

A hybrid CS-SA intelligent approach to solve uncertain dynamic facility layout problems considering dependency of demands

Ghorbanali Moslemipour¹

Received: 7 January 2017 / Accepted: 11 July 2017 / Published online: 18 July 2017
© The Author(s) 2017. This article is an open access publication

Abstract This paper aims at proposing a quadratic assignment-based mathematical model to deal with the stochastic dynamic facility layout problem. In this problem, product demands are assumed to be dependent normally distributed random variables with known probability density function and covariance that change from period to period at random. To solve the proposed model, a novel hybrid intelligent algorithm is proposed by combining the simulated annealing and clonal selection algorithms. The proposed model and the hybrid algorithm are verified and validated using design of experiment and benchmark methods. The results show that the hybrid algorithm has an outstanding performance from both solution quality and computational time points of view. Besides, the proposed model can be used in both of the stochastic and deterministic situations.

Keywords Clonal selection · Simulated annealing · Stochastic dynamic · Facility layout problem

Introduction

Nowadays, manufacturing industries make inexorable attempt to achieve the competitive benefit of the design of an optimal layout of facilities. Due to the fact that the facility layout problem (FLP) significantly affects the high manufacturing cost involved in intelligent manufacturing systems, it can be viewed as a crucial issue in the design of modern production systems. Facility layout is the problem of

determining the relative locations of facilities on the shop floor. The optimal arrangement of these facilities leads to minimising the total manufacturing cost and maximising the productivity. The Material Handling Cost (MHC) forms 20–50% of the total manufacturing costs and it can be reduced by at least 10–30% by designing an optimal layout (Tompkins et al. 2003). According to the nature of product demands and time planning horizon, the FLP can be classified into four problems as follows: (1) static facility layout problem (SFLP) with deterministic and constant flow of materials over a single time period, (2) dynamic (multi-period) facility layout problem (DFLP) with different deterministic flow of materials in each period, (3) stochastic static facility layout problem (SSFLP) with stochastic flow of materials over a single time period, and (4) stochastic dynamic facility layout problem (SDFLP) where product demands are random variables so that their parameters change from period to period. In the SFLP and SSFLP an optimal layout of facilities is designed so that the total MHC is minimised. On the other hand, for each period of the DFLP and SDFLP an optimal facility layout is designed so that the total material handling and rearrangement costs is minimised. The SDFLP is the most complicated form of the FLP so that the other forms can be regarded as the special case of the problem. In general, the FLP with discrete representation and equal-sized facilities, which are assigned to the same number of locations, can be formulated as the Quadratic Assignment Problem (QAP). In discrete representation, the shop floor is divided into a number of equal-sized locations. Koopmans and Beckman (1957) proposed the first QAP model for the FLP. In this paper, the SDFLP is formulated using a QAP-based mathematical model. Designing the optimal layout of facilities is a crucial competitive advantage of production industries. To find the optimal solution of the FLP formulated by a QAP-based mathematical model, the

✉ Ghorbanali Moslemipour
ghmoslemipour@pnu.ac.ir; ghmoslemipour@yahoo.com

¹ Department of Industrial Engineering, Payame Noor University (PNU), P.O.BOX, 19395-3697 Tehran, Iran

resolution approaches such as exact methods, heuristic algorithms, and intelligent approaches can be used. Since the QAP is a Non-deterministic Polynomial (NP)-complete problem (Sahni and Gonzalez 1976; Vinoba and Indhumathi 2015), it is very difficult to solve the problem using exact and heuristic methods, especially in large sizes. Thus, it is essential to use intelligent approaches as promising tools for solving the SDFLP formulated by the QAP in a reasonable computational time. Regarding the use of intelligent approaches for solving the DFLP, the following previous researches can be mentioned: Pourvaziri and Naderi (2014) developed a hybrid multi-population Genetic Algorithm (GA) to cope with the DFLP. Azadeh et al. (2014) solved a DFLP having equal-sized facilities using Data Envelopment Analysis (DEA) and diversification strategy of Tabu Search (TS) algorithm. Fazlelahi et al. (2015) suggested a model to design a robust facility layout in dynamic environment utilizing a permutation-based GA. Hasani et al. (2015) proposed a hybrid intelligent approach for solving the DFLP. Derakhshan Asl and Kuan (2015) dealt with static and dynamic FLPs having unequal-sized facilities using a modified Particle Swarm Optimisation (PSO) approach. Bashiri and Karimi (2012) solved a QAP model using some heuristic and meta-heuristic algorithms including SA, TS and PSO approaches. Intelligent approaches are able to find a sub-optimal (near to optimal) solution for the problem at hand. Therefore, there is an urgent need for improving the quality of the obtained solution. Hybrid resolution approaches are common methods to enhance the solution quality of the optimisation problems. Simulated Annealing (SA) is a promising method to solve the FLP especially in volatile environments because it has some advantages such as low computational time, free of local optima, easy for implementation, and convergent property (Moslemipour et al. 2012). Kulturel-Konak and Konak (2015) proposed a simulated annealing meta-heuristic approach to solve a cyclic FLP as a special case of DFLP where product mix and volume are changed seasonally. Experimental studies on SA demonstrate that selecting a good initial solution for this algorithm leads to improvement in both the obtained solution quality and execution time. Abtahi and Bijari (2017) proposed a hybrid intelligent approach using imperialistic competition, harmony search and SA algorithms.

