

Multi-agent approach based on a design process for the optimization of mechatronic systems

Amir Guizani^{1,*}, Moncef Hammadi¹, Jean-Yves Choley¹, Thierry Soriano¹, Mohamed Slim abbes², and Mohamed Haddar²

¹ QUARTZ EA7393, SUPMECA-Paris, 3 rue Fernand Hainaut, 93407 Saint-Ouen, France

² Mechanics, Modelling and Production Research Laboratory, National School of Engineers of Sfax, University of Sfax, BP. 1173, Sfax 3038, Tunisia

Received: 4 June 2016 / Accepted: 11 December 2016

Abstract. Mechatronic design optimization is a complex process characterized by an important number of requirements, design variables, constraints and objectives. Therefore, it is very important to decompose efficiently the system design problem into a set of partitions to minimize the computational cost while profiting from the spatial distribution of design tools, working teams and expertise. However, the optimization of the overall design requires incorporating the relevant partitions in order to find the optimum mechatronic design. Efficient strategies of partitioning and coordination should be specified at the conceptual level to have a successful optimization process. In this paper, a new approach based on multi-agent paradigm is proposed for mechatronic design optimization. The proposed approach is applied to the preliminary design case of an electric vehicle to demonstrate its validity and effectiveness.

Keywords: Multi-agent approach / mechatronic design / modelling / simulation / optimization

1 Introduction

A typical mechatronic design involves multidisciplinary teams to work together on the design of a mechatronic system. Challenges associated to a collaborative design of multidisciplinary teams require dividing the work into smaller design problems easier to manage and coordinating between different design sub-problems in order to avoid misleading conclusions that could make the mechatronic system expensive or useless.

In this paper, a new approach for a mechatronic design optimization based on multi-agent technologies is presented. The suggested approach has been applied to the case of an electric vehicle design optimization to illustrate its efficiency in facilitating the complex design of mechatronic systems.

This paper is organized as follows: After the introduction, the related works addressing the optimization of complex and mechatronic systems are presented. In the subsequent section a detailed description of our approach for optimizing the mechatronic systems design is offered. Afterwards, a preliminary design of an electric vehicle is

proposed to illustrate our approach. The results of the application are presented and discussed in section five. The last section is devoted to the conclusion.

2 Related works

The complex system design process begins by analysing how to break a multidisciplinary design problem and large scale individual elements. Several approaches for system partitioning may be used, and the choice depends on the system and the analysis environment. Wagner [1] identified four categories of system partitioning methods: by object, by aspect, sequential, or matrix. Object decomposition involves dividing a system to physical component or function. Aspect partitioning divides the system to discipline. Sequential partitioning is appropriate for flow processes. Matrix partitioning is applied to large systems of mathematical equations.

Plume and Mitchell [2] stressed that a good design can only be achieved by focusing on the parts, because the parameters of one part may affect the level of other parts. There is therefore a risk that one part is evaluated incorrectly which might make the system unnecessary, expensive or inefficient.

Several approaches have been proposed in previous works to solve this issue. The Multidisciplinary design optimization (MDO) seems to be the most promising solution for the optimization of mechatronic systems, as it

* e-mail: amir.guizani@supmeca.fr

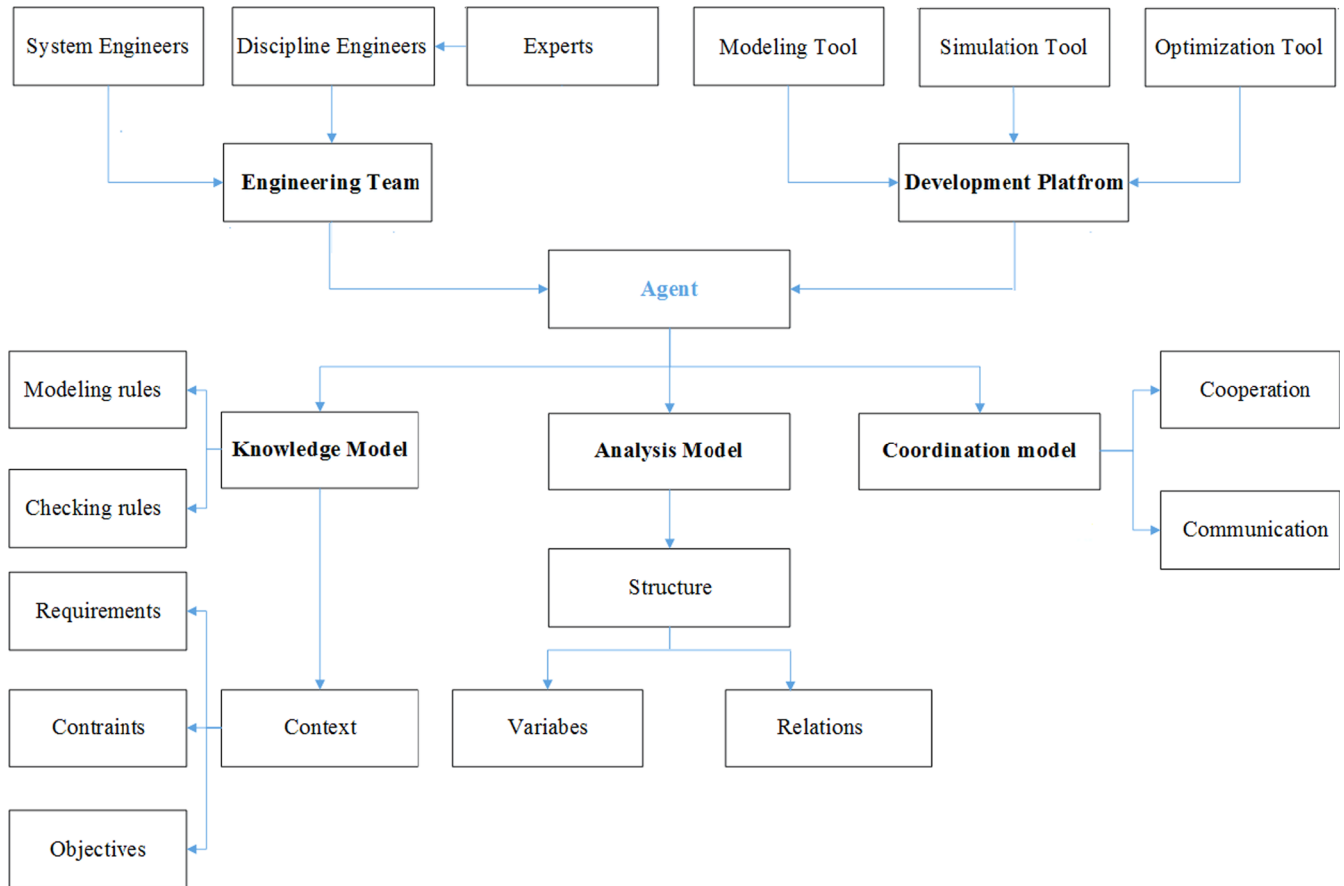


Fig. 1. The definition of an agent for the optimization of mechatronic design.

allows designers to incorporate all relevant disciplines simultaneously. MDO has been introduced in numerous fields, including aerospace design [3], electronics [4], automobile design [5], etc.

