

# GESOS: A MULTI-OBJECTIVE GENETIC TOOL FOR PROJECT MANAGEMENT CONSIDERING TECHNICAL AND NON-TECHNICAL CONSTRAINTS

Claude Baron<sup>1</sup>, Samuel Rochet<sup>1</sup>, Daniel Esteve<sup>2</sup>

<sup>1</sup> LESIA, INSA, 135 av. de Rangueil, 31077 Toulouse cedex 04, France

<sup>2</sup> LAAS, CNRS, 7 av. du colonel Roche, 31077 Toulouse cedex 04, France  
claude.baron@insa-tlse.fr, rochet@insa-tlse.fr, daniel.esteve@laas.fr

**Abstract:** Project managers have a difficult issue to deal with: identify tasks to plan during project management, with their technical and non-technical parameters, determine a target to reach, and effectively reach it to avoid financial penalties. This paper presents a tool able to join system design and project management. We think thus facilitate the construction of an architecture of planning and the optimization of this one according to methods that we propose to develop. Our main motivation is to prevent the obvious incompatibilities between technical objectives and socio-economical requirements in the enterprise. Thus, we want to help decision makers to chose a project skeleton, called scenario, at the launching of the project, but also during its management, in order to quickly react in case of the occurrence of any perturbation. A method using evolutionary algorithms seemed adapted. We will see the benefits that result from this approach and concludes on the perspectives of larger applications we can envisage thanks to the tool that supports it, GESOS.

**Key words:** evolutionary computing, genetic algorithms, metaheuristic optimization, system design, project management, multi-criteria decision making

## 1. INTRODUCTION

Our work takes place in the domain of development of methodologies for the project management. The originality of our approach is to couple this domain with the one of system design. In a simplified way, the question is to

succeed in making simultaneously evolve the organization of the project and design. With the process of project management we associate some mechanisms of robustness and adaptability related to external disturbances of technical, social or economic nature, in order to respect the laid down objectives.

The problem thus becomes a question of minimization of the distance between the objective effectively achieved by the scenario (in terms of costs, times, quality...) and the laid down initial objective. However, the choice of the various tasks to be carried out during a project is a complex optimization problem. Only some heuristic methods can enable us to find a solution that is close to the optimum in a reasonable time, like evolutionary algorithms. Indeed, considering this problem, we could note that such methods seemed best adapted because these research algorithms allow taking into account multiple parameters of which they seek many combinations simultaneously. We validated our approach using a tool based on the use of the genetic algorithms, GESOS (Genetic Evaluation, Selection and Optimization of Scenarios).

## **2. GENERAL CONTEXT OF THE PROJECT**

### **2.1 Objective: reach the targets**

On one side, the objectives of a system design process consist in obtaining an exact conformance of the system to technical requirements: performances, quality, reliability, testability, etc. On the other side, the project manager has to deal with a project management process : define targets to reach (in terms of costs, duration, resources...), then define tasks and their associated means and suppliers, that have to be scheduled in order, precisely, to reach the target within the previously defined constraints : duration, means (financial, manpower, machines, suppliers, etc.).

So the project's objectives consist in reaching a group of technical or economical targets. In order to measure how precisely these targets are reached, we have decided to use the Taguchi's approach in which quality should be measured by the deviation from a specified target value. This concept will be developed in section 3.4.2.

### **2.2 The notion of scenario**

We are convinced that the first description level for tasks can be derived from the functional components description obtained at the preliminary

design step [Bar04b]. This operation is processed on the base of technical considerations, has to consider non technical objectives relative to the project management: costs, market, supply constraints, supplying delays, quality, certification, any types of risks...

