplying the velocity equation of the particle by a constriction coefficient k originally developed by Clerc and Kennedy [21]. The other main difference with OMOPSO is that SMPSO applies a polynomial mutation as turbulence factor to the 15% of the particles.

A comparative study [13] based on two different criteria (the convergence speed to the Pareto front and the quality of the resulting approximation sets), and between SMPSO and four highly competitive multi-objective algorithms which are: NSGA-II [14], MOCell [16], SPEA2 [20] and AbYSS [22] and, show that SMPSO is the best in accuracy and speed.

4.3 NSGA-II approach

In 1994, Srinivas and Deb [23] proposed a multiobjective evolutionary algorithm called "Nondominated Sorting Genetic Algorithm (NSGA)". This algorithm is based on several fronts of classifications of individuals, each of which corresponds to a group of individuals having the same degree of Pareto dominance. This ensures a population progression to the Pareto-optimal surface. In addition, NSGA also uses a sharing function to maintain a wide diversity of population and to more effectively distribute solutions along the entire Pareto front. However, the major drawbacks of this approach lie in the great computational complexity, and the difficulty of specifying the value of sharing parameter σ_{share}. Accordingly, an improved approach of the NSGA algorithm, called NSGA-II [14], was invented in 2002 to resolve all NSGA criticisms.

To fully understand how this approach works, we look at a generation t. A population of parents (P_t) of size (N) and a population of children (Q_t) of size (N) coexist. The first step is to assemble these two populations to form a population ($R_t = P_t \cup Q_t$) of size (2N). This assembly ensures elitism and significantly reduces the calculation complexity.

Then, the population (R_t) is sorted in ascending order according to a non-dominance criterion to identify the different ranks. Hence the fronts F1, F2, F3, etc. The best individuals are found in the first fronts.

After that, the process is to build a new population (P_{t+1}) containing the (N) best individuals. This is done by adding the complete fronts (first front F1, second front F2, etc.) as long as the number of individuals present in (P_{t+1}) is less than (N). So, at this stage, it remains ($N - | P_{t+1} |$) individuals to include in (P_{t+1}). For this, a distance crowding procedure is applied on the first following front, (F_i), not included in (P_{t+1}). The ($N - | P_{t+1} |$) best individuals within the meaning of this crowding distance are inserted into (P_{t+1}). This crowding mechanism makes it possible to maintain the diversity of the solutions and it does not require any parameter to be fixed. The final step is to apply selection, crossover, and mutation operations to individuals in (P_{t+1}) to create a new population of children (Q_{t+1}).

4.4 NSGA-III approach

In most real-world problems that include multiple players and features, there are often many optimization issues that require four or more objectives (sometimes 10 to 15 objectives). These types of problems pose several difficulties to any optimization algorithm [24]. Indeed, the increase in number of objectives also causes the increase of number of solutions. So there is not much space to create new solutions in a generation. This slows down, the search process, and as a result, the multiobjective evolutionary algorithm becomes ineffective. On the other hand, the implementation of a diversity preservation operator increases the complexity of calculation. In addition, the visualization of a large front becomes a difficult task, since the decision maker encounters a difficulty concerning the choice of an optimal solution.

In this regard, the authors in 2014, proposed a third improved version of the NSGA approach, called NSGA-III [15], with the aim of solving the difficulties of the aforementioned optimization algorithms. Generally, this algorithm remains similar to the NSGA-II algorithm, but it uses an improved mechanism for the selection operator. But unlike NSGA-II, maintaining diversity is aided by adaptively providing a number of well-distributed reference points. The solutions that correspond to each reference point can be accentuated to explore and exploit a set of well-diversified optimal solutions.

4.5 MOCell approach

In [16], the authors introduced an optimization algorithm for continuous multiobjective problems, called MOCell, and that is based on cellular genetic algorithms [25]. This algorithm uses an external archive to preserve the non-dominated solutions found during the optimization process. After each iteration, the new solutions randomly replace the existing solutions in the archive. It also uses a measure based on crowding distance in order to decide which solutions will be removed from the archive when it becomes full, and in order to diversify the solutions in the Pareto front.

