10th World Congress on Structural and Multidisciplinary Optimization
May 19 - 24, 2013, Orlando, Florida, USA

Uncertainty propagation in multi-agent systems for multidisciplinary optimization problems

L. Jaeger[1], T. Jorquera[2], S. Lemouzy[3], C. Gogu[4], S. Segonds[5], C. Bes[6]

1PhD student, jaeger@lgmt.ups-tlse.fr, ICA (Institut Clément Ader), Université de Toulouse, UPS; **2**PhD student, tom.jorquera@irit.fr, IRIT(Institut de Recherche Informatique de Toulouse), Université de Toulouse, UPS; **3**Post-doctoral, lemouzy@irit.fr, IRIT, Université de Toulouse; **4**Assistant professor, christian.gogu@univ-tlse3.fr, ICA, Université de Toulouse, UPS;**5**Assistant professor, stephane.segonds@univ-tlse3.fr, ICA, Université de Toulouse, UPS; **6**Professor, christian.bes@univ-tlse3.fr, ICA, Université de Toulouse, UPS

1. Abstract

Because of uncertainties on models and variables, deterministic multidisciplinary optimization may achieve under-sizing (without design margins) or over-sizing (with arbitrary design margins). Thus, it is necessary to implement multidisciplinary optimization methods that take into account the uncertainties in order to design systems that are both robust and reliable. Probabilistic methods such as reliability-based design optimization (RBDO) or robust design methods, provide designers with powerful decision-making tools but may involve very time-consuming calculations. New optimization approaches have been developed to deal with such complex problems. Auto-adaptive Multi-Agent Systems (AMAS) is a new approach developed recently, allowing to take into account the various aspects of a multidisciplinary optimization problem (multi-level, computation burden etc.). This approach was suggested for solving complex deterministic optimization problem. Now, the question of the integration of uncertainties in this multi-agent based optimization arises. The aim of this paper is to propose a new methodology for integrating the treatment of uncertainties in an adaptive multi-agent system for sequential optimization. The developed method employs a single loop process in which cycles of deterministic optimization alternate with evaluations of the system reliability. For each cycle, the optimization and the reliability analysis are decoupled from each other. The reliability analysis is carried out at agent level and only after the resolution of the deterministic optimization, to verify the feasibility of the constraints under uncertainties. Following the probabilistic study, the constraints violated (with low reliability) are shifted to the area of feasibility by integrating adaptive safety coefficients whose calculations are based on the agent-level reliability information. The method developed is applied to a conceptual aircraft design problem.

2. Keywords: Optimization, uncertainty, multi-agent, security coefficient, multidisciplinarity.

3. Introduction

The models which are used in preliminary design are often derived from empirical equations or simplified physical models. Modeling uncertainties generated by the models are prevalent and the resulting system following a deterministic optimization may not achieve the required performance when subjected to a more detailed analysis in the following phases of the design. Therefore it is necessary to implement methods taking into account the uncertainties since the early stages of design optimization in order to design systems that are both reliable and robust and thus ensure the success of the project on time and budget. Probabilistic methods such as reliability-based design optimization (RBDO) and robust design methods [1] provide designers with powerful decision-making tools but imply sometimes very time-consuming calculations, since the optimization procedure involves a double loop: iteration of the deterministic optimization process followed by a probabilistic analysis. Indeed, in order to determine the probabilistic characteristics of the system performances at a design point, it is necessary to perform some analysis, either by using approaches such as Monte Carlo simulation, importance sampling, subset simulations, or by using various analytical methods for probabilistic analysis as FORM, SORM. Many studies have been conducted before on this basis leading to the development of efficient technics with low computational time and to take into account the uncertainties in the design process of systems engineering.

Alongside these developments, new optimization approaches have also been proposed to deal with large industrial problems. Indeed, nowadays, systems are increasingly complex and design methods deployed involve aspects of multilevel and multi-disciplines [2]. The classical optimization methods can be difficult to implement in order to address these complex problems for which the number of variables and models is too high [3]. The difficulties of solving such optimization problems have motivated the development of new methods based on the adaptive multi-agent system (AMAS) approach. The multi-agent approach has been especially proposed for solving complex problems such as aircraft preliminary design in which we are particularly interested in this paper. The question of the integration of uncertainty in the multi-agent framework arises. Our work attempts to give a first answer to this question by proposing a new methodology for taking into account uncertainties in a sequential multi-agent optimization problem.

