Expert Systems With Applications 91 (2018) 46–62

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Mathematical modeling and multi-start search simulated annealing for unequal-area facility layout problem

Maral Zafar Allahyari, Ahmed Azab*

Production & Operations Management Research Lab, Industrial and Manufacturing System Engineering Department, University of Windsor, Windsor, Canada 401 Sunset Ave., CEI Building, Windsor, N9B 3P4, Canada

ARTICLE INFO

Article history:
Received 12 November 2015
Revised 27 July 2017
Accepted 28 July 2017
Available online 29 July 2017

Keywords:
Facility layout problem
Unequal-area facilities
Mathematical modeling
Heuristics
Metaheuristics

ABSTRACT

In this paper, a mixed integer nonlinear programming model (MINLP) is formulated to allocate the position of a number of unequal-area rectangular facilities within the continuum of a planar plant site with a predetermined fixed area. Facilities have predetermined dimensions and are not orientation-free. A continuous approach to the problem is taken. Constraints are developed to eliminate the possible overlap between the different facilities. The model accommodates for aisles, whether vertical or horizontal, as well as blocks and preference locations, where no facilities are allowed to be placed. The problem seeks to minimize total material handling the cost. Four test cases including one from the local industry is used to justify the developed model. The problem at hand is computationally intractable; hence, a novel Simulated Annealing (SA) algorithm is developed to solve large instances of the problem. A unique heuristic algorithm is used for initialization. A multi-start search mechanism is implemented to increase the diversity and mitigate the chances of getting entrapped in local optima. For validation, a group of benchmark problems is being used.

1. Introduction

The facility layout problem (FLP) is the arrangement of a given number of non-equal sized facilities within a given space with the objective of improving machine utilization and part demand quality and reducing setup time, work-in-process inventory, and material handling cost. According to research findings, the total material handling cost is reduced on average by 30%–70% by applying optimized layout designs (Yaman, Gethin, & Clarke, 1993). Any changes to layout imply expenditures usually in the form of opportunity cost due to shut downs of production.

The different approaches developed to tackle the FLP problem can be classified into discrete and continuous. The discrete layout problem is the traditional approach to the FLP. The most popular mathematical model for this approach is the Quadratic Assignment Problem (QAP). The two main assumptions used in the QAP are: 1) all facilities have equal size and shape; 2) the candidate locations of facilities are known a priori. Presuming locations are predetermined ahead of time is a grave assumption that might not be practical and doable at all times. Moreover, by specifying locations ahead of time, the chances of arriving at a more efficient layout design are lessened; this is because the resulting possible solution space is shrunk by cutting on possible facilities' arrangements and fine tweaks that can prove of better quality in case a discrete approach is followed. Furthermore, this discrete approach to FLP is even more unsuitable when facilities have unequal-area and shape or if there are different clearances between the facilities. Formulating unequal-area facilities is the subject of this paper.

According to the literature, there are two different ways to tackle the Unequal-Area Facility Layout Problem (UA-FLP). Chronologically, the first approach attempts at dividing each facility into smaller size unit blocks, where the total area of those blocks is approximately equal to the area of the facility. There are two drawbacks to this method: firstly, the problem size grows as the total number of blocks increases; secondly, the exact shape of the facilities is ignored. The second approach is continuous layout where the exact shape and dimensions of facilities are considered; it assumes that all facilities can be placed anywhere in the allowed planar space continuum of the shop floor. Mathematical formulations for such continuous approach are usually mixed integer programming (MIP) with the objective of minimizing functions that are usually travel-based. Two set of constraints are usually used under this approach: one to eliminate overlap between facilities and another to ensure facilities stay within in-site boundaries. A continuous approach for FLP is adopted in this paper. The related work in the literature is reviewed in Section 2. A description of a

* Corresponding author.
E-mail addresses: zafaram@uwindsor.ca (M.Z. Allahyari), azab@uwindsor.ca, azab.ahmed@gmail.com (A. Azab).

http://dx.doi.org/10.1016/j.eswa.2017.07.049

developed mixed-integer non-linear programming model is mentioned in Section 3. The model has been linearized. Also, a real industry case study is demonstrated in this section to validate the developed mathematical model. Section 4 presents details of the developed metaheuristics for the FLP. To justify and validate the developed models, a group of benchmark problems from the literature is given and utilized in Section 5. Finally, conclusions are provided in Section 6.

2. Literature review

In academia and industry, the Facility Layout Problem (FLP) has attracted much attention. Comprehensive surveys have been conducted to plot the trends, models, and solution methods developed for the problem at hand (Arikaran, Jayabalan, & Senthilkumar, 2010; Drira, Pierreval, & Hajri-Gabouj, 2007; Kusiak & Heragu, 1987; Meller & Gau, 1996). The FLP, which can be defined as the problem of finding the most efficient arrangement of m indivisible departments with unequal area requirements within a facility or plant is originally being formulated by Armour and Buffa (1963). In general, there are two types of FLP: detailed layout design and block layout design. The detailed layout determines the exact position of facilities/department, designs the layout of sub-facilities within each facility/department including aisle structures and pickup and pickoff points. The block layout design, on the other hand, is limited only to determining the relative location of the main facilities/departments constituting the shop floor with no regard to the details, internal restrictions, and sub-facilities making the departments (Meller & Gau, 1996). As defined in the literature, the objective of the block layout design problem is the minimization of the material handling cost within a facility taking into consideration the following two sets of constraints: (a) departmental and floor area requirements and (b) departmental locational restrictions. Constraints set (a) ensures departments are located within the given perimeters of the plant, while constraints set (b) prevents departments from overlapping (Castillo, Westerlund, Emet, & Westerlund 2005; Chiang, 2001; Ketcham, Malstorm, & McRoberts, 1989; Kusiak & Heragu, 1987; Meller & Gau, 1996; Yamane et al., 1993).

Several solution methods for FLP are developed in the literature; these can be categorized as follows: (1) exact optimization approaches, (2) heuristics, and (3) metaheuristics. The authors find the taxonomy provided by Drira et al. (2007) to be quite comprehensive; few issues are noted, though. Firstly, the authors in their figure one summarizing their classification are mistakenly labeling the first category as "Manufacturing System". Under the category, they include the four different types of layouts: Process, Product, Fixed, and Cellular (Hybrid Process Product), though. This category should be better labeled simply "Layout"; the type of manufacturing system is obviously not exactly the layout adopted. The two, Layout and Manufacturing System, are related yet different. Under manufacturing system, we have job-shop, dedicated manufacturing system, Flexible Manufacturing System (FMS), Reconfigurable Manufacturing System (RMS), etc.

Secondly, for what they term Resolution approaches (for solution methods), they include Exact Methods, Heuristics, and Intelligence approaches. They miss an important class of solution methods, which is approximation algorithms. Thirdly, Drira et al. (2007) fail to include a complete list of all exact methods; branch and cut and cutting plane are missing. Fourthly, it is not clear what they mean by Intelligence Methods. If they mean Artificial Intelligence (AI) methods, then the search methods such as Breadth-first, Depth-first, Best-first, A*, etc. can be categorized as heuristics. Finally, in their classification of material handling, they include Automated Guided Vehicles (AGV), we believe that they could have better used a more general term such as Vehicles or Carriers which could include AGVs, Forklifts, and else.

For the exact solution methods, Meller and Gau (1996) state that the first MIP formulation for FLP is presented by Montreuil (1990). However, the developed MIP only can solve six or fewer departments optimally. Afterward, Heragu and Kusiak (1991) develop the special case of problem tackled by Montreuil's (1990), where the length, width, and orientation of facilities are known in advance. They represent two mathematical models: 1) linear continuous; 2) linear mixed integer. A heuristic method is presented to solve their models. Meller, Narayanan, and Pamela (1999) reformulate Montreuil's model (1990) by redefining the 0–1 binary variables. They propose improved surrogate perimeter constraints. This set of constraints enforces the area of the department to commit to the changes in its perimeter. Sherali, Fraticelli, and Meller (2003) develop an improved MIP model, which is quite similar to that developed by Meller et al. (1999). They claim the developed algorithms produce more accurate approximation in department areas by deriving a polyhedral outer approximation of the area requirements and branching priorities. Castillo et al. (2005) develop a mixed integer nonlinear programming model to find optimal solutions for the block layout design problem. Their proposed framework consists of an exact representation of the underlying nonconvex and hyperbolic area restrictions using a decision variable transformation and symmetric convex lower bounds, together with simple constraints to avoid symmetric layout solutions. Their computational results consistently yield optimal global solutions on several well-known test problems and benchmarks.

The main drawback of exact approaches is that they cannot solve large size instances of the FLP problem for optimality because of the intractable and combinatorial nature of the problem (Ripon, Glette, Khan, Hovin, & Torresen, 2013). Tate and Smith (1995) develop a Genetic Algorithm (GA) using an adaptive penalty function for the shape-constrained UA-FLP. They use the flexible structure bay concept to restrict the departmental shapes to a subset of possible shapes. Balakrishnan, Cheng, and Wong (2003) develop a user-friendly approach to solving FLP with unequal-area facilities. The approach was taken to model the unequal-area facilities is to divide the area of facilities into equal-sized squares; this way each facility occupies a different number of units on the grid depending on its area. However, the size of the problem grows significantly taking this approach. They develop a spaced-filling approach (SFC) and employ both simulated annealing and genetic algorithms metaheuristics. Wang, Hu, and Ku (2005) also tackle UA-FLP using a discrete approach. They apply SFC to handle the unevenness with the facilities size. With their objective function, they combine material flow, shape ratio, and area utilization. A genetic algorithm is used to solve the problem. Nordin, Zainuddin, Salim, and Rajeswari (2009) develop a mathematical model for the UA-FLP with fixed flow between departments to minimize the distance traveled. Constraints such as the restricted areas, reserved departmental locations, and the irregularity of the shapes of manufacturing layout are considered in their study. Finally, a hybridized GA-SA (Genetic Algorithm-Simulated Annealing) meta-heuristic method is developed to solve the problem. Komarudin and Wong (2010) develop an Ant System (AS) algorithm, which is a variant of Ant Colony Optimization, with slicing tree representation to solve the UA-FLP. Kulturel-Konak and Konak (2011) propose a hybrid particle swarm optimization (PSO) and local search approach for UA-FLP using a relaxed, flexible bay structure (FBS). In this study, facilities do not have fixed dimensions. Bozer and Wang (2012) present a heuristic procedure based on the graph-pair representation and simulated annealing to solve UA-FLP. GRAPH creates an initial layout by solving Linear Programming (LP) derived from a graph-pair. The initial solution is used as starting point for the SA-based search. Kulturel-Konak and Konak (2013) represent a hybrid GA/LP approach to solve UA-FLP, which uses a new coding scheme that is called location/shape representation. The proposed encoding system represents relative department positions based on their centroid and orientation. Finally, LP is used to determine the actual locations and shapes of the departments. Xiao, Yoonho, and Minseok (2013) develop an MIP model and a two-step heuristic for designing unequal area facilities with fixed shapes and input/output (I/O) points by introducing a zone concept. The zone concept is used to reduce the solution space, where a layout solution is first generated by using an interconnected zone algorithm and SA. That solution is then improved using reduced MIP. Kulturel-Konak and Konak (2014) develop a large-scale hybrid simulated annealing (LS-HAS) and MIP for the cyclic facility layout problem. Facilities are assumed unequal size and rectangular; however, their dimensions are not predetermined. Gonçalves and Resende (2015) develop a hybrid approach combining biased random-key genetic algorithm (BRKGA) and LP for the UA-FLP. BRKGA is used to determine the order of placement and dimensions of each facility, and LP is applied to fine-tune the solutions. Niroomand, Hadi-Vencheh, Şahin, and Vizvari (2015) propose a modified migrating birds optimization algorithm for closed-loop layout.

