Zoning Constrained Machine Layout Problem with Mutual Clearances

Zeynep Uruk* and Nevra Akbilek

Department of Industrial Engineering, Sakarya University, Sakarya, Turkey

Abstract

In this paper, a single row machine layout problem is considered with zoning constraints and mutual clearances under an enhanced objective of minimizing material flow cost and machine installation cost. The problem is restricted by positive and negative zoning constraints to represent real life problems. Moreover, clearances needed between machine pairs are divided into two types, which are must and extra clearances, and extra clearances are reduced by mutual use between adjacent machines to decrease material flow costs. Objective function also considers fixed costs of locating machines which usually neglected in machine layout problems in literature but a necessity in real life problems. Two mathematical models are formulated, which are nonlinear and linear mixed integer programs, to solve the problem optimally and to compare the effect of linearity/nonlinearity in mathematical programming formulations in terms of solution quality and time. The mathematical models are not effective in terms of time for large problem instances; therefore, a genetic algorithm is proposed generating high quality solutions in reasonable time. It is shown that the genetic algorithm outperforms both the nonlinear and linear mathematical models with lower cost and shorter time.

Key words: Restricted single row machine layout problem, Zoning constraints, Mutual clearances, Machine installation cost, Flexible manufacturing systems

1. Introduction

In this research, a restricted single row machine layout problem with mutual clearances is studied. The objective function is enhanced by considering machine installation cost in addition to minimizing material flow cost. There exist restrictions on the placement of machines, namely positive and negative zoning constraints. Moreover, two types of clearances, which are must and extra clearances, are considered between machines and extra clearance is used mutually between adjacent machines. There are several studies on zoning constraints and mutual clearances separately in the literature. However, this is the first research which simultaneously addresses zoning constraints and mutual clearances.

A single-row machine layout is the placement of machines on a straight line which is widely used in flexible manufacturing systems (FMS). There are five layout types in FMS using material handling devices in practice [1]. These are single row layout(a), single cell layout(b), loop layout(c), double row layout(d) and cluster layout(e) which can be seen in Figure 1. Single cell layout(b) and loop layout(c) can also be taken as a single row layout, that encourages the study of single row layout.

Braglia [2] used genetic algorithm to minimize backtracking for single row machine layout problem in FMS. Braglia [3] also presented several algorithms for a single row flexible manufacturing system for the n/m/F/Fmax problem. Ramkumar and Ponnambalam [4] proposed a genetic algorithm and simulated annealing algorithm for the single row machine layout design in FMS. Ficko et al. [5] designed a model with genetic algorithm for single and multiple rows layout in FMS. Solimanpur et al. [6] formulated a nonlinear mathematical model and developed an ant colony algorithm with sequence dependent machine distances regarding this subject. Ozcelik [7] applied a hybrid genetic algorithm with a local search method for single row layout problem with different machine sizes and clearances in FMS. Jahromi et al. [8] studied a modified Evolutionary Algorithm and an Evolutionary Algorithm with the island model for scheduling of an FMS considering operation allocation and dynamic machine-tool selection. Tubaileh and Siam [9] discussed single, double and multi-row layouts in FMS to minimize the total cost of material transportation between machines using Ant Colony and Simulated Annealing algorithms. Rai and Jayswal [10] considered the loop layout in FMS to minimize the number of machines which the part types cross and solved the problem using particle swarm optimization algorithm. Delavar et al. [11] considered a bi-objective mathematical model to design a four-dimensional cellular manufacturing system under the objective of minimizing total costs and maximizing skill level of operators. They offered a multi-objective vibration damping optimization algorithm, a non-dominated sorting genetic algorithm, a multi-objective particle swarm optimization, and a multi-objective invasive weeds optimization. Yadav and Jayswal [12] addressed analysis and modelling of an FMS problem for performance enhancement. The effect of layout type, part mix, and batching condition on system performance was investigated in terms of productivity, system utilization rate, and cycle time using a simulation analysis and Taguchi's experimental design. Hafiz et al [13] studied a mixed-model carousel-based FMS by means of Coloured Petri net approach, the design of experiment and response surface methods to minimize cycle time and work-in-process and to maximize throughput.

There are several studies in the literature using genetic algorithm to solve single row machine layout. Braglia [2], Ramkumar and Ponnambalam [4], Ficko et al. [5] and Ozcelik [7] are mentioned above. Ponnambalam and Ramkumar [14] analysed flow path characteristics on a single row layout using genetic algorithm. Lin [15] offered a hierarchical order-based genetic algorithm. Datta et al. [16] used a permutation-based genetic algorithm by combining rule based and random permutation with proposed crossover and mutation operators in unconstrained single row layout problem. Lenin et al. [17] developed a genetic algorithm for multi-products of different process sequences under a multi objective function which minimizes the total flow, investment cost and machines arranged in single row layout. Kothari and Ghosh [18] proposed an efficient genetic algorithm which improves some benchmark instances in literature. Ghadirpour et al. [19] proposed three mixed integer nonlinear mathematical models and genetic algorithm to solve unequal–area stochastic dynamic facility layout problems with routing flexibility.

The single row layout problem is usually considered as unconstrained in the literature, in which the machines or facilities can be located without any restriction. However, in manufacturing environments mostly machines can't be located to any of the available sites, but there exist restrictions on the locations of machines. Some of these restrictions are named as zoning constraints which are considered in this study. When a positive zoning constraint exists for some machines, then these machines must be placed next to each other because of operational/technical dependencies or safety considerations. In case of a negative zoning constraint for some machines, these machines can't be located near each other because the processing of some machine might affect the processing of other machines. Kouvelis et al. [20] studied a single row machine layout problem using QAP formulation by proper modifications to account for zoning constraints. They proposed compulsion and penalty procedures for simulated annealing algorithm. Wang et al. [21] constructed a model for intra cell and inter cell layout problems in cellular manufacturing systems to minimize the total distance of material handling. They proposed an improved simulated annealing algorithm with a neighbourhood generation mechanism which satisfies zoning constraints. Brunese and Tanchoco [22] addressed the single row layout model with an implied within-building constraint. They evaluated the impact of this constraint on several benchmark problems using the time and objective function value by using linear and nonlinear mixed-integer formulations. Kalita and Datta [23] studied a single row layout problem with ordering and positioning constraints. Addition to the mathematical model, they offered a permutation-based genetic algorithm to minimize the overall material handling cost between the facilities. Liu et al. [24] also considered a constrained single row facility layout problem with ordering, positioning, and relation constraints. They proposed mixed-integer programming models and an improved fireworks algorithm to minimize the material handling cost. Yang and Utamima [25] searched single row facility layout problem by studying fixed cost of assigning facilities and safety constraints. They presented a hybrid estimation of distribution algorithm, tabu search and particle swarm optimization. They also built a hybrid genetic algorithm as benchmark. This study also considers the clearance between the two facilities differently from above studies. Maier and Taferner [26] considered the same problem as Li et al. [24] and offered a new integer linear model which was the best exact approach in literature. Kalita and Datta [27] studied a constrained single-row facility layout problem which requires the placement of certain facilities in set places and/or in specified orders, with or without permitting the placement of any other facility between two ordered facilities. They implemented a permutation-based genetic algorithm with some repairing mechanisms. Coppé et al. [28] proposed a mixed integer model and a decision diagram-based approach for this problem.

