

2010

Heuristics for the Dynamic Facility Layout Problem with Unequal-area Departments

Alan McKendall

Artak Hakobyan

Follow this and additional works at: https://researchrepository.wvu.edu/faculty_publications



Part of the [Operations Research, Systems Engineering and Industrial Engineering Commons](#)



Production, Manufacturing and Logistics

Heuristics for the dynamic facility layout problem with unequal-area departments

Alan R. McKendall Jr. *, Artak Hakobyan

Department of Industrial and Management Systems Engineering, West Virginia University, 325A Mineral Resources Building, Morgantown, WV 26506, USA

ARTICLE INFO

Article history:

Received 30 June 2008

Accepted 23 February 2009

Available online 1 March 2009

Keywords:

Facilities planning and design
 Dynamic facility layout problem
 Unequal-area departments
 Tabu search
 Boundary search
 Meta-heuristic

ABSTRACT

The dynamic facility layout problem (DFLP) is the problem of finding positions of departments on the plant floor for multiple periods (material flows between departments change during the planning horizon) such that departments do not overlap, and the sum of the material handling and rearrangement costs is minimized. In this paper, the departments may have unequal-areas and free orientations, and the layout for each period is generated on the continuous plant floor. Because of the complexity of the problem, only small-size problems can be solved in reasonable time using exact techniques. As a result, a boundary search (construction) technique, which places departments along the boundaries of already placed departments, is developed for the DFLP. The solution is improved using a tabu search heuristic. The heuristics were tested on some instances from the DFLP and static facility layout problem (SFLP) literature. The results obtained demonstrate the effectiveness of the heuristics.

Published by Elsevier B.V.

1. Introduction

The static facility layout problem (SFLP) is a well-researched problem of finding positions of departments on the plant floor such that departments do not overlap while some objective is optimized. The most commonly used objective is minimizing material handling cost (i.e., minimizing the sum of the product of the flow of materials, distance, and transportation cost per unit per distance unit for each pair of departments). When material flows between departments change during the planning horizon, the problem becomes the dynamic facility layout problem (DFLP). Some of the factors, which may change material flows, are as follows and were taken from [Shore and Tompkins \(1980\)](#):

- Changes in the design of an existing product.
- The addition or deletion of products.
- Replacement of existing production equipment.
- Shorter product life cycles.
- Changes in the production quantities and associated production schedules.

For a review of the SFLP, see [Kusiak and Heragu \(1987\)](#) and [Meller and Gau \(1996\)](#).

The DFLP is the problem of finding positions of departments on the plant floor for multiple periods such that departments do not overlap, and the sum of the material handling and rearrangement costs is minimized. In other words, for each period in the planning horizon, the layout is determined such that the sum of the material handling cost for each layout and the cost of rearranging departments between each pair of consecutive layouts is minimized. [Rosenblatt \(1986\)](#) was the first to present solution techniques (i.e., optimal and heuristic procedures based on dynamic programming) for the DFLP. However, the author considered equal area departments and used the discrete representation of the layout (used equal-size grids to represent departments on the plant floor). Most of the solution techniques available in the literature for the DFLP use the discrete representation of the layout and are approximation techniques, since the DFLP is computationally intractable. For a review of the problem assumptions and solution techniques for the DFLP, see [Balakrishnan and Cheng \(1998\)](#) and [Kulturel-Konak \(2007\)](#).

In this paper, a boundary search technique, which places departments along the boundaries of already placed departments, is used to construct a solution for the DFLP under the following assumptions.

* Corresponding author. Tel.: +1 304 293 4607.

E-mail address: armckendall@mail.wvu.edu (A.R. McKendall Jr.).

- (1) Departments may have unequal-areas. In other words, departments are either square or rectangular in shape.
- (2) Department areas are fixed for each period but may vary from one period to another.
- (3) Departments may have free orientations (i.e., departments may be either horizontally or vertically oriented). If the longer side of the department is parallel to the x -axis, the department is horizontally oriented. In contrast, if the longer side is parallel to the y -axis, the department is vertically oriented.
- (4) The layout for each period uses the continuous representation of the plant floor.
- (5) The objective of the DFLP is to obtain a layout for each period in the planning horizon such that the sum of material handling and rearrangement costs is minimized. Rearrangement cost is incurred when a department is said to be rearranged (i.e., department centroid or dimensions change in consecutive periods).

Once a solution is constructed using the boundary search technique, a tabu search heuristic is used to improve the solution.

Montreuil and Venkatadri (1991) presented the first formulation for the DFLP with unequal-area variable shape departments. Their mathematical formulation does not require binary variables, since the relative positions of pairs of departments are known. This linear programming (LP) model was improved by Montreuil and Laforge (1992) by relaxing some assumptions. Other papers, which considered the DFLP with unequal-area variable shape departments, are Lacksonen (1994, 1997). In both cases, the author used a mixed integer linear programming (MILP) model to solve their DFLP. The only papers, in the literature, known to the authors, which consider the DFLP under assumptions (1)–(5), are Yang and Peters (1998) and Dunker et al. (2005). Yang and Peters (1998) considered time windows when solving the DFLP. The authors solve a series of SFLPs, one for each time window, using a MILP formulation. The structured hexagonal adjacency graph from Goetschalckx (1992) is used to fix the binary variables corresponding to relative positions of departments in each time window. Dunker et al. (2005) used a hybrid approach, which combined dynamic programming with a genetic algorithm. Each gene stores information about the relative positions of departments in a layout for a period. The solution corresponding to a gene is obtained by solving a relaxed MILP formulation for the SFLP in which the only unknown binary variables are variables representing the orientations of departments and configurations of input/output stations. Dynamic programming is used to evaluate the fitness of each gene.

All the solution techniques, which use the continuous representation of the layout for the DFLP with unequal-area departments, use either a LP or MILP formulation to determine the layout for each period (i.e., the layout plan). However, in this paper, the boundary search technique, which is less costly computationally, is used to determine the layout plan.

The structure of this paper is as follows. In Section 2, a mathematical formulation for the DFLP with unequal-area departments is presented. In Section 3, a boundary search heuristic and a tabu search heuristic are presented for the DFLP. Computational results for the heuristics are presented in Section 4, and Section 5 concludes the paper.

2. A mathematical formulation for the DFLP

In this section, a MILP formulation is presented for the DFLP with unequal-area departments. A similar formulation can be found in Dunker et al. (2005), except that in this paper input/output stations are at the centroids of the departments. First, the notation is given as follows. It is important to note that most of the indexes and parameters defined in this section are used later for the heuristics.

Indexes

$i, j = 1, \dots, N$ where N is the number of departments
 $t = 1, \dots, T$ where T is the number of periods

Parameters

F_{tij} cost to transport materials a unit distance from department i to department j in period t
 $F'_{tij} = F_{tij} + F_{tji}$ total cost to transport materials a unit distance between departments i and j in period t (upper triangular matrix)
 R_{ti} rearrangement cost of shifting department i at the beginning of period t
 Sh_{ti} shorter side length of department i in period t
 Lng_{ti} longer side length of department i in period t

$$DeptOrient_{ti} = \begin{cases} 0 & \text{if department } i \text{ in period } t \text{ can have any orientation} \\ 1 & \text{if department } i \text{ in period } t \text{ is restricted to horizontal orientation} \\ 2 & \text{if department } i \text{ in period } t \text{ is restricted to vertical orientation} \end{cases}$$

L length of the plant floor
 W width of the plant floor
 M a large number

Variables

(x_{ti}, y_{ti}) the centroid (or location) of department i in period t
 l_{ti}, w_{ti} the length and width of department i in period t
 x_p_{tij}, y_p_{tij} horizontal and vertical distances between the centers of departments i and j in period t
 $h_{ti} = \begin{cases} 1, & \text{If department } i \text{ has horizontal orientation in period } t \\ 0, & \text{Otherwise} \end{cases}$
 $left_{tij} = \begin{cases} 1 & \text{if department } i \text{ is to the left of department } j \text{ in period } t \text{ (i.e., } x_{ti} + 0.5 * l_{ti} \leq x_{tj} - 0.5 * l_{tj}) \\ 0 & \text{Otherwise} \end{cases}$
 $below_{tij} = \begin{cases} 1 & \text{if department } i \text{ is below department } j \text{ in period } t \text{ (i.e., } y_{ti} + 0.5 * w_{ti} \leq y_{tj} - 0.5 * w_{tj}) \\ 0 & \text{otherwise} \end{cases}$
 $r_{ti} = \begin{cases} 1 & \text{if department } i \text{ is rearranged at the beginning of period } t \\ 0 & \text{otherwise} \end{cases}$

