IIE Transactions

Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713772245

Bi-objective facility expansion and relayout considering monuments

To cite this Article: Kulturel-Konak, Sadan, Smith, Alice E. and Norman, Bryan A. , 'Bi-objective facility expansion and relayout considering monuments', IIE Transactions, 39:7, 747 - 761
To link to this article: DOI: 10.1080/07408170600805943
URL: http://dx.doi.org/10.1080/07408170600805943

IIE Transactions (2007) **39**, 747–761
Copyright © "IIE"
ISSN: 0740-817X print / 1545-8830 online
DOI: 10.1080/07408170600805943

Bi-objective facility expansion and relayout considering monuments

SADAN KULTUREL-KONAK[1,*], ALICE E. SMITH[2] and BRYAN A. NORMAN[3]

[1] *Management Information Systems, Penn State-Berks, Tulpehocken Road, P.O. Box 7009, Reading, PA 19610, USA*
E-mail: sadan@psu.edu
[2] *Department of Industrial and Systems Engineering, Auburn University, 206 Dunstan Hall, Auburn University, AL 36849-5346, USA*
E-mail: aesmith@eng.auburn.edu
[3] *Department of Industrial Engineering, University of Pittsburgh, 1048 Benedum Hall, Pittsburgh, PA 15261, USA*
E-mail: banorman@pitt.edu

Received December 2002 and accepted December 2005

In this paper, the unequal area facility expansion and relayout problem is studied. The facility relayout problem is important since both manufacturing and service entities must modify their layouts over time when their operational characteristics change. A bi-objective approach is proposed to solve the relayout problem for cases of both a fixed facility area and an expanded facility area. Material handling costs and relayout costs are minimized using a tabu search meta-heuristic optimizer. This heuristic randomly alternates the objective function between the two objectives of the problem in each step and, by doing so, eliminates the difficulty of weighting and scaling the two objectives. The approach is flexible in handling various aspects of the problem such as stationary portions of departments (i.e., monuments), addition of new departments, and changes in existing department and facility areas. Computational experiments show that the bi-objective tabu search approach is effective and tractable. The use of the Pareto front of designs is demonstrated by showing a few approaches to analyzing the trade-offs between initial costs (relayout cost) and ongoing expenses (material handling costs).

Keywords: Bi-objective unequal-area facility re-layout problem, facility expansion, monuments, fixed departments, Pareto front, tabu search

1. Introduction

Most of the literature on facility layout focuses on greenfield design, which is the design of a new facility without the influence or constraint of an existing facility. In practice, however, the Facility Relayout Problem (FRLP) is more common than a greenfield design since both service and manufacturing industries operate in highly volatile environments that motivate redesign of their layouts. Although the greenfield design problem and the FRLP have common characteristics, the FRLP requires additional constraints and objectives. This paper considers several of the most significant aspects of the FRLP, namely fixed portions of departments, departmental area changes, expansion of the bounding facility and the two primary cost objectives: (i) relayout costs; and (ii) material handling costs.

In the FRLP, it may be impossible to relocate certain pieces of equipment and machinery such as large presses, furnaces, and painting units. In the literature, such stationary equipment is usually assumed to consume an entire department at a corner of the facility (Bozer *et al.*, 1994; Meller and Gau, 1996; Meller and Bozer, 1997). In the area of dynamic (i.e., multi-period) Facility Layout Problems (FLPs), Kouvelis *et al.* (1992) assumed that a fixed piece of equipment (i.e., a fixed department) was not allowed to change its location during the planning periods. In this paper, a fixed piece of equipment is called a "monument", and a flexible approach is used to handle departments with monuments so that only the monument portion of the department remains stationary. Thus, if a department contains a large, difficult to move machine but also contains several smaller machines or workbenches that can be readily moved, the proposed method only requires that the large machine be kept at its current location. The rest of the department can be moved as long as the new department location still contains the large machine. In addition, monuments are not limited to a certain location within the facility such as corners or edges of the layout.

*Corresponding author

Facility relayout is usually considered because of changes in the material flow pattern or modification of departments, such as the addition of new departments or changes in the sizes of the existing ones. Therefore, the FRLP can be modeled in two distinct categories: (i) relayout within an existing area; and (ii) relayout with expansion. The approach in this paper can readily accommodate either with a few simple assumptions related to the format of the expansion.

The solution technique is a Tabu Search (TS) that has been modified to consider two objectives. Specifically, both material handling costs and relayout costs are minimized, and the TS procedure finds the Pareto front (i.e., the set of non-dominated solutions) with respect to both objectives. The problem is formulated as an unequal-area layout problem with rectangular departments within a rectangular facility; this alone is a substantial relaxation over most of the previous relayout papers, which assumed interchangeable departments (i.e., a quadratic assignment problem (QAP) formulation).

2. Relevant literature

2.1. *The dynamic facility layout problem*

The Dynamic FLP (DFLP) considers flows over multiple time periods in an environment in which material flow between departments changes over time. The FRLP is a special case of the DFLP with two restrictions: (i) the first time period is the present and has a specified layout; and (ii) there are only two time periods (a similar idea is postulated by Lacksonen (1994)). A comprehensive survey on DFLP algorithms can be found in Balakrishnan and Cheng (1998). Rosenblatt (1986) first dealt with the dynamic nature of the FLP by assuming a deterministic environment in which product demands were known for each period. The major goal was to decide the layout for each period given the from-to flow matrices for each period. Following this original paper other techniques have been developed to solve the problem either optimally or heuristically. The optimal solution methods proposed by Montreuil and Venkatadri (1991), Balakrishnan *et al*. (1992), Montreuil and Laforge (1992), Urban (1992, 1998), and Lacksonen and Enscore (1993) are only able to solve relatively small instances. The heuristic approaches by Urban (1993), Conway and Venkataramanan (1994), Kaku and Mazzola (1997), Yang and Peters (1998), Kochhar and Heragu (1999), Balakrishnan and Cheng (2000), Balakrishnan *et al*. (2000), and Baykasoglu and Gindy (2001) include a steepest-descent pairwise-interchange procedure, a construction procedure, and meta-heuristics, such as Genetic Algorithms (GAs), TS, and simulated annealing.

2.2. *Tabu search*

Tabu search is a meta-heuristic optimization method with its current form being suggested by Glover (1989, 1990). TS is an effective heuristic method that has been used to solve combinatorial optimization problems with large and complex search spaces in the areas of scheduling, telecommunications, transportation, routing, network design, graph theory, manufacturing, financial analysis, and constraint satisfaction. Comprehensive material about TS can be found in Glover *et al*. (1993) and Glover and Laguna (1997). TS is used to solve the FLP problem because of the existence of non-linearities in the objective function and/or the constraint set, the strong and consistent neighborhood structure of the block layout and the necessity to use a global, rather than local, optimizer. TS guides a local (neighborhood) search procedure to find a global or near-global optimum. Therefore, TS with its local search property is an appropriate optimization tool to search neighborhoods efficiently. Kaku and Mazzola (1997) used a TS approach to solve the DFLP. Their TS heuristic is a two-stage procedure. In the first stage, a number of different starting solutions are generated using a diversification strategy to ensure that different regions of the search space are explored. Then, a modified TS procedure is applied to each solution, and the starting solutions providing the best solutions are selected for use in the second stage. In the second stage, neighborhoods around the best solutions found in the first stage are searched more intensively. At the end of the second stage, the best solution obtained for the problem is the final solution found by the TS heuristic. They used an objective function that summed the material handling costs and costs of moving the departments. They did not consider altering either the departmental or facility areas, nor did they consider fixed portions (monuments) in the redesign. Chiang (2001) provides a Windows® based (i.e., implemented using Microsoft Visual Basic) visual facility design system that uses TS as the search engine. The purpose of embedding a graphical user interface with a search engine is to provide layout designers with a user-friendly environment to facilitate user interaction and thereby improve the design process. Since locations and shapes of the departments are assumed to be more flexible than in a classic QAP formulation, two type of moves, namely interchange and move (i.e., insertion) transitions, are considered to better cover the search space. (For more information about the neighborhood definitions, see Chiang (2001)). The approach requires a long computational time because of the complex nature of the problem and the low compilation rate of Visual Basic. However, with the advent of faster computers in recent years, it should be possible to use this embedded algorithm to solve large problems in a reasonable time.

The relayout problem involves two objectives: (i) minimizing material handling costs; and (ii) minimizing the relayout costs. These are difficult to scale and weight relative to one another. One is an ongoing, operational cost whereas the other is an initial capital cost. Identifying the Pareto front (that is, the set of non-dominated designs) is of value to the decision maker in this case. Therefore, the technique of alternating objective functions which was devised and studied by Kulturel-Konak *et al.* (2006) was used.