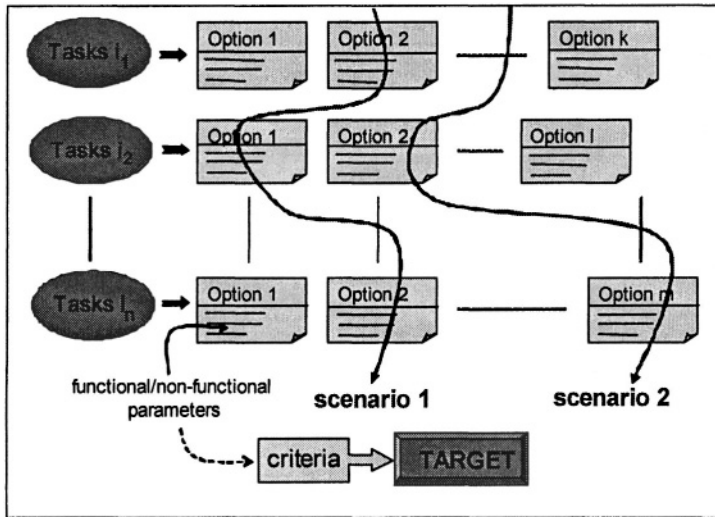


Figure 1. Generation of scenarios

The intersection between the tasks' content and the technical and non-technical objectives leads to establish several possibilities for the project organization which are coherent with general specifications of products, as illustrated on figure 1; we call them *scenarios*. Scenarios are deduced from the options attached to each task they correspond to global solutions respecting both technical specifications and strategic project requirements. In a simplified way, we can say that a scenario is a combination of options and the main point of this paper is to present a procedure to define the best appropriated combination.

### 3. THE TOOL GESOS

#### 3.1 Why an evolutionary choice?

Choosing various options associated to project's tasks is an optimization problem for which no exact polynomial algorithm is known. The use of an exact method of optimization is then not very realistic for large-sized problems; that's why we need to use a heuristic method. The evolutionary

algorithms seemed interesting for several reasons. First, the research algorithms are well adapted to multiple parameters of which they consider many combinations at the same time, which allows offering the decision maker a range of solution at each step. One thus lay out with each stage of calculation a unit of available solutions and not a single solution, as the method of simulated annealing or taboos do. It is an advantage in our case where a choice of solutions will have to be offered at the user. Second, they use a very simple criterion of evaluation by allocating a note to each individual according to its performance. This avoids using more complex mathematic tools like the gradient or the derivative, which are not easily usable or not very representative in a similar problem. Third, using an evolutionary approach is also interesting considering the possibility it offer to treat heterogeneous types of representations. Last but not least, the fact that only one mark by individual is necessary in the evolutionary process makes the evaluation and optimization procedures independent. One can thus imagine implementing several types of evaluation and let the decision maker chose one, or try and compare different ones...

## **3.2 Project representation**

Our work take place in a more ambitious project [Bar04a] where several research and industrial partners are involved. For instance, data for GESOS will provide from two other programs developed at the LAAS. The first one, HILES [Ham03], realizes a product formal description and the second, LORA [Bri04] allows representing temporal relations between project's tasks and in particular the sequence and resources allocation. In a simplified way, we can consider that the information coming from HILES, which provides technical elements, completed by LORA's project management information, permits a complete representation of the project. GESOS will use these resources to construct one complete project representation before determining the optimal scenario. After this step of optimization, solutions can be re-introduced at HILES and LORA in order to permit the decision maker to respectively simulate the operations of the product and project scheduling.

The first constraints types treated by GESOS are the temporal relations between tasks and more precisely the precedence relations between them which traduce the case, when one action can't be done before the end of the others. It's the main relation type behind the project tasks. We represent tasks to realize by numbered squares and precedence relations by arrows. We can with this representation use a graph to visualize task's sequencing.

This last relation allows treating classical problems of scheduling in which project structure doesn't have any alternatives. But in our problematic,

options associated to each tasks are unknown as well as the project architecture. For instance, the choice of one material affects the machining steps and only machining steps compatible with this one can be used. That's why we need to introduce one element representing a choice in the project sequence; it's what we call decision nodes. To represent the choices that appear in the project we add two numbered circles into the tasks graph. Each one represents a node and it's the ensemble of these two nodes which forms a decision. Thus, we have an initial node and a final node. The first circle on figure 2 with a double line is the initial node. Only one way will be activated after one initial node, all the others will be deactivated. The final node has a single line and all the relations after him must be realized.