MDO allows designers to reach the optimum of the collaborative design which is more advantageous compared to the design found by optimizing each discipline sequentially, since it can exploit the interactions between the disciplines. However, including all disciplines simultaneously increases significantly the complexity of the problem and computation time.

The optimization of mechatronic systems with a detailed level of design requires efficient strategies of partitioning and coordination to lead to a successful optimization workflow that reduces the complexity and computation time. One idea for an optimal design of mechatronic systems is to use the multi-agent approach.

The Multi-agent approach is considered as an emerging development of a combination of trends including artificial intelligence, object-oriented programming and concurrent object-based systems [6]. Multi-agent systems are already present in several multi-domain industrial applications such as: software development, intelligent manufacturing and intelligent transportation systems [7]. Several past studies have dealt with multi-agent technologies to address the problems of distributed analysis and collaborative design. Villanueva et al. [8] proposed the use of multi-agent systems with a new technique of adaptive partitioning of

the design space to solve optimization problems involving expensive digital simulators. The technique consists in partitioning the design space between different agents using different meta-models to approximate their sub-region and coordinating to change the boundaries of their sub-regions. Hao et al. [9] worked on a framework for engineering design and optimization based on agents. Distributed decision-making is among the main services of the developed application. La Rocca and van Tooren [10] proposed a new knowledge-based application for aircraft multidisciplinary design and optimization. The authors in [10] argued that the use of a knowledge-based approach presents more flexibility and automation, which provides solutions to the urgent problems including the exploitation of the MDO approach in large distributed design frameworks. Ren et al. [11] presented a comparison study between MDO and multi-agent technology to optimize the construction design. Authors in [11] concluded that agent based systems may be adopted into the collaboration process in the MDO to enhance the efficiency of the process.

3 Description of the multi agent approach for mechatronic design optimization

Our approach is based on a multi agent platform containing only software agents, which communicate with physical agents (designers, engineers, etc.) and design software (modelling, simulation and optimization).

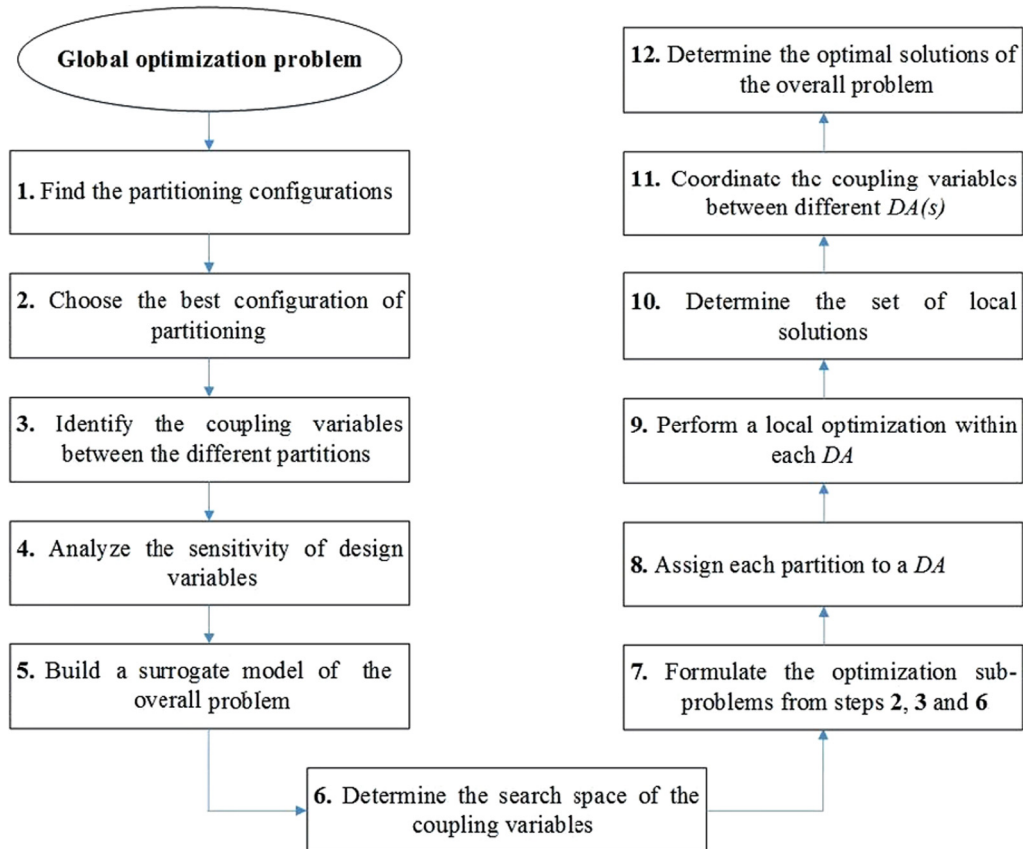


Fig. 2. The design process for optimizing a mechatronic system.

For each mechatronic optimization problem, we choose the multi agent platform to be composed of two types of agents: Design Agents (DA s) and a Coordinating Agent (CA). This choice is justified by the fact that in every optimization problem, there are global objectives related to the system and local objectives linked to components or subsystems. The CA objective is to participate in the research of global optimal solutions for the mechatronic optimization problem, while each DA is affected to a local partition of the optimization problem in order to find local optimal solutions.

Each software agent (CA or DA) is associated to an engineering team and a development platform (Fig. 1). The engineering team may be composed of engineers and experts in the various fields of Systems Engineering. The platform consists of various development tools for modelling, simulation and optimization.

