Interactive optimization strategies for layout problems

Julien Bénabès, Fouad Bennis, Emilie Poirson, Yannick Ravaut

International Journal on Interactive Design and Manufacturing manuscript No.
(will be inserted by the editor)

Interactive optimization strategies for layout problems

Julien Bénabès(1) · **Fouad Bennis**(1) · **Emilie Poirson**(1) ·
Yannick Ravaut(2)

June 23^{rd}, 2010

Abstract Component and facility layout plays an important role in the design and usability of many engineering products and systems as mechanical design, process plan, management and architecture including ship compartment layout,... Because of the great complexity of most industrial layout problems, the decision of the acceptable layout is a hard and critical task since the special layout can have a significant consequence on the user satisfaction, the economic cost and broadly speaking the global performances. Thus, in order to propose to the designer an optimal spatial arrangement in a reasonable time, this paper develops an interactive optimization strategy based on a genetic algorithm coupled with a separation algorithm. The proposed method is tested on the layout problem of a shelter. The resolution of this problem is innovative because it introduces the concept of space of accessibility in the layout problem formulation.

(1) Centrale de Nantes, Irccyn, UMR CNRS 6597, Nantes, France
Tel.: (+33)2.40.37.69.42
Fax: (+33)2.40.37.69.30
E-mail: julien.benabes@irccyn.ec-nantes.fr

(2) Thales Communications, Cholet, France
Tel.: (+33)2.41.64.56.60
E-mail: Yannick.ravaut@fr.thalesgroup.com

Keywords Layout problem · Interactive optimization · Genetic algorithm

1 Introduction

Layout problem is inherently a multidisciplinary task (Giassi et al, 2004). It covers all the aspects of the product design life cycle from the conceptual to the detailed stage and makes necessary the collaboration between experts of technical and economical disciplines. In fact, layout design is usually formulated as an optimization problem: *find the best arrangement (location and orientation) of components in a given available space satisfying geometrical and functional constraints.* A non-overlapping constraint is basically a common geometrical constraint for all three-dimensional layout problems, while alignment, orientation or gathering components refer to functional constraints. Because of the geometrical complexity, the three-dimensional layout optimization problems are generally considered as non-linear and NP-hard problems. It means that the problem is intrinsically harder than those which can be solved by a non-deterministic Turing machine in polynomial time. The objective and constraints evaluation is generally time consuming.

It is essential to distinguish between Cutting and Packing (C&P) problems and layout problems. In C&P problems, components are only geometrically related to each other, whereas in layout problems, components are geometrically and functionally connected. This difference leads different tools and methods to solve each class of problem being aware of the common non-overlap constraints in the two problems.

Typologies of C&P problems have been proposed (Dyckhoff, 1990), but as far as we know, there is no

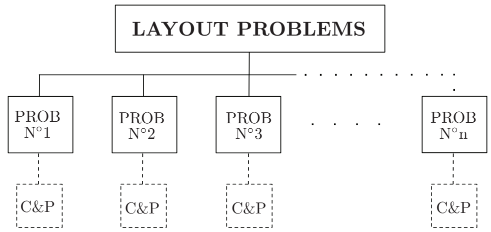


Fig. 1 Typology of layout problems

general typology of layout problems. Drira et al. have described a tree representation of facility layout problems (Drira et al, 2007), that depends on design constraints and objective functions of the location of facilities inside a plant.

Actually, as shown in figure 1, layout problems can be divided into several kinds of specific problems, which have their own solving method. C&P problems can be assimilated as a particular application of each specific problem.

Layout problems can also be classified according to three criteria: the compactness of the problem, the number and type of design constraints and objectives and the geometrical complexity of the design components. Let us consider four examples of layout problems: the container loading problem (1), the engine layout design (2), the room layout design (3) and the manufacturing facility layout design (4). Figure 2 illustrates the classification of these four problems according to the three criteria. For example, the engine layout design is a problem with an important compactness. Constraints and objectives are multiple (non-overlap and functional constraints, accessibility objective,...) and the different parts of the engine have complex lines.

The formulation of layout problems uses single or multi-objective optimization. The designer can make an early decision by using an aggregation function in order to transform a multi-objective optimization problem into a single-objective one. This approach is only effective when all data and information on the aggregation are available or if the designer is familiar with the specific layout problem. In this paper, multi-objective optimization is used. The decision on the preferences between objective functions is delayed so that the designer can use the Pareto-front in order to select the most appropriate solution. In this approach, the designer has to simultaneously optimize two or more conflicting objectives subject to constraints.