Bazargan-Lari and Kaebernick and team have written several papers about the design of cellular manufacturing (Bazargan-Lari, 1999; Bazargan-Lari & Kaebernick, 1997; Bazargan-Lari, Kaebernick, & Harraf, 2000; Kaebernick, Bazargan-Lari, & Arndt, 1996). Bazargan-Lari and Kaebernick (1997) present a continuous plane approach, where different constraints, such as cell boundaries, overlap elimination, closeness relationships, location restrictions/preferences, orientation constraints, and traveling distances are considered. A hybrid method, which combines nonlinear goal programming (NLGP) and SA, for the machine layout problem is presented. Imam and Mir (1993) introduce a heuristic algorithm (FLOAT) to place unequal-area rectangular facilities in a continuous plane by introducing their new concepts of controlled coverage and envelop blocks. For initialization, facilities are placed randomly within the allocated site area; however using the envelop blocks area for each facility, which is larger than the actual size of the facility. Envelop blocks are calculated by multiplying magnification factor with the facilities' actual dimensions. Afterward, during the heuristic iterations, the size of envelop blocks is gradually decreased by reducing the magnification factor. This procedure continues until the dimensions of envelopes match the exact dimensions of their corresponding facilities. One of the limitations of this approach is the fact that facilities are just allowed to move horizontally or vertically; no diagonal movement is allowed, however. Diagonal movement is addressed by Mir and Imam (2001), where they develop a hybrid model using SA to obtain a sub-optimal initial feasible solution that is to be improved afterward using a steepest descent solution method. Allahyari and Azab (2014) develop a bi-level mixed integer non-linear programming model to solve unequal-area Cellular Manufacturing System (CMS) facility layout problem at the Intra- and inter-cellular layout level sequentially. Firstly, the leader problem solves intracellular layout problem for one cell at a time. Secondly, the follower problem solves the intercellular layout problem for the whole production floor by locating cells on the whole shop floor. Allahyari and Azab (2015) present a heuristic construction algorithm to locate facilities radially in continuous planar site. They validate their developed mathematical modeling and heuristic using a case study from the cutting tool industry.

Table 1 shows a synthesis matrix, where the gaps and overlaps in the literature are being reported with a focus on continuous formulations and solution methods to the problem. Firstly, it can be concluded that only a dearth of the literature did address

Table 1
Synthesis matrix.

Paper	FLP Approach	Facilities			Orientation	Overlap constraint	Block/Aisle structures	Restricted area	Distance	Heuristic
		Dimension	Size	Shape						
Heragu and Kusiak (1991)	Continuous	Fixed	Unequal	Regular	Fixed	*		*	Rectilinear	Mathematical Modeling and Heuristic (Penalty Method)
Mir and Imam (2001)	Continuous	Fixed	Unequal	Regular	Fixed	*			Rectilinear	Hybrid(Heuristic and SA)
Imam and Mir (1993)	Continuous	Fixed	Unequal	Regular	Fixed	*			Rectilinear	Heuristic (Controlled coverage)
Bazargan-Lari and Kaebernick, (1997)	Continuous	Fixed	Unequal	Regular and Irregular	Free	*	*	*	Rectilinear	Mathematical Modeling and Hybrid(GP and SA)
Hassan et al. (1986)	Generalized Assignment Problem Formulation	Changeable	Unequal	Regular	Free				Rectilinear	Heuristic (Constructive algorithm)
Bazaraa (1975)	Quadratic Set covering Problem	Fixed	Unequal	Regular and Irregular	Fixed			*	Rectilinear	Heuristic
Van Camp et al. (1991)	Continuous	Fixed	Unequal	Regular	Free	*		*	Euclidean	Heuristic(Constructive algorithm- NLT)
Allahyari and Azab (2014)	Continuous	Fixed	Unequal	Regular	Fixed	*	*	*	Rectilinear	Mathematical Modeling
Allahyari and Azab (2015)	Continuous	Fixed	Unequal	Regular	Fixed	*	*	*	Rectilinear	Mathematical Modeling and Heuristic (Constructive algorithm)
This paper	Continuous	Fixed	Unequal	Regular	Fixed	*	*	*	Rectilinear	Mathematical Modeling and Metaheuristic (Constructive algorithm) and SA

the problem mathematically using full and proper formulations, and mathematical programming. In fact, most of the formulations in the literature have not been attempted and solved; it is often that metaheuristic algorithms are developed instead to find sub- or near-optimal solutions. Exceptions to this one are Heragu and Kusiak (1991) and Bazargan-Lari and Kaebernick (1997) who present two of the most important mathematical models in the literature. Also, Bazargan-Lari and Kaebernick's (1997) develop a comprehensive, formulation. Secondly, it has been noted that the simpler block layout problem has been tackled more often than the detailed layout problem. In other words, those set of constraints such as aisle and block/preference constraints have not been considered enough. Bazargan-Lari and Kaebernick (1997) develops comprehensive sets of constraints, which represent the detailed layout problem. Thirdly, it is important to highlight the fact that the assumptions taken by most of the papers included in the synthesis matrix as shown in Table 1 are different from the ones adopted in this paper. For example, the irregularity aspects of the shapes of the facilities are rarely considered in the literature. Bazargan-Lari and Kaebernick (1997) and Bazaraa (1975) consider this set of constraints, however. Another set of constraints that is relaxed and is critically playing a role constricting the solution space is the orientation constraint. Bazargan-Lari and Kaebernick (1997) relax this constraint and assume facilities are orientation free.

In this paper, a comprehensive mathematical model for pure continuous detailed FLP is presented, which is applicable for special cases of FLP such as CM layout design. Operation sequencing, parts' demand, aisle structures, block and preference locations are considered. The developed model can be used in manufacturing systems, where different carriers with different capacities are used to transport parts among facilities/department.

3. Mathematical modeling

In this paper, a Mixed Integer Nonlinear Programming (MINLP) mathematical model for the facility layout problem is developed. That model is later linearized in sub-Section 3.7. The problem is to layout the unequal-area facilities in a continual planar site. The site has a rectangular shape with a specified length *(L)* and width *(W)*, which has some location restrictions because of the aisle structure and/or blocks. Facilities are not allowed to be placed within their boundaries. As shown in Fig. 1, each facility has a rectangular shape, with its position represented by the coordinates of its centroid. The facilities' dimensions and orientation are predetermined. Product demand is being considered. Material handling cost is assumed to be a function of both the material handling and flow type being used. The material is transferred using different carriers, which have different capacities. The objective is to minimize total travel-distance material flow cost.

According to the taxonomy of the FLP problem provided by Drira et al. (2007), the generalized job-shop is the type of manufacturing system targeted in this paper since it is a general mathematical model and facility layout problem pursued in this paper. The layout for job-shops is cellular layout if the product mix (product variety) is high yet soft. Note that Cellular layout is a combination/hybridization of both of that of Process and Product. The layout evolution in our case is static. Regular facility shapes with fixed dimensions are considered. For material handling, vehicles have been used; it is important to note as explained in Section Two that Drira et al. (2007) taxonomy labels this AGV; we do believe though as explained that Vehicles is a more a generalized term. As for developed mathematical model, a mixed integer continuous formulation is formulated. Finally, for the solution method both mathematical programming and metaheuristics (as explained later in Section 4) are being employed.

The problem formulation is provided in the following Sections (3.1–3.6).

3.1. Assumptions

The problem is formulated under the following assumptions:

1. Facilities are not in the same size.
2. Facilities are considered as rigid blocks; *i.e.*, facilities are not orientation free.
3. Facilities must be located within a given area.
4. Facilities must not overlap with each other.
5. The demand for each part type is known a priori and is constant.

3.2. Sets

$M = \{1, 2, 3, \ldots, M\}$ Index set of facility type
$P = \{1, 2, 3, \ldots, P\}$ Index set of material type
$O_p = \{1, 2, 3, \ldots, o_p\}$ Index set of operations for material p

3.3. Parameters

L	Length of site
W	Width of site
Y_{AUpC}	Vertical coordinate of upper side of aisle
Y_{ALowC}	Vertical coordinate of lower side of aisle
X_{ALftC}	Horizontal coordinate of left side of aisle
X_{ARtC}	Horizontal coordinate of right side of aisle
l_i	Length of facility i
w_i	Width of facility i
D_j	Demand quantity for material j
C_j	Transfer unit cost for material j
U_{joi}	is 1 if operation o of material j can be done by facility i; otherwise it is, 0

3.4. Decision variables

x_i	Horizontal distance between center of facility i and vertical reference line
y_i	Vertical distance between center of facility i and horizontal reference line
Z_{iu}	1, if facility u is arranged in the same horizontal level as facility i, and 0 otherwise
Z_i	1, if facility i is arranged in out of aisle horizontal boundaries, and 0 otherwise
W_i	1, if facility i is arranged in out of aisle vertical boundaries, and 0 otherwise

3.5. Objective function

The objective represented by Eq. (1) minimizes the cost of total travel and material flow and takes into consideration the shape, size and geometric characteristics of facilities and the given allotted space. Rectilinear distance is used as the distance measurement system.

$$\text{Min} \sum_{j=1}^{P} \sum_{i=1}^{M} \sum_{\substack{u=1 \\ u \neq i}}^{M} (|x_i - x_u| + |y_i - y_u|)\ D_j C_j \tag{1}$$

If materials have a different sequence of processing, the objective function is given by Eq. (2):

$$\text{Min} \sum_{j=1}^{P} \sum_{o=1}^{o_p-1} \sum_{\substack{i,u=1 \\ i \neq u}}^{M} U_{joi}\ U_{jo+1u}(|x_i - x_u| + |y_i - y_u|) CA_j D_j \tag{2}$$

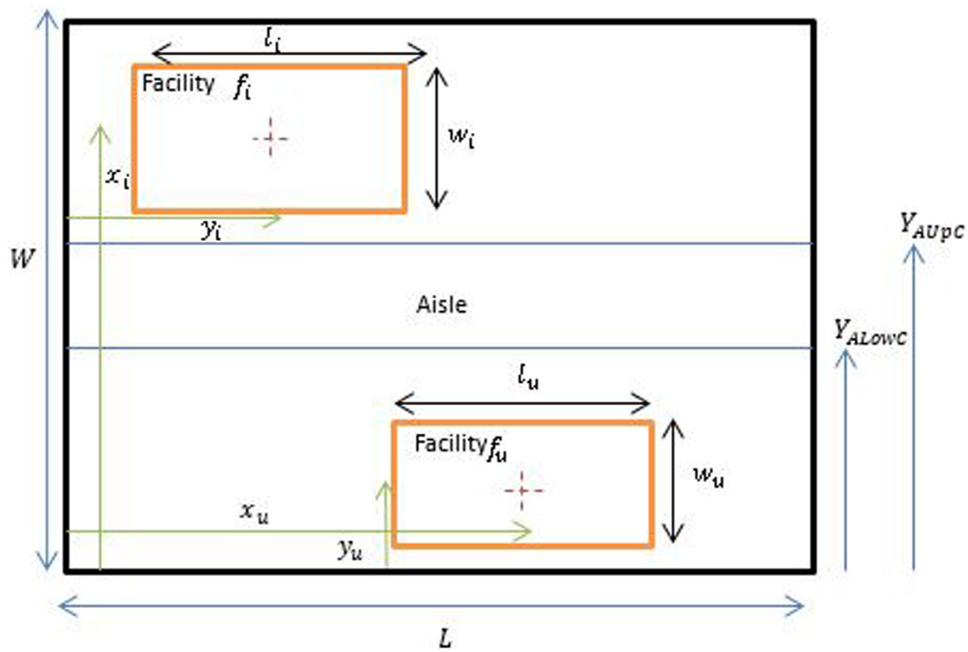

Fig. 1. Scheme of the site.