Each software agent can share, communicate or coordinate with physical agents throughout the analysis model, the knowledge model and the coordination model:

- the analysis model is used to evaluate the design assigned to a partition (CAD model, finite element model);
- the knowledge model is based on the rules of modelling, control and contextual information of the agent. The modelling rules help the designers to define the mathematical models that can be used in the mechatronic system modelling. The control rules are used to verify the consistency of the analysis model. The agent context is based on information about design requirements, constraints and optimization objectives;

- the coordination model contains information for cooperation and communication between DA (s) such as coupling variables and shared variables, etc.

In order to explain our approach, the rest of this part details the principal tasks realized along the optimization process to facilitate the collaborative distributed design. Thus, we indicate the steps where CA and DA intervene and the tasks to be realized. Figure 2 shows the design process used in the multi-agent approach for optimizing a mechatronic system.

The first task in the design process is to define the optimal number of DA (s) necessary for the optimization and the optimal repartition of objectives, constraints and design variables between different agents. In order to accomplish this task, it is necessary to execute the steps below:

1. The first step consists in finding the possible configurations to partition the optimization problem. Several techniques may be used to ensure this step. The simplest technique is that the partitioning is manually made by studying the coupling degree between the objective functions, the constraints and the design variables or by following approaches that already exist in the literature such as: the graph theory [12], the game theory [13], the Augmented Lagrangian Coordination [14], etc.

2. Once the configurations are found, the second step of the design process is to find the best configuration to be used to solve the global optimization problem. To obtain the best configuration, we propose to use two criteria: the

calculation time of the overall solution C_1 and the number of partitions C_2 of each configuration. Both criteria must be minimized.

With

$$C_1 = t_{global} = \max t_{local} + \max t_{exchange}, \quad (1)$$

where

- t_{global} : the global calculation time of the configuration,
- t_{local} : the local calculation time for each partition (can be evaluated for one iteration)
- $t_{exchange}$: the data transfer time (the information amount to be exchanged) between two partitions (depends on the number of coupling variables, number of shared variables, used tools, etc.)

The software agent CA can intervene in this step in order to help us to choose the best configuration. For this, we propose to use the SysML [15] language. In fact, the optimization configuration can be represented with an Internal Block Diagram (IBD). Each block can represent a partition. The necessary information to describe the partition such as execution time can be attributed to the properties of the block that define the partition. The SysML ports ensure the specification of the exchange variables and can contain other information about exchange time throughout the ports. A CA that can be described via Java program can read the information defined in the IBD via a XML format, which ensures the evaluation of criterion C_1 .

3. The third step is to determine the coupling variables between the partitions of the chosen configuration. These coupling variables are subsequently used to facilitate the coordination process between the different partitions in order to find the optimal design of the global problem.

Once the coupling variables are identified, the designer verifies if these variables have variation intervals. If it is the case, the design process goes directly to the seventh step. Otherwise, the designer has to determine the variables having unknown validity domains to execute the fourth, the fifth and the sixth steps.

4. In the fourth step, a sensitivity analysis of the design variables is necessary to identify those with negligible influence on the behaviour of the partitions already identified in the previous step. These variables have to be eliminated in order to maximize the precision and minimize the calculation time of the meta-model.

5. Once the sensitivity analysis is done, the physical agent builds a meta-model (surrogate model) having as inputs the design variables and as outputs the objective functions, the constraints and the coupling variables identified in the third step.

6. After building the surrogate model, the role of the designer is to determine the validity range of the coupling variables based on the optimization of the surrogate model.

7. The following step is to formulate the optimization sub-problems of the different partitions of the selected configuration issuing from the previous steps. The formulation of each optimization sub-problem requires:

- the choice of a partition of the configuration selected in the second step (specify the design variables, the local constraints and objectives);

- the determination of the coupling variables that are linked to the chosen partition according to the third step;
- the specification of the validity domain of every coupling variable according to the third or the sixth step.

8. Next, the CA assigns each partition (sub-problem) to a DA . In fact, starting from the XML file already built in the second step, the CA can transform the data of each partition into messages containing the optimization formulations of every DA .

After the reception of the formulation messages from the CA , each DA informs the associated designer about the proposed formulation in order to accomplish a local optimization in his specific partition and determine the optimal local solutions compared to the internal objectives to meet the requested requirements (steps 9 and 10 in the design process). The local optimization of each DA requires the:

- specification of an analysis model to evaluate the local constraints and objectives;
- allocation of a multi-objective optimization algorithm to determine the local optimal solutions;

Once the local optimizations are performed and the solutions are generated, the system engineer's task, with the help of CA , is to find the best strategy to link the coordination variables (the coupling variables and the shared variables between different $DA(s)$) and to find the global solutions that respect all the required performances (steps 11 and 12).

4 Application to the case of a preliminary design of an electric vehicle

4.1 Introduction

We consider the case of a preliminary design of an electric vehicle. The objective of this study is to optimize the battery, the electric motor and the gear ratio to meet the performance requirements related to the maximum velocity and the acceleration test.

4.2 Modelling the electric vehicle with Modelica

The electric vehicle model has been modelled using Modelica language [16]. This model is composed of Figure 3: an input model for velocity demand, a battery (DC voltage source), a rectifier (DC-AC inverter) and a propulsion system. The latter is composed of an electric motor (permanent magnet synchronous motor) connected to a one-ratio gear box, a wheel component and a translating mass representing the vehicle mass. This mass is supporting resistive forces due to air drag, rolling resistance of the wheels and grade angle resistance. A detailed model of the electric vehicle is developed in [17].

4.3 Simulation results

For the sake of simulation and to reproduce a road path with different driving conditions, we will be using the New European Driving Cycle (NEDC) [18]. In this study, the parameters used in the electric vehicle model are given in Table 1.