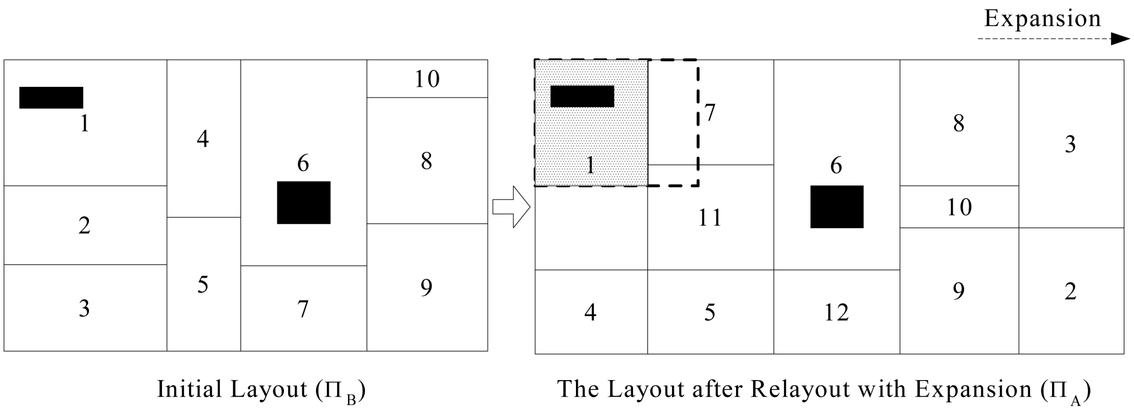


Fig. 1. Overlap of the locations of department 1 after relayout.

3. Problem statement

The FRLP studied in this paper is defined as follows. Given an existing layout (Π_B: the partition "Before" relayout) and a new flow matrix, a new layout (Π_A: the partition "After" relayout) will be devised to minimize material handling costs and relayout costs and to accommodate the addition of new departments and/or modify the areas and locations of existing departments. For instance, in Fig. 1, for a ten-department problem, the current layout and the modified layout after rearranging the locations of the existing departments and adding two new departments (departments 11 and 12) are given. In this example, to accommodate the additional space requirements for the new departments, the building is expanded one way in the horizontal direction. Some pieces of equipment are assumed to be fixed, and these are called monuments. Since the monuments cannot be relocated, the departments with monuments can only be partially moved. In Fig. 1, the monument portions are shown with black solid rectangles in departments 1 and 6. As seen in Π_A, although department 1 has been slightly repositioned, it still retains its monument in the original location. (The dashed rectangle in Π_A shows the location of department 1 before relayout.)

A comprehensive general model for the FRLP is given in Montreuil and Laforge (1992). The FRLP studied in this paper is as follows. Given a maximum allowable aspect ratio for each department and the locations of monuments (i.e., the departments with monuments can only be moved such that they still contain the locations of their monuments), the objective is to minimize both material handling costs and relayout costs:

min (the material handling cost (Equation (1))
and relayout cost ($\sum_{i=1}^{N}$ Equation (7))),
subject to
maximum aspect ratio of the departments (Equation (8)),
monuments (Equations (10)–(13)).

3.1. *Material handling costs*

The material handling costs are calculated as:

$$MHC(\Pi) = \sum_{i=1}^{N} \sum_{j=i+1}^{N} f_{ij} c_{ij} d_{ij}(\Pi), \tag{1}$$

where f_{ij} is the flow volume between departments i and j, c_{ij} is the unit handling and transport cost between departments i and j, N is the total number of departments and $d_{ij}(\Pi)$ is the distance (using a pre-specified metric) between departments i and j in partition Π. Since the whole region is partitioned into departments, a specific layout is called "partition Π".

3.2. *Relayout cost*

As stated in Section 1, departments as well as the entire facility are assumed to be rectangular. It is also assumed that the cost of relocating a department is a function of the percent overlap between the old and new locations of the department. If the new and old locations do not overlap, then a fixed cost is used regardless of how far the new location is from the old one. The rationale underlying this assumption derives from the following consideration. Once the decision is made to move a machine or fixture, the distance that it will be moved can be ignored since the majority of the cost comes from disassembling and re-assembling the machine and its material handling system. One can think of the overlap metric as a surrogate for the number of machines (or fixtures) moved, where departmental area is assumed to be correlated with the number of items to move.

To determine the overlap between the new and old locations of a department, the coordinates of the upper-right and lower-left corners of a department are used. The following notation is used to represent the location of the departments.

$(x_i^{\mathrm{BL}}, y_i^{\mathrm{BL}})$ = coordinates of the lower-left corner of department i before relayout;
$(x_i^{\mathrm{BU}}, y_i^{\mathrm{BU}})$ = coordinates of the upper-right corner of department i before relayout;

(x_i^{AL}, y_i^{AL}) = coordinates of the lower-left corner of department i after relayout;
(x_i^{AU}, y_i^{AU}) = coordinates of the upper-right corner of department i after relayout;
(x_i^{OL}, y_i^{OL}) = coordinates of the lower-left corner of overlap area of department i;
(x_i^{OU}, y_i^{OU}) = coordinates of the upper-right corner of overlap area of department i.

Given the locations of department i before and after relayout, the corner coordinates of the overlap area can be found using the following equations:

$$x_i^{OL} = \max\{x_i^{BL}, x_i^{AL}\}, \tag{2}$$
$$y_i^{OL} = \max\{y_i^{BL}, y_i^{AL}\}, \tag{3}$$
$$x_i^{OU} = \min\{x_i^{BU}, x_i^{AU}\}, \tag{4}$$
$$y_i^{OU} = \min\{y_i^{BU}, y_i^{AU}\}. \tag{5}$$

Then, the area of the overlap corresponding to department i, O_i, is calculated as follows:

$$O_i = \left(x_i^{OU} - x_i^{OL}\right) \times \left(y_i^{OU} - y_i^{OL}\right). \tag{6}$$

Then, the relayout cost of department i, RC_i, is given as:

$$RC_i = rc_i \times (Area_i - O_i), \tag{7}$$

where $Area_i$ is the area of department i before relayout and rc_i is the fixed relayout cost of department i per unit square area moved. In Equation (7), if a department is moved and the new and old locations of the department overlap 80% or more, it is assumed that 20% of the total RC is incurred. In Fig. 1, Π_A shows the location of department 1 before (dashed lines) and after (solid lines) relayout. The overlap area, O_i, is shown with shading.

3.3. *Infeasibilities*

The proposed methodology considers two types of infeasibility: (i) department shape infeasibility; and (ii) monument infeasibility. Each of these is described in more detail below.

3.3.1. *Shape infeasibility*

The aspect ratio for department i is used to constrain its shape and is defined as $\alpha_i = \max\{l_i, w_i\}/\min\{l_i, w_i\}$, where l_i and w_i are the length and width of department i, respectively. The smaller the aspect ratio, the more constrained the problem. For each department i, the following inequality must hold:

$$\alpha_i \le \alpha. \tag{8}$$

Therefore, the degree of shape feasibility violation of department i can be calculated as follows:

$$\max\{\alpha_i - \alpha, 0\}, \tag{9}$$

where α is the pre-defined maximum allowable aspect ratio.

3.3.2. *Monument infeasibility*

A monument, which is assumed to be rectangular, is represented by the coordinates of its upper-right and lower-left corners, (x_i^{FU}, y_i^{FU}) and (x_i^{FL}, y_i^{FL}), respectively. A layout is feasible with respect to the departments with monuments if each of those departments includes its corresponding monument. In other words, the following four conditions must be satisfied for a department i with a monument for this department to be *monument feasible*:

$$x_i^{AL} \le x_i^{FU} \le x_i^{AU}, \tag{10}$$
$$x_i^{AL} \le x_i^{FL} \le x_i^{AU}, \tag{11}$$
$$y_i^{AL} \le y_i^{FU} \le y_i^{AU}, \tag{12}$$
$$y_i^{AL} \le y_i^{FL} \le y_i^{AU}, \tag{13}$$

The degree of monument feasibility violation of a department i with a monument, θ_i, can be calculated as follows:

$$\theta_i = \max\{x_i^{AL} - x_i^{FL}, x_i^{FU} - x_i^{AU}, 0\} + \max\{y_i^{AL} - y_i^{FL}, y_i^{FU} - y_i^{AU}, 0\} \tag{14}$$

Equation (14) calculates the minimal amount that department i must be moved in the x and y directions to include the corresponding monument. Figure 2 shows an example of monument infeasibility.

4. The Bi-objective TS algorithm

The layout is represented by using the flexible bay construct of Tate and Smith (1995) with a variable length string encoding, which concatenates a permutation of the department order and the bay break position. A boustrophedon ordering of the departments is used. (Boustrophedon is a reading/writing style that alternates direction every line.) The flexible bay structure is a continuous layout representation allowing the departments to be located only in parallel bays

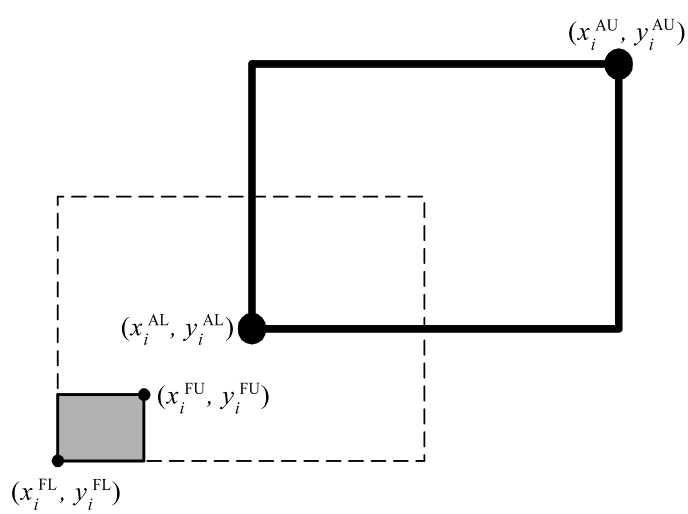


Fig. 2. Monument infeasibility.

with varying widths. The width of a bay depends on the total area of the departments in the bay. A bay is bounded by straight aisles on both sides, and departments are not allowed to span over multiple bays. Meller (1997, p. 229) mentioned that "In many manufacturing facilities, the facility layout is based on a parallel bay design, where material moves primarily within the bays and inter-bay movement is handled via an inter-bay material handling system at one (or both) end(s) of the bays". Examples of such facilities arise in the context of heavy manufacturing; e.g., steel production, bridge crane manufacturing, and in retail environments.