3.6. Constraints

A number of restrictions which are considered in the developed FLP model are: 1) Overlap elimination constraints; 2) Within-site boundaries constraints; 3) Aisle structure Constraints; 4) Carrier constraints; 5) Block and preference location constraints. Detailed explanations of these constraints are provided in the following sub-sections.

3.6.1. Overlap elimination constraints

No overlap is allowed between Facilities. The facility f_i and facility f_j do not overlap if either the absolute horizontal distance between their centroid, $|x_i - x_u|$ is greater than or equal to the half of the summation of their length $(l_i + l_u)/2$, or the absolute vertical distance between their centroid $|y_i - y_u|$ is greater than or equal to the half of the summation of their width $(w_i + w_u)/2$. These constraints are provided by Eqs. (3) and (4).

$$|x_i - x_u| \geq Z_{iu}(l_i + l_u)/2 \quad i, u = 1, .., M \tag{3}$$

$$|y_i - y_u| \geq (1 - Z_{iu})(w_i + w_u)/2 \quad i, u = 1, .., M \tag{4}$$

3.6.2. Within-site boundaries constraints

If the given space has exact boundaries, facilities are not permitted to be located outside of its perimeter. These constraints are expressed by Eqs. (5)–(8).

$$x_i + \frac{l_i}{2} \leq L \quad i = 1, .., M \tag{5}$$

$$x_i - \frac{l_i}{2} \geq 0 \quad i = 1, .., M \tag{6}$$

$$y_i + \frac{w_i}{2} \leq W \quad i = 1, .., M \tag{7}$$

$$y_i - \frac{w_i}{2} \geq 0 \quad i = 1, .., M \tag{8}$$

3.6.3. Aisle constraints

In the case of having aisles (whether vertical or horizontal), facilities are not allowed to be placed within the boundaries of the aisle. If the aisle structure is horizontal, either-or constraints are provided by Eqs. (9) and (10). To illustrate, facility f_i is not located within a horizontal aisle if either the upper boundary of the facility $(y_i + w_i/2)$ is less than or equal to the lower boundary of the aisle Y_{ALowC}, or if the lower boundary of the facility $(y_i - w_i/2)$ is greater than or equal to the upper boundary of the aisle Y_{AUpC}.

- ***Horizontal aisle constraint***

$$(y_i + w_i/2) - Y_{ALowC} \leq \ MZ_i \quad i = 1, .., M \tag{9}$$

$$Y_{AUpC} - (y_i - w_i/2) \leq M(1 - Z_i) \quad i = 1, .., M \tag{10}$$

Similarly, if the aisle has vertical structure, facility f_i is not located inside the aisle if either the right boundary of the facility $(x_i + l_i/2)$ is less than or equal to the left boundary of the aisle X_{ALftC}, or either the left boundary of the facility, $(x_i - l_i/2)$ is greater than or equal to the upper boundary of the aisle X_{ARtC}. These constraints are defined by Eqs. (11) and (12).

- ***Vertical aisle constraint***

$$(x_i + l_i/2) - X_{ALftC} \leq MW_i \quad i = 1, .., M \tag{11}$$

$$X_{ARtC} - (x_i - l_i/2) \leq M(1 - W_i) \quad i = 1, .., M \tag{12}$$

3.6.4. Block constraints

It is quite often that manufacturers cannot assign certain locations to any departments since they might be used to hold finished or raw material inventory or perhaps because of safety issues. In those cases, areas are assumed to be prohibited blocks with exactly given length, width, and coordinates- see Fig. 2. Eqs. (13) and (14) present the formulation for the block constraint.

$$|x_i - xblock_k| \geq Z'_{ik}(l_i + lblock_k)/2 \quad i = 1, .., M, k = 1, .., K \tag{13}$$

$$|y_i - yblock_k| \geq \left(1 - Z'_{ik}\right)(w_i + wblock_k)/2 \ \ i = 1, .., M, \ \ k = 1, .., K \tag{14}$$

where,

K Number of blocks

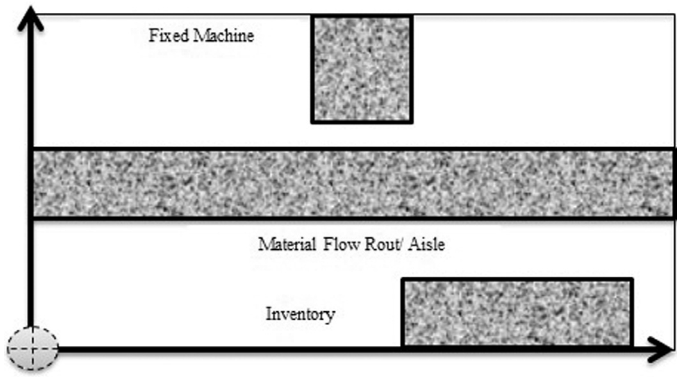

Fig. 2. The scheme of block constraints.

$xblock_k$ The horizontal coordinate of block k
$yblock_k$ The vertical coordinate of block k
$lblock_k$ The length of block k
$wblock_k$ The width of block k
Z'_{ik} 1 if facility i is arranged in the same horizontal level as block k, and 0 otherwise

3.6.5. Use of carriers for material handling

In most cases, materials are being transported by a carrier among the facilities. Hence, the frequency of trips is usually considered for material flows between facilities (Heragu and Kusiak, 1988).

Assuming:

v_{iu}^p Volume of material type p to be carried from facility i to facility u in a given time horizon
n_{iu} Number of different material types to be carried from facility i to facility u in a given time horizon
u^p Number of material type p to be carried in a single trip of the carrier

Based on the above notation by Heragu and Kusiak (1998), the total frequency of trips between facility i and facility u in a given time horizon $flow_{iu}$ is defined as:

$$flow_{iu} = \sum_{p=1}^{n_{iu}} \frac{v_{iu}^p}{u^p} \tag{15}$$

Where * is the smallest integer greater than or equal to *.

Therefore, the objective provided by Eq. 14 is to minimize the total distance traveled by materials between facilities.

$$\text{Min} \sum_{i=1}^{M} \sum_{\substack{u=1\\u\neq i}}^{M} (|x_i - x_u| + |y_i - y_u|)\ flow_{iu} \tag{16}$$

In this paper, Herague and Kusiak's (1988) formulation for machine layout in Flexible Manufacturing System (FMS) where material is moved within cells by a carrier (such as a robot or an AGV) is improved. The following practical considerations were taken into account when formulating the developed model; these were ignored by Herague and Kusiak's (1988):

1. The material is moved between facilities using different carriers.
2. Carriers have different capacity.
3. Costs associated with movements are a function of carrier-type and are directly proportional to distance traveled.

Hence, the following set and parameters are defined:

Set

$CR_j = \{1, 2, 3, \ldots, CR_j\}$ Index set of carrier type j

Parameters

v_{iu}^p Volume of material type p to be carried from facility i to facility u in a given time horizon
n_{iuj} Volume of different material types to be carried from facility i to facility u by carrier j in a given time horizon
u_j^p Volume of material type p to be carried in a single trip of the carrier j
C_j^{CR} Travel unit cost for carrier type j

Based on the above notation, the total frequency of trips between facility i and facility u $flow_{iuj}$ is being transported using carrier j in a given time horizon is defined in Eq. (15). The new objective function considering material being transported by carriers is provided in Eq. (16).

$$flow_{iuj} = \sum_{p=1}^{n_{iuj}} \frac{v_{iu}^p}{u_j^p} \tag{17}$$

$$\text{Min} \sum_{j=1}^{CR} \sum_{i=1}^{M} \sum_{\substack{u=1\\u\neq i}}^{M} (|x_i - x_u| + |y_i - y_u|)\ flow_{iuj} C_j^{CR} \tag{18}$$

The overlap elimination constraints (3.6.1), Within-site Boundaries Constraints (3.6.2), Aisle Constraints (3.6.3) are also applicable in case of carriers.

3.6.6. Non-negativity constraints

All decision variables are non-negative; Eq. (19) is the non-negativity constraint.

$$x_i,\ \ y_i \geq 0, Z_i, W_i, Z_{iu}, Z'_{ik} \text{ are binary} \quad i, u = 1, .., M, k = 1, .., K \tag{19}$$

3.7. Linearization

Since both overlap eliminations constraints and objective function are nonlinear. In order to linearize the model, the following variables are defined and given by Eqs. (20) and (21).

$$|x_i - x_u| = x_{iu}^+ - x_{iu}^- \tag{20}$$

$$|y_i - y_u| = y_{iu}^+ - y_{iu}^- \tag{21}$$

The two above terms are replaced by absolute terms (the modulus function) in the objective function Eq. (14)). Moreover, Eqs. (20) and ((21) are added to the sets of constraints. The linearized objective function of the problem is given by Eq. (22).

$$\text{Min} \sum_{i=1}^{M} \sum_{\substack{u=1\\u\neq i}}^{M} \left(\left(x_{iu}^+ - x_{iu}^-\right) + \left(y_{iu}^+ - y_{iu}^-\right)\right) \times flow_{iu} \tag{22}$$

To linearize the overlap elimination constraints, Eqs. (3) and (4), two auxiliary variables QX_{iu} and QX_{iu} are introduced. Constraints (23) to (26) substitute constraints (3) and (4).

$$(x_i - x_u) + M \times QX_{iu} \geq Z_{iu}(l_i + l_u)/2 \quad i, u = 1, .., M \tag{23}$$

$$(x_i - x_u) - M \times (1 - QX_{iu}) \leq (-Z_{iu})(l_i + l_u)/2 \quad i, u = 1, .., M \tag{24}$$

$$(y_i - y_u) + M \times QY_{iu} \geq (1 - Z_{iu})(w_i + w_u)/2 \quad i, u = 1, .., M \tag{25}$$

$$(y_i - y_u) - M \times (1 - QY_{iu}) \leq -(1 - Z_{iu})(w_i + w_u)/2 \quad i, u = 1, .., M \tag{26}$$

Table 2
Machine tools and cell descriptions.

ID	Machine	Cell	
		Name	Size
M1,M2	Blanchard (2)	Primary cell	35 × 25
M3	Double Disk (1)		
M4,M5, M6	Wendt (3)		
M7	Polish (1)		
M8, M9	Surface grinding (2)	Grinding cell	26 × 20
M11, M12	Swing fixture (2)		
M13	V-bottom (1)		
M10	Surface grinding	Diamond cell	30 × 20
M14, M15	Wire-cutting (2)		
M16	Laser M/C (1)		
M17	Brazing (1)		
M18	Ewag (1)		
M19	ETCH (1)	Final cell	30 × 20
ST1	Inspection (1)		
ST2	Wash (1)		
ST3	Shipment and Packing (1)		

*The number in brackets indicates the number of copy of the corresponding machine tool.

3.8. Test case one

Allahyari and Azab (2015) validate their developed mixed integer nonlinear mathematical modeling and heuristic construction algorithm using a case study from a local metalworking tools industry. The current layout adopted in the case study is a process one. Five different types of a family of cutting insert tools are produced. Each part has specific monthly demand. Machines tools used for manufacturing are in different size and shape with identical copies on the floor. Table 2 represents the characteristics of machine tools and cells. The demand is being shared among the different copies of those machine tools. Each type of part families has a specific operation sequence which is shown in Table 3. As demonstrated in Table 3, it is obvious that the number of operations performed on each insert tool is large enough; hence, the amount of travel taking place every day on the production floor is quite significant.