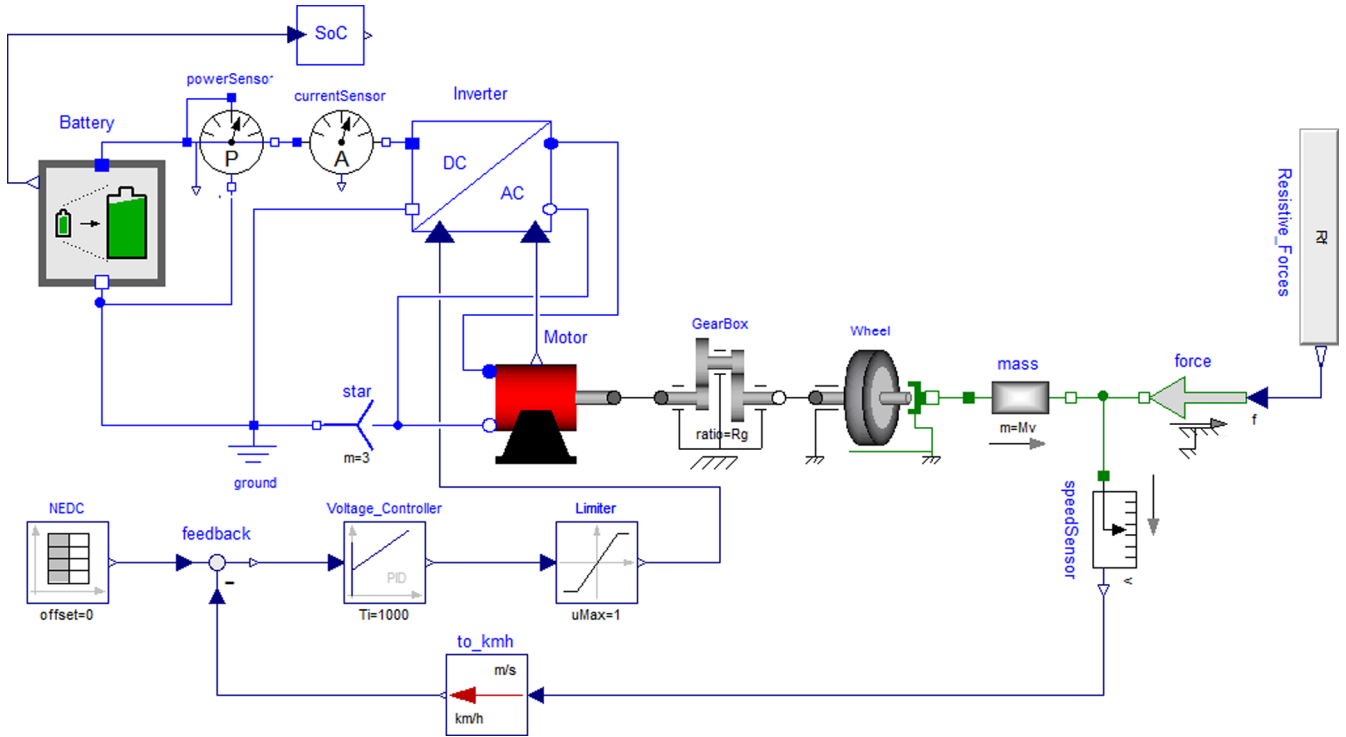


Fig. 3. Electric vehicle model developed with Modelica.

Table 1. Electric vehicle parameters.

Parameters	Description	Value	Unit
M	Vehicle mass	1540	kg
ρ	Air density	1.2	$\text{kg}\cdot\text{m}^{-3}$
A_{front}	Vehicle frontal area	1.8	m^2
C_{ro}	Rolling coefficient	0.2	–
C_d	Aerodynamic drag coefficient	0.013	–
α	Grade angle	0	rad
g	Gravitational acceleration constant	9.81	$\text{m}\cdot\text{s}^{-2}$
r_w	Wheel radius	0.28	m
R_g	Gearbox ratio	$1 \leq R_g \leq 12$	–
Emf	Constant of the electromotive force	$0.1 \leq Emf \leq 2$	$\text{N}\cdot\text{m}\cdot\text{A}^{-1}$
nu	Inverter efficiency	0.98	–
N_s	Number of series connected	$1 \leq N_s \leq 100$	–
N_p	Number of parallel strings	$1 \leq N_p \leq 10$	–
SoC_{init}	Initial state of charge	0.8	–
V	Speed vehicle	Variable	km/h
SoC	State-of-charge	Variable	–

The objective is to simulate the proposed model. The default values of the N_s , N_p , Emf and R_g are respectively 75, 4, 0.5 and 6.

The results shown in Figure 4 indicate a good agreement between the input driving cycle and the output vehicle velocity. By modifying the input to constant speed, we can simulate other different performance test cases, such as the maximum vehicle velocity, the velocity at 10 seconds and the velocity in a grade road.

The state-of-charge SoC is the history of the battery when a driving cycle used. Figure 5 shows the variation of the SoC during the NEDC cycle. The first SoC was 80%. The operating range is between 80% and 67%, i.e., the battery has lost 16.25% of its capacity during the NEDC cycle. With a minimum state-of-charge $SoC_{min}=0.2$ and the same values of N_s , N_p , Emf and R_g the electric vehicle can repeat 4.6 times the NEDC cycle.

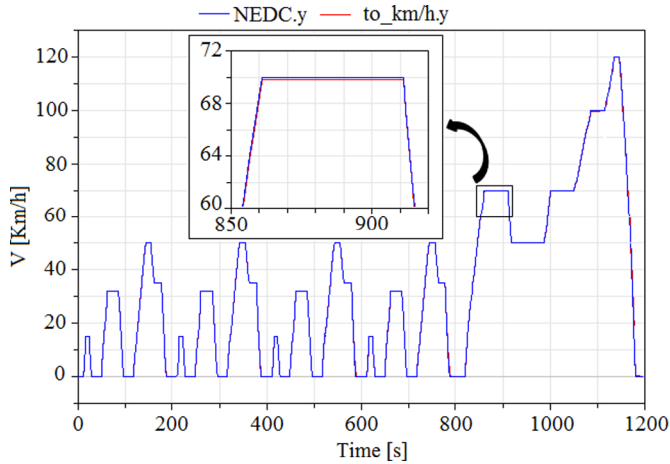


Fig. 4. Comparison between the input NEDC cycle and the output vehicle velocity.

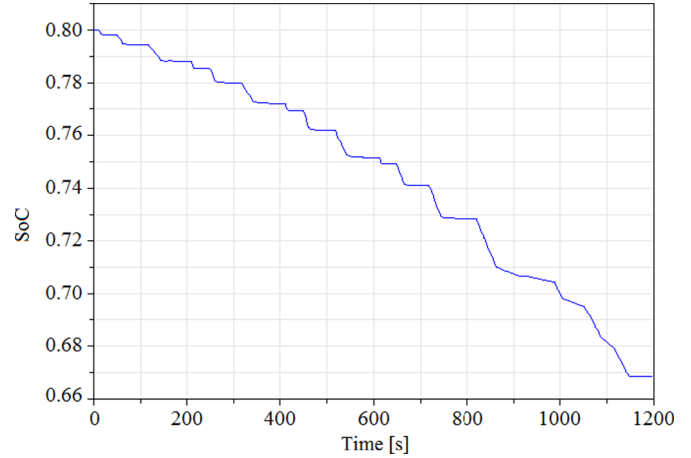


Fig. 5. State-of-Charge for the NEDC.

