

Solving facility layout problems with strict geometric constraints using a two-phase genetic algorithm

J.A. Diego-Mas*, M.C. Santamarina-Siurana, J. Alcaide-Marzal and
V.A. Cloquell-Ballester

*Departamento de Proyectos de Ingeniería, Universidad Politécnica de Valencia,
Camino de Vera s/n, Valencia, Spain*

(Received 26 March 2007; final version received 4 September 2007)

This article puts forward a two-phase genetic algorithm that is able to solve facility layout problems strictly respecting the geometric constraints imposed on activities. In the first phase the algorithm attempts to locate an optimum slicing tree to group the activities appropriately. In the second phase the layout is obtained from this tree. In order to assess the slicing trees in the first phase we propose an evaluation function able to predict if, by making the appropriate cuts, the tree structure is able to generate layouts that satisfy the geometric restrictions imposed on the facilities to be arranged, and to minimize the cost of transporting materials between the production activities. It also permits the determination of the most suitable aspect ratio of the layout zone in order to minimize non-compliance with the geometric restrictions. The algorithm and the method of calculating the indicator proposed in the evaluation function are described, and the results obtained in the experiments carried out are also given.

Keywords: slicing tree; layout problem; genetic algorithm

1. Introduction

Of the various techniques for solving the facility layout problem (FLP), those based on the slicing tree structure are efficient in the search for geometrically acceptable solutions. In a slicing tree each leaf of the tree is assigned a unique integer corresponding to the identifier of a facility. Each internal node represents the way (vertical or horizontal) a rectangular partition is cut, allocating the facilities in each branch of the node on each side of the partition. In Tam's proposal (Tam 1992b), when the slicing tree has been obtained, the different layouts are generated by modifying the type of cut that is made in the internal nodes of the slicing tree. The slicing tree is created by applying numerical clustering techniques (Tam used the average linkage method), grouping the departments according to the flow of materials between them. The same author (Tam 1992a) used this method of generating layouts by means of slicing trees in a simulated annealing algorithm, and subsequently, in Tam and Chan (1998) he modified his initial proposal, pointing out that

*Corresponding author. Email: jodiemas@dpi.upv.es

the use of a fixed slicing tree in the search for the optimum layout can only cover part of the possible solutions space. Therefore, he introduced a genetic algorithm that simultaneously searches for an optimal tree and a layout obtained from it. This allows searching throughout the entire solution space. This procedure was subsequently improved in Tam and Chan (1998), Gau and Meller (1999), Al-Hakim (2000), Hanafi (2000), Wu and Appleton (2002), and Shayan and Chittilappilly (2004). In these proposals, the geometric quality of the slicing tree is evaluated only on the basis of the quality of the layout obtained from it.

There are thus two approaches: a sequential search for the tree and then for the layout, or a simultaneous search for the grouping of the facilities and for the most suitable types of cut for each slicing tree generated. The sequential approach can only cover part of the possible solution space, since the slicing tree chosen limits the search zone. The simultaneous search attempts to simultaneously optimize both, the slicing tree structure and the cuts that generate layouts, thus widening the search space. In these procedures, the evaluation of the slicing tree is implied in the evaluation of the layouts that it generates. Even if the search space is not limited, these methods are less efficient in the search for possible solutions in a given slicing tree. Using a fixed slicing tree could be attractive if one is able to find an optimum slicing tree according to the different criteria used to evaluate the layouts. In this way the search space would be limited to the most promising zone, which can then be subjected to an intensive search.

In this paper we define a genetic algorithm in two phases. The first phase will attempt to find the optimum slicing tree according to the layout evaluation criteria. The second phase will attempt to obtain the best possible layout from it by modifying the type of cut that is made in the internal nodes of the slicing tree. Since a slicing tree can provide multiple layouts, it is rather difficult to know in the first phase, before the types of cut that will produce the definitive layout are known, if any of them is a good candidate in the search for solutions to reduce to the minimum the distance travelled by materials between different departments. Moreover, it is necessary to know if this good candidate provides geometrically acceptable solutions. Therefore, the first stage in the definition of the genetic algorithm must be to obtain a proper evaluation function of the fitness of the slicing trees.

In the case of the geometric criterion, this could be carried out by means of the *slicing tree's geometric potential*. The slicing tree's geometric potential (Diego-Mas *et al.* 2006) is an estimator of the degree to which a slicing tree is able to generate geometrically acceptable solutions to a given problem. The search could be directed by this estimator by including it in the formulation of the objective function. It would thus be possible to intensify the search in the most promising slicing trees from the geometric point of view, and to discard those that do not comply with the restrictions, whatever cuts are made. The evaluation function will be completed with a new estimator, the *slicing tree's relational potential*. This estimator is an indicator of the capacity of a slicing tree to produce layouts in which the distance between facilities is optimal according to the criterion considered in the problem (cost of material flow, need for closeness between activities due to security, hygiene...).

In this paper, Section 2 will describe the method proposed to evaluate the fitness of the slicing trees. In Section 3 the genetic algorithm in two phases is suggested. In Section 4 the results of using the algorithm in the search of solutions for well-known layout problems will be presented, and in Section 5 we will discuss the proposed method and its results.

2. Evaluation of the slicing trees

The FLP considers the layout of a group of departments of known and unequal areas, with no overlaps, in a flat area of given dimensions, in such a way that costs relating to the interactions between the departments are kept to a minimum. The costs arising from the layout of departments include, in this case, the cost of transporting materials between the facilities in each department and those arising from the non-compliance with certain quantitative and qualitative preferences. To generate a slicing tree, the departments to be arranged have to be grouped according to the criteria chosen to evaluate the solutions. Different layouts can be generated by modifying the type of cut that is made in the internal nodes. Therefore, the slicing tree itself is not a single solution, but rather a set of solutions, accessible through the subsequent determination of the manner in which the cut in each node is carried out to generate the different spaces. To evaluate the capacity of a slicing tree to generate geometrically acceptable layouts, reducing as far as possible the distance travelled by materials between different departments, we propose an additive evaluation function composed of the geometric potential and the relational potential of the tree. The following subsections describe these indicators.