The first novelty of this paper is proposing a new QAP-based mathematical model for designing an optimal facility layout in each period of the multi-period time planning horizon of the SDFLP. In this model, product demands are assumed to be dependent normally distributed random variables with known Probability Density Function (PDF) and covariance that changes from period to period at random. Product demands have also been considered as normally distributed or at least random variables in layout design problems in a number of previous studies (Forghani et al.

2013; Nematian 2014; Vafaeinezhad et al. 2016; Vitayasak et al. 2016; Zhao and Wallace 2014, 2015). The second novelty of this paper is suggesting a new hybrid algorithm named CS-SA for solving the SDFLP by combining the Clonal Selection (CS), and SA algorithms. In this algorithm, a population of randomly generated initial solutions is improved using the CS algorithm. By doing so, the performance of the SA algorithm is likely to be improved from both solution quality and computational time points of view.

Literature review

Hitchings (1970) considered dynamic behaviour of the FLP for the first time. Palekar et al. (1992) designed the SDFLP using quadratic integer programming model. They considered three degrees of uncertainties named optimistic, most likely, and pessimistic for product demands by assigning probability of happenings to these degrees. Montreuil and Laforge (1992) addressed the SDFLP using a scenario tree of probable futures. In their method, a number of alternative layouts were built for future. Krishnan et al. (2008) proposed three mathematical models for designing a facility layout in an uncertain environment by considering multiple product demand scenarios. Tavakkoli-Moghaddam et al. (2007) suggested a novel formulation using the QAP to concurrent design of the optimal machine and cell layouts in a single time period planning horizon of a cellular manufacturing system (CMS) by considering the stochastic independent product demands with known normal PDF. Moslemipour and Lee (2012) proposed a new QAP-based mathematical model for designing an optimal machine layout for each period of the SDFLP, in which the product demands are independent normally distributed random variables with known PDF that changes from period to period at random. In this paper, in addition to the aforementioned assumptions, time value of money and dependency of product demands are also considered so that the expectation, variance and covariance of demands are randomly changed from period to period. Shafiq et al. (2015) proposed a novel mathematical model for designing dynamic distributed layouts.

Clonal selection algorithm

CS algorithm belongs to Artificial Immune System (AIS) approaches. The biological immune system protects the human body against foreign invaders such as viruses and bacteria called antigens. The molecules named antibodies, which recognise the presence of an antigen, are rapidly increased by cloning during the clonal selection process. The affinity of the new cloned antibodies is improved by mutations, which in turn, leads to neutralisation and

elimination of the antigen. The probability of selecting antibodies for mutation is proportional to their affinity to the antigen. After mutation, the receptor editing process is started by eliminating some percentage of the ineffective antibodies and introducing the same percentage of the new ones. The simulation of the natural immune system leads to development of a new intelligent algorithm named AIS. The AIS starts with a randomly generated population of individuals (antibodies) as the possible solutions. At each iteration of this algorithm, first, the affinity of each antibody is calculated using the objective function of the problem. Next, a number of antibodies with the best affinity value are selected and cloned. Then, each clone is mutated and the improved antibodies are preserved for the next generation. Finally, using the receptor editing process, a pre-specified number of antibodies with low affinity values are replaced with the new ones, which are generated at random. Ulutas and Islier (2009) proposed a CS algorithm for solving the DFLP. Satheesh Kumar et al. (2009) proposed an AIS algorithm to solve the unidirectional loop layout problem by minimising the total congestion of all parts and minimising the maximum congestion amongst a partfamily. Ulutas and Kulturel-Konak (2012) suggested a CS algorithm to deal with a FLP having unequal-sized facilities and flexible bay arrangement. Ulutas and Kulturel-Konak (2013) suggested an improved CS algorithm for designing the FLP with unequal-sized facilities. They proposed a novel encoding method and a guideline for optimising the parameters of the CS algorithm. Ulutas and Islier (2015) proposed a CS algorithm for solving a real DFLP named footwear industry.