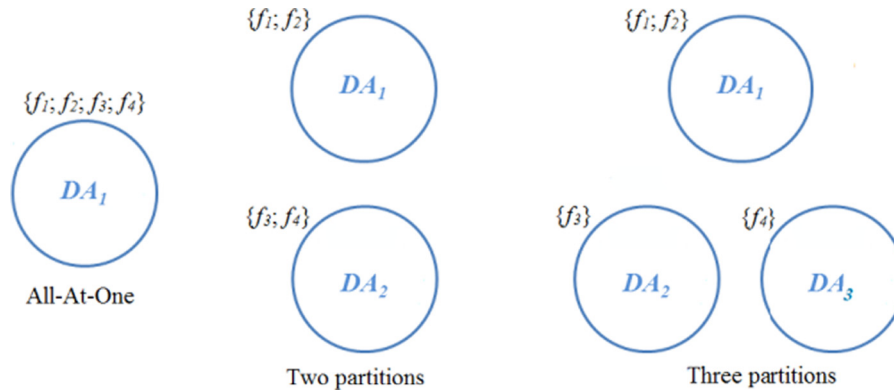


Fig. 6. Three possible configurations for partitioning.

Defining one analysis model for different test cases such as acceleration test and maximum velocity is a complex task and computationally expensive when classical MDO techniques are used. Therefore, the multi-agent approach is used to solve these problems.

To develop our approach, it is convenient to choose a platform, adopting open-sourced JADE [19] as the software development environment fully implemented in JAVA language. JADE follows the Agent Management References Model (AMRF) specification, which provides a standard infrastructure with well-defined functions. Multiple agents can be easily created, deployed, registered, located and communicated in JADE.

4.4 Use of the Multi-Agent approach

We consider the optimization of an electric vehicle with four objective functions and three design constraints:

- Objective f_1 : minimizing the mass of the energy storage system in order to choose the optimal battery;
- Objective f_2 : maximizing the state-of-charge of the battery at the end of a driving cycle;
- Objective f_3 : minimizing the electric power required by the propulsion and the transmission systems in order to choose the optimal electric motor;

Objective f_4 : minimizing the gear ratio to reduce the volume of the gearbox;

- Constraint c_1 : Acceleration test: Vehicle velocity after 10 seconds of start-up (V_{10}) shall be equal to 60 ± 5 km/h;
- Constraint c_2 : Maximum velocity: Vehicle maximum velocity (V_{max}) shall be equal to 120 ± 5 km/h;
- Constraint c_3 : Battery Mass: The mass of the battery should be less than 200 kg.

For the design variables, we will limit our study to the number of series connected cells N_s , the number of parallel strings N_p , the electromotive force Emf of the electric motor and the gear ratio R_g .

The first step of the design process is to find the possible configurations for the partitioning of the optimization problem. An important measure, which is used in this case, is the degree of coupling between the objective functions, constraints and design variables.

According to the formulation of the optimization problem, the two constraints V_{10} and V_{max} are related to the overall system performance. It is difficult to assign these two constraints to specific partitions. Furthermore, the decomposition following the design variables is very effective if the whole problem is modeled in a mathematical form. The problem of the electric vehicle is modeled at the

Table 2. Choice of the best configuration.

Criterion	Configuration1	Configuration2	Configuration3
C_1 (for one evaluation)	5.17 s	2.6 s	2.1 s
C_2	1	2	3

Table 3. Optimization results of the surrogate model.

RSM					
N_s	74–83	M_{bat} (kg)	149.7–198.2	V_{max} (km/h)	115.4–123.9
N_p	4–5	SOC	0.722–0.754	I (A)	141.6–155.3
Emf (Nm/A)	0.203–0.381	P (kW)	54.6–65.7	U (V)	385.8–422.7
R_g	4.73–7.91	V_{10} (km/h)	55.1–64.4		

same time by components and mathematical models. Hence, it is indispensable to study the degree of coupling between the objective functions, which are associated with specific components (battery, motor and gearbox) to decompose the problem into multiple partitions and to assign constraints and internal design variables to each partition.

In our case, several combinations can be studied; some of them are realizable. The three possible configurations are: a partition, two partitions and three partitions (Fig. 6).

One partition is a multidisciplinary optimization problem where all objective functions are assigned to a single partition which corresponds to all-at-one (AAO) problem. In this case one analysis model is needed to solve the overall problem. AAO currently used for the optimization of multidisciplinary systems is expensive in computation time, difficult to implement, and inflexible with the preliminary design phase where the objectives and design constraints change frequently.

In the configuration with two partitions, both objective functions f_1 and f_2 are tightly coupled, so they must be assigned to the same partition. However, the degree of coupling between the two objective functions f_3 and f_4 is low. Therefore, the solution is to consider two analysis models assigned to both partitions. The first model is the model of the battery to analyze f_1 and f_2 and the second model is the propulsion system to treat f_3 and f_4 . These analysis models are associated to both partitions that may be run in parallel either on the same computer or on two different computers, which reduces the computation time.

In the configuration with three partitions, the degree of coupling between f_3 and f_4 is low so that each function can be associated with an analysis model. In this case, three analysis models are needed to treat this problem: The battery model for f_1 and f_2 , engine model for f_3 and last model is the transmission model for f_4 .

The second step is finding the best configuration to be used to solve this problem. The two criteria that can be considered here are: the computation time of the overall solution C_1 and the number of partitions for each

configuration C_2 . Both criteria must be minimized. According to Table 2, the choice is trivial since both criteria are in favor of configuration with two partitions.

In our case, the coupling variables between the two partitions of the selected configuration are the voltage U and the current I of the battery. The search space of these variables is unknown and therefore it is necessary to perform the steps 4–6 during the design process to determine the validity domain of U and I .