The general formulation of a multi-objective optimization problem can be formulated by:

$$\begin{cases} \text{find the design variable } \mathbf{x}^* = (x_1, x_2, ..., x_n) \\ \mathbf{x}^* = \operatorname{argmin} F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x})) \\ \text{s.t: } g(\mathbf{x}) \leq 0 \text{ and } h(\mathbf{x}) = 0 \end{cases} \tag{1}$$

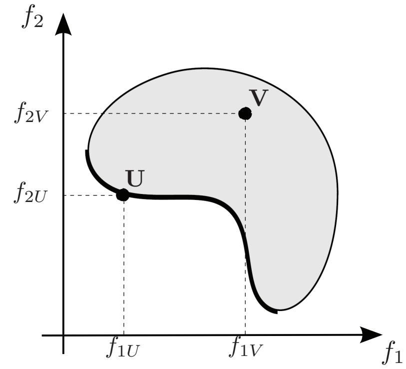


Fig. 3 Pareto front of a multi-objective problem

where m is the number of objective functions and n the number of design variables.

The designer has to compare two solutions represented by two vectors of objectives $F_U = (f_{1U}, f_{2U}, ..., f_{mU})$ and $F_V = (f_{1V}, f_{2V}, ..., f_{mV})$ where f_{iU} is the ith component of the vector of objectives F for the design variable U. In fact, U dominates V (Pareto dominance) if U is as good as V for all the objectives and U is better than V for at least one objective. Mathematically, this can be formulated by:

$$\begin{cases} \forall i \in [1, ..., n] \, f_{iU} \leq f_{iV} \\ \exists j \in [1, ..., n] \, f_{jU} < f_{jV} \end{cases} \tag{2}$$

Multi-objective optimization searches for the set of non-dominated points (assimilated to Pareto-optimal points in the next sections of this paper) in the objective space given by efficient solutions. Figure 3 represents the Pareto-front for an optimization problem defined by two objectives (min f_1, min f_2), where U dominates V.

One finds multiple single or multi-objective solving approaches to solve layout optimization problems in two or three dimensions. Traditional optimization approaches for three dimensional layout problems are described by Cagan et al. (Cagan et al, 2002). Some approaches use genetic algorithms (Yi et al, 2008), simulated-annealing algorithms (Szykman and Cagan, 1995, 1997) or extended pattern search algorithms (Su and Cagan, 2000). Most search algorithms are developed for a specific problem and they provide an effective optimization strategy for it. Therefore, they are not generic and can not be adapted to other layout problems. In this paper, the proposed method is a based on a generic technique for solving layout problems. The design strategy uses a genetic algorithm coupled with a separation algorithm. This approach insures a good diversity of solutions computed by the algorithm and allows the designer to interact with the Pareto-optimal solutions, in order to make a final decision.

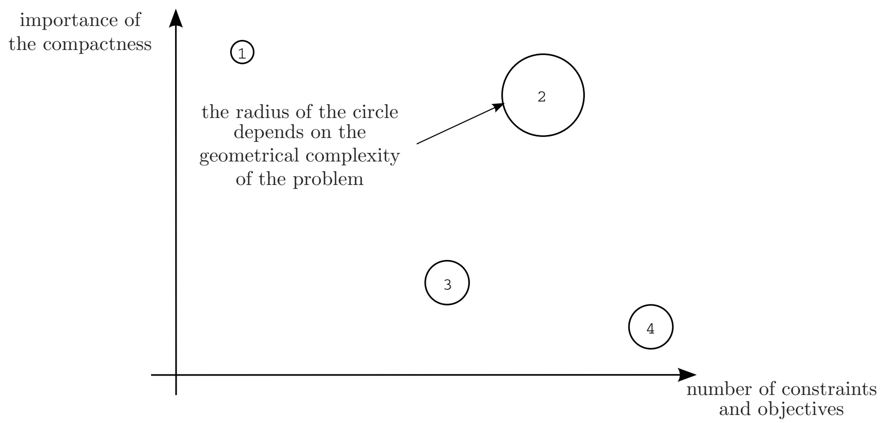


Fig. 2 Layout problems representation

This paper is organized as follows: in section 2, the synopsis of the proposed optimization method is presented. In section 3, the proposed method is tested on the layout problem of a shelter. The problem formulation and the results obtained by the method are described and analyzed. Sections 4 and 5 are dedicated to an outlook on future work and the conclusion.

One finds multiple single or multi-objective solving approaches to solve layout optimization problems in two or three dimensions. Traditional optimization approaches for three dimensional layout problems are described by Cagan et al. (Cagan et al, 2002). Some approaches use genetic algorithms (Yi et al, 2008), simulated-annealing algorithms (Szykman and Cagan, 1997) or extended pattern search algorithms (Su and Cagan, 2000). Most search algorithms are developed for a specific problem and they provide an effective optimization strategy for it. Therefore, they are not generic and can not be adapted to other layout problems. In this paper, the proposed method is based on a generic technique for solving layout problems. The design strategy uses a genetic algorithm coupled with a separation algorithm. This approach insures a good diversity of solutions computed by the algorithm and allows the designer to interact with the Pareto-optimal solutions, in order to make a final decision.

This paper is organized as follows: in section 2, the synopsis of the proposed optimization method is presented. In section 3, the proposed method is tested on the layout problem of a shelter. The problem formulation and the results obtained by the method are described and analyzed. Sections 4 and 5 are dedicated to an outlook on future work and the conclusion.