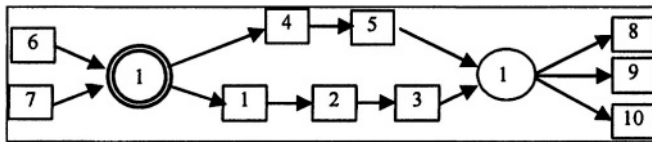


Figure 2. Project's structure with decision nodes

Thus, only one task sequence is possible between an initial and final node. For example, on figure 2 two ways are possible. As showed on figure 3, the tasks 1, 2 and 3 can be active and 4 and 5 inactive or tasks 4 and 5 actives and 1, 2 and 3 inactive.

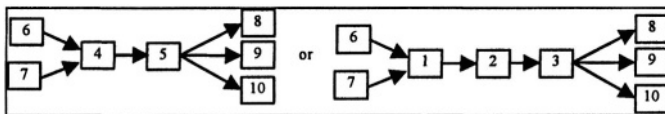


Figure 3. Possible Structures issues to Figure 2

Nodes allow the representation of the mutual exclusion of certain tasks by others and we can then combine nodes to represent complex scenarios structures.

### 3.3 Individual coding

This section presents the individual representation. The choice of an appropriated representation allows obtaining an efficient operation of the algorithm.

Our objective is to determine the best way into the options to reach the fixed target. This means select for each task an option and for each node a way. Parameters linked to tasks and options being determined, remain two

object types which can vary into the individual selected: to each task, the option selected and to each node, the decision chose. Our coding is very simple, on the first chromosome part will be coded the options associated to each tasks and on the second chromosome part, the way of each nodes as showed on figure 4.

For a problem with  $n$  tasks with  $i_n$  options each and  $m$  nodes with  $i_m$  choices, on the  $n$  first genes, gene  $i$  correspond to task  $i$  and on genes  $n+1$  to  $n+m$ , gene  $i$  correspond to node  $i-n$ . In each of themes, one integer between 1 and  $i_n$  or  $i_m$ , function of the case, does reference to the option or way chosen.

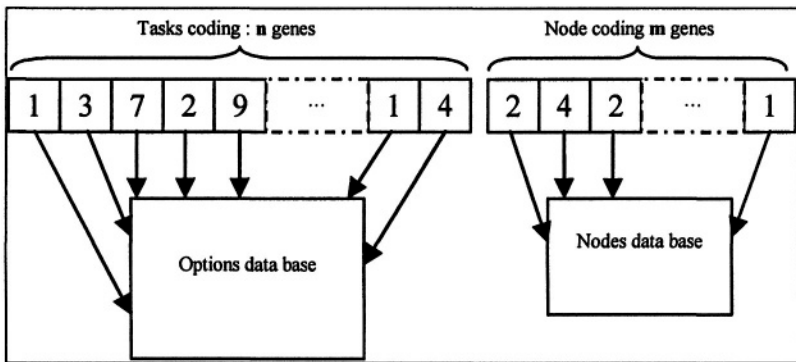


Figure 4. Chromosome signification

To decode each individual, we look into two data bases, the “options data base” and the “nodes data base”. The options data base contains all tasks with these own parameters and options parameters and, the nodes data base all information needed too define the way between each group of two nodes which corresponds to each project’s choice.

This representation using only integers allow to manipulated individuals with restricted size and can easily be decoded before the individual evaluation step. In addition, this coding allows using directly classical methods of crossover and mutation without reconstruct non-viable individuals. This is due to the fact that we use a coding in which an object does not depends of a gene chain but only of one gene; so a gene modification can not change all the meaning of the chromosome but only the object corresponding to the modified gene.

### 3.4 Genetic procedure

In this section will be described the genetic procedure implemented in GESOS. It’s a strictly classical algorithm. After a random population

initialization, we apply the classical sequence of evaluation, selection, crossover, mutation as shown on figure 5. The point requiring to be developed is the evaluation process. It will be explained in the following paragraphs. The other algorithms used, roulette wheel method for the selection, single point crossover and one gene mutation are the standards methods [Gol94] and don't need to be explain in details in this paper.