Since the company's business in the few years has grown quite significantly, their product mix has changed and hence, the required processing as well. This in turn has forced the company to acquire new machine tools and work centers. These new machines and centers had been located without any consideration of the impact of the sequence of operations and associated material flow. The following issues have been identified in the authors' communication with the company:

1. Increase in unnecessary travel between machine tools and work centers due to the inappropriate process layout adopted.
2. Increase in material handling cost due to unnecessary travel.
3. The increase in waiting time and queue length, and hence, delays in customer due dates and increased opportunity cost.
4. The increase in scrap and reworked products.
5. Conflicts in production planning.

Upon careful examination of the company's needs and issues, the cellular layout has been chosen and implemented. The layout problem for a cellular manufacturing system is two-folded: inter- and intra-cellular (Allahyari & Azab, 2014 and 2015). Only the inter-cellular layout problem is being solved using the generalized.

The inter-cellular layout problem is solved using the developed nonlinear model which lays out the different cells on the shop floor; *i.e.*, a facility in this case is a manufacturing cell. The problem becomes to optimally determine the position of all manufacturing cells on the overall site. An upper limit for the length and width of each cell is defined by plant manager as shown in Table 2. At this

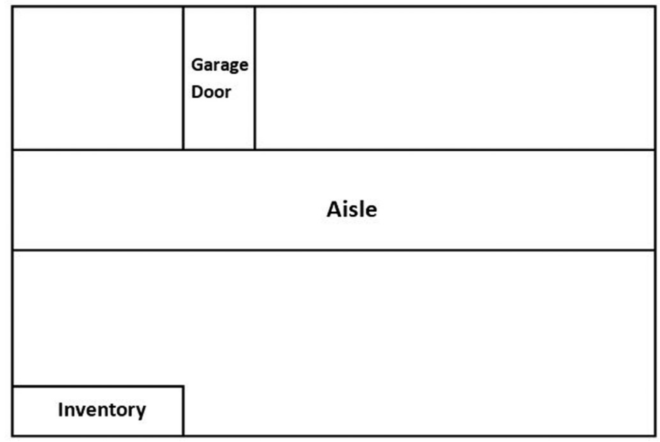

Fig. 3. Scheme of company's shop floor.

inter-cellular level, the travel is defined as that between the different cells on the whole production floor. Since the Cell Formation is done in advance, machine assignment is already predetermined; *i.e.*, operations of part *j* processed in cell *k* are known as well ahead of time.

The company's shop floor does not have complete rectangular shape. There is an inventory, a horizontal aisle for material flow and transportation in the middle, and a garage door within the boundaries of the shop floor. Fig. 3 shows the empty unallocated overall sketch/layout of the shop floor.

The inter-cell costs for Dog Bone, S shape, Triangular, Top Notch, and Diamond families are ¢12, ¢12, ¢18, ¢15, and ¢15, respectively. The objective function values (Material Handling Costs) for the intercellular problem for both linear and nonlinear methods are provided in Table 4. The scheme of both linear and non-linear models for intercellular is shown in Fig. 4. Clearly, in comparison to the nonlinear model, the linear model results are superior since they are exactly optimal. Both models have considered the block and preference constraints, as well as the aisle constraints. An Xpress MP algebraic modeling language and its nonlinear solvers "mmxslp" and "mmxnlp" have been used to solve the developed model.

4. Metaheuristics (Simulated annealing)

The facility layout problem belongs to the class of NP- hard problems (Castillo et al., 2005; Heragu & Kusiak, 1991). Hence the need of designing heuristic algorithms are raised. Simulated Annealing (SA) is a stochastic neighborhood search technique which was initially developed by Metropolis et al. (1953) and applied to combinatorial problems by Kirkpatrick, Geltt, and Vecchi (1983). SA is based on the physics and theories put to explain the annealing heat treatment process (Golden & Skiscim, 1986). SA borrows on the concept of cooling and starts the search at high temperature, where chances of accepting a lower-quality neighborhood solution are relatively high (enabling hill-climbing). The temperature is gradually decreased with outer-loop iterations. Search to find the lower energy (better solution) continues until the system reaches equilibrium (near- or sub-optimal). Based on the literature reviewed SA is widely applied to the FLP (Koulamas, Antony, & Jaen, 1994; Mavridou & Pardalos, 1997; Wilhelm & Ward, 1987, Bazargan et al., 1997, Heragu & Alfa, 1992). The core of SA algorithm is Metropolis algorithm (Kirkpatrick et al., 1983; Press, Teukolsky, Vetterling, & Flanner, 2007), which allows uphill moves sometimes; Metropolis algorithm has four main elements (Kirkpatrick et al., 1983; Press et al., 2007). The explanation of SA parameters and settings are explained in coming sub-sections. For development, C++

Table 3
Parts' operations sequence.

Part	Demand	Operation sequence									
Dog Bone	1200	M3	M11, M12	M8, M9	M13	M19	ST2	ST1	ST3		
S Shape	900	M3	M4, M5, M6	M13	M11, M12	M8, M9	M19	ST2	ST1	ST3	
Triangular	300	M3	M7	M4, M5, M6	M13	M8, M9	M19	ST2	ST1	ST3	
Top Notch	300	M3	M4, M5, M6	M8, M9	M19	ST2	ST1	ST3			
Diamond (1)	600	M1, M2	M4, M5, M6	M10	M14, M15	M17	M18	M19	ST2	ST1	ST3
Diamond (2)	600	M1, M2	M4, M5, M6	M10	M14, M15	M17	M16	M19	ST2	ST1	ST3
Diamond (3)	200	M17	M18	M19	ST2	ST1	ST3				

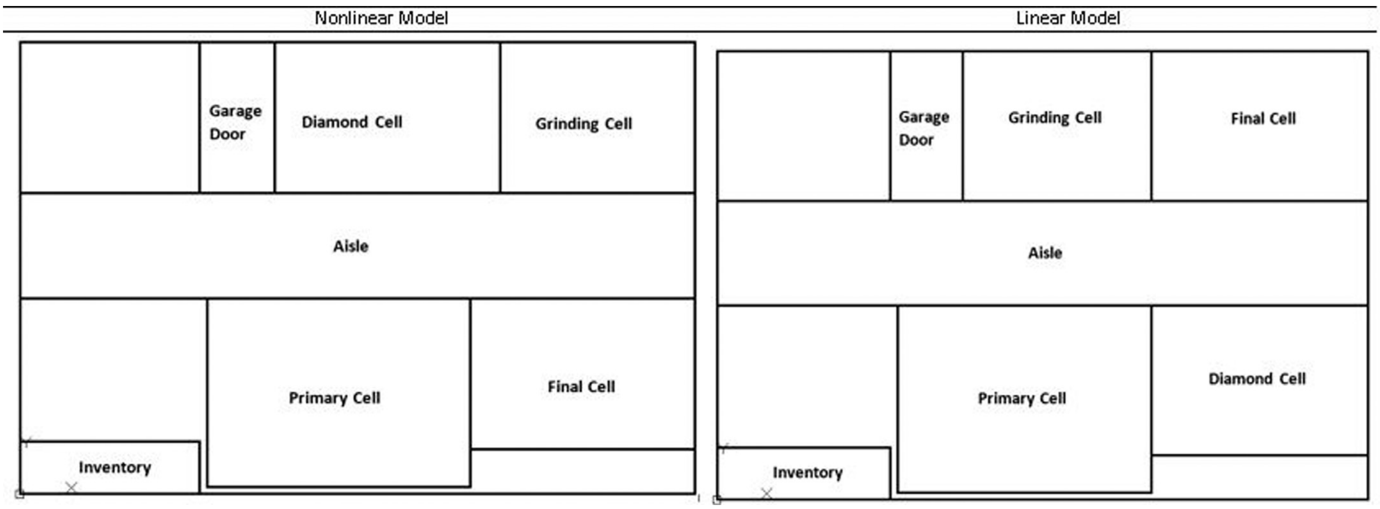

Fig. 4. Inter-cell layout designs- non-linear and linear models.

Table 4
Inter-cell layout-linear and nonlinear model.

Cells	Centroid			
	Linear model		Nonlinear model	
	X	*Y*	*X*	*Y*
Primary	42.5	13.5	42.5	13.5
Grinding	47	50	77	50
Diamond	75	16	49	50
Final	75	50	75	16
Blocks	***X***	***Y***	**Length**	**Width**
Garage door	29	50	10	20
Inventory	11.37	3.5	6.5	23
Aisle	45	32.5	90	60
Shop dimension: 90×60	**MHC:** $4213.90		**MHC:** $6467.70	

procedural programming language and the Microsoft Visual Studio Application Development Environment are used. The pseudo code of developed SA is represented in Appendix A.

4.1. Initialization

A unique heuristic is used to generate a feasible initial solution for SA algorithm (Allahyari & Azab, 2015). The explanation of the developed heuristic is provided in Section 4.1.1.

4.1.1. Initialization heuristic

The mechanics of the developed algorithm are very different from any of the available heuristics in the literature. The whole idea behind our algorithm is to place facilities radially along vectors $\overrightarrow{r_f}$ that are originated from the centroid of the space considered, where all facilities are to be placed as shown in Fig. 8. The radial vectors along which all facilities are to be placed are distant radially by an angle $\theta = \frac{360^0}{M}$.

-Heuristic algorithms starts with dividing the given area into four equal-size quadrants; *i.e.,* $Q_1, Q_2, Q_3, and\ Q_4$. Afterwards, all facilities are placed on top of each other in the middle of the given area. The developed heuristic algorithm consists of the two nested loops.

- **Outer loop**

For each iteration of the outer loop, one random facility (called target facility) f_G is chosen and located radially along the radius (r_f), which is making an angle θ' with the abscissa as shown in Fig. 8.

$$\theta' = i \times \theta, \ \ i = 1, 2, \ldots, M \tag{27}$$

Facilities are permitted to be placed within the boundaries of the given area. To satisfy this constraint, vector $\vec{a}$, which is a vector of random magnitude along vector's $\overrightarrow{r_f}$ direction, is taken, and facility, f_G, is placed at the end of this vector. The length of vector $\vec{a}$ is a random number between$[0, |\overrightarrow{r_f}| - r]$, where r is the length of the diagonal of facility f_G. The next step is checking the possibilities for overlap between all facilities. If any overlap occurs between the target facility f_G and the given area's boundaries or between target facility f_G and the previously placed facilities, the inner loop is triggered. It should be noted that facilities coordinates for each is calculated based on an origin that is located at the bottom-left corner of the site as shown in Fig. 5.

- **Inner loop**

Different repair functions based on the type of overlap are being developed to eliminate overlap. Repair functions guarantee the elimination of overlap between facilities and allocation of the facility within the boundaries of its corresponding quadrant. However,

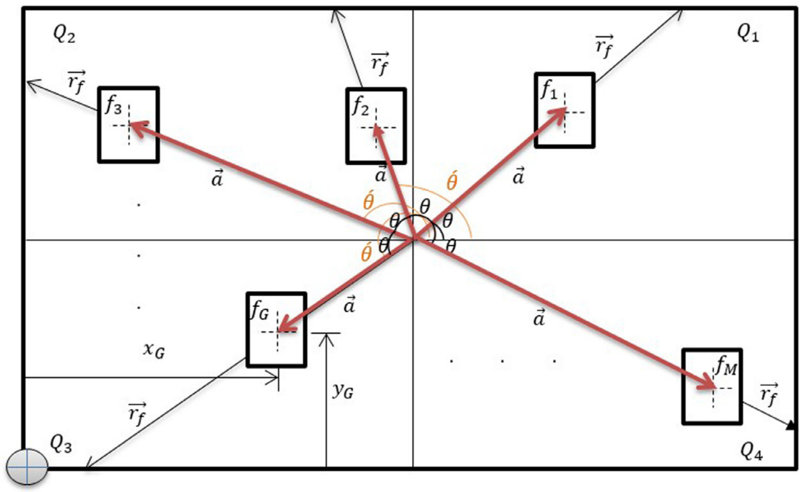

Fig. 5. The mechanics of developed heuristic.