To fully understand how this algorithm works, the approach begins by creating an empty Pareto front. Every individual in the population belongs to a neighborhood containing the individuals who can cooperate with them. For each generation and for each individual, the process consists of choosing two parents belonging to the individual neighborhood; and the genetic operators (crossover, mutation) applied to them in order to obtain an offspring. After the evaluation of the results, they are preserved in the auxiliary archive and the Pareto front if they are not dominated by the current individuals. Finally, after each generation, a fixed number of individuals in the auxiliary population randomly replace the individuals from the old one.

MOCell has been compared against two evolutionary multiobjective algorithms which are NSGA-II [14] and SPEA2 [20], and evaluated with a wide range of problems (including unconstrained and constrained ones). The comparison results show that MOCell is very competitive regarding to the convergence metric, and it distinctly outperforms, according the diversity of solutions, the other two algorithms in all the test problems.

5. FORMALIZATION OF OPTIMIZATION PROBLEM

Once the definition of the analytical model of the studied machine is complete, the next step of the optimal design is to formulate the optimization problem based on this model. Thus, the optimization algorithms will be applied to the optimization problem in order to find the optimal geometry of the machine. In this section, the decision variables are given first. Next, the objectives functions are addressed. Finally, the problem constraints are investigated.

5.1 Decision variables

The machine geometry is well defined by 10 dimensioning parameters [1]. Indeed, these parameters represent the decision variables to optimize, and they are provided with their symbols in Table 1.

5.2 Constraints

The optimization problems can be with or without constraints. In our problem, we applied a set of inequality constraints that the decision variables must respect them in order to satisfy the operating requirements. Indeed, we generally find two types of inequality constraints [1]:

- Type 1: Constraints of type $B_{i\inf} \leq x_i \leq B_{i\sup}$: the values of $\vec{x}$ that satisfy these constraints define the "search space".
- Type 2: Constraints of type $c\left(\vec{x}\right) \leq 0$ or $c\left(\vec{x}\right) \geq 0$: the values of $\vec{x}$ that satisfy these constraints define the "space of achievable values".

Thus, the definition of the constraints is a rather delicate task. Indeed, the exploitation of the constraints makes it possible to find more quickly the solutions sought, but a strongly constrained problem doesn't make it possible to estimate a terminal of the Pareto front and causes afterwards a bad convergence.

In our problem, we applied these two types of constraints. Indeed, the first inequality constraints are given by the value ranges of the variables in Table 1, and for the second type, there are eight constraints imposed and which are given in Table 2 [4].

These constraints are imposed in order to avoid the magnetic saturation of the magnetic circuit (dimensions of the stator tooth width and the stator slot, dimensions of the diameter's machine and the axial length), to prevent leakage flux (air gap length, permanent magnets dimensions), and to guarantee the mechanical rigidity (slot width, tooth width, slot wedge width, slot wedge height).

Table 2: Inequality constraints of type 2

Variables	**Constraints**
Slot width	$0.15h_s \leq b_{ss2} \leq 0.5h_s$
Tooth width	$b_{ts} \geq 0.3\tau_s$
Slot wedge width	$b_{ss1}.\ k_{open} \geq 2$ mm
Slot wedge height	$h_{sw} \geq 2$ mm
Stator tooth flux density	$B_{st} \leq 2.2$
Stator yoke flux density	$B_{sy} \leq 1.8$
Rotor yoke flux density	$B_{ry} \leq 1.8$
Air gap flux density	$B_g \leq 0.95$

Where b_{ss2} is the outer stator slot, τ_s is the slot pitch and b_{ss1} is the inner stator slot width.