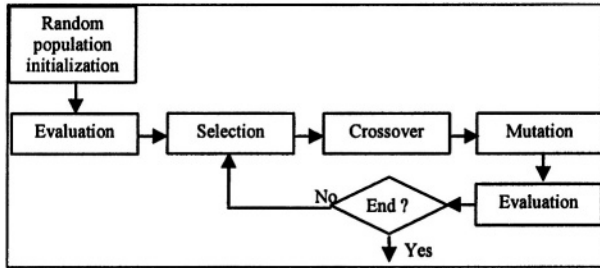


Figure 5. Genetic procedure

In the following paragraph we will detail the evaluation process. The different steps of its decomposition will be explained before the criteria used and to finish we will explain the method of fitness attribution.

### 3.4.1 Evaluation steps

The evaluation implies to:

- determine the individuals projects sequences starting from a complex project in which all tasks aren't activated
- and find an efficient multi-objective algorithm in order to valorize the apparition of good schemata and keeping the solutions diversity.

For that, the following steps are successively realized:

#### 1. Problem reduction :

For each individual, the choices done during the project are known during the evaluation. In function of the individual's chromosome, we know what options and ways have been chosen. However, in function of the choices linked to nodes, certain tasks or others nodes will be inactive. These ones will be absent from the scenario realized, that's why it's useless to keep them during the rest of evaluation. So, a second scenario is generated in which all inactive tasks and nodes are removed.

For instance, on figure 6 we can see the second scenario obtained of one chromosome. Stay only tasks 1, 2, 6, 7 and 8 with respectively the 3<sup>rd</sup>, 5<sup>th</sup>, 8<sup>th</sup>, 5<sup>th</sup> and 2<sup>nd</sup> of their associated options.

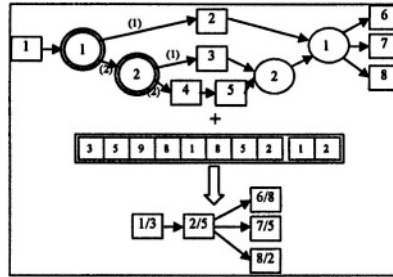


Figure 6. Scenario reduction

## 2. Determination of optimization parameters

Once the previous operation of problem reduction completed, we obtain a classical scheduling problem in which intervene only precedence relations between tasks and resources availability. In this case, we could easily determine a project sequence by classical methods, and by the same occasion evaluate parameters; this last point is detailed in the paragraph 3.4.2.

Presently, we use distance between cost and budget and distance between project duration and desired duration for parameters. The first is determined by the summation of the cost of each realized task. The second using a PERT method variant in which is introduce moreover tasks antecedence relations, constraints linked to resources availability.

## 3. Multi-criteria evaluation:

In our problem we try to minimize the distance of all parameters by report there respecting objectives (see 3.4.2); it's a typically multi-criteria problem. To make our selection, we have developed a method based on the Strength Pareto Evolutionary Algorithm. The selection process will be developed in paragraph 3.4.3.

### 3.4.2 Evaluation criteria

We have seen on paragraph 2.1 that the objective is to reach the technical and economical targets. In order to measure how precisely the targets are reached, we decided to use the Taguchi's approach. Taguchi methods, developed by Dr. Genichi Taguchi [War89], can be described in a basic idea: Quality should be measured by the deviation from a specified target value, rather than by conformance to preset tolerance limits.

Taguchi believes that the customer becomes increasingly dissatisfied as performance departs farther away from the target. He suggests a quadratic curve to represent a customer's dissatisfaction with a product's performance. The curve is centered on the target value, which provides the best performance in the eyes of the customer. This is a customer-driven design



rather than an engineer's specification. There is some financial loss incurred at the upper consumer tolerance. This could be a warranty charge to the organization or a repair expense for example.

This synthetic and very attractive approach invites us to represent in terms of costs every consequence of the product requirements. Indeed, originally, the Taguchi's loss function establishes a financial measure of the user dissatisfaction with a product's performance as it deviates from a target value. In our context, the project manager has to define a target considering technical as well as financial objectives. He thus has to simultaneously lead a product development and a pure project management processes.

Value of each optimization parameters will be the value of loss function associated to the objective which permits us to work on a minimization problem in which we determine individuals with the minimum loss function.

$$L = K(O - T)^2$$

With L loss function value, T target value, O objective value and K user parameter which permit to adapt each loss function to objective type.