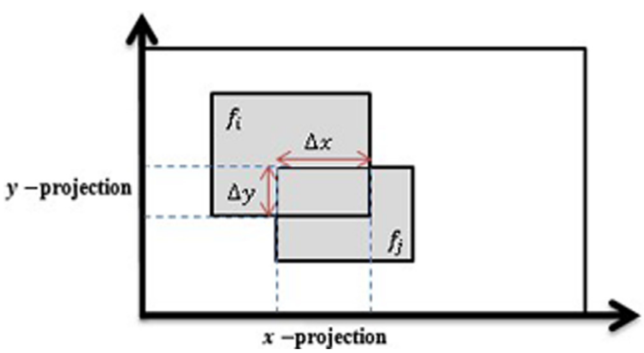

Fig. 6. Scheme of overlap between two facilities f_i and f_j.

if the corresponding quadrant is too congested, the overlapped facility can be placed partially in a different quadrant. Nevertheless, no facilities are allowed to violate the given area boundaries. The inner loop has two main steps. In the first step, the overlap between facility f_G and the overlapped facility f_j is repaired. Afterwards, overlap checking is performed for all facilities starting from the last placed facility to the first one to see if repair done in previous step has caused further overlaps or not. If no overlap takes place, the inner loop is ended and algorithm goes back to the outer loop to place another facility, given a facility is still left to be placed. However, if overlap is detected when checking for overlap between all the facilities, the second step of the inner loop is enacted.

The second step of the inner loop consists of few iterations. In each iteration, as explained one facility f_i is selected as target facility, and then the possibility of overlap between the target facility and rest of previously placed facilities is checked. If there is overlap between the target facility f_i and facility f_j, overlap elimination algorithms are enacted. The overlap has two main projections: one in x-direction, Δx and another in y-direction Δy. Δx represents the horizontal overlap between the two facilities f_i and f_j. In a similar fashion, Δy shows the vertical overlap between the two overlapped facilities as demonstrated in Fig. 6. If $\Delta x \le \Delta y$ the overlap is fixed by removing overlap in tx-projection direction; otherwise it does that in the y-direction. The repair mechanism starts by moving target facility f_i by the overlap distance Δ in appropriate direction.

Since no facility is allowed to violate the given area's boundaries, there is a need to know how much distance left between facility f_i and cell/floor (or quarter) boundaries. If the distance left is less than overlap Δ, then overlap elimination is carried out for the facility f_j. Moreover, If the distance left between the facility f_j and site (or quarter) boundaries is not less than overlap Δ, the overlap distance Δ should be applied to both facilities f_i and f_j. At the end of each iteration the overlap is checked once again to tackle any possibly newly occurred overlap. This loop is repeated until all overlap and intersection between facilities are repaired. The summary of the developed initialization heuristic is represented in Fig. 7.

4.2. Configuration changes

By moving from one configuration to another one, a new neighborhood solution is generated. In this paper, two operators called the *displacement operator,* and the *swap operator* has been designed to generate new neighborhood solution.

4.2.1. Displacement operator

The idea behind the developed displacement operator is to reduce the distance between the two facilities by moving one facility toward the other. To do that, one facility, which is called the goal facility f_G, is chosen randomly. Afterwards, facility f_G is moved towards its closest unattached facility which is called facility f_C, by taking randomly selected move steps. To illustrate more, the following concepts are defined:

(X_G, Y_G)	Horizontal and vertical coordinates of centroid of facility f_G, respectively.
(X_C, Y_C)	Horizontal and vertical coordinates of centroid of facility f_C, respectively.
(L_G, W_G)	Length and width of facility f_G, respectively.
(L_C, W_C)	Length and width of facility f_C, respectively.
O'	Centroid of f_G
O''	Centroid of facility f_C
$\overrightarrow{O'O''}$	Vector from centroid of facility f_G to the centroid of facility f_C.
$\vec{r'}$	Vector from O' to the closest boundary of facility f_G toward the facility f_C.
$\vec{r''}$	Vector from O'' to the closet boundary of the facility f_C toward facility f_G .
C	The conjunction of vector $\vec{r'}$ and the closest boundary of facility f_G to the facility f_C.
C'	The conjunction of vector $\vec{r''}$ and the closest boundary of facility f_C to facility f_G.

Step 1: Place all facilities on top of each in the centroid of the given area.

Step 2: Divide the given area into four quadrants and calculate angle between facilities. $\theta = \frac{360^{\circ}}{M}$

Step 3: Outer loop

Step 3.1: Randomly Choose one facility as target f_G

Step 3.2: Take radial movement:

Step 3.2.1: Calculate angle $\hat{\theta}$ of corresponding target facility f_G and specify radial r_f

Step 3.2.2: Find a vector $\vec{a}$ along vector $\vec{r_f}$

Step 3.3: Place target facility f_G at the end of vector $\vec{a}$

Step 3.4: overlap checking: if there is any overlap between target f_G and facilities that have already been placed go to step 5, otherwise go to step 5

Step 4: Inner loop

Step 4.1: Specify the corresponding quarter of facilities f_i and f_j

Step 4.2: Calculate the overlap Δ based on the comparision between Δx and Δy projections of the overlap between the two overlapped facilities f_i and f_j

Step 4.3: Apply an appropriate repair function

Step 4.4: Overlap checking: if there is any overlap go to step 4, otherwise go to step 5

Step 5: if all facilities placed on the floor (cell) go to step 6, otherwise go to step 3

Step 6: End

Fig. 7. Summary of developed initialization heuristic algorithm.

$|\overrightarrow{CC'}|$ Maximum_movable_distance.
θ_1 The angle between vector $\overrightarrow{O'O''}$ and horizontal line.
θ_2 The angle between vector $\overrightarrow{O'O''}$ and vertical line.

The steps of the move operator algorithm are explained below:

Step 1. One facility is chosen randomly as a goal facility, called facility f_G.
Step 2. Border(s) of facility f_G which no facilities have been attached to it is specified, called Free Border(s) (FB).
Step 3. Facilities which have been located on the same side of the free border(s), FB are specified, called Unattached Facilities (UF).
Step 4. The Euclidean distance between the centroid of facility f_G and facilities in UF are calculated.

$$Dis_{Gi} = \sqrt{(X_G - X_i)^2 - (Y_G - Y_i)^2} \ \forall \ i \in UF \tag{28}$$

Step 5. Facility which has the least Dis_{Gi} is chosen as the closest facility called facility f_C.
Step 6. Facility f_G is divided into four equal-sized quadrants by origin of its centroid, O'.
Step 7. The quadrant of facility f_G where facility f_C has been located is found.
Step 8. The maximum movable distance $|\overrightarrow{CC'}|$ which is along vector $\overrightarrow{O'O''}$ is calculated as given by Eqs. (29) to (33) and shown in Figs. 8 and 9.

$$\theta_1 = tan^{-1}\frac{|Opposite\ side|}{|Adjacent\ side|} = tan^{-1}\frac{|Y_G - Y_C|}{|X_G - X_C|} \tag{29}$$

$$\theta_2 = tan^{-1}\frac{|Opposite\ side|}{|Adjacent\ side|} = tan^{-1}\frac{|X_G - X_C|}{|Y_G - Y_C|} \tag{30}$$

Also : $\theta_2 = 90 - \theta_1$

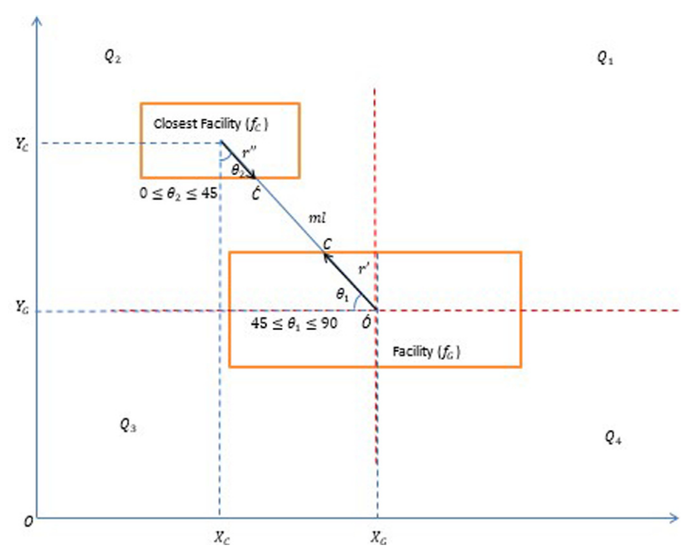

Fig. 8. Angle calculation in displacement operator (I).

The length of both vectors $\vec{r'}$ and $\vec{r''}$ depends on their corresponding angles θ_1 and θ_2 as provided by Eqs. (31) and (32).

$$\left|\vec{r'}\right| = \begin{cases} \frac{Adjacent\ side}{Cos\theta_1} = \frac{L_G/2}{Cos\theta_1} & if\ 0 \le \theta_1 \le 45^0 \\ \frac{Opposite\ side}{Sin\theta_1} = \frac{W_G/2}{Sin\theta_1} & if\ 45^0 \le \theta_1 \le 90^0 \end{cases} \tag{31}$$

Table 5
New coordinate of facility f_G.

New coordinates of facility f_G	Q_1	Q_2	Q_3	Q_4
$\boldsymbol{X_G}$	$X_G + ml^*Cos\theta_1$	$X_G - ml^*Cos\theta_1$	$X_G - ml^*Cos\theta_1$	$X_G + ml^*Cos\theta_1$
$\boldsymbol{Y_G}$	$Y_G + ml^*Sin\theta_1$	$Y_G + ml^*Sin\theta_1$	$Y_G - ml^*Sin\theta_1$	$Y_G - ml^*Sin\theta_1$

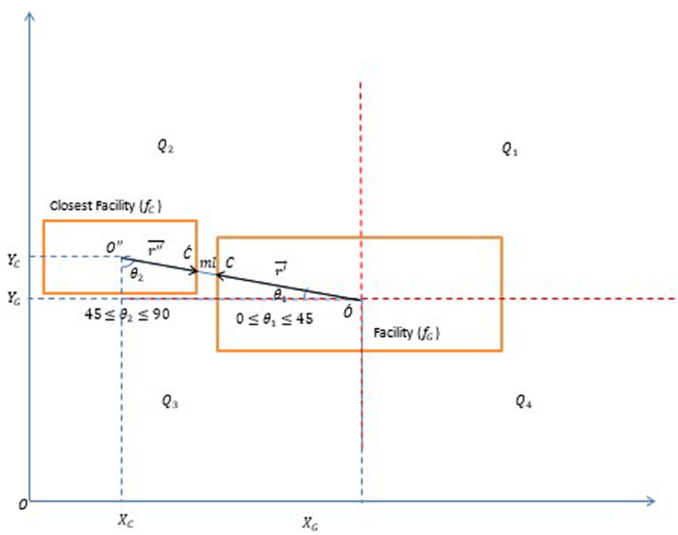

Fig. 9. Concept of Angle in displacement operator (II).

Table 6
Special cases for used for repair function.