Since an initial layout exists in the FRLP, the bi-objective TS algorithm starts with this initial layout rather than starting with a random layout. If new departments are to be added to the initial layout, a new bay is added as the last bay, and the new departments are placed in this bay. The algorithm uniformly and randomly uses either material handling cost (*MHC*) or relayout cost (*RC*) as the objective to select the best candidate solution in each move, that is, with $p = 0.5$. Instead of a single best solution, all non-dominated solutions are kept on the non-dominated solutions list, which is updated each time a new solution is found. Two objective functions are used. The first is based on the *MHC* and the second is based on the *RC*:

$$F = MHC(\Pi_A) + (MHC_{\text{ND\_worst}} - MHC_{\text{all}}) \times \left(\left(\frac{\sum_{i=1}^{n} \max(0,\ \alpha_i - \alpha)}{\text{NFT}_\alpha} \right)^k + \left(\frac{\sum_{i=1}^{n} \theta_i}{\text{NFT}_\theta} \right)^k \right), \quad (15)$$

$$F = RC(\Pi_A,\ \Pi_B) + (RC_{\text{ND\_worst}} - RC_{\text{all}}) \times \left(\left(\frac{\sum_{i=1}^{n} \max(0,\ \alpha_i - \alpha)}{\text{NFT}_\alpha} \right)^k + \left(\frac{\sum_{i=1}^{n} \theta_i}{\text{NFT}_\theta} \right)^k \right). \quad (16)$$

In Equation (15), $MHC(\Pi_A)$ is the material handling cost of layout Π_A. In Equation (16), $RC(\Pi_B, \Pi_A)$ is the relayout cost of changing from layout Π_B to layout Π_A. The last part of each objective function corresponds to a penalty function for infeasibility. The penalty function has two parts: one for shape infeasibility and one for monument infeasibility. In this paper, the penalty functions use the notion of Near-Feasibility Threshold (NFT). NFT was first defined by Smith and Tate (1993) and then improved by Coit *et al.* (1996) in their work on penalty functions for GAs and Kulturel-Konak *et al.* (2004) for TS. Solutions are penalized according to their distance from feasibility as follows. Within the NFT region, infeasible solutions are penalized relatively lightly whereas beyond the NFT region, a heavy penalty is applied. The NFT for shape infeasibility (NFT_α) and for monument infeasibility (NFT_θ) are updated according to the feasibility ratio of the recent moves kept in the tabu list as described in Kulturel-Konak *et al.* (2004). The NFT is dynamically updated based on the recent search history stored in the tabu list and the state of the current move as follows. If most of the recent moves have been feasible, which indicates that the search is either in the feasible region or close to it, the NFT value is increased and a high value of the NFT decreases the penalty applied. Therefore, exploration of infeasible solutions is encouraged and as a result, the search is directed towards the boundary of feasible and infeasible solutions with the expectation that optimal solutions reside near the boundary of the feasible region. If most of the recent moves have been infeasible, the search is either in the infeasible region or close to it. Therefore, the NFT value is decreased to increase the penalty applied. As a result, feasible solutions and infeasible solutions with lower levels of infeasibility become more attractive, which moves the search towards the feasible region.

When updating NFT_α, only shape infeasibility is considered. Likewise, for NFT_θ, only monument infeasibility is used. The exponent, k, is a user-defined severity parameter amplifying the behavior of the ratio in parenthesis, and it is set to two, which results in a Euclidean distance metric between the infeasible solution and feasibility. The method is not sensitive to the value of this exponent, however. The definitions of the terms in the multiplier of each penalty function are as follows:

$MHC_{\text{ND\_worst}}$ = the worst *MHC* in the non-dominated solutions list;
MHC_{all} = the best *MHC* found so far;
$RC_{\text{ND\_worst}}$ = the worst *RC* in the non-dominated solutions list;
RC_{all} = the best *RC* found so far.

A swap move is used to produce candidate solutions. The swap operator exchanges the positions of department pair $[i, j]$ in the department permutation array for $i = 1, \ldots, N$ and $j = (i + 1), \ldots, N$. The best permutation after exhaustively applying the swap move is chosen, and the swap move is applied to all department pairs regardless of whether or not a department has a monument. It is impossible to visit all possible solutions of the problem using only the swap move without altering the bay structure of a solution. Therefore, bay break moves are considered after the swap operator is complete. In a bay move, combinations of bay break positions for the same, one less and one more bay breaks of the best candidate swap move are investigated. If the best candidate is infeasible, all possible bay break combinations are considered. However, if the best candidate is feasible in terms of both aspect ratio and monuments, only feasible bay break combinations are considered.

The tabu list includes the pair of departments swapped to produce the best candidate solution and the bay structure of the related solution. Therefore, a department pair is not allowed to be swapped while it is on the tabu list unless the bay structure is different. A dynamic length tabu list is used, which varies every 20 iterations according to a uniform distribution ranging between eight and 15. The stopping criterion is defined as the maximum number of iterations conducted without updating the list of non-dominated solutions and is set to 1000. The TS reported here is not sensitive to either the exact tabu list size or the termination criterion.

A diversification scheme based on restart is used. If the list of non-dominated solutions has not been updated in the last (stopping criterion/4) moves, one of the non-dominated solutions found during the search is randomly selected as the new current solution, the tabu list is reset to empty, and the search restarts from this solution. Experiments show that this diversification scheme improves performance by exploring a wide Pareto front. Pareto fronts found without diversification were too narrow.

The detailed steps of the algorithm are described below.

Step 0. Read the existing layout as the initial layout and assign it to the current solution. Initialize the tabu list and the non-dominated solutions list to empty. If the current solution is feasible, add it to the non-dominated solutions list.

Step 1. Randomly choose either Equation (15) or Equation (16) as the fitness function to evaluate candidate solutions.

Step 2. Search the neighborhood of all possible *swap* moves for the department permutation of the current solution. Compare each feasible candidate solution with the current non-dominated solutions list using *MHC* and *RC* as follows. If a candidate solution dominates some current non-dominated solutions, remove these dominated solutions from the non-dominated solutions list and add the candidate to the non-dominated solutions list. If a candidate solution is not dominated by any current non-dominated solution, simply add this candidate solution to the non-dominated solutions list. Choose the non-tabu (or if it is tabu, but it dominates any solution in the non-dominated solutions list) candidate solution with the best objective as the best candidate solution.

Step 3. For the departmental sequence of the best candidate solution chosen in Step 2, investigate all possible bay break positions using the same, one less, and one more bay breaks of the best candidate. (If the best candidate solution is feasible, consider only feasible arrangements.) Compare candidate solutions to the best candidate solution using the fitness function selected in Step 1 and update the best candidate solution if needed. In addition, each time a feasible candidate solution is found, compare it with the current non-dominated solutions list and update this list as defined in Step 2. To increase efficiency, Step 3 is performed once in every ten swap moves.

Step 4. (*Diversification*) If the list of the non-dominated solutions has not been updated in the last (stopping criterion/4) moves, randomly choose one of the current non-dominated solutions and assign it to the current solution, reset the tabu list to empty and return to Step 1.

Step 5. Enter the solution selected by Steps 2 and 3 on the tabu list and set the current solution equal to the best candidate solution. Check the stopping criterion and if it is not satisfied, return to Step 1.

5. Computational results

Three well-known test problems: (i) the ten-department problem of Van Camp *et al.* (1991) (that will simply be referred to as the Van Camp problem); (ii) the 14-department problem of Bazaraa (1975); and (iii) the 20-department problem of Armour and Buffa (1963) were used to develop relayout problems. These problems divide a planar region into ten, 14, and 20 unequal-sized departments, respectively. The total facility area and the areas of the departments were taken directly from the published problems. A set of products with their routings among departments and production volumes were defined. Additional test problem data along with the original problem data are given in Appendix A. In the Van Camp and Bazaraa problems a new product was introduced, and in the Armour-Buffa problem production was stopped on one current product, and in all three problems the routings and the volumes of the existing products were changed. Tables A1–A7 for the Van Camp problem, Tables A8–A13 for the Bazaraa problem, and A14–A17 for the Armour-Buffa problem give the input data. The FRLP with expanded area includes the addition of three new departments to the Van Camp problem and two new departments into the Bazaraa problem. Department 9 in the Van Camp problem, departments 3 and 4 in the Bazaraa problem, and departments 5, 19, and 20 in the Armour-Buffa problem were chosen to include monuments, and the coordinates of the lower-left and the upper-right corners of the monuments are given in Tables A7, A13, and A17, respectively.

For the Van Camp problem, rc_i was taken to be \$1, for the Bazaraa problem \$20, and for the Armour-Buffa problem \$250. If a department includes a monument, then these unit costs are doubled since relocating the movable portions of a department with a monument is more difficult than relocating a department without a monument. The problems were solved by considering positively correlated product demands, since this case is more realistic, and the effect of the variance on the *MHC* is easy to see. The algorithm was coded in C and run using an Intel Pentium IV with a 2.2 GHz processor and 1 GB RAM. To gauge variability, ten different initial random seeds were used. It was observed that the Pareto fronts found are almost identical.