2.1 *The geometric potential of a slicing tree*

The geometric potential of a slicing tree indicates if, given the conditions of the problem under consideration, it will be possible to obtain from this tree, layouts that comply with the geometric requirements demanded by the facilities for all the possible plant ratios[†]. The calculation process of the geometric potential could be found in Diego-Mas *et al.* (2006). In Figure 1, a slicing tree for a 7 free-orientation facilities problem is presented. The area and minimum and maximum ratios of the facilities are also shown. The geometric potential of this tree for each plant ratio is depicted.

Strictly speaking, the geometric potential of a slicing tree measures the incapacity of the tree to generate valid layouts more than its capacity to produce them. That is to say, a tree with a geometric potential equal to 0 for the plant ratio will be able to generate layouts that comply with the geometric restrictions of the problem. A tree with a high geometric potential will be a bad candidate in the search for the optimum layout. The search process of the optimum slicing tree from the geometric point of view will consist of finding the tree with lowest geometric potential. In Figure 1, the geometric potential indicates that, given the conditions of the problem under consideration, it will not be possible to obtain from this slicing tree a layout that complies with the geometric requirements demanded by the facilities for any plant ratio.

It can also be observed that the plant domain ratios between 0.523 and 0.611 (or symmetrically between 0.960 and 1.047), will minimize the geometric non-compliance. Therefore, in those cases in which this initial domain is not previously fixed (the construction of a new plant) it will be possible to determine the most advisable ratio from the geometric point of view. This does not mean that this form is necessarily the best from the point of view of other factors (e.g. the flow of materials between departments).

[†] In this paper we will use the height-width ratio, which can be defined as the height of the facility or the plant divided by its width, but in Figure 1, the geometry of a facility is measured by means of the angle (in radians) formed by the diagonal of its rectangular area and the horizontal (Diego-Mas *et al.* 2006).

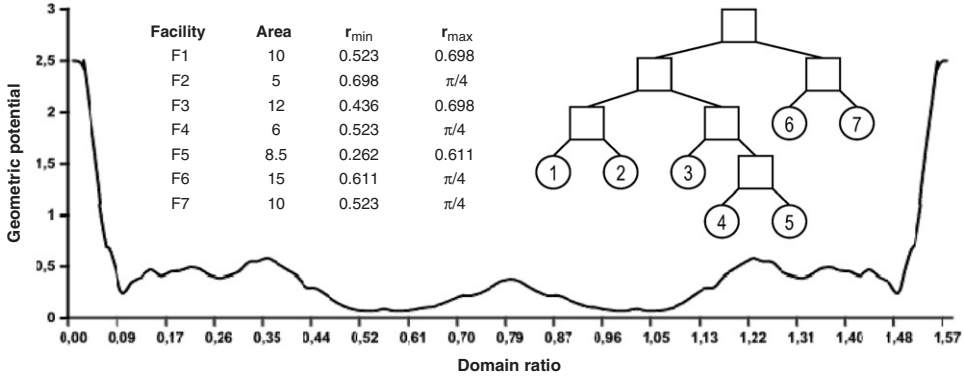


Figure 1. Geometric potential of a slicing tree.

2.2 The relational potential of a slicing tree

The relational potential of a slicing tree will be used to estimate a slicing tree's capacity to generate layouts minimizing the cost of material flow between activities, by allowing the adjacency of the facilities that require proximity according to the criteria defined in the problem. This indicator will be calculated as expression (1):

$$R_p = \sum_{i=1}^n \sum_{j=1}^n \left(P(i, j) \cdot \sum_{u=1}^k \alpha_u \cdot f_{u,ij} \right) \quad (1)$$

In expression (1): $P(i, j)$ is the adjacency potential of the facilities i and j that will be explained further on; n is the number of facilities; k is the number of criteria used to evaluate the slicing tree; α_u is the weight of the criterion u ; and $f_{u,ij}$ is the value of the criterion u for the facilities i and j . For example, one criterion customarily used is the cost of material flow between activities. If this is criterion number 1, $f_{1,34}$ will be the cost of the flow of materials between facilities 3 and 4. In the event of using a qualitative criterion a numerical value will have to be assigned to each level of the criterion.

Given a particular slicing tree, the adjacency potential of two activities $P(i, j)$, estimates the possibility of these two facilities being adjacent in the layouts that are generated from this tree. The adjacency potential is a function of the difference between the levels at which these facilities are located on the tree and the first common node to both facilities. For example, in the slicing tree shown in Figure 3 the facilities 3 and 7 are located on levels 3 and 5 respectively. The first common node to both facilities is the root node, which is located on level 0. The adjacency potential of two activities is defined as:

$$P(i, j) = P(i) \cdot P(j) = \frac{4 \cdot 3^{s-1}}{4^s} \cdot \frac{4 \cdot 3^{s'-1}}{4^{s'}} = \left(\frac{3}{4} \right)^{s+s'-2} \quad (2)$$

In expression (2): $P(i)$ is the adjacency potential of facility i in respect of facility j ; $P(j)$ is the adjacency potential of facility j in respect of activity i ; s is the level on the slicing tree of facility i less the level on the tree of the first common node with facility j ; and s' is the level on the slicing tree of facility j less the level on the tree of the first common node with facility i .

To justify this formula the example given in Figure 3 will be used. As can be appreciated, the difference between the levels of facility 3 and the first common node with facility 7 is 3, therefore $s = 3$. This means that for facility 3 to be definitively located in the plant there have to be 3 cuts made on the domain. Given that there are four possible types of cuts[†] there are $4^s = 64$ possibilities of placing the facility within the group of facilities hanging from the same side of the common node (1, 2, 3, 4 and 5). Given that the facilities are considered to be square, there are 4 sides on which i can be adjacent to facility 7. This will nevertheless only occur if facility 3 has one of its sides on the outer perimeter of the group of facilities 1, 2, 3, 4 and 5. The number of sides of facility 3 which comply with this property on the 4^s arrangements can be calculated.