2 Optimization strategy

Layout problems are generally defined as complex optimization problems. The search of a "feasible" design, it means a design which respects all the design constraints, is a hard task. This complexity is linked to the parceling of the design space, because of the geometric complexity of layout components, the non-overlap constraints and the relative location between components. This property leads us to recognize that there is no choice but using stochastic or heuristic techniques for solving two or three-dimensional layout problems. These algorithms make it possible to explore efficiently the design space and avoid local optimum. A multi-objective genetic algorithm is used in the optimization strategy proposed in this paper

Since the genetic algorithm is based on stochastic operators and parameters, the progression of the optimization process is time consuming. It depends also on the number of design variables, the number of components and the types of design constraints. Moreover, since the genetic algorithm is randomly initialized with a population of designs with high number of overlapped components, the algorithm fails to find feasible solutions. Then, in order to improve the performances of the algorithm, two new steps are introduced into the global process of the genetic algorithm: separation techniques and interaction with the designer.

The objective of this approach is to generate an uniformly distributed global Pareto-front for layout optimization problems. Our strategy consists of initializing the multi-objective optimizer with a population of individuals which have been locally modified by a separation algorithm and by interaction with designer in order to reduce the violation of placement constraints. In fact, this strategy is based on three complementary approaches, which are clearly separated:

1. firstly, the generation of a database of mixed designs that respect non-overlap constraints,
2. secondly, the multi-objective optimization of this database by considering all the design objectives,

3. thirdly, the interactive choice of the appropriate solution made by the designer, by using the Pareto-front of designs generated by the genetic algorithm.

1. Separation algorithm

Several separation algorithms have been proposed (Imamichi and Nagamochi, 2007, 2008). However, the key idea is always the same: given a configuration that does not satisfy location constraints, the objective of the separation algorithm is to minimize the non-respect of overlap between components and protrusion (overlap between components and the non allowed space).

For solving simple layout problems in two dimensions, the separation problem is formulated as an unconstrained minimization problem defined by:

$$(Sep\,Algo)\begin{cases}\min F(X)=\sum A_{ij}\\ i,j\in[1,...,n]\,,\,i\neq j\end{cases}\tag{3}$$

where A_{ij} represents the intersection area between the components i and j. Consequently, it is possible to define the violation of placement constraints F as the total sum of intersection areas between the different elements which make up the layout design. For example, let us consider that all the components of a two-dimensional layout design are rectangles. The intersection area between the components i and j is equal to:

$$\begin{aligned}A_{ij}= \;& \max[\,0\,,\,min(\,x_i+\frac{l_i}{2}\,,\,x_j+\frac{l_j}{2}\,)\\ &-\max(\,x_i-\frac{l_i}{2}\,,\,x_j-\frac{l_j}{2}\,)]\\ &\times\max[\,0\,,\,min(\,y_i+\frac{L_i}{2}\,,\,y_j+\frac{L_j}{2}\,)\\ &-\max(\,y_i-\frac{L_i}{2}\,,\,y_j-\frac{L_j}{2}\,)]\end{aligned}\tag{4}$$

where (x_i, y_i) are the coordinates of the geometric center of the rectangle i. L_i and l_i represent respectively the length and the width of the rectangle i.

The algorithm used to minimize F is based on the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. This algorithm computes a finite-difference approximation of the gradient and the hessian of the function F in order to locally modify the optimization variables and to minimize F. The algorithm stops after a certain number of iterations.

In order to understand the principle of the separation algorithm, let us consider a layout problem test. The dimensions of the square container are $10\times 10 = 100\,m^2$. The objective of this two-dimensional layout problem is to place, in the container, N square items whose dimensions are $1\times 1 = 1\,m^2$. It means that the