$\Delta x \le \Delta y$		$\Delta x > \Delta y$	
Case *I*	Case *II*	Case *III*	Case *IV*
$\boldsymbol{x_i \le x_j}$	$\boldsymbol{x_i > x_j}$	$\boldsymbol{y_i \le y_j}$	$\boldsymbol{y_i > y_j}$
$x_j = x_j + \Delta x$	$x_j = x_j - \Delta x$	$y_j = y_j + \Delta y$	$y_j = y_j - \Delta y$

$$\left|\vec{r''}\right| = \begin{cases} \dfrac{Adjacent\ side}{Cos\theta_2} = \dfrac{W_C/2}{Cos\theta_2} & if\ 0 \le \theta_2 \le 45^0 \\ \dfrac{Opposite\ side}{Sin\theta_2} = \dfrac{L_C/2}{Sin\theta_2} & if\ 45^0 \le \theta_2 \le 90^0 \end{cases} \tag{32}$$

Hence, the length of vector $|\overrightarrow{CC'}|$ is:

$$\left|\overrightarrow{CC'}\right| = \sqrt{(X_C - X_{C'})^2 + (Y_C - Y_{C'})^2} \tag{33}$$

Step 9. The length of the movement called *ml* which is a random number in interval (0, $|\overrightarrow{CC'}|$] is calculated.
Step 10. Facility f_G is moved by *ml* along $\overrightarrow{CC'}$.
Step 11. The new coordinates of facility f_G is calculated which are shown in Table 5.
Step 12. The overlap check function is performed to check the possibility of overlap between facilities. In the case of overlap, appropriate repair function is performed. The detail of overlap check function and repair function is brought in Appendix B.

4.2.2. Swap operator

This operator switches the location of two randomly selected facilities. After the swap, check for overlap is performed, and in the case of any overlap, the appropriate repair function is implemented.

4.3. Objective function

The objective function represents the quantitative measurement of goodness of a system. The objective function has the same mechanism as developed mathematical modeling.

4.4. Annealing/cooling schedule

The annealing schedule determines the following:

4.4.1. Initial temperature

Since the annealing of solids is the base of the SA approach, the initial temperature is the melting point of SA algorithm, and it should be defined in such a way that the solutions generated by high acceptance probability approximately close to one. Kirkpatrick et al. (1983) noted that the initial temperature has to be large enough that 80% of generated solutions are accepted.

4.4.2. Cooling rate

The rate by which the temperature is decreased. In this paper the equation suggested by Kirkpatrick et al. (1983) to decrease the temperature is used. Assume:

t_i	Temperature at iteration *i*
10	Initial temperature
0.9	Cooling rate
k	The total number of temperature iterations

Hence, the temperature reduction equation is:

$$t_i = 10(0.9)^{i-1} \quad i = 1, 2, \ldots, k \tag{34}$$

4.4.3. Length of outer loop

The total number of iterations which the annealing is performed.

4.4.4. Termination

There are different approaches to stopping criteria such as: 1) A Specific number of temperature iterations, 2) Exact final temperature and, 3) No improvement for a specific number of temperature iteration.

Based on our experiment, we found the total temperature iterations $k = 75$ and the total number of generated new configurations ($100 \times M$), where *M* is the total number of facilities produces better results.

4.5. Multi-start search

Despite the fact that SA is a hill-climbing algorithm, it is still found after analysis of the primary computational experiments that the developed SA still gets entrapped in local optima and is unable to reach the global optimum. Hence, a multi-start search mechanism is implemented to increase the diversification and exploration capacity of the search. Firstly the system starts annealing for a specific number of iterations (α). Afterwards, a counter (β) counts the consecutive temperatures, where no improvements are made. To test for improvement, the average of total objective functions of accepted solutions at temperature iteration *i* is subtracted from that at iteration $i-1$. If the subtraction is less than or equal

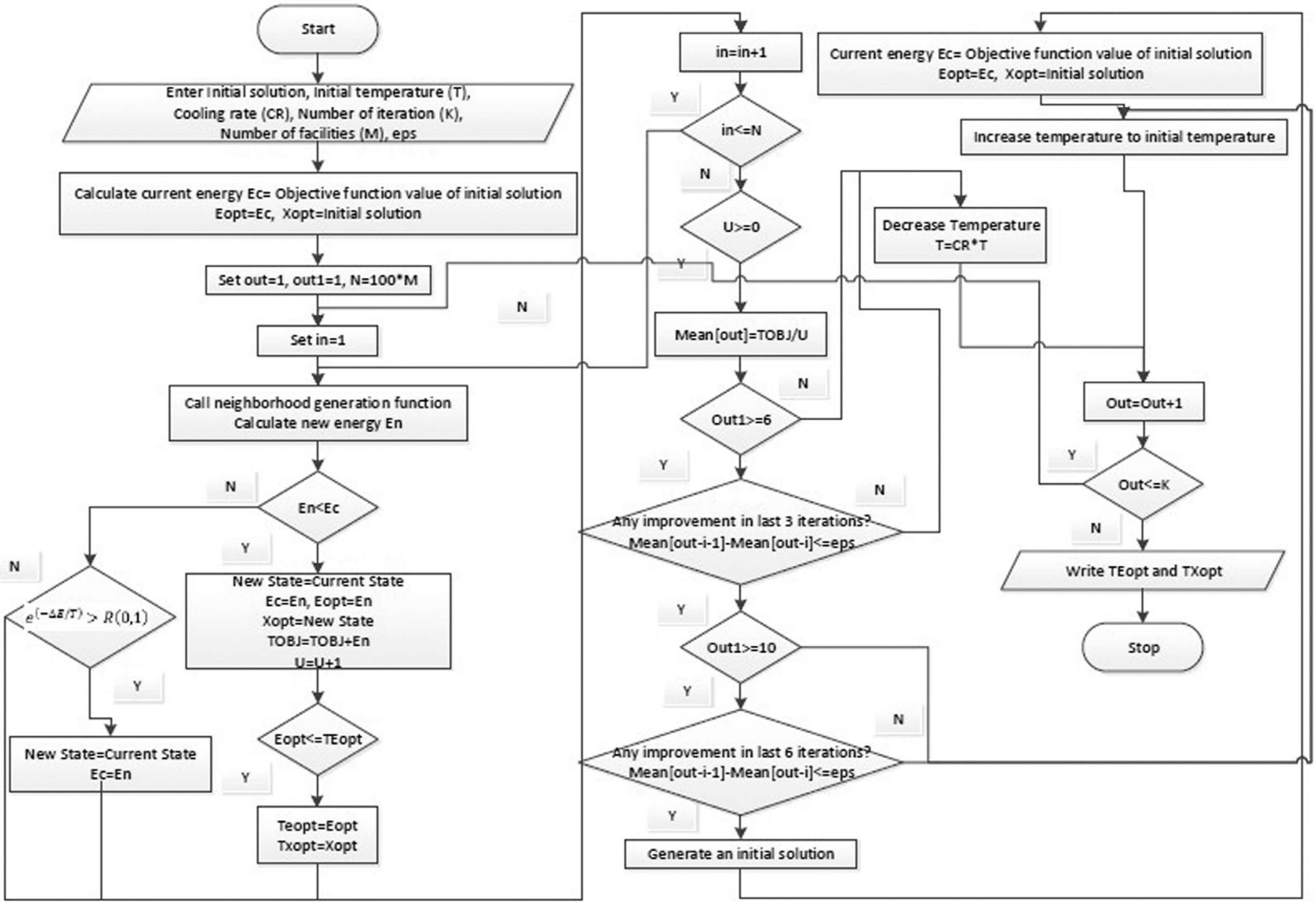

Fig. 10. Flowchart of developed multi-start search SA.

to predetermined value of ε, then an improvement is made. Otherwise, the counter β is incremented. If the counter β reaches a predetermined number (*fix*), the temperature is reset to initial temperature. If after that, improvement is made, β is also reset. Otherwise this procedure continues by a specific number of iterations γ. However, if after $\alpha+\gamma$ temperature steps, the counter β reaches a predetermined value of 2 $\times$*fix*, then it is likely that resetting the temperature is not going to allow for hill climbing of the current local optimum. Then, the whole search is then restarted using another random solution. The general algorithm of developed SA is summarized in Fig. 10 and the pseudo, code of is shown in Appendix A.

4.6. Overlap checking function and repair function

The developed repair function for SA ensures that the solution generated by the different neighborhood functions are feasible. It could be summarized in the following steps.

Assume two facilities f_i and f_j as the target facility and overlapped facility respectively have overlapped; *i.e.* $|x_G - x_i| \leq ((l_G + l_i)/2)$ and $|y_G - y_i| \leq ((w_G + w_i)/2)$. The overlap distance has two main projections in the x-direction ($\Delta\, x$) and in the y-direction (Δy). Δx (Eq. (35)) represents the horizontal overlap between the two facilities f_i and f_j .In a similar fashion, Δy (Eq. (36)) shows the vertical overlap between the two overlapped facilities as demonstrated in Fig. 6.

$$\Delta x = ((l_G + l_i)/2) - |x_G - x_i| \tag{35}$$

$$\Delta y = ((w_G + w_i)/2) - |y_G - y_i| \tag{36}$$

Based on the comparison between Δx and Δy and the coordinates of facility f_i and facility f_j four basic repair mechanisms are developed:

The pseudo code of the developed overlap checking and overlap elimination are given in Appendix.

5. Computational results

In this section, three major benchmark problems from the literature are used for verification and evaluation of the developed metaheuristc. The first one is extracted from Khare, Khare, and Neem (1988). The second test case is taken from Mir and Imam (2001). Finally, the third test case from Bazaraa (1975) which has two sub-cases is used. All test cases are run ten times; statistical mean and variance metrics are used for comparison purposes.

5.1. Test case two

The second test case is taken from Khare et al. (1988) is for the 10-equal size facility layout problem. They have developed a multi-goal facility layout design. When setting the weight $W1$ to zero, their objective function (Khare et al., 1988) becomes identical to one developed in Section 3.5. Other literature (Imam & Mir, 1993; Mir & Imam, 2001) have used this test problem as well. Imam and Mir (1993) run ten times with ten different initial solutions to test their model performance. For the purpose of

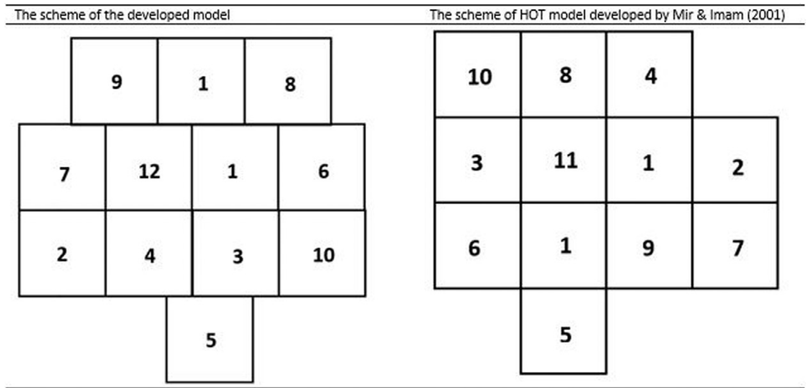

Fig. 11. A Comparison between the best-obtained layout designs for the 10-facility test case from Khare et al. (1988).

Table 7
The Comparison results for the first test case taken from Khare et al. (1988).

Model	Optimal cost		
	Best design	Mean	Std. dev.
Khare et al. (1988)	506.00	541.07	14.57
Imam and Mir (1993) (FLOAT)	504.96	517.80	9.87
Mir and Imam (2001) (HOT)	496.00	502.36	5.72
Developed multi-start search SA	489.89	494.34	3.20
Nonlinear model	684		

comparison the developed model is run ten times. Table 9 compares the results obtained from the developed model and that of Khare et al. (1988) and Imam and Mir (1993).