Clearances are needed between machine pairs for operative considerations and safety issues, namely, maintenance, ventilation, avoiding of unwanted interactions between machines such as vibrations and/or emissions, work in process storage and enough space for workers. In literature, some studies do not consider machine dimensions or clearances, and some consider equal clearances between machine pairs, which are not realistic scenarios. In some papers, variable clearances between every pair of machines are taken into consideration either symmetrically or asymmetrically. In symmetric case, the clearance between adjacent machines (i,j) doesn't depend on machine sequence. In asymmetric case, the clearance between adjacent machines (i,j) differ from the clearance between adjacent machines (j,i). Manzke et al. [29] studied an artificial bee colony algorithm to solve the single row layout problem with sequence-dependent asymmetric clearances. Safarzadeh and Koosha [30] considered a multi-row facility layout problem with fuzzy clearances to minimize the material handling and lost opportunity costs. They modelled a nonlinear mixed integer programming with fuzzy constraints and transformed it into a linear mixed integer programming. They also offered a genetic algorithm. Keller [31] proposed a modified mathematical model for the single row layout problem with machine spanning clearances to minimize the weighted sum of distances. Three construction heuristics are also developed to generate an initial solution. In this paper, a different approach is addressed. Defined clearances above can be divided in to two as must and extra clearances. Must clearance describe the space needed for maintenance, ventilation, avoiding of unwanted interactions between machines such as vibrations and/or emissions. Extra clearance is necessary for work in process storage and enough space for workers. Mutual use of extra clearances among adjacent machines is often possible which allows decreasing the distance between machines and hence material flow cost. Yu et al. [32] proposed a tabu search to solve a single row machine layout problem under the objective of minimizing the cost of material flow. They divided clearances in to two, which are minimum and additional clearances, and the latter are shared between adjacent machines. They treated the additional clearance as one or both sides of each facility and it can be located either on the left or right side of machine for one side type. They applied a heuristic rule to determine the position of one-sided additional clearance for each facility. Zuo et al. [33] considered the same concept in a double row layout problem with a bicriteria objective of minimizing layout area and material flow cost. A mixed integer linear mathematical model is developed and a hybrid tabu algorithm combined with a heuristic rule is proposed. Akbilek [34] examined a safety measure integrated single-row machine configuration with uneven size and clearance. She suggested a mathematical model, a tabu search algorithm and a genetic algorithm with a novel heuristic rule support for multiple objectives.

This paper fills the gap in literature by considering both zoning constraints and mutual clearance concept at the same time through an enhanced objective function which minimizes material flow costs and machine installation costs.

2. Problem definition and modelling

In this paper, a restricted single row machine layout problem with mutual clearances in flexible manufacturing systems is studied under an enhanced objective function of minimizing material flow costs and machine installation costs.

There exist restrictions on the placement of machines, namely positive and negative zoning constraints. When a positive zoning constraint exists for some machines, then these machines must be placed next to each other. Robots need to be located together in semi-enclosed environment to protect the workers; the machines should be placed as close as possible because of the operations that must strictly follow each other. In case of a negative zoning constraint for some machines, these machines cannot be located near each other because the processing of some machine might affect the processing of other machines. The machines that need fine adjustments cannot be placed near machines having loud noises or extreme vibrations. Because of safety considerations, the machines dealing with flammable material cannot be close to the machine generating high heat. In the problem set, a negative zoning constrained set is defined for machines (i,j) where i and j cannot be adjacent to each other and also a negative zoning constrained set is defined machines (i,j) where i and j have to be adjacent to each other.

Machines are defined to be rectangular with various widths and the distances among the machines are computed regarding their centroids. Clearances needed between machine pairs for operative considerations and safety issues are divided in to two, namely must and extra clearances. Must clearance stand for maintenance, ventilation, avoiding of unwanted interactions between machines such as vibrations and/or emissions. Extra clearance is necessary for work in process storage and enough space for workers. All machines have must clearances in between with respect to the factors mentioned above. Extra clearances depend on the products produced, technical properties of machines and the location of load/unload port of each machine. Some machines need extra clearance on both side of the machine and some machines need it just on the left or right. The location of extra clearances defined cannot be changed because it is not possible to relocate the load/unload port of machine or change the orientation of machine. Extra clearances between adjacent machines are used mutually if both machines need extra clearance on the adjacent sides and maximum of them ($\max\{e^r, e^l\}$) is applied. Figure 2 visualizes the situation.

Objective function is also enhanced by considering installation cost of machines. Machine installation costs may depend on the locations which is mentioned by Sule [35]. Some examples for machine installation costs are as follows: Vibration damper, floor levelling and fortification are needed for press machines. Air, water, oil, and refrigerant lines are installed for some machines that the installation costs directly depend on the location of machines. Although these types of costs are usually neglected during the machine layout design in the literature and the design is based on just material flow between machines, installation costs of machines are huge expenses which may affect the decisions of machine locations.

The notation used throughout the paper is given below:

Decision Variables

$X_{i,a}$: Binary variable that controls if machine i is placed to the location a or not

$d_{a,b}$ ($d_{i,j}$): The cost of material handling based on distance/material flow between locations a and b (machines i and j)

$Z_{i,j}$: Binary variable that controls if machine j is placed immediately to the right of machine i

Co^P_a: Coordinate of position a

Co^M_i: Coordinate of machine i

$MCP_{a,b}$: Must clearance between positions a and b

$ECP_{a,b}$: Extra clearance between positions a and b

Parameters

$f_{i,j}$: Material flow between machines i and j per unit time

w_i: Width of machine i

$c_{i,j}$: Must clearance between machine i and machine j, where machine j is located to the right of machine i

e^l_i : Extra clearance for the left side of machine i

e^r_i : Extra clearance for the right side of machine i

m: The number of machines to be assigned

$IC_{i,a}$: Installation cost of machine i to location a

NZC: A set of machines (i,j) where i and j cannot be adjacent to each other

PZC: A set of machines (i,j) where i and j have to be adjacent to each other

M_1 ,M_2, M_3: Sufficiently big numbers

In this paper, two mathematical programming formulations are proposed, namely nonlinear and linear mathematical model. In machine layout literature, generally one type of mathematical model is formulated. On the contrary, two types are presented and compared in terms of both solution quality and time in this research.