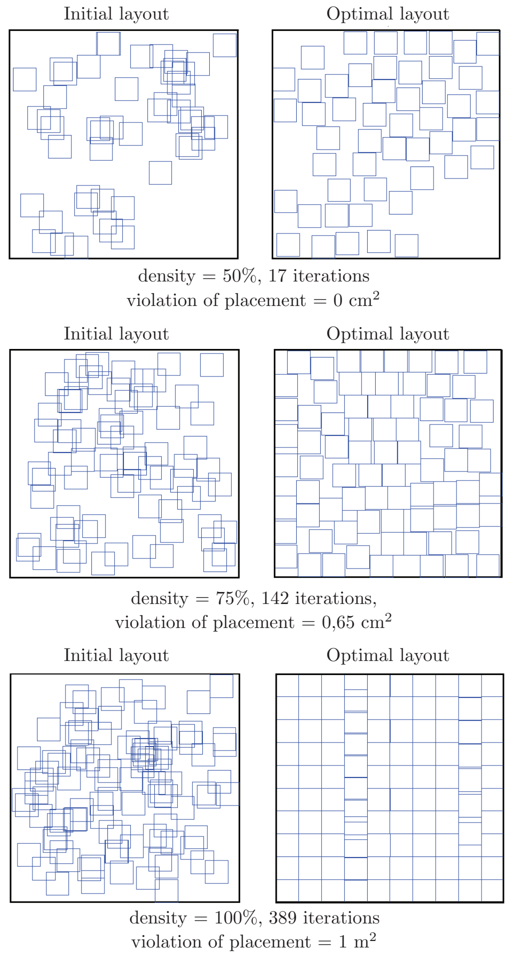


Fig. 4 Separation algorithm BFGS test

algorithm searches the optimal configuration that reduces the violation of placement constraints F, which has been previously defined as the total sum of intersection areas between the square items and the container. Figure 4 shows simulations results, considering different values of the problem density.

2. Multi-objective optimization

The multi-objective genetic algorithm used in the approach, proposed in this paper, is the MOGA-II (Poles, 2003). The MOGA-II is an efficient Multi-Objective Genetic Algorithm (MOGA) (Deb, 1998) that uses a smart multi-search elitism. This new elitism operator is able to preserve some excellent solutions without bringing premature convergence to local-optimal frontiers. For simplicity, MOGA-II requires only very few user-provided parameters. Several other parameters are internally settled in order to provide robustness and efficiency to the optimizer. The three main genetic operators of this algorithm are the directional cross-over, the selection and the mutation.

Number of individuals in the initial generation	240
Probability od Directional Cross-Over	0.1
Probability of Selection	0.05
Probability of Mutation	0.45
Probability of Classical Cross-Over	0.4
Elitism	enabled
DNA String Mutation Ratio	0.5
Number max of generations	100

Table 1 Multi-Objective Genetic Algorithm parameters

For the application studied in section 3, these genetic operators have been set as shown in the table 1. The number of individuals in the initial generation is equal to 240 because a rule of thumb suggests possibly to accumulate an initial population of at least 16 design configurations and possibly more than $2 \times Number\,of\,variable \times Number\,of\,objectives = 2 \times 24 \times 5 = 240$.

3. Interactive process
In general, the development of an engineering object is considered as a single process involving multi-criteria identification of the mathematical model followed by multi-criteria optimization of the object design on the basis of this mathematical model. The process of statement-solution of engineering design problems without the interference of the design is impossible. For solving the design problem, the designer almost always has to correct either the mathematical model, the dimension of the vectors of design variables and criteria, the design variable ranges, and so on. This creative process of correcting an initial statement is natural when solving engineering problems. The direct participation of the designer in the construction of the feasible design and non-formal analysis are the essential stage of the search for the optimal design (Serna et al, 2005; Rabeau et al, 2007). The simulation tools provide powerful solutions for planning and designing of complex mechanical system. The problem of these is the representation and the interpretation of the results by the engineer. The important for the engineer is not only the value on the point but its variation and the information about the most favorable directions. The exploitation of the results is not obvious and the link with the performance value of the real phenomenon is not trivial. When one analyzes the communication between the operator and the computer, he can perceive that the operator immersion in the digital model is very weak.

In the optimization approach, proposed in this paper, two interactive steps are successively used:

- firstly, the interactivity with the designer is used to select and locally modified solutions optimized by the separation algorithm. The objective is to consider these designs as initial individuals of the genetic algorithm. This first interactive step is limited to the geometrical non-overlap of components,
- secondly, the interactivity with the designer is used to make a final decision on optimal solutions. Our optimization strategy uses multi-objective optimization. The genetic algorithm is stopped after a fixed number of iterations. Then, since all the non-dominated designs, generated by the algorithm, are potentially good acceptable solutions of the layout problem, the designer has to explore these solutions and select the best one. However, it is well recognized by the expert of optimization that it is always very hard for the designer to make a final decision on optimal solutions. All the designer's requirements, for example the qualitative or subjective criteria, can not easily be expressed by simple mathematical expressions. Then, in order to take into account the personal judgment of the designer, this paper proposes an interactive numerical environment, used to support the decision of the designer. The designer can visualize a design and locally modify the position and the orientation of some components.

These two interactive steps are different. In the second one, the designer has the evaluation of both the geometric and functional requirements. The interactivity is not only limited to the geometric evaluation of the design, as it is made in the first interactive step. The numerical values of objective functions and there deviation are used in this second interactive step.

3 Application

In this section, our layout optimization strategy is tested on a real-world application which deals with the search of the optimal arrangement of components inside a shelter.

3.1 Problem description

Eight components, including electrical and energetic cabinets, desks and electrical boxes, have to be arranged in a shelter. The CAD model of the shelter is presented in figure 5.

