MATHEMATICAL PROGRAMMING MODELS FOR SOLVING UNEQUAL-SIZED FACILITIES LAYOUT PROBLEMS - A GENERIC SEARCH METHOD

R. Tavakkoli-Moghaddam

Department of Industrial Engineering, Faculty of Engineering
Tehran University, P.O. Box 11365-4563, Tehran, Iran

(Received: July 13, 1998 - Accepted in Revised form: June 3, 1999)

Abstract This paper present unequal-sized facilities layout solutions generated by a genetic search program. named LADEGA (LAyout DEsign using a Genetic Algorithm) [9]. The generalized quadratic assignment problem requiring pre-determined distance and material flow matrices as the input data and the continuous plane model employing a dynamic distance measure and a material flow matrix are discussed. Computational results on test problems are reported as compared with layout solutions generated by the branch - and bound algorithm a hybrid method merging simulated annealing and local search techniques, and an optimization process of an enveloped block.

Key Words Facilities Layout Problems, Mathematical Models, Heuristics, Genetic Algorithms

چکیده در این مقاله حل‌هایی از استقرار (جانمایی) ماشین‌آلات با ابعاد مختلف ارایه می‌گردد. در این خصوص یک برنامه جستجوژنتیکی به نام LADEGA (طراحی استقرار با استفاده از الگوریتم ژنتیک) نوشته شده است. دو مدل برنامه‌ریزی ریاضی جهت فرموله کردن مسایل استقرار تسهیلات ارایه خواهد شد: ۱) مدل مسأله تخصیص مضاعف عمومی شده که نیاز به ماتریس مسلفت از پیش تعیین شده و جریان مواد (هر دو به عنوان اطلاعات ورودی) دارد. ۲) مدل صحفه پیوسته که نیاز به اندازه‌گیری مسافت دینامیکی (به عنوان متغیر تصمیم‌گیری) و ماتریس جریان مواد (به عنوان اطلاعات ورودی) دارد. چندین مسأله مورد امتحان و بررسی خواهد گرفت که نتایج حاصله برنامه با نتایج حاصله از: ۱) روش Branch and Bound ۲) روش هیبریدی متشکل از روش‌های جستجو موضعی و Simulated Annealing و ۳) روش بهینه سازی بلوکی مقایسه خواهد شد.

1. INTRODUCTION

Manufacturing facilities layout design is a class of NP-hard optimization problems [8]. A great number of algorithms and methods have been introduced and proposed to solve equal and unequal-sized facilities layout problems [7, 12]. The quadratic assignment problem (QAP) [6] is one of the combinatorial optimization models for equal-sized facilities layout problems. Computational results of equal-sized facilities layout problems have been reported by Tavakkoli-Moghaddam and Shayan [10].

In this paper, a generalized quadratic assignment problem (G-QAP) and a continuous plane formulation are used to solve unequal-sized facilities (or machines) layout problems. In the former formulation, a pre-determined distance matrix must be first constructed. However, in the latter one, a dynamic distance between each pair of facilities is measured by a center - to - center and rectilinear distance method to find the objective value. The goal of these models is to assign all the facilities to the given boundaries (areas) in such a way that the total material handling cost is minimized. In the continuous plan model, the computational time for evaluating the linear objective function is much less than the quadratic objective function used in the G-QAP.

To optimally solve such problems, an

efficient optimizing algorithm can hardly be found to generate layout solutions in a reasonable time when the problem size is more than 20 equal-sized facilities (or locations), due to combinatorial explosion occurrence in the problem space.

Due to the above limitation, many intelligent heuristic techniques have been proposed [12]. From among these heuristic techniques, genetic algorithms (GAs) are chosen as a robust and stochastic search method in order to find feasible and promising solutions within a reasonable computation time. Based on the GA methodology, a specific GA program named LADEGA LAyout DEsign using a Genetic Algorithm [9] has been implemented on an IBM compatible PC to solve each test problem conducted in this paper.

2. MATHEMATICAL PROGRAMMING MODELS

The quadratic assignment problem (QAP) model cannot solve unequal-size facilities layout problems. However, Bazaraa [1] modified a quadratic set covering model to solve unequal-sized facilities layout problems resulting in the formulation of the generalized quadratic assignment problem (G-QAP). The following mathematical programming named G-QAP Model is used to solve two test problems of 12 and 14 facilities. These facilities must be assigned to 58 and 63 standard (equal-sized) blocks (locations) respectively.

The number of locations indicates the size of the distance matrix ($N \times N$). A centroid and rectilinear distance between each pair of locations is measured. In the G-QAP Model, the number of locations is greater than the number of facilities ($N>m$). However in the QAP, the number of facilities is equal to the number of locations. In the case of unequal-sized facilities layout problem, each facility consisting of one or greater than the number of equal blocks can take irregular shapes.