Next, a MILP formulation is presented for the proposed problem:

$$\text{Minimize total cost} = \sum_{t=1}^T \sum_{i=1}^N \sum_{j>i}^N F'_{ij}(x \cdot p_{tij} + y \cdot p_{tij}) + \sum_{t=2}^T \sum_{i=1}^N R_{ti} r_{ti} \quad (2.1)$$

$$\text{Subject to } (x_{ti} + 0.5l_{ti}) - (x_{tj} - 0.5l_{tj}) \leq M(1 - \text{left}_{tij}) \quad \forall t, i, j, \quad (2.2)$$

$$(y_{ti} + 0.5w_{ti}) - (y_{tj} - 0.5w_{tj}) \leq M(1 - \text{below}_{tij}) \quad \forall t, i, j, \quad (2.3)$$

$$\text{left}_{tij} + \text{left}_{tji} + \text{below}_{tij} + \text{below}_{tji} = 1 \quad \forall t, i, j, \quad (2.4)$$

$$x_{ti} + 0.5l_{ti} \leq L \quad \forall t, i, \quad (2.5)$$

$$x_{ti} - 0.5l_{ti} \geq 0 \quad \forall t, i, \quad (2.6)$$

$$y_{ti} + 0.5w_{ti} \leq W \quad \forall t, i, \quad (2.7)$$

$$y_{ti} - 0.5w_{ti} \geq 0 \quad \forall t, i, \quad (2.8)$$

$$x \cdot p_{tij} \geq x_{ti} - x_{tj} \quad \forall t, i, j > i, \quad (2.9)$$

$$x \cdot p_{tji} \geq x_{tj} - x_{ti} \quad \forall t, i, j > i, \quad (2.10)$$

$$y \cdot p_{tij} \geq y_{ti} - y_{tj} \quad \forall t, i, j > i, \quad (2.11)$$

$$y \cdot p_{tji} \geq y_{tj} - y_{ti} \quad \forall t, i, j > i, \quad (2.12)$$

$$l_{ti} = \text{Lng}_{ti} h_{ti} + \text{Sh}_{ti}(1 - h_{ti}) \quad \forall t, i, \quad (2.13)$$

$$w_{ti} = \text{Lng}_{ti}(1 - h_{ti}) + \text{Sh}_{ti} h_{ti} \quad \forall t, i, \quad (2.14)$$

$$h_{ti} = 1 \quad \forall t, i, \quad \text{DeptOrient}_{ti} = 1, \quad (2.15)$$

$$h_{ti} = 0 \quad \forall t, i, \quad \text{DeptOrient}_{ti} = 2, \quad (2.16)$$

$$x_{ti} - x_{t-1,i} \leq Mr_{ti} \quad \forall i, t > 1, \quad (2.17)$$

$$-x_{ti} + x_{t-1,i} \leq Mr_{ti} \quad \forall i, t > 1, \quad (2.18)$$

$$y_{ti} - y_{t-1,i} \leq Mr_{ti} \quad \forall i, t > 1, \quad (2.19)$$

$$-y_{ti} + y_{t-1,i} \leq Mr_{ti} \quad \forall i, t > 1, \quad (2.20)$$

$$w_{ti} - w_{t-1,i} \leq Mr_{ti} \quad \forall i, t > 1, \quad (2.21)$$

$$-w_{ti} + w_{t-1,i} \leq Mr_{ti} \quad \forall i, t > 1, \quad (2.22)$$

$$x_{ti}, y_{ti}, l_{ti}, w_{ti}, x \cdot p_{tij}, y \cdot p_{tij} \geq 0 \quad \forall t, i, j, \quad (2.23)$$

$$h_{ti} = 0 \text{ or } 1, \quad r_{ti} = 0 \text{ or } 1, \quad \text{left}_{tij} = 0 \text{ or } 1, \quad \text{below}_{tij} = 0 \text{ or } 1 \quad \forall t, i, j.$$

The first term in objective function (2.1) is used to obtain material handling costs, and the second term is for rearrangement costs. Constraints (2.2)–(2.8) are very similar to those presented in Sherali et al. (2003). These constraints ensure that the departments do not overlap and are within the boundaries of the plant floor. Constraints (2.9)–(2.12) are used to obtain the rectilinear distances between centers of departments. Similar constraints are used by Sherali et al. (2003). Constraints (2.13)–(2.16) are used to control the orientations of the departments. Similar constraints are used in Dunker et al. (2005). Constraints (2.17)–(2.22) ensure that the department has the same values of length, width, and center coordinates in any two consecutive periods in which the department is not rearranged. These constraints are similar to constraints in Dunker et al. (2005), except that the constraints in Dunker et al. (2005) ensure that the departments are not rearranged, by forcing the centers and orientations of not rearranged departments to be the same, versus the center coordinates and the widths of departments, as in constraints (2.17)–(2.22). In addition the constraints in Dunker et al. (2005) ensure that the input/output stations of not rearranged departments are not rearranged (recall, the input/output stations in this paper are located at the centers of departments). Last, the restrictions on the variables are given in constraints (2.23). The MILP formulation can be used to solve small problem instances only. Therefore, heuristic techniques, which obtain good, but not necessarily optimal solutions for larger problem instances are presented next.

3. Solution Technique for the DFLP

The solution technique for the DFLP under assumptions (1)–(5) consists of the following three stages.

- (1) Use flow data to determine the order in which departments are to be selected for placement on the plant floor (selection procedure).
- (2) Use a placement procedure to place departments on the plant floor. The output is a layout plan and its cost.
- (3) Use tabu search to improve the layout plan obtained at stage 2.

As a result, the major components of the proposed solution technique for the DFLP are the solution representation, the selection procedure, the placement procedure, and the tabu search heuristic (improvement procedure). These components as well as other components embedded within the major components (e.g., descent heuristic embedded within the tabu search heuristic) are discussed in detail. It is important to note that the boundary search technique consists of stages (1) and (2).

3.1. Solution representation

For the proposed heuristic, the DFLP solution can be represented as a vector of department period pairs as follows:

$$\pi = \{(i_1, t_1), (i_2, t_2), \dots, (i_{NT}, t_{NT})\}.$$

The first department selected for placement is department i_1 in period t_1 . If department pair (i_k, t_k) precedes (i_r, t_r) in this vector (i.e., $k < r$), then department i_k in period t_k is selected for placement before department i_r in period t_r , where $k, r \in \{1, 2, \dots, NT\}$.

3.2. Boundary search heuristic

As previously stated, the boundary search heuristic (BSH) consists of a selection and placement procedure. First, the selection procedure constructs a solution π in such a way, that the following two conditions are satisfied:

$$(1) \sum_{t=1}^T \sum_{d \neq i}^N F'_{tid} \geq \sum_{t=1}^T \sum_{d \neq j}^N F'_{tjd}$$

$$(2) \sum_{d \neq i}^N F'_{t_1 id} \geq \sum_{d \neq i}^N F'_{t_2 id}$$

In other words, the departments, which have higher cumulative flow with other departments over all periods, are placed in vector π first according to condition (1). Precisely stated, in condition (1), each department i is placed in all periods in vector π and only then some other department j (with the next largest cumulative flow) is placed in all periods. This process is repeated until all departments have been placed in vector π . However, the order of the periods in which department i is placed in vector π is determined by the flow of department i with all other departments in each of these periods, according to condition (2). In other words, since (in condition 2) the total flow between department i and the other departments are higher in period t_1 than in period t_2 , department period pair (i, t_1) is placed in vector π before (i, t_2) . For the special case, $T = 1$, an initial solution can be obtained for this SFLP by using condition (1) without index t . Welgama and Gibson (1993) and Imam and Mir (1998) presented a similar, but slightly different, selection procedure for the SFLP. In Welgama and Gibson (1993), the first department to be placed is the department which interacts with the maximum number of departments. The next department to be selected for placement is the department having the greatest flow with already placed departments. The subsequent departments are placed using the same rule. Imam and Mir (1998) use the same selection procedure as Welgama and Gibson (1993), except that the first department to be placed is the department having the highest flow with all other departments.

Once an initial solution π is constructed using the above selection procedure, the following placement procedure is used to assign departments to locations on the plant floor for each period. First, department i_1 in period t_1 is placed at the center of the plant floor. Second, department i_2 in period t_2 is either placed at the center of the plant floor if $t_1 \neq t_2$ or placed on the boundary of department i_1 in period t_1 if $t_1 = t_2$. For the latter case, the best location on the boundary is selected for the location of department i_2 with respect to minimizing the sum of material handling costs. After assigning several departments to locations on the plant floor, department i_k in period t_k may be assigned to either the best location on the boundary of already placed departments or the location of department i_k in period $t_k - 1$ or $t_k + 1$. If rearrangement cost for department i_k is relatively low with respect to its material handling cost, it will be the former case as opposed to the latter. It will be the latter, if rearrangement cost for department i_k is relatively high and department i_k has been assigned to a location in period $t_k - 1$ or $t_k + 1$. Nevertheless, the best location is selected such that the sum of material handling and rearrangement costs is minimized. The process continues until all departments in each period have been assigned to locations on the plant floor. Details of the major components of the placement procedure are given below.

