Journal of Industrial and Systems Engineering
Vol. 11, No. 1, pp. 180-204
Winter (January) 2018

A heuristic method for combined optimization of layout design and cluster configuration in continuous productions

Aliasghar Miri[1], Hamideh Razavi[1*]

[1]*Industrial Engineering Department, Ferdowsi University of Mashhad, Mashhad, Iran*

miri_aam@yahoo.com, h-razavi@ um.ac.ir

Abstract

Facility layout problems have been generally solved either hierarchically or integrated into other phases of plant design. In this paper, a hybrid method is introduced so that clustering and facilities layout can be simultaneously optimized. Each cluster is formed by a group of connected facilities and selection of the most appropriate cluster configuration is aimed. Since exact method by Mixed Integer Programming (MIP) is limited to small problems, a heuristic algorithm including constructive and improving phases is developed. In order to enhance the performance of the algorithm, systematic generation of intersection points inside available area together with shaking, split groups and Tabu list techniques are used.Then, two different examples are presented and the comparison of the results supports the merit of the proposed algorithm. For further validation, 18 test problems are solved both by the proposed algorithm and MIP by CPLEX. Comparison of the results reveals that for up to 13 facilities, the best solutions of the algorithm are equal to optimum solution of MIP but achieved in shorter times. For larger problems with higher number of facilities, even though processing times for MIP is much longer, in almost all cases, it cannot produce the best solutions of the proposed algorithm.

Keywords: Facility layout problem, heuristic algorithm, cluster configuration, unequal facility sizes

1-Introduction

In most facility layout problems (FLPs), the main objective is the least cost of material handling between stations (Meller et. al, 1998). However, there are other factors to be considered such as product design, manufacturing processes, workstation and equipment design, production flow, sequencing and scheduling, available space, material transport and storage systems as well as layout constraints such as fixed stations, I/Os and safety concerns. Each of these factors can significantly change the final layout and ignoring each of them may produce an ineffective layout. Therefore, with incorporating these factors simultaneously or sequentially in plant design, two different approaches arise including hierarchical and integrated approach.

*Corresponding author
ISSN: 1735-8272,

In hierarchical plant design, facility layout is one of the secondary stages which starts after the completion of the product and process design (Deisenroth and Apple, 1972). The objective of the layout design is usually the least cost of the material handling. Other parameters of the model are generated by the process design such as specification of the workstations, from-to charts and cost of material handling between workstations. This information is generated when the process is thoroughly defined and its requirements for the tools and machineries as well as production sequence and scheduling is specified. In this approach, there is always a risk that the joint solution for the process and layout design is obtained far from the optimum.

Recently, the hierarchical approach has proceeded to an integrated approach which simultaneously optimizes facility layout together with process design, automation and scheduling (Realff et. al, 1996), (Bock and Hoberg, 2007). Figure1 shows the interactions of layout design and the product design, process design and its scheduling (Francis et. al, 1992).

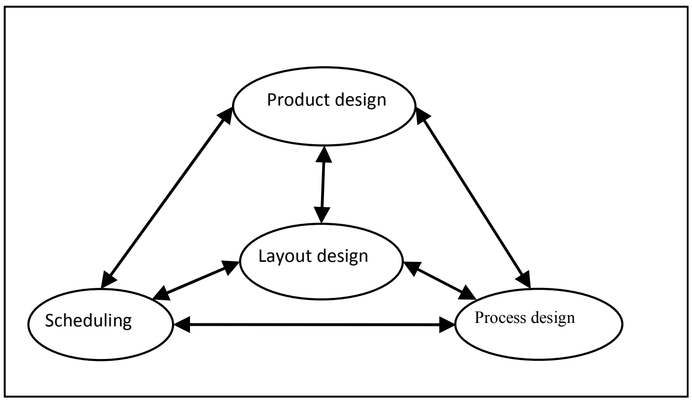


Fig 1. Relationship between plant design activities

Integrated approach is used for simultaneous selection of equipment, scheduling and layout design along with the intermediate storage design in continuous productions (Realff et. al, 1996). In the same time, an MIP model is built and solved by Penteado and Ciric for layout and facility design as well as financial and safety risks (Penteado and Ciric,1996).

In 2002, Barbosa et al. published a paper on facility and layout design (Barbosa-Póvoa et. al,2002). Later on in 2005, Patsiatziset al. proposed a MLP for simultaneous layout design, connecting structures and production planning (Patsiatzis and Papageorgiou ,2005). In 2007, an integrated method was developed for layout design and production routing (Taghavi and Murat,2011). Its authors divided the available area into unit cells and implemented the technique on these grids. A nonlinear model is also suggested by Taghavi and Muratfor simultaneous design of material flow and facility layout using a heuristic algorithm (Bock and Hoberg ,2008). A review of the above research with integrated approach has been performed by Barbosa-Povoa (2007). It quotes some of the shortcomings for batch production or discrete systems such as unbalance of multiple objectives, huge costs of integrated layout designs and heavy computations.

Integrated approach makes the solution space to expand drastically because of creating many new status. On the other hand, one has to simplify the complicated models to be able to solve them. This may cause the details to be lost so final solutions become inoperable. Therefore, the extent of integration in plant design is an impotant decision. It is generally known that economic, time and organizational factors affect the modelling extent in facilities design and the cost of formulation and its solution are mainly concerned (Francis et. Al,1992). Based on this concept, in the current research, the integrated approach is converted into a new formulation in which instead of a unique solution from process design phase, a set of candidate clusters are introduced into the layout phase in order to find the optimum solution.

Early discrete models for process layout were solved by quadratic assignment method (QAP). Koopmans et al.,(1957), Bland et al.,(1994), Loiola et al.(2007) and Moslemipour (2017) have reported different QAP based techniques. Afterwards, Hassan et al. (1987) used graph theory in order to make a network representation of the model. Similar to QAP, this technique cannot obtain optimum solutions when facilities are unequal in dimensions. Moreover, discrete models are not capable of incorporating I/O points, rotation of facilities or inter-facilities distance limitations. Hence, continuous models with open field layouts were introduced. These models were firstly introduced by Montreuil (1991) as an extension of QAP and used mixed integer programming. Meller et al. (1998) and Drira et al. (2017) have later employed the same method. In their approach, even though the dimensions of facilities, i.e. length (l_i) and width (d_i) can be different, a fixed constant ratio ($\lambda_i = l_i \times d_i$) , always relate them to each other. Such relationship is not common in process industries, so the method has not been practised in this field. Additionally, nonlinear constraints in the mathematical formulation complicated the model.

In 2003, Sherali et al.(2003) used an approximation method for area limitations and determined the accuracy of the result by linear constraints. They tried to improve the computational time by reducing problem symmetry, replacing constraints and branching method.Afterwards, Castillo et.al. (2005) presented two different models with mixed integer programming and symmetry breaking constraints. Similarly, Jankovits et al.(2011) used a two stage model, a primary stage for rough placements of facilities based on convex releasing and a second stage by semi-definite optimization . Their model was not appropriate for small sized problems but was efficient for larger problems.

Positioning of the I/O points is another issue in facility layout. The distance between two facilities, $d(i,j) = | X^O_j - X^I_i | + | Y^O_j - Y^I_i |$ is expressed by the distance between output points of the origin facility to the input point of the destination facility. $X^O{}_j$ and $Y^O{}_j$ are coordination of j^{th} origin output; and $X^I{}_i$ and $Y^I{}_i$coordination of the i^{th} input destination. Kimet al. (1999) added these information to the models and used mathematical programming for minimizing the cost of material handling between I/O points. Later on, Barbosa-Povoa et al.(2001) used mixed integer programming for multiple I/O points, irregular shapes and rotations of facilities. However, the dimension ratio had to be constant and a least cost layout in a continuous production system was intended.

In terms of heuristic and meta-heuristic algorithms for continuous models, Tam (1992) adapted genetic algorithm (GA) only for open field layout problem. He used slicing tree technique in order to generate different solutions. Later on, Chiang (2001) proposed an algorithm based on Tabu search and Shayan et al. (2004) developed a hybrid meta-heuristic from slicing tree and genetic algorithm. They used a hierarchical method and ignored I/O locations in layout design. Afterwards, Chwif et al.(1998) used simulated annealing and developed a heuristic algorithm. In their model, facility dimensions had to be proportional and each facility could only have a single I/O point. As mentioned, these assumptions are not valid in process industries where multiple I/Os are present and dimensions are not necessarily proportional.

Because of the computational limitations, the classic exact methods can only solve the problems with maximum 15 stations. Hence, for problems with greater number of stations, non-exact methods must be used. These solution methods in FLP can be categorized into heuristics, meta-heuristics and artificial intelligence techniques (Sharma and Singhal,2016). In 2016, a two-step technique was suggested for larger problems.In the first step a nonlinear model is used to determine the location of each station and in the second step convex optimization finds the most feasible solution (Anjos and Vieira, 2016).