This paper is organized as follows. In the section 4, we explain the developped reliability design optimization method which is based on a single loop calculations: in a cycle, a deterministic optimization is solved, then a probabilistic analysis is conducted based on the determination of the reliability constraints. The violated constraints will be shifted with the introduction of a global safety coefficient. After having first explained the principle of solving a deterministic optimization problem by multi-agent method in section 4.1, we explain the proposed method for taking into account uncertainties in a multi-agent system in sections 4.2 and 4.3. In the section 5 we apply the developed method for managing uncertainty on a preliminary aircraft design test case. Finally, conclusions and perspectives will be presented.

4. Methodology for managing uncertainties in a problem solving by multi-agent

4.1. Principle for solving a deterministic problem by a multi-agent system

Nowadays, most of design problems are complex multidisciplinary optimization problems involving both the optimization of design parameters (masses, geometries, etc..) and performance optimization (for exemple the maximum takeoff length in the case of an airplane). The use of appropriate methodologies of MDO type (Multidisciplinary Design Optimization) facilitates the convergence and reduces the computational time to obtain a performant optimal solution. In the current work conducted within the ID4CS project (Integrated Design for Complex System), a specific theory called Adaptive Multi-Agent System (AMAS) is applied to carry out the solving of an MDO problem. The goal of the project is to develop a system which permits to take into account all MDO aspects and help engineers in solving complex optimization problems, with applications being implemented in the domain of aeronautical design.

The AMAS system is a system composed of several autonomous entities called agents. An agent has one or more local targets and interacts with other agents to meet its objectives. Studies have shown that this type of system can be used for solving optimization problems [3].

We explain the principle of a multi-agent system with a simple test case: a study case inspired by Alexandrov [4]. This study case 1 is representative of a multidisciplinary optimization problem (feedback loops between the disciplines):

$$\begin{cases} min_{l_1,l_2,s}\frac{1}{2}(a_1{}^{\mathbf{2}}+10a_2{}^{\mathbf{2}}+5(s-3)^{\mathbf{2}}) \\ s-10l_1-1\leq 0 \\ -s+l_2+2\leq 0 \end{cases} \tag{1}$$

with $a_1 = \frac{l_1-a_2}{2}$ and $a_2 = \frac{l_2-a_1}{2}$. This problem can be decomposed as shown on figure 1 (It's one of the possible solution, it could be decomposed in different ways).

Each of the elements introduced in the multidisciplinary optimization problem can be encapsulated in an independent entity: an agent. There are different types of agents (see figure1):

- a model agent: represents the model of a discipline. A model agent takes the values of its variable agents connected as input (i.e. l_1 and a_1 for m_1) in order to compute one or several output values. Those values are sended to the corresponding agents connected as output (i.e. a_1 for m_1). In order to satisfy the requests receved from its output variables, the model agent is able to determine the necessary modification of its input variables (by any mean, such as an internal optimisation algorithm). These modifications are then requested to its corresponding input variables.

- an objective agent: represents an optimisation objective (minimise or maximise a value of the problem, i.e. o_1). This kind of agent is a specialisation of a model agent where there is only one ouput value. The agent sends directly some requests to its outputs in order to minimise or maximise this output value.

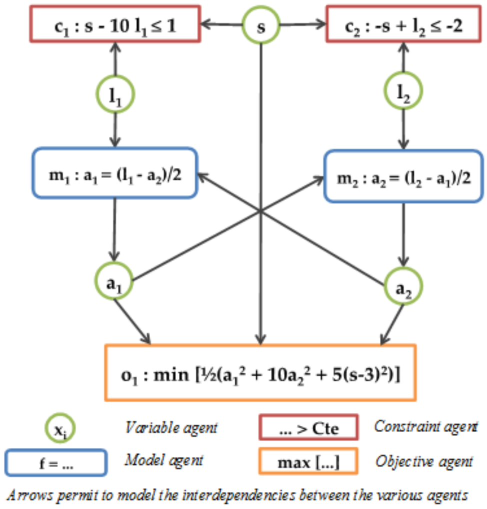


FIGURE 1 – Decomposition of Alexandrov test case

- a constraint agent: represents a constraint of the MDO problem (i.e. c_1). Like the objective agent, it is a specialisation of a model agent and it computes only one ouput value. It acts in order to keep this value greater or lower than a constant value (generaly 0). The agent sends directly requests to its input variables in order to achieve this goal.

- a variable agent: represents the variables of the problem. A variable agent can be connected as an input of one or several model, constraint or objective. In this case, the variable provides its value which enable connected agents to figure out their own ouput values. A variable agent can also be connected to one (and only one) model agent as an output. Its value is then the output value of the model agent. When a variable agent is only connected as an input, it is able to satisfy any value modification request by directly changing its internal value. When the agent is connected as an output of the model m, and because it's own value depends on the value computed by m, it will be only able to change its value indirectly by sending a request to m.