5.3 Objective functions

The search for the optimal geometry of the in-wheel motor must be done according to three objectives: the first is to maximize the efficiency of the machine; the second is to minimize its mass, and the third is to minimize its torque ripple. The evaluation of these objectives functions is performed as follows [4]:

$$f_1 = \min\left(1 - \eta\right) \tag{20}$$

$$f_2 = \min\left(mass\right) \tag{21}$$

$$f_3 = \min\left(T_{rip}\right) \tag{22}$$

Where η is efficiency machine, and it's equal to [4]:

$$\eta = \frac{P_{out}}{P_{out} + P_{co} + P_{st} + P_{sy} + P_{ry}} \tag{23}$$

P_{out} is the output power equal 26 kW.

6. COMPUTATIONAL RESULTS

After fixing the optimization problem and specifying the decision variables, the objective functions and the imposed constraints, the next task is to find the optimization results such as the geometric dimensions, the efficiency, the mass and the torque ripple of the machine. It should be noted that the used algorithms are able to find multiple trade-off solutions in one single run.

We present in this section the non-dominated solutions found by each approach in the form of a Pareto front. Then we choose a single solution of each algorithm to present the optimal geometry of the machine. Finally, we also present the results of the magnetic flux densities.

6.1 Trade-off solutions

We use Matlab to present the Pareto fronts and thus the compromise solutions of each approach. They are represented by the following five figures:

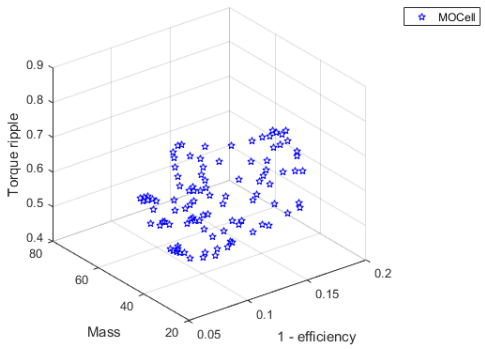

Figure 2 MOCell Pareto front

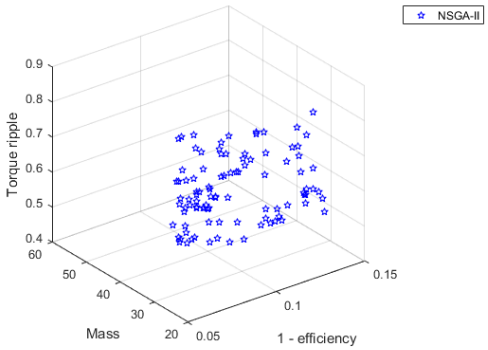

Figure 3 NSGA-II Pareto front

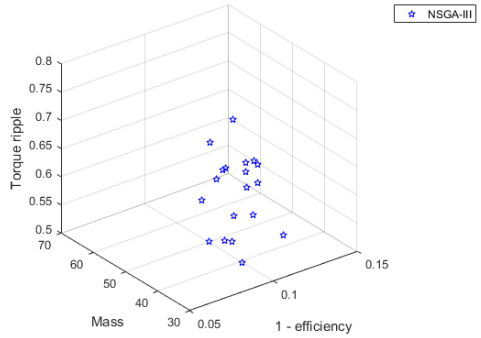

Figure 4 NSGA-III Pareto front

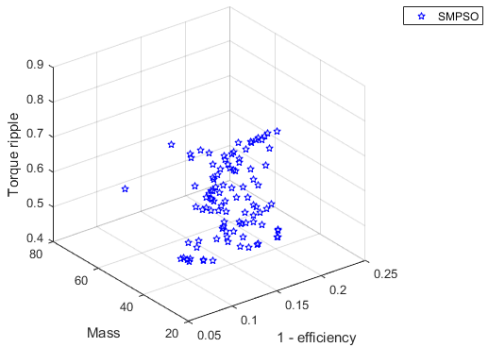

Figure 5 SMPSO Pareto front

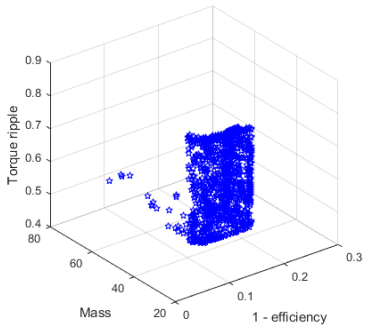

Figure 6 OMOPSO Pareto front

6.2 Optimal geometry of the machine

As the Pareto fronts show, each algorithm can provide a multitude of non-dominated solutions. We select a single significant solution from each front in order to present the geometry of the in-wheel motor and the results of the objective functions. These results are shown in Table 3. The results show that all the constraints imposed on the decision variables (type 1 constraints) are well respected.