5.1. *Problems using the existing facility area–Without and with department area changes*

First, the problems were solved with new demand flows and using the existing departmental and whole facility areas.

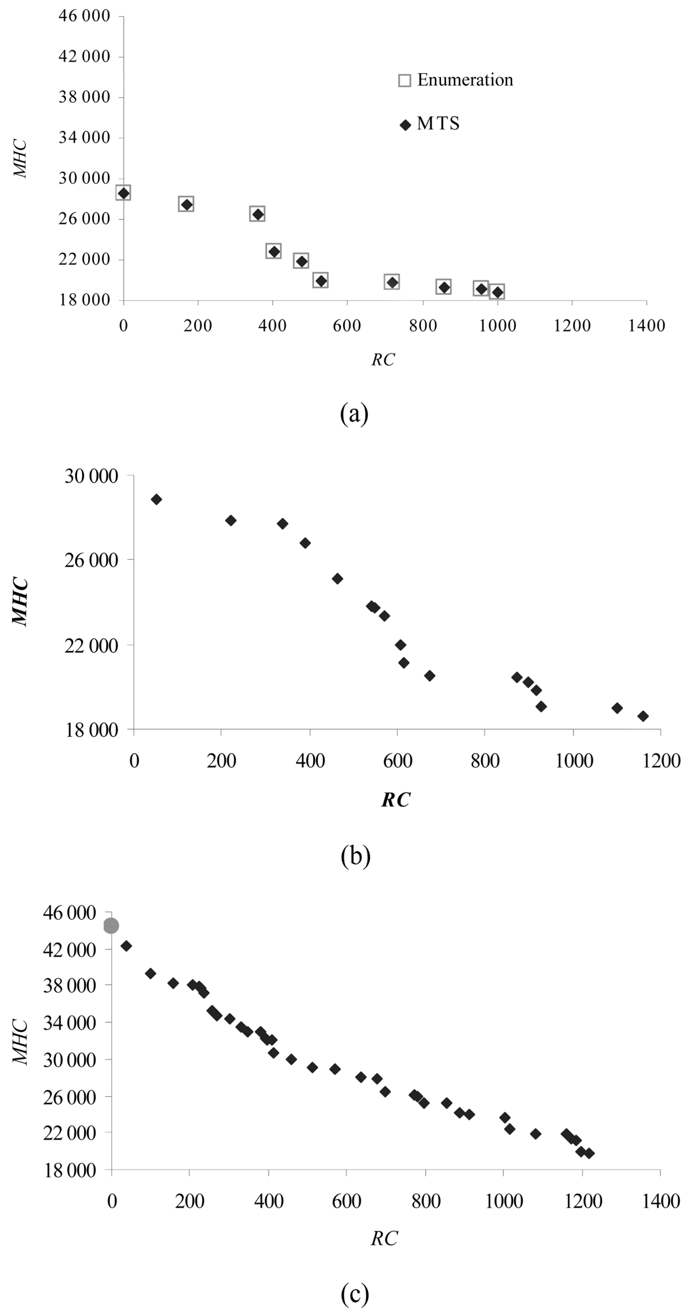


Fig. 3. Pareto front of the Van Camp problem: (a) with the existing facility area and no departmental area change (TS finds all Pareto optimal solutions); (b) with the existing facility area and some departmental area changes; and (c) with expansion.

Figures 3(a), 4, and 5, respectively, show the Pareto fronts for the Van Camp, Bazaraa, and Armour-Buffa problems. The non-convex Pareto front shape of Figs. 3(a) and 5 can be explained by the discrete structure and non-linear nature of the objectives of the problem. Rosenblatt and Sinuany-Stern (1986) study a discrete efficient frontier approach to the FLP. For an objective function of the type $(w_1 f_1 + w_2 f_2)$, where f_1 and f_2 represent two objectives and w_1 and w_2 their corresponding weights. However, this approach cannot be adopted here since the problem under study does not consider a weighted sum of the objectives. In the figures, the *MHC* of the initial (existing) layout (i.e., *RC* equal to zero) is shown with a large circle. The CPU time is 2.90 seconds for a single replication for Fig. 3(a), 5.82 seconds for Fig. 4, and 187.24 seconds for Fig. 5. Six diversifications were performed for the Van Camp problem, three for the Bazaraa problem, and seven for the Armour-Buffa problem. It is easy to notice that solutions can have a similar *MHC* while their *RC* is different. This can be explained by the assumption that a minimum of 20% of the total relayout cost is required even if a department changes its location less than this percentage. For both problems, the algorithm finds non-dominated solutions spread over a wide Pareto front. For the Bazaraa problem, since an aspect ratio of two is very constrained, the Pareto front consists of fewer solutions but is still spread over a wide range. It is clear that the layout with a zero relayout cost is the solution that corresponds to the left-end of the Pareto front. To test whether solutions at the right-end of the Pareto front (high *RC* and low *MHC*) represent the true right-end of the Pareto front, both problems are solved by defining a zero relayout cost for each department (i.e., as a single-objective problem to minimize *MHC*). In both instances, the single-objective TS

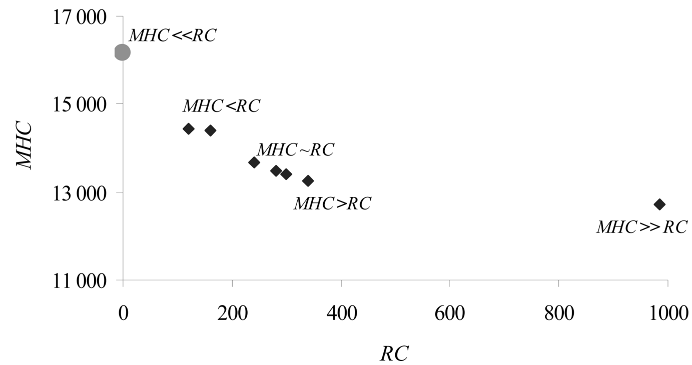


Fig. 4. Pareto front of the Bazaraa problem with the existing facility area and no departmental area change.

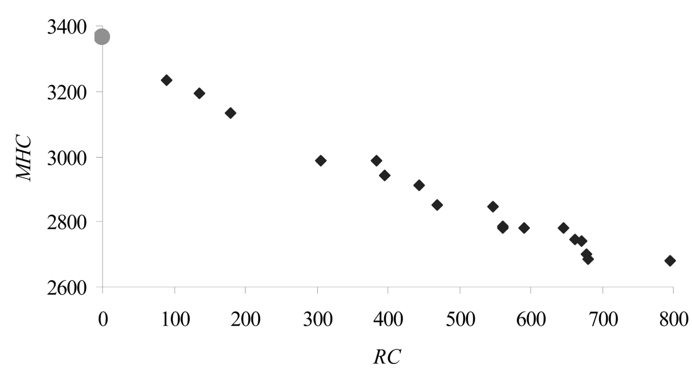


Fig. 5. Pareto front of the Armour-Buffa problem with the existing facility area and no departmental area change.

approach could not find solutions with a *MHC* lower than the solutions found by the bi-objective TS approach. The Pareto set is given in ascending order of *RC* in Tables B1 for the Van Camp B2 for the Bazaraa, and B3 for the Armour-Buffa problems in Appendix B.

Although all three problems are very constrained and it is hard to find feasible solutions, the algorithm performs well in locating departments without violating shape and monument feasibility. Even if departments partially changed their locations and shapes, they retained their monuments at the correct location. An exhaustive enumeration algorithm was coded to generate all possible department permutations and all possible bay break positions for a given permutation. This algorithm was used to find the theoretical frontier of the non-dominated solutions for the Van Camp problem. As seen in Fig. 3(a), TS was able to find all theoretical non-dominated solutions. It took slightly longer than 36 hours to obtain these solutions using exhaustive enumeration whereas the TS ran in only a few seconds.

Then, the Van Camp problem was solved with new demand flows and area changes of some departments using the existing facility area. (The area changes are given in Table A3.) Figure 3(b) shows the Pareto front for this case. There is no zero relayout cost design in this case since there are departmental area changes. Similar interpretations as those for Fig. 3(a) can be made about the results displayed in Fig. 3(b). The CPU time is 2.74 seconds for a single replication, and five diversifications was performed. The Pareto set is given in ascending order of *RC* in Table B4 in Appendix B. It is important to note that the method can handle departmental area changes without any change in the algorithm.

5.2. *Problems using an expanded facility area*

Then the Van Camp and Bazaraa problems with new demand flows and additional departments in an expanded area were considered. Figures 3(c) and 6 depict the Pareto fronts of both problems. The CPU time is 40.96 seconds for a single replication for Fig. 3(c) and 49.00 seconds for Fig. 6. In Figs. 3(c) and 6, the *MHC* of the initial (existing) layout (i.e., *RC* equal to zero) is shown with a large circle.

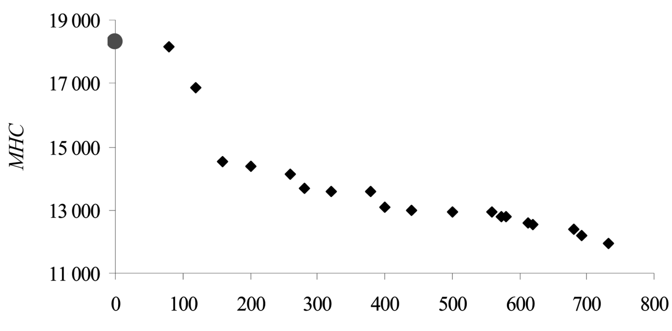


Fig. 6. Pareto front of the Bazaraa problem with expansion.