The layout optimization of this shelter is a three-dimensional layout problem. However, for this application, thanks to the fact that the cabinets are full height

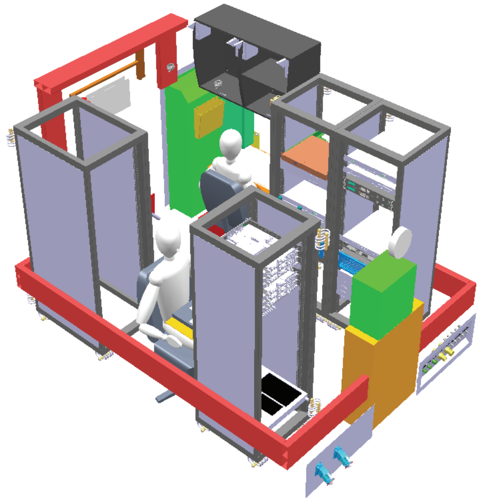


Fig. 5 Overall view of the shelter

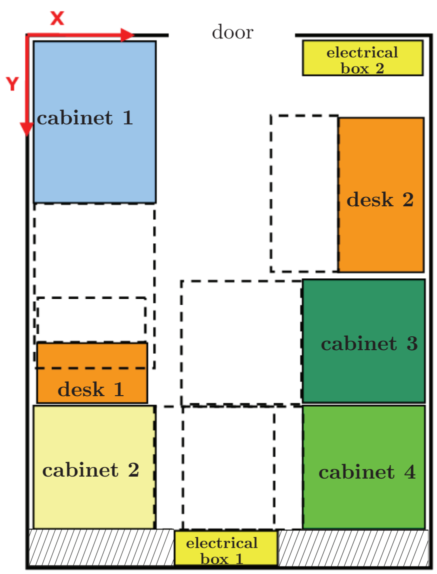


Fig. 6 Configuration model of the shelter in 2D

of the shelter and prevent a superposition of elements, the model is simplified and conceptualized in two dimensions. The simplified model of the shelter is shown in figure 6.

The formulation of this layout problem is innovative because the components can be classified in two categories: those which have a mass (*material components*) and those which no have mass (*virtual components*). Here, the virtual components represent the spaces of accessibility of the cabinets and the desks. For example, the space of accessibility of a cabinet is the required space to insert some materials into the cabinet. These spaces are symbolized in figure 6 by dotted rectangles. With this problem formulation, the design constraints depend on the category of components. It means that

Component	Mass (kg)	Dim /X (mm)	Dim /Y (mm)
shelter		2150	2740
cabinet 1	400	600	600
cabinet 2	300	600	600
cabinet 3	300	600	600
cabinet 4	300	600	600
desk 1	10	465	350
desk 2	30	525	800
electrical box 1	50	580	200
electrical box 2	35	430	250

Table 2 Data of the problem

overlap is allowed between two spaces of accessibility, considering that operations of materials loading are sequentially made, whereas overlap has to be minimized between two material components.

Moreover, the space, represented by a hatched rectangle in figure 6, is the space below the air-conditioner where no cabinet can be placed. This space is also a virtual component that is fixed during the optimization process. Besides, the free corridor, located in the middle of the shelter is a space of living that can be also considered as a fixed virtual component.

The dimensions described in the table 2 match with the configuration presented in figure 6. The density of this configuration, without considering the spaces of accessibility of the different components, is equal to 50%. If the spaces of accessibility are taken into account, this density increases up to 90%.

3.2 Problem formulation

The problem formulation is a very important step of the optimization process. The optimization problem studied here is an under constrained multi-objective optimization problem. Let us see how the variables, constraints and design objectives are defined.

Optimization variables

Each layout component has three optimization variables (X, Y, α): the coordinates of each element (a continued variable along X axis and an other one along Y axis) and the rotation angle (one discreet variable along Z axis). Consequently, the number of optimization variables for this problem is equal to 24 (= 8 items $\times$ 3 coordinates). Because of the rotation of each component, the variables X and Y are bounded, according to the following relation (for the variable X_i for example):

$$min(l_i, L_i) < X_i < l_{sh} - min(l_i, L_i) \qquad (5)$$

where l_{sh} represents the width of the shelter. Here, l_i is the dimension of the component along X axis (it does not have to be confused as the width of the component i).

Design constraints
The design constraints of this layout problem are only non-overlap constraints. They are divided in four categories, according to the following classification:

– non-overlap constraints between material components (C1),
– non-overlap constraints between material components and the spaces of accessibility (dotted rectangle represented in figure 6) (C2),
– non-protrusion constraints between components, spaces of accessibility and the shelter (C3),
– non-overlap constraints between cabinets and the space below the air-conditioner (hatched rectangle represented in figure 6) (C4).

The rectangular shape of components simplifies the formulation of the design constraints. Thus, the non-overlap constraint between the rectangles i and j is equal to the intersection area between the component i and j (in cm^2). This area has been defined in equation 4. Actually, the objective function of the separation algorithm is defined as:

$$F = C1 + C2 + C3 + C4 \tag{6}$$

Design objectives
In collaboration with the engineering experts of this specific problem, we have considered for this layout optimization problem the five following design objectives:

– to minimize the distance between the center of gravity of components and the geometrical center of the shelter, in order to balance the masses inside the shelter (O1),
– to maximize the distance between the cabinet 1 and the cabinets 2 and 3 and the electrical box 2, in order to limit interactions between energetic and electrical network (O2, O3, O4),
– to minimize the distance between the electrical box 2 and one of the shelter's walls, in order to establish a connection with exterior (O5).