Table 3: Optimal geometry of each algorithm

	MOCELL	**NSGA-II**	**NSGA-III**	**SMPSO**	**OMOPSO**
D_{rint} (mm)	285	285	287.06	285	286.18
δ (mm)	1.90	1.70	2.42	3.00	3.00
h_{sw} (mm)	2.06	2.05	3.10	3.63	2.00
h_m (mm)	2.00	2.00	2.05	2.00	2.00
h_s (mm)	30.18	30.00	32.98	30.00	30.00
b_{ts} (mm)	26.26	25.04	27.68	38.46	25.00
l (mm)	50.02	50.00	55.66	50.03	50.00
k_{open}	0.21	0.29	0.54	0.47	0.20
D_{sint} (mm)	194.35	194.99	176.05	195	195.00
α (°)	20.84	20.57	20.73	20.00	20.00
efficiency	0,904	0.906	0.905	0.845	0.879
weight (kg)	28.779	28.588	32.626	27.746	27.833
Torque ripple	0.683	0.737	0.702	0.809	0.805
Execution time (s)	13.265	4.554	0.600	4.117	5.912

6.3 Results of magnetic flux densities

The following table presents the magnetic flux densities corresponding to each selected solution in Table 3. These results also show that the magnetic constraints are well respected.

Table 4: Magnetic flux densities

	MOCell	**NSGA-II**	**NSGA-III**	**SMPSO**	**OMOPSO**
$B_{sy} \leq 1.8$	1.608	1.666	1.076	1.342	1.329
$B_{st} \leq 2.2$	1.908	2.005	1.578	1.292	1.502
$B_{ry} \leq 1.8$	1.744	1.791	1.650	1.240	1.399
$B_g \leq 0.95$	0.747	0.776	0.622	0.559	0.584

7. PERFORMANCE METRIC

The optimization algorithms used incorporate a stochastic aspect. Indeed, several executions do not always give the same results. Moreover, the different approaches also give different results. In this respect, we use a relative metric in order to compare the efficiency of these methods. This metric, called “covariance” [26], makes it possible to compare two compromise surfaces S' and S". The value C (S', S") is a real number between 0 and 1, which corresponds to the percentage of elements of S" which are dominated by, at

least, one of the elements of S'. The calculation of this ratio is done by the following formula:

$$C\left(S',S''\right)=\frac{card\left(\left\{\overrightarrow{v''}\in S'';\exists\overrightarrow{v'}\in S'\middle|\overrightarrow{v'}\prec\overrightarrow{v''}\right\}\right)}{card\left(S''\right)} \tag{24}$$

Thus, C (S', S") = 1 implies that the front S" is totally dominated by S'. Conversely, C(S',S") = 0 implies that none of the points of S" is dominated by a point of S'. As a result, the closer the value is to 0, the better will be the front S" with respect to S'. Moreover, since this metric is not symmetrical, we must always consider C (S', S") and C (S", S') to obtain a more reliable measurement of the two fronts to be compared.

The following table gives the results of this metric for the used optimization approaches:

Table 5: Covariance

C (column, line	MOCell	NSGA-II	NSGA-III	SMPSO	OMOPSO
MOCell	*	0.6400	0.0800	0.1200	0.5300
NSGA-II	0.0900	*	0	0	0.3200
NSGA-III	0.9474	1	*	0.9474	0.6316
SMPSO	0.4800	0.9100	0.0300	*	0.9100
OMOPSO	0.1713	0.5780	0.0153	0.0641	*

8. DISCUSSION OF THE RESULTS

The algorithms used in the process of the optimal sizing of the in-wheel motor show that they are able to explore the search space. Indeed, after several executions, the Pareto fronts of each algorithm show that they are able to find a multitude of solutions, making a compromise between the three contradictory objective functions. All the found solutions are feasible, since they respect all the imposed constraints. Thus, the user can select the solution that corresponds to his needs.

