# MULTIMODAL OPTIMIZATION OF SIMULATED SYSTEMS

**Ahlem Baccouche[1,2], Anne-lise Huyet[2], Henri Pierreval[2], Besma Feyech[1]**

[1]University of Tunis, Research Unit of Technologies Information and Communication, School of Sciences and Technologies of Tunis, Bab menara 1008 Tunis, Tunisia

[2]Clermont University, IFMA, LIMOS, UMR CNRS 6158, Campus de Clermont Ferrand, Les Cézeaux, F-63175 Aubière, Cedex, France

*henri.pierreval@ifma.fr*

**Abstract**

Many approaches have been suggested to solve simulation optimization problems. In classical problems, these approaches provide only one solution (optimal or "near optimal"), which is the one that gave the best results on a given performance criterion, on the basis of the simulation experiments performed. In such types of problems as design problems, this "best solution" may not be considered as the most suited for design makers. Indeed, other considerations as the measured performance can have to be taken into account (e. g., certain solutions can be more difficult to implement, or may induce supplementary costs); the simulation model cannot incorporate all the elements that are important to the decision maker. To provide decision makers with sufficient flexibility in their final choice, we suggest to provide several efficient solutions (the best found and some other local optima with acceptable performance) instead of only one. Then, the decision makers can make their choice based on other considerations than only the performance evaluated by simulation. To address this problem, we propose to combine a multi-modal evolutionary algorithm with a simulation model. We show how such an approach can be implemented by adapting the recently published Crowding Clustering Genetic Algorithm (CCGA) and connecting it with ARENA. The benefits of this multimodal simulation optimization approach are illustrated with a supply chain problem where several parameters have to be optimized.

**Keywords: Multimodal optimization, simulation optimization, supply chain.**

**Presenting Author's biography**

Ahlem Baccouche is preparing her Ph.D. in computer science within a collaboration between LIMOS-UMR 6158 (Laboratory of Computing, System Modeling and Optimization) in France and UTIC 02/UR/14-03 (Research unit of technologies of information and communication) in Tunisia. She is interested in simulation optimization of such systems as manufacturing systems under uncertainty.

# 1 Introduction

In the design of several types of systems, it is often necessary to determine the best values of certain parameters, so as to obtain the best performance. For example, the number of cranes in a harbor, the number of servers in a communication network, the buffer sizes and transport lot sizes in a manufacturing system, etc. In this respect, simulation is often used to evaluate the performance [1, 2], which can be based on customer's satisfaction, costs, delay, number of works in progress, etc. When there are numerous parameters whose values are to be determined, the number of possible solutions to evaluate can be too high. Thus, an exhaustive search is impossible. It is necessary to use simulation optimization methods, which incorporate a search strategy to find the optimal solution (e.g. the best performance). Many approaches suggested in the literature can be used for simulation optimization [3, 4, 5, 6, 7] to obtain the solution that gives the best results, on the basis of the evaluated performance.

However in various types of concrete problems (e. g., design problems), all the considerations that have to be taken into account cannot be included in a simulation model or in an objective function. For example, certain solutions can be more difficult than others to implement, certain costs can turn out to be too difficult to evaluate, etc. Therefore, the final choice for a solution can be based on other criteria than only the performance, as measured by the simulation model. Furthermore, the simulation model cannot incorporate every detail about the system and certain simplifying assumptions have sometimes to be made.

As a consequence, in certain cases it may be interesting to provide the decision makers with more than one solution (the one found to be the best through a simulation optimization process), which is the approach generally encountered in the literature. Decision makers can be interested in selected a solution with slightly lowest performance as far as this solution provides other advantages, which cannot be measured using simulation (e. g. difficulties in implementing the solution). This is the key idea that we develop in this article, where we suggest searching for a set of *k* efficient solutions. These *k* efficient solutions represent local optima, which we would like to be as close to the (unknown) optimal solution as possible and to be as different as possible each other. Such a problem can be addressed using a multimodal optimization metaheuristic, connected to the simulation model, as in other simulation optimization methods. To our knowledge, this approach has not been studied in the literature and is not reported in surveys, such as [3, 4, 5, 6, 7].