Let us consider the case of Figure 2. In this example node k is the first common node between facilities i and j . Facility j will always be adjacent to the group of facilities i , a and e . The more sides that facility i has on the outside of said group, the likelihood of it being adjacent to j will be greater. Figure 2 shows all the possible arrangements of the group of facilities, and the forms in which j may turn out adjacent to i in each of these. This value can be calculated as $4 \cdot 3^{s-1} = 36$, where $s = 3$. Clearly the greater number of facilities that have to be grouped with i before j is grouped (that is the greater the s) the fewer the possibilities of the facilities i and j being adjacent. In Figure 3, facilities 3 and 7 have few possibilities of working out adjacent (as can be seen in layouts a) and b) in the figure), though it is possible for this to occur, as in case c). This figure shows the calculation of the adjacency potential between certain activities.

The relational potential of a slicing tree, defined in expression (1), will be greater for the trees in which the facilities which require being adjacent have a greater possibility of proving so.

2.3 The evaluation function

The evaluation function to be used in the first phase of the genetic algorithm for evaluating the slicing trees will now be determined. It has already been stated that the potential of the slicing tree will be a weighted additive function of the two types of potential defined, relational and geometric. The expression (3) will be used:

$$E_f = C_g \cdot G_p + C_r \cdot \frac{1}{1 + R_p}, \quad (3)$$

where: E_f is the value of the evaluation function; G_p is the geometric potential of the slicing tree; R_p is the relational potential of the slicing tree; and C_g and C_r are the weighting coefficients of the relative importance between the geometric criterion and the relational criterion respectively. The relational potential of the slicing tree is a fixed value for each slicing tree. The geometric potential is however a function of the plant ratio. If this ratio is previously set the geometric potential will be the value of this function for that ratio.

The most adapted slicing trees have a greater relational potential, for which reason the relational potential has been introduced in the denominator of one of the terms of the function. To avoid terms under the unit in this denominator, 1 is added to the relational potential. Given this definition of the evaluation function, the optimization process in the first phase of the algorithm will consist of locating the slicing tree with lowest value of the

[†] It is common practice to consider only two types of cuts to be made in each node (vertical or horizontal) (Tam and Chan 1998). Here, however, two possible vertical cuts and two horizontal ones are distinguished, depending on which side of the cut the activities hanging from the node are located on (Tam 1992a, 1992b).

evaluation function; that is, minimizing the geometric and maximizing the relational potential.

3. The genetic algorithm

As has already been stated, the genetic algorithm proposed has a structure in two phases. In the first of the phases, the slicing tree with greatest capacity for generating layouts complying with geometric requirements, and also optimum from the standpoint of the different criteria used in the evaluation, will be determined.

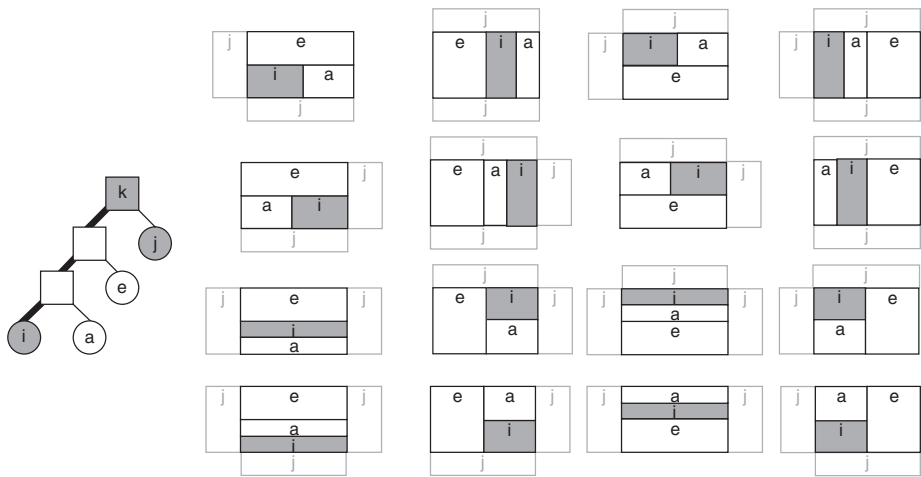


Figure 2. Adjacency between one facility and a group of facilities.

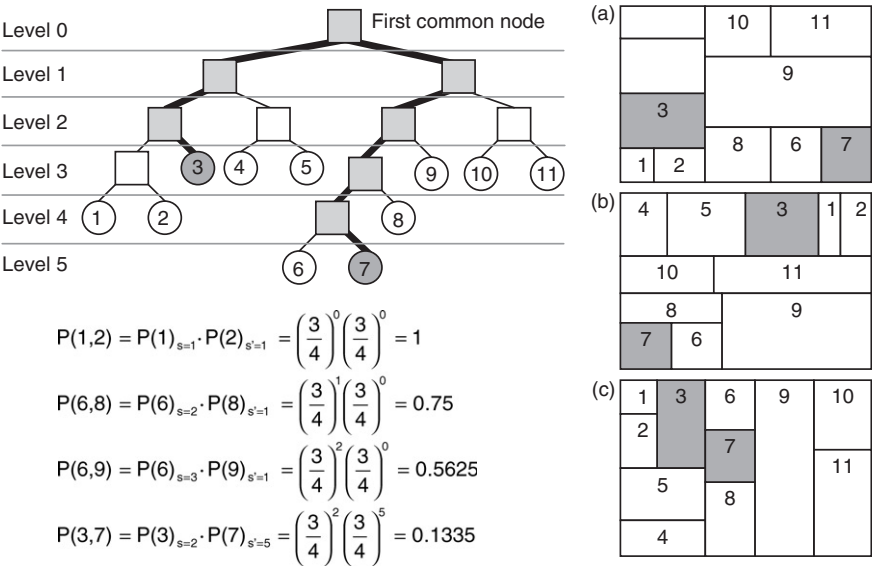


Figure 3. Adjacency potential of two facilities in a slicing tree.