### 3.4.3 Multi-criteria evaluation

Our problem is then to minimize the loss functions ensemble in order to obtain scenarios as near of the targets as possible. This evaluation must valorize the scenarios diversity too to allow us to suggest to the decision maker a wide range of solutions and not a group centered on the same scenario with only little variations. To realize this, we have chosen to use a method inspired by the Strength Pareto Evolutionary Algorithm. This technique presented in 1998 by E. Zitzler and L. Tiele ([Zit88] and [Zit89]) put down on the determination of the Pareto front associated to the problem and allows us to reach ours objectives of solution quality and population diversity.

During a mono-criterion evaluation, it's easy to compare two individuals by comparing a scalar fitness. In a multi-criteria optimization case, relations between individuals are more complexes. We can use the dominance notion. In a minimization problem, considering two objective vectors  $\mathbf{v}$  and  $\mathbf{u}$ . If all  $\mathbf{v}$  components are less or equal to  $\mathbf{u}$  components, with at least one component strictly less, so  $\mathbf{v}$  vector correspond to a better solution than  $\mathbf{u}$ . In this case, we tell  $\mathbf{v}$  dominate  $\mathbf{u}$  on Pareto sense. On formal way, we can write:

$$\mathbf{v} <^p \mathbf{u} \text{ if } \mathbf{v} <^p \mathbf{u} \Leftrightarrow \forall i \in [1; c], v_i \leq u_i, \text{ and } \exists j \in [1; c]: v_j < u_j$$

The whole of objective vectors which couldn't be dominated constitutes problem optimal values in Pareto sense. These values belong to the Pareto front. The Pareto-optimal whole is defined like unit which contain solutions in the search space with values are into the Pareto front.

We work with two sets of individuals. P, the population, and P', which contain Pareto front individuals. Individual fitness determination is decomposed in two steps:

*Step 1:* at each P' individual is associated a value representing there strength equal to the number of solutions dominate by i in population P divided by P height plus one (1).

*Step 2:* fitness  $f_i$  of all individual appertaining to P is equal to the inverse of the sum of P' individual strength which dominate it plus one (2).

$$s_i = \frac{n}{\mu + 1} \quad (1)$$

$$f_i = \frac{1}{1 + \sum_{i', j} s_{i'}} \quad (2)$$

So, one individual is as less performing than it is dominated by individuals of P'. One selection method like roulette wheel is used after in order to favorite the good fitness individual. Figure 7 illustrate performances calculation.

With this technique, Pareto front individuals are valorized compared to others individuals. The firsts' one have a fitness equal at one and the seconds' inferior at one, so the Pareto front individuals have more chances than the other to be present into the next generation. This point permits to obtain a convergence to good scenarios.

For the second objective of the evaluation, valorize the population diversity, we can notice that a technique based on niches is implicitly effectuated by fitness calculation above. Each figure 7 rectangle is considered like a niche in Pareto dominance terms. If one niche contains a high number of solutions, its contribution to individual strength is strong. In return, this high strength implies a weak fitness for individuals belong to this niche. It's what we can notice in niche (1), in spite of this individuals aren't dominate only by a single other, they obtain a mediocre note. Inverse niches with few individuals will have dominants with weak strength which imply a high fitness for its. For example, niche (2) individuals obtain the best population fitness except Pareto front individuals. Thus we obtain a stabilization of the subpopulations height into niches.

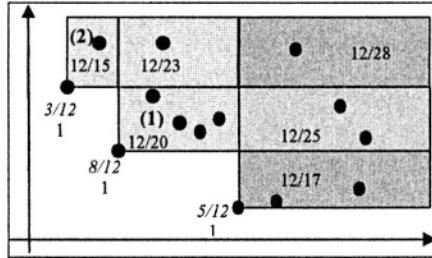


Figure 7. Fitness affectation

We can notice that actually, we use a clustering method in order to limit the number of individuals in the Pareto front.

## 4. RESULTS AND FUTURE EVOLUTIONS

### 4.1 Experimental results

Presently, to validate this tool we use a module to randomly generate problems. Tests on real problems from industry aren't still being considered but are envisaged in near future once the different tools (HILES, LORA and GESOS) linked. Results presented below have been obtained on middle size problem with 60 tasks with 15 options each. Fixed objectives were 5000 € for the project cost and 500 h for desired duration.