A concurrent solution of facility layout and material handling was obtained for the first time by Hu et al. (2007) using genetic algorithm. In this research, only a single I/O was considered for each facility and a sequential algorithm was developed. Afterwards, Scholz et al.(2009) presented a combined Tabu search and slicing tree algorithm with fixed or variable dimensions for facilities. They extended their research by fixed-position facilities, aisles and internal barriers (Scholz et.al,2010). In 2012, Aiello et al. (2012) extended the previous work and presented a genetic based meta-heuristic algorithm for optimization of material handling, dimension ratio and proximity of facilities.

Recently, multi objective models are developed in order to integrate inter-cell layout and material handling (Leno et.al,2013). In terms of solution techniques, Kulturel-Konak et al.(2013), proposed a hybrid genetic algorithm and linear programming (LP) approach to solve the unequal area facility layout problem (UAFLP). They used a new encoding scheme, called location/shape, which represents the relative facility positions based on the centroids and orientations of the facilities. Once the relative facilities positions are set by the GA, the actual facility locations and shapes are determined by LP solution. Despite their novel modelling and solution techniques, these researches are dedicated to hierarchical layout design and cannot incorporate process characteristics.

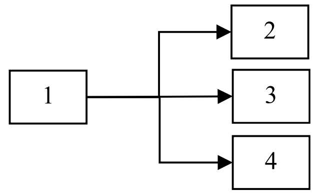
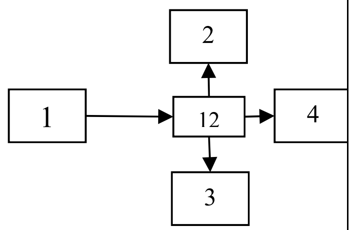
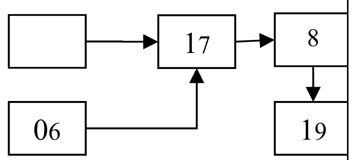
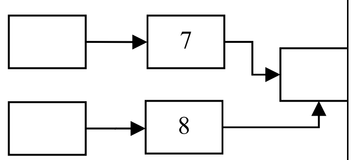
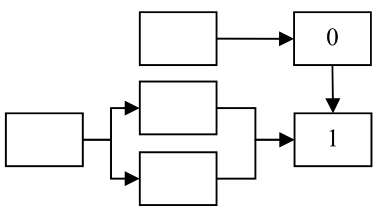
In summary, there are still many limitations on simultaneous optimization of layout and process design. Strictly adhering to either hierarchical or integrated approach is the main limitation of the existing models. Furthermore, in most studies, rotation of facilities and arbitrary number of I/O points are not allowed.

The current research tackles these limitations and introduces a hybrid method for interactive phases of process and layout design. The new method, which is called candidate clusters method, starts with several suggested cluster structures by process designers and attempts to decide among the choices while searching for optimum layout of facilities. The innovation of this approach can be summarized in the consideration of a set of candidate clusters coming from previous phase for process and operations design. The possibility of adding and eliminating stations in candidate clusters as well as having different dimensions for a station in different structures, are other innovative characteristics of this approach. It has the advantages of hierarchical approach such as low cost and less time consuming due to less complexity of the model together with the advantages of the integrated approach such as incorporating the process and operation specifications. In the rest of this paper, after a short description of cluster structures, a formulation of the MIP model is presented and the constraints, parameters and variables are briefly defined. Then, the heuristic algorithm is explained in detail accompanied by several examples. Finally, 18 sample problems have been solved by both heuristic and exact methods and results are compared and discussed.

2-Cluster configuration

In the proposed hybrid model, optional clusters of facilities aresuggestedby process designersbefore layout planning. Each cluster consists of a group of facilities with certain inter-connections. This configuration is widely used in continuous chemical processes for instance when a group of facilities such as mixers, boilers, etc. are connected to a reservoir (Moran, 2015). The concept is completely different to conventional hub groups by Farahani et al. (2013) in which hub problems are concerned with the optimum location of the hubs regarding the total costs. In clusters, the layout of the facilities in a continuous production, regarding multiple objectives is desired. For example, consider the cluster models in table1. It includes 12 facilities in two clusters. Each cluster has two different structures which are suggested by process designers.

Table 1. Cluster structures suggested by process designers

<table>
<tr><th>Cluster No.</th><th>Structure</th><th>Cluster links</th><th>From-To chart</th></tr>
<tr><td rowspan="2">1</td><td>1</td><td>
</td><td><table>
<tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>COIH</th></tr>
<tr><td>O_{11}</td><td>I_{41}</td><td>18</td></tr>
<tr><td>O_{12}</td><td>I_{21}</td><td>20</td></tr>
<tr><td>O_{13}</td><td>I_{31}</td><td>20</td></tr>
</table></td></tr>
<tr><td>2</td><td>
</td><td><table>
<tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>COIH</th></tr>
<tr><td>O_{11}</td><td>$I_{12,1}$</td><td>31</td></tr>
<tr><td>$O_{12,1}$</td><td>I_{41}</td><td>18</td></tr>
<tr><td>$O_{12,2}$</td><td>I_{21}</td><td>20</td></tr>
<tr><td>$O_{12,3}$</td><td>I_{31}</td><td>20</td></tr>
</table></td></tr>
<tr><td rowspan="2">2</td><td>1</td><td>
</td><td><table>
<tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>COIH</th></tr>
<tr><td>O_{51}</td><td>I_{71}</td><td>17</td></tr>
<tr><td>O_{61}</td><td>I_{72}</td><td>17</td></tr>
<tr><td>O_{71}</td><td>I_{81}</td><td>20</td></tr>
<tr><td>O_{81}</td><td>I_{91}</td><td>21</td></tr>
</table></td></tr>
<tr><td>2</td><td>
</td><td><table>
<tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>COIH</th></tr>
<tr><td>O_{51}</td><td>I_{71}</td><td>17</td></tr>
<tr><td>O_{61}</td><td>I_{81}</td><td>17</td></tr>
<tr><td>O_{71}</td><td>I_{91}</td><td>19</td></tr>
<tr><td>O_{81}</td><td>I_{92}</td><td>19</td></tr>
</table></td></tr>
<tr><th colspan="4">Other links</th></tr>
<tr><td colspan="2">
</td><td colspan="2"><table>
<tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>COIH</th></tr>
<tr><td>O_{41}</td><td>I_{51}</td><td>15</td></tr>
<tr><td>O_{41}</td><td>I_{61}</td><td>15</td></tr>
<tr><td>O_{91}</td><td>$I_{10,1}$</td><td>16</td></tr>
<tr><td>O_{51}</td><td>$I_{11,2}$</td><td>17</td></tr>
<tr><td>O_{61}</td><td>$I_{11,2}$</td><td>17</td></tr>
</table></td></tr>
</table>

As can be seen in table 1, facility No. 1 supplies the flow to facilities No. 2, 3 and 4, either directly or via a cluster centre. In cluster No. 2, facilities No. 5 to 9 can have two different connections depending on the process design. Indeed, structure no. 1 has divided the operation between facilities No. 7 and 8, while in structure No. 2 the operation is fully performed in parallel configuration. Each cluster has an individual cost including operational and non-operational costs which are different to other clusters.

In the cluster approach, the presence of some facilities depends on the structure of the cluster. For instance, if structure No. 2 in cluster No. 1 is opted, facility No. 12 is present otherwise it is eliminated from the process and layout design. This situation may happen by various reasons. For example, an operation can be either completed on one facility or be broken into several parts and completed on multiple facilities. In this research, those facilities which their presence is conditioned to the selected structure of the clusters such as semi-finished stockings or distribution centres are named cluster centres.

3- Model description

Facility layout in this research is confined to a two dimensional (x,y) space. One or more clusters can be defined, each consisting of at least two different connection structures, i.e. from-to charts with specified inner links and connecting structures (piping, conveyors or similar means of material flow).

Cluster centres are well defined if present. Transportation costs and those costs related to cluster structures, e.g. extra equipment, semi-finished inventories, etc. are known in advance. Each facility $j$is rectangular with independent fixed length and width and can have single or multiple I/O points identified by O_{ji}/I_{ji}. Distances between I/O points are rectilinear. Facilities can rotate counter-clockwise by integer multiples of 90 degrees.Total available area is limited and pre-specified.

It should be noted that each cluster may contain several I/O points of different facilities. Meanwhile, I/O points and their corresponding facilities may contribute to several clusters simultaneously.

Figure 2 shows a single facility in its base position with zero degree rotation. It has one input point as $I_{11}$and two output point as O_{11} and O_{12}.