G-QAP Model

$$Min \sum_{i=1}^{m} \sum_{j=1}^{N} \sum_{k=1}^{m} \sum_{l=1}^{N} \frac{f_{ik}\, d_{jl}\, x_{ij}\, x_{kl}}{n_i \times n_k} \qquad (1)$$

s.t.:

$$\sum_{j=1}^{N} x_{ij} = n_i \qquad i = 1,2,\ldots\ldots,m \qquad (2)$$

$$\sum_{i=1}^{m} x_{ij} \le 1 \qquad j = 1,2,\ldots\ldots,N \qquad (3)$$

$$x_{ij} = 0 \text{ or } 1 \qquad \forall i,j \qquad (4)$$

where,

$$x_{ij} = \begin{cases} 1 & \text{if facility } i \text{ is assigned to block } j \\ 0 & \text{otherwise,} \end{cases}$$

m = number of facilities,

N = total number of blocks occupied by all the m facilities and a dummy facility,

f_{ik} = material flow between two facilities i and k,

d_{jl} = distance between two blocks j and l,

n_i = total number of blocks for facility i,

The purpose of the above model is to assign each facility to a number of blocks, depending on the facility area, with the minimum objective value representing the material handling cost. The first constraint (Eq. 2) ensures that the total number of blocks occupied by facility i are correctly met and the second constraint (Eq. 3) ensures that each block is only used by one facility. The last Equation (Eq. 4) defines the binary decision variables for the G-QAP Model.

The following is the continuous plane model named Cont-l Model for unequal-sized facilities layout problems as well. The dimensions of the floor plan and each facility are known *a priori*. However the user may change the floor dimensions to generate alternative layouts.

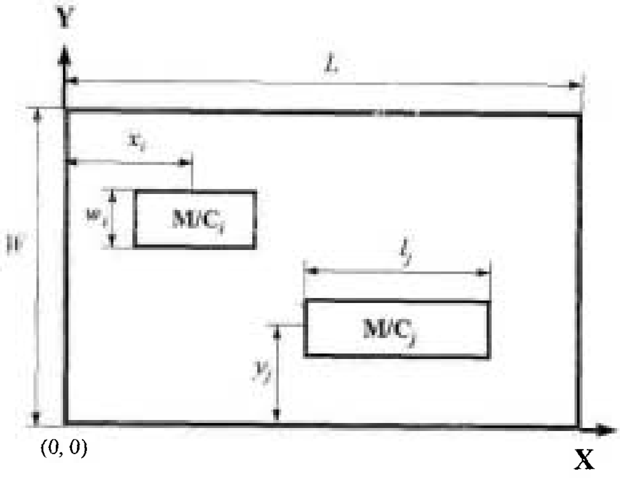


Figure 1. Schematic of facilities (or machines) layout desing.

Figure 1 helps to follow the notations used in the mathematical model. The reference coordinate is assumed to be the lower-left hand corner of the floor plan. The objective is to find the locations of the center coordinates of facilities (decision variables) so that the total flow-distance cost is minimized.

Cont-1 Model:

$$Min. \sum_{i=1}^{n} \sum_{j=1}^{n} f_{ij} \left[|x_i - x_j| + |y_i - y_j| \right] \quad (5)$$

s.t.

$$|x_i - x_j| \geq \frac{1}{2}\left[l_i + l_j\right] \qquad \forall_i \; and \; \forall_j \quad (6)$$

$$|y_i - y_j| \geq \frac{1}{2}\left[w_i + w_j\right] \qquad \forall_i \; and \; \forall_j \quad (7)$$

$$|x_i - x_j| \leq L - \frac{1}{2}\left[l_i + l_j\right] \qquad \forall_i \; and \; \forall_j \quad (8)$$

$$|y_i - y_j| \leq W - \frac{1}{2}\left[w_i + w_j\right] \qquad \forall_i \; and \; \forall_j \quad (9)$$

$$x_i, y_i \leq 0 \qquad \forall_i \quad (10)$$

where,

n = number of facilities to be assigned,

f_{ij} = frequency of material flow between facilities i and j,

l_i and w_i = length and width of the horizontal and vertical side of facility i,

x_i and y_i = x and y - coordinates of the centroid of facility i,

L and W = length and width of the floor plan respectively.

Any of the first two set of constraints (Eqs. 6 and 7) ensures the non-overlapping condition for each pair of facilities. The set of constraints given in Eqs. 8 and 9 ensures that each facility is assigned within the floor boundaries and the last set of constraints (Eq. 10) is non-negative decision variables. Additional constrainst are given in the next section. The input data (parameters) to the model are the number of facilities, floor dimensions, facilities dimensions, and material flow between each pair of facilities. In this paper a test problem of 20 facilities [5], named Imam-20, has been also carried out using the Cont-1 Model.

The above model considers only rectangular facilities. If a facility is of irregular shape, then a rectangular block is needed to fit the irregular facility. The model finds the x and y coordinates (real values) of all the facilities to represent a layout solution to a given problem. In contrast, in the G-QAP Model, The distance between each of the paired blocks is given and the model determines how each facility will fit with each set of blocks (pre-determined locations).

3. GENETIC ALGORITHMS

Genetic algorithms [3,4] are parallel search techniques based on the mechanics of natural selection and natural genetics. They work on a set of feasible solutions called a population of chromosomes indicating a sequence of locating a facility into the relevant location. Each feasible solution is encoded into a chromosome represented as a permutation of integers. The goodness (fitness value) of each chromosome is then evaluated by a fitness function, a mapping of the objective function of a facilities layout problem. The objective function can be expressed as the total costs incurred by the material flow between each pair of facilities and distance between the two relevant locations and/or any other measures.

New chromosomes are created by duplicating a few members of the current population based on the selection scheme and by mating member (s) selected from the current population. This process is called a "generation" and it is expected that the average quality of layout solutions be improved in the succeeding generation.