To determine potential locations available for department i_k in period t_k , BSH uses four vectors, which identify the feasible boundaries of already placed departments. More specifically, these vectors are ordered special structures of four types of boundary segments:

$hor_segms_d_t, hor_segms_u_t$ = vectors, the elements of which are vectors themselves, storing the coordinates of horizontal boundary segments in period t facing downward and upward, respectively.
 $vert_segms_l_t, vert_segms_r_t$ = vectors, the elements of which are vectors themselves, storing the coordinates of vertical boundary segments in period t facing leftward and rightward, respectively.

Each horizontal or vertical segment s used in these vectors has the form $s = \langle c, c_1, c_2 \rangle$. If segment s is horizontal then c is the Y-coordinate of the segment endpoints, and c_1 and c_2 are the X-coordinates of the segment endpoints. In a similar manner, if segment s is vertical then c is the X-coordinate of the segment endpoints, and c_1 and c_2 are the Y-coordinates of the segment endpoints. Every time some department i is placed in the layout in period t , BSH ensures that the vectors $hor_segms_d_t, hor_segms_u_t, vert_segms_l_t,$ and $vert_segms_r_t$ are modified such that the following two conditions are satisfied. Note: these conditions are demonstrated below considering only $vert_segms_r_t$; however, the conditions for vectors $hor_segms_d_t, hor_segms_u_t, vert_segms_l_t$ are exactly the same.

- All the segments $\langle c, c_1, c_2 \rangle$ in $vert_segms_r_{tq}$ have the same value of c components, and are ordered in ascending order based on the value of the c_1 component, where $q = 1, \dots, |vert_segms_r_t|$. For example, if the partially constructed layout in period 2 is as defined in Fig. 1, then $q = 1, \dots, 6$ (cardinality of $vert_segms_r_2$ is 6)

$$vert_segms_r_2 = \{ [vert_segms_r_{21}], [vert_segms_r_{22}], \dots, [vert_segms_r_{26}] \},$$

$$vert_segms_r_2 = \{ [(i, h), (e, d)], [(c, b)], [(g, f)], [(v, u)], [(r, q), (n, m)], [(t, s), (p, o)] \},$$

$$vert_segms_r_2 = \{ [\langle 2.5, 3.5, 6 \rangle, \langle 2.5, 7, 9.5 \rangle, \langle 3, 9.5, 11.5 \rangle, \dots, \langle 8.5, 1.5, 3.5 \rangle, \langle 8.5, 6.5, 8.5 \rangle] \}.$$

- The segments $\langle c, c_1, c_2 \rangle$ in $vert_segms_r_{tq}$ have smaller value of c component than segments in $vert_segms_r_{tq+1}$, for $q = 1, \dots, |vert_segms_r_t| - 1$. For example, the value of the c component of segments in $vert_segms_r_{21}$ is 2.5, and the value of the c component of segments in $vert_segms_r_{22}$ is 3.

These conditions allow BSH (specifically, a binary search algorithm embedded within BSH) to efficiently search for feasible areas along the boundary of already placed departments. Recall, based on the selection procedure given above, the first department being placed is assigned to a location for each period, first. Then the next department is assigned to a location for each period, and so on. Therefore, the selection procedure cannot generate the type of layout given in period 2 (i.e., separation of clusters of departments) in Fig. 1. However,

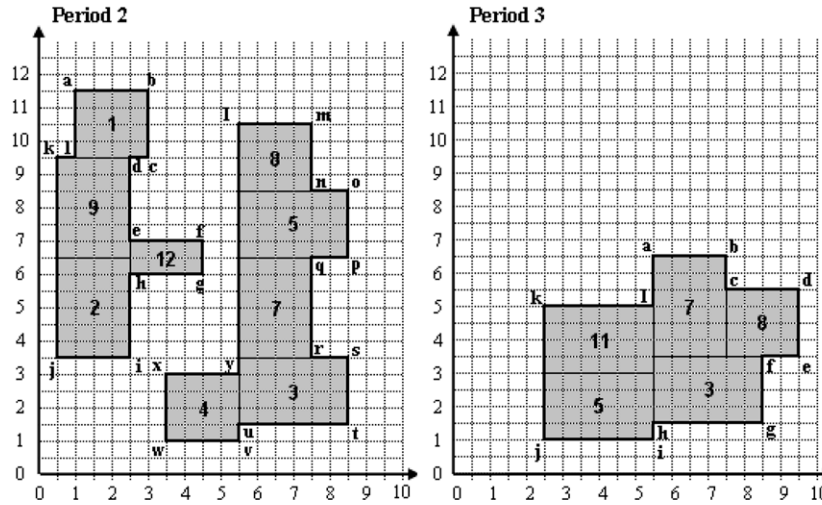


Fig. 1. Example of boundary segments in a partially constructed layout plan.

this example considers a more general case for some solution π (i.e., different selection procedure). Also, this type of layout in period 2 may occur either when rearrangement costs are high or when some departments are assigned to fixed locations. Nevertheless, this issue is considered in both BSH and the improvement procedure given below.

Once the feasible boundaries of already placed departments have been identified, the best location for some department i in period t is determined such that material handling cost is minimized. More specifically, at each iteration, BSH selects the next available department period pair (i_{curr}, t_{curr}) from π and places the department i_{curr} on the plant floor in period t_{curr} . Before placing the department, the coordinates of the center of gravity (i.e., approximately the most favorable location on the plant floor) for the department are calculated as follows:

$$cg_x = \frac{\sum_{\forall j \in \pi_per_{t_curr}} (F'_{t_curr, i_curr, j} x_{t_curr, j})}{\sum_{\forall j \in \pi_per_{t_curr}} F'_{t_curr, i_curr, j}},$$

$$cg_y = \frac{\sum_{\forall j \in \pi_per_{t_curr}} (F'_{t_curr, i_curr, j} y_{t_curr, j})}{\sum_{\forall j \in \pi_per_{t_curr}} F'_{t_curr, i_curr, j}},$$

where $\pi_per_t =$ ordered list of departments already placed in period t such that department i precedes j , if and only if (i, t) precedes (j, t) in π . The center of gravity (cg_x, cg_y) will be closer to the departments already placed in period t_{curr} , which have higher flow with department i_{curr} . More importantly, (cg_x, cg_y) may be within the boundary of an already placed department. As a result, the best available position with respect to minimizing material handling cost is a location either close to or at the center of gravity (cg_x, cg_y) . To find the best position for placing department i_{curr} in period t_{curr} , the binary search algorithm (see Reynolds (1979) for the description of the binary search algorithm) searches the vectors $hor_segms_d_t$, $hor_segms_u_t$, $vert_segms_l_t$, or $vert_segms_r_t$, as well as the vectors contained in these vectors to quickly identify rectangular area(s) (i.e., feasible areas) for each department orientation (i.e., horizontal and vertical). Within these feasible areas, the department being placed, i_{curr} , may be moved along the corresponding segments without overlapping other departments. The locations within the feasible areas closest to the center of gravity are called candidate locations. Each candidate location (x_cand, y_cand) , for each department orientation, is evaluated with respect to material handling cost. More exactly, the material handling cost of placing department i_{curr} at location (x_cand, y_cand) in period t_{curr} (i.e., $flow(x_cand, y_cand)$) is obtained for each candidate location as follows:

$$flow(x_cand, y_cand) = \sum_{\forall j \in \pi_per_{t_curr}} F'_{t_curr, i_curr, j} (|x_cand - x_{t_curr, j}| + |y_cand - y_{t_curr, j}|),$$

where $(x_{ti}, y_{ti}) = x$ - and y -coordinate of the centroid (location) of department i in period t . Also, if department i in period t is horizontally (vertically) oriented, then the department length and width are $l_{ti} = Lng_{ti}$ and $w_{ti} = Shi_{ti}$ ($l_{ti} = Shi_{ti}$ and $w_{ti} = Lng_{ti}$), respectively. Infeasible layout plans may occur when the boundaries of the plant floor are too restrictive with respect to the total area required for all departments. If all candidate locations result in *plant floor infeasible* layout plans (i.e., a layout constructed for a period is not within the boundaries of the plant floor), then the candidate location with the lowest value of $flow(x_cand, y_cand)$ is selected. Otherwise, the location with the lowest value of $flow(x_cand, y_cand)$ among the locations, resulting in a *plant floor feasible* layout plan (i.e., layout for each period is within the boundaries of the plant floor), is selected. More precisely, to determine whether the candidate location will result in a plant floor feasible or infeasible layout, the following parameters are used: the minimum values of left most (*MinLeft*) and lower most (*MinLow*) coordinates as well as the maximum values of right most (*MaxRt*) and upper most (*MaxUp*) coordinates of already placed departments over all periods. These parameters are used to determine the horizontal and vertical span of the already placed departments. More specifically, the horizontal span is determined using *MinLeft* and *MaxRt*, and the vertical span is determined using *MinLow* and *MaxUp*. Given the horizontal and vertical span of the departments as well as the centroid and dimensions of the placed department corresponding to candidate location, the spanning parameters are recalculated. If the resulting horizontal span of departments is less or equal to the plant floor length (L), and the resulting vertical span of departments is less or equal to the plant floor width (W), then the resulting layout is *plant floor feasible*. Otherwise, it is a *plant floor infeasible* layout. Note, the layout plan may have departments with negative coordinates, and still be *plant floor feasible*. Some of the above concepts is illustrated on an example below.