The aim is here to allow a choice to be made among the *k* efficient solutions, which is not only based on the performance evaluated by simulation, but also on other considerations that the decision maker finds relevant, on the basis of his or her expertise.

This paper is organized as follows. Section 2 introduces classical simulation optimization approaches. Section 3 presents multimodal optimization and introduces the Crowding Clustering Genetic Algorithm. Section 4 presents the suggested approach and Section 5 illustrates its benefits on a supply chain problem. Finally, Section 6 provides our concluding remarks and future research directions.

# 2 Simulation optimization

Simulation optimization approaches are used to search the best possible values of a vector of input variables to a simulation model (e. g., number of resources or number of operators), so as to optimize an objective function (e. g., expectation) of an output variable. Tekin and Sabuncuoglu [7] formalize a simulation optimization problem as follows:

$$\text{Min/Max } F(\mathrm{X}) \qquad (1)$$
$$\mathrm{X} \in \mathrm{D}$$

where $F(\mathrm{X})$ is generally of the form: $E\,[G\,(\mathrm{X}, \omega)]$, the performance measure of the problem. The quantity $G\,(\mathrm{X}, \omega)$ is called the sample performance, $\omega$ represents the stochastic effects in the system and $X = (X_1, \ldots, X_n)$ is a vector of input variables that belongs to domain $D = \otimes D_i$, where $D_i$ represents the respective domain of $X_i$.

In the literature, a variety of methods have been proposed to solve the above simulation optimization problem, for instance metaheuristics such as simulated annealing and evolutionary algorithms [8]. Several surveys discuss foundations, theoretical developments and applications of these techniques [6, 7, 8, 9]. However they focused on the problem given in Eq. (1), trying to find best solution. They do not discuss how several different enough solutions can be found, which is what we are interested in. In this respect, we suggest in the following to use a multimodal approach as the search method in the simulation optimization process. The basic background about multimodal optimization is given in the next section.

# 3 Multimodal optimization

## 3.1 Principle

Multimodal optimization aims at locating multiple local optima in a search space (Fig. 1) in a single optimization run. It is used in a large variety of engineering problems and is more and more popular in different areas, such as electronic engineering, telecommunication and medicine [10, 11, 12, 13].

Several metaheuristics can be used for multimodal optimization (e. g., [14, 15]). The efficiency of evolutionary approaches for this purpose has been reported by many researchers [16]. As a matter of fact, the use of a population of solutions allows niches [15, 16] to be determined, which divide the population into different subpopulations and drive them towards different local optima.

Niching methods aim at maintaining enough diversity in the population and at reducing the effect of genetic drift (i.e., convergence to single optima) resulting in the standard optimization algorithms. Various niching approaches are used in the literature [17, 18, 19]: crowding, clearing procedures, clustering or sharing schemes, etc. Among existing approaches, the recently published CCGA multimodal genetic algorithm has been found to give good results, comparing to other niching methods on standard multimodal tests functions [20].

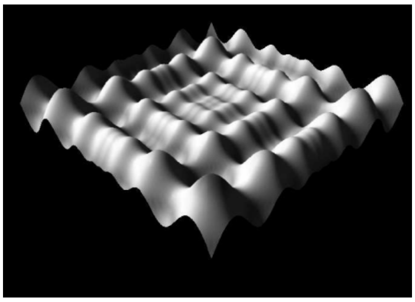

Fig. 1 Multimodal function

Several approaches have been published in the literature to addressed multimodal optimization [17]. In the next section, we focus on CCGA, which is a well known evolutionary algorithm adopted for this purpose.

### 3.2 Crowding Clustering Genetic Algorithm

CCGA provides a mechanism for generating multiple optima by combining a standard crowding method to form niches [18] with a clustering strategy to eliminate genetic drift effectively. The idea of clustering is already in crowding algorithm. The niching strategy consists en two parts. Firstly, the fittest individual of each niche is selected to survive into the next generation. Secondly competitions are made between these individuals to prevent converging to a single optimum. The main steps of the algorithm are as follows [20]:

Step 1. *Initialization* of distributed population. Number of population is *PopNum*.

Step 2. *Recombination* of parents to generate *PopNum children*.