We use for the moment only these two criteria. Other criterion like risk associated to the project or product quality aren't yet implemented. Actually, during our experimentations, we voluntary use a high number of generations and this to verify the algorithm convergence and see the different algorithm phases.

On figure 8, we can note that average values of loss functions decrease before converge. In some cases, we note an increase of one of the criteria whereas the other decreases this translates simply the existing dependence between these two values. Of course, if we begin with a population very near of one of our objective but far to the second, the algorithm will reduce the quality on the first criterion to raise the second in order to obtain a best compromise. On figure 9, we can see the population (represented by the crosses) and Pareto front individuals (the points) repartition after research of the optimum. On this figure, we refund the classical Pareto front in minimization problem and we can notice that the population is good attracted on its borders. On figure 10, the individuals repartition is doing on a cross form centered on the fixed objectives values. In addition, we can see

that the population is well distributed in the space of research and well covers a vast field of scenarios. It should be noted that, in this application, we could find a solution very close to the optimal with one duration to 500,7 h and a cost to 5003 €.

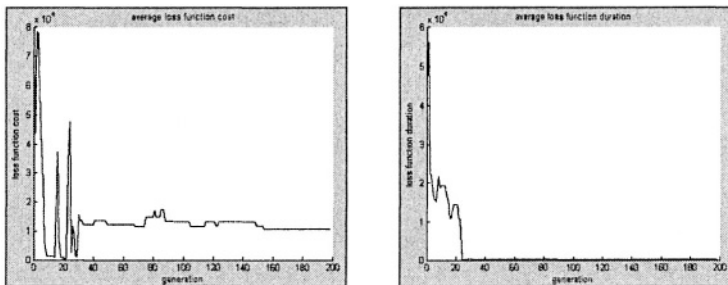


Figure 8. Average value of cost and duration loss function

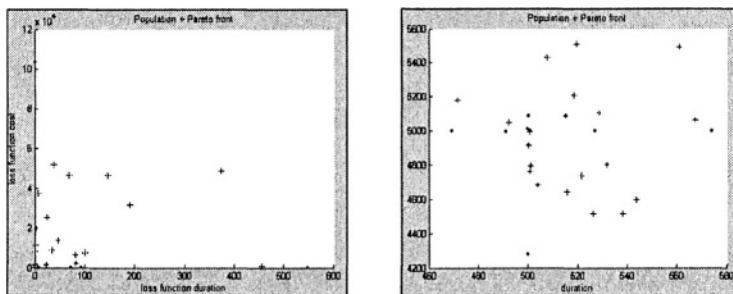


Figure 9. Repartition function of loss functions Figure 10. Repartition function parameters

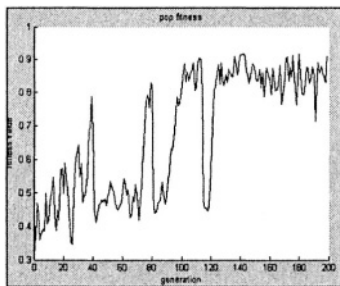


Figure 11. Average fitness value

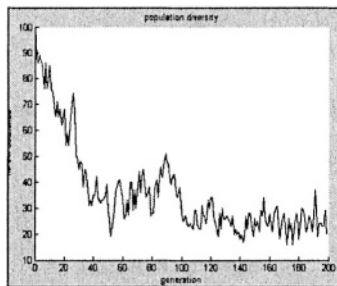


Figure 12. Population diversity

In this example an optimal solution was obtained in less of 40 generations (figure 8). After, the fitness (figure 11) and its diversity

(figure 12) variations are not related to the research of Pareto front individuals but to the second phase of the algorithm during which, once time the Pareto front established, population will tender to be distributed uniformly along the Pareto front. In this phase, it's no more the population which will push the performances of the Pareto front but the front which will attract the individuals of the population.