The design objectives O2, O3, O4 and O5 are formulated by the distance between the centers of gravity of elements. For example, the distance d_{ij} between the components i and j is equal to:

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_j)^2} \tag{7}$$

where x and y are the coordinates of the centers of gravity of the items i and j.

Let us consider the coordinates of the center of gravity of all the elements which are placed in the shelter. These coordinates are equal to:

$$X_{gra} = \frac{\sum_{i=1}^{N}(x_i \times m_i)}{\sum_{i=1}^{N} m_i} \; , \; Y_{gra} = \frac{\sum_{i=1}^{N}(y_i \times m_i)}{\sum_{i=1}^{N} m_i} \tag{8}$$

where N is equal to the number of elements which have a mass: the cabinets, the desks, the electrical boxes and the air-conditioners. Then, by considering the equation 7, the objective 1 (O1) is computed.

More designer's knowledge could be incorporated in the layout problem formulation. It means for example, in the layout design of the shelter studied in this paper, the design objective O5 can be deleted and the degree of freedom of the electrical box 2 can be reduced, in order to force it to only move along one of the shelter's walls. The designer's contribution for the problem formulation should simplify the search of feasible solutions by reducing the number of possible designs.

3.3 Results and analysis

The resolution of this optimization problem has been firstly realized only with the multi-objective optimizer MOGA-II. The algorithm has been randomly initialized with a population of 240 designs. Most of these initial designs did not respect the non-overlap constraints because they had been randomly generated. Because of the great density of this layout problem, only one or two feasible variants were generated by the genetic algorithm. A variant is defined as: the design j is a new variant if it differs from the design i by at least one of the following criteria:

– one of the layout components has been displaced from at least $\triangle$ mm along one of the axis X or Y ($\triangle$ is set to 500 mm in this application),
– one of the components has been rotated,
– the minimum difference between the values of the objective functions of the two designs is bigger than a limit for example fixed to 10 cm.

These results lead us to use the method proposed in this paper in order to generate, with only one optimization simulation, a set of well distributed Pareto-optimal designs.

Thus, the results obtained for each step of the method are described here:

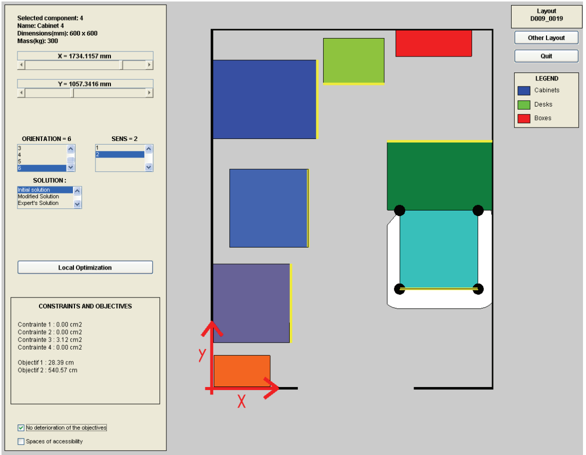


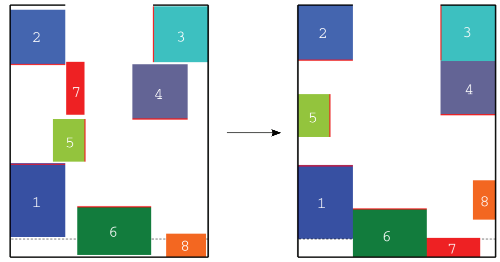
Fig. 8 Interactive environment for decision making

1. Separation algorithm and first interaction with the designer:
the algorithm has also been randomly initialized with designs that did not respect non-overlap constraints. Then, a set of feasible designs have been computed and by interacting directly with them and by relaxing the design constraints (until 150 cm^2), the designer has selected 78 different designs. Then, this population has been completed with 162 individuals, randomly generated, in order to create the first population of the genetic algorithm (240 individuals) and to guarantee the diversity of the solutions.

2. Multi-Objective Genetic Algorithm:
The algorithm has searched optimal solutions by considering the design constraints and all the objectives of the problem. Then, after a hundred of generations, a set of 14 variants have been computed. 7 of these solutions are Pareto-optimal designs. Figure 8 represents these 7 Pareto-optimal variants and the solution, initially created by the engineering expert. It is important to mention that this initial solution is an intuitive solution which has been generated only by considering geometric aspects.