3.1 Search for the slicing tree

To encode the slicing tree a procedure similar to the one described in Tam and Chan (1998) will be used. The coding scheme of these authors represents, by means of a chain with three parts, the structure of the slicing tree, the types of cuts of the internal nodes and the facilities occupying the leaves of the tree. That is, each chain represents a layout. In this case it is only necessary to encode the structure of the tree and the facilities which will be located on the leaves. The second phase of the algorithm will determine the types of cuts in the internal nodes. The part of the chain that encodes the types of cuts will thus be eliminated from the coding scheme.

The first part of the chain will represent the structure of the tree encoded using Gambler's ruin. The second part encodes the facilities which occupy the leaves of the tree by means of the inversion table of a permutation sequence with size N_{fac} (number of facilities), as shown in Tam and Chan (1998). Figure 4 shows an example of the encoding used.

Al-Hakim (2000) noted that the chromosome representation in Tam and Chan was not efficient and suggested some improvements to remove infeasibility. In the algorithm proposed, a mechanism for filtration of the chains representing no feasible solutions is applied, which has given good results in the experiments performed.

The initial population of the slicing trees is generated randomly. The selection of the surviving and reproducing trees is done by roulette wheel selection (Goldberg 1989). After selecting the parent chains, the crossover operator acts by choosing the crossover point as a random number between 1 and $S_1 - 1$, where S_1 is the length of the chain. The offspring will be the result of exchanging the sub-chains of the parents which are left at each side of the crossover point. The offspring replace the parents in the population. The mutation acts by randomly selecting a gene from the chain and modifying its value.

3.2 Search for the layout

After completing the first stage of the algorithm, the structure of the slicing tree for generating layouts complying with the requirements of the problem will have been obtained. In the second phase, the types of cuts that have to be made in the internal nodes

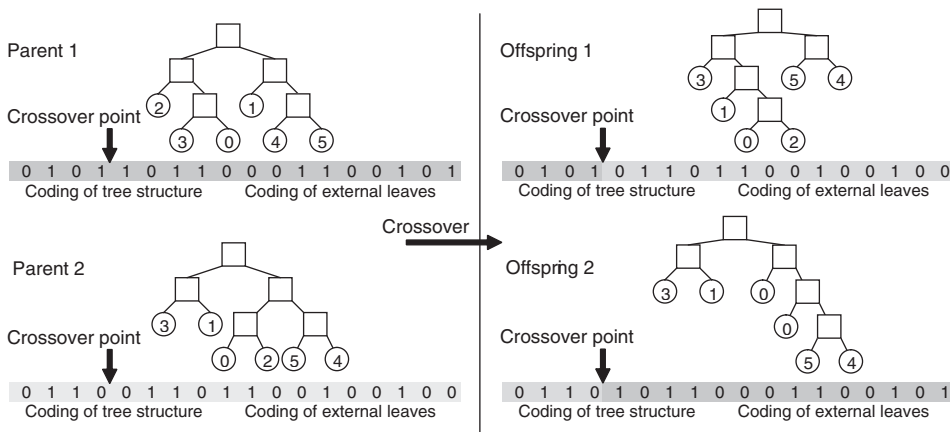


Figure 4. Coding scheme example.

of the tree for generating the best layout that can be obtained from this will be determined. Given that the structure of the tree is now fixed, the algorithm described in Tam and Chan can now be used (Tam 1992b). The difference in this case is that, if the first phase of the algorithm has been done successfully, the second phase will focus the search on a space for particularly promising solutions, given that the slicing tree used will potentially be able to generate good solutions for the problem.

The coding schema used represents only the type of slice made in each node following a pre-order traversal of the tree. There are 4 types of cut: 'u' is a horizontal cut which places the facilities on the left-hand side of the node in the upper sub-domain obtained after the cut; 'b' is a horizontal cut which places the facilities on the left-hand side of the node in the lower sub-domain; 'l' is a vertical cut which places the facilities of the left-hand side of the node in the left-hand sub-domain, and 'r' is a vertical cut which places the facilities of the left-hand side of the node in the right-hand sub-domain. The evaluation function considers both the cost of the flow of materials between the facilities and the non-compliance of the geometric restrictions imposed on the activities.

4. Experimental results

To verify the operation of the genetic algorithm three experiments were performed. In the first we chose the problem considered by Armour and Buffa (1963), one of the best known unequal-area facility layout test problems, to compare the results with the ones obtained by other solution methods. In the second experiment, the problem tackled in Tam and Li (1991) was used. It was proposed as a complex and highly geometrically-restricted model problem. An attempt was thus made to find out the algorithm's capacity to deal with problems with major geometric restrictions. In the third experiment an attempt was made to find out the algorithm's capacity to solve problems in which the areas and the geometric restrictions of the facilities were very different from one another, both in area and in form and degree of formal tolerance. To demonstrate this capacity it was considered advisable to generate a problem whose characteristics were a high number of facilities, with very different spatial needs as regards size and shape, and formal restrictions with different degrees of tolerance.