Step 3. *Standard Crowding* and *clustering* to form niches:

For each parent $P_j$, j=1... *PopNum*,

construct *PopNum* clusters $\{P_j, CS_j\}$ using a standard crowding model with $CF = PopNum$ under a distance metric $D$ (),

For each cluster,

select the fittest individual $CC_j$ as the center.

Sort all clusters according to their centers.

Step 4. *Definition of Reserved Clusters*: *RC*. Each element in *RC* has a center of cluster $RCC_i$ and a radius of cluster $RCR_i$.

For ( j=1…*PopNum*)

Compare the j^th cluster with all clusters $RCC_i$ in *RC*,

For all i,
if ($D(CC_j, RCC_i) > RCR_i$ or Peak $(CC_j, RCC_i) = 1$) Then
place $CC_j$ into *RC*,
set the radius of cluster as min ($CR_j$, $D(CC_j, RCC_i)$).

Step 5. *Definition of the next Generation*: Define the number of elements in *RC* as *NRC*; generate (*PopNum* - *NRC*) uniformly distributed individuals in the feasible solution space. These individuals and the centers of clusters in *RC* enter the next generation.

Step 6. *Repeat* step 2 to step 6 until the maximum generation number (*MaxGen*) reaches.

The initial population can be randomly generated and *PopNum* denotes the population size. Then, each couple of parents is randomly selected from the individuals for crossover process. Two offspring's are generated per couple and we obtain the population of children. The size of this last population is called *PopNum children.* Then, an iterative procedure is applied. It begins by grouping each child $C_i$, $i$=1, 2….*CF* with its nearest parent under $D$ () using a *standard crowding method* with *CF* (Crowding factor) = *PopNum* to eliminate selection error. The individuals within each cluster are sorted according to their fitness values and the individual with the highest fitness value is called the center $CC_j$. The competition between centers is made using the distance metric $D$ (), which avoids genetic drift in the crowding algorithm, and the concept of *peak detection* which links crowding and clustering to prevent niches from being destroyed by mistake. The algorithm terminates when a given number generations is reached.

It does not use any mutation operator and diversity handled by introducing a proportion of new individuals in each generation (see step 5).

We use this algorithm in the rest of the paper. We chose it because of the good performances reported in terms of quantity, quality and precision of the solutions found [20].

## 4 Proposed approach

As explained in the introduction, we are interested in obtaining several good solutions in our simulation optimization process, so as to provide decision makers with more flexibility in their final choice. To be good enough, these solutions must obtain performances that are as close as possible to the optimal solution (local optima) and that have characteristics different enough in order to offer alternative choices. The decision making process is based on three stages, as depicted in Fig. 2.

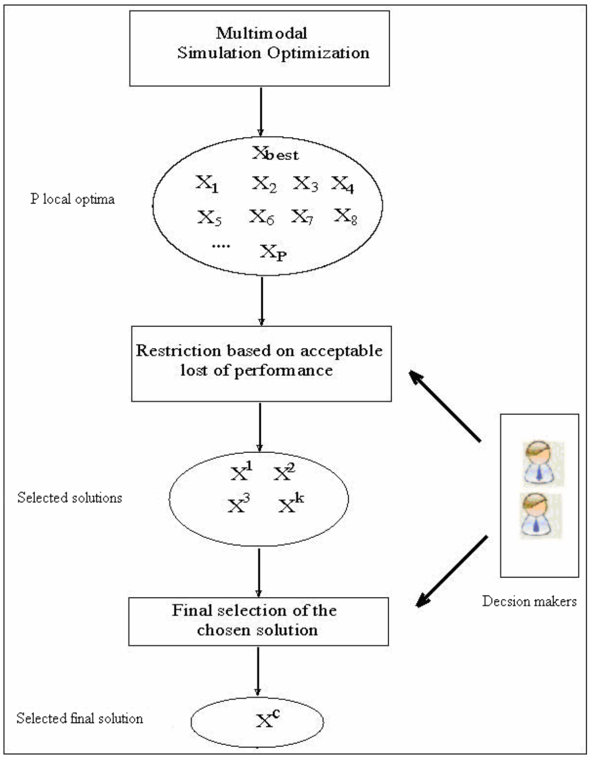


Fig. 2 Decision process

In the first stage, we aim at determining the set $S$ of $p$ solutions, different enough, which are local optima of Eq. (1). Such a problem can be solved using a multimodal optimization search connected with the simulation model that evaluates the system response (see Fig. 3). The multimodal simulation optimization search allows us to find $p$ solutions: $X_1,\ldots, X_p$ which are local optima.