In this paper, the CS algorithm is utilised to improve the quality of randomly generated initial solutions. The initial solutions are applied to SA algorithm described in the following section. Doing so, a hybrid algorithm named CS-SA is constructed so that it has an outstanding performance from both solution quality and computational time points of view with respect to the SA having just a single initial solution.

Simulated annealing algorithm

SA algorithm is a simulation of physical annealing process of solids in statistical mechanics, which starts with a known or randomly generated initial solution and a high initial value of temperature. It is formed by two loops namely, the inner loop to search for a neighbouring solution, and the outer loop for decrease the temperature to reduce the probability of accepting the non-improving neighbouring solutions in the inner loop. The quality of the solution obtained by SA depends upon various factors such as initial temperature, initial solution, and the exchange approach for generating neighbouring solutions in the inner loop. Due to the fact that SA is a single search algorithm, the quality of its final

solution is affected by the initial solution (Ashtiani et al. 2007; Chen and Jing 2005). Using hybrid algorithms, the disadvantage of some algorithms are compensated by the advantage of some other ones. Ram et al. (1996) proposed two distributed algorithm namely, Clustering Algorithm (CA) and Genetic Clustering Algorithm (GSA), which generate a good initial solution and a good population of initial solutions for SA algorithm, respectively. They applied the two algorithms to the Job Shop Scheduling (JSS) problem and the Travelling Salesman Problem (TSP) and concluded that the performance of SA was improved from both computational time and solution quality standpoints. Ashtiani et al. (2007) developed multiple initial solutions for SA approach to solve the DFLP. Dong et al. (2009) used SA algorithm to solve the DFLP by considering the possibility of removing/adding facilities in different periods. Reza zadeh et al. (2009) and Tajbakhsh et al. (2009) combined SA algorithm with PSO method to solve the FLP. Sahin et al. (2010) used SA algorithm to deal with the DFLP by regarding budget constraint. Madhusudanan-Pillai et al. (2011) solved their proposed robust layout design model using SA in DFLP. Moslemipour and Lee (2012) utilised SA algorithm to solve their developed model in the SDFLP. Lee et al. (2012) proposed a novel hybrid Ant Clony (AC)/SA approach having outstanding performance to solve the SDFLP. Moslemipour et al. (2012) reviewed intelligent approaches for solving layout problems comprehensively. Khosravian-Ghadikolaei and Shahanaghi (2013) proposed a solution approach based on SA algorithm for solving the multi-floor DFLP. Shirazi et al. (2014) developed a multi-objective SA-based algorithm to deal with a group layout. Li et al. (2015) suggested an improved SA algorithm for designing the SDFLP. Palubeckis (2015) dealt with the single row facility layout problem by proposing a fast SA algorithm having better performance than the iterated TS algorithm. Palubeckis (2016) also proposed a multi-start SA approach to design the single row facility layout. Kulturel-Konak and Konak (2015) developed a large-scale hybrid SA algorithm for solving the cyclic FLP as a special case of the DFLP, which is formulated by a Mixed Integer Programming (MIP) model. In the cyclic FLP, the production cycle replicate itself by going to the first time period after the last one owing to the seasonal feature of products. Matai (2015) developed an improved SA algorithm for solving the FLP. Tayal and Singh (2016) used a hybrid meta-heuristic approach, which is based on Firefly and Chaotic SA algorithms for solving the multi-objective SDFLP by considering stochastic demands. Tayal and Singh (2017) proposed an integrated resolution approach by combining the SA algorithm with the DEA and TOPSIS as practical decision-making methods for solving a multi-objective SDFLP. They considered some quantitative and qualitative objectives, such as total MHC, flow distance, closeness ratio and maintenance issues.

The proposed model

The QAP is a nonlinear Combinatorial Optimisation Problem (COP). This problem aims to find the optimum appointing of several items to the same number of known positions so that a quadratic objective function is minimised. In general, the FLP having discrete representation and equal-sized facilities assigned to the same number of known locations is usually formulated by the QAP model. In discrete representation, the shop floor is divided into a number of equal-sized facility locations. In this section, the SDFLP is formulated by a new QAP-based mathematical model to design the dynamic machine layout. The parameters and indexes used in the models are shown in Table 1.