The problem can be formulated as a mixed integer nonlinear mathematical model as given below:

Model 1: Nonlinear Mathematical Model

$$Min \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{a=1}^{m-1}\sum_{b=a+1}^{m}(f_{i,j}+f_{j,i})d_{a,b}\,X_{i,a}X_{j,b} + \sum_{i=1}^{m}\sum_{a=1}^{m} IC_{i,a}\,X_{i,a} \tag{1}$$

st

$$\sum_{i=1}^{m} X_{i,a} = 1 \qquad \forall a \in \{1,2,\ldots,m\} \tag{2}$$

$$\sum_{a=1}^{m} X_{i,a} = 1 \qquad \forall i \in \{1,2,\ldots,m\} \tag{3}$$

$$X_{i,a} \le 1 - \left(X_{j,a-1} + X_{j,a+1}\right) \qquad \forall (i,j) \in NZC \text{ and } \forall a \in \{1,2,\ldots,m\} \tag{4}$$

$$X_{i,a} \le X_{j,a-1} + X_{j,a+1} \qquad \forall (i,j) \in PZC \text{ and } \forall a \in \{1,2,\ldots,m\} \tag{5}$$

$$d_{a,b} = \sum_{i=1}^{m}\sum_{t=a+1}^{b-1} w_i\, X_{i,t} + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{t=a}^{b-1}(c_{i,j} + \max\left\{e_i^r\ ,e_j^l\right\})\, X_{i,t}\, X_{j,t+1} + \sum_{i=1}^{m}\frac{w_i\,(X_{i,a} + X_{i,b})}{2} \quad \forall a,b: 1 \le a < b \le m \tag{6}$$

$$X_{i,a} \in \{1,0\} \qquad \forall i \in \{1,2,\ldots,m\}\, and\ \forall a \in \{1,2,\ldots,m\} \tag{7}$$

The objective function (Equation 1) minimizes the cost which includes material flow and machine installation costs. Constraint 2 (Equation 2) provides that only one machine can be placed to each location. Constraint 3 (Equation 3) ensures that at each machine must be placed to one location. Constraint 4 (Equation 4) considers the negative zoning constraints where NZC= {(i, j): machines i and j cannot be located next to each other}. Constraint 5 (Equation 5) considers the positive zoning constraints where PZC={(i, j): machines i and j have to be adjacent to each other}. Constraint 6 (Equation 6) computes distances between machines considering machine locations, machine widths, must and extra clearances. Constraint 7 (Equation 7) assigns $X_{i,a}$ as a binary variable.

The objective function and constraint 5 (Equation 5) of above model is nonlinear. The nonlinear terms can be linearized by addition of new variables and can be modelled as a mixed integer linear mathematical formulation as follows:

Model 2: Linear Mathematical Model

$$Min \sum_{i=1}^{m}\sum_{j=1}^{m} f_{i,j}\, d_{i,j} + \sum_{i=1}^{m}\sum_{a=1}^{m} IC_{i,a}\, X_{i,a} \tag{8}$$

st

$$\sum_{i=1}^{m} X_{i,a} = 1 \qquad \forall a \in \{1,2,\ldots,m\} \tag{9}$$

$$\sum_{a=1}^{m} X_{i,a} = 1 \qquad \forall i \in \{1,2,\ldots,m\} \tag{10}$$

$$Z_{i,j} + Z_{j,i} = 0 \qquad \forall (i,j) \in NZC \tag{11}$$

$$Z_{i,j} + Z_{j,i} = 1 \qquad \forall (i,j) \in PZC \tag{12}$$

$$Z_{i,j} \ge X_{i,a} + X_{j,a+1} - 1 \qquad \forall i \in \{1,2,\ldots,m\}, \forall j \in \{1,2,\ldots,m\} \setminus i\, and\ \forall a \in \{1,2,\ldots,m-1\} \tag{13}$$

$$\sum_{i=1, i\ne j}^{m} Z_{i,j} = \sum_{a=2}^{m} X_{j,a} \qquad \forall j \in \{1,2,\ldots,m\} \tag{14}$$

$$\sum_{j=1, j\ne i}^{m} Z_{i,j} = \sum_{a=1}^{m-1} X_{i,a} \qquad \forall i \in \{1,2,\ldots,m\} \tag{15}$$

$$Co_1^P = 0 \tag{16}$$

$$Co_a^P = Co_{a-1}^P + MCP_{a-1,a} + ECP_{a-1,a} + \sum_{i=1}^{m}\frac{w_i\,(X_{i,a-1} + X_{i,a})}{2} \qquad \forall a \in \{2,\ldots,m\} \tag{17}$$

$$MCP_{a,a+1} \ge c_{i,j}\, Z_{i,j} - M_1\left(2 - X_{i,a} - X_{j,a+1}\right) \quad \forall i \in \{1,2,\ldots,m\}, \forall j \in \{1,2,\ldots,m\} \setminus i, \forall a \in \{1,2,\ldots,m-1\} \tag{18}$$

$$MCP_{a,a+1} \le c_{i,j}\, Z_{i,j} + M_1\left(2 - X_{i,a} - X_{j,a+1}\right) \quad \forall i \in \{1,2,\ldots,m\}, \forall j \in \{1,2,\ldots,m\} \setminus i, \forall a \in \{1,2,\ldots,m-1\} \tag{19}$$

$$ECP_{a,a+1} \ge e_i^r \, X_{i,a} \qquad \forall i \in \{1,2,\ldots,m\} \, and \, \forall a \in \{1,2,\ldots,m-1\} \tag{20}$$

$$ECP_{a,a+1} \ge e_j^l \, X_{j,a+1} \qquad \forall j \in \{1,2,\ldots,m\} \, and \, \forall a \in \{1,2,\ldots,m-1\} \tag{21}$$

$$\begin{gathered} ECP_{a,a+1} \le \max\left\{e_i^r \, , e_j^l\right\} + M_2\left(2 - X_{i,a} - X_{j,a+1}\right) \\ \forall i \in \{1,2,\ldots,m\}, \forall j \in \{1,2,\ldots,m\} \setminus i, \forall a \in \{1,2,\ldots,m-1\} \end{gathered} \tag{22}$$

$$Co_i^M \ge Co_a^P - M_3\left(1 - X_{i,a}\right) \qquad \forall i \in \{1,2,\ldots,m\} \, and \, \forall a \in \{1,2,\ldots,m\} \tag{23}$$

$$Co_i^M \le Co_a^P + M_3\left(1 - X_{i,a}\right) \qquad \forall i \in \{1,2,\ldots,m\} \, and \, \forall a \in \{1,2,\ldots,m\} \tag{24}$$

$$d_{i,j} \ge Co_i^M - Co_j^M \qquad \forall i \in \{1,2,\ldots,m\} \, and \, \forall j \in \{1,2,\ldots,m\} \setminus i \tag{25}$$

$$d_{i,j} \ge Co_j^M - Co_i^M \qquad \forall i \in \{1,2,\ldots,m\} \, and \, \forall j \in \{1,2,\ldots,m\} \setminus i \tag{26}$$

$$X_{i,a} \in \{1,0\} \qquad \forall i \in \{1,2,\ldots,m\} \, and \, \forall a \in \{1,2,\ldots,m\} \tag{27}$$

$$Z_{i,j} \in \{1,0\} \qquad \forall i \in \{1,2,\ldots,m\} \, and \, \forall j \in \{1,2,\ldots,m\} \tag{28}$$