In the second stage, we are interested in restricting this set to the $k$ solutions that have acceptable performances. Assume that the best solution is $X^{Best}$ and $F(X^{Best})$ the relative performance. Then, we want that $F(X^1)>F(X^{Best})$-$MAXDEG$, …., $F(X^k)>F(X^{Best})$-$MAXDEG$, where $MAXDEG$ represents the maximum degradation of performance that the decision maker is willing to accept.

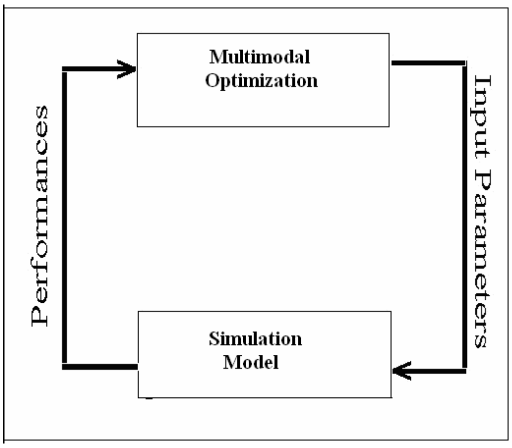


Fig. 3 Multimodal simulation optimization

The final choice is made in the last stage, where the alternatives are analyzed. The solution $X^c$ among the $k$ selected is chosen on the basis of the decision maker's expertise. This choice is not only based on the performance. Other considerations, as the solution cost, the easiness of implementation (e. g. technical difficulties), and the final users acceptance can play an important role in the final decision. Such types of considerations can be quite difficult to include in a simulation model. More formal methods, to make the final selection, such as *Analytic Hierarchy Process* (AHP) [21] can be used, if needed, at this stage.

This multimodal simulation optimization approach has been implemented using CCGA (in *C++* language), as the search method connected with ARENA [22]. CCGA has needed to be adapted to take into account other variables as real variables (CCGA has been designed for optimization in $IR^N$).

## 5 Application to a supply chain problem

This application problem is inspired from industry and is described in [23]. The supply chain is composed of 2 sites. The supplier manufactures the products, which are then transported to a distribution center by a lorry. The customers bring the products in the distribution center when they are available. If the products are not available, then orders are sent to the suppliers. Three types of parts are produced: $P_m$ for $m$= 1 to 3. The demand of products is based on the following mix: 65% $P_1$, 25% $P_2$ and 10% $P_3$.

The supplier manufacturing system consists of a production line, which carries out three operations, according to a Just In Time philosophy. Every operation requires a constant amount of processing time. Production is performed by batch (a given batch size for each type of product). The stock for raw material is assumed to be infinite. Kanban loops are used between each machine of the production line (for each part type). A lorry delivers batches of products (transportation lots of a given size for a given product) to the distribution center; it also brings back the orders for new products. Lorries visit the suppliers every 4 hours.

The optimization problem consists in determining the following three parameters for each part type:

- $KB_m$ = [10…30]: number of kanbans (for the production line) for the parts of type *m*,
- $TLU_m$ = [1…20]: transport load units associated with part (transport batch size) *m*,
- $PLU_m$ = [1…10]: production load units associated with part (production batch size) *m*.

These parameters define the input variables and the vector $X = [KB_m, TLU_m, PLU_m]$ for $m = 1$ to 3. This vector defines a configuration of our system.

For economical reasons, we have to minimize the costs induced by order-to-delivery lead time and the amount of WIP. The objective function to minimize can be expressed in Eq. (2):

$$F(X) = C_{WIP} * NBE + \sum C_m * TpsC_m \qquad (2)$$

*F(X)* is stochastic (the time between arrival of customers orders follows an exponential distribution).

With:

- *NBE*: number of WIP,
- $C_{WIP}$ : cost of one WIP,
- $C_m$: cost of one unity of time for orders delay of part *m*,
- $TpsC_m$: time to satisfy a command of part *m*,
- *m*: type of part..