The assumptions are as follows:

1. Equal-sized facilities (machines) are assigned to the same number of known locations.
2. The discrete representation of the SDFLP is considered.
3. For simplicity, it is assumed that demands of a particular part in different time periods are independent of each other (i.e. $\text{Cov}(D_{tk}, D_{t'k}) = 0$). Shumsky and Zhang (2009), for instance, has already made this assumption.
4. Demands of products are assumed to be dependent (i.e. $\text{Cov}(D_{tk}, D_{t'k'}) \neq 0$) normally distributed random variables with known expected value, variance, and covariance that change from period to period at random.
5. The parts are moved in batches between facilities.
6. Time value of money is considered.
7. There is no constraint for dimensions and shapes of the shop floor.
8. The data on number of facilities, number of periods, machine sequence, present value of part movement cost, transfer batch size, distance between facility locations, money interest rate for each period (e.g., year), present value of facility rearrangement cost, the expected value, variance, and covariance of part demands in each period are known as inputs of the model.
9. Machines can be laid out in any configuration such as rectangular and U-shaped configurations as shown in Fig. 1, where L1, ..., L12 are the known machine locations.

Designing a dynamic facility layout is a method to cope with fluctuations and uncertainties in product demands in the SDFLP. In the dynamic layout design method, an optimal layout is designed for each period so that the total cost including the MHC and the relocation cost is minimised. Using this approach, the layout of facilities can be changed from period to period in accordance with changes in product demands. Therefore, although this method has the advantage of having optimal layout for each period, it

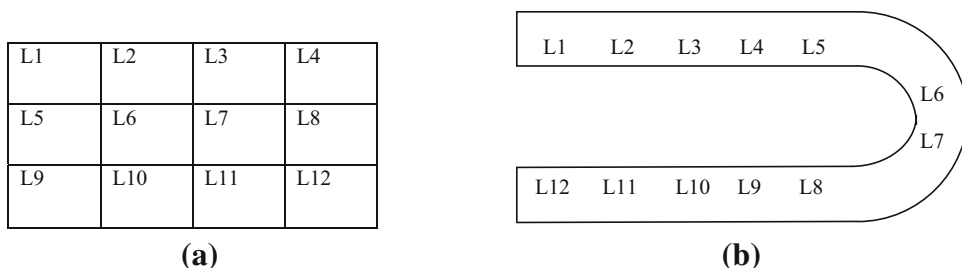
Table 1 Notations of the proposed models

Notation	Description
K	Total quantity of parts
M	Total quantity of machines/locations of machine
T	Total quantity of periods
k	Part index ($k = 1, 2, \dots, K$)
t	Period indicator ($t = 1, 2, \dots, T$)
i, j	Machine indices ($i, j = 1, 2, \dots, M$); $i \neq j$
l, q	Machine location indices ($l, q = 1, 2, \dots, M$); $l \neq q$
N_{ki}	Process number for the process performed on part k by machine i
f_{ijk}	Materials flow linking machines i and j in period t created by part k
f_{ijk}	Materials flow linking machines i and j created by part k
f_{ij}	Materials flow linking machines i and j in period t created by all parts
f_{ij}	Materials flow linking machines i and j created by all parts
D_{tk}	Part k demand during period t
B_k	Part k batch volume
C_{tk}	Cost of movements for part k in period t
C_k	Present value of the movement cost per batch for part k
I_r	Interest rate
a_{iilq}	Cost of shifting machine i from location l to location q in period t
a_{oilq}	Present value of cost of shifting machine i from location l to location q
d_{lq}	Distance from machine location l to machine location q
x_{iil}	Decision variable for dynamic machine layout problem
$C(\pi)$	Total cost of layout π
Z_p	Value of the standard normal variable Z by considering confidence level p
$E()$	Expectation
$\text{Var}()$	Variance
$\text{Cov}()$	Covariance
$U(\pi, p)$	Maximum value (upper bound) of $C(\pi)$ with the confidence level p
OFV_{dm}	The objective function of the dynamic machine layout design model

suffers from the disadvantage of having the facility rearrangement cost. Actually, in the dynamic approach, considering each period as a stage, the multi-period problem can be considered as a multi-stage dynamic system with optimal behaviour from stage to stage. In this section, the SDFLP is formulated using the dynamic approach so that an optimal machine layout is obtained for each period. According to the assumption of assigning the equal-sized machines to the same number of known locations, the QAP model of the DFLP proposed by Balakrishnan et al. (1992) is used for formulating the SDFLP. This model is given in equations from (1)–(5).