3. Interactive decision making:
These 7 Pareto-optimal variants do not dominate the initial solution. On the other hand, the initial solution does not dominate either these solutions, generated by the proposed method. Actually, as it is explained in the section 2, the designer is the only person who can make a final decision on these optimal layout designs. In order to make this decision, an interactive geometric and numerical visualization of the design is used. This environment is illustrated in figure 8.

This interactive environment allows the designer to locally modify a selected design and compare the modified solution with the initial one and the solution initially proposed by the engineering experts. In figure 8, we can see that a white area is displayed around the layout component 4. The area represents the set of positions where the designer can place the layout component 4 without damaging the design objectives. It means that, considering the current orientation of the layout components, if the component 4 moves inside the white area, the new solution will not be dominated by the current one.



initial solution 3 modified solution 3

Fig. 9 Local modification of the solution 3

Design Objective	Initial solution	Improved solution 3
O1 (minimize)	25,41 cm	3,48 cm
O2 (maximize)	240,58 cm	240,58 cm
O3 (maximize)	198,50 cm	198,50 cm
O4 (maximize)	165,80 cm	172,51 cm
O5 (minimize)	0 cm	0 cm

Table 3 Industrial solution vs solution 3

Let us consider that, according to his personal judgment, the designer decides to select the solution 3 illustrated in figure 7. By locally changing the location of some components of the shelter, he improves the performances of the design. A local modification of this layout design, as shown in figure 9, improves the design objectives.

Table 3 describes all the objectives values for the initial solution and the solution 3 locally modified by the designer. Actually, thanks to the interaction of the designer with the final solutions, a new solution, better than the initial one, is created.

4 Review and outlooks

This paper has introduced a new interactive optimization strategy for solving layout problems. The method can be divided into several steps, as shown in figure 10. Firstly, a population of designs, randomly initialized, is optimized by the separation algorithm. Then, the designer interacts with the solutions computed by the separation algorithm and selects some individuals accord-

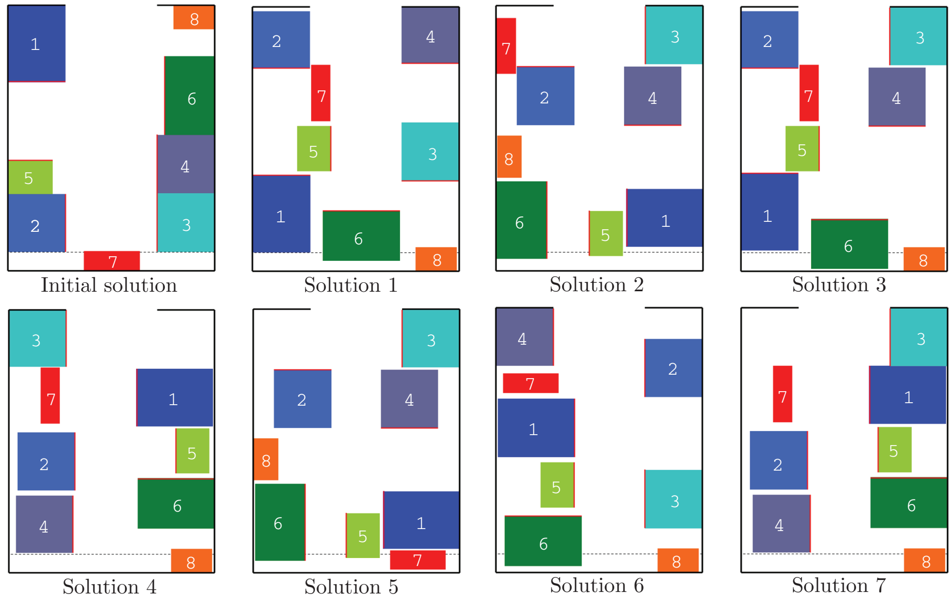


Fig. 7 Initial solution and Pareto-optimal solutions

ing to the design constraints. Secondly, the new population is optimized by the multi-objective optimizer by considering all the design objectives. Then, the designer can locally modify some computed designs in order to improve their objectives. Actually, our strategy has the innovative particularity to allow the designer to interact with the optimization process in order to improve the performances of the Pareto-optimal designs and to keep a good diversity in computed solutions.

This innovative optimization process proposed in this paper suggests that the method could be improved according to the designer preferences:

- qualitative fitness could be inserted into design process (**?**). In some layout problems, all constraints and objectives can not be easily formulated as simple mathematical expressions. It means that these constraints and objectives could be replaced by subjective criteria, defined by the designer in order to characterize the design. For example, this qualitative fitness could be represented by a mark and considered by the algorithm as a design objective. Brintrup et al. have already developed an interactive genetic algorithm based framework for handling qualitative criteria in design optimization (Brintrup et al (2007)),
- the designer could interact with design variables during the optimization process. Stopping the optimizer would allow the designer to firstly analyze a specific solution, secondly locally modify the design configuration and then decide to keep this modified design in the next generation of the genetic algorithm. We can find in (Michalek and Papalambros (2002)) a significant contribution to this concept applied to the design optimization of architectural layouts.