To identify the validity areas of the coupling variables, first it is necessary to make a sensitivity analysis of the design variables. The four design variables N_s , N_p , Emf and R_g have a great influence on the required system performance. Indeed, the objective function f_1 is expressed according to N_s and N_p ($M_{bat} = N_s \cdot N_p \cdot M_{cel}$). The function f_3 depends on the internal characteristics of the motor in particular the electromotive force Emf and f_4 represents the reduction ratio R_g .

The next step is to build a surrogate model of the overall problem. The input vector of the surrogate model is $X = \{N_s, N_p, Emf, R_g\}$ and output vector is $Y = \{M_{bat}, SoC, P, R_g, V_{10}, V_{max}, U, I\}$. The design field is defined by: $1 \leq N_s \leq 100$, $1 \leq N_p \leq 10$, $0.1 \leq Emf \leq 2$ and $1 \leq R_g \leq 12$.

Several techniques have been used to build a surrogate model. The polynomial Response Surface Method (RSM) [20] is commonly considered as the first surrogate modelling technique. It uses a polynomial formulation to approximate an exact function. Other techniques such as Kriging and Radial Basis Function Neural Networks (RBFNNs) [21] are also used to model complex relationships between inputs and outputs. In this example, the technique RSM provided in the Model Center library is used to construct the surrogate model.

Once the surrogate model is built, the designer's task is to define an optimizer to determine the range of validity of the coupling variables. In the current study, the optimization of the surrogate model is performed using the Non-dominated Sorting Genetic Algorithm II (NSGA II) [22] with Model Center software. NSGA II is a multi-objective optimization technique that uses a non-dominated sorting

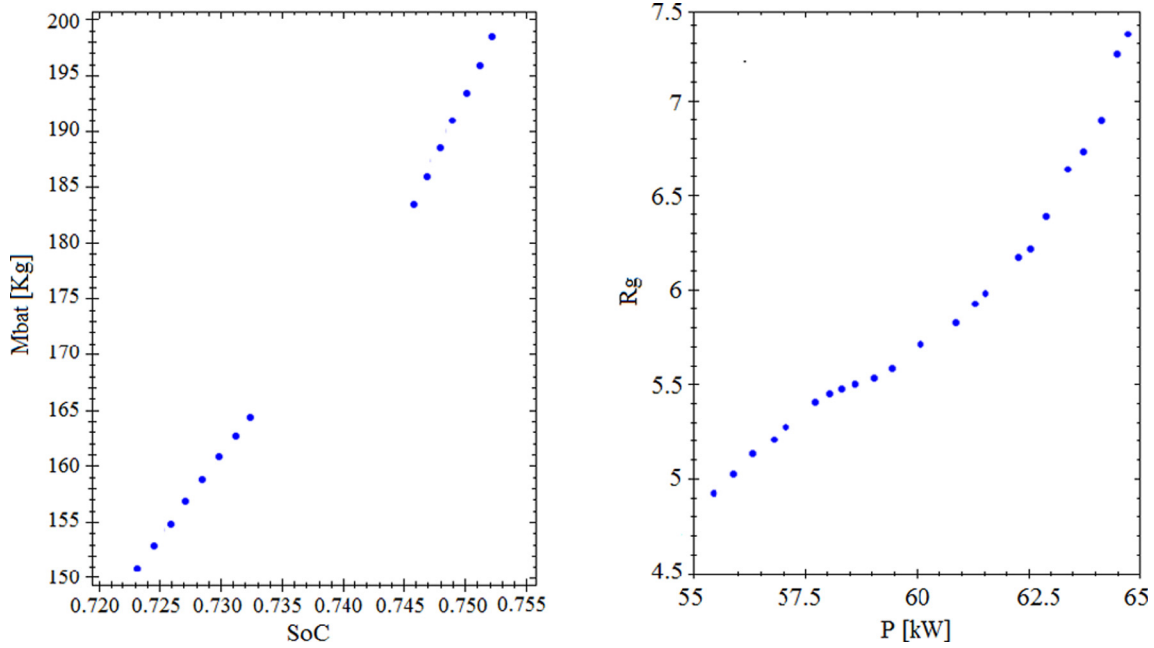


Fig. 7. Pareto fronts for DA1 (left) and DA2 (right).

genetic algorithm. A design is said to be dominated if there is another design that is superior to it in all objectives. NSGA II is a fast sorting algorithm to compute Pareto set.

The optimization problem is multi-objective, the surrogate model was not a single solution but a set of solutions whose lower and upper limits of the vectors X and Y are given in Table 3.

The next step is to formulate the optimization problems of $DA(s)$ from the previous steps. For the selected configuration with two partitions, we will consider two design agents DA_1 and DA_2 . Each DA has an analysis model (Modelica) to evaluate its constraints and local objectives. For each DA we allocate the same optimization algorithm NSGA II with Model Center software.

The first analysis model consists of the battery in series with an equivalent resistance R_{eq} representing the rest of the electric vehicle model. The value of this resistance is determined from the optimization results of the surrogate model:

$$R_{eq} = U_{\max}/I_{\max} = U_{\min}/I_{\min} = 2.72\Omega. \quad (2)$$

The second analysis model is the model of the propulsion system. To get an accurate simulation of this system it must connect with a constant voltage source U to replace the battery model.

The formulation of each optimization sub-problem is given by:

$$DA_1 : \begin{cases} \min M_{bat} \\ \max SoC \\ M_{bat} \leq 200 \\ 141.6 \leq I \leq 155.3 \\ 385.8 \leq U \leq 422.7 \\ 1 \leq N_s \leq 100 \\ 1 \leq N_p \leq 10 \end{cases} \quad DA_2 : \begin{cases} \min(P, R_g) \\ 55 \leq V_{10} \leq 65 \\ 115 \leq V_{\max} \leq 125 \\ 141.6 \leq I \leq 155.3 \\ 385.8 \leq U \leq 422.7 \\ 0.1 \leq Emf \leq 2 \\ 1 \leq R_g \leq 12 \end{cases} \quad (3)$$

The optimization results of each DA and the coordination process of the CA are explained in the following section.

4.5 Results and discussion

After the complete specification of two $DA(s)$, a local optimization was performed within each agent. Each agent is characterized by a local optimizer and an analysis model. Both optimizations are based on the electric vehicle parameters and variables listed in Table 1. Both optimizations are multi-objective; these problems do not have a unique solution, but a set of solutions (Pareto Front) as shown in Figure 7.

Each point of the Pareto front is characterized by an input vector X (design variables to optimize) and an output vector Y (objective functions to be achieved and constraints to be respected).