Results obtained in experiments are thus encouraging, since they show a rapid convergence of the face of Pareto towards the optimal solutions. Another advantage of this algorithm is its insensitivity to the distance between project objectives and limits. We can notice that even on problems with objectives very distant from those attainable, acceptable results are obtained from the viewpoint of the solution's quality. It is a considerable advantage; numerous other heuristic are very sensitive to this phenomenon and give in this case poor quality solutions.

## **4.2 GESOS prospects and evolutions**

The use of evolutionary algorithms in this type of application is quite new because until now they were used in problems of scheduling where tasks are known but not the order in which they must be followed. Here, the problem is the opposite, the general order of the tasks is predetermined and we try to find which chain of tasks would lead us as close as possible to our objectives. It is thus a new problem that is posed. Thanks to the tool, we obtained encouraging results that validate our approach on examples elaborated for the purpose. In particular, we have shown that, when the genetic parameters are well-adjusted, convergence is obtained within few seconds, for quite complex problems (100 tasks with 100 options each [Bar04c]). Of course, the tool deserves to be improved, on several aspects which can be either relative to the management and design processes by taking into account more tasks parameters or relative to the genetic procedure (e.g. a more sophisticated stop criterion to elaborate, studying the impact of another chromosome coding, test multiple cut points and variants on mutation and crossover, realize a precise adjustment of algorithm parameters like mutation and crossover rates...).

## **5. CONCLUSION**

Nowadays, project management is basically funded onto tasks scheduling and resources management considerations. It supervises product design tasks in the way that the decisions made determine the allocation of resources.

This situation is not totally satisfactory because it induces misunderstandings. In previous papers [Bar04a] [Bar04b], we have proposed the use of some models and tools. In this one, we focused on the selection and optimization methods and tools that can be envisaged. The paper showed that genetic techniques can be employed to reduce computation delays in order to help the project manager to quickly define project scenarios. This corresponds to a real industrial need. We exposed our strategy, justified its choice and presented the benefits that can be obtained on the case of a complex example, though our approach still merits improvements. In conclusion, GESOS seems to be a well-adapted tool for decision makers to manage their projects. Considering the whole project in which we are involved, we determined ambitious objectives, and we are now in the phase of integrating the several research tools developed by our partners. We are now contemplating the possibility to use this platform and test it on real complex industrial applications.

## REFERENCES

- [Bar04a] Baron C., Esteve D., “Towards a Shared Process for Product Design and Project”, **14<sup>th</sup>** Annual International International Conference on System Engineering (INCOSE), Toulouse, France, June 2004.
- [Bar04b] Baron C., Esteve D., Rochet S. “How evolutionary computation can be introduced to select and optimize scenarii along a product design process”, Transactions on Systems, N. Mastorakis Editor, World Scientific and Engineering Academy and Society Publishers, ISSN 1109-2777, pp. 888-893, issue 2, vol.3, April 2004.
- [Bar04c] Baron C., Rochet S., Esteve D. “A genetic approach to support decision makers during project management”, invited communication at 4th Int. Conf. on Soft Computing, Optimization, Simulation & Manufacturing Systems, Miami, Florida (USA), April 2004.
- [Bri04] Briand C., Doucet J.-E., Esquirol P., Huget M.-J., Lopez P., “Projet de plate-forme logicielle LORA – specification 1.1 Représentation de problèmes d’ordonnancement de tâches et d’affectation de ressources”, Rapport LAAS N°01551, December 2001, 24p, February 04.
- [Gol94] Goldberg D., “Algorithmes génétiques”, Addison-Wesley, 1994.
- [Ham03] Hamon J.C., Esteve D, Pampagnin P., “HiLeS Designer: A tool for systems design”. International Symposium Convergence 03: Aeronautics, Automotive & Space, Paris, December 2003.
- [War89] Warner J.C., O’Connor J., “Molding Process is Improved by Using the Taguchi Method”, Modern Plastics: 65-68, 1989.
- [Zit99] Zitzler E., Thiele L., “Multiobjective Evolutionary Algorithms : A comparative Case Study and the Strength Pareto Approach”. IEEE Trans. On Evolutionary Computation, tome 3, n°4, 257-271, 1999.
- [Zit98] Zitzler E., Thiele L., “An Evolutionary Algorithm for Multiobjective Optimization : The Strength Pareto Approach”, TIK report, N°43, May 1998.