5 Conclusion

This article presents an innovative layout problem formulation including the concept of virtual component defined in section 3. It shows that the problem formulation is a very important step in the optimization process because it has a great impact on computed solutions. Secondly, the hybridization of the separation algorithm and the multi-objective algorithm is a very efficient method to ensure a good diversity in Pareto-optimal solutions. Moreover, the strategy is designed to allow the interaction between the user and the optimization process in order to improve the performances of Pareto-optimal designs.

Actually, for industrial experts, design optimization has great advantages. On the one hand, it allows the designers to explore more alternative solutions to their problem. It is a very good way to encourage the innovation. On the other hand, using design optimization allows the designer to easily make a final decision and justify it with quantitative values related to the problem formulation.

Acknowledgment
The authors would like to acknowledge Thales Communications for the application study.

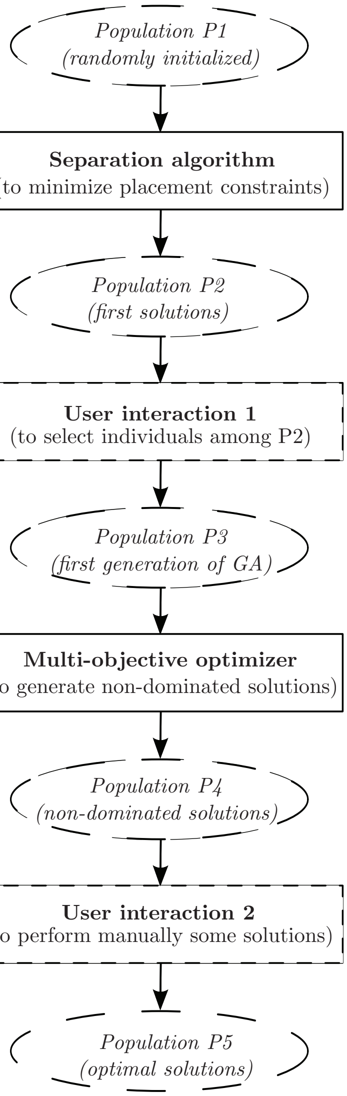


Fig. 10 Schematic representation of the optimization strategy

References

Brintrup AM, Ramsden J, Tiwari A (2007) An interactive genetic algorithm-based framework for handling qualitative criteria in design optimization. Computers in Industry 58:279–291

Cagan J, Shimada K, Yin S (2002) A survey of computational approaches to the three-dimensional layout problems. Computer-Aided Design 34:597–611

Deb K (1998) Multi-objective genetic algorithms: Problem difficulties and construction of test problems. Evolutionary Computation 7:205–230

Drira A, Pierreval H, Hajri-Gabouj S (2007) Facility layout problems: A survey. Annual Reviews in Control 31:255–267

Dyckhoff H (1990) A typology of cutting and packing problems. European Journal of Operational Research 44(2):145–159

Giassi A, Bennis F, Maisonneuve JJ (2004) Multidisciplinary design optimisation and robust design approaches applied to concurrent design. Structural and Multidisciplinary Optimization 28:356–371

Imamichi T, Nagamochi H (2007) A multi-sphere scheme for 2d and 3d packing problems. In: SLS 2007: Proceedings of Engineering Stochastic Local Search Algorithms. Designing, Implementing and Analyzing Effective Heuristics, Springer, Heidelberg, Lecture Notes in Computer Science, vol 4638, pp 207–211, DOI 10.1007/978-3-540-74446-7_19

Imamichi T, Nagamochi H (2008) Designing algorithms with multi-sphere scheme. In: ICKS '08: Proceedings of the International Conference on Informatics Education and Research for Knowledge-Circulating Society (icks 2008), IEEE Computer Society, Washington, DC, USA, pp 125–130

Michalek JJ, Papalambros PY (2002) Interactive design optimization of architectural layouts. Engineering Optimization 34:485–501

Poles S (2003) Moga-ii, an improved multi-objective genetic algorithm. ESTECO, Technical Report 2003-006 p 1/16

Rabeau S, Depince P, Bennis F (2007) Collaborative optimization of complex systems: a multidisciplinary approach. International Journal on Interactive Design and Manufacturing 1:209–218

Serna L, Mejia R, Bennis F, Fischer X (2005) Some ways of implementation of interactive design. In: Virtual Concept 2005, Biarritz

Su Y, Cagan J (2000) An extended pattern search algorithm for threedimensional component layout. Journal of Mechanical Design 122:102–108

Szykman S, Cagan J (1995) A simulated annealing-based approach to three-dimensional component packing. Journal of Mechanical Design 117:308–314

Szykman S, Cagan J (1997) Constrained three-dimensional component layout using simulated annealing. Journal of Mechanical Design 119:28–35

Yi M, Fadel GM, Gantovnik VB (2008) Vehicle configuration design with a packing genetic algorithm. International Journal of Heavy Vehicle Systems 15:433–448