4.1 Experiment 1

The problem considered by Armour and Buffa (1963) is a problem with 20 free-orientation departments with symmetrical flows (the original flow matrix was corrected as in Scriabin and Vergin (1975), Huntley and Brown (1991) and Tate and Smith (1995)), arranged within a domain with height-width ratio of 2/3. Tate and Smith (1995) offer solutions to the problem for the different maximum permitted height-width aspect ratios[†] equal for all the facilities (1000, 50, 25, 15, 10, 7, 5, 4, 3, 2, 75). In fact, the solutions found for ratios higher than 3 arrange the facilities in departments of excessive length, thus making them unsuitable for any type of productive work. Tate and Smith tried as well to obtain a layout in which the facilities were as near to square as possible. The best of these layouts had the same minimum feasible aspect ratio for all the facilities, 1.706/67, and a cost of the flow of materials between departments of 6662.9. Another solution to the problem is given in

[†] In Tate and Smith (1995) the aspect ratio is defined as the width divided by height of the activity. Since the activities are freely oriented, height/width can be used in the same way.

Wang *et al.* (2005) with a cost of 6777. However, these authors solve the problem by using a discrete model, and the shapes of the departments prove complex in general. In Kim and Kim (1998) the problem is solved by means of a procedure which uses location matrices and a continuous model of space in the plant. From the geometric standpoint Kim and Kim provide the best result for the problem that has been possible to find in the bibliography. The maximum ratio allowed for the activities is 1.4 and the cost of transporting the materials is 6824.66.

Experiments were carried out varying the minimum feasible aspect ratio of all the facilities from 1.70 to 1.35, finding in all cases layouts that complied with the geometric restrictions and with material movement costs lower than those obtained by other authors (Table 1 and Figures 5(a–c)). Furthermore, we tried to obtain a layout in which the facilities were as near to square as possible. The best of these layouts had the same minimum feasible aspect ratio for all the facilities, 1.225, and a cost of material flow between departments of 8897.38 (Figure 5(d)). This cost is higher than the other solutions due to the fact that the required geometric restrictions were much more severe.

Each experiment was performed five times, obtaining feasible solutions in all of these, even in the cases in which geometric restrictions were highly severe. This indicates the soundness of the algorithm proposed in the search for geometrically acceptable solutions. In Tate and Smith (1995) ten runs of the Armour and Buffa test problem were performed for each ratio, of which only three feasible executions were obtained for the ratio 1.75, and eight for ratio 1.706/67.

As indicated in Diego-Mas *et al.* (2006), the geometric potential of the slicing tree enables determination of the most convenient plant ratio for the activities to meet the geometric requirements imposed. Therefore, in those cases in which this initial domain is not previously fixed (the construction of a new plant where the shape of the building can be adapted to the requirements of the production process) it will be possible to determine the most advisable one from the geometric point of view. To this end, after completing the first phase of the algorithm, the ratio of the plant providing lowest geometric potential of the slicing tree obtained within a certain interval is selected. Hence, by allowing the plant ratio to vary in the [0.4, 2.5] interval, it was possible to locate solutions for maximum feasible aspect ratio of the facilities of 1.12. The best solution obtained had a cost of 8646.19 with a plant ratio of 0.435 (Figure 5(e)).

Table 1. Costs of the solutions for the Armour and Buffa (1963) test problem.

Maximum feasible aspect ratio of the facilities	Tate and Smith (1995)	Kim and Kim (1998)	Two phase algorithm
2.000	6171.1		
1.750	7205.4		
1.706/67	6662.9		
1.700			6084.62
1.550			6181.50
1.400		6824.66	6553.46
1.350			6632.65
1.225			8897.38

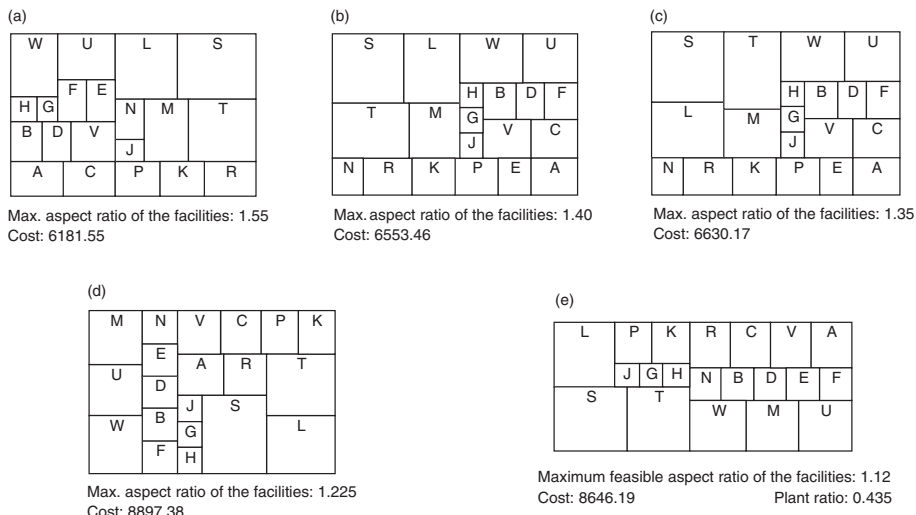


Figure 5. Solutions of the Armour and Buffa (1963) test problem.

4.2 Experiment 2

The problem described in Tam and Li (1991) consists of 30 facilities of very different sizes whose areas range from 3 to 36 units, with an average of 11.93 and a typical deviation of 8.55, all with free orientation. The ratios allowed for each facility are also very diverse. In some cases the geometrical requirements are very flexible and in others very strict. There are four activities with no geometrical flexibility which are required to be perfect squares (activities 17, 18, 27 and 28). In the adaptation of the problem to solve this by means of the algorithm proposed, these activities were given a minimum flexibility, enabling their aspect ratios to vary through interval $[0.95, 1]$. The plant ratio is not set.

The aforementioned article presents a method based on a *divide and conquer* strategy for solving the plant layout problem respecting the geometric constraints. The method divides the problem into several groups of activities by means of clustering techniques, generating an independent layout for each group of activities. Later on a global layout is generated considering each cluster as a major activity. The drop in effectiveness meant by dividing the problem into sub-problems and generating independent layouts for each of these is considered by the authors as a minor disadvantage which on the other hand enables large problems with major geometric restrictions to be tackled. The solutions obtained by this system are not compact, with numerous dead areas between activities (Figure 6(a)).