As discussed earlier, the best location for some department i in period t may be along the boundary of already placed departments (as in previous paragraph). This occurs when either rearrangement cost of the department is relatively low with respect to its material handling cost or rearrangement cost is not considered. However, the best location for some department i in some period t may be at the location of department i in period $t - 1$ or $t + 1$, when rearrangement cost of department i is relatively high with respect to its material handling cost. As a result, the best position for some department i_{curr} in period t_{curr} , with respect to minimizing material handling cost, is compared to the location of department i_{curr} in period $t_{curr} - 1$ (if the position is not occupied and department i_{curr} has already been placed in period $t_{curr} - 1$) and the position of department i_{curr} in period $t_{curr} + 1$ (if the position is not occupied and department i_{curr} has already been placed in period $t_{curr} + 1$). When comparing any two locations from the resulting candidate locations, the preference is given first to the location, which will result in a *plant floor feasible* layout plan. If both locations result in a *plant floor feasible* layout plan or both locations result in a *plant floor infeasible* layout plan, then the sum of the material handling cost between department i_{curr} and all placed departments in period t_{curr} as well as the rearrangement cost of department i_{curr} in periods t_{curr} (which considers the cost of rearranging the location of department i_{curr} from period $t_{curr} - 1$ to t_{curr} , if $t_{curr} > 1$) and $t_{curr} + 1$ (which considers the cost of rearranging the location of department i_{curr} from period t_{curr} to $t_{curr} + 1$, if $t_{curr} < T$) is used as a comparison criteria. This cost is defined as $TC_{i_{curr}, t_{curr}}$. Obviously, if department i_{curr} has not been placed in period $t_{curr} - 1$ and period $t_{curr} + 1$, then the rearrangement cost of department i_{curr} in periods t_{curr} and $t_{curr} + 1$ are zero. If the vector $\pi_{per_{t_{curr}}}$ is empty (i.e., no departments have been placed in period t_{curr}), and department i_{curr} has not been placed in neither period $t_{curr} - 1$ nor period $t_{curr} + 1$, then the department is centered on the rectangle, enclosing all the placed departments in all periods.

In Fig. 2, the example of finding the best location of department 11 in period 2 along the segment $(y,l) = \langle 5.5, 3, 10.5 \rangle$ is presented. The current layout corresponding to period 3 is shown in Fig. 1. Also, no departments have been placed in period 1. In this example, BSH searches for feasible rectangular areas within which department 11 can move parallel to segment (y,l) , without overlapping with other departments (i.e., dark gray regions). BSH tries both horizontal and vertical orientations of the department (demonstrated in Fig. 2a and b, respectively) and searches for feasible rectangular areas within the rectangle with corner points A, B, C, and D. Within each feasible area, BSH considers candidate locations for department 11 (e.g., rectangles outlined by dashed lines in Fig. 2), which is closest to the center of gravity for department 11 (e.g., point $(3.61, 6.14)$). For each candidate location $(cand_x, cand_y)$, BSH calculates the value *flow* $(cand_x, cand_y)$. BSH evaluates all the candidate locations along all the boundary segments. See Table 1 for evaluation of candidate locations along segment (y,l) . The first two candidate locations are illustrated in Fig. 2a, and candidate locations 3 and 4 are illustrated in Fig. 2b. Since candidate location 1 gives the lowest material handling cost of 157 and produces a feasible layout plan, it is selected as the best candidate location on the boundary of placed departments. However, the location of department 11 in period 3 is different from the best candidate location found in period 2; therefore, rearrangement cost of 50 is added such that $TC_{11,2} = 207$. Next, this location is compared to the location of department 11 in period 3. See Fig. 1 for the location of department 11 in period 3, and its orientation. Considering all possible positions, it is best to locate department 11 at the same location of the department in period 3, since $TC_{11,2} = 183$. Also, note that BSH identified two feasible areas along segment (y,l) for each of the orientations of department 11. Therefore, the number of feasible areas along each boundary segment may be either greater or equal one or zero (i.e., may not find any feasible areas along some segments).

The steps of the BSH are outlined as follows.

Stage 1: Selection Procedure

- Step 1:* Initialize π (using the selection procedure), total cost of solution π (i.e., $TC(\pi) = 0$), and feasible status of solution π (i.e., $feas_st = 1$). If $feas_st = 1$, then the current layout plan is plant floor feasible, and zero otherwise. Set $k = 1$.

Stage 2: Placement Procedure

- Step 2:* Set the values of i_{curr} and t_{curr} as the k th department period pair in π .

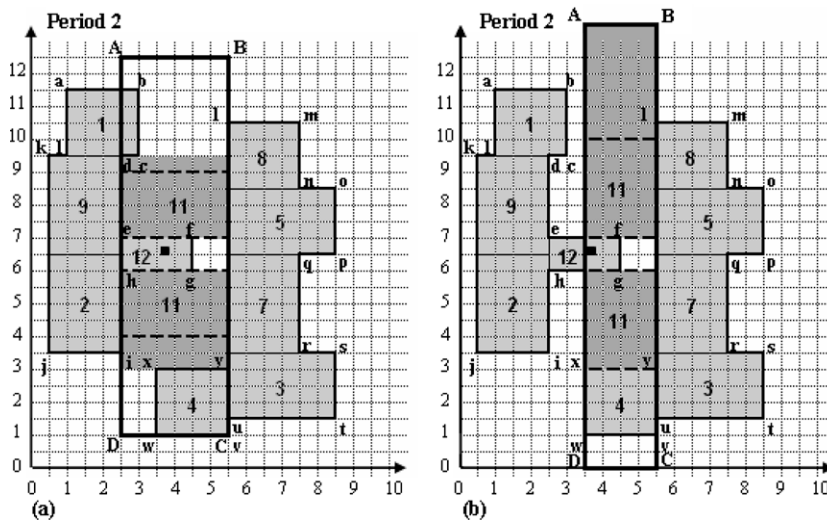


Fig. 2. Candidate positions for department 11 in period 2 along vertical segment (y,l) : (a) for horizontal orientation of department; (b) for vertical orientation of department.

Table 1Material handling cost of candidate positions of department 11 along vertical segment (l,y) .

	X	Y	flow(X,Y)
Candidate position 1	4.0	5.0	157
Candidate position 2	4.0	8.0	158
Candidate position 3	4.5	4.5	175
Candidate position 4	4.5	8.5	180

If $|\pi_{per_{t_{curr}}}| = 0$ and department i_{curr} has not been placed in neither period $t_{curr} - 1$ nor $t_{curr} + 1$.

Determine orientation of department i_{curr} . If $L \geq W$, then department i_{curr} is horizontally oriented. Otherwise, it is vertically oriented.

Place department i_{curr} at the center of the plant floor in period t_{curr} and initialize the location and dimensions of the department (i.e., set the values $x_{t_{curr},i_{curr}}$, $y_{t_{curr},i_{curr}}$, $l_{t_{curr},i_{curr}}$, and $w_{t_{curr},i_{curr}}$ from the position of the department).

Else

Calculate the coordinates of the center of gravity (cg_x, cg_y) for department i_{curr} in period t_{curr} .

Find the best candidate location along all segments in $vert_segms_{l_{t_{curr}}}$, $vert_segms_{r_{t_{curr}}}$, $hor_segms_{d_{t_{curr}}}$, $hor_segms_{u_{t_{curr}}}$, considering both horizontal and vertical orientation of department i_{curr} .