Constraints 8 to 12 (Equations 8-12) serve the same purpose as constraints 1 to 5 (Equations 1-5). Constraint 13 (Equation 13) makes sure that $Z_{i,j}$ is equal to 1 if machine i and j are adjacent (j is placed immediately to the right of i). Constraints 14 and 15 (Equations 14 and 15) ensures that each machine i can be adjacent (j is placed next to the right of i) to one machine j. Constraint 16 (Equation 16) initializes coordinate of 1st position. Constraint 17 (Equation 17) computes the coordinates of subsequent locations according to machine widths, must and extra clearances. Constraints 18-19 (Equations 18 and 19) enforce to apply must clearances between adjacent positions. Similarly, constraints 20 and 21 (Equations 20 and 21) enforces to apply extra clearances between adjacent positions. Constraint 22 (Equation 22) limits the extra clearance as $\max\left\{e_i^r, \, e_i^l\right\}$ between adjacent positions if they are able to share extra clearances. Constraints 23 and 24 (Equations 23 and 24) match the coordinates of machines and locations. That is, if i^{th} machine is placed in a^{th} position, then the coordinates of i^{th} machine and a^{th} position should be the same. Constraints 25 and 26 (Equations 25 and 26) compute the distance between machines i and j according to their positions. Constraint 27 and 28 (Equations 27 and 28) assigns $X_{i,a}$ and $Z_{i,j}$ as binary variables.

3. Proposed Genetic Algorithm

Genetic Algorithm imitates Darwinian evolutionary mechanisms based on the principle of the survival of fittest principle [36]. A genetic algorithm is developed in this research to design the restricted single row layout with zoning constraints and mutual clearances. Below the structure of the algorithm is explained.

Permutation representation is used as the individual structure, which means the solution will be a permutation of the machines to be arranged. As can be seen in Figure 3, each cell represents a gene, and all genes form a chromosome. There exist m genes in one chromosome.

The individuals are initialized by random permutation and the population is created. To avoid infeasible individuals (the ones that are violating the constraints), penalty functions are used in objective function. Roulette wheel selection operator [36] is used in this study. A proportion of the wheel is assigned to each of the individuals based on their fitness value. The better fitted individual has the larger probability of survival and mating. A one-point crossover operator, combined with a rule to avoid infeasible solutions, is used to generate children by mating the parents selected from the population through roulette wheel selection operator with a predefined crossover ratio. The individuals are then mutated through swapping genes with a low mutation ratio. Afterwards, a simplified elite preservation operation is applied to make sure that individuals with high fitness are survived over to the next generation, and the ones with low fitness are eliminated. Original population, crossed over individuals, and mutants are combined, sorted in terms of their fitness values, and the best individuals (as much as the starting population number) are kept as the population for the next generation. The process of evolution is continued till a specified maximum number of generations are realized.

The pseudo code of the proposed algorithm is as follows:

Genetic Algorithm

Input: Problem instance, population size (N), maximum iteration, crossover ratio, mutation ratio
Output: Single row machine layout, material flow and machine installation cost

1. Generation of initial population (N individuals)
2. Evaluation of individuals in terms of fitness
3. Best solution $\leftarrow$ The best fitted individual
4. For 1 : maximum iteration
5.   Roulette wheel probabilities $\leftarrow$ Based on fitness of individual
6.   Selection of parents to be mated $\leftarrow$ Roulette wheel selection operator by crossover ratio
7.   Perform the ruled one-point crossover
8.   Perform swapping genes of mutation
9.   Apply elite preserving mechanism
10.   Merging population, crossed over individuals and mutants
11.   Evaluation of individuals in terms of fitness
12.   Population for the next generation $\leftarrow$ Best N fitted individual
13.   Best solution $\leftarrow$ The best fitted individual
14. End
15. Output the solution $\leftarrow$ Best solution

The parameters of genetic algorithm are determined as the following for computational analysis: Size of population is 100, crossover ratio is 0.7 and mutation ratio is 0.01.

4. Computational Results

A computational analysis is carried out in this section to check the efficiency of the proposed approximation algorithm by comparing it to two mathematical formulations, namely a nonlinear model and a linear model. Nonlinear and linear mixed integer models are formulated in GAMS 2.25 and solved with DICOPT and CPLEX solvers, respectively, using a 2 GB memory computer and an Intel Pentium processor. Proposed genetic algorithm is built in and complied with MATLAB R2015b using the same computer.

Machine number m determines the size of the problem. 5, 10, 15, 20, 25 and 30 machines are used to test proposed algorithm in problem instances. With the increasing machine number, huge CPU time is required for the solution of the problem by exact approaches. That’s why approximation algorithm is proposed to present good quality solutions in small CPU times. In Table 1, other parameters are presented where U[a, b] means a uniform distribution in interval [a, b]. The flow of materials is generated according to the number of product types, number of products for each type, and rate of machines visited by each product type using a MATLAB code designed. f_{ij} is calculated as the sum of products whose process routes comprise machine i which immediately precedes machine j. 10 problem instances are created randomly using parameters in Table 1 for every problem size.

The sets for positive and negative zoning constraints are also created randomly. For problem instances with m=5 and 10, each of PZC (set of machines (i,j) where i and j have to be adjacent) and NZC (set of machines (i,j) where i and j cannot be adjacent) includes 1 pair of machines. For problem instances with m=15 and 20, each of PZC and NZC sets have 2 pairs of machines. For remaining instances, both include 2 pairs of machines.

Each run is repeated for 10 times with seed 1 to 10 and best results are reported.

4.1. Analysis of Instances in term of Objective Function Values and CPU Times

Table 2 shows objective function values of mathematical models and genetic algorithm for each problem size and instances. Nonlinear model fails to obtain any result for the first instance of m=20 and cannot get any results for larger instances because of resource limit of DICOPT. Similarly, linear model can give an integer result for just 1^{st}, 6^{th}, 9^{th} instances for m=30 and cannot for the remaining instances. Although linear mathematical model is better than nonlinear model in terms of presenting an integer result, nonlinear model can generate lower cost solutions than linear one. Genetic algorithm can give a feasible result for all instances and the solution quality is much better than others, in other words presented costs of proposed genetic algorithm are much lower.

CPU values of mathematical models and genetic algorithm for each problem size and instances are visualized in Table 3. As can be seen, linear model can only give result in a reasonable time for just m=5 instances. For the remaining instances it did not stop before the default 1000 seconds time limit of CPLEX solver exceeded and printed the best integer results found which are poor. Although CPU values of nonlinear model are acceptable, it cannot get any result for m=25 and larger instances because of resource limits of DICOPT and the results are worse than approximation algorithm because of getting stuck in local optima. CPU values of genetic algorithm are reasonable, and it can find better results than both mathematical models. The slight increase of CPU time with the increasing size of instances is also acceptable. A detailed deviation analysis is presented in the next section.

Table 4 visualizes min, mean and max CPU times for problem sizes. Linear model stops by the default 1000 seconds time limit of CPLEX solver for the instances m=10 and larger instances. As can be seen, mean CPU times of nonlinear model and proposed genetic algorithm increase by the increasing problem size. Although CPU times of nonlinear model is smaller than genetic algorithm for smaller instances, genetic algorithm succeeds to find results in smaller time than nonlinear model for larger instances. Moreover, nonlinear model cannot present any results for the instances m=25 and larger because of resource limit and presented cost value of nonlinear model is larger than genetic algorithm. The slight increase of CPU time with the increasing size of instances is also acceptable for proposed genetic algorithm.