Based on the results shown in Table 7, the best design produced by the developed multi-start search SA algorithm has the least cost in comparison to other comparable ones from the literature. Moreover, the mean of the developed algorithm outperforms all other algorithms from the literature. Finally, the standard deviation of the developed algorithm is significantly smaller than Mir and Imam (2001), which has the best-obtained results so far in the literature. Fig. 11 shows the best layout scheme of the developed multi-start search SA in comparison with the best layout design found in the literature (Mir & Imam, 2001). Finally, results obtained from multi-start search SA algorithm have been compared to that obtained by the nonlinear model in Table 7; it shows how the developed multi-start search SA algorithm outperforms the results obtained from the nonlinear model, which is expected. SA guarantees sub-optimality, whereas all the nonlinear model guarantees are feasibility.

5.2. Test case three

The third test case problem data for 20-unequal facilities is extracted from Imam and Mir (1993). Bazargan-Lari and Kaebernick (1997) and Mir and Imam (2001) also have used the same test problem to evaluate the performance of their models. Imam and Mir (1993) and Mir and Imam (2001) have performed their developed model ten timed with ten different initial solutions; however, Bazargan-Lari and Kaebernick (1997) run their model ten times with the same initial solution. In this paper, the proposed multi-start search SA algorithm runs for two sets of ten runs each; two sets of runs, ten-runs each, have been performed in this case to provide a richer collective data set of comparison with the literature.

As shown in Table 8, for the first set of 10-runs, the developed multi-start search SA has generated the least (best) mean when compared to other all models from the literature. Additionally, the best layout generated by the developed multi-start search SA has significantly less optimal cost. However, the standard deviation of this first ten runs is larger than that of Bazargan-Lari and Kaebernick (1997). With regards to the second set of runs, the mean of the developed multi-start search SA is almost the same as that of the other models. While this set of runs has reached the least standard deviation, the generated best layout design is higher than that by Bazargan-Lari and Kaebernick (1997) and Mir and Imam (2001). Fig. 12 summarizes the comparison between the obtained layout *versus* the best obtained one from the literature. Lastly, as presented in Table 8; the results obtained from the developed multi-start search SA algorithm outperforms the results from the nonlinear model.

5.3. Test case four

Bazaraa (1975) has introduced two sub-test cases for unequal and irregular shape facility layout problem. The first and second sub-test case involves 12 facilities and 14 facilities respectively. These two test cases have been used by Bazargan-Lari and Kaebernick (1997), Hassan, Hogg, and Smith (1986), and Van Camp et al., (1991). Tables 9 and 10 represent the best layout design optimum cost. Moreover, Fig. 13 shows the layout scheme of 12-facility and 14-facility test problems.

At this point, no evaluation can be made for the generated mean, and standard deviation of the developed model for two test problems since none of the previous studies have provided those statistics. The range of interaction matrix or material flow rate in both test cases are significantly high; 288 and 162 for 12-facility and 14-facility problem respectively. This is seen as a potential cause of the relatively higher standard deviation evaluated; because minor changes of the location of one facility may have caused large discrepancies in the obtained objective function value.

The 12- facility test case run for ten times and 14-facility test problem run for two set of 10-runs; the results are provided in Tables 9 and 10 respectively. Based on the literature, Hassan et al. (1986) have the best results for both test problems. However, they have taken a certain assumption which makes the comparison difficult. The assumption is no restrictions on the dimensions of the facilities is taken, and facilities can take nonrectangular shape. This assumption terminates to the design of facilities in a different shape from what Bazaraa (1975) proposed originally. However, in this paper facilities have fixed dimension. To compare with the Nonlinear Programming (NLP) model presented by Van Camp, Carter, and Vannelli (1991), the results have been found in this paper is significantly better than them despite the fact that they used Euclidean distances to measure objective cost.

Table 8
Comparison results for second test case taken from Mir and Imam (1993).

Model		Optimal cost		
		Best design	Mean	Std. deviation
TOPOPT (Imam and Mir, 1989)		1320.72	1395.64	45.67
FLOAT (Imam & Mir, 1993)		1264.94	1333.81	27.51
HOT (Mir & Imam, 2001)		1225.40	1287.29	21.94
Bazargan-Lari and Kaebernick (1997)		1248.89	1286.6	14.9
Developed multi-start search SA	First set of runs (10 runs)	1218.746	1270.55	20.09
	Second set of runs (10 runs)	1260.43	1288.27	14.48
Nonlinear model		4578		

Cost for TOPOPT is reported from Imam and Mir (1993).

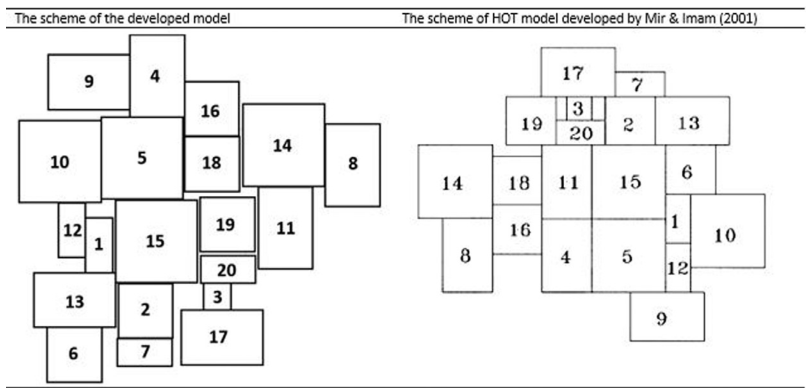

Fig. 12. A comparison between the best-obtained layout design for the 10-facility test case from Imam and Mir (1993).

Table 9
The Comparison results of the best-obtained layout design for 12-facility test case from Bazaraa (1975).

Model		Optimum cost of best layout design
		12-facility
Bazarra (1975)		14,079
Bazargan-Lari and Kaebernick (1997)		13,342
PLANET (Appel and Deisenroth, 1972)		11,664–11,808
SHAPE (Hassan et al., 1986)		10,578–11,140
NLP (Van Camp et al., 1991)		11,910
Developed multi-start search SA	Best layout design:	11,397.19
	Mean:	11,437.31
	Std. Dev:	41.223
Nonlinear model		24,752

Cost for PLANET and SHAPE are reported from Hassan et al. (1986).

The developed multi-start search SA obtain better results than that of Bazaraa (1975)'s and Bazargan-Lari and Kaebernick (1997)'s. However making a comparison between these models might not be as straightforward, because different assumptions have been taken. Firstly, the irregularity aspect of the shape of the facilities is not addressed in this paper. However Bazaraa (1975)'s and Bazargan-Lari and Kaebernick (1997) do address irregularity of shapes. The only irregular shape facility in the 12-facility test case is the third facility; in the 14-facility test case are facilities number four and eight have irregular shapes. Hence the majority of the facilities in those two test cases have a rectangular shape. Since the number of irregular facilities is too few, this aspect of facilities is believed not to affect the objective function value significantly. Secondly, the fixed dimensions of the area, which encloses the facilities are relaxed in this paper; however, this assumption has been taken into consideration using their study. Finally, both Bazaraa (1975) and Bazargan-Lari and Kaebernick (1997) assume facilities are orientation free; however in this paper orientation is fixed which is a critical constraint that usually results in constriction of the formed solution space as explained in synthesis matrix (Table 1). Lastly, based on Tables 9 and 10, the results obtained from the developed multi-start search SA algorithm for both subtest cases shows superiority than that from the nonlinear model.

Table 10
The Comparison results of the best-obtained layout design for 14-facility test case from Bazaraa (1975).

Model			Optimum cost of Best Layout Design
			14-facility
Bazarra (1975)			8170.5
Bazargan-Lari and Kaebernick (1997)			7409
PLANET (Appel and Deisenroth, 1972)			6399–6480
SHAPE (Hassan et al., 1986)			6339–6462
NLP (Van Camp et al., 1991)			6875
Developed multi-start search SA	First set of runs (10 runs)	Best layout design:	6543.18
		Mean:	6607.477
		Std. Dev:	49.51
	Second set of runs (10 runs)	Best Layout Design:	6469.9
		Mean:	6564.52
		Std. Dev:	48.05
Nonlinear model			14,881.5

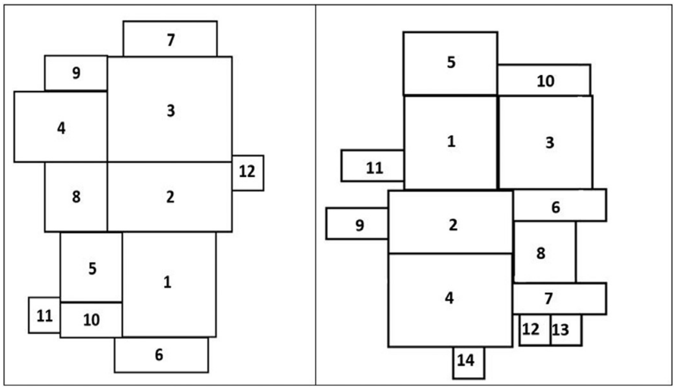

Fig. 13. The best-obtained layout design for 12-facility and 14-facility test cases from Bazaraa (1975).

6. Conclusion

In this paper, a Mixed Integer Nonlinear Programming (MINLP) mathematical model is developed to solve the facility layout problem (FLP) for unequal-area rectangular facilities to be placed in a continuous planar site. The suggested model has addressed some main issues related to the layout problem such as with-in site boundaries, overlap elimination, and aisle and block considerations. A near-optimal Simulated Annealing metaheuristic that uses a multi-start search mechanism to enhance the algorithm exploration capacity and escape local search traps is developed. A novel constructive heuristic is presented for initialization. The heuristic is unique in the sense that it places facilities radially. The developed multi-start search Simulated Annealing algorithm uses two neighborhood functions, namely the swap operator and the newly designed displaced operator. A repair function has been utilized to eliminate possible resulting overlap between facilities.

For evaluation and justification, a set of case studies and benchmark problems is used to demonstrate the merit of the collectively developed approach. Test case one, which is a case study from local industry, is used to validate the developed mathematical model. When solving test case two, the developed multi-start search SA completely outperforms its counterparts from the literature regarding both the mean and standard deviation of obtained cost-based objective function. For the test case three, two sets of runs (ten runs each) have been used. For the first set, both the obtained mean and best-found layout outperform its counterparts from the literature, but not the standard deviation. However, for the second set better results were obtained for the standard deviation outdoing comparable work in the literature, whereas a slight increase in values of both the mean and best-obtained objective function are observed. The original test case has four assumptions differing than ones taken; thus, it was not fair to compare our results against existing ones in literature. For future research, hybridizing SA with iterated local search is believed to diversify the solution space and speed up the search. Also, it will be added value especially for more specialized FLP problems of Cellular Manufacturing, if developed metaheuristic recognizes constraints of within-site boundaries.

Acknowledgment

Research presented in this paper has been supported by an NSERC DiscoveryGrant #811008, and a Direct Research Fund received from local industry.