Compare the best candidate position with the position of department i_{curr} in period $t_{curr} - 1$ and period $t_{curr} + 1$ (if department i_{curr} has been placed in period $t_{curr} - 1$ and period $t_{curr} + 1$, respectively).

Select the best position among these positions and initialize the values of $x_{t_{curr},i_{curr}}$, $y_{t_{curr},i_{curr}}$, $l_{t_{curr},i_{curr}}$, and $w_{t_{curr},i_{curr}}$ from the best position and orientation found.

Set $TC_{i_{curr},t_{curr}}$ from the best position found.

Step 3: Update the vectors $hor_segms_{d_{t_{curr}}}$, $hor_segms_{u_{t_{curr}}}$, $vert_segms_{l_{t_{curr}}}$, and $vert_segms_{r_{t_{curr}}}$ according to the position of department i_{curr} in period t_{curr} .

Set $TC(\pi) = TC(\pi) + TC_{i_{curr},t_{curr}}$.

Set $feas_st = 1$, if placing department i_{curr} at location found in period t_{curr} will result in plant floor feasible layout plan, and 0 otherwise.

Add i_{curr} to vector $\pi_{per_{t_{curr}}}$.

If $k < NT$, then set $k = k + 1$, and go to Step 2.

Else

Output the values of $TC(\pi)$, x_{ti} , y_{ti} , l_{ti} , w_{ti} , for all $t = 1, \dots, T$ and $i = 1, \dots, N$, and terminate heuristic.

BSH presented above for the DFLP is a modification of the cluster boundary algorithm (CBA) presented by Imam and Mir (1998) for the SFLP. The main differences between CBA and BSH are as follows.

- CBA is used for solving only SFLPs, and departments are always placed along the boundary of already placed departments. In contrast, BSH can be used to solve both DFLPs and SFLPs. More importantly for the DFLP, departments are placed either along the boundary of already placed departments or at the location of the department in a preceding or succeeding period.
- CBA may be computationally costly when searching for feasible rectangular areas for placing departments; however, in BSH the center of gravity, ordered special structures of boundary segments, and binary search algorithm are used to find feasible areas very quickly.

CBA heuristic moves the department being placed along the boundary segments in a stepwise manner to determine the feasible positions for placing the department. At each step, the OFV should be evaluated. If the step size is too small, then the computational time will be extremely high. On the other hand, using a larger step size will decrease the solution space and may result in poor layouts. Also, it is unclear how CBA considers infeasible solutions (layouts) when plant floor dimensions are very constraining. Another construction heuristic for the SFLP, which constructs the layout by consecutively placing departments along the boundary of already placed departments is presented in Welgama and Gibson (1993). However, Welgama and Gibson (1993) consider only certain positions along each side of placed departments, when placing current/entering department. This decreases the search space and may result in poor solutions. In addition, many evaluated positions may be occupied by other placed departments, which may increase computational time (as in Imam and Mir, 1998).

3.3. Tabu search heuristic

Once an initial layout plan is obtained using BSH, a tabu search (TS) heuristic, called TS/BSH (stage 3), is used to improve the layout plan. Glover (1986) was the first to introduce the TS heuristic. Since then, TS has been successfully used to solve many types of combinatorial optimization problems. Skorin-Kapov (1990) was the first to use TS to solve the SFLP using the discrete representation of the layout. Similarly, Taillard (1991), Battiti and Tecchiolli (1994), Skorin-Kapov (1994), Chiang and Kouvelis (1996), and Chiang and Chiang (1998) presented effective TS heuristics for the discrete SFLP, to mention a few. However, Kaku and Mazzola (1997) presented the only TS heuristic for the discrete (equal-area) DFLP, known to the authors. Since the TS heuristic performed well on related problems, a TS heuristic is presented for the continuous DFLP presented in this paper.

It is important to note that the discrete DFLP is much harder to solve than the discrete SFLP, since the DFLP solution space is much larger (DFLP is a generalization of the SFLP as discussed earlier). More precisely, for the discrete SFLP with N equal-area departments there are $N!$ solutions (or layouts). In contrast, there are $(N!)^T$ solutions (or layout plans) for the discrete DFLP with T periods and N equal-area departments. Hence, the solution space for the discrete DFLP is exponentially larger than the solution space for the discrete SFLP. However, for π , there are $(N!)^T$ permutations. For each permutation, there are infinitely many layout plans, since there are infinitely many layouts (i.e., infinitely many locations for the departments on the continuous plant floor) for each period. Therefore, the solution space for the continuous

(unequal-area) DFLP is tremendously larger than the solution space for the discrete (equal-area) DFLP. As a result, there is a limited number of solution techniques available in the literature for the continuous DFLP, and there are no papers, known to the authors, which uses TS to solve this problem. The authors believe that this is the case, since it may be difficult to develop a simple TS heuristic to effectively search the solution space of the continuous DFLP. This will be discussed later. Nevertheless, a simple TS heuristic is presented below for the continuous DFLP presented in this paper.

In general, TS/BSH starts with an initial solution π (from the selection procedure) as well as its corresponding layout plan and total cost obtained from BSH (specifically from the placement procedure). The solution π is defined as the current solution. Next, TS/BSH explores the entire neighborhood of the current solution, $N(\pi)$. More specifically, all possible pairwise exchanges between department locations in each period are considered, and the best M exchanges (or moves) are selected and ranked (in ascending order) with respect to their estimated total costs (i.e., the cost in which the layout plan is estimated to have if the move is performed). Then, for the first move, which yields $\bar{\pi}$, a layout plan and its total cost ($TC(\bar{\pi})$) are obtained quickly using a modified BSH. If $TC(\bar{\pi}) < TC(\pi)$, then the neighboring solution becomes the current solution (i.e., set $\pi = \bar{\pi}$). Otherwise, consider second move, which gives, say $\bar{\pi}$, and repeat process until either $TC(\bar{\pi}) < TC(\pi)$ for one of the M exchanges or all M exchanges have been considered such that $TC(\bar{\pi}) \geq TC(\pi)$. For the latter case, the best solution with respect to total cost is selected as the current solution. The move used to produce the new current solution is defined as tabu for a certain number of iterations, called tabu duration. More precisely, a move that is tabu cannot be performed unless it satisfies the aspiration criterion (i.e., a move that is declared tabu can be performed if it gives a better estimated cost than the best layout plan found and have been tabu for a number of iterations). The tabu status and tabu duration of each move are maintained in the upper half of the tabu list structure $tabu[t][i][j]$, where $i < j$. Then, the neighborhood of the new current solution, π , is explored, and the process is repeated until a stopping criterion is met. That is, TS/BSH terminates after a certain amount of run time (Max_Time).

When exploring the neighborhood of solution π , $N(\pi)$, $[N(N - 1)/2]T$ exchanges (or moves) are considered if there are N departments and T periods. More exactly, at each TS/BSH iteration, $[N(N - 1)/2]T$ exchanges need to be performed and evaluated. Instead of performing each move to obtain the corresponding solution, and then using BSH to determine the layout plan and its total cost for each move, an estimated total cost is quickly obtained for each move. To do this, only the centroids of the exchanged departments will be swapped in the layout plan for solution π (ignoring the lengths and widths of the exchanged departments). Then, the total cost is updated, which yields an estimated total cost. Obviously, the estimated total cost may not be equivalent to the actual total cost of the resulting layout plan, since the actual positions of the exchanged departments as well as some other departments may be different when the actual move is performed, if the dimensions of the exchanged departments are different. Nevertheless, the best M exchanges are selected and ranked (in ascending order) with respect to the estimated total cost. As discussed above, a modified BSH is used to quickly obtain either an improved layout plan from solution $\bar{\pi}$ (i.e., $TC(\bar{\pi}) < TC(\pi)$) or the best layout plan (from $\bar{\pi}$) of the M layout plans, which is not an improvement over the current layout plan from solution π (i.e., $TC(\bar{\pi}) \geq TC(\pi)$). BSH is modified such that the displacement of not exchanged departments is minimized. More exactly, consider the move (t, u, v) used to produce solution $\bar{\pi}$, where the locations of departments u and v are exchanged in period t . In period t , not exchanged departments before the first exchanged department in solution $\bar{\pi}$ have the same locations as well as lengths and widths in the layout plan obtained for solution π . However, for each department i being placed (i.e., including the first exchanged department and the others that follow), high flow is given between department i and an artificial department (with zero length and width). If department i is the exchanged department u (v), then the artificial department is at the location of department u (v) in the layout plan obtained for solution π . Otherwise, the artificial department is at the location of department i in the layout plan obtained for solution π . Whatever the case may be, BSH (specifically the placement procedure) presented earlier is then used to place each department i . As a result, for solution $\bar{\pi}$ a layout plan and its total cost are obtained. For example, consider the current solution π as shown in Table 2a, and its corresponding layout plan in Fig. 3a. Then, move (2, 1, 6), corresponding to exchanging the locations of departments 1 and 6 in period 2, results in $\bar{\pi}$ and its corresponding layout plan, which are shown in Table 2b and Fig. 3b, respectively. Notice when placing any department period pair (i, t) , which precedes both department period pairs (1, 2) and (6, 2), the modified BSH places department i in period t at the same position as in the layout plan constructed for π (i.e., same x_{ti} , y_{ti} , l_{ti} , w_{ti}). This improves computational time, since the positions of such departments are known and are not calculated. When placing department period pairs 21 through 36 that are not exchanged, they are in the vicinity of their locations in the layout plan for solution π (i.e., the layout plan in Fig. 3a) by using artificial departments and BSH (i.e., placement procedure) as discussed above. Similarly, the modified BSH ensures that the exchanged department 1 (6) is in the vicinity of the location of department 6 (1) in period 2 in the layout plan for solution π .