After the generation of the Pareto fronts, the best design solutions obtained by $DA(s)$ are summarized in Table 4.

For instance, the optimization performed by DA_2 shows that the maximum power required by the motor P ranges between 55.4 kW and 64.8 kW, a variation on ΔP is about 14.5%. The gear ratio of the gearbox R_g varies between 4.92 and 7.36, which has a ΔR_g variation is about 33%. The two criteria therefore vary strongly along the optimal front, which means that there is a compromise between them. If we find that one of the criterion is showing a significant variation with respect to the other, e.g., negligible ΔP to ΔR_g , this would mean that it is possible to improve the criterion of the maximum power required by the motor P without having a significant impact on the criterion of gear ratio R_g . The compromise is likely to exist in this case of DA_2 and also in the case of DA_1 . The choice of a solution on the fronts is difficult. Therefore, it is very difficult to

Table 4. Optimization results of $DA(s)$.

Results after optimization					
DA_1	N_s	75–82	DA_2	Emf (Nm/A)	0.214–0.348
	N_p	4–5		R_g	4.92–7.36
	I (A)	142.1–152.5		I (A)	142.6–151.6
	U (V)	387.9–416.1		U (V)	392.1–419.3
	M_{bat} (kg)	150.8–198.4		P (kW)	55.4–64.8
	SOC	0.723–0.752		V_{10} (km/h)	56.5–63.8
	Computing time (min)	14.56		V_{max} (km/h)	116.2–123.5
	Number of solutions	15		Computing time (min)	21.36
			Number of solutions	23	

Table 5. Optimal solutions of the multi-agent approach.

Solutions	N_s	N_p	Emf	R_g
1	79	4	0.264	5.83
2	80	4	0.251	6.18
3	81	4	0.244	6.41
4	82	4	0.226	6.91
5	76	5	0.309	5.14
6	77	5	0.280	5.46
7	78	5	0.270	5.59

Table 6. Optimization results of the AAO method.

	AAO
N_s	74–83
N_p	4–5
Emf (Nm/A)	0.214–0.413
R_g	4.52–7.76
M_{bat} (kg)	150.8–198.4
SOC	0.722–0.756
P (kW)	53.4–66.3
V_{10} (km/h)	55.1–64.6
V_{max} (km/h)	115.2–123.6
Computing time (min)	43.22
Number of solutions	10

determine the values of the design variables of DA_1 and DA_2 that meet the requirements requested in order to find the optimal solutions for the overall problem. The advantage of the agent based approach is to define rules allowing the CA to make decisions in such cases of conflicts.

The simple design rule used by the CA consists in coordinating the coupling variables between the different $DA(s)$ to find consistency between the good values of the design variables that meet the overall system performance.

In our case, the two coupling variables are U and I . The coordination algorithm used by the CA to determine the values of the design variables of DA_1 and DA_2 is given by:

```

For i = 1 to n
  For j = 1 to m
    If  $\|U_1(i) - U_2(j)\|^2 \leq 1$  and  $\|I_1(i) - I_2(j)\|^2 \leq 1$  then
      Return ( $N_s(i), N_p(i)$ ) of  $DA_1$  and ( $Emf(j), R_g(j)$ )
    End If
  End For
End For

```

where, n and m are the numbers of optimal solutions along the Pareto fronts of DA_1 and DA_2 respectively. (U_1, I_1) and (U_2, I_2) are the values of the coupling variables of DA_1 and DA_2 respectively.

The implementation of the coordination algorithm allowed us to determine 7 optimal solutions whose values are given in [Table 5](#).

[Figure 8](#) shows the variation of the vehicle speed for the first 4 optimal solutions for which we have $57.1 \leq V_{10} \leq 62.5$ and $118.3 \leq V_{max} \leq 122.4$.

The choice of the global optimal solution is based on decision making rules that we add to the CA . For example, if the priority is given to minimizing the mass of the battery, the characteristics of the optimal solution are: ($N_s = 79, N_p = 4, Emf = 0.264$ and $R_g = 5.83$). However, if the priority is given to minimizing the volume of the gearbox, the best solution in this case is: ($N_s = 76, N_p = 5, Emf = 0.309, R_g = 5.14$).

To validate our approach, a comparative study with classical optimization methods is required. For this reason, we used one of the most used multidisciplinary optimization methods in the literature: the AAO method which corresponds to the first partitioning configuration of our problem where all the objective functions are associated with a single partition. In addition, even if this case-study seems to be relatively simple, the exact solution is not easy to find to solve this problem. Therefore it is necessary to have an analysis model; this model has been presented in [Figure 3](#). The optimization results of this configuration are shown in [Table 6](#).

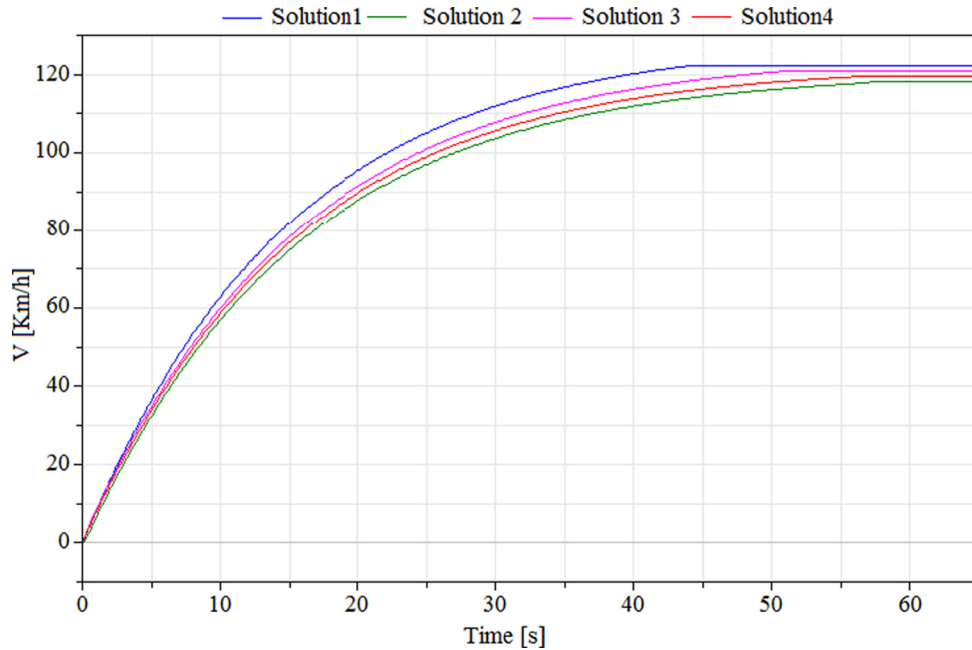


Fig. 8. The vehicle velocity for the optimal configurations.