Given that Tam and Li do not indicate the form of location domain, two experiments were performed to solve the problem, the first setting the plant ratio as 1 (square plant), and the second letting the algorithm determine the most appropriate ratio within an interval $[0.5, 1]$. In the first case a better result was obtained by the layout shown in Figure 6(c), with a cost due to the flow of materials of 17809.38. In the second case the layout of Figure 6(b) was obtained, with a plant ratio of 0.76 and a material transport cost of 17058.4.

In Tam and Li (1991) no data is provided about the solution proposed, nor distances between activities or material transport costs. They only indicate the value of the objective



Figure 6. Solutions of the Tam and Li (1991) test problem.

function for the partial layouts with which the final solution was generated. To act as a reference for comparing this solution with the solutions obtained with the two-phase algorithm, an estimation of the cost of material transport for the solution proposed and the area occupied by the facilities was made. A value of 19 880 units was obtained for the cost[†], determining that a domain of 27.87×20.25 length units was required, which entails a surface area of 564.52 area units. Given that the sum of the areas required by the activities is 358, the percentage of surface area not occupied is 36.6%[‡].

In the solutions obtained by means of the two-phase algorithm the cost due to material flow is under that of the solution provided in Tam and Li. The solutions also have a high geometric quality, the strict geometric constraints imposed in the original problem are respected and compact layouts without dead areas are obtained, which has an influence on the lower cost of material transport. This reduces the size of the area needed to house the production process, this being 63.4% of the size required to contain the layout obtained by means of Tam and Li's hierarchical method.

4.3 Experiment 3

The problem set is the layout of 40 free orientation activities in a plant with a height-width ratio of 0.9. The areas, maximum ratios and minimum ratios of the departments were

[†]The Euclidean metric was used in this calculation.

[‡]The estimation made is bound to be inaccurate. However, after obtaining this, it could be used to determine the material transport cost of the partial layouts generating the solution, being very similar to the ones given by the authors.

generated at random, later introducing certain modifications to ensure that the problem complied with the objectives for which it was set. The full data on this, including the material flow between activities, is given in the Appendix. This shows that in some cases it is required for the shape of the facilities to be practically square (for example facilities 12 and 13), in others it is necessary for the shape to be slender (for example facilities 14, 17, 19 or 40). There are facilities with high formal tolerances (such as facilities 1, 4 or 14) and others with strict ones (2, 13 or 38)[†].

The experiment was repeated ten times. In all of these the first phase of the algorithm was able to locate a slicing tree with no geometrical potential, which guaranteed the existence of geometrically acceptable layouts stemming from said trees. In all the executions the second phase of the algorithm located layouts complying with the geometrical restrictions. On average 1067.6 generations of the first phase and 1194.8 of the second were needed to locate the best distributions. The average cost due to the material flow of the ten distributions obtained was of 338 092.64 units. The best of the distributions is the one presented in Figure 7(a). This was obtained in the third execution and the cost associated with the material flow between activities in this is 326 623.53 units.

Furthermore, an experiment was made considering that the height-width ratio of the plant was not set *a priori* and could range from 0.5 and 1. The cost associated with material transport of the best layout obtained was 324 274.16 units, which represents a reduction of 2349.37 units in respect of the best solution obtained with a fixed plant ratio. The height-width ratio of the plant housing this solution is 0.72 (Figure 7(b)).

5. Discussion and conclusion

For the solutions obtained to an FLP to be really applicable they must necessarily be subject to certain geometric restrictions. The shapes of the different departments must allow the machines used in them to be correctly arranged, and must also allow the workers to carry out their activities with no hindrance. Rectangular areas are usually required, with certain maximum and minimum limits in the height-width ratio of the departments. Solutions that do not conform to these limits are considered unacceptable. However, as has been pointed out by Levary and Kalchick (1985), Kusiack and Heragu (1987) and Tam and Chan (1998), many studies on the subject have either ignored the geometric restrictions, or have established unreasonably large validity intervals for the ratios of the departments. In the real world, in one and the same production process, a specific facility may require a large narrow space with little dimensional tolerance, another requires a smaller space with great formal tolerance allowing it to take diverse forms, and another may, for example, require a slightly long shape in the vertical sense with no tolerance in the narrowness. The algorithm presented in this work enables location of layouts in which the areas and geometrical restrictions of the facilities are different to one another, both in the form and the degree of tolerance. The geometrical potential of the slicing tree will act as an indicator of the capacity of the trees examined to generate appropriate layouts, even in cases of activities with very different shapes, sizes and restrictions.

The experiments carried out have vouched for the fact that the cost associated with the material flows of the solutions found also improves on the solutions proposed by

[†] One should remember that the use of the height-width ratio as a measurement of the form of the areas of the activities causes the non-existence of linearity between the difference in ratios and the difference of forms. Hence, a difference in ratios of 0.1 when these take values close to the unit entails figures that are more similar to each other than if these ratios have values farther from the unit.

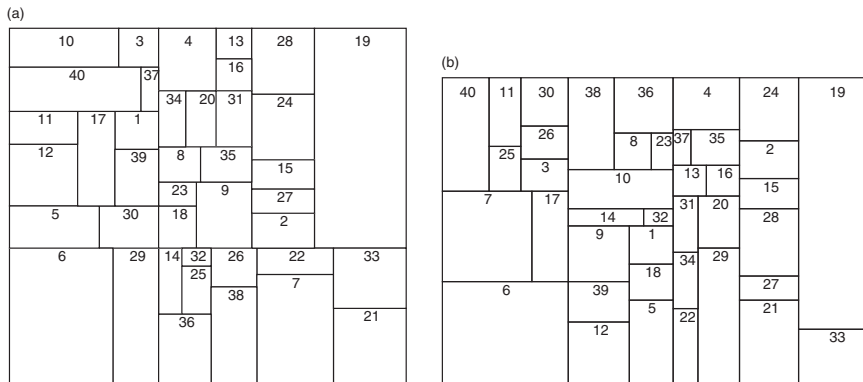


Figure 7. Best solution for the test problem: (a) fixed plant ratio and (b) free plant ratio.

other authors. The algorithm proposes a sequential solution of the problem: the first phase attempts to find the optimum slicing tree according to the layout evaluation criteria; and the second phase attempts to obtain the best possible layout from it by modifying the type of cut that is made in the internal nodes of the tree. This sequential approach limits the space for searching for solutions to the ones which can be obtained from the slicing tree generated in the first phase. The first search phase has to be done efficiently to avoid restricting the search in the second phase to a zone of the solution space that has little promise.