4.2. Percent Deviations for the Instances

Table 5 shows deviations of approximation algorithm from mathematical models for all problem sizes and instances. Zero deviation means that genetic algorithm finds the same result with the mathematical models. A negative deviation means that genetic algorithm can find a better result. There exist no instances that mathematical models find better solutions than proposed genetic algorithm.

For the instances with 5 machines, genetic algorithm finds the same solution (optimal solution) with the mathematical models except the first and fifth instances, which nonlinear mathematical model resulted in a worse result. For the instances with 10 machines and larger, genetic algorithm generates better results than both mathematical models. For m=25 and larger instances, nonlinear algorithm cannot produce any result, therefore the deviations are not available. Although linear model ends up with an integer solution at the instances with 25 machines, they are very poor such that genetic algorithm improves the result of linear model by 16.5-33.26%. For 1st, 6th, 9th instances of m=30, genetic algorithm finds integer results which are 40.88%, 38.98% and 53.63% better than linear mathematical model, respectively. Linear model fails to produce any integer result for the other instances. Table 6 visualizes min, mean and max deviations for problem sizes. As can be seen, mean deviations increase by the increasing problem size, and they are considerably big which means genetic algorithm enhances mathematical models to a large extent.

Table 7 presents overall deviations of approximation algorithm from mathematical models. In average, proposed genetic algorithm improves linear mathematical model by 18.15% and nonlinear mathematical model by 2.07%. Moreover, the proposed genetic algorithm never gives worse results than the mathematical models.

4.3. Cost Reduction by Considering Mutual Clearances and Machine Installation Costs

The proposed model in this paper improves the cost of machine layout design both in terms of material flow and machine installation costs. The distances between machines are reduced by mutual clearances between adjacent machines to decrease material flow costs. Moreover, installation costs of machines are considered in the objective function because they are huge expenses which may affect the decisions of machine locations.

In this section, a comparison is carried out to observe the cost improvement of the proposed model in this research. A nonlinear mathematical model, which is not including mutual clearances and machine installation costs in the objective function, is formulated, and presented below:

Model 3: Nonlinear Mathematical Model (without mutual clearances and installation costs)

$$Min \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{a=1}^{m-1}\sum_{b=a+1}^{m}(f_{i,j}+f_{j,i})d_{a,b}X_{i,a}X_{j,b} \quad (29)$$

st

$$Z_{real}=\sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{a=1}^{m-1}\sum_{b=a+1}^{m}(f_{i,j}+f_{j,i})d_{a,b}X_{i,a}X_{j,b}+\sum_{i=1}^{m}\sum_{a=1}^{m}IC_{i,a}X_{i,a} \quad (30)$$

$$d_{a,b}=\sum_{i=1}^{m}\sum_{t=a+1}^{b-1}w_i X_{i,t}+\sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{t=a}^{b-1}(c_{i,j}+e_i^r+e_j^l)X_{i,t}X_{j,t+1}+\sum_{i=1}^{m}\frac{w_i(X_{i,a}+X_{i,b})}{2} \quad (31)$$

$$\forall a,b: 1\le a<b\le m$$

$$Constraints\,(2)-(5)\,,(7) \quad (32)$$

The objective function (29) minimizes the cost which includes only material flow costs. The real cost (Z_{real}), including material flow and machine installation costs, is calculated in Constraint (30). Constraint (31) calculates the distances between machines according to the decided locations without mutual clearances. Remaining constraints are given previously which are (2)-(5) and (7).

The runs are performed with the instances presented previously using DICOPT solver of GAMS. The results (Z_{real}) are presented in Table 8 and compared with the cost values of the proposed nonlinear mathematical model in Table 9. Since nonlinear mathematical model cannot produce any integer solutions for instances with 25 and 30 machines, they are not presented in the table.

As can be seen in Table 9, the costs are improved for all instances and are reduced by 3.45%-20.10%. Mean cost reductions are also presented in the table for each problem size.

Total cost of machine layout problem (material flow and machine installation costs) is reduced by 11.36% in average by the proposed model.

4.4. Analysis of Genetic Algorithm Parameters

An analysis of proposed genetic algorithm parameters is carried out to test the solution quality for instance 5 with 15 machine numbers. The analysis is done by changing three parameters one by one, namely, size of population, crossover, and mutation ratio. Size of population is selected to be 50, 100, 150 and 200. Crossover and mutation ratio levels are “0.5, 0.7, 0.9” and “0.1, 0.01, 0.001” respectively. Each run is realized with seed 1 to 10 and best results are reported. Iteration number is selected to be 1000 for all instances. The results are presented in Table 10.

Objective function values (material flow and machine installation costs) changes between 174378 and 189760. Minimum cost appears at two points, which are “size of population=200, crossover ratio=0.9, mutation ratio=0.01” and “size of population=100, crossover ratio=0.9, mutation ratio=0.1”. Maximum cost appears at “size of population=50” as expected with “crossover ratio=0.5 and mutation ratio=0.001”. Even the worst result of proposed genetic algorithm is better than the result of linear mathematical model, which is 192618, for the same instance.

Generally, the results are improved with increasing population number, crossover, and mutation rate. CPU time slightly increases with the increasing mutation rate; however, it rises proportionally with crossover rate and population size. Therefore, the parameters should be selected by considering both solution quality and CPU time.

5. Conclusions

In this paper, a restricted single row machine layout problem with mutual clearances is considered under an enhanced objective function of minimizing material flow and machine installation costs in flexible manufacturing systems. Positive and negative zoning constraints restrict the problem on the placement of machines which is common in real life problems. Moreover, the distances between machines are shortened and hence material flow costs are reduced by mutual use of clearances between adjacent machines. Nonlinear and linear mathematical programming formulations are offered for optimal solutions; however, the mathematical models are failed to give optimal solutions greater than five machine problem instances because of the complexity of the problem and time limit. Therefore, a genetic algorithm is proposed generating high quality solutions in reasonable time. Several problem instances are created by changing problem sizes and the efficiency of the proposed genetic algorithm is tested in terms of solution quality and time. The performances of mathematical models are also compared. Nonlinear mathematical model performs better than linear mathematical model both in terms of objective function value and solution time for small size instances. However, nonlinear model cannot present any result for larger instances because of resource limit in DICOPT while linear model presents an integer solution for these instances which are poor compared to proposed genetic algorithm. It is demonstrated that genetic algorithm performs better than both nonlinear and linear mathematical model and presented machine arrangements with smaller cost in smaller time. A sensitivity analysis is also performed to test the effect of genetic algorithm parameters on results. Moreover, the proposed model is compared with the one which ignores mutual clearances and installation costs in the objective function. Total cost of machine layout problem is reduced by 11.36% by the proposed model. As future studies, double or multirow machine layouts can be considered and/or additional constraints can be included to the problem.