As the function of the different agents was introduced, we can briefly introduce now the resolution approach for solving multidisciplinary optimization problem through a multi-agent system. In this simple example it is assumed that the variables and the input parameters have arbitrary initial values. We also assume that the objective-function agent (in which the performance to minimize is encapsulated) decides based on some internal decision algorithm to ask the variable a_2 to take another value in order to improve the system performance. The approach followed by the multi-agent system is now explained. a_2 is a variable associated with a model, it can not choose to change its own value, it must forward the request to the model. A local optimization is then performed at the level of the model agent to meet this demand. The optimizer returns the values of the model (a_1 and l_2) which are solutions of the problem. Then the model agent sends the necessary queries to its inputs. To simplify the explanations we assume that l_2 has the correct value (therefore no query is sent). Because the variable a_1 can not change the value itself, a query is sent to its model. The model uses the optimizer (which may be different from the one previously used) to process the request. The optimizer returns a value for a_2 and l_1 (again, we chose to simplify the example, a_2 has the required value). The model sends a message to the entry l_1. l_1 is a parameter, it may itself decide to change its value. It accepts the request and informs the entities related to him that there is a change. The model can now calculate the new value of a_1. As a_1 changes its value, it informs the entities related to it. The model can now calculate the new value of a_2. Since the value of a_2 is changed, the model agent informs the entities related to him that there is a change. Thus, the model must calculate a new value for a_1, etc.. This behavior is repeated up to a stopping criterion, generally up to stabilization of the system.

4.2. Sequential optimization methods implemented

To incorporate uncertainty management in an optimization problem solved by this multi-agent system, we opt in this paper for a single loop strategy. We adapt classical methods based on adaptive safety factors [5]. The implementation approach has the advantage of being easily coupled to the logic of a multi-angent resolution as discussed in more detail in this section. We present below a description of this approach.

In a probabilistic design, most calculations are performed in order to allow the assessment of the system reliability. Therefore, to improve the overall efficiency of the probabilistic optimization we need to minimize the number of reliability evaluations. To achieve this goal, we use a single loop method: a series of cycles of deterministic optimization and reliability assessment. Each cycle of the optimization process includes two decoupled analysis: a deterministic optimization and an analysis of the reliability of the system at the deterministic optimum point. The reliability study permits to verify if each constraint is met with a certain level of reliability which is imposed. In the approach we propose, the optimization problem is solved by a multi-agent system (see 4.1) then the reliability of the obtained optimum is measured (analytically or by Monte Carlo simulations according to the problem being addressed). If the performance is not achieved, we introduce a global safety factor in the calculation of the limit states, so that the constraints are met with a level of reliability at least equal to 90%. Then, we start a new determinist optimization cycle from the last obtained point (used for uncertainty propagation). We can notice that it is possible that the new constraints (with safety factors) are not satisfied at this point of the search space. If it could be a problem for standard optimisation methods like Pattern Search, the mutli-agent optimization approch used in this study doesn't need such kind of constraint satisfaction for the starting point. This method requires fewer iterations of the optimization process and reliability assessments in order to converge, thus making the process more efficient than double loops reliability methods. This process of probabilistic optimization problem solving is detailed in the following sections.

The proposed approach begins with a deterministic optimization cycle. In mathematical terms, the optimization can be expressed as follows [6]:

$$\begin{cases} min_x f(x) & x \in S \\ g_i(x) \leq 0, & i = 1, ..., p \end{cases} \tag{2}$$

with S the search space. Following this deterministic cycle, uncertainties are incorporated by introducing the reliability problem formulation:

$$\begin{cases} min_x E[f(x,\delta)] & x \in S \\ Prob(g_i(x,\delta)) \leq P_{f_i}^{g_i} & i = 1, ..., p \end{cases} \tag{3}$$

We assume that the random variables are gaussian or can be approximated by normal distributions. Once the optimum point associated with the equation 2 is determined, the reliability of the different constraints is measured by $Prob[g_i(x,\delta) \leq 0] \quad i = 1, ..., p$ where δ is the random vector representing the uncertainties involved in the problem. The reliability analysis of the system allows the proposal of coefficients (called safety factors) which will be used in a deterministic approach to ensure a certain level of reliability during the design phase for the system holding [7]. The use of the safety factor is a common practice in optimization [5].

We implement a first sequential method based on two levels. The solving of the reliability based optimization is composed of two independent steps. The first one searches the deterministic optimal parameters in the physical space of the optimization variables and the second one allows to incorporate the uncertainties by solving the reliability problem given by the equation 3. The transition between the deterministic optimization and the reliability based optimization is performed once detecting a change in the constraints and in the objective function below a certain treshold Δ during a number n of consecutive steps.