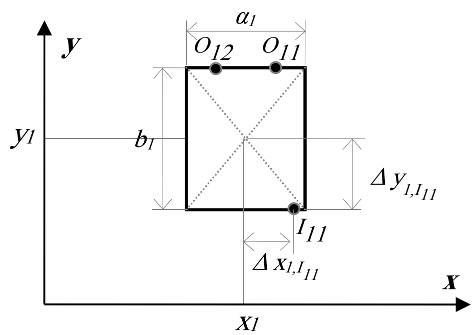


Fig 2. A typical primary position of facility

As can be seen in figure 2, I/O points are linked to the geometrical centre of facilities by distance parameters. For instance, for input point I_{11} , distance parameters are defined as $\Delta x_{1,I11}$and $\Delta y_{1,I11}$. Cost matrix is a cross product of flow matrix and transportation costs matrix. The final solution contains optimum plant layout, facilities orientations and likely cluster centres as well as optimum connections inside clusters.

3-1- Model formulation

A mathematical representation of the MIP model is presented in equations (1) to (38) and the explanations are given in the next sections.

$$\min \ TC = \sum_{\substack{j,i,j',i' \\ (\forall i,j,i',j',h : O_{ji} \notin links_h \ or \ I_{j'i'} \notin links_h)}} C_{OI\ j,i,j',i'} . D_{OI\ j,i,j',i'} + \sum_{h=1} CH_h \tag{1}$$

s.t

$$\sum_{\substack{j,i,j',i' \\ O_{ji},\ I_{j'i'} \in links_h}} C_{OIH\ j,i,j',i',k,h}\ D_{OI\ j,i,j',i'} + \Pi_{hk} - (1 - w_{hk}) M^d \le CH_h \qquad ;\forall h,k \tag{2}$$

$$\sum_k w_{kh} = 1 \qquad ;\forall h \tag{3}$$

$$E_j = \sum_{k \in structure_{jh}} w_{kh} \qquad ;\ \forall j,h\ :\ j \in links_h, GH \tag{4}$$

$$x_{O_{ji}} = x_j + r1_j\,\Delta x_{j,O_{ji}} - r2_j\,\Delta y_{j,O_{ji}} - r3_j\,\Delta x_{j,O_{ji}} + r4_j\,\Delta y_{j,O_{ji}} \quad ;\forall j,i \tag{5}$$

$$y_{O_{ji}} = y_j + r1_j\,\Delta y_{j,O_{ji}} + r2_j\,\Delta x_{j,O_{ji}} - r3_j\,\Delta y_{j,O_{ji}} - r4_j\,\Delta x_{j,O_{ji}} \quad ;\forall j,i \tag{6}$$

$$x_{I_{ji}} = x_j + r1_j\,\Delta x_{j,I_{ji}} - r2_j\,\Delta y_{j,I_{ji}} - r3_j\,\Delta x_{j,I_{ji}} + r4_j\,\Delta y_{j,I_{ji}} \quad ;\forall j,i \tag{7}$$

$$y_{I_{ji}} = y_j + r1_j\,\Delta y_{j,I_{ji}} + r2_j\,\Delta x_{j,I_{ji}} - r3_j\,\Delta y_{j,I_{ji}} - r4_j\,\Delta x_{j,I_{ji}} \quad ;\forall j,i \tag{8}$$

$$r1_j + r2_j + r3_j + r4_j = 1 \quad ;\forall j,i \tag{9}$$

$$r_j = r2_j + r4_j \qquad ;\forall j \tag{10}$$

$$r_j \le E_j \qquad ;\forall j \in GH \tag{11}$$

$$x_j - x_i + M^d(E1_{ji} + E2_{ji}) \ge (l_j + l_j)/2 + Zx_{ji}^{\min} \quad ; \quad \forall j > i \ \text{و}\ i,j \notin GH \tag{12}$$

$$x_i - x_j + M^d(1 - E1_{ji} + E2_{ji}) \ge (l_j + l_j)/2 + Zx_{ji}^{\min} \quad ; \quad \forall j > i \ \text{و}\ i,j \notin GH \tag{13}$$

$$y_j - y_i + M^d(1 + E1_{ji} - E2_{ji}) \ge (d_j + d_j)/2 + Zy_{ji}^{\min} \quad ; \quad \forall j > i \ \text{و}\ i,j \notin GH \tag{14}$$

$$y_i - y_j + M^d(2 - E1_{ji} - E2_{ji}) \ge (d_j + d_j)/2 + Zy_{ji}^{\min} \quad ; \quad \forall j > i \ \text{و}\ i,j \notin GH \tag{15}$$

$$x_j - x_i + M^d(1 + E1_{ji} + E2_{ji} - E_i) \ge (l_j + l_i)/2 + Zx_{ji}^{\min} \quad ;\forall j,i : i \in GH,\ j \notin GH \tag{16}$$

$$x_i - x_j + M^d(2 - E1_{ji} + E2_{ji} - E_i) \ge (l_j + l_j)/2 + Zx_{ji}^{\min} \quad ;\forall j,i : i \in GH,\ j \notin GH \tag{17}$$

$$y_j - y_i + M^d(2 + E1_{ji} - E2_{ji} - E_i) \ge (d_j + d_j)/2 + Zy_{ji}^{\min} \quad ;\forall j,i : i \in GH,\ j \notin GH \tag{18}$$

$$y_i - y_j + M^d(3 - E1_{ji} - E2_{ji} - E_i) \ge (d_j + d_j)/2 + Zy_{ji}^{\min} \quad ;\forall j,i : i \in GH,\ j \notin GH \tag{19}$$

$$E1_{ji} + E2_{ji} \le 2E_i \qquad ;\forall j,i : i \in GH,\ j \notin GH \tag{20}$$

$$x_j - x_i + M^d(2 + E1_{ji} + E2_{ji} - E_i - E_j) \ge (l_j + l_i)/2 + Zx_{ji}^{\min} \quad ; \forall j,i \in GH,\ j > i \tag{21}$$

$$x_i - x_j + M^d(2 - E1_{ji} + E2_{ji} - E_i - E_j) \ge (l_j + l_j)/2 + Zx_{ji}^{\min} \quad ; \forall j,i \in GH,\ j > i \tag{22}$$

$$y_j - y_i + M^d(2 + E1_{ji} - E2_{ji} - E_i - E_j) \ge (d_j + d_j)/2 + Zy_{ji}^{\min} \quad ;\forall j,i \in GH,\ j > i \tag{23}$$

$$y_i - y_j + M^d(3 - E1_{ji} - E2_{ji} - E_i - E_j) \ge (d_j + d_j)/2 + Zy_{ji}^{\min} \quad ;\forall j,i \in GH,\ j > i \tag{24}$$

$$E1_{ji} + E2_{ji} \le 2E_i \qquad ; \forall j,i \in GH,\ j > i \tag{25}$$

$$E1_{ji} + E2_{ji} \le 2E_j \qquad ; \forall j,i \in GH,\ j > i \tag{26}$$

$$x_{O_{ji}} - x_{I_{j'i'}} = \Delta x^{+}_{O_{ji},I_{j'i'}} - \Delta x^{-}_{O_{ji},I_{j'i'}} \qquad ;\forall j,i,j',i' \tag{27}$$

$$y_{O_{ji}} - y_{I_{j'i'}} = \Delta y^{+}_{O_{ji},I_{j'i'}} - \Delta y^{-}_{O_{ji},I_{j'i'}} \qquad ;\forall j,i,j',i' \tag{28}$$

$$DOI_{j,i,j',i'} = \Delta x^{+}_{O_{ji},I_{j'i'}} + \Delta x^{-}_{O_{ji},I_{j'i'}} + \Delta y^{+}_{O_{ji},I_{j'i'}} + \Delta y^{-}_{O_{ji},I_{j'i'}} \quad ;\forall j,i,j',i' \tag{29}$$

$$x_j \ge l_j / 2 \qquad ; \qquad \forall j \tag{30}$$

$$y_j \ge d_j / 2 \qquad ; \qquad \forall j \tag{31}$$

$$x_j + l_j / 2 \le X^{\max} \quad ; \qquad \forall j \tag{32}$$

$$y_j + d_j / 2 \le Y^{\max} \qquad ; \qquad \forall j \tag{33}$$

$$x_j, y_j, l_j, d_j, CH_h \geq 0 \quad ; \quad \forall j, h \tag{34}$$

$$D_{ji}, \Delta x^+_{ji}, \Delta x^-_{ji}, \Delta y^+_{ji}, \Delta y^-_{ji} \geq 0 \quad ; \quad \forall i, j \tag{35}$$

$$r_j, E_j, E1_{ji}, E2_{ji}, w_{hk} \in \{0,1\} \quad ; \quad \forall i, j, h, k \tag{36}$$

3-1-1- Objective Function

The objective function of the model is defined based on minimum connection costs by equation (1). It is the sum of the investments on the physical connections, material flow costs and structural expenses of clusters.