From the point of view of the solutions diversity, and as the figures of the Pareto fronts show, the best algorithm is OMOPSO because it generates a well-diversified Pareto front and which contains 1000 solutions at each execution. The algorithms SMPSO, MOCell and NSGA-II are very competitive and they make it possible to generate acceptable Pareto fronts and which contains each of them 100 solutions at each execution. However, NSGA-III algorithm loses its diversity very quickly, and it only generates a few dozen solutions for each run.

In addition, the performance metric shows the following results:

- NSGA-III is dominated by all other algorithms. This is explained by its generation of a small set of solutions.
- SMPSO is also dominated by all algorithms except NSGA-III.
- MOCell dominated by NSGA-II, OMOPSO and dominates NSGA-III, SMPSO.
- OMOPSO dominates all algorithms except NSGA-II.
- NSGA-II dominates all algorithms.

Finally, the algorithms are able to solve the multi-objective problem in a reasonable execution time.

9. CONCLUSIONS

This paper deals with the optimal design of an in-wheel motor for an electric vehicle. It consists of optimizing the geometric dimensions of a synchronous radial flux permanent magnets machine with an external rotor and concentrated windings. The optimization is done according to three contradictory objectives which are the maximization of the machine efficiency, the minimization of its mass and the minimization of its torque ripple. The optimization also depends on certain mechanical and magnetic constraints. To achieve this goal, the magnetic and electrical properties of the analytical model are presented and five intelligent algorithms (MOCell, NSGA-II, NSGA-III, SMPSO, and OMOPSO) are implemented. The optimization results show that all the algorithms make it possible to explore the search space and to find a well-diversified set of solutions. However, NSGA-III algorithm loses its diversity very quickly and it generates an incomplete Pareto front. On the contrary, OMOPSO algorithm allows generating a very rich Pareto front. The results of the performance metric show that NSGA-II dominates all the other algorithms, and NSGA-III dominated by all algorithms. In addition, the execution times of the algorithms are very reasonable. The machine efficiency can reach up to 90% and the mass can reach up to 27.7 kg. The user has a wide margin of choice.

In future work, our perspective is to modify the analytical model by modifying the stator structure with an unequal teeth structure to improve the machine performance. In addition, our attempt is to modify an optimization algorithm to deal with the constraints of the optimization problem.

REFERENCES

[1] L. Zaaraoui, and A. Mansouri, “Conception optimale d’un moteur-roue destiné à un véhicule électrique,” Editions universitaires européennes, Février 2019.

[2] H. Bouker, “Conception et optimisation des machines synchrones à aimants permanents à haute vitesse dédiées aux véhicules électriques hybrides, ” Thèse de doctorat, Université Paris-Saclay, 2016.

[3] D. Martinez, “Design of a Permanent-Magnet Synchronous Machine with Non-Overlapping Concentrated Windings, ” Degree project in Electrical Engineering Master of Science, School of Electrical Engineering, Kungliga Tekniska Hogskolan, Stockholm, 2012.

[4] A. Mansouri, “Conception et optimisation multiobjectifs d’un moteur à aimants permanents destiné pour un véhicule électrique, ” Habilitation universitaire, Génie électrique, Université de Gafsa, 2016.

[5] Z. Rahman, “Evaluating radial, axial and transverse flux topologies for 'in-wheel' motor, ” Proceedings of the International Conference on Power Electronics in Transportation, Novi, MI, USA, Oct. 2004.

[6] M. Lehr, K. Reis and A. Binder, “Comparison of axial flux and radial flux machines for the use in wheel hub drives, ” e & i Elektrotechnik und Informationstechnik, Vol. 132, pp. 25-32, Feb. 2015.

[7] A. Abeed, “Design and Optimization of a Novel Permanent Magnet Synchronous Machine Based on Transverse Flux Topology,” Ph.D. Dissertation, Electrical Engineering, Raleigh, North Carolina, 2018.