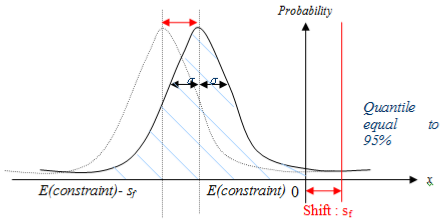
A second more advanced approach is also proposed. This technique is inspired by the sequential method SORA [8].Our approach is based on the use of global safety factors [5], which lead to a better integration in the multi-agent solving approach developed in the ID4CS project. In the developed method we employ a single loop process [9] wherein a series of cycles of deterministic optimization and reliability system evaluation succeed each other. For each cycle, the optimization and the reliability are decoupled and the evaluation of the reliability is only carried out after the resolution of the deterministic optimization problem in order to verify the feasibility of the constraints under uncertainties. When the reliability constraints of the optimization problem are not satisfactory, we introduce an adaptative safety factor in the calculation of the limit states, so that the constraints are met with a level of reliability equal to a target value (in our case this value is often taken as equal to 90%). In our study, the global safety coefficient s_{f_i} is determined under the assumption of normality as follows:

$$\begin{cases} Prob[g_i(x,\delta) - s_{f_i}] \leq \alpha_i \\ s_{f_i} = E[g_i(x,\delta)] + \Phi^{-1}(\alpha_i)\sigma_{g_i} \end{cases} \tag{4}$$

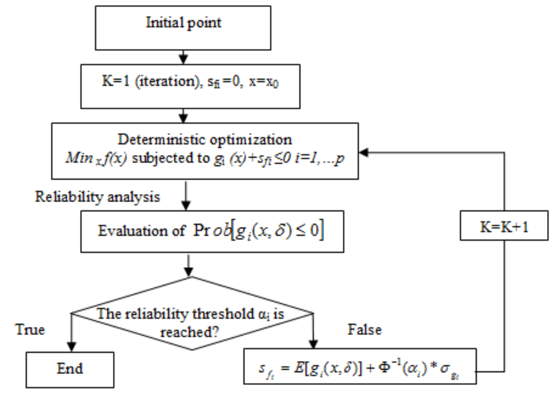
where α_i is the required level of reliability and σ_{g_i} the standard deviation of the studied constraints. An illustration is given in 2a. The new agent-level search point of the next optimization cycle $K+1$ is then the last evaluated search point which was used for updating the safety factors $s_{f_i}^{K+1}$. We consider that the algorithm has converged to a solution when the evolution of all the s_{f_i} is becoming relatively small, that is to say:

$$|s_{f_i}^{K} - s_{f_i}^{K+1}| \leq \epsilon, \forall i \tag{5}$$

where ϵ is the convergence detection threshold. A flowchart of this method is given in figure 2b.



(a) Modification of the constraint



(b) Flowchart of the reliability optimization method

Figure 2

For the first cycle (K = 1), the value of the current point is arbitrarily imposed and the safety factor is equal to zero. Following the resolution of the equation 2 (deterministic optimization), some constraints can be active. For an active constraint g, the optimum point X is located on the boundary of this deterministic constraint. When considering the random nature of X, the current reliability (probability that the constraint are respected) may be inferior to the imposed threshold. Thus, if after assessing the reliability, the deterministic optimum does not meet the required reliability threshold, we modify the deterministic constraints by introducing a global safety factor to ensure the feasibility of the constraints. For the next cycle ($K = 2$), we start a new deterministic optimization where each constraints g_i are updated with a new safety factor $s_{f_i}^{K}$. This optimisation starts from the last evaluated point of the search space. These steps are repeated until convergence. The sequential single loop method requires fewer iterations of the optimization process and reliability assessments, in order to converge (no reliability assessment at each current point of the optimization), thus making the process more efficient than double loop reliability methods.

4.3. Agent level uncertainty propagation and management

Since a multi-agent system is a set of agents operating in a common environment, our goal is to propagate uncertainties through this set of agents, from the inputs to the outputs and define a strategy permitting the multi-agent system to satisfy the new constraints which are formulated in terms of reliability and robustness. Uncertainties can be of two different types: the modeling uncertainty and the parameter uncertainties. Modeling uncertainties are due to the difference between the model predictions and the reality. Another potential source of uncertainty in the outputs of our system comes from the uncertainty in the input parameters. In preliminary aircraft design for example, these variations can result from the differences between the idealized models which are used and the final design system at the end of the various design phases.