In equation (1), $links_h$ is the set of I/O for the members of the h^{th} cluster. The first statement of *TC*, expresses the connection costs when I/O of one of member facilities do not belong to the h^{th}cluster $(\forall O_{ji}\ or I_{ji} \notin links_h)$. The second statement defines the connection costs of the facilities whoseI/Os belong to the h^{th}cluster $(\forall O_{ji}, I_{ji} \in links_h)$. The assignment cost of the cluster center and other expenses are evaluated by the process design team (Π_{hk}) by equation (2).

In the h^{th} cluster, inter-facility connections can be provided only through a single structure which is selected among the suggestions of the process design team (equation (3)).

3-1-2- Constraints

The constraints in equation (4) are related to assignment of the cluster centers based on the cluster structures. Equations (5) to (11) define the rotations statusof the facilities. Overlaps are eliminated by equations (12) to (19) and safety concerns are regarded by equations (20) to (26).

The distances between I/Os are rectilinear and calculated by equation (37).

$$DOI_{j,i,j',i'} = |x_{O_{ji}} - x_{I_{j'i'}}| + |y_{O_{ji}} - y_{I_{j'i'}}| \quad : \forall j, i, j', i' \tag{37}$$

Euclidean distances are linearized by Equations (27) to (29) and total area is restricted by equations (30) to (33).

3-1-3-Parameters and variables

The model parameters as well as continuous and binary variables are defined in table2.

	symbol	Definition
Parameters	a_i , b_i	dimensions of equipment i alongr the x and $y$axis respectively
	Π_{hk}	cost of structure $k^{th}$of h^{th} cluster
	M^d	upper bounds of the distance between two facilities
	H	number of clusters
	m	number of facilities
	$COI_{j.i,j',i'}$	cost per distance, defined between the output point$O_{ji}$and the input point $I_{j'i'}$ $(\forall i, j, i', j', h : O_{ji} \notin links_h \; or \; I_{ji} \notin links_h)$
	$COIH_{j.i,j',i',k,h}$	cost per distance, defined between the output point$O_{ji}$and the input point $I_{j'i'}$in structure k^{th} of h^{th}cluster $(\forall i, j, i', j', h : O_{ji}, I_{ji} \in links_h)$
	$\Delta y_{j,Oji}$,$\Delta x_{j,Oji}$	relative distance between the output$O_{ji}$and the geometrical centre of the facility j respectively in the x- and y-axis, as
	$\Delta y_{j,Iji}$,$\Delta x_{j,Iji}$	relative distance between the input$I_{ji}$and the geometrical centre of the facility j respectively in the x- and y-axis, as
	X^{max} و Y^{max}	maximum area x- and y-coordinates
Continuous variables	l_i, d_i	length and depth of equipment facility i
	CH_h	cost of h^{th} cluster
	TC	value of objective function
	$DOI_{j.i,j',i'}$	total rectilinear distance between the output point $O_{ji}$and the input point $I_{j'i'}$
	x_{Oji} ،y_{Oji}	coordinates of the output point O_{ji}
	$x_{I_{ji}}$,$y_{I_{ji}}$	coordinates of the input point I_{ji}
	$\Delta y^+_{Oji,Ij'i'}$, $\Delta x^+_{Oji, Ij'i'}$	relative distance in x-coordinates and y-coordinates between the output point $O_{ji}$and the input point $I_{j'i'}$ when$x_{Oji}- x_{Ij'i'}\geq 0$,$y_{Oji}- y_{Ij'i'}\geq 0$
	$\Delta y^-_{Oji,Ij'i'}$,$\Delta x^-_{Oji, Ij'i'}$	relative distance in x-coordinates and y-coordinates between the output point $O_{ji}$and the input point $I_{j'i'}$ when respectively:$x_{Oji}- x_{Ij'i'}\leq 0$,$y_{Oji}- y_{Ij'i'}\leq 0$
Binary variables	r_i	Facility no.i orientation; equal to 1 if the length of facility is parallel to the x-axis otherwise 0
	w_{hk}	Structure no.i of h_{th} cluster; equal to1 if selected otherwise0
	E_j	facility j existence, which equals 1 if the facility j is present in the solution problem otherwise 0
	$E1_{ji}$, $E2_{ji}$	non-overlapping binary variable
	$r1_j$,$r2_j$,$r3_j$,$r4_j$	Facility no.j anti-clockwise rotation, expressed in integer multiples of 90 (respectively 0, 90, 180, 270) from the original equipment representation.

Table 2. Definition of the parameters and variables

3-1-4-Solution

The model is programmed in ILOGCPLEX 12.1 and solved for different test problems. A set of results are listed in Table 11. It can be seen that for 13 facilities, the processing time exceeds 3 hours. Since the number of facilities in a continuous production plant can easily reach to 20 or 30, the solution time by MIP becomes unacceptable. Therefore, a heuristic algorithm is introduced.

4-Heuristic algorithm

The proposed heuristic algorithm consists of two consecutive parts including constructive and improving algorithms. Two more techniques including split groups and Tabu lists are also devised in order to reduce the processing time of the algorithm. The programming is done in Microsoft visual studio 2010 C++ environment and ran by a Core(TM) i7 CPU 2.1GHz and 6GB RAM. In all sections, random numbers are generated based on uniform distribution.

4-1-Constructive algorithm

This algorithm is similar to Planet algorithm by Apple et al. (1972), albeit with some modifications. Initially, a facility is chosen by random and placed in an arbitrarylocation. Next, other facilities are located one by one, based on their rank among connection costs to previously located facilities. In fact, single facility location problem is solved sequentially until all facilities are located. Apseudo code for this algorithm is shown in figure3.Finding a proper location for each facility is the challenge of continuous models. In the suggested method, at first the total area is meshed so that each intersection point can be a potential location of a corner or an I/O point of the facility. The intersection points can be systematically produced by intersection of abscissa and ordinate lines of I/Os, corners of facilities and boundaries of the area.

```
Constructive algorithm(){
        for( h=0 ; h < H ; h++ ){
                ks_h= Rand;
                Update the costs and lists of facilities considering cluster information.
                } //End for
        for( j=1 ; j ≤ 2m ; j++ ){
                Select a facility by random and place it in an arbitrary location.
                for( i=1 ; i ≤ m ; i++ ){
                        Among the rest of unlocated facilities, select the facility with maximum
                        connection costs to located ones and locate it.
                        If( solution is not  feasible )
                             break;
                } //End for
                If( solution is feasible){
                        Calculate the cost (Tc_0).
                        break;
                } //End if
        }//End for
        If( solution is not  feasible )
                printf( "There was no solution" );
        else
return solution_0;}
```

Figure3. Pseudo code of the constructive algorithm

In the constructive algorithm, a solution is considered feasible when there is no overlap with previously located facilities. ks_h is the selected structure of the h^{th} cluster, H is the number of clusters and m is the number of facilities.

4-1-1- Systematic generation of intersection points

Intersection points in a confined area with multiple facilities can be divided into four groups including intersections of: **1.** Abscissa and ordinate lines of I/O points belong to different facilities, **2.** Facility corners and boundaries of the workshop area, **3.** Abscissa lines of I/O points and ordinate lines of facility corners and boundaries of the workshop area and **4.** Ordinate lines of I/O points and abscissa lines of facility corners and boundaries of the workshop area.

To define the above intersections, horizontal and vertical coordination of I/O points and j^{th} facility corners are collected in S_{xoi},S_{yoi},S_x and $S_y$sets respectively. Then, these sets are sorted in ascendingarrangement and repeated entries are excluded so that the sets are downsized. The area boundaries include 8 intersection points for which the coordination are obtained by equation (38) to (45).

$$xf_1 = \min(S_x) + X^{\max} \quad (31)$$

$$xf_2 = \min(S_x) + Y^{\max} \quad (32)$$

$$xf_3 = \max(S_x) - X^{\max} \quad (3)$$

$$xf_4 = \max(S_x) - Y^{\max} \quad (4)$$

$$yf_1 = \min(S_y) + X^{\max} \quad (5)$$

$$yf_2 = \min(S_y) + Y^{\max} \quad (6)$$

$$yf_3 = \max(S_y) - X^{\max} \quad (7)$$

$$yf_4 = \max(S_y) - Y^{\max} \quad (8)$$

Where xf_k is the maximum available area for positioning the new facility either at the right side of the located facilities when $k=1,2$, or at the left side when $k=3,4$. Similarly,yf_k is the maximum available areaeither at the topside of the located facilities when $k=1,2$ or at the bottom side when $k=3,4$.Additionally, the distinction between k values are defined by equation (46).

$$k = \begin{cases} 1,3 \text{ , if the default orientation is considered for the workshop area} \\ 2,4 \text{ , if the default orientation of the workshop area is rotated 90 degree counter-clockwise} \end{cases} \quad (46)$$

Furthermore, the coordination of the workshop boundaries are added to S_x and S_y. When these sets are completed, next facility should be placed on one of the intersection points so that total cost (Tc) is minimized. For this purpose, the position of this facility is examined by the placement of its corner or I/O points on four points including intersection of S_{xoi} and S_{yoi},S_x and S_y, S_{xoi} and S_y; and S_x and S_{yoi}. At each of these intersections, rotational status of 0, 90, 180 and 270degrees for the facility are also tested.