References

1. Singh, N. and Rajamani, D. “Cellular manufacturing systems, design, planning and control”, London: Chapman & Hall (1996).

2. Braglia, M. “Optimisation of a simulated-annealing-based heuristic for single row machine layout problem by genetic algorithm”, *International Transactions in Operational Research*, 3(1), pp. 37-49 (1996).

3. Braglia, M. “Heuristics for single-row layout problems in flexible manufacturing systems”, *Production Planning & Control: The Management of Operations*, 8(6), pp. 558-567 (1997).

4. Ramkumar, A.S. and Ponnambalam, S.G. “Design of single-row layouts for Flexible Manufacturing Systems using genetic algorithm and simulated annealing Algorithm”, *Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems,* Singapore (2004).

5. Ficko, M., Brezocnik, M., and Balic J. “Designing the layout of single- and multiple-rows flexible manufacturing system by genetic algorithms”, *Journal of Materials Processing Technology,* 157–158, pp. 150-158 (2004).

6. Solimanpur, M., Vrat P., and Shankar R. “An ant algorithm for the single row layout problem in flexible manufacturing systems”, *Computers & Operations Research*, 32, pp. 583–598 (2005).

7. Ozcelik, F. “A hybrid genetic algorithm for the single row layout problem”, *International Journal of Production Research*, 50(20), pp. 5872-5886 (2012).

8. Jahromi, M.H.M.A., Tavakkoli-Moghaddam, R., Makui A., et al. “A novel mathematical model for a scheduling problem of dynamic machine-tool selection and operation allocation in a flexible manufacturing system: A modified evolutionary algorithm”, *Scientia Iranica E*, 24(2), pp. 765-777 (2017).

9. Tubaileh, A. and Siam, J. “Single and multi-row layout design for flexible manufacturing systems”, *International Journal of Computer Integrated Manufacturing,* 30(12), pp. 1316-1330 (2017).

10. Rai, R.S. and Jayswal, S.C. “Design and Optimization of Loop Layout in Flexible Manufacturing System using Particle Swarm Optimization”, *Int J Adv Technol*, 9(2) (2018).

11. Aghajani-Delavar, N., Mehdizadeh, E., Tavakkoli-Moghaddam, R., et al. “A multi-objective vibration damping optimization algorithm for solving a cellular manufacturing system with manpower and tool allocation”, *Scientia Iranica E* (2020).

12. Yadav, A. and Jayswal, S.C. “Evaluation of batching and layout on the performance of flexible manufacturing system”, *Int J Adv Manuf Technol*, 101, pp. 1435–1449 (2019).

13. Hafiz, Z.N., Tauseef, A., Fahid, R. “Modeling, analysis and optimization of carousel-based flexible manufacturing system”, *Journal of Industrial and Production Engineering*, pp. 1-15 (2022).

14. Ponnambalam, S.G. and Ramkumar, V. A “Genetic Algorithm for the Design of a Single-Row Layout in Automated Manufacturing Systems”, *The International Journal of Advanced Manufacturing Technology*, 18, pp. 512–519 (2001).

15. Lin M.T. “The single-row machine layout problem in apparel manufacturing by hierarchical order-based genetic algorithm”, *International Journal of Clothing Science and Technology*, 20(5), pp. 258-270 (2008).

16. Datta, D., Amaral, A.R.S., and Figueira, R. “Single row facility layout problem using a permutation-based genetic algorithm”, *European Journal of Operational Research*, 213, pp. 388–394 (2011).

17. Lenin, N., Kumar, M.S., Islam, M.N., et al. “Multi-objective optimization in single-row layout design using a genetic algorithm”, *The International Journal of Advanced Manufacturing Technology* , 67, pp. 1777–1790 (2013).

18. Kothari, R. and Ghosh, D. “An efficient genetic algorithm for single row facility layout”, *Optim Lett*, 8, pp. 679–690 (2014).

19. Ghadirpour, M., Rahmani, D., Moslemipour, G. "Routing flexibility for unequal–area stochastic dynamic facility layout problem in flexible manufacturing systems", *IJIEPR*, 31(2), pp. 269-285 (2020).
20. Kouvelis, P., Chiang, W.C., and Fitzsimmons J. "Simulated annealing for machine layout problems in the presence of zoning constraints", *European Journal of Operational Research*, 57, pp. 203-223 (1992).
21. Wang, T.Y., Lin, H.C., and Wu, K.B. "An improved simulated annealing for facility layout problems in cellular manufacturing systems", *Computers Ind. Engng*, 34(2), pp. 309-319 (1998).
22. Brunese, P.A. and Tanchoco, J.M.A. "On implied within-building constraints for machine layout", *International Journal of Production Research*, 51(6), pp. 1937–1952 (2013).
23. Kalita, Z. and Datta, D. "A constrained single row facility layout problem", *The International Journal of Advanced Manufacturing Technology*, 98, pp. 2173–2184 (2018).
24. Li, S., Zhang, Z., Guan, C., at al. "An improved fireworks algorithm for the constrained single-row facility layout problem", *International Journal of Production Research*, 59(8), pp. 2309-2327 (2021).
25. Yang, C.O. and Utamima, A. "hybrid estimation of distribution algorithm for solving single row facility layout problem", *Computers & Industrial Engineering*, 66, pp. 95–103 (2013).
26. Maier, K and Taferner, V. "Solving the constrained Single-Row Facility Layout Problem with Integer Linear Programming", *International Journal of Production Research*, pp. 1-16 (2022).
27. Kalita, Z. and Datta, D. "The Constrained Single-Row Facility Layout Problem with Repairing Mechanisms", In *Bennis, F., Bhattacharjya, R. (eds) Nature-Inspired Methods for Metaheuristics Optimization*. Modeling and Optimization in Science and Technologies, Springer International Publishing, Cham, 16, pp. 359-383 (2020).
28. Coppé, V., Gillard, X., and Schaus, P. "Solving the Constrained Single-Row Facility Layout Problem with Decision Diagrams", *28th International Conference on Principles and Practice of Constraint Programming*, 235, pp. 14-18 (2022).
29. Manzke, L., Keller, B., and Buscher, U. "An Artificial Bee Colony Algorithm to Solve the Single Row Layout Problem with Clearances", *International Conference on Information Systems Architecture and Technology – ISAT 2017*, 657 (2018).
30. Safarzadeh, S. and Koosha, H. "Solving an extended multi-row facility layout problem with fuzzy clearances using GA", *Applied Soft Computing*, 61, pp. 819-831 (2017).
31. Keller, B. "Construction heuristics for the single row layout problem with machine-spanning clearances", *INFOR: Information Systems and Operational Research*, 57(1), pp. 32-55 (2019).
32. Yu, M., Zuo, X., and Murray, C.C. "A tabu search heuristic for the single row layout problem with shared clearances", *IEEE Congress on Evolutionary Computation (CEC)*, Beijing, China (2014).
33. Zuo, X.Q., Murray, C.C., and Smith, A.E. "Sharing clearances to improve machine layout", *International Journal of Production Research*, 54(14), pp. 4272-4285 (2016).
34. Akbilek, N. "Safety-integrated single-row machine layout problem optimization using GA and TS with a novel heuristic rule support", *Soft Comput*, 25, pp. 13533–13547 (2021).
35. Sule R.D. Manufacturing facilities location, planning, and design (3rd ed.). CRC Press, Taylor & Francis Group 2008.
36. Goldberg, D.E. "Genetic Algorithms in Search, Optimization, and Machine Learning", *Addison-Wesley,* New Jersey, USA (1989).