Once modified BSH identifies a layout plan from move (t, u, v) out of the M moves, as the new current layout plan, the tabu list $tabu[t][i][j]$, where $i < j$, is updated. In other words, the entry $tabu[t][u][v]$ is set to the current iteration ($curr_ts_iter$). When this exchange

Table 2
Solution π : (a) before move (2,1,6) is performed; (b) after move (2,1,6) is performed.

Dept. Per. Pair	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Department(i)	8	8	8	2	2	2	11	11	11	9	5	5	9	4	7	3	1	3	3	7	1	6	6	6	12	12	12	10	10	10	1	5	4	7	4	9
Period(t)	1	3	2	1	3	2	1	3	2	3	1	3	2	3	3	3	1	1	2	1	2	1	2	3	1	2	3	2	3	1	3	2	2	2	1	1
(a)																																				
Dept. Per. Pair	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Department(i)	8	8	8	2	2	2	11	11	11	9	5	5	9	4	7	3	1	3	3	7	6	6	1	6	12	12	12	10	10	10	1	5	4	7	4	9
Period(t)	1	3	2	1	3	2	1	3	2	3	1	3	2	3	3	3	1	1	2	1	2	1	2	3	1	2	3	2	3	1	3	2	2	2	1	1
(b)																																				

is considered at a latter iteration, when determining the best M moves, the exchange is determined to be tabu by considering one of the following conditions.

- If the move (t, u, v) is estimated to produce a layout plan that is not better than the best layout plan found thus far, then the tabu duration is Ten_Len . If $curr_ts_iter - tabu[t][u][v] \leq Ten_Len$, then the exchange is tabu restricted.
- If the move (t, u, v) is estimated to produce a layout plan that is better than the best layout plan found thus far, then the tabu duration is $0.5Ten_Len$. If $curr_ts_iter - tabu[t][u][v] \leq 0.5Ten_Len$, then the exchange is tabu restricted. This considers the above aspiration criterion.

Considering the second condition, it is important to note that care should be taken when selecting a move that is declared tabu, since a move can repeatedly result in an estimated total cost, which is better than the actual total cost of the best solution (best layout plan), but the move may actually result in a worse total cost of the layout plan. Therefore, if this move is not tabu for some iterations, cycling may occur.

TS/BSH for the DFLP is outlined below.

- Step 1: Find initial solution π using selection procedure. Find its corresponding layout plan (i.e., values for $x_{ti}, y_{ti}, l_{ti}, w_{ti}$ for all $t = 1, \dots, T$ and $i = 1, \dots, N$), total cost (i.e., $TC(\pi)$), and feasible status (i.e., $feas_st$) using placement procedure. These are defined as the current solution, current layout plan, current total cost, and current feasible status. Also, set as the best solution (i.e., set $\pi^* = \pi$), best layout plan (i.e., set $x_{ti}^* = x_{ti}, y_{ti}^* = y_{ti}, l_{ti}^* = l_{ti}$, and $w_{ti}^* = w_{ti}$ for all t and i), best total cost (i.e., set $TC^*(\pi^*) = TC(\pi)$), and best feasible status (i.e., set $feas_st^* = feas_st$). Initialize tabu duration (i.e., Ten_Len), tabu list $tabu[t][i][j]$ for $i < j$ (i.e., set $tabu[t][i][j] = -Ten_Len$ for $i < j$), current iteration (i.e., set $curr_ts_iter = 0$), current time (i.e., $curr_time = 0$), and maximum run time (i.e., Max_Time).
- Step 2: Set $curr_ts_iter = curr_ts_iter + 1$. Explore $N(\pi)$ and find the best M nontabu moves with respect to the estimated total cost. From the M moves, find the most favorable move (t, u, v) using the modified BSH, which results in the most favorable solution $\bar{\pi}$ as well as its layout plan, total cost, and feasible status. These become the current solution (i.e., set $\pi = \bar{\pi}$), current layout plan (i.e., update values for $x_{ti}, y_{ti}, l_{ti}, w_{ti}$ for all $t = 1, \dots, T$ and $i = 1, \dots, N$), current total cost (i.e., set $TC(\pi) = TC(\bar{\pi})$), current feasible status (i.e., $feas_st$).
- Step 3: If $(feas_st > feas_st^*)$ or $(feas_st = feas_st^* \text{ and } TC(\pi) < TC^*(\pi^*))$, then set $\pi^* = \pi$, update best layout plan (i.e., set $x_{ti}^* = x_{ti}, y_{ti}^* = y_{ti}, l_{ti}^* = l_{ti}$, and $w_{ti}^* = w_{ti}$ for all t and i), $TC^*(\pi^*) = TC(\pi)$, and $feas_st^* = feas_st$.
- Step 4: If $curr_time < Max_Time$, then set $tabu[t][u][v] = curr_ts_iter$, and go to step 2. Else, terminate the heuristic and return π^* , value for all $x_{ti}^*, y_{ti}^*, l_{ti}^*$, and $w_{ti}^*, TC^*(\pi^*)$, and $feas_st^*$.

TS/BSH presented above is much more effective than a typical, simple, TS heuristic for searching the tremendously large solution space of the continuous DFLP because of the following.

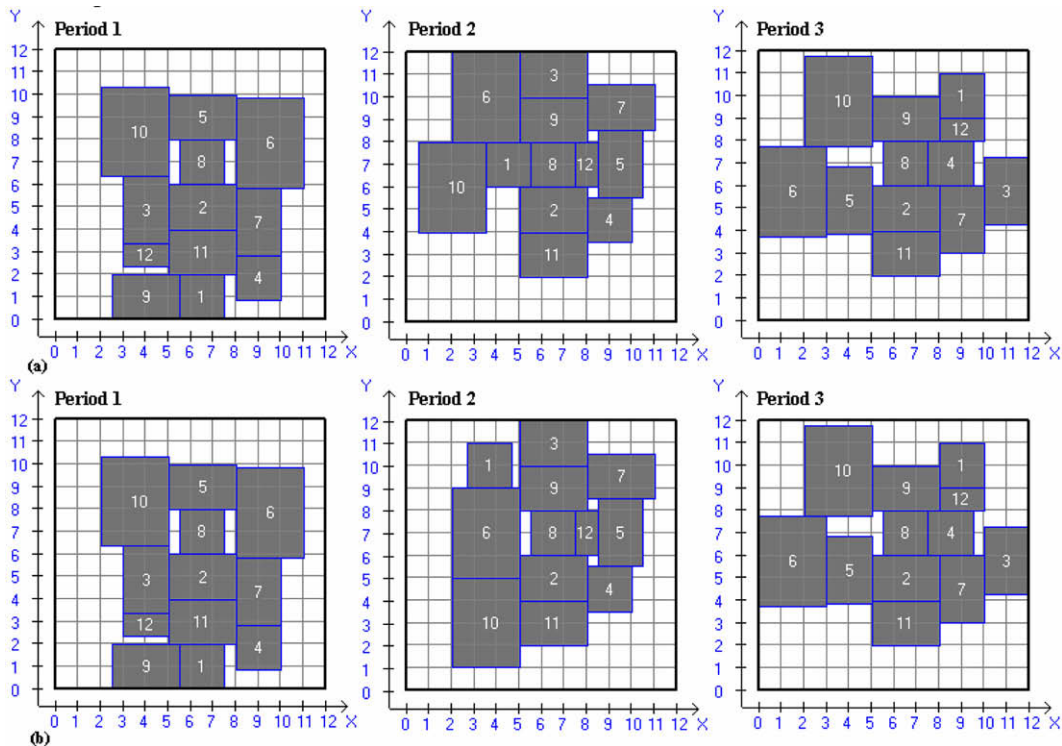


Fig. 3. Layout plans: (a) before move (2,1,6) is performed; (b) after move (2,1,6) is performed.