Seven diversifications were performed for the expanded Van Camp problem and eight for the expanded Bazaraa problem. Compared to Figs. 3(a) and 4, more non-dominated solutions distributed over a wider Pareto front are found. The reason for finding more solutions is the addition of new departments and introducing new interdepartmental flow relations, which enlarge the search space. Therefore, as expected, the larger search space results in more non-dominated solutions for these problems. The Pareto set is given in ascending order of *RC* in Tables B5 and B6 in Appendix B for the Van Camp and the Bazaraa problems, respectively. As with departmental area changes, the method can accommodate facility area changes without changing the underlying algorithm.

5.3. *Comparison with mixed integer non-linear programming*

The effectiveness of the TS approach was analyzed by attempting to solve the Bazaraa and Armour-Buffa problems using GAMS to solve the Mixed-Integer non-linear programming problems that result from modeling these relayout problems. GAMS was not able to improve the efficient frontier solutions found by the TS. In all cases GAMS terminated in 24 to 36 hours having exhausted the computer memory due to the size of the resulting branch-and-bound tree. This approach was also tested by seeding GAMS with efficient frontier solutions found by TS, however, GAMS was not able to improve them. While these experiments do not prove that the TS-identified Pareto fronts are truly optimal, they do provide additional evidence that the optimization by TS is: (i) necessary as the model cannot be solved exactly for larger problems; and (ii) effective in finding optimal or near-optimal solutions.

5.4. *Design decision making*

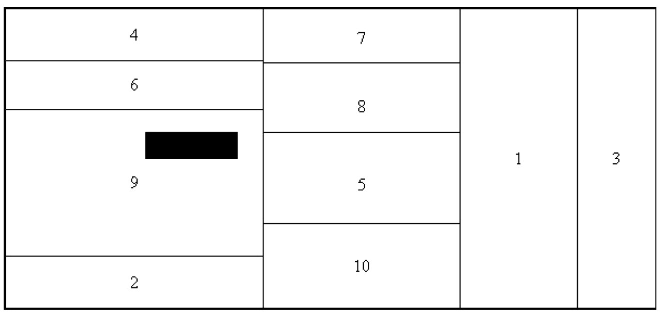
Most of the current research on multiobjective metaheuristics approaches as well as the TS concentrates on generating non-dominated solutions. In practice, the (human) decision-maker will need to select a single solution. This can be done using a multicriteria decision making approach (Keeney and Raiffa, 1993; Horn, 1997; Coello, 2000; Cvetkovic and Parmee, 2002). As an example, a preference approach similar to the one defined in Cvetkovic and Parmee (2002), which is adapted from the linguistic ranking methods of Chen *et al.* (1992, p. 265), is shown in Table 1. The Pareto front of the Bazaraa problem using these relationships between the two objectives is marked in Fig. 4.

As another example, multiple and very different layouts can be in the Pareto set depending on the decision makers' preferences. Three layouts (the first one being the initial layout) from the Pareto set of the Van Camp problem are shown in Fig. 7(a–c). If the decision maker is willing to pay the upfront costs of relayout, the layout in Fig. 7(c) would

Table 1. The preference approach

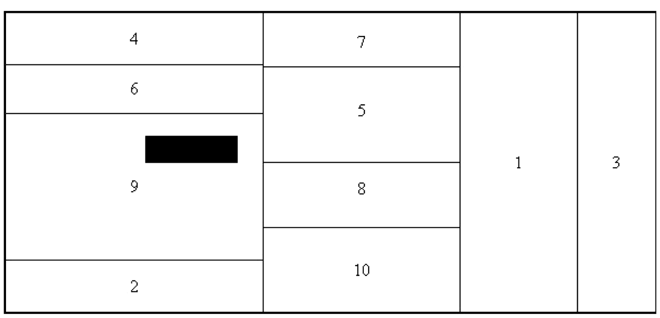
Relation	*Intended meaning*
≫	Much more important
>	More important
∼	Equally important
<	Less important
≪	Much less important

be chosen as it has the lower *MHC*. However, if the initial capital cost is the main issue, the layout in Fig. 7(a) or that in Fig. 7(b) would be preferred with a much lower *RC* but higher *MHC* over the operating life of the facility.

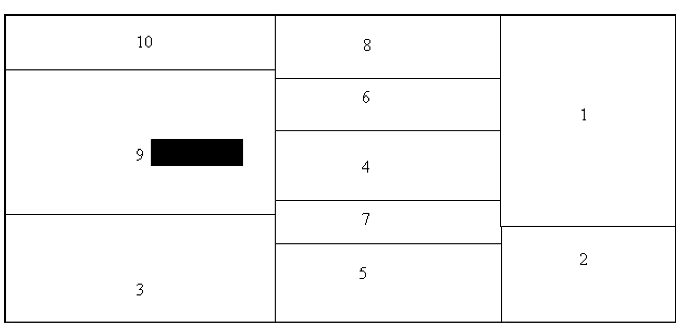


(a)

Initial VanCamp problem layout
Encoding: 4 6 9 2 10 5 8 7 1 3 | 4 8 9



(b)



(c)

Fig. 7. Three alternative designs of the Pareto front of the Van Camp problem: (a) $RC = 0$ and $MHC = 28\,577.016$; (b) $RC = 170$ and $MHC = 27\,415.232$; and (c) $RC = 999.826$ and $MHC = 18\,817.596$.

6. Conclusions

Since facilities typically have long lifetimes and operational requirements change over time, the FRLP is a more common practical problem than a greenfield design. The main motivation in the FRLP studied in this paper is to consider unequal area redesign problems including fixed areas and expanded facilities with a flexible, effective and computationally tractable algorithm. In this study, stationary areas within departments, called monuments, which represent portions of departments or entire departments that cannot be moved, are allowed to be specified anywhere in the layout.

In the proposed method, the relayout problem is treated as a bi-objective problem, minimizing the material handling cost and the relayout cost, which are measured on different time and magnitude scales. Previous approaches weight these two objectives to solve the problem with a single objective (The TS could also optimize this problem using a single objective or more than two objectives with little change to the algorithm.) However, the method can efficiently find a set of non-dominated solutions for the problem, instead of a single solution, while not requiring the two objectives to be weighted. This eliminates problems of scaling and *a priori* assessment of relative importance. Computational experiments show encouraging evidence that the TS algorithm improves the layouts and adapts to the new production environment while still accommodating monuments.

Further utilization of the approach was shown by demonstrating a few approaches to choosing among those designs along the Pareto front. Basically, the decision maker will need to examine first costs (relayout costs) versus operating costs (material handling costs) to choose the best layout for the given situation. Considerations would include the expected lifetime of the facility design and the availability of initial capital funds.

Acknowledgement

This work was supported by the US National Science Foundation under grant DMI 9908322.

References

Armour, G.C. and Buffa, E.S. (1963) A heuristic algorithm and simulation approach to relative location of facilities. *Management Science,* **9**(2), 294–309.

Balakrishnan, J. and Cheng, C.H. (1998) Dynamic layout algorithms: a state-of-the-art survey. *International Journal of Management Science,* **26**(4), 507–521.

Balakrishnan, J. and Cheng, C.H. (2000) Genetic search and the dynamic layout problem. *Computers & Operations Research*, **27**(6), 587–593.

Balakrishnan, J., Cheng, C.H. and Conway, D.G. (2000) An improved pair-wise exchange heuristic for the dynamic plant layout problem. *International Journal of Production Research,* **38**(12), 3067–3077.

Balakrishnan, J., Jacobs, F.R. and Venkataramanan, M.A. (1992) Solutions for the constrained dynamic facility layout problem. *European Journal of Operational Research,* **57**, 280–286.

Baykasoglu, A. and Gindy, N. (2001) A simulated annealing algorithm for dynamic layout problem. *Computers & Operations Research*, **28**(14), 1403–1426.

Bazaraa, M.S. (1975) Computerized layout design: a branch and bound approach. *AIIE Transactions*, **7**(4), 432–438.

Bozer, Y.A., Meller, R.D. and Erlebacher, S.J. (1994) An improvement–type layout algorithm for single and multiple floor facilities. *Management Science*, **40**(7), 918–932.

Chen, S.-J., Hwang, C.-L. and Hwang, F.P. (1992) *Fuzzy Multiple Attribute Decision Making*, Springer-Verlag, Berlin, Germany.

Chiang, W.-C. (2001) Visual facility layout design system. *International Journal of Production Research*, **39**(9), 1811–1836.

Coello, C.A.C. (2000) Handling preferences in evolutionary multiobjective optimization: a survey, in *Proceedings of the Congress on Evolutionary Computation*, IEEE Service Center, Piscataway, NJ, pp. 30–37.

Coit, D.W., Smith, A.E. and Tate, D.M. (1996) Adaptive penalty methods for genetic optimization of constrained combinatorial problems. *INFORMS Journal on Computing*, **8**, 173–182.

Conway, D.G. and Venkataramanan, M.A. (1994) Genetic search and the dynamic facility layout problem. *Computers & Operations Research*, **21**(8), 955–960.

Cvetković, D. and Parmee, I.C. (2002) Preferences and their application in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, **6**(1), 42–57.

Glover, F. (1989) Tabu search part I. *ORSA Journal on Computing*, **1**, 190–206.