As an example, the coordination sets for a facility after filtering the repeated entries, is defined by S_{xoi}={95, 105},S_{yoi}={100},S_x={95, 105, 125, 130, 75, 70}and S_y={96, 104, 126, 131, 74, 69}. The intersection points of these sets are generated by crossings of the dashed lines in figure 4.

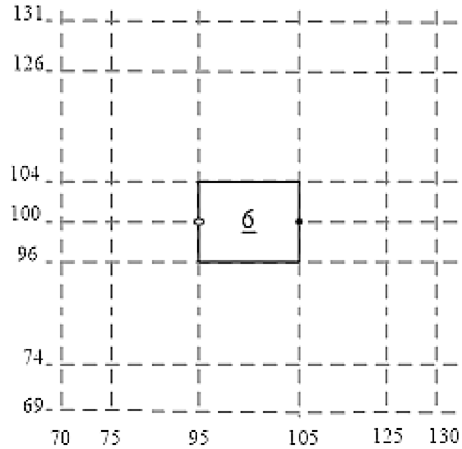


Fig 4. Intersection points for the sample sets of coordination

4-2-Improving algorithm

In each iteration of this algorithm as shown in figure 5, several facilities are randomly selected and relocated using single facility location method. Initial solution or *solution0* is taken from constructive algorithm and is saved in *best solution*. Then, the algorithm moves to an outer loop and runs *sub local search 1* algorithm. It runs until locating each facility in its best possible place within the current layout. Next, *sub local search 2*algorithm relocates *fch* number of facilities so that current layout can be improved.

```
Improving algorithm(){
        Constructive algorithm();
        bestSolution = solution_0;
        BestTc=Tc_0;
        for( i=1 ; i ≤ plo ; i++){
                sub local search1();
                sub local search2(fch , Arg=1);
                if( Tc_i < Tc_i-1 )
                        if( Tc_i <BestTc ){
                                bestSolution = solution_i ;
                                BestTc=Tc_i ;
                        } // End if
                else{
                        if( Rand < psh ){
                                Shaking; // shaking is  sub local search2(fchs, Arg=2)
                                if( Tc_i <BestTc ){
                                        bestSolution = solution_i ;
                                        BestTc=Tc_i ;
                                } // End if
                        } // End if
                } // End else
} //End for }
```

Fig 5. Pseudo code for improving algorithm

When *sub local search 2* is completed, if layout cost is increased compare to previous iteration, i.e. $Tc_i > Tc_{i-1}$, *shaking* subroutine is likely executed (probability of execution is symbolized by *psh*) which is designed to escape from local optimums.If the cost is less than former execution, the loop is truncated and it moves to the next iteration. In addition, if the current layout cost (*Solution*$_i$) is less than best available layout (*best Solution*), the best solution is replaced by the current solution. Finally, if the iteration number is equal to a predefined number (*plo*), the algorithm is terminated.

4-2-1- Sub local search 1 subroutine

In figure 6, *sub local search 1* subroutine is shown. It is aimed to find the best location for facilities in the current layout. At first, current solution or *solution*$_i$ is temporarily saved in *solution*^{t}. Then, facilities are relocated one at a time so that its relative costs are minimized. In each iteration, if $Tc_i > Tc^t$, the current layout is returned to the former saved layout in *solution*^{t}. Else if $Tc_i < Tc^t$, *solution*^{t} is updated. In this subroutine, fne_{jkh} is the binary variable, indicating the presence of the j^{th} facility in the k^{th} structure of the h^{th} cluster.

```
Sub local search1(){
    Solution^t = solution_i ;
    Tc^t = Tc_i ;
    For( j=1 ; j ≤ m ; j++ ){
        If( Fne_jkh=1 ){
            Remove facility j from current location then place in the best location.
            Calculate TC_i.
            If( Tc_i<Tc^t){
                Solution^t = solution_i ;
                Tc^t = Tc_i ;
            } // End if
            else{
                solution_i =Solution^t;
                Tc_i=Tc^t  ;
            } // End else
        } // End if
    } // End for}
```

Fig 6. *Sub local search 1* subroutine

4-2-2- Sub local search 2 subroutine

This subroutine is developed for improving the facility layout. As in figure 7, in each run, *fch* number of facilities are excluded from the layout and relocated in the least cost possible place. For supporting *Shaking* algorithm, at times of retrieval a statement is sent to *sub local search 2.*If this statement has the value of 2, *Shaking* is executed. At the beginning of the program, the current solution, (*solution*$_i$) is temporarily saved in *solution*^{t} . Then the *Arg* statement is checked. Since its primary value is 1 and not 2, *fch* facilities are randomly selected and excluded from the layout. Identifications of these facilities are saved in $facility_{fnb}$($fnb=1,\ldots,fch$) in order. In the next step, each of the selected facility is placed in its best location. If no facility is found to have the required conditions, the subroutine is run again with a new set of *fch* facilities. The subroutine is terminated when the costs are reduced or the number of iteration is over the limit. For escaping the local optimums, the solutions with higher costs might be accepted by a probability equal to *p*.

```
Sub local search2(){
    For( itt=1 ; itt<pli ; itt++)
        Solution^t = solution_i;
        If( Arg==2){
            solution_i=BestSolution ;
            for( h=0 ; h ≤ H ; h++){
                ks_h=Rand;
                Update cost and list of facilities considering cluster information.
```

```
        } // End if
    } //End if
    else{
        solution_i = Solution^t ;
        Randomly, select fchs facility and removed from layout. Save number of
        facility in the array Facility_i.
    }
    For( fnb = 1 ; fnb ≤ fch ; fnb++){
        j=Facility_fnb ;
        placingj facility in the best location, in according with the condition of the
        placement.
        If( solution is not feasible )
            break;
    } // End for
    If( solution is feasible ){
        CalculateTc_i.
        If( Tc_i<Tc_i-1 or Rand<p )
            break;
    }
} // End for }
```

Fig 7. *Sub local search 2* subroutine

4-2-3-Shaking subroutine

This subroutine is similar to *sub local search 2*unless the number of selected facilities, *fschs*, is different. Another distinction is the random structure which is chosen for the clusters. This subroutine is run on the best available layout and it can effect on the selected structure of the cluster as well as relocations of the facilities. In figure6, *Shaking* subroutine is recalled by *sub local search 2* and *Arg*=2.

4-2-4- Split groups

The number of intersection points can reach to a square function of the number of facilities and I/O points. As the population of the points grows, searching process expands and hence the execution time of the algorithm increases.

A solution to this difficulty can be the elimination of points to those facilities which are related to the locating facility. The refined set of facilities is called a split group.In this technique which is shown in figure8, each facility belongs to a split group based on the cluster structure. The number of members in the i^{th} split group is limited to a minimum, equal to *fng* and a maximum, equal to all facilities directly related to i^{th} facility.

```
                    Split group formation (){
For( i = 1 ; i ≤ M ; i++){
    For( u = 1 ; u<fng; u++){
        For( j = 1 ; j ≤ M; j++){
            if( Cm_ij> 0){
                Gp_iu = j ;
                u = u + 1 ;
            } //End if
        } // End for
        Temp = 0;
        For( j = 1 ; j<M; j++){
            if( temp < ∑_{j,q} Cm_iq , ∀q: j ≠ Gp_iq , q = 0, 1, …, u − 1)
```

$$temp = \sum_{j,q} \mathrm{Cm}_{iq} \ , \forall q : j \neq Gp_{iq} \ , q = 0, 1, \dots, u-1;$$

} // End for

} // End for

} // End for

}

Fig 8. Pseudo code for split group formation

In the above chart, Gp_{iu} is the code of the u^{th} facility belong to the i^{th} split group and Cm_{ij} is the total cost of the connections between i^{th} and j^{th} facilities obtained by equation (47).

$$Cm_{ij} = \sum_{io,ji} (COI_{i,io,j,ji} + COI_{j,ji,i,io}) + \sum_{io,ji,k,h} (COIH_{i,io,j,ji,k,h} + COIH_{j,ji,i,io,k,h}) : \forall i, j \tag{47}$$

In formation of the i^{th} group, initially all of the related facilities to i^{th} facility, i.e. Cm_{ij}>0 are chosen. Then, if the population of the split group is less than *fng*, among the rest of facilities, those facilities which have the higher relations to the split group are selected. Afterwards, the refined intersection points are extracted. These set of points are formed by S_{xoi} and S_{yoi} from I/O coordination which are directly connected to i^{th} facility ($Cm_{ij}>0$). The S_x and S_y point sets include the workshop boundaries latitude and longitude and facility corners coordination belong to the i^{th} split group. Finally, the coordination of the split group intersection points are obtained.