A general approach can be constructed using the following main steps:

1. Select a good representation (genetic encoding) to encode the search space.
2. Determine an appropriate fitness function and scaling method.
3. Find a good selection scheme.
4. Choose and design genetic operators.
5. Set the GA parameters to control and terminate the process.
6. Terminate the algorithm when the stoppling condition is met.
7. Report the best solution searched so far.

A number of typical test problems have been demonstrated and solved by genetic algorithms, one of which can be found in Goldberg [3]. Figure 2 is a flow chart related to simple genetic algorithm with a reproduction method. It should be noted that the reproduction rate is computed as follows: reproduction rate = 1- (crossover rate + mutation rate).

Based on the G.A concept, a computer program has been implemented to run the GA program named LADEGA. This program generates promising layout solutions to the given facilities layout problems. A few test problems of unequal-sized facilities/machines considering a number of constraints are conducted to examine the validity of the above program. To solve the layout problems tested in this paper by LADEGA, the genetic coding (chromosome or genotype) of a layout solution phenotype) must be designed. LADEGA always generates valid chromosomes by satisfying all the constraints (Eqs. 6-10) of the Cont-1 Model which forms the chromosome. At the initial stage, all the chromosome bits are set to zero. The following pseudo code is illustrated in order to generate a valid chromosome in the initial population.

Generate (valid chromosome)

Read the GA parameters and input data.

do

Initialize (or set) all the chromosome bits to zero. Not all the facilities are processed. Select a facility at random.

Assign the selected facility to the top-left corner of the available floor plan, either horizontally or vertically depending on how dimensions of the facility are stored in the input file.

Deduct the facility area from available space.

Allocate the facility number to the respective locations of the chromosome (this facility is processed).

for (1 to N, number of blocks)

if (the chromosome bit is zero) then

Select an unprocessed (new) facility at random.

if (the new selected facility neither exceeds the floor boundaries nor overlaps with another facility, ie. it does not violate the constraints.) then

Assign the new selected facility to the right

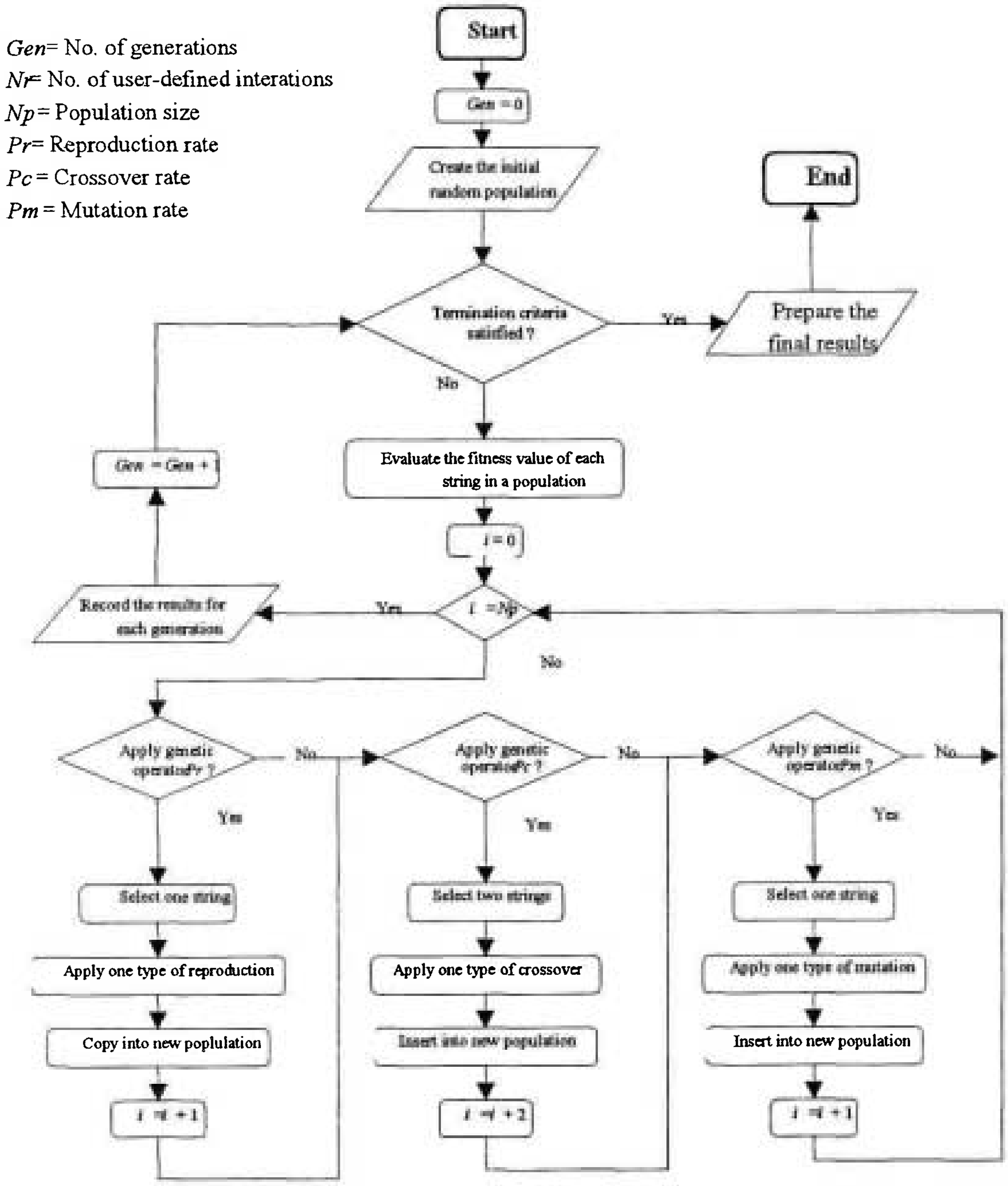


Figure 2. Flow chart related to simple genetic algorithm with a reproduction method.