To propagate uncertainties we integrate additional information in each agent. The random variables can be represented in different ways: by a cloud of sample points or by their probability distribution. To keep a relatively generic approach for uncertainty representation we have chosen the beta-mystique distribution developed at Airbus which allows to represent through four parameters the most commonly used probability laws: the uniform law, the normal law, the Gumbel distribution and the triangular distribution as well as possible variations in these laws. We describe now how to manage the uncertainties for each type of agent. There are different types of agents that are involved in the propagation:

- The model agent: a model representing a certain physical system is encapsulated. The agent takes a vector of random parameters x_c as input. Each random parameter x_i component of the vector x_c is characterized by its mean x_{m_i} at the current optimization point and its random characteristics (eg its standard deviation σ_{x_i} or the four parameters of the beta-mystique law, etc.). The output of the agent is then written: $f(x_i) + \delta_{model}$ where δ_{model} is the modeling uncertainty. The uncertainty δ_{model} is characterized by one of the representation given above (a cloud of sample points, and various probability laws can be for example represented thanks to the beta-mystique law) and obtained by the uncertainties propagator associated with this representation. These can be either exact propagators in simple cases, or propagators that give an approximation based on first order calculation. It is then sent to the rest of the multi-agent system (variable agents taking as input the agent output: the consumers agents).

- The objective-function agent: the specific objective of this agent is to minimize the objective function of the system (this agent encapsulates the performance function of the optimization problem). It ensures that the obtained performances are consistent with the requirements of the problem. The agent takes as input a vector of random parameters x_c with their associated uncertainty representation. The output of the agent is $E[f(x_i)]$. The mean of the function f is calculated on the basis of the results of the uncertainty propagation which also permits to determine the standard deviation of the objective function (we can then obtain information about the robustness of the system).

- The constraint agent: the constraint agent can determine the level of reliability reached by each constraint. It takes as input a vector of random parameters x_c. Each random component x_i has an associated uncertainty representation (cloud of sample points or probability distribution). The agent checks internally whether the constraint is satisfied by measuring the level of reliability. This probability can be written as $Prob[g_i(x_i) \leq 0]$ and obtained directly from samples for an uncertainty representation by a cloud of sample points or it can be calculated by $Prob[g_i(x_i) \leq 0] = \Phi(-\frac{\mu_{contrainte}}{\sigma_{contrainte}})$ under the hypothesis of normality for uncertainties represented by probability distributions. For non-Gaussian variables, such expressions may still represent a good approximation for the reliability evaluation, considering the central limit theorem and the fact that for complex problems the constraint agent usually depends on a large number of uncertainty source. A normality test can be implemented prior to the estimation of the probability in order to test this hypothesis. If the constraints do not meet reliability thresholds which are imposed α_i, a global safety factor s_f is introduced (see equation 4) in order to take into account the dispersion of the constraint (see figure 2a): $Prob[g_i(x_i) - s_{f_i}^K \leq 0] = \alpha_i$ where α_i is the required level of reliability. The new search point for the following optimization cycle is then the last point used for reliability study. As already said before, it is possible that the safety factor implies that the new constraints are not satisfied at this point of the search space. If it could be a problem for standard optimisation methods like pattern search, the mutli-agent optimization approch used in this study doesn't need such kind of constraint satisfaction for the next optimization cycle. The interactions defined in the constraint agent are shown on figure 3.

Note that when uncertainties are modeled by normal laws characterized by variance σ and mean μ, the uncertainty propagation can be simplified. According to the available information at the output of the agent (accessibility of the gradient or the structure model), the uncertainty propagation is different. Then, we differentiate two cases:

- The encapsulated model is linear or quadratic: mean and standard deviation of the agent output are determined analytically.

- The encapsulated model is ordinary: if the gradient is available or can be evaluated the propagation is performed analytically using a first order approximation. Otherwise, we use Monte Carlo simulations in order to define the characteristics of the uncertainties on the agent output.

In this section a method for managing uncertainty for solving optimization problems by multi-agent system was introduced. In the next section we set up a test case to validate the proposed approach. Thus,

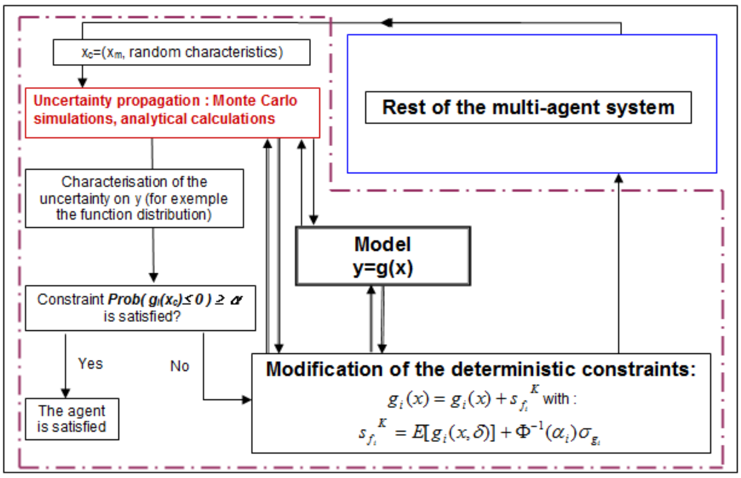


Figure 3: Uncertainty propagation through a constraint agent

the methods described in this section are applied in section 5 in the context of a preliminary aircraft design problem.