4-2-5- Tabu list

In order to escape the local optimum, one solution is generating a list by the length of *ptb*, which stores the previous solutions for each iteration. It can avoid repeating the former layout solutions using the positions and orientations of the facilities. In order to check whether a layout is repeated, an index is evaluated by equation (48).

$$indx_i = Tc_i + \sum_{j}^{m} x_j y_j rotate_j \quad , \qquad \forall i \tag{48}$$

Here, Tc_i is the cost of the i^{th} layout by the algorithm; $rotate_j$ is the rotation value of the j^{th} facility (1, 2, 3 and 4 for 0, 90, 180 and 270 degrees respectively). In the Tabu list, the *ptb* value of the last layout is saved. If $index_i$ of the current layout is equal to the $index_i$ of any of the Tabu list items, this layout is repeated before and must be eliminated.

5-Numerical examples

In order to compare the findings of this research with other research, an example by Papageorgiou and Rotsteinis considered (Papageorgiou and Rotstein ,1998). It has 11 stations and its connecting structure is presented in figure 9. Flow of material and the dimensions are listed in tables 3 and 4. They assumed that I/Os are placed in the center of each station and solved the model based on the assumptions that flow diagrams and candidate groups are diverse and arbitrary, a few of which are presented in figure 10. The optimum structure is selected along with the optimization of the layout.

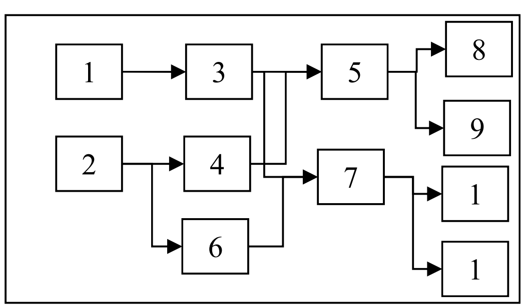


Fig 9. Default connecting structure

connection costs			*connection costs*		
i	***j***	C_{ij}	***i***	***j***	C_{ij}
1	3	1	**6**	7	5
2	4	20	**5**	8	10
2	6	5	**5**	9	10
3	5	10	**7**	10	1
3	7	1	**7**	11	1
4	5	20			

Dimension of facility			*Dimension of facility*		
j	a_j	b_j	*j*	a_j	b_j
1	5	3	7	5	5
2	6	6	8	5	3
3	6	6	9	6	6
4	5	5	10	2	1
5	6	6	11	3	2
6	4.5	4.5			

Table 3. Facilities information

The optimum objective function for the information given in Table4 is obtained equal to 470.

Coordinate of facility				*Coordinate of facility*			
j	x_j	y_j	r_j	***j***	x_j	y_j	r_j
1	3	9.5	0	7	14.5	7.75	0
2	9	3	0	8	13.5	14	1
3	3	14	0	9	9	20	0
4	9	8.5	0	10	15.5	11.25	1
5	9	14	0	11	18	4.25	0
6	14.25	3	0				

Table4. Optimum facility locations

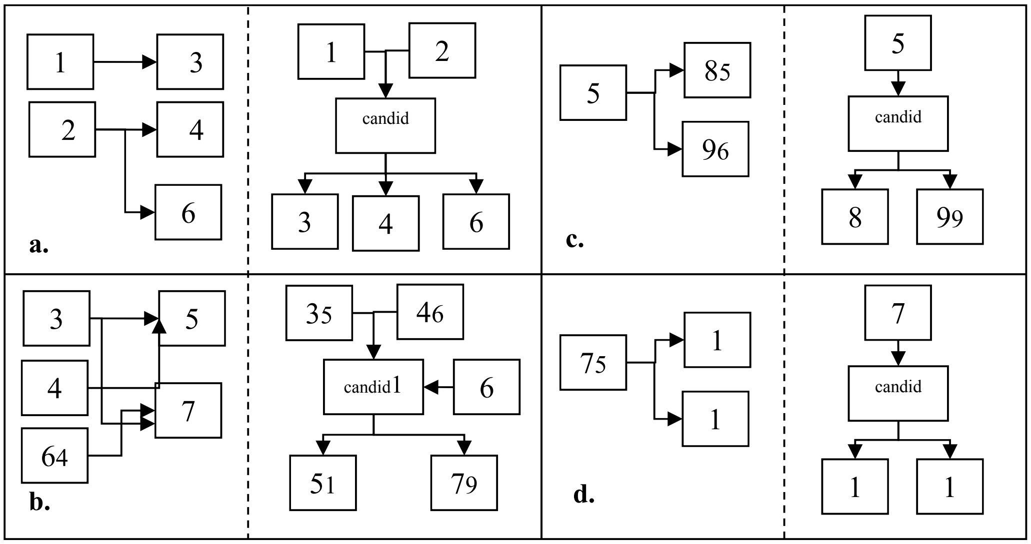


Fig10. Suggested connecting structure

In the current research, the above example is reconsidered for two candidate structures in table 5 and the cost of each structure is evaluated. Then it is solved both by the method of exact and the technique presented in this paper and a summary of the final results are presented in Table 6 and 7 respectively. For both methods, ILOGCPLEX 12.1 software on a computer with a Core(TM) i7 CPU2.1 GHz processor and 64GBRAM is used. The first approach reaches to the optimum solution of 451 in 461.5 sec. In this solution, in group 1 and 2, structure 2 and 1 are selected respectively ($w_{11}=0$,$w_{12}=1$, $w_{21}=1$, $w_{22}=0$) and station 12 is assigned.
Using the heuristic method suggested in this paper, the example is solved in 11.94sec with objective function equal to 451 which supports the merit of the proposed algorithm.

Table 5. Cluster structures suggested by process designers

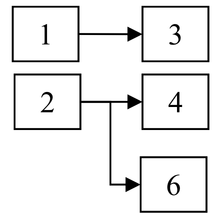
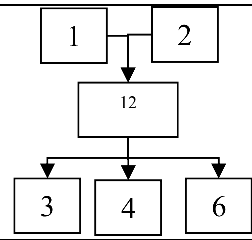
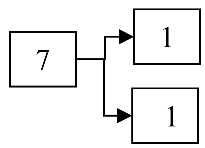
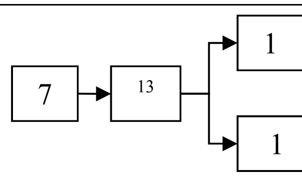
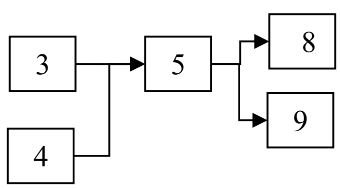
<table>
<tr><th colspan="5">Cluster information</th></tr>
<tr><th>h</th><th>k</th><th>π_{hk}</th><th>connection structures</th><th>connection costs ($COIH_{j,i,j',i',k,h}$)</th></tr>
<tr><td rowspan="2">1</td><td>1</td><td>0</td><td></td><td><table><tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>$COIH$</th></tr><tr><td>O_{11}</td><td>I_{31}</td><td>1</td></tr><tr><td>O_{21}</td><td>I_{41}</td><td>20</td></tr><tr><td>O_{21}</td><td>I_{61}</td><td>5</td></tr></table></td></tr>
<tr><td>2</td><td>0</td><td></td><td><table><tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>$COIH$</th></tr><tr><td>O_{11}</td><td>$I_{12,1}$</td><td>20</td></tr><tr><td>O_{21}</td><td>$I_{12,1}$</td><td>20</td></tr><tr><td>$O_{12,1}$</td><td>I_{41}</td><td>1</td></tr><tr><td>$O_{12,1}$</td><td>I_{41}</td><td>20</td></tr><tr><td>$O_{12,1}$</td><td>I_{51}</td><td>5</td></tr></table></td></tr>
<tr><td rowspan="2">2</td><td>1</td><td>0</td><td></td><td><table><tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>$COIH$</th></tr><tr><td>O_{71}</td><td>$I_{10,1}$</td><td>1</td></tr><tr><td>O_{71}</td><td>$I_{11,1}$</td><td>1</td></tr></table></td></tr>
<tr><td>2</td><td>0</td><td></td><td><table><tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>$COIH$</th></tr><tr><td>O_{71}</td><td>$I_{13,1}$</td><td>1.5</td></tr><tr><td>$O_{13,1}$</td><td>$I_{10,1}$</td><td>1</td></tr><tr><td>$O_{13,1}$</td><td>$I_{11,1}$</td><td>1</td></tr></table></td></tr>
<tr><th colspan="5">other connections</th></tr>
<tr><th colspan="4">connection structures</th><th>connection costs ($COI_{j,i,j',i'}$)</th></tr>
<tr><td colspan="4"></td><td><table><tr><th>O_{ji}</th><th>$I_{j'i'}$</th><th>COI</th></tr><tr><td>O_{31}</td><td>I_{51}</td><td>10</td></tr><tr><td>O_{41}</td><td>I_{51}</td><td>20</td></tr><tr><td>O_{51}</td><td>I_{81}</td><td>10</td></tr><tr><td>O_{51}</td><td>I_{91}</td><td>10</td></tr></table></td></tr>
</table>