Appendices

Figures and Tables

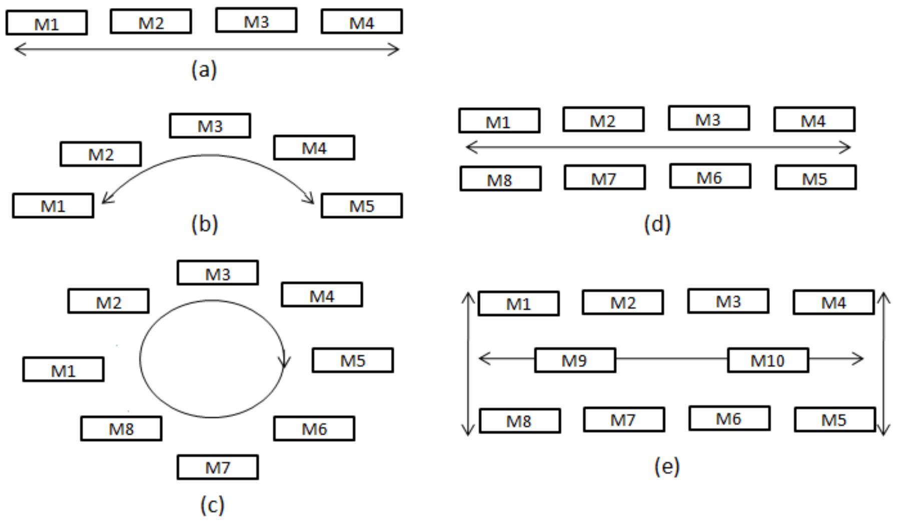

Figure 1. Types of layouts in flexible manufacturing systems using material handling devices

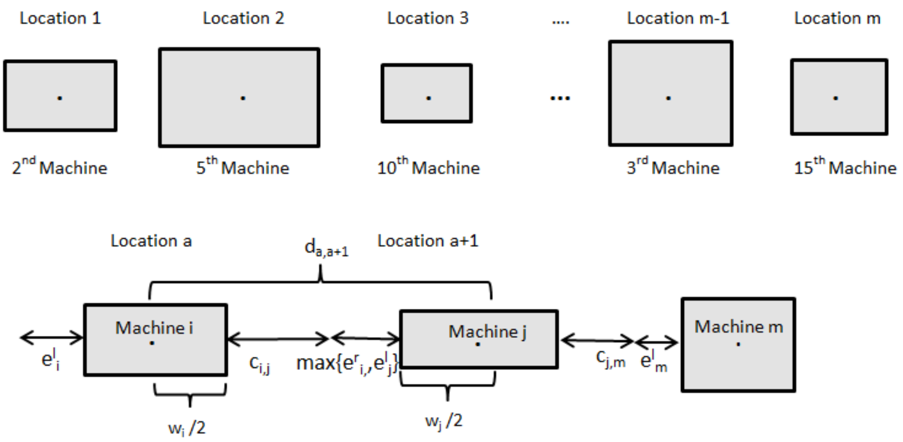

Figure 2. Location of machines with respect to dimensions and clearances

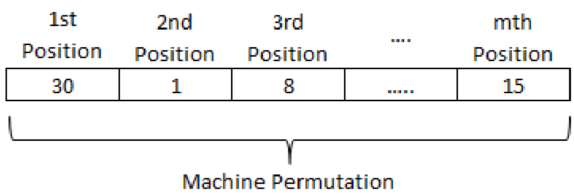

Figure 3. Individual (chromosome) structure

Table 1. Parameters used in problem instances

Definition	Factor	Level	
Machine width	w_i	U[1,3]	
Must clearance	c_{ij}	U[0.5,1.5]	
Extra clearances	e^r_{i}, e^l_i	w_i x U[0.3,0.4]	
Installation cost	$IC_{i,a}$	U[0,500]	
Flow of materials	f_{ij}	Number of product types	U[10,30]
		Number of products for each type	U[30,70]
		Rate of machines visited by each product type	U[0.4,0.8]

Table 2. Objective function values (costs) of mathematical models and genetic algorithm

m	Linear Model Instances									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
5	14663	17812	11288	7637	12424	11485	10719	10995	21173	17655
10	47119	60465	33704	52116	78945	56506	76118	53225	25978	71801
15	104404	115895	89618	80141	192618	163420	118430	146479	185726	158238
20	153688	169404	205996	322892	150436	157224	241434	229833	263231	343381
25	467817	239704	311251	486282	355985	249011	362506	301799	289465	285195

30	362891	-	-	-	-	571240	-	-	232471	-
m	Nonlinear Model Instances									
	1	2	3	4	5	6	7	8	9	10
5	14672	17812	11288	7637	12536	11485	10719	10995	21173	17655
10	48247	61157	33351	50988	74523	55909	75813	52656	26452	68060
15	89059	102858	76992	70815	179940	153570	103774	136033	180450	142710
20	-	136418	185484	270582	128907	123523	201991	199507	209190	311878
25	-	-	-	-	-	-	-	-	-	-
30	-	-	-	-	-	-	-	-	-	-
m	Genetic Algorithm Instances									
	1	2	3	4	5	6	7	8	9	10
5	14663	17812	11288	7637	12424	11485	10719	10995	21173	17655
10	47119	58978	32567	50537	74354	54416	74617	52147	25847	67652
15	88500	97787	76005	69104	177446	141548	100358	135155	172730	137814
20	133981	135246	178083	269468	124277	120197	188145	192585	205059	297306
25	384111	179881	239267	417409	270362	201553	302890	240314	233570	226501
30	257586	423615	484409	432464	238953	411017	625421	713026	151322	682623

Table 3. CPU of mathematical models and genetic algorithm

M	Linear Model Instances									
	1	2	3	4	5	6	7	8	9	10
5	0.20	0.34	0.37	0.19	0.31	0.23	0.19	0.17	0.20	0.22
10	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
15	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
20	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
25	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
30	1000	-	-	-	-	1000	-	-	1000	-
M	Nonlinear Model Instances									
	1	2	3	4	5	6	7	8	9	10
5	0.87	0.83	1.17	0.97	1.26	0.41	0.76	0.84	0.70	0.34
10	3.65	4.91	3.16	2.90	6.14	5.40	2.18	4.36	3.03	1.88
15	29.41	17.60	14.44	22.22	27.39	17.44	9.89	10.77	9.05	9.51
20	-	53.11	89.86	55.69	61.24	99.56	92.51	44.73	102.21	175.02
25	-	-	-	-	-	-	-	-	-	-
30	-	-	-	-	-	-	-	-	-	-
m	Genetic Algorithm Instances									
	1	2	3	4	5	6	7	8	9	10
5	20.11	20.29	19.92	20.23	19.89	21.19	21.61	21.37	21.40	21.60
10	25.11	28.30	26.40	27.61	26.88	26.73	26.68	27.71	28.95	29.31
15	38.39	39.36	42.21	30.49	35.20	29.63	26.62	32.26	32.88	36.21
20	58.05	55.23	53.39	58.04	59.35	54.48	57.44	52.47	63.59	56.00
25	81.74	83.27	86.99	89.07	89.28	85.46	88.73	84.25	80.78	82.34
30	130.58	122.59	128.15	130.27	130.77	127.44	125.63	127.28	129.98	130.64