of the previous facility assigned to the floor plan or to the new row of the floor plan (this facility is now prosessed.)
else if (the selected facility neither exceeds the floor boundaries nor overlaps with another facility, ie. it does not violate the constraints.) then
Assign the new selected facility to the right of the previous facility assigned to the floor plan or to the new row of the floor plan (this facility is now processed and rotated).
else (the selected facility is not processed)

end if

end for

end do (till a valid chromosome is created, ie. all the facilities are assigned and processed)

Figure 3a demonstrates the structure of a genetic coding (chromosome) representing a valid layout solution of the Imam-20 case, as shown in Figure 2c. According to the floor dimensions (W=12 and L=9), there will be 108 equal-sized blocks to house all the 20 facilities requiring 101 blocks and dummy facilities which require 7 blocks. It should be noted that the chromosome length for the above layout solution (Phenotype) is 108 (12 × 9), where the location configuration is given in Figure 2*b*.

4. COMPUTATIONAL RESULTS

The best objective value found by LADEGA for the Baz-12 case is 25,738 which is associated with a layout solution as shown in Figure 3*a*. The above objective value has been reported by Tavakkoli-Moghaddam and Shayan [11]. This layout solution summarized in Table 1 is superior to the solutions reported by Bazaraa [1] and Bazargan-Lari and Kaebernick [2], in which the solution quality is defined as follows:

$$\text{Solution quality} = \left[1 + \frac{\text{Solution (reported by others - obtained by LADEGA)}}{\text{Lower solution obtained(or reported)}}\right] * 100\%$$

It should be noted that the LADEGA's solution is compared with the solution generated by the branch-and-bound technique, due to the same space utilization. The space utilization of the layout solution generated by LADEGA is the same as that of Bazaraa [1] and 3% better than the layout solution reported by Bazargan-Lari and Kaebernick [2]. The GA parameters are: population size-10000; number of facilities-12; chromosome length-58; generation number-1000; crossover rate-0.8; mutation rate-0.1; random seed-0.5; and tournament size-2.

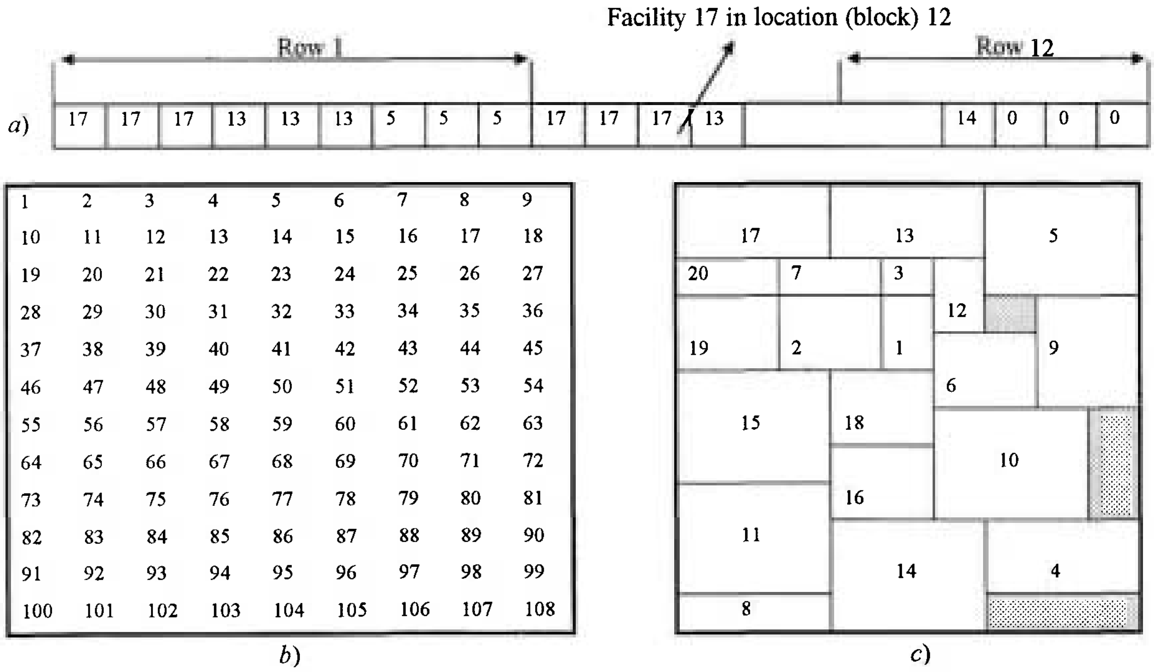


Figure 3. *a*) A valid genetic coding (chromosome) for the 20-facilities problem, *b*) Location (block) configuration, and *c*) A feasible layout solution.

TABLE 1. Computational Result of Three Different Algorithms.

problem	B/B[a]	Hybrid model b	LADEGA	Solution quality (%)c
Baz-12	28158	26684	25738	109.4
Baz-14	16341	14818	14407	113.4

a) Branch-and-bound (B/B) technique [1].

b) Hybrid model by merging simulated annealing and local search methods [2].

c) Note: solution quality shows a performance measure of LADEGA with the B/B method.