- In TS/BSH, when exploring the neighborhood of the current solution, an estimated total cost is obtained quickly for each neighbor, as opposed to the actual total cost obtained in a typical TS heuristic, which can be computationally costly for large problems. Only the actual total costs (layout plans) of M neighbors are obtained quickly using modified BSH.
- Since TS/BSH obtains the actual total costs of only the best M neighboring solutions, based on an estimated total cost, the neighboring solution selected may not be the best in the neighborhood, as opposed to a typical TS heuristic. Therefore, TS/BSH uses a descent-type strategy, which may allow for more diverse solutions, as opposed to a typical TS heuristic which uses a steepest-descent (less diverse) approach.

Thus, many more diverse neighborhoods may be explored in TS/BSH, as opposed to a typical simplistic TS heuristic. The authors believe the statement that “search diversification is possibly the most critical issue in the design of TS heuristics,” as stated by Gendreau (2003), and believes this is the reason why TS/BSH is so successful, especially for large problems, as illustrated in the next section.

Based on the solution techniques available in the continuous DFLP literature, the heuristics currently available obtain the total cost (layout plan) by solving a series of LPs or MILPs. In either case (especially the latter), it is computationally intractable for large DFLPs. However, TS/BSH is the only heuristic, known to the authors, which does not use an LP or MILP formulation to solve the DFLP. Instead, it uses modified BSH to quickly obtain the total cost (layout plan) for some of the neighbors of solution π . However, the major drawback of TS/BSH is that it may not out-perform a LP- or MILP-based heuristic for smaller problems, since BSH is not an exact approach (i.e., does not use LP or MILP formulation to obtain layout plan), as illustrated in the next section.

4. Computational results

Two sets of test problems are used in this paper in order to test the performances of the proposed heuristics. More specifically, data set I consists of two DFLP instances taken from Dunker et al. (2005). The first problem, P6, is a 6-department problem with six periods, and the second problem, P12, is a 12-department problem with four periods. These problems were first presented in Yang and Peters (1998), and were slightly modified in Dunker et al. (2005). Yang and Peters (1998) consider low and high rearrangement cost of 50 and 200, respectively, for each department. Dunker et al. (2005) solved the problems in Yang and Peters (1998), but used the rearrangement cost of 19 and 50 for problems P6 and P12, respectively, to allow for more department rearrangements in the solution. In addition, the problems in this data set use an initial layout in period 0 (i.e., the relative positions as well as lengths and widths of departments in initial layout prior to period 1 are specified); therefore, the rearrangement costs in period 1 should be considered.

Since there exist only two test problems in the literature for the DFLP presented in this paper (data set I), test problems from the SFLP literature (called data set II) are also used to test the performances of the proposed heuristics. Data set II consists of five SFLP instances. The first four problems are 20-, 50-, 100-, and 125-department benchmark problems (problems P20, P50, P100, and P125) from the commercial software VIP-PLANOPT 2006, which is used to solve SFLPs. For further details about VIP-PLANOPT 2006, see Engineering Optimization Software, 2005. The first test problem for data set II (20-department problem for the SFLP) was taken from VIP-PLANOPT 2006 and was generated in Imam and Mir (1993). The authors explained how the problem instance was generated and used it to test the performance of their heuristic. The second (50-department), third (100-department), and fourth (125-department) test problems for data set II were also taken from VIP-PLANOPT 2006. Mir and Imam (2001) used the 50- and 100-department problems to test their hybrid technique, and the authors explained how these two problems were generated. Imam and Mir (1998) used the 100- and 125-department problems to test their cluster boundary search technique. However, the authors did not explain how the test problems were generated. Since VIP-PLANOPT 2006 outperformed the above heuristics with respect to solution quality, the proposed heuristics are compared to the results obtained by VIP-PLANOPT 2006 for the first four problems of data set II for the SFLP. The last problem in this data set is a 62-department SFLP (P62), presented in Dunker et al. (2003). In data set II, there are no restrictions on the plant floor boundaries. Therefore, the length (L) and width (W) of the plant floor for each problem was obtained from the results obtained by the VIP-PLANOPT 2006 software and Dunker et al. (2003). Also, note that the Euclidean distance measure is used in problem P50, as opposed to the rectilinear distance measure as in the other problems. In all experiments, a Pentium IV 3.6 GHz PC was used to solve the problem instances in data sets I and II using the proposed heuristics. The proposed heuristics were coded using the C++ programming language.

The parameter values for TS/BSH were obtained by experimentation and are shown in Table 3. Recall, the parameters for TS/BSH are M , Ten_Len , and Max_Time . Consider the two extreme cases when setting the parameter M . First, if $M = 1$, the actual layout plan and total cost for only one move would be determined based on the move which gives the best estimated total cost. As a result, the smaller values of parameter M may result in a much more random search, which may produce inferior solutions (more diversification but much less intensification). Second, if $M = \lfloor N(N - 1)/2 \rfloor$ (i.e., the size of the neighborhood), the actual layout plan and total cost for all possible moves would be determined, as in a typical steepest-descent type of TS heuristic (deterministic search). The larger values of parameter M may result in better solutions at each iteration (more intensification and less diversification). However, the drawback of using a large M value result in

Table 3
Parameter settings for TS/BSH: (a) for data set I; (b) for data set II.

Problem	Max_Time (hours)	M	Ten_Len
<i>(a)</i>			
P6	1	10	.30
P12	5	20	.15
<i>(b)</i>			
P20	5	40	.15
P50	5	20	.15
P100	5	80	.15
P125	5	40	.15
P62	5	60	.15

less iterations (more computations per iteration) and less search diversification. Therefore, these issues need to be considered in order to make TS/BSH more effective. The parameter *Ten_Len* is determined by multiplying the size of the neighborhood by 0.15 or 0.30. Smaller values of *Ten_Len* tend to result in poor solutions, since TS/BSH spends too much time repeating the same moves, or the heuristic may get trapped in local optima (i.e., cycling). On the other hand, using too large of value for *Ten_Len* results in restricted solution space, and too many potentially good moves may be overlooked.

Table 4 summarizes the results obtained by BSH and TS/BSH for data set I. More precisely, the total cost of the layout plan generated (OFV) and the amount of computational time used to find the best solution (Runtime) for each problem are given for both heuristics. Recall, total maximum run time was given in Table 3 under *Max_Time*. Also in the last column, the percent deviation the OFV of TS/BSH is below the OFV of BSH (% Dev) for each problem is given. Table 5 compares the results obtained from TS/BSH for data set I with the best results available in the literature, which was obtained from the hybrid genetic algorithm (HGA) presented in Dunker et al. (2005). Each test problem was solved only once using TS/BSH. However, Dunker et al. (2005) indicated that their HGA was solved 20 times for each test problem on a Pentium IV 1.5 GHz PC. For problem P6, HGA obtained a better OFV than TS/BSH (OFV of TS/BSH is 2.16% above the best OFV of HGA). However, for problem P12, TS/BSH outperformed HGA (OFV of TS/BSH is 0.94% below best OFV of HGA) and obtained a better solution than HGA in 2.6 minutes. As previously discussed, HGA (an MILP-based heuristic) outperformed TS/BSH on the smaller problem instance which may be contributed to the fact that BSH is not an exact approach. Dunker et al. (2005) used a relaxed MILP formulation, in which the only binary variables are the variables used for orientations and rearrangement statuses of departments. The authors mentioned that the number of binary variables in the relaxed MILP increases linearly with increase in problem size, which could theoretically result in an exponential increase in computational time. On the other hand, TS/BSH is capable of solving larger problem instances in computationally reasonable time. For instance, problem P12 was solved in less than 1 millisecond by BSH, and modified BSH requires even less computation time; therefore, TS/BSH is capable of performing a large number of iterations in a relatively short amount of time, which may be the reason why TS/BSH outperformed HGA for the larger problem.

Table 4
Summary of results for BSH and TS/BSH for data set I.

Pb	Initial Sol. (BSH)		TS/BSH		% Dev
	OFV	Runtime (milliseconds)	OFV	Runtime (minutes)	
P6	6967.9	0	6648.3	27.77	4.81
P12	29779.6	0	26845.5	82.05	10.93

Table 5
Summary of results for TS/BSH and HGA for data set I: (a) for problem instance P6; (b) for problem instance P12.