Table 6. Optimum locations using candidate groups

Coordinate of facility				Coordinate of facility			
j	x_j	y_j	r_j	j	x_j	y_j	r_j
1	14.75	8.5	·	8	5.25	14	·
2	3	26	·	9	9.75	20	·
3	15.75	14	·	10	17.5	0.5	·
4	9.75	8.5	·	11	21	3.5	·
5	9.75	14	·	12	12.75	8.5	·
6	12.75	3.5	·	13	–	–	–
7	17.5	3.5	·				

Table 7. Optimum locations using heuristic algorithm

Coordinate of facility				*Coordinate of facility*			
j	x_j	y_j	r_j	j	x_j	y_j	r_j
1	6.75	10	2	8	16.25	15.5	2
2	14.25	3	1	9	11.25	20	1
3	5.25	14	1	10	5.5	3	1
4	11.75	9	1	11	1.5	5.5	2
5	11.25	14	1	12	10.75	11	1
6	9	4.5	1	13	–	–	-
7	4	4	1				

The next example is considered to be a 30x35m rectangular workshop with 9 facilities specified in table 8.

Table 8. Facilities information

Dimension of facility

j	a	b
1	7	12
2	6	9
3	6	9
4	6	9
5	11	9
6	10	8
7	10	10
8	9	12
9	4	4

Specifications of input

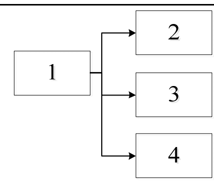
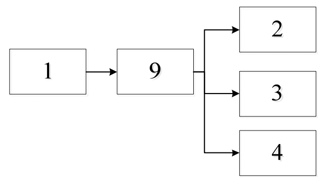
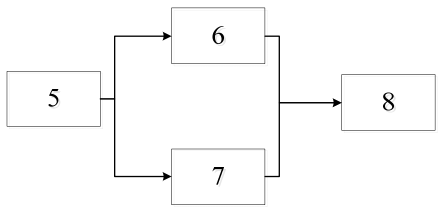
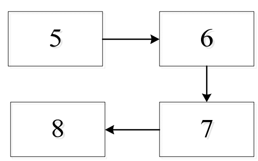
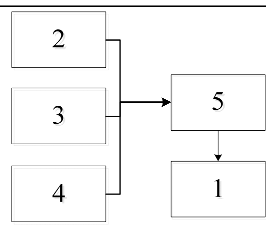
O_{ji}	Δx	Δy
O_{11}	-3.5	0
O_{21}	-3	0
O_{31}	-3	0
O_{41}	-3	0
O_{51}	-5.5	0
O_{61}	-5	0
O_{71}	-5	0
O_{81}	-4.5	0
O_{91}	-2	0

Specifications of outputs

I_{ji}	Δx	Δy
I_{11}	3.5	0
I_{21}	3	0
I_{31}	3	0
I_{41}	3	0
I_{51}	5.5	0
I_{52}	0	-4.5
I_{61}	5	0
I_{71}	5	0
I_{81}	2	0

The connecting structures are categorized in two clusters. Cluster No.1 consists of two structures and a cluster centre, and cluster No. 2 contains two structuresbut has no cluster centre. Information of the clusters and the facilities inter-connections are graphed and listed in table 9. The cost saving by structure No. 1 is 120.

Table 9. Connection structures suggested by process design department

Cluster information				
h	k	π_{hk}	*connection structures*	*connection costs* ($COIH_{j,i,j',i',k,h}$)
1	1	0		O_{ji} / $I_{j'i'}$ / $COIH$ O_{11} / I_{21} / 20 O_{11} / I_{31} / 20 O_{11} / I_{41} / 20
	2	120		O_{ji} / $I_{j'i'}$ / $COIH$ O_{11} / I_{81} / 32 O_{91} / I_{21} / 20 O_{91} / I_{31} / 20 O_{91} / I_{41} / 20
2	1	0		O_{ji} / $I_{j'i'}$ / $COIH$ O_{51} / I_{61} / 14 O_{51} / I_{71} / 14 O_{61} / I_{81} / 15 O_{71} / I_{81} / 15
	2	0		O_{ji} / $I_{j'i'}$ / $COIH$ O_{51} / I_{61} / 20 O_{61} / I_{71} / 21 O_{71} / I_{81} / 19
other connections				
connection structures				*connection costs* ($COI_{j,i,j',i'}$)
				O_{ji} / $I_{j'i'}$ / COI O_{52} / I_{11} / 13 O_{21} / I_{51} / 16 O_{31} / I_{51} / 16 O_{41} / I_{51} / 16

Clusters No.1 and No. 2 are randomly selected by the algorithm and the data are updated in table 10.

O_{ji}	$I_{j'i'}$	COI		O_{ji}	$I_{j'i'}$	COI
O_{11}	I_{21}	20		O_{71}	I_{81}	19
O_{11}	I_{31}	20		O_{52}	I_{11}	13
O_{11}	I_{41}	20		O_{21}	I_{51}	16
O_{51}	I_{61}	20		O_{31}	I_{51}	16
O_{61}	I_{71}	21		O_{41}	I_{51}	16

Table 10. Updated inter-connection costs

Since structure No.1 is selected for cluster No.1 and facility No. 9 does not exist in that structure, this facility is eliminated from the model and 8 other facilities remain. Then, facility No. 6 is selected by chance and is placed at (100,100) position. Among the rest of unallocated facilities, facility No. 7 is selected for location, because it has the maximum connection cost to facility No. 6. This algorithm continues until all facilities are located, as shown in figure 11. Orientation of the numbers indicates the rotation of the facilities from their original status. The cost of this layout is 1532.5 and the solution has been achieved in 0.006sec.

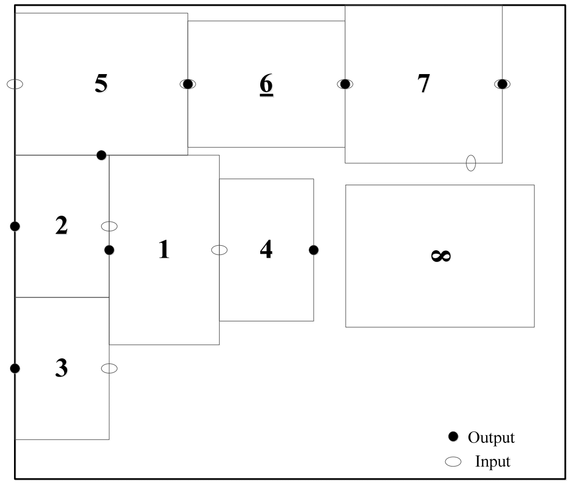


Figure 11. Final layout by constructive algorithm

6-Results comparison and discussion

The performance of the heuristic algorithm has been compared with the MIP exact method. Table 11comprises the specification of 18 test problems with their solutions obtained by both methods. It reveals that for any problem with less than 15 facilities, MIP has reached to the global optimum solution. However, for larger problems due to long processing time, the CPLEX has been stopped after 7200sec.The heuristic algorithm has been solved 5 times for each problem and a summary of the results are listed in Table11. The columns of *Best T*C, *Worst T*C and *Ave.T*C are representing the best, the worst and the average of the 5 solution runs. *Ave. time* is the average of the processing time of 5 solution runs. For CPLEX results, *T*C designates the resulting objective function and *Opt. Gap* is the gap between the result and the optimum solution.