Glover, F. (1990) Tabu search-part II. *ORSA Journal on Computing*, **2**, 4–32.

Glover, F. and Laguna, M. (1997) *Tabu Search*, Kluwer, Boston, MA.

Glover, F., Taillard, E. and de Werra, D. (1993) A user's guide to tabu search. *Annals of Operations Research*, **41**, 3–28.

Horn, J. (1997) Multi-criterion decision making. in *Handbook of Evolutionary Computation*, Bäck, T., Fogel, D, and Michalewicz, Z. (eds.), volume 1, IOP Publishing Ltd. and Oxford University Press, pp. F1.9:1–F1.9:15.

Kaku, B.K. and Mazzola, J.B. (1997) A tabu search heuristic for the dynamic plant layout problem. *INFORMS Journal on Computing*, **9**(4), 374–383.

Keeney, R.L. and Raiffa, H. (1993) *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, Cambridge University Press, Cambridge, UK.

Kochhar, J.S. and Heragu, S.S. (1999) Facility layout design in a changing environment. *International Journal of Production Research*, **37**(10), 2429–2446.

Kouvelis, P., Kuawarwala, A.A. and Gutierrez, G.J. (1992) Algorithms for robust single and multiple period layout planning for manufacturing systems. *European Journal of Operational Research*, **63**, 287–303.

Kulturel-Konak, S., Norman, B.A., Coit, D.W. and Smith, A.E. (2004) Exploiting tabu search memory in constrained problems. *INFORMS Journal on Computing*, **14**(3), 241–254.

Kulturel-Konak, S., Smith, A.E. and Norman, B.A. (2006) Multi-objective tabu search using a multinomial probability mass function. *European Journal of Operational Research*, **169**(3), 915–931.

Lacksonen, T.A. (1994) Static and dynamic layout problems with varying areas. *Journal of the Operational Research Society*, **45**(1), 59–69.

Lacksonen, T.A. and Enscore, E.E. (1993) Quadratic assignment algorithms for the dynamic layout problem. *International Journal of Production Research*, **31**(3), 503–517.

Meller, R.D. (1997) The multi-bay manufacturing facility layout problem. *International Journal of Production Research*, **35**(5), 1229–1237.

Meller, R.D. and Bozer Y.A. (1997) Alternative approaches to solve the multi-floor facility layout problem. *Journal of Manufacturing Systems*, **16**(3), 192–203.

Meller, R.D. and Gau, K.-Y. (1996) Facility layout objective functions and robust layouts. *International Journal of Production Research*, **34**(9), 2727–2742.

Montreuil, B. and Laforge, A. (1992) Dynamic layout design given a scenario tree of probable futures. *European Journal of Operational Research*, **63**, 271–286.

Montreuil, B. and Venkatadri, U. (1991) Strategic interpolative design of dynamic manufacturing systems layouts. *Management Science*, **37**(6), 682–694.

Rosenblatt, M.J. (1986) The dynamics of plant layout. *Management Science*, **32**(1), 76–86.

Rosenblatt, M.J. and Sinuany-Stern, Z. (1986) A discrete efficient frontier approach to the plant layout problem. *Material Flow*, **3**, 277–281.

Smith, A.E. and Tate, D.M. (1993) Genetic optimization using a penalty function, in *Proceedings of the 5th International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 499–505.

Tate, D.M. and Smith, A.E. (1995) Unequal area facility layout using genetic search. *IIE Transactions*, **27**, 465–472.

Urban, T.L. (1992) Computational performance and efficiency of lower-bound procedures for the dynamic facility layout problem. *European Journal of Operational Research*, **57**, 271–279.

Urban, T.L. (1993) A heuristic for the dynamic facility layout problem. *IIE Transactions*, **25**(4), 57–63.

Urban, T.L. (1998) Solution procedures for the dynamic facility layout problem. *Annals of Operations Research*, **76**, 323–342.

Van Camp, D.J., Carter, M.V. and Vanelli, A. (1991) A nonlinear optimization approach for solving facility layout problems. *European Journal of Operational Research*, **57**, 174–189.

Yang, T. and Peters, B.A. (1998) Flexible machine layout design for dynamic and uncertain production environments. *European Journal of Operational Research*, **108**, 49–64.

Appendices

Appendix A

1. The Van Camp problem

Facility:
$W = 51$ m $H = 25$ m Facility Area: $25 \times 51 = 1275$ m^2
Distances are *Rectilinear*.
Maximum aspect ratio is five.

Table A1. Departmental areas

	Department									
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Area (m^2)	238	112	160	80	120	80	60	85	221	119

Table A2. Estimated volumes and routings of products

Product	*Mean*	*Routing*
1	10	3-5-10
2	100	1-5-8-7
3	50	2-9-6-4
4	50	2-9-5-8-7

Table A3. Departmental area changes for relayout with the existing facility area

Dept.	*Old area(m^2)*	*New area(m^2)*	*Change in area (m^2)*
1	238	188	−50
7	60	80	+20
10	119	149	+30

Table A4. Estimated volumes and routings of products for relayout with the existing facility area

Product	*Mean*	*Routing*
1	20	3-5-7-9
2	75	3-5-8-10
3	100	1-2-5-6-4
4	50	1-2-5-7-8-10
5	150	5-7-4-9-10

Table A5. Additional departmental areas for relayout with expansion

New Dept.	*Area (m^2)*	*Overall facility area (m^2)*
11	110	$25 \times 51 = 1275$ (old)
12	90	$25 \times 61 = 1525$ (new)
13	50	

Table A6. Estimated volumes and routings of products for relayout with expansion

Product	*Mean*	*Routing*
1	20	3-5-7-11-9
2	75	3-5-8-10-12
3	100	1-6-4-13
4	50	2-5-8-10-7-11
5	150	5-13-7-9-10-12

Table A7. Locations of the monuments for relayout

Fixed dept.	*Monument (x_i^{FL}, y_i^{FL}) (x_i^{FU}, y_i^{FU})*
9	(10, 13) (18, 15)

2. *The Bazaraa problem*

Facility:
$W = 9$ blocks $H = 7$ blocks Facility area: $7 \times 9 = 63$ block^2
Distances are *Rectilinear*. Maximumaspect ratio is two.

Table A8. Departmental areas

	Department													
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*
Area (block^2)	9	8	9	10	6	3	3	3	2	3	2	1	1	3

Table A9. Estimated volumes and routings of products

Product	*Mean*	*Routing*
1	100	2-8-9-1
2	150	3-8-9-11-12-6
3	50	7-10-13-6
4	100	5-9-12-4-1
5	100	6-8-5-4-11-10
6	120	4-13-12

Table A10. Estimated volumes and routings of products for relayout with the existing facility area

Product	*Mean*	*Routing*
1	120	2-8-7-10
2	100	3-8-9-12-6
3	75	7-10-12-9-6
4	90	5-9-12-4-14-1-13-11
5	125	6-5-14-4-12-10
6	100	4-3-5-13-11-12
7	200	2-8-4-14-1

Table A11. Additional departmental areas for relayout with expansion

Dept.	*Area (block^2)*	*Overall facility area (block^2)*
15	6	$7 \times 9 = 63$ (old)
16	8	$7 \times 11 = 77$ (new)

Table A12. Routings and estimated volumes of products for relayout with expansion

Product	*Mean*	*Routing*
1	120	2-8-7-1
2	100	3-8-11-12-6
3	75	7-10-15-12-9-6
4	90	5-9-12-4-16-13-1
5	125	6-5-4-16-12-10
6	100	4-3-5-13-11-12
7	200	2-15-4-16

Table A13. Locations of the monuments for relayout

Fixed dept.	*Monument (x_i^{FL}, y_i^{FL}) (x_i^{FU}, y_i^{FU})*
3	(5, 3.5) (6, 4.5)
4	(0, 0) (1, 2)

3. *The Armour-Buffa problem*

Facility:
$W = 3$ $H = 2$ Facility area: $3 \times 2 = 6$.
Distances are *Rectilinear*. Maximum aspect ratio is three.

Table A14. Departmental areas

Dept.	*Area*	*Dept.*	*Area*	*Dept.*	*Area*	*Dept.*	*Area*
1	0.27	6	0.18	11	0.60	16	0.75
2	0.18	7	0.09	12	0.42	17	0.64
3	0.27	8	0.09	13	0.18	18	0.41
4	0.18	9	0.09	14	0.24	19	0.27
5	0.18	10	0.24	15	0.27	20	0.45

Table A15. Estimated volumes and routings of products

Product	*Mean*	*Routing*
1	100	1-3-7-8-16-19
2	25	7-8-2-9-11
3	50	4-5-12-13-15-17
4	40	6-7-1-10-14-15
5	125	18-12-7-8-11-13-17
6	10	3-8-9-10-12-15
7	50	5-6-2-3-8-19-20-15
8	50	2-4-10-17-19

Table A16. Estimated volumes and routings of products for relayout with the existing facility area

Product	*Mean*	*Routing*
1	75	1-3-7-2-8-16-19
2	75	7-8-2-9-11-15-17
3	75	1-4-3-5-12-13-15-11-17
4	100	6-7-10-14-15-18
5	100	18-17-6-12-7-8-17
6	20	3-8-7-9-10-15-18
7	25	3-8-7-9-10-15-18-20

Table A17. Locations of the monuments for relayout

Fixed dept.	*Monument* (x_i^{FL}, y_i^{FL}) (x_i^{FU}, y_i^{FU})
5	(0.1, 0.05) (0.25, 0.25)
19	(1.2, 1.2) (1.4, 1.4)
20	(2.3, 1.5) (2.6, 1.9)