The model being stochastic, a long simulation run [1] of 80000 hours has been used, with a warm-up period of 4000 hours.
The Euclidian distance *D()* is used in CCGA as the distance metric to form niches and in the selection of reserved clusters (respectively steps 2 and step 5 see section 3.2). After several empirical tests, the population size chosen was 50 and the stopping criterion 500 generations.

We obtained 39 solutions, which are the centers of the reserved clusters (see section 3.2). The best solution $X^{Best}$ has a performance of 8314.11. It is considered that the performance lost should not be more than 9145.52, which represents 10 % of degradation. The selected solutions are presented in Tab. 1.

If decision makers want to take into account the $CO_2$ production of this supply chain, then larger transportation batches can be preferred (within the limit of capacity of the lorries available). In this respect, the solution 5 might be considered as more interesting than the solution 1. This choice reduces the number of transport cycles. Nevertheless, this solution implies large production load units. This aspect could have negative influence in case of quality problems. Indeed, when a default is detected on a product, generally the concerned batch is not send to the customer to prevent quality feedbacks. Taking this into account would lead to favor solutions 1 and 9 as the most relevant ones, because of their smaller production batches. We could also consider that configurations with few kanbans would provide more flexibility to the production process, which is useful for example in case of urgent orders or unexpected breakdowns.

Tab. 1 Selected solutions

| N° Solution | Performance | $KB_1$ | $KB_2$ | $KB_3$ | $TLU_1$ | $TLU_2$ | $TLU_3$ | $PLU_1$ | $PLU_2$ | $PLU_3$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 8314.11 | 20 | 13 | 12 | 18 | 12 | 6 | 1 | 2 | 7 |
| 2 | 8537.2 | 20 | 13 | 10 | 20 | 6 | 7 | 10 | 6 | 4 |
| 3 | 8752.36 | 22 | 19 | 10 | 18 | 18 | 3 | 5 | 6 | 5 |
| 4 | 8880 | 20 | 16 | 23 | 19 | 9 | 9 | 1 | 8 | 10 |
| 5 | 8932.74 | 18 | 18 | 14 | 18 | 14 | 14 | 6 | 5 | 7 |
| 6 | 8961.64 | 15 | 12 | 11 | 11 | 6 | 7 | 10 | 6 | 4 |
| 7 | 8982.33 | 22 | 11 | 17 | 17 | 7 | 10 | 6 | 4 | 7 |
| 8 | 8984.73 | 27 | 27 | 12 | 19 | 19 | 5 | 3 | 9 | 7 |
| 9 | 9040.66 | 24 | 16 | 17 | 16 | 16 | 9 | 7 | 2 | 2 |

Such comparisons illustrate how, in some situations, it could be acceptable to lost performance to take into consideration other important issues, such as the environmental impact or the flexibility level.

# 6 Conclusion

In a certain number of optimization problems (e. g., in industry), loosing performance can be accepted as far as other considerations, such as the cost of the proposed solution, its easiness to be implemented in the enterprise, can be considered. Therefore, a simulation optimization approach providing several efficient solutions instead of one may turn out to be very useful. This objective can be achieved through a multimodal optimization approach of simulation models, which provides the decision maker with degrees of freedom in his/her choice of efficient solutions. We have illustrated this paradigm of “multimodal simulation optimization” by adapting the CGGA algorithm to problems with integer variables and connecting it with ARENA. The reserved centres of clusters obtained at the last iteration of CGGA provide a set of local optima, which constitutes alternative solutions from which the most relevant can be determined by the decision makers on the basis of their several types of other considerations (costs, implementation, robustness, etc.) and on their respective expertise. A supply chain problem, where lot sizes have to be determined has shown the relevance of this approach.

Our future researches directions are concerned with distributing this approach to save computing time [24], incorporating statistical comparisons of solutions (when the simulation model is stochastic), taking into consideration simulation configuration problems [25] and testing other multimodal algorithms.

# 7 References


[1] A.-M. Law, W.-D. Kelton,. *Simulation Modeling & Analysis*. McGraw-Hill, 2000.

[2] J-P-C.Kleijnen. *Design and Analysis of Simulation Experiments*. Heidelberg: Springer-Verlag, 2008.