Table 11. Comparison of the proposed algorithm with CPLEX results

Problem Id.	m	H	Heuristic algorithm					CPLEX			δ
			Best TC	Average TC	Worst TC	Ave. time (sec)	Var	TC	Opt Gap	Time (sec)	
I8-1	8	2	1010.0	1010	1010.0	21.97	-	1010.0	0%	48.559002	0
I8-2	8	1	1028.0	1028	1028.0	20.66	-	1028.0	0%	32.504997	0
I9-1	9	1	607.0	608.8	616.0	32.32	4.02	607.0	0%	4942.51416	0
I9-2	9	2	614.0	614	614.0	28.57	-	614.0	0%	9.065001	0
I11-1	11	1	823.0	823	823.0	71.84	-	823.0	0%	27008.3789	0
I11-2	11	1	807.0	862.5	900.5	84.03	37.46	807.0	0%	675.960999	0
I13-1	13	1	824.5	827.5	839.5	66.75	6.71	824.5	0%	10402.4502	0
I13-2	13	2	768.0	768	768.0	93.48	-	768.0	0%	12152.2	0
I15-1	15	1	1301.0	1309.6	1344.0	135.22	19.23	1336.0	50.1%	7200	-2.7%
I15-2	15	2	1145.5	1152.7	1154.5	224.06	4.02	1126.5	17.2%	2759.83	1.7%
I19-1	19	1	661.0	681.8	716.0	154.32	23.99	1027.0	91.3%	7200	-55.4%
I19-2	19	2	879.5	906.3	950.5	217.40	2.39	1126.5	86%	7200	-28.1%
I23-1	23	2	1546.5	1698.7	1820.5	363.71	117.24	1917.5	100%	7200	-24.0%
I23-2	23	1	976.5	1006.1	1039.0	329.98	26.69	1218.5	100%	7200	-24.8%
I27-1	27	3	986.5	1088.1	1121.5	413.10	7.23	1287.0	100%	7200	-30.5%
I27-2	27	2	1007.5	1070.5	1130.5	429.50	58.37	1339.0	100%	7200	-32.9%
I30-1	30	2	872.0	907.9	926.5	596.99	22.43	1239.5	100%	7200	-42.1%
I30-2	30	3	814.5	872.7	889.5	655.74	2.73	1113.0	100%	7200	-36.6%

In table 11, δ is the deviation percentage between the CPLEX results and the best solution of the algorithm. The negative values indicate the preference of the algorithm results. It can be found from Table 11 that for all of the problems in which CPLEX has merged to an optimum solution; our algorithm has reached to that solution in much shorter time (less than 100sec). For the rest of the problems, the best solution of the algorithm is up to 55% better than CPLEX results, unless for problem I15-2 which has a final solution 1.7% worse than CPLEX solution. For problems with more than 15 facilities, even the worst solution of the algorithm is better than CPLEX results. In terms of processing time, the proposed algorithm is faster than CPLEX program for all 18 problems, as can be seen in table 11.

7-Conclusion

In this paper, a hybrid approach for facility layout and cluster configuration was presented. A set of cluster structures were initially proposed by process designers. This information together with other data for conventional layout model was used to find an optimum solution both for the cluster configuration and facility layout. Mixed integer programming by CPLEX software has been used for this purpose, albeit time consuming and inefficient. Therefore, a heuristic algorithm was developed consisting two sub-algorithms. The Constructive algorithm found the initial layouts and the improving algorithm relocated the facilities for enhancing the layout. Other techniques such as shaking, systematic intersection point generation, split groups, Tabu lists were also implemented in the algorithm. Then, two different examples were solved and the comparison of the results supported the merit of the proposed algorithm. For further validation, 18 different problems were defined and solved both with the heuristic algorithm and MIP model by CPLEX. It was proved that up to 13 facilities, the algorithm has reached to same solutions in much shorter time. For the rest of the problems, the best solution of the algorithm is generally up to 55% better than CPLEX results.

References

Aiello, G., La Scalia, G., & Enea, M. (2012). A multi objective genetic algorithm for the facility layout problem based upon slicing structure encoding. Expert Systems with Applications, 39(12), 10352-10358.

Anjos, M. F., & Vieira, M. V. (2016). An improved two-stage optimization-based framework for unequal-areas facility layout. Optimization Letters, 10(7), 1379-1392.

Barbosa-Povoa, A. P., Mateus, R., & Novais, A. Q. (2001). Optimal two-dimensional layout of industrial facilities. International Journal of Production Research, 39(12), 2567-2593.

Barbosa-Póvoa, A. P. (2007). A critical review on the design and retrofit of batch plants. Computers & Chemical Engineering, 31(7), 833-855.

Barbosa-Póvoa, A. P., Mateus, R., & Novais, A. Q. (2002). Optimal 3D layout of industrial facilities. International Journal of Production Research, 40(7), 1669-1698.

Bland, J. A., & Dawson, G. P. (1994). Large-scale layout of facilities using a heuristic hybrid algorithm. Applied Mathematical Modelling, 18(9), 500-503.

Bock, S., & Hoberg, K. (2007). Detailed layout planning for irregularly-shaped machines with transportation path design. European Journal of Operational Research, 177(2), 693-718.

Castillo, I., & Westerlund, T. (2005). An ε-accurate model for optimal unequal-area block layout design. Computers & Operations Research, 32(3), 429-447.

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

Chwif, L., Barretto, M. R. P., & Moscato, L. A. (1998). A solution to the facility layout problem using simulated annealing. Computers in industry, 36(1), 125-132.

Deisenroth, M. P., & Apple, J. M. (1972). A computerized plant layout analysis and evaluation technique. In Annual AIIE Conference.

Drira, A., Pierreval, H., & Hajri-Gabouj, S. (2007). Facility layout problems: A survey. Annual Reviews in Control, 31(2), 255-267.

Farahani, R. Z., Hekmatfar, M., Arabani, A. B., & Nikbakhsh, E. (2013). Hub location problems: A review of models, classification, solution techniques, and applications. Computers & Industrial Engineering, 64(4), 1096-1109.

Francis, R. L., McGinnis, L. F., & White, J. A. (1992). Facility layout and location: an analytical approach. Pearson College Division.

Hassan, M. M., & Hogg, G. L. (1987). A review of graph theory application to the facilities layout problem. Omega, 15(4), 291-300.

Hu, G. H., Chen, Y. P., Zhou, Z. D., & Fang, H. C. (2007). A genetic algorithm for the inter-cell layout and material handling system design. The International Journal of Advanced Manufacturing Technology, 34(11), 1153-1163.
Jankovits, I., Luo, C., Anjos, M. F., & Vannelli, A. (2011). A convex optimisation framework for the unequal-areas facility layout problem. European Journal of Operational Research, 214(2), 199-215.

Kim, J. G., & Kim, Y. D. (1999). A branch and bound algorithm for locating input and output points of departments on the block layout. Journal of the operational research society, 50(5), 517-525.

Koopmans, T. C., & Beckmann, M. (1957). Assignment problems and the location of economic activities. Econometrica: journal of the Econometric Society, 53-76.

K. Y. Tam, Genetic algorithms, function optimization, and facility layout design. European Journal of Operational Research, 63(1992) 322–346.

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.

Leno, I. J., Sankar, S. S., Raj, M. V., & Ponnambalam, S. G. (2013). An elitist strategy genetic algorithm for integrated layout design. The International Journal of Advanced Manufacturing Technology, 66(9-12), 1573-1589.

Loiola, E. M., de Abreu, N. M. M., Boaventura-Netto, P. O., Hahn, P., & Querido, T. (2007). A survey for the quadratic assignment problem. European journal of operational research, 176(2), 657-690.

Meller, R. D., Narayanan, V., & Vance, P. H. (1998). Optimal facility layout design. Operations Research Letters, 23(3), 117-127.

Montreuil, B. (1991). A modelling framework for integrating layout design and flow network design. In Material Handling'90 (pp. 95-115). Springer, Berlin, Heidelberg.

Moran, S. (2015). An applied Guide to process and plant design. Butterworth-Heinemann.

Moslemipour, G. (2017). Robust inter and intra-cell layouts design model dealing with stochastic dynamic problems. Journal of Industrial and Systems Engineering, 10(4), 123-140.

Papageorgiou, L. G., & Rotstein, G. E. (1998). Continuous-domain mathematical models for optimal process plant layout. Industrial & engineering chemistry research, 37(9), 3631-3639.

Patsiatzis, D. I., Xu, G., & Papageorgiou, L. G. (2005). Layout aspects of pipeless batch plants. Industrial & engineering chemistry research, 44(15), 5672-5679.

Penteado, F. D., & Ciric, A. R. (1996). An MINLP approach for safe process plant layout. Industrial & engineering chemistry research, 35(4), 1354-1361.

Realff, M. J., Shah, N., & Pantelides, C. C. (1996). Simultaneous design, layout and scheduling of pipeless batch plants. Computers & chemical engineering, 20(6-7), 869-883.

Sharma, P., & Singhal, S. (2016). A review of objectives and solution approaches for facility layout problems. International Journal of Industrial and Systems Engineering, 24(4), 469-489.

Shayan, E., & Chittilappilly, A. (2004). Genetic algorithm for facilities layout problems based on slicing tree structure. International Journal of Production Research, 42(19), 4055-4067.
Sherali, H. D., Fraticelli, B. M., & Meller, R. D. (2003). Enhanced model formulations for optimal facility layout. Operations Research, 51(4), 629-644.

Scholz, D., Jaehn, F., & Junker, A. (2010). Extensions to STaTS for practical applications of the facility layout problem. European Journal of Operational Research, 204(3), 463-472.

Scholz, D., Petrick, A., & Domschke, W. (2009). STaTS: a slicing tree and tabu search based heuristic for the unequal area facility layout problem. European Journal of Operational Research, 197(1), 166-178.

Taghavi, A., & Murat, A. (2011). A heuristic procedure for the integrated facility layout design and flow assignment problem. Computers & Industrial Engineering, 61(1), 55-63.