In the first resolution phase there is no clear idea of what layout will finally be adopted, or even of the shape of the site area. This uncertainty makes it hard to assess the quality of the slicing trees. The evaluation function proposed for the first phase uses the *geometric potential of the slicing trees* as an indicator of the capacity of a slicing tree to generate layouts that comply with the geometric requirements of a given problem. Apart from this the *relational potential of a slicing tree* is put forward for estimating a tree's capacity to generate layouts minimizing the cost of the flow of materials between activities and which enable the adjacency of the facilities which require this. The experimental results show the efficiency of these indicators.

Finding the geometric and relational potentials involves a considerable amount of calculation, and the time taken up in this process will vary according to the number of items in the layout and, to a lesser extent, depending on the size of the geometric validity intervals of the items and of the site domain. An efficient software implementation will have a decisive influence on the calculation time. In the 40 facilities problem shown in Section 4, on a computer with a 1.5 GHz processor and 1 Gb of RAM, the average processing time was 323.5 seconds. 89.6% of the time was used in the first phase of the algorithm.

Finally, it could be of interest to extend the scope of the application to non-rectangular domains with possible internal restricted areas, and to different approaches to the FLP, like a dynamic facility layout problem or multiple level layout problems (Conway and Venkataramanan 1994, Zhang *et al.* 2002, Balakrishnan *et al.* 2003).

Acknowledgments

We would like to thank the R&D+i Linguistic Assistance Office at the Universidad Polit cnica de Valencia for their help in translating this paper.

References

- Al-Hakim, L., 2000. On solving facility layout problems using genetic algorithms. *International Journal of Production Research*, 38, 2573–2582.
- Armour, G.C. and Buffa, E.S., 1963. A heuristic algorithm and simulation approach to relative location of facilities. *Management Science*, 9, 294–309.
- Balakrishnan, J., Cheng, C.H. and Wong, K.F., 2003. FACOPT: a user friendly FACility layout OPTimization system. *Computers & Operations Research*, 30, 1625–1641.
- Conway, D.G. and Venkataramanan, M.A., 1994. Genetic search and the dynamic facility problem. *Computers and Operations Research*, 21, 955–960.
- Diego-Mas, J.A., Santamarina-Siurana, M.C., Cloquell-Ballester, V.A. and Alcaide-Marzal, J., 2006. Slicing tree's geometric potential: an indicator for layout problems based on slicing tree structure. *International Journal of Production Research*, December 2006, DOI 10.1080/00207540600932061. Available online at: <http://dx.doi.org/10.1080/00207540600932061>
- Gau, K.Y. and Meller, R.D., 1999. An iterative facility layout algorithm. *International Journal of Production Research*, 37, 3739–3758.
- Goldberg, D.E., 1989. *Genetic algorithms in search, optimization and machine learning*. Reading, MA: Addison-Wesley Publishing Company Inc..
- Hanafi, R., Design of a genetic algorithm to solve the facility layout problem. Minor thesis, IRIS, Swinburne University of Technology, Melbourne, 2000.
- Huntley, C.L. and Brown, D.E., 1991. A parallel heuristic for quadratic assignment problems. *Computers and Operations Research*, 18, 275–289.
- Kim, J.G. and Kim, Y.D., 1998. A space partitioning method for facility layout problems with shape constraints. *IIE Transactions*, 30, 947–957.
- Kusiack, A. and Heragu, S.S., 1987. The facility layout problem. *International Journal of Production Research*, 29, 229–251.
- Levary, R.R. and Kalchick, S., 1985. Facilities layout-a survey of procedures. *Computers & Industrial Engineering*, 9, 141–148.
- Scriabin, M. and Vergin, R.C., 1975. Comparison of computer algorithms and visual-based methods for plant layout. *Management Science*, 22, 172–181.
- Shayan, E. and Chittilappilly, A., 2004. Genetic algorithm for facilities layout problems based on slicing tree structure. *International Journal of Production Research*, 42, 4055–4067.
- Tam, K.Y. and Li, S.G., 1991. A hierarchical approach to the facility layout problem. *International Journal of Production Research*, 29, 165–184.
- Tam, K.Y., 1992a. A simulated annealing algorithm for allocating space to manufacturing cells. *International Journal of Production Research*, 30, 63–87.
- Tam, K.Y., 1992b. Genetic algorithms, function optimization, and facility layout design. *European Journal of Operational Research*, 63, 322–346.
- Tam, K.Y. and Chan, S.K., 1998. Solving facility layout problems with geometric constraints using parallel genetic algorithms: experimentation and findings. *International Journal of Production Research*, 36, 3253–3272.
- Tate, D.M. and Smith, A.E., 1995. Unequal-area facility layout by genetic search. *AIIE Transactions*, 27, 465–472.
- Wang, M.J., Hu, M.H. and Ku, M.Y., 2005. A solution to the unequal area facilities layout problem by genetic algorithm. *Computers in Industry*, 56, 207–220.
- Wu, Y. and Appleton, E., 2002. The optimization of block layout and aisle structures by a genetic algorithm. *Computers & Industrial Engineering*, 41, 371–387.
- Zhang, G.Q., Xue, J. and Lai, K.K., 2002. A class of genetic algorithm for multiple-level warehouse layout problems. *International Journal of Production Research*, 40, 731–744.