Appendix B

Table B1. Pareto set of the Van Camp problem with existing facility area and with no departmental area change

Solution	*RC*	*MHC*	*Encoding*
1	0.000	28 577.016	4 6 9 2 10 5 8 7 1 3 | 4 8 9
2	170.000	27 415.232	4 6 9 2 10 8 5 7 1 3 | 4 8 9
3	361.357	26 508.751	4 6 9 2 10 8 5 7 3 1 | 4 8
4	402.734	22 875.939	4 6 9 10 2 5 8 7 1 3 | 4 8 9
5	478.565	21 907.487	4 6 9 10 8 5 2 7 1 3 | 4 8 9
6	529.091	19 996.678	4 6 9 10 8 2 5 7 1 3 | 4 8 9
7	720.448	19 680.336	4 6 9 10 8 2 5 7 3 1 | 4 8
8	877.919	19 224.261	6 4 9 10 8 7 5 2 1 3 | 4 6 8
9	975.933	19 173.084	6 4 9 10 8 7 5 3 1 2 | 4 6 8
10	999.826	18 817.596	10 9 3 5 7 4 6 8 1 2 | 3 8

Table B2. Pareto set of the Bazaraa problem with existing facility area and with no departmental area change

Solution	*RC*	*MHC*	*Encoding*
1	0.000	16 171.706	2 5 4 11 12 13 9 8 3 6 10 7 1 14 | 3 8 11
2	120.000	14 452.623	2 5 4 11 12 13 9 8 3 6 10 7 14 1 | 3 8 11
3	160.000	14 415.679	2 5 4 11 12 9 13 8 3 6 10 7 14 1 | 3 8 11
4	240.000	13 660.262	2 5 4 11 12 13 9 8 3 6 14 1 10 7 | 3 8 11
5	280.000	13 478.262	2 5 4 11 13 12 9 8 3 6 14 1 10 7 | 3 8 11
6	300.000	13 419.734	2 5 4 11 12 13 9 8 3 14 6 1 10 7 | 3 8 11
7	340.000	13 268.845	2 5 4 11 13 12 9 8 3 14 6 1 10 7 | 3 8 11
8	985.411	12 729.807	6 14 4 8 12 9 13 11 5 3 2 7 10 1 | 3 8 11

Table B3. Pareto set of the Armour-Buffa problem with existing facility area

Solution	*RC*	*MHC*	*Encoding*
1	0.000	3364.494	16 4 6 5 1 2 9 10 14 15 19 8 7 3 12 13 17 20 11 18 | 4 9 14 17 20
2	90.000	3233.879	16 4 13 5 1 2 9 10 14 15 19 8 7 3 12 6 17 20 11 18 | 4 9 14 17 20
3	135.000	3192.018	16 13 4 5 1 2 9 10 14 15 19 8 7 3 12 6 17 20 11 18 | 4 9 14 17 20
4	180.000	3133.112	16 13 4 5 1 2 9 10 14 15 19 8 7 3 6 12 17 20 11 18 | 4 9 14 17 20
5	305.345	2989.327	16 4 3 5 1 2 9 10 14 15 19 8 7 6 12 13 17 20 11 18 | 4 9 14 17 20
6	383.140	2986.687	16 1 4 5 3 2 9 10 14 15 19 8 7 6 12 13 17 20 11 18 | 4 9 14 17 20
7	395.345	2942.875	16 4 3 5 1 2 9 10 14 13 19 8 7 6 12 17 15 20 11 18 | 4 9 14 17 20
8	443.027	2910.243	16 4 3 5 1 2 9 10 14 13 19 8 7 6 12 15 17 20 11 18 | 4 9 14 17 20
9	469.391	2850.044	16 4 3 5 1 2 9 10 14 13 19 8 7 6 12 17 15 20 11 18 | 4 9 14 17 20
10	547.186	2847.405	16 1 4 5 3 2 9 10 14 13 19 8 7 6 12 17 15 20 11 18 | 4 9 14 17 20
11	560.497	2787.040	16 4 3 5 1 2 10 14 9 19 8 7 6 12 13 17 20 15 11 18 | 4 8 13 16 20
12	562.058	2782.523	16 4 3 5 1 2 10 14 9 19 8 7 6 12 13 17 20 11 15 18 | 4 8 13 16 20
13	591.895	2781.382	16 3 4 5 1 2 10 14 9 19 8 7 6 12 13 17 20 11 15 18 | 4 8 13 16 20
14	645.808	2779.557	16 1 4 5 3 2 10 14 9 19 8 7 6 12 13 17 20 11 15 18 | 4 8 13 16 20
15	661.356	2744.479	16 4 3 5 1 2 10 14 19 8 9 7 6 12 13 17 20 15 11 18 | 4 7 13 16 20
16	671.974	2740.420	16 4 3 5 1 2 10 14 19 9 8 7 6 12 13 17 20 15 11 18 | 4 7 13 16 20
17	679.005	2699.873	16 4 3 5 1 10 14 9 19 8 2 7 6 12 13 17 20 15 11 18 | 4 7 13 16 20
18	680.566	2687.923	16 4 3 5 1 10 14 9 19 8 2 7 6 12 13 17 20 11 15 18 | 4 7 13 16 20
19	795.981	2682.708	16 4 3 5 2 1 10 14 19 9 8 7 6 12 17 13 20 15 11 18 | 4 7 13 16 20

Table B4. Pareto set of the Van Camp problem with existing facility area and with some departmental area changes

Solution	*RC*	*MHC*	*Encoding*
1	50.000	28 826.833	4 6 9 2 10 5 8 7 1 3 | 4 8 9
2	220.000	27 853.619	4 6 9 2 10 8 5 7 1 3 | 4 8 9
3	337.868	27 706.359	4 6 9 2 10 7 8 5 1 3 | 4 5 9
4	391.400	26 766.496	4 6 9 8 10 5 2 7 1 3 | 4 8 9
5	464.114	25 072.624	4 6 9 8 10 2 5 7 1 3 | 4 8 9
6	542.012	23 832.851	4 9 10 2 6 5 8 7 1 3 | 3 8 9
7	549.224	23 748.993	4 9 10 6 2 5 8 7 1 3 | 3 8 9
8	569.535	23 333.522	10 9 4 6 2 5 8 7 1 3 | 3 8 9
9	607.818	21 978.815	4 9 10 2 8 5 6 7 1 3 | 3 8 9
10	615.480	21 120.193	4 9 10 8 2 5 6 7 1 3 | 3 8 9
11	674.778	20 520.613	4 9 10 8 2 5 7 6 1 3 | 3 8 9
12	871.274	20 429.762	10 9 4 6 7 5 2 8 1 3 | 3 8
13	899.970	20 188.196	9 8 10 2 7 5 4 6 1 3 | 3 8 9
14	916.322	19 864.114	9 8 10 2 5 7 6 4 1 3 | 3 8 9
15	928.105	19 102.007	9 8 10 2 5 7 4 6 1 3 | 3 8 9
16	1100.979	19 004.261	9 8 10 2 5 7 4 6 3 1 | 3 8
17	1159.345	18 581.410	9 10 8 7 4 6 5 3 2 1 | 3 5 7

Table B5. Pareto set of the Van Camp problem with expansion

Solution	*RC*	*MHC*	*Encoding*
1	0.000	44 460.031	4 6 9 2 10 5 8 7 1 3 13 11 12 | 4 8 9 10
2	38.095	42 231.456	4 6 9 2 10 5 8 7 1 3 13 11 12 | 4 8 9 11
3	97.600	39 256.934	4 6 9 2 10 5 8 7 13 1 3 11 12 | 4 8 10 11
4	158.711	38 184.598	4 6 9 2 10 5 8 7 13 1 12 3 11 | 4 8 10
5	207.600	38 000.934	4 6 9 2 10 5 8 7 13 1 12 11 3 | 4 8 10 12
6	221.157	37 902.437	4 6 9 2 10 5 8 13 7 1 3 11 12 | 4 8 10 11
7	227.400	37 714.885	4 6 9 2 10 5 13 7 1 3 11 8 12 | 4 8 9 10
8	236.296	37 156.637	4 6 9 2 10 5 8 7 13 1 12 3 11 | 4 8 11
9	257.432	35 205.938	4 6 9 2 10 5 13 7 1 8 3 11 12 | 4 8 10 11
10	270.000	34 762.831	4 6 9 2 10 5 8 7 13 11 12 3 1 | 4 8 11 12

(Continued on next page)