In the case of the 14-facilities problem (Baz-14), it is required to assign 14 facilities occupying 61 blocks to an area of 63 blocks. The floor dimensions of 7 (width) by 9 (length) are known a priori. Again, LADEGA generates a significantly improved result with the value of 14,407 for the Baz-14, as shown in Figure 3b, which is still superior to other solutions reported, as shown in Table 1. The above objective value has been reported in Tavakkoli-Moghaddam and Shayan [11]. The GA parameters are population size 5000; number of facilities-14; chromosome length-63; generation number-1000; crossover rate-0.8; mutation rate-0.1; random seed number-0.5; and tournament size-2.

In the case of the 20-facilities problem named Imam-20, a total number of 101 standard blocks are required for all the 20 facilities. Imam and Mir [5] applied an optimization process to the enveloped blocks to improve the initial solution and finally reported the solution of 2,529,89 without considering the floor boundaries. However, the associated layout solution requires 126 equal-sized blocks to assing all the 20 facilities, ie. the floor width and length are 9 and 14 respectively. LADEGA is an improved method which considers the floor boundaries by the user. It should be noted that the dead space ratio (DSR) [13] of the solution reported by Imam and Mir [5] is 0.2, in which reduction of the DSR results in the best space utilization.

This DSR is determined by the total amount of unused areas (dead space) against the total rectangular areas required to fit all the machines within the floor plan (or a rectangular shape) in order to achieve the best space utilization. The aim of the DSR is to minimize the unused space (or slack areas) by forcing the facilities as close together as possible.

LADEGA reports the best objective value of 2,509 as shown in Figure 6.8. Its solution quality is 100.83% compared with the solution reported by Imam and Mir [5]. The associated x and y-center coordinates of 20 facilities are shown in

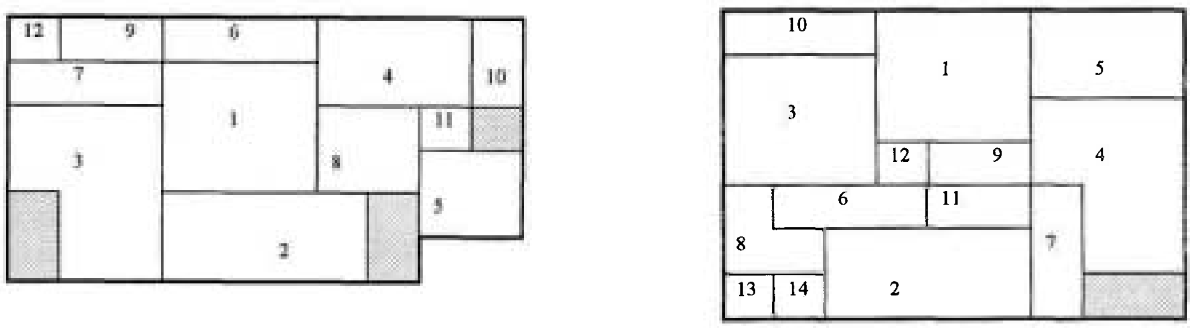
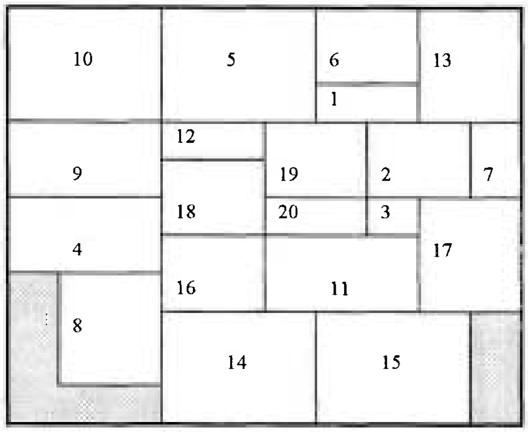


Figure 4. Layout solutions with the best objective value of **a)** 25738 for the Baz-12 case and **b)** 14407 for the Baz-14 case.

Table 2. In addition to the above promising solution, the related layout has the dead space ratio of 0.082, about 40% (2.5times) of the DSR computed from the solution reported by Imam and Mir [5] showing the better space utilization, as summarized in Table 2. It should be noted that the floor dimensions are 11 (width) and 10 (length) for the above solution.

Further investigation were carried out in solving the Imam-20 problem using both the G-QAP and Cont-1 Models. Let the floor width and length be 11 and 10 respectively. To use the G-QAP Model, the distance matrix (110 by 110) must be constructed as input data. The objective value of 3,092.38 is found by LADEGA. The associated layout solution is drawn to find the x and y-coordinates of each facility. Thus the rectilinear distance between each pair of facilities is dynamically measured from the above layout solution. Then the Cont-1 Model is used yielding smaller objective value of 3,018 with a gap of 2.46% obtained in much less computational time. The associated layout solution has the sequence of 11, 2, 14, 8, 17, 18, 16, 15, 9, 20, 12, 4, 19, 7, 1, 13, 3, 10, 6 and 5. This sequence is obtained by scanning and reading the facility numbers given from left to right and top to bottom of the layout configuration.