[3] F. Azadivar. A tutorial on simulation optimization. Proceedings of the 1992 Winter Simulation Conference. 1992.

[4] S. Andradottir. Simulation optimization. In Handbook of simulation. Jerry Banks (Ed.), John Wiley & sons inc. New York . 1998.

[5] M. Fu. Optimization for simulation: theory vs. practice. *INFORMS Journal Computing*, 14: 192-215, 2002.

[6] R. J. Swisher, P-D. Hyden, S-H. Jacobson and L-W. Schruben,. A survey of recent advances in discrete input parameter discrete-event simulation optimization. *IIE Transactions*, 36(6): 591 – 600, 2004.

[7] E. Tekin, and I. Sabuncuoglu. Simulation Optimization: A. Comprehensive Review on Theory and applications. *IIE Transactions*, 36: 1067-1081, 2004.

[8] H. Pierreval and L. Tautou. Using evolutionary algorithms and simulation for the optimization of manufacturing systems. *IIE Transactions, Special issue on Operations Engineering on Simulation*, V 29 (3): 181-189, 1997.

[9] Y.F Li, S. H Ng, M. Xie and T. N Goh. A systematic comparison of metamodeling techniques for simulation optimization in decision support systems. *Applied. Soft. Computing. Journal*, 17 pages, 2010

[10]M. Gatt, A-D. Anderson, B-S. Reddy, P. Hayward-Sampson, I-C. Tring and J. MacFie. Randomized clinical trial of multimodal optimization of surgical care in patients undergoing major colonic resection. *British Journal of Surgery*, 92 (11): 1354-1362. 2005.

[11]C. Y. Lin and F. H. Wang. Sequential simulated annealing for multimodal design optimization. *Journal of the Chinese Institute of Engineers*, 26(1): 57-70. 2003.

[12]O. Hajji, S. Brissets and P. Brochet. A new tabu search method for optimization with continuous parameters. *IEEE transactions on magnetics*, 40 (2): 1184-1187. 2004.

[13] http://en.wikipedia.org/wiki/ModeFRONTIER.

[14] D.Beasley, D-R. Bull and R-R. Martin. A sequential niche technique for multimodal function optimization. *Evolutionary computation*, 1(2): 101-125. 1993.

[15]A. Petrowski. A clearing procedure as a niching method for genetic algorithms. Proceedings of the IEEE International Conference on Evolutionary Computation. 1996.

[16]G. Singh and K. Deb. Comparison of multi-modal optimization algorithms based on evolutionary algorithms. ACM, GECCO. 2006.

[17]B. Sareni. Méthodes d'optimisation multimodales associées à la modélisation numérique en électromagnétisme. PhD. Thesis 1999, Ecole Centrale de Lyon, France.

[18]K-A. De Jong. An Analysis of the behavior of a class of genetic adaptative systems. PhD thesis 1975, University of Michigan, Dissertation Abstracts International, 36, (10), 5140B, university Microfilms n° 76-9381.

[19]D-E. Goldberg and J. Richardson. Genetic algorithms with sharing for multimodal function optimization. Proceedings of the 2nd International Conference on Genetic Algorithms. 1987.

[20]Qing Ling, W. Gang, Y. Zaiyue and W. Qiuping. Crowding clustering genetic algorithm for multimodal function optimization. *Applied Soft Computing*, 8: 88-95. 2008.

[21]E. Triantaphyllou. *Multi-Criteria Decision Making Methods: A comparative Study* (Applied Optimization, Volume 44). Springer 1 edition, 2000.

[22] http://www.arenasimulation.com.

[23]A-L. Huyet, and J-L. Paris. Setting and Analysis of Multiproduct Supplier Links managed by kanban using a Learnig+ Evolutionary Optimization. Proceedings Of ISC 2003, the Industrial Simulation Conference. 2003.

[24]H. Pierreval and J. L., Paris. Distributed evolutionary algorithms for simulation optimization. *IEEE transactions on man, systems, and cybernetics* , 30(1): 15-24. 2000.

[25]H. Pierreval and J. L., Paris. From 'Simulation Optimization' to 'Simulation Configuration' of Systems. *Simulation Modelling Practice and Theory*, 11(1): 5-19. 2003.