Appendix A. Pseudocode of the developed simulated annealing (SA) algorithm

- Initialize Solution (run the heuristic algorithm), Calculate the current energy E_c as objective function value of initial solution $E_{opt} = E_c$, X_{opt} as initial solution
- Initialize temperature (T), Cooling rate (CR), Number of iteration (K), Number of machines (M), eps
- Initialize counters Set Out_A=1, Out_B=0
- Set N=100*M

➢ **Execute (Outside Loop A)**

➢ Set $n = 1, u = 0, T_{obj} = 0$

  ❖ **Execute (Inside Loop B)**
  - Call neighborhood generation functions
  - Calculate new energy (E_n)
  - If $E_n < E_c$ then
    - ✓ $X_N = X_C$
    - ✓ $E_n = E_c$, $E_{opt} < E_n$, X_{opt} = New state, $T_{obj} = T_{obj} + E_n$, $u = u + 1$
    - ✓ If $E_{opt} \le T_{E_{opt}}$ then
      - ○ $T_{E_{opt}} = E_{opt}$ and $T_{X_{opt}} = X_{opt}$

      End if

  Else
    - ✓ If $e^{(-\Delta E/T)} > R(0,1)$ then
    - ○ $X_N = X_C$ and $E_n = E_c$

      End if

  End if
  - $n = n + 1$

  ❖ **Loop until** $n \le N$ **(Loop B)**

  ❖ **Multi-Start Search Algorithm**

  If $u > 0$ then
  - ▪ $Out_B = Out_B + 1$
  - ▪ $Mean_{out_B} = T_{obj}/u$
  - ▪ If $Out_B \ge 6$ then
    - If $Mean_{out_B}$ improvement in last 3 iteration is less than equal to eps then
      - ○ If $Out_A \ge 10$ then

        if $Mean_{out_B}$ improvement in last 6 iteration is less than equal to eps? then
        1. Generate an initial solution using developed heuristic
        2. Current energy E_c as objective function value of initial solution $E_{opt} = E_c$, X_{opt} as initial solution
        3. Increase temperature to initial temperature

        Else
        - Increase temperature to initial temperature

        End if

      Else
      - ○ Increase temperature to initial temperature

      End if

    ELSE

    Decrease the temperature $T = CR \times T$

    End if

  Else

  Decrease the temperature $T = CR \times T$

  End if

  Else

  Decrease the temperature $T = CR \times T$

  End if

  ❖ $Out_A = Out_A + 1$

➢ **Loop until** $Out_A \le K$ **(Loop A)**

Appendix B. Pseudocode of the repair algorithm

Initialize parameters M
Initialize counters $G = 1, i = 0$
➢ Execute (outside loop)
❖ Set $i = 1$
❖ Execute (inside loop)
Execute **overlap checking function**
• If facility f_i overlaps with facility f_G then
▪ Calculate Δx and Δy
▪ Find which "case" is valid for facility f_G and facility f_i
▪ Repair facility f_i based on the Table (6)
▪ Set $k = 0, j = 1$
▪ Execute (Loop A)
✓ If facilities f_j has the shared border with facility f_i and the same "case" as of facility f_G and facility f_i then
$k = k + 1$ and $RepA[k] = j$
✓ End if
✓ $j = j + 1$
▪ Loop until $j \leq M$ (End of Loop A)
▪ Set $v = 1$
Execute (Loop B)
Repair facility $f_{RepA[v]}$ based on the Table (6)
Execute repair function 2 (parameter: $v, case$)
✓ Set $u = 1,\ p = 1$
✓ Execute (Loop C)
▪ If facility f_v has overlap with facility f_u then
○ overlap=overlap+1
○ $p = p + 1$
○ $ovlp[p] = u$
○ If ($case = I\ or\ case = II$) then
If $x_v \leq x_u$ then
Apply $case = I$ for facility f_u
Else
Apply $case = II$ for facility f_u
End if
○ Else
If $y_v \leq y_u$ then
Apply $case = III$ for facility f_u
Else
Apply $case = IV$ for facility f_u
End if
○ End if
▪ End if
▪ $u = u + 1$
✓ Loop until $u \leq M$ (Loop C)
✓ If ($Ovlap \neq 0$) then
▪ Set $q = 1$
▪ Execute (Loop D)
○ **Execute repair function 2 (parameter:$ovlp[q], case$)**
○ $q = q + 1$
▪ Loop until $q \leq p$ (Loop D)
✓ End if
○ $v = v + 1$
Loop until $v \leq k$ (Loop B)
○ $i = i + 1$
❖ Loop until $i \leq M$ (inside loop)
• $G = G + 1$
➢ Loop until $G \leq M$ (outside loop)

References

Allahyari, M. Z., & Azab, A. (2014). A Novel bi-level continuous formulation for the cellular manufacturing system facility layout problem. *CIRP Procedia, 33*, 87–92.
Allahyari, M. Z., & Azab, A. (2015). Improved bi-level mathematical programming and heuristics for cellular manufacturing facility layout problem. In *4th Proceedings of the 10th ASME manufacturing science and engineering conference*.
Arikaran, P., Jayabalan, V., & Senthilkumar, R. (2010). Analysis of unequal areas facility layout problems. *International Journal of Engineering, 4*(1), 44–51.
Armour, G. C., & Buffa, E. S. (1963). A heuristic algorithm and simulation approach to relative location of facilities. *Management Science, 9*(1), 294–309.
Balakrishnan, J., Cheng, C.-H., & Wong, K.-F. (2003). FACOPT: A user friendly FACility layout OPTimization system. *Computers & Operations Research, 30*, 1625–1641.
Bazaraa, M. S. (1975). Computerized layout design: A branch and bound approach. *AII Transactions, 7*(4), 432–438.
Bazargan-Lari, M. (1999). Layout designs in cellular manufacturing. *European Journal of Operational Research, 11*(2), 258–272.
Bazargan-Lari, M., & Kaebernick, H. (1997). An approach to the machine layout problem in a cellular manufacturing environment. *Production Planning & Control, 8*(1), 41–55.
Bazargan-Lari, M., Kaebernick, H., & Harraf, A. (2000). Cell formation and layout designs in a cellular manufacturing environment – A case study. *International Journal of Production Research, 38*(7), 1689–1709.
Bozer, Y. A., & Wang, C.-T. (2012). A graph pair representation and MIP model based heuristic for the unequal area facility layout problem. *European Journal of Operational Research, 218*(2), 382–391.
Castillo, I., Westerlund, J., Emet, S., & Westerlund, T. (2005). Optimization of block layout design problems with unequal areas: A comparison of MILP and MINLP optimization methods. *Computers and Chemical Engineering, 30*, 54–69.
Chiang, W.-C. H. (2001). Visual facility layout design system. *International Journal of Production Research, 39*(9), 1811–1839.
Drira, A., Pierreval, H., & Hajri-Gabouj, S. (2007). Facility layout problems: A survey. *Annual Reviews in Control, 31*(2), 255–267.
Gonçalves, J. F., & Resende, M. G. C. (2015). A biased random-key genetic algorithm for the unequal area facility layout problem. *European Journal of Operational Research, 246*(1), 86–107.
Golden, B. L., & Skiscim, C. H. (1986). Using simulated annealing to solve routing and location problems. *Naval Research Logistics Quarterly, 33*(2), 261–279.
Hassan, M. M. D., Hogg, G. L., & Smith, D. R. (1986). SHAPE: A construction algorithm for area placement evaluation. *International Journal of Production Research, 24*(5), 1283–1295.
Heragu, S. S., & Alfa, A. S. (1992). Experimental analysis of simulated annealing based algorithms for layout problem. *European Journal of Operational Research, 57*(2), 190–202.
Heragu, S. S., & Kusiak, A. (1988). Machine layout problem in flexible manufacturing systems. *Operations research, 36*(2), 258–268.
Heragu, S. S., & Kusiak, A. (1991). Efficient models for the facility layout problem. *European Journal of Operational Research, 53*(1), 1–13.
Imam, M. H., & Mir, M. (1993). Automated layout of facilities of unequal areas. *Computers & Industrial Engineering, 24*(3), 355–366.
Kaebernick, H., Bazargan-Lari, M., & Arndt, G. (1996). An integrated approach to the design of cellular manufacturing. *CIRP Annals – Manufacturing Technology, 45*(1), 421–425.
Ketcham, R. L., Malstorm, E. R., & McRoberts, K. L. (1989). A comparison of three computer assisted facilities design algorithms. *Computer Industrial Engineering, 16*(3), 375–386.
Khare, V. K., Khare, M. K., & Neem, M. L. (1988). Combined computer-aided approach facilities design problem and estimation of the distribution parameter in the case of multigoal optimization. *Computers and Industrial Engineering, 14*(4), 465–476.
Kirkpatrick, S., Geltt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Sience, 220*, 671–680.
Komarudin, & Wong, K. Y. (2010). Applying ant system for solving unequal area facility layout problems. *European Journal of Operational Research, 202*(3), 730–746.
Koulamas, C., Antony, S. R., & Jaen, R. (1994). A survey of simulated annealing applications to operations research problems. *Omega, 22*(1), 41–56.
Kulturel-Konak, S., & Konak, A. (2011). A new relaxed flexible bay structure representation and particles warm optimization for the unequal area facility layout problem. *Engineering Optimization, 43*(12), 1263–1287.
Kulturel-Konak, S., & Konak, A. (2013). Linear programming based genetic algorithm for the unequal area facility layout problem. *International Journal of Production Research, 51*(14), 4302–4324.
Kulturel-Konak, S., & Konak, A. (2014). A large-scale hybrid simulated annealing algorithm for cyclic facility layout problems. *Engineering Optimization*, 1–16.
Kusiak, A., & Heragu, S. S. (1987). The facility layout problem. *European Journal of Operational Research, 29*, 229–251.
Mavridou, T. D., & Pardalos, P. M. (1997). Simulated annealing and genetic algorithms for the facility layout problem: A survey. *Computational Optimization and Applications, 7*, 111–126.
Meller, R. D., & Gau, K-Y. (1996). The facility layout problem: Recent and emerging trends and perspectives. *Journal of Manufacturing Systems, 15*, 351–366.
Meller, R. D., Narayanan, V., & Pamela, H. V. (1999). Optimal facility layout design. *Operations Research Letters, 23*, 117–127.
Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics, 21*(6), 1087–1092.
Mir, M., & Imam, M. H. (2001). A hybrid optimization approach for layout design of unequal-area facilities. *Computers & Industrial Engineering, 39*, 49–63.
Montreuil, B. (1990). *A modelling framework for integrating layout design and flow network design*. Berlin Heidelberg: Springer.
Niroomand, S., Hadi-Vencheh, A., Şahin, R., & Vizvari, B. (2015). Modified migrating birds algorithm for closed loop layout with exact distances in flexible manufacturing systems. *Expert Systesm and Applications, 42*(19), 6586–6597.
Nordin, N. N., Zainuddin, Z. M., Salim, S., & Rajeswari, R. (2009). Mathematical modeling and hybrid heuristic for unequal size facility layout problem. *Journal of Fundamental Sciences, 5*, 79–87.
Press, W., Teukolsky, S. A., Vetterling, W. T., & Flanner, B. P. (2007). Numerical recipes the art of sientific computing. 549–555.
Ripon, K. S. H. N., Glette, K., Khan, K. N., Hovin, M., & Torresen, J. (2013). Adaptive variable neighborhood search for solving multi-objective facility layout problems with unequal area facilites. *Swarm and Evilutionary Computation, 8*, 1–12.
Sherali, H. D., Fraticelli, B. M. P., & Meller, R. D. (2003). Enhanced model formulation for optimal facility layout. *Operations Research, 51*(4), 629–644.
Tate, D. M., & Smith, A. E. (1995). Unequal-zare facility layout by genetic search. *IIE Transaction, 27*, 465–472.
Van Camp, D. J., Carter, M. W., & Vannelli, A. (1991). A nonlinear optimization approach for solving facility layout problems. *European Journal of Operational Research., 57*, 174–189.

Wang, M.-J., Hu, M. H., & Ku, M.-Y. (2005). A solution to the unequal area facilities layout problem by genetic algorithm. *Computers in Industry, 56*, 207–220.

Wilhelm, M. R., & Ward, T. L. (1987). Solving quadratic assignment problems by simulated annealing. *IIE Transactions*, 107–119.

Xiao, Y., Yoonho, S., & Minseok, S. (2013). A two step heuristic algorithm for layout design of unequal-sized facilities with input/output points. *International Journal of Production Research, 51*(14), 4200–42222.

Yaman, R., Gethin, D. T., & Clarke, M. J. (1993). An effective sorting method for facility layout construction. *International Journal of Production Research, 31*, 413–427.