Due to lack of input data for large-scale layout problems of unequal-sized facilities, two data sets are uniformly generated at random for the dimensions of 30 and 50 facilities. These two test problems are named Tav-30 and Tav-50 by Tavakkol-Moghaddam [9]. Each facility generated consists of a number of standard (equal-sized) blocks. The total number of blocks required for a facility is determined by



Facility	1	2	3	4	5	6	7	8	9	10
X	7.0	8.0	7.5	1.5	4.5	7.0	9.5	2.0	1.5	1.5
y	8.5	7.0	5.5	5.0	9.5	10.0	7.0	2.5	7.0	9.5

Facility	11	12	13	14	15	16	17	18	19	20
X	6.5	4.0	9.0	4.5	7.5	4.0	9.0	4.0	6.0	6.0
y	4.0	7.5	9.5	1.5	1.5	4.0	4.5	6.0	7.0	5.5

Figure 5. Best layout solution, including their x and y-center coordinates, with the objective value of 2509 and the DSR of 0.082 (dimensions of 11 × 10) for the Imam-20.

TABLE 2. Comparison of Computational Result for 20 Unequal-sized Facilities.

Problem	Objective function value			Dead space ratio (DSR)	
	FLOAT[a]	LADEGA	%[b]	FLOAT	LADEGA
Imam-20	2529.89	2509.00	100.83	0.200	0.082

a) Optimization process of an enveloped block [5].

b) Solution quality of LADEGA compared with the FLOAT.

multiplying its width and length. Let the shape of all the facilities be rectangular. The total sum of equal-sized blocks used for all the 30 and 50 facilities are 270 and 443 respectively. These Figures are needed for determining the DSR of a layout solution.

As an initial start, the floor dimensions of $W=20$ (width) by $L=15$ (length) and $W=25$ by $L=20$ are used in LADEGA for the Tav-30 and Tav-50 cases respectively. These dimensions determine the chromosome length by 300 and 500. It should be noted that the value and the position of each gene within the chromosome represent the facility number and the related location respectively.

Three cases are proposed to investigate the performance of the solutions generated by LADEGA, based on how the dimensions of facilities are exchanged. A small number of population and generations, as 100 and 50 respectively, are chosen in each of these three cases in order to test the validity of LADEGA in handing a large chromosome length, depending on the floor dimensions. The objective values are given in Table 3. The solution quality for each case is poor at this stage. In each experiment, the population size 100, generation 50, crossover rate 0.8, mutation rate 0.1, random seed 0.5 and tournament size 2 are used in LADEGA.

Each proposed case uses its own input data as required for the dimensions of the facilities. There is a possibility of proposing another case which randomly selects one dimension as the length and the other one as the width of the facility. This case has not been the subject of an experiment with LADEGA. In all experiments reported in Table 4, the floor dimensions have not been changed for each test problem (Tav-30 or Tav-50).

Further experiments have been carried out when the floor dimensions are only changed by the user. The user can determine the most suitable dimensions of the site in advance. Equally important, it can be used in cases where the total layout area required is less than the available area under cover in a building. Determination of the most suitable space will maximize the utilization. The GA parameters are the same as the one used above. The objective values of 14 test problems reported by LADEGA are given in Tables 4 and 5. The performances of LADEGA on solving Tav-30 and Tav-50 are shown in Figures 5 and 6 respectively. The relevant information of 14 different tests is extracted from the above tables, in which each test represents one specific floor dimension. In the case of Tav-30, the chromosome length must be greater than 270, showing the total standard (equal-sized) blocks required to accommodate all the 30 facilities. This can be done by finding an appropriate width and length of the floor plan in such a way that the product of these dimensions satisfy the above area (ie., greater than 270). The same concept is applicable to Tav-50, where the chromosome length must be greater than 443.

Given Table 4, it is concluded that the best objective values for Tav-30 can be found with a chance of 14.3% in Case 1, 62.3% in Case 2 and 76.6% in Case 3. It is also shown that the best objective values for 50 facilities can be found in Case 1, Case 2 and Case 3 with 7%, 43% and 57% respectively, from the computational results given in Table 5. As an example, the chance of finding the best objective value in Case 1 of Tav-30 is determined as follows: only 2 solutions out of 14 experiments carried out have the best objective values, test 1 and 2. Therefore the chance is 2/14 = 14.3%.

TABLE 3. Computational Results for the Tav-30 and Tav-50 Cases Using Three Cases.

Problem	Case	Width	Length	Objective value	DSR
Tav-30	1	20	15	21119	0.10
	2	20	15	20983	0.10
	3	20	15	20989	0.10
Tav-50	1	25	20	135478	0.11
	2	25	20	133266	0.11
	3	25	20	133454	0.11

Case 1: Original input data for the dimensions of the facilities are used

Case 2: Length of a facility is always larger than its width

Case 3: Original input data for the dimensions of the facilities are exchanged

TABLE 4. Comparison of the Objective Values for Tav-30 in Three Cases, Varying and Exchanging Dimensions of the Floor Plan and their Facilities.

Test no	Floor dimensions		Objective Value			DSR
	Width	Length	Case 1	Case 2	Case 3	
1	29	10	22531	22685	22979	0.0690
2	10	29	23453	23911	23773	0.0690
3	26	11	22553	21399	22117	0.0559
4	11	26	22626	23143	22164	0.0559
5	24	12	21493	21213	22229	0.0625
6	12	24	22651	21817	22411	0.0625
7	22	13	21485	21229	21299	0.0559
8	13	22	22571	21324	21357	0.0559
9	20	14	21539	19733*	20553*	0.0357
10	14	20	20961*	20535	21873	0.0357
11	20	15	22531	20983	20989	0.1000
12	15	20	23051	21362	20653	0.1000
13	19	15	21065	20617	20797	0.0526
14	15	19	21633	20707	20651	0.0526

* Best objective function value in each case.