5. Preliminary aircraft design test case

5.1. Presentation of the problem

The preliminary aircraft design problem that we consider (inspired from [10] [11]) involves the ratio power/weight $\frac{P_{TO}}{m_{TO}}$ and the wing loading $\frac{m_{MTO}}{S_W}$ as aircraft variables. The objective of this problem is to minimize the weight of the aircraft at takeoff m_{MTO} and get the lowest possible power/weight ratio and the highest possible wing loading. The deterministic optimization problem is formulated as follows:

$$\begin{cases} Min_{\frac{m_{MTO}}{S_W},\frac{P_{TO}}{m_{MTO}}} m_{MTO}(\frac{m_{MTO}}{S_W},\frac{P_{TO}}{m_{MTO}}) \\ g_i(\frac{m_{MTO}}{S_W},\frac{P_{TO}}{m_{MTO}}) \leq 0 \qquad\qquad i=1,2,3 \end{cases} \tag{6}$$

where the constraints g_i are only the active constraints: the cruise speed(block 1, figure 4a), the landing field length (block 2) and the takeoff field length (block 3).

We first determine the optimum point of our problem without uncertainty through a standard algorithm. The algorithm used is of Pattern Search type, which provides the optimum in 110 evaluations of the objective function: $\frac{m_{MTO}}{S_W} = 377kg/m^2$; $\frac{P_{TO}}{m_{MTO}} = 187W/kg$ and $m_{MTO} = 20945kg$.

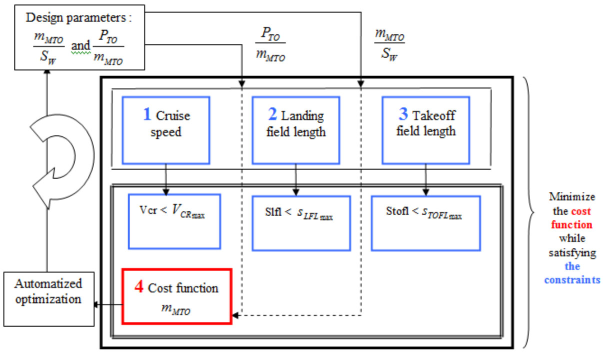
The preliminary design is a design phase which proves to be crucial but complex for several reasons: decisions are made in a context where few things are defined and the data of the problem are still poorly known. The search space of the solutions must remain general enough to not exclude potential solutions, and finally the models used are often relatively coarse. At this stage of the design, uncertainties are significant and are mainly modeling uncertainties. We wish to design a system weakly sensitive to changes and having statistically the best performance. To this end the deterministic problem is modified to include reliability constraints. The problem is as follow:

$$\begin{cases} Min_{\frac{m_{MTO}}{S_W},\frac{P_{TO}}{m_{MTO}}} E[m_{MTO}(\frac{m_{MTO}}{S_W},\frac{P_{TO}}{m_{MTO}},\delta)] \\ Prob[g_i(\frac{m_{MTO}}{S_W},\frac{P_{TO}}{m_{MTO}},\delta) \leq 0] \geq 90\% \qquad i=1,2,3 \end{cases} \tag{7}$$

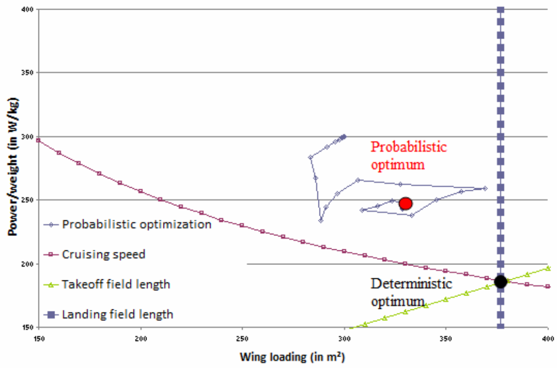
Uncertainties are introduced on the following parameters: the maximum lift coefficient during landing $C_{L,max,L}$, the maximum lift coefficient during the takeoff phase $C_{L,max,TO}$, the lift to drag ratio E and the propeller efficiency $\eta_{P,CR}$. These random variables were chosen on the advice of experts.