Appendix: data of the problem proposed: areas of the facilities, maximum and minimum ratios allowed, and flow of materials between the facilities

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	Area	r _{min}	r _{max}
1	*	10	9	20	10	11	7	11	8	15	22	6	2	15	13	8	12	11	7	9	3	6	14	9	17	12	3	16	2	15	13	3	5	11	15	13	13	2	11	14	56.843	0.51	1
2		*	10	4	7	16	14	6	15	2	14	13	6	14	15	4	12	10	15	14	17	5	4	37	14	16	15	10	14	9	7	10	16	10	13	9	8	14	10	79.943	0.56	0.70	
3			*	15	16	7	16	9	11	5	7	37	8	17	4	5	12	14	17	5	7	5	12	13	14	11	13	6	14	10	4	11	9	9	6	6	15	8	43	58.103	0.57	1	
4				*	3	11	17	5	9	11	4	5	5	14	14	16	13	11	22	4	3	7	42	3	13	11	13	9	3	12	12	5	12	8	26	15	16	2	27	7	129.315	0.49	1
5					*	14	14	7	16	4	12	80	16	16	6	7	3	13	35	5	14	16	12	14	16	6	14	6	50	10	1	2	3	7	7	13	13	6	4	7	138.165	0.47	0.65
6						*	70	8	8	11	10	12	15	12	7	16	9	11	9	3	2	17	16	15	8	8	9	14	8	10	13	12	7	7	12	7	2	16	5	8	491.470	0.76	1
7							*	3	2	13	5	6	6	16	16	4	51	7	11	4	15	6	7	10	9	6	12	15	2	8	16	10	3	14	13	9	13	15	9	297.180	0.69	1	
8								*	17	13	11	11	9	6	12	4	5	14	16	6	4	3	15	9	6	14	5	10	13	9	6	4	25	15	7	5	6	6	4	51.800	0.73	0.98	
9									*	0	16	14	17	3	8	13	13	62	5	15	19	10	10	16	10	17	8	4	4	4	11	16	4	19	10	14	15	39	12	8	126.968	0.76	0.92
10										*	3	15	16	7	10	12	13	2	5	7	6	2	12	17	5	2	12	14	2	2	9	7	7	8	9	10	12	7	11	156.980	0.31	0.43	
11											*	5	17	4	16	4	15	11	12	9	3	3	7	11	10	15	6	3	8	11	63	4	14	16	9	7	10	16	11	16	80.645	0.34	0.49
12												*	13	7	11	14	12	5	3	9	13	3	14	8	2	16	4	16	4	6	13	9	4	14	5	7	3	15	2	151.130	0.83	1	
13													*	9	12	9	4	3	42	5	11	14	4	11	10	4	9	7	3	43	4	9	4	16	4	8	9	10	14	38.735	0.83	1	
14														*	14	3	12	8	3	4	3	15	5	10	16	5	14	11	12	56	5	14	8	14	8	11	17	15	10	5	50.165	0.15	0.33
15															*	12	6	17	8	17	32	2	4	42	14	10	15	10	16	14	13	8	10	16	7	6	3	9	4	11	68.150	0.45	0.75
16																*	10	9	76	11	7	17	12	16	11	16	2	10	12	15	9	10	3	12	1	5	7	4	10	37.463	0.80	1	
17																	*	10	6	16	10	12	7	7	14	4	10	4	9	14	9	5	7	14	4	10	4	15	13	5	120.968	0.32	0.46
18																		*	8	4	13	14	12	9	6	7	9	14	2	2	12	13	15	14	8	14	15	6	16	55.770	0.67	0.92	
19																			*	14	6	16	8	9	11	7	3	105	16	5	15	12	90	12	12	11	4	16	8	5	698.500	0.25	0.41
20																			*	12	16	34	10	5	9	15	9	16	16	10	16	8	15	17	15	5	5	13	2	80.290	0.66	0.96	
21																				*	28	2	4	5	3	11	17	10	8	5	14	93	4	17	6	17	7	6	4	191.770	0.62	1	
22																				*	12	16	6	8	5	13	13	2	16	13	35	8	30	16	8	9	9	10	70.300	0.30	0.44		
23																				*	10	14	13	9	3	9	8	12	11	9	16	5	16	6	2	3	14	31.450	0.57	0.69			
24																				*	2	10	11	10	11	12	5	11	15	6	4	13	14	11	4	10	144.525	0.81	1				
25																				*	16	10	16	6	14	7	10	13	5	13	8	12	10	14	70	51.118	0.59	0.78					
26																				*	11	5	4	15	4	15	4	15	9	3	9	15	11	15	17	10	60.643	0.66	0.89				
27																				*	9	9	12	6	14	9	4	6	4	16	10	8	16	49.513	0.28	0.41							
28																				*	10	11	30	17	7	13	7	13	10	9	32	14	147.955	0.73	1								
29																				*	7	3	14	5	7	3	6	10	9	11	15	214.650	0.29	0.38									
30																				*	7	5	14	4	3	7	14	3	5	88.245	0.68	1											
31																				*	4	10	14	4	15	4	10	9	5	52.725	0.42	0.55											
32																				*	12	6	10	4	5	14	17	9	18.560	0.55	0.79												
33																				*	15	4	7	13	17	6	12	152.440	0.55	0.85													
34																				*	7	10	8	12	4	5	50.350	0.42	0.52														
35																				*	3	15	4	15	10	64.463	0.68	0.98															
36																				*	5	62	3	54	126.140	0.69	1																
37																				*	3	5	13	26.270	0.39	0.53																	
38																				*	3	9	151.340	0.41	0.50																		
39																				*	3	87.945	0.54	0.79																			
40																				*	203.645	0.30	0.41																				

Copyright of International Journal of Production Research is the property of Taylor & Francis Ltd and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.