The two criteria selected here to compare the multi-agent approach with the AAO method are the total number of optimal solutions and overall calculation time. The multi-agent approach allowed us to identify 7 solutions in 21.36 min (this is the maximum computation time of different partitions of the selected configuration) while with the AAO method we detected 10 solutions in 43.22 min. Comparing these optimization results, it can be deduced that 70% of the total number of solution is determined with an important reduction of 50.6% in the overall calculation time.

In this paper, we have described a multi-agent approach that enables agents to follow a design process in order to facilitate the coordination of distributed design. We decomposed the design problem into two design agents and we assigned to each agent its objectives and constraints. A coordinating agent was considered to coordinate between design agents to achieve optimal decisions.

The proposed approach allows designers to integrate the optimization process requirements in the conceptual design level, simplify the complexity of a mechatronic design optimization and reduce the computation time. Global results were found and the optimal design simulation results were presented.

5 Conclusion

Optimizing mechatronic systems with classical MDO techniques is quite challenging. As a result, using multi-agent technologies is considered to be the solution to handle this problem. Our main contribution in this study is proposing a multi-agent approach to decompose the design optimization problem into partitions and coordinate between them. For each partition, we allocate a design

agent to make a local optimization. To find the global optimal design of mechatronic systems, we considered another type of agent called the coordinating agent. The proposed approach is based on a design process to facilitate the collaborative distributed design of mechatronic systems. This approach was applied to the case of a preliminary design of an electric vehicle to illustrate how the use of the multi-agent paradigm helps designers in making effective decisions and to achieve an optimal decision of the overall problem.

Acknowledgment. The authors would like to thank Mrs. Samia Ammar, English language teacher in the National Engineering School of Gafsa (ENIGafsa), for her time in English language proofreading.

References

- [1] T.C. Wagner, A general decomposition methodology for optimal system design, Thesis, University of Michigan, 1993
- [2] J. Plume, J. Mitchell, Collaborative design using a shared IFC building model-learning from experience, *Autom. Constr.* 16 (2007) 28–36
- [3] J. Sobieszcanski-Sobieski, R.T. Haftka, Multidisciplinary aerospace design optimization: survey of recent developments, *Struct. Optim.* 14 (1997) 1–23
- [4] M. Hammadi, J.Y. Choley, O. Penas, J. Louati, A. Rivière, M. Haddar, Layout optimization of power modules using a sequentially coupled approach, *Int. J. Simul. Model.* 10 (2011) 122–132
- [5] A. Guizani, M. Hammadi, J.Y. Choley, T. Soriano, M.S. Abbes, M. Haddar, Multidisciplinary optimization of mechatronic systems: application to an electric vehicle, *Mechatron. Syst.: Theor. Appl.* (2014) 1–14
- [6] M. Wooldridge, Agent-based software engineering, *IEE Proc. Softw. IET* 144 (1997) 26–37

- [7] S. Park, V. Sugumar, Designing multi-agent systems: a framework and application, *Expert Syst. Appl.* 28 (2005) 259–271
- [8] D. Villanueva, G. Picard, R. Le Riche, R.T. Haftka, Optimisation multi-agent par partitionnement adaptatif de l'espace de conception, 20èmes Journées Francophones sur les Systèmes Multi-Agents JFSMA, 2012
- [9] Q. Hao, W. Shen, S.W. Park, J.K. Lee, Z. Zhang, B.C. Shin, An agent-based e-engineering services framework for engineering design and optimization, *Innov. Appl. Artif. Intell.* (2004) 1016–1022
- [10] G. La Rocca, M.J. van Tooren, Knowledge-based engineering to support aircraft multidisciplinary design and optimization, *Proc. Inst. Mech. Eng. G: J. Aerosp. Eng.* 224 (2010) 1041–1055
- [11] Z. Ren, F. Yang, N. Bouchlaghem, C. Anumba, Multi-disciplinary collaborative building design—a comparative study between multi-agent systems and multi-disciplinary optimisation approaches, *Autom. Constr.* 20 (2011) 537–549
- [12] D.J. Pate, J. Gray, B.J. German, A graph theoretic approach to problem formulation for multidisciplinary design analysis and optimization, *Struct. Multidiscip. Optim.* 49 (2014) 743–760
- [13] E. Ghotbi, Bi-and multi-level game theoretic approaches in mechanical design, Thesis, University of Wisconsin-Milwaukee, 2013
- [14] S. Tosserams, L. Etman, J.E. Rooda, Augmented lagrangian coordination for distributed optimal design in MDO, *Int. J. Numer. Methods Eng.* 73 (2008) 1885–1910
- [15] S. Friedenthal, A. Moore, R. Steiner, A practical guide to SysML: the systems modelling language, Morgan Kaufmann, 2014
- [16] P. Fritzson, Principles of object-oriented modelling and simulation with Modelica 2.1, John Wiley and Sons, 2010
- [17] A. Guizani, M. Hammadi, J.Y. Choley, T. Soriano, M.S. Abbes, M. Haddar, Electric vehicle design, modelling and optimization, *Mech. Ind.* 17 (2016) 405
- [18] T.J. Barlow, S. Latham, I. McCrae, P. Boulter, A reference book of driving cycles for use in the measurement of road vehicle emissions, 2009
- [19] Telecom Italia, JAVA Agent Development Framework, available from <http://jade.tilab.com>
- [20] G.E. Box, K. Wilson, On the experimental attainment of optimum conditions, *J. R. Stat. Soc. Ser. B: Methodol.* 13 (1951) 1–45
- [21] M.J. Powell, Radial basis functions for multivariable interpolation: a review, *Algorithms for approximation*, 1987, Clarendon Press, 143–167
- [22] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multi objective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.* 6 (2002) 182–197

Cite this article as: A. Guizani, M. Hammadi, J.-Y. Choley, T. Soriano, M. Slim abbes, M. Haddar, Multi-agent approach based on a design process for the optimization of mechatronic systems, *Mechanics & Industry* **18**, 507 (2017)