5.2. Result obtained by the multi-agent method

The deterministic problem defined by the equation 6 is solved by a multi-agent method, the optimum is represented by a black dot on the figure 4b.



(a) Description of the simplified conceptual aircraft phase



(b) Resolution of the probabilistic optimization by Monte Carlo simulation with the multi-agent method

Figure 4

These results are validated by comparison with a standard optimization method (see table 1). The results obtained through the two different methods are close, however there is a difference in the number of iterations required to achieve convergence: the multi-agent method requires 42 iterations while the classical optimization algorithm (Pattern Search) requires 110 iterations, or about 3 times more. The convergence is considered as reached once detecting a change in the constraints and in the objective function below a certain treshold during a number m_s of consecutive steps.

Method	**Multi-agent**	**Pattern Search algorithm**
Results of the deterministic optimization $(m_{MTO}; \frac{m_{MTO}}{S_W}; \frac{P_{TO}}{m_{MTO}})$	(20948 *kg*, 377 *kg.m*²; 187 *W / kg*)	(20945 *kg*, 376 *kg.m*²; 187 *W / kg*)
Number of evaluations	42	110

Table 1: Comparison of the results between the multi-agent method and the classical optimization in the case of the deterministic optimization

We now take into account the uncertainties and solve the problem given by equation 7. Different cases are implemented in order to test the developped methods. Thus the uncertainties are taken into account by:

- Monte-Carlo simulations (50 000 draws are done), the variables are modelized by normal distributions with constant mean and standard deviation which are specific to each uncertain variable.
- Analytic calculations, the mean and the standard deviation of each random variable are determined analytically. We assume that all the random variables are uncorrelated. This approximation will be verified by implementing the same method but with Monte Carlo simulations.

The different results obtained are presented in table 2a and compared to a classical optimization method in table 2b.

In the various studied cases we observe that the multi-agent approach provides solutions close to those obtained by a conventional optimization method. Nevertheless, we can notice a slight difference between the results obtained through the two algorithms for the propagation of uncertainties by Monte-Carlo simulations when they are modeled by log-normal law. Indeed the Pattern Search algorithm does not provide stable results (for the same initial point) as opposed to multi-agent. The Pattern Search algorithm is sensitive to noise caused by the Monte-Carlo draws, and this sensitivity to uncertainty is even more important when the variables follow a log-normal law. We also note that the assumptions made in order to perform an analytical uncertainty propagation are validated. Indeed, in the table 2a we can see that

Multi-agent method	Monte Carlo simulations, gaussian random variables	Monte Carlo simulations, log-normal random variables	Analytical calculations, log-normal random variables
Reliability optimum $(m_{MTO};\frac{m_{MTO}}{S_w};\frac{P_{TO}}{m_{MTO}})$	(21313 kg; 328.7 kgm²; 244W / kg)	(21276 kg; 333 kg.m²; 238W / kg)	(21276 kg; 333 kgm²; 238W / kg)
Number of iterations	41	42	44

(a) Results obtained by the multi-agent method for the reliability optimization

Pattern Search algorithm	Monte Carlo simulations, gaussian random variables	Monte Carlo simulations, log-normal random variables	Analytical calculations, log-normal random variables
Reliability optimum $(m_{MTO};\frac{m_{MTO}}{S_w};\frac{P_{TO}}{m_{MTO}})$	(21311kg; 328kgm²; 244W / kg)	(21324 kg; 332kgm²; 244W / kg)	(21281 kg; 332kgm²; 239W / kg)
Number of iterations	406	362	214

(b) Results obtained by a classical algorithm for the reliability optimization

Table 2

we get the same results when the uncertainties are propagated analytically or by Monte-Carlo simulations (the uncertainties are modeled by log-normal law).

The multi-agent approach appears to significantly reduce the number of simulations. On the other hand, whatever the initial selected point, the multi-agent method also appears more robust because it permits to obtain the same optimum point each time, which is not the case for the Pattern Search algorithm.

When we compare the deterministic optimization results with those obtained for the reliability optimizations we see an increase in the cost function (see table 1 and figure 4b): the takeoff weigth is more important in the case of the reliability optimization (normal tendency when setting up a reliability and/or robust optimization problem: the presence of margins in reliability optimization causes the moving of the optimum). The constraints represented in 4b are deterministic constraints. The takeoff weigth of the airplane increased but we have taken into account the uncertainties impacting the parameters which generate the greater sensitivity of the objective function and performances, we are now able to guarantee a design satisfying a certain level of reliability (the active constraints have a reliability threshold equal to 90%) in contrast to the deterministic optimization.