Table B5. Pareto set of the Van Camp problem with expansion *(Continued)*

Solution	*RC*	*MHC*	*Encoding*
11	300.118	34 435.194	4 6 9 2 7 13 8 5 10 1 3 12 11 | 4 6 9 10 11
12	300.900	34 429.669	4 6 9 2 10 5 7 13 1 8 3 11 12 | 4 8 10 11
13	331.600	33 558.189	4 6 9 2 11 5 13 7 1 3 12 10 8 | 4 8 9 10
14	347.951	33 021.898	4 6 9 2 8 5 13 7 1 10 3 11 12 | 4 8 10 11
15	380.831	32 990.315	4 6 9 2 10 7 13 5 8 1 3 12 11 | 4 7 9 10 11
16	391.212	32 264.978	4 6 9 2 11 5 13 7 1 3 8 10 12 | 4 8 9
17	395.951	32 118.053	4 6 9 2 8 5 7 13 1 10 3 11 12 | 4 8 10 11
18	409.062	32 081.398	4 6 9 2 8 5 13 7 1 10 12 3 11 | 4 8 10
19	414.800	30 678.685	4 6 9 2 10 5 13 7 11 8 12 3 1 | 4 8 11 12
20	458.269	30 029.269	4 6 9 2 10 5 7 13 11 8 12 3 1 | 4 8 11 12
21	510.418	29 110.633	4 6 9 2 10 7 8 13 5 12 11 3 1 | 4 8 10 12
22	570.342	28 999.856	4 6 9 2 10 7 13 12 11 8 5 3 1 | 4 6 8 11 12
23	636.303	28 030.837	4 6 9 2 7 13 5 8 10 12 11 1 3 | 4 6 10
24	676.776	27 856.689	10 8 9 12 11 5 13 7 1 3 4 6 2 | 4 8 9 10
25	699.156	26 487.168	10 9 11 7 12 8 5 13 1 3 4 6 2 | 3 5 8 9 10
26	770.681	26 026.442	8 10 9 12 7 11 5 13 1 3 4 6 2 | 4 6 8 9 10
27	782.192	25 896.595	8 10 9 12 11 7 5 13 1 3 4 6 2 | 4 8 9 10
28	798.737	25 267.670	3 9 2 10 8 5 12 11 7 13 4 6 1 | 3 5 7 11
29	855.452	25 168.190	8 9 10 12 7 11 5 13 1 3 4 6 2 | 4 6 8 9 10
30	889.556	24 228.368	10 9 11 7 12 8 5 13 4 6 2 3 1 | 3 5 8 11 12
31	914.015	23 961.126	11 9 10 12 7 4 13 5 8 2 6 3 1 | 3 5 9 11 12
32	1002.015	23 638.913	11 9 10 12 7 4 13 5 8 3 6 1 2 | 3 5 9 11
33	1014.969	22 423.449	11 12 10 9 7 13 4 5 8 2 6 3 1 | 2 5 9 11 12
34	1082.305	21 935.217	12 11 7 9 10 8 5 2 13 4 6 3 1 | 2 5 9 12
35	1162.211	21 823.194	11 2 12 10 9 7 5 8 13 4 6 3 1 | 2 5 7 9 12
36	1173.118	21 369.549	11 12 10 9 7 8 5 13 4 2 3 1 6 | 2 4 6 8 11
37	1184.936	21 147.796	11 12 10 9 7 8 5 13 4 6 3 2 1 | 2 4 6 8 11
38	1199.255	19 853.848	11 2 12 10 9 7 8 5 13 4 6 3 1 | 2 5 7 9 12
39	1217.922	19 752.506	11 2 12 10 9 7 8 5 13 4 3 1 6 | 2 5 7 9 11

Table B6. Pareto set of the Bazaraa problem with expansion

Solution	*RC*	*MHC*	*Encoding*
1	0.000	18 301.464	2 5 4 11 12 13 9 8 3 6 10 7 1 14 15 16 | 3 8 11 14
2	80.000	18 167.298	2 5 4 9 12 13 11 8 3 6 10 7 1 14 15 16 | 3 8 11 14
3	120.000	16 882.119	2 15 4 11 12 13 9 8 3 6 10 7 1 14 5 16 | 3 8 11 14
4	160.000	14 551.464	16 5 4 11 12 13 9 8 3 6 10 7 1 14 2 15 | 3 8 11 14
5	200.000	14 374.520	16 5 4 11 12 9 13 8 3 6 10 7 1 14 2 15 | 3 8 11 14
6	260.000	14 141.187	16 5 4 9 12 11 13 8 3 6 10 7 1 14 2 15 | 3 8 11 14
7	280.000	13 667.833	16 5 4 11 12 13 9 6 3 8 10 7 1 14 2 15 | 3 8 11 14
8	320.000	13 607.556	16 5 4 11 12 9 13 6 3 8 10 7 1 14 2 15 | 3 8 11 14
9	380.000	13 577.000	16 5 4 11 12 9 13 6 3 10 8 7 1 14 2 15 | 3 8 11 14
10	400.000	13 075.750	5 16 4 11 12 13 9 6 3 8 10 7 1 14 2 15 | 3 8 11 14
11	440.000	12 970.750	5 16 4 11 12 9 13 6 3 8 10 7 1 14 2 15 | 3 8 11 14
12	500.000	12 940.194	5 16 4 11 12 9 13 6 3 10 8 7 1 14 2 15 | 3 8 11 14
13	560.000	12 917.976	5 16 4 11 12 9 13 6 3 15 10 1 14 7 8 2 | 3 8 10 13
14	572.235	12 801.457	16 5 4 11 12 13 9 6 3 15 10 8 7 14 1 2 | 3 8 10 14
15	580.000	12 766.619	5 16 4 11 12 13 9 6 3 8 10 15 1 14 7 2 | 3 8 11 13
16	612.235	12 601.179	16 5 4 11 12 9 13 6 3 15 10 8 7 14 1 2 | 3 8 10 14
17	620.000	12 521.619	5 16 4 11 12 9 13 6 3 8 10 15 1 14 7 2 | 3 8 11 13
18	680.000	12 382.333	5 16 4 11 12 9 13 6 3 15 10 8 1 14 7 2 | 3 8 10 13
19	692.235	12 209.373	5 16 4 11 12 13 9 6 3 15 10 8 7 14 1 2 | 3 8 10 14
20	732.235	11 964.373	5 16 4 11 12 9 13 6 3 15 10 8 7 14 1 2 | 3 8 10 14

Biographies

Sadan Kulturel-Konak, is currently an Assistant Professor of Management Information Systems at Penn State Berks. She received her degrees in Industrial Engineering; B.S. from Gazi University, Turkey in 1993, M.S. from Middle East Technical University, Turkey in 1996 and from the University of Pittsburgh in 1999, and Ph.D. from Auburn University in 2002. Her research interests are in modeling and optimization of complex systems and robustness under uncertainty with applications to facility layout, reliability, and scheduling. She has published her research in *IIE Transactions*, *OR Letters*, *INFORMS Journal on Computing*, *International Journal of Production Research*, *European Journal of Operational Research*, and the *Journal of Intelligent Manufacturing*. She is a member of INFORMS, IIE, SWE, Alpha Phi Mu, and Phi Kappa Phi.

Alice E. Smith, is Professor and Chair of the Industrial and Systems Engineering Department at Auburn University. Previous to this position, she was on the faculty of the Department of Industrial Engineering at the University of Pittsburgh, which she joined in 1991 after 10 years of industrial experience with Southwestern Bell Corporation. She has degrees in engineering and business from Rice University, Saint Louis University and University of Missouri—Rolla. Her research in analysis, modeling and optimization of manufacturing processes and engineering design has been funded by NASA, the National Institute of Standards (NIST), Lockheed Martin, Adtranz (now Bombardier Transportation), the Ben Franklin Technology Center of Western Pennsylvania and the National Science Foundation (NSF), from which she was awarded a CAREER grant in 1995 and an ADVANCE Leadership grant in 2001. International research collaborations have been sponsored by the federal governments of Japan, Turkey, United Kingdom and the US. She has served as a principal investigator on over $3 million of sponsored research. She was named a Philpott-WestPoint Stevens Distinguished Professor in 2001 by the Auburn University College of Engineering. For outstanding achievements in research and scholarly activity she received the annual Senior Research Award of the College of Engineering at Auburn University in 2001 and the University of Pittsburgh School of Engineering Board of Visitors annual Faculty Award in 1996. She holds one US patent and several international patents and has authored over 50 publications in journals including articles in *IIE Transactions*, *IEEE Transactions on Reliability*, *INFORMS Journal on Computing*, *International Journal of Production Research*, *IEEE Transactions on Systems, Man, and Cybernetics*, *Journal of Manufacturing Systems*, *The Engineering Economist*, and *IEEE Transactions on Evolutionary Computation*. She won the E. L. Grant Best Paper Award in 1999 and the William A. J. Golomski Best Paper Award in 2002. She holds editorial positions on *INFORMS Journal on Computing*, *Computers & Operations Research*, *International Journal of General Systems*, *IEEE Transactions on Evolutionary Computation* and *IIE Transactions*. Five of her doctoral students have obtained tenure track positions at US universities and two of these are NSF CAREER awardees. She is a fellow of IIE, a senior member of IEEE and SWE, a member of Tau Beta Pi, INFORMS, NSBE and ASEE, and a Registered Professional Engineer in Industrial Engineering in Alabama and Pennsylvania.

Bryan A. Norman, is an Associate Professor of Industrial Engineering at the University of Pittsburgh. He received his Ph.D. degree in Industrial and Operations Engineering from the University of Michigan in 1995, where he was a National Science Foundation Fellowship holder, and has B.S.I.E. and M.S.I.E. degrees from the University of Oklahoma. His research interests primarily focus on the modeling of complex problems in manufacturing and production systems and applied optimization. His areas of application include scheduling, sequencing, job rotation, assembly line balancing, facility layout, material handling system design, network design and energy modeling. His research has been funded by several sources including the National Science Foundation and local industry. He has published his research in *IIE Transactions*, *Naval Research Logistics*, *INFORMS Journal on Computing*, *International Journal of Production Research*, *European Journal of Operational Research*, *Annals of Operations Research* and *Computers and Industrial Engineering*. He is a member of IIE and INFORMS.