	HGA	TS/BSH
<i>(a)</i>		
Average OFV	6569.0	6648.3
Worst OFV	6613.0	
Best OFV	6507.5	
Average runtime (minutes)	29.4	27.77
Fastest runtime (minutes)	8.4	
Longest runtime (minutes)	57.86	
<i>(b)</i>		
Average OFV	27748.0	26845.5
Worst OFV	28344.5	
Best OFV	27098.5	
Average runtime (minutes)	160	82.0
Fastest runtime (minutes)	54	
Longest runtime (minutes)	314	

Table 6
Summary of results for data set II.

Pb	VIP-PLANOPT		BSH		TS/BSH		Time TS/BSH found better solution	% Dev
	OFV	Runtime (seconds)	OFV	Runtime (seconds)	OFV	Runtime (seconds)		
<i>(a)</i>								
P20	1157.0 (1157.0)	0.44	1369.5	0.000	1151.4	10351.86	10351.86	0.49
P50	78224.7 (79,234)	7.00	80510.2	0.016	71291.4	7626.52	3.80	9.73
P100	538193.1 (547,990)	14.00	553041.9	0.047	496820.4	11397.19	33.33	8.33
P125	1,084,451 (1,096,800)	35.00	1,115,334	0.096	1,008,839	9250.28	41.91	7.49
<i>(b)</i>								
P62	3,939,362	4996	4,122,123	0.030	3,812,825	7304.05	286.06	3.32

Table 6 summarizes the results obtained by BSH, TS/BSH, VIP-PLANOPT 2006, and a coevolutionary algorithm (CA) presented by Dunker et al. (2003) for data set II, which consists of instances for the SFLP. More specifically, for Table 6a, OFV and runtime (i.e., amount of computational time used to find the best solution) for each problem are given for VIP-PLANOPT 2006, BSH, and TS/BSH. Also, the time it took TS/BSH to obtain a better OFV than VIP-PLANOPT 2006 and the percent deviation the OFV of TS/BSH is below the OFV of VIP-PLANOPT 2006 for each problem is given under % Dev column. For example, TS/BSH obtained the OFV of 1,008,839 for problem P125 in 9250.28 seconds in which this OFV is 7.49% below the OFV of VIP-PLANOPT 2006 (1,084,451). In fact, TS/BSH found a better OFV than VIP-PLANOPT 2006 in 42 seconds. Nevertheless, TS/BSH clearly outperformed VIP-PLANOPT 2006 with respect to solution quality where average percent deviation is 6.51. The runtimes shown in Table 6 corresponding to VIP-PLANOPT 2006 are reported at the website of the software and correspond to the solutions obtained from the older version of the software, which are given in parentheses under OFV. However, the OFVs shown in Table 6 for VIP-PLANOPT 2006 are the best results obtained by the most recent version of the software (i.e., slightly improved results over the results obtained from the old version of the software which are given in parentheses), and the runtimes corresponding to these solutions were not available to the authors. In Table 6b, OFV and runtime are given for problem P62 for CA, BSH, and TS/BSH. Dunker et al. (2003) indicated that their CA was solved 81 times for problem P62 on a Pentium IV 1.5 GHz PC. TS/BSH with one run outperformed CA with respect to solution quality and computational time. Note, for data set II, even though the lengths and widths of the plant floors used by TS/BSH were obtained from the results from VIP-PLANOPT 2006 software and Dunker et al. (2003), all the layouts obtained by TS/BSH are *plant floor feasible*.

5. Conclusion

The continuous DFLP with unequal-area and fixed department shapes is a computationally intractable problem, and there exists no exact techniques, which optimally solves the problem in polynomial time. Therefore, a construction and improvement type heuristic were developed to solve the problem in reasonable computational time. The construction heuristic (i.e., BSH) finds solutions quickly. The improvement heuristic (i.e., TS/BSH) uses BSH to generate layout plans, unlike other solution techniques in the literature, which uses either LP- or MILP-based heuristics to generate layout plans. TS/BSH heuristic performed well, especially for the large size problems, for a DFLP data set and a SFLP data set both taken from the literature. The following recommendations are given for future research:

- Develop a type of TS/BSH heuristic for a more general DFLP where variable shape departments as well as input and output stations are considered.
- Develop a more effective BSH-type heuristic, which performs better for smaller problems as well as larger problems.

References

- Balakrishnan, J., Cheng, C.H., 1998. Dynamic layout algorithms: A state-of-the-art survey. *Omega* 26 (4), 507–521.
- Battiti, R., Tecchioli, G., 1994. The reactive tabu search. *ORSA Journal on Computing* 6, 126–140.
- Chiang, W.-C., Chiang, C., 1998. Intelligent local search strategies for solving facility layout problems with the quadratic assignment problem formulation. *European Journal of Operational Research* 106, 457–488.
- Chiang, W.-C., Kouvelis, P., 1996. An improved tabu search heuristic for solving facility layout design problems. *International Journal of Production Research* 34 (9), 2565–2585.
- Dunker, T., Radons, G., Westkamper, E., 2003. A coevolutionary algorithm for a facility layout problem. *International Journal of Production Research* 41 (15), 3479–3500.
- Dunker, T., Radons, G., Westkamper, E., 2005. Combining evolutionary computation and dynamic programming for solving a dynamic facility layout problem. *European Journal of Operational Research* 165 (1), 55–69.
- Engineering Optimization Software, 2005. VIP-PLANOPT 2006, <<http://www.planopt.com/>>.
- Gendreau, M., 2003. An introduction to tabu search. In: Glover, F., Kochenberger, G.A. (Eds.), *Handbook of Metaheuristics*, Vol. 57. Springer, New York, pp. 37–54.
- Glover, F., 1986. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research* 13 (5), 533–549.
- Goetschalckx, M., 1992. An interactive layout heuristic based on hexagonal adjacency graphs. *European Journal of Operational Research* 63 (2), 304–321.
- Imam, M.H., Mir, M., 1993. Automated layout of facilities of unequal areas. *Computers and Industrial Engineering* 24 (3), 355–366.
- Imam, M.H., Mir, M., 1998. Cluster boundary search algorithm for building-block layout optimization. *Advances in Engineering Software* 29 (2), 165–173.
- Kaku, B.K., Mazzola, J.B., 1997. A tabu-search heuristic for the dynamic plant layout problem. *INFORMS Journal on Computing* 9 (4), 374–384.
- Kulturel-Konak, S., 2007. Approaches to uncertainties in facility layout problems: Perspectives at the beginning of the 21st century. *Journal of Intelligent Manufacturing* 18 (2), 273–284.
- Kusiak, A., Heragu, S.S., 1987. The facility layout problem. *European Journal of Operational Research* 29 (3), 229–251.
- 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., 1997. Preprocessing for static and dynamic facility layout problems. *International Journal of Production Research* 35 (4), 1095–1106.
- Meller, R.D., Gau, K.Y., 1996. The facility layout problem: Recent and emerging trends and perspectives. *Journal of Manufacturing Systems* 15 (5), 351–366.
- Mir, M., Imam, M.H., 2001. A hybrid optimization approach for layout design of unequal-area facilities. *Computers and Industrial Engineering* 39 (1), 49–63.
- Montreuil, B., Laforge, A., 1992. Dynamic layout design given a scenario tree of probable futures. *European Journal of Operational Research* 63 (2), 271–286.
- Montreuil, B., Venkatadri, U., 1991. Strategic interpolative design of dynamic manufacturing systems layouts. *Management Science* 37 (6), 682–694.
- Reynolds, J.C., 1979. Reasoning about arrays. *Programming Languages* 22 (5), 290–299.
- Rosenblatt, M.J., 1986. The dynamics of plant layout. *Management Science* 32 (1), 76–86.
- Sherali, H.D., Fraticelli, B.M.P., Meller, R.D., 2003. Enhanced model formulations for optimal facility layout. *Operations Research* 51 (4), 629–644.
- Shore, R.H., Tompkins, A., 1980. Flexible facilities design. *AIIE Transactions* 12 (2), 200–205.
- Skorin-Kapov, J., 1990. Tabu search applied to the quadratic assignment problem. *ORSA Journal on Computing* 2, 33–45.
- Skorin-Kapov, J., 1994. Extensions of a tabu search adaptation to the quadratic assignment problem. *Computers and Operations Research* 21 (8), 855–865.
- Taillard, E., 1991. Robust taboo search for the quadratic assignment problem. *Parallel Computing* 17, 443–455.
- Welgama, P.S., Gibson, P.R., 1993. A construction algorithm for the machine layout problem with fixed pick-up and drop-off points. *International Journal of Production Research* 31 (11), 2575–2589.
- Yang, T., Peters, B.A., 1998. Flexible machine layout design for dynamic and uncertain production environments. *European Journal of Operational Research* 108 (1), 49–64.