In this first solution of the preliminary design problem, probabilistic uncertainties are taken into account from the beginning of the optimization. Thus, if the initial point of the optimization algorithm is far from the reliability optimum and when Monte-Carlo simulations are required, the management of uncertainties can be very expensive. So, we now want to implement the previously developed method using a single loop procedure in which a series of cycles of deterministic optimization and reliability assessments of the system succeed each other. This method is thus applied and compared with a two loop sequential method. The results are given in table 3. As for the analytical test case, the optima which are obtained by the two methods are close and despite of a higher number of deterministic evaluations, the sequential single loop method is more efficient because it requires only three reliability assessments.

Multi-agent method	Double level sequential optimization method	Single loop sequential optimization method
Reliability optimum $(m_{MTO};\frac{m_{MTO}}{S_w};\frac{P_{TO}}{m_{MTO}})$	(21311kg; 328kgm²; 244W / kg)	(21303kg; 331kg.m²; 242W / kg)
Total number of evaluations	2 300 036	150 216
Number of deterministic evaluations	36	216
Number of reliability evaluations	46	3

Table 3: Sequential optimization results

The developed methods were used to take into account the uncertainties of different natures (modeling uncertainties and parameter uncertainties) and in order to set up a system having statistically the highest level of performance in a multi-agent method. The multi-agent system is very promising for solving complex problems (multidisciplinary problem with interrelated disciplines). This resolution can be very complicated when implementing a standard optimization method (computation time, important problems of convergence etc..).

6. Conclusion

Nowadays systems to optimize are more and more complex: the number of parameters to be determined is high, sometimes several interrelated disciplines are involved, as well as different levels of granularity. The classical optimization methods (algorithm based on the calculation of gradients, research methodology etc..) can thus be difficult to implement in order to solve such complex problems. This motivates the development of new methods to address optimization problems with a large number of variables which may have multi-level and multi-disciplines aspects. Methods based on the use of adaptive multi-agent systems (AMAS) are interesting alternatives to address this problem. The question of the integration of uncertainties in the multi-agent problem-solving approach arises. In this paper we detail a new methodology we developed which allows the consideration of uncertainties in a sequential multi-agent optimization problem. Thus we have proposed several methods for managing uncertainty, based on the calculation of adaptive safety factors. The proposed approach solves the problem sequentially (deterministic resolution then optimization under uncertainty) in order to accelerate the convergence in most cases. Finally, the last proposed method reduces even more the computation time by developing a sequential approach with adaptive criterion for assessing the level of reliability. These three approaches have been tested and validated on a preliminary aircraft design test case. Future work will be dedicated to test our method on a more detailed preliminary aircraft design.

References

[1] Hans-Georg Beyer and Bernhard Sendhoff. Robust optimization – a comprehensive survey. *Computer Methods in Applied Mechanics and Engineering*, 196(33-34):3190–3218, August 2007.

[2] J. Sobieszczanski-Sobieski and R. T. Haftka. Multidisciplinary aerospace design optimization: survey of recent developments. *Structural Optimization*, 14(1):1–23, August 1997.

[3] Jean-Baptiste. B. Welcomme, Marie-Pierre Gleizes, and Romaric Redon. A self-organising multi-agent system managing complex system design application to conceptual aircraft design. *System and Information Sciences Notes (SIWN)*, 1(2):208–221, November 2009.

[4] N. Alexandrov and R. Lewis. Analytical and computational aspects of collaborative optimization. Technical Report TM-2000-210104, NASA, April 2000.

[5] Y. T. Wu, Y. Shin, R. Sues, and M. Cesare. Safety-factor based approach for probabilistic-based design optimization. Seattle, Washington, 2001.

[6] R. Le Riche, O. Roustant, X. Bay, and G. Pujol. Formulation de l'optimisation avec incertitudes dans le projet RNTL/OMD. Technical report, Ecole des Mines, Saint Etienne, 2007.

[7] M. Pendola. *Fiabilité des structures en contexte d'incertitudes statistiques et d'écarts de modélisation.* PhD thesis, Université Blaise Pascal, Clermont II, Clermont-Ferrand, 2000.

[8] Tae Min Cho and Byung Chai Lee. Reliability-based design optimization using convex linearization and sequential optimization and reliability assessment method. *Structural Safety*, 33(1):42–50, January 2011.

[9] Xiaoping Du and Wei Chen. Sequential optimization and reliability assessment method for efficient probabilistic design. *Journal of Mechanical Design*, 126(2):225, 2004.

[10] D. Scholz and M. Nita. Preliminary sizing of large propeller driven aeroplanes. pages 1–19, Brno, Czech Republic, October 2008.

[11] L. Jaeger, C. Gogu, S. Segonds, and C. Bes. Aircraft multidisciplinary design optimization under both model and design variables uncertainty. *Journal of Aircraft*, pages 1–11, February 2013.