Table 4. Problem Size Based CPU of mathematical models and genetic algorithm

m	Linear Model			Nonlinear Model			Genetic Algorithm		
	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max

5	0.17	0.24	0.37	0.34	0.81	1.26	19.89	20.76	21.61
10	1000	1000	1000	1.88	3.76	6.14	25.11	27.37	29.31
15	1000	1000	1000	9.05	16.77	29.41	26.62	34.32	42.21
20	1000	1000	1000	44.73	85.99	175.02	52.47	56.80	63.59
25	1000	1000	1000	-	-	-	80.78	85.19	89.28
30	1000	1000	1000	-	-	-	122.59	128.33	130.77

Table 5. Deviations of Approximation Algorithm from Mathematical Models for All Instances

M	% Deviation from Linear Model Instances									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
5	0.00	0.00	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.0
10	0.00	-2.52	-3.49	-3.12	-6.2	-3.84	-2.01	-2.07	-0.50	-6.1
15	-17.97	-18.52	-17.91	-15.97	-8.6	-15.45	-18.01	-8.38	-7.52	-14.8
20	-14.71	-25.26	-15.67	-19.83	-21.0	-30.81	-28.32	-19.34	-28.37	-15.5
25	-21.79	-33.26	-30.09	-16.50	-31.7	-23.55	-19.68	-25.59	-23.93	-25.9
30	-40.88	-	-	-	-	-38.98	-	-	-53.63	-
M	**% Deviation from Nonlinear Model Instances**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
5	-0.06	0.00	0.00	0.00	-0.90	0.00	0.00	0.00	0.00	0.00
10	-2.39	-3.69	-2.41	-0.89	-0.23	-2.74	-1.60	-0.98	-2.34	-0.60
15	-0.63	-5.19	-1.30	-2.48	-1.41	-8.49	-3.40	-0.65	-4.47	-3.55
20	-	-0.87	-4.16	-0.41	-3.73	-2.77	-7.36	-3.59	-2.01	-4.90
25	-	-	-	-	-	-	-	-	-	-
30	-	-	-	-	-	-	-	-	-	-

Table 6. Problem Size Based Deviations of Approximation Algorithm from Mathematical Models

M	% Deviation from Linear Model			% Deviation from Nonlinear Model		
	Min	**Mean**	**Max**	**Min**	**Mean**	**Max**
5	0.00	0.00	0.00	-0.90	-0.10	0.00
10	-6.17	-2.99	0.00	-3.69	-1.79	-0.23
15	-18.52	-14.31	-7.52	-8.49	-3.16	-0.63
20	-30.81	-21.89	-14.71	-7.36	-3.31	-0.41
25	-33.26	-25.20	-16.50	-	-	-
30	-53.63	-44.50	-38.98	-	-	-

Table 7. Overall Deviations of Approximation Algorithm from Mathematical Models

% Deviation from Linear Model			% Deviation from Nonlinear Model		
Min	**Mean**	**Max**	**Min**	**Mean**	**Max**
-53.63	-18.15	0.00	-8.49	-2.09	0.00

Table 8. Z_{real} Without Considering Mutual Clearances and Installation Costs

m	Total Costs of Instances									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
5	15606	19265	13554	8181	12968	13141	12804	13205	24636	19906

10	53235	67260	35342	57979	88747	66970	87168	61217	30422	78850
15	97261	109011	85621	81613	194121	159594	117347	151267	193433	154149
20	-	154651	196793	295395	136447	141995	216991	218456	232628	328754

Table 9. Cost Reduction by Considering Mutual Clearances and Machine Installation Costs

m	**% Cost Reduction for Instances**										
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Mean**
5	-6.37	-8.16	-20.07	-7.13	-3.45	-14.42	-19.45	-20.10	-16.36	-12.75	-12.83
10	-10.34	-9.98	-5.97	-13.71	-19.09	-19.78	-14.98	-16.26	-15.01	-15.85	-14.10
15	-9.21	-5.98	-11.21	-15.25	-7.88	-3.92	-13.08	-11.20	-7.19	-8.02	-9.29
20	-	-13.37	-6.10	-9.17	-5.85	-14.95	-7.43	-9.50	-11.20	-5.41	-9.22
25	-	-	-	-	-	-	-	-	-	-	-
30	-	-	-	-	-	-	-	-	-	-	-

Table 10. Analysis of proposed genetic algorithm parameters

Size of Population	**Crossover Ratio**	**Mutation Ratio**	**Cost Value**	**CPU**	**Size of Population**	**Crossover Ratio**	**Mutation Ratio**	**Cost Value**	**CPU**
200	0.9	0.1	175643	95	100	0.9	0.1	174378	48
		0.01	174378	89			0.01	178554	44
		0.001	180111	88			0.001	181069	44
	0.7	0.1	176734	76		0.7	0.1	176326	38
		0.01	177051	70			0.01	177446	35
		0.001	178439	69			0.001	180819	34
	0.5	0.1	177274	56		0.5	0.1	175643	28
		0.01	177147	50			0.01	178509	25
		0.001	179303	50			0.001	182674	25
150	0.9	0.1	176326	78	50	0.9	0.1	178824	25
		0.01	176509	67			0.01	178823	23
		0.001	180140	66			0.001	188150	23
	0.7	0.1	175779	57		0.7	0.1	178824	20
		0.01	178509	53			0.01	178345	19
		0.001	178601	52			0.001	186726	18
	0.5	0.1	177078	43		0.5	0.1	177790	15
		0.01	176265	38			0.01	178509	14
		0.001	177327	38			0.001	189760	13

Biographies

Zeynep Uruk is a PhD student at Industrial Engineering, Sakarya University since September 2019. She has received her M.Sc. in Industrial Engineering from Bilkent University, Turkey, in 2011 and B.Sc. in Chemical Engineering from the Middle East Technical University, Turkey, in 2008.

Nevra Akbilek is an assistant professor in Sakarya University, Turkey. She received her B.E., M.E., and Ph.D. degrees from Sakarya University, Sakarya, Turkey, in 2000, 2003, and 2009, respectively. She was a research assistant from 2001 to 2009 in Sakarya University engineering faculty, Industrial engineering department. Then she received a post-doctorate fellowship at Auburn University in the USA in 2011 and finished her post-doctorate in 2012.