TABLE 5. Comparison of the Objective Values for Tav-50 in Three Cases, Varying and Exchanging Dimensions of the Floor Plan and their Facilities.

Test no	Floor dimensions		Objective Value			DSR
	Width	Length	Case 1	Case 2	Case 3	
1	31	15	140212	138742	139016	0.0473
2	15	32	144208	144305	148848	0.0770
3	29	16	136380	138982	134320	0.0453
4	16	30	145700	146908	143631	0.0771
5	28	17	137832	133536	133708	0.0693
6	17	28	143566	141004	138508	0.1014
7	26	18	136112	132202*	136790	0.0534
8	18	27	138575	137635	140958	0.0884
9	25	19	136176	134126	137026	0.0674
10	19	26	139196	139816	135446	0.1032
11	25	20	135478	133266	133458	0.1140
12	20	25	137398	137132	135426	0.1140
13	24	20	132700*	134316	127710*	0.0771
14	20	24	137156	135731	132798	0.0771

* = Best value in each case.

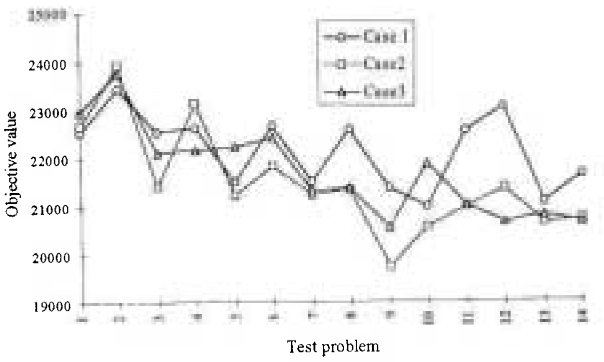


Figure 6. Performance of LADEGA obtained from Table 4 for the Tav-30 case.

To further investigate the performance of LADEGA, experimental cases have incorporated lager population size and a greater number of generations. Table 6 summarizes the results found for the three cases using LADEGA. Figures 8 and 9 show the performance of LADEGA when small and large population sizes are used to solve the 30 and 50 facilities layout problems.

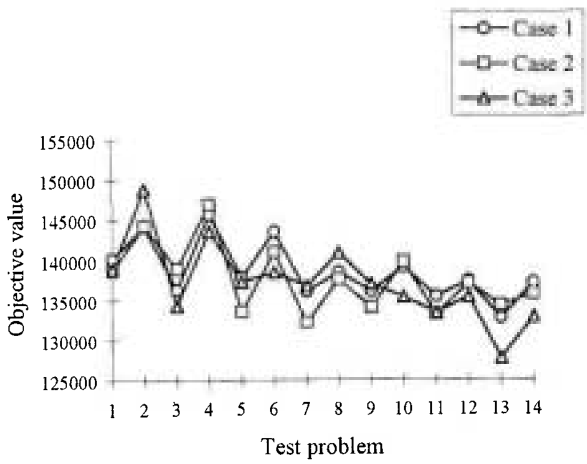


Figure 7. Performance of LADEGA obtained from Table 5 for the Tav-50 case.

For all three cases discussed above, Figure 9 shows the convergence rate of average objective values verses number of generations when the floor dimensions of W=20 and L=15 are used in the Tav-30 case.

TABLE 6. Computational Results for the 30 and 50 Facilities of Unequal-sized Areas.

Problem	Case	Objective function		Floor dimension		Dead space ratio
		Small population[a]	Large population[b]	Width	Length	
Tav-30	1	20961	20319	14	20	0.036
	2	19733	**19509**	20	14	0.036
	3	20553	20173	20	14	0.036
Tav-50	1	132700	128676	23	20	0.037
	2	132202	126170	26	18	0.053
	3	127710	**125246**	24	20	0.077

Case 1: The original input data for the dimensions of the facilities are used.

Case 2: The length of a facility is always larger than its width.

Case 3: The original input data for the dimensions of the facilities are exchanged.

a) : Population size = 100, and b): Population size = 10000.

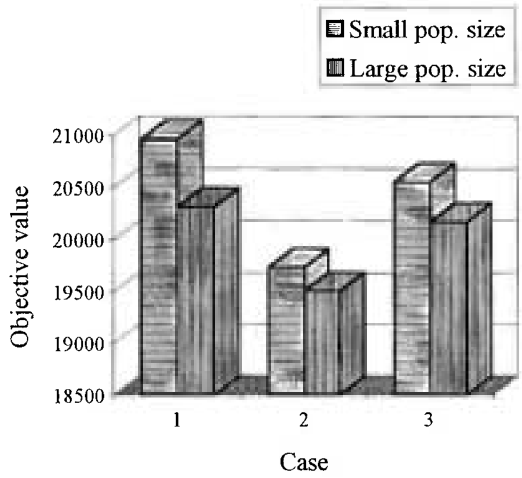


Figure 8. Comparison of computational results between small population size, 100, and large population size, 10000, obtained by LADEGA for the Tav-30 case.