[8] A. J. Rix et M. J. Kamper, “Radial-Flux Permanent-Magnet Hub Drives: A Comparison Based on Stator and Rotor Topologies,” IEEE Trans. on Industrial Electronics, Vol. 59, pp. 2475 - 2483, 2012.

[9] Y. Y. Choe et al. , “Comparison of Concentrated and Distributed Winding in an IPMSM for Vehicle Traction, ” SciVerse ScienceDirect Energy Procedia, Vol. 14, pp. 1368 – 1373, 2012.

[10] V.X. Hung, "Modeling of exterior rotor permanent magnet machines with concentrated windings," Ph.D. thesis, Faculty of electrical engineering, mathematics and computer science, Netherlands, 2012.
[11] A. Mansouri, N. Smairi, and H. Trabelsi, "Multi-objective optimization of an in-wheel electric vehicle motor, " International Journal of Applied Electromagnetics and Mechanics, vol. 50, pp. 449–46, 2016.
[12] M.R. Sierra and C.A.C. Coello, "Improving PSO-based multi-objective optimization using crowding, mutation and ε-dominance, " Proceedings In third International Conference on Evolutionary Multi-Criterion Optimization, EMO, Guanajuata, pp. 505-519, 2005.
[13] A.J. Nebro et al. , "SMPSO: A new PSO-based metaheuristic for multi-objective optimization, " Proceedings of the IEEE Symposium on Computational Intelligence in Multi-criteria Decision-Making, pp. 66-73, 2009.
[14] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm:NSGA-II, " IEEE Transactions on Evolutionary Computation, vol. 6(2), pp. 182–197, 2002.
[15] K. Deb, and H. Jain, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-point Based Non-dominated Sorting Approach, Part I: Solving Problems with Box Constraints, " IEEE Transactions on Evolutionary Computation, vol. 18, pp. 577 – 601, 2014.
[16] A. J. Nebro et al. , "MOCell: A cellular genetic algorithm for multiobjective optimization, " International Journal of Intelligent Systems, vol. 24, pp.726-746, 2009.
[17] C.R. Raquel and P.C. Naval, "An effective use of crowding distance in multiobjective particle swarm optimization, " Proceeding of the 7th annual conference on Genetic and evolutionary computation, Washington DC, USA, pp. 257 -264, 2005.
[18] M. Laumanns, L. Thiele, K. Deb and E. Zitzler, "Combining Convergence and Diversity in Evolutionary Multi-objective Optimization, " Evolutionary Computation, pp. 263–282, 2002.
[19] A. C. Godinez, L. E. Mancilla Espinosa, E. M. Montes, "An Experimental Comparison of MultiObjective Algorithms: NSGA-II and OMOPSO, " Proceedings of the 2010 IEEE Electronics, Robotics and Automotive Mechanics Conference. Morelos, Mexico, 2010.
[20] E. Zitzler, M. Laumanns, L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization, " Proceedings of the EUROGEN2001 Conference, Barcelona, Spain, CIMNE, pp. 95–100, 2002.
[21] M. Clerc, J. Kennedy, "The Particle Swarm: Explosion, Stability and Convergence in a Multi-Dimensional complex Space, " IEEE Transactions on Evolutionary Computation, vol. 6, 2001.
[22] A.J. Nebro et al. , "AbYSS: Adapting Scatter Search to Multiobjective Optimization, " IEEE Transactions on Evolutionary Computation, pp. 439-457, 2008.
[23] N. Srinivas, and K. Deb, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms, " Evolutionary Computation, vol. 2, pp. 221–248, 1994.
[24] K. Deb, "Multi-Objective Optimization Using Evolutionary Algorithms, " Chichester, UK: Wiley, 2001.
[25] E. Alba, and B. Dorronsoro, "Cellular Genetic Algorithms, " Operations Research/Computer Science Interfaces Series, 2008.
[26] Y. Colette, and P. Siarry, "Optimisation multiobjectif, " Eyrolles, 2002.