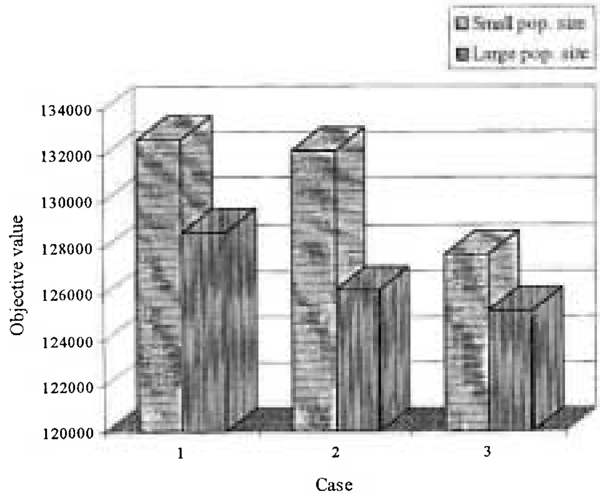


Figure 9. Comparison of computational results between small population size, 100, and large population size, 10000, obtained by LADEGA for the Tav-50 case.

5. CONCLUSION

This paper presented layout solutions generated by LADEGA for unequal-sized facilities layout problems using the generalized QAP and continuous plane models. Three test problems of 12, 14 and 20 unequal-sized facilities have been conducted to test the performance and efficiency of the LADEGA program to compare with the branch-and-bound technique, a hybrid method merging simulated annealing and local search methods, and an optimization process of of enveloped blocks.

In the case of the continuous plane model, the dead space ratio has been taken into consideration as the second objective criterion to maximize the space utilization. The computational experiments showed that

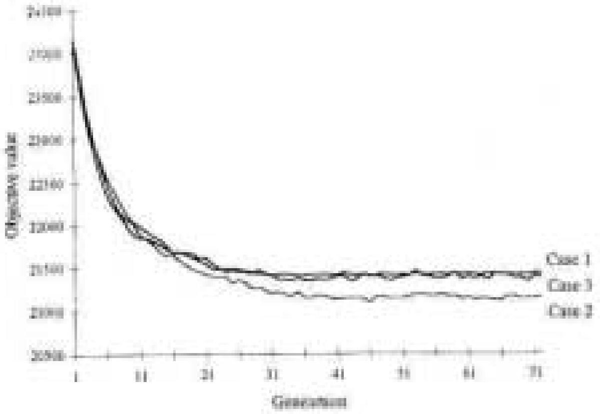


Figure 10. Convergence rate of LADEGA used in the Tav-30 case.

LADEGA generates layout solutions which are better than the solutions reported in the literature so far. A further investigation has been carried out in solving the 20-facilities layout problem by employing the generalized QAP and continuous plane models. The latter model yields lower objective value and less computational time than the former model.

Two data sets of 30 and 50 facilities were generated at random, and three cases with changed dimensions of its facilities were planned and investigated. In the case of solving 30 and 50 facilities, three cases have been proposed and the best solutions have been presented using small and large population sizes.

REFERENCES

1. M. S. Bazaraa, "Computerized Layout Design: A Brach and Bound Approach", ***AIIE Transaction***, 7 (1975) 432-438.
2. M. Bazargan-Lari and H. Kaebernick, "An Efficient Hybrid Method to Solve Equal and Unequal-Size Facilities Layout Problems", ***Int. J. of Ind. Eng. - Applications and Practice***, 3, No. 1 (1996) 51-63.
3. D. E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley Publishing Co., (1989).
4. J. H. Holland, "Adaptation in Natural and Artificial Systems: an Introductory Analysis With Applications to Biology, Control, and Artificial Intelligence", MIT Press, Cambridge, (1975), (2nd edition in 1992).
5. M. H. Imam and M. Mir, "Automated Layout of Facilities of Unequal Areas", Computers and Industrial Eng. 24, No. 3 (1993) 355-366.
6. T. C. Koopmans and M. Beckmann, "Assignment Problems and the Location of Economic Activities", ***Econometrica***, 25 (1957) 53-76.
7. A. Kusiak and S.S. Heragu, "The Facility Layout Problem", ***European J. Oper. Res.***, 29 (1987) 229-251.
8. S. Sahni and T. Gonzalez, "P-Complete Approximation Problems", ***J. of the Association for Computing Machinery***, 23 (1976) 555-565.
9. R. Tavakkoli-Moghaddam, "Design of a Genetic Algorithm to Solve Manufacturing Facilities Layout Problems", PhD Dissertation, Swinburne University of Technology, Melbourne, Australia, (1997).
10. R. Tavakkoli-Moghaddam and E. Shayan, "An Analysis of The Genetic Operators Affecting the Performance of Genetic Algorithms for Facilities Layout Problems", ***Proc. of the 7th Int. Conf. on Manufacturing Eng., Institute of Engineers***, Cairns, Australia (1997a) 51-60.
11. R. Tavakkoli-Moghaddam and E. Shayan, "Facilities Layout design by genetic algorithms", Computers and Industrial Engineering, 35, Nos. 3-4 (1997b) 527-530.
12. R. Tavakkoli-Moghaddam and E. Shayan, "Manufacturing Facilities Design, A State-of-the-Art Survey of Advanced Modeling, ***Proc. of the 2nd Int. Mechanical Engineering Conf***, Society of Mechanical Engineering, Shiraz, Iran (1996) 877-885.
13. P. S. Welgama and P. R. Gibson, "A Construction Algorithm for the Machine Layout Problem With Fixed Pick-up and Drop-Off Points", ***Int. J. Prod. Res.***, 31, No. 11 (1993) 2575-2590.