Fig. 1 **a** Rectangular configuration and **b** U-shaped configuration



$$\begin{aligned} \text{Minimize } & \sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^M \sum_{l=1}^M \sum_{q=1}^M f_{ij} d_{lq} x_{til} x_{tjq} \\ & + \sum_{t=2}^T \sum_{i=1}^M \sum_{l=1}^M \sum_{q=1}^M a_{tilq} y_{tilq} \end{aligned} \tag{1}$$

Subject to:

$$\sum_{i=1}^M x_{til} = 1; \forall t, l \tag{2}$$

$$\sum_{l=1}^M x_{til} = 1; \forall t, i \tag{3}$$

$$x_{til} = \begin{cases} 1 & \text{if facility } i \text{ is assigned to location } l \text{ in period } t \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

$$y_{tilq} = x_{(t-1)il} \times x_{tiq} \tag{5}$$

where, f_{ij} denotes the total materials current in period t from facility i to facility j and vice versa. The distance between facility locations l and q is represented by d_{lq} . The fixed cost of shifting facility i from location l to location q in period t is denoted by a_{tilq} . The objective function given in Eq. (1) is the sum of two terms, including the total MHC and the total rearrangement cost. The constraints (2) and (3) ensure assigning each facility in each period to exactly one location and vice versa. Equation (4) represents the decision variables that are the solutions to the problem so that they determine the location of each facility in each period. Equation (5) indicates that $y_{tilq} = 1$ if the facility i is shifted from location l in period $t - 1$ (i.e. $x_{(t-1)il} = 1$) to location q in period t (i.e. $x_{tiq} = 1$). The flow of materials linking machines i and j in period t created by part k can be calculated using Eq. (6), where the condition $|N_k - N_{k'}| = 1$ refers to two consecutive operations, which are done on part k by machines i and j . Since the demand is divided by the batch size, the quantity of the flow should be discrete value. As mentioned in the assumptions of the problem, the demand for part k in period t (D_{tk}) is a random variable with normal distribution. Therefore, according to Eq. (6), the materials current created by part k in period t from facility i to facility j and vice versa (f_{ijk}) is also a random variable with a normal distribution having the expectation and variance given in Eqs. (7) and (8), respectively.

$$f_{ijk} = \begin{cases} \frac{D_{tk}}{B_k} C_{tk} & \text{if } N_{ki} - N_{kj} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

$$E_{ijk} = \begin{cases} \frac{D_{tk}}{B_k} C_{tk} & \text{if } N_{ki} - N_{kj} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

$$\text{Var}_{ijk} = \begin{cases} \frac{D_{tk}}{B_k^2} C_{tk}^2 & \text{if } N_{ki} - N_{kj} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

The total materials current linking machines i and j in period t created by all parts (i.e. f_{ij}) is obtained using Eq. (9) in which f_{ijk} is a random variable with normal distribution and thereby f_{ij} is also a random variable with a normal distribution having the expectation and variance shown in Eqs. (10) and (11), respectively. According to assumption (4), in each period, demands of parts are dependent. It means that $\text{Cov}(D_{tk}, D_{tk'}) \neq 0$ and thereby $\text{cov}(f_{ijk}, f_{ijk'}) \neq 0$. Kulturel-Konak et al. (2004) has already used an equation similar to Eq. (11), but for a single period problem. Inserting Eqs. (7) and (8) into Eqs. (10) and (11) leads to the new form of the expectation and variance of f_{ij} as represented in Eqs. (12) and (13), respectively. Utilising Eqs. (1) and (5), the total cost for a given dynamic machine layout π_{dm} , which is denoted by $C(\pi_{dm})$, is calculated using Eq. (14). In this equation, the total cost is equal to the summation of the total MHC (the first term) and the total rearrangement cost (the second term).

$$f_{ij} = \sum_{k=1}^K f_{ijk} \tag{9}$$

$$E(f_{ij}) = \sum_{k=1}^K E(f_{ijk}) \tag{10}$$

$$\text{Var}(f_{ij}) = \sum_{k=1}^K \text{Var}(f_{ijk}) + 2 \sum_{k=1}^K \sum_{k'=k+1}^K \text{cov}(f_{ijk}, f_{ijk'}) \tag{11}$$

$$E(f_{ij}) = \sum_{k=1}^K \frac{E(D_{tk})}{B_k} C_{tk} \tag{12}$$

$$\begin{aligned} \text{Var}(f_{ij}) = & \sum_{k=1}^K \frac{\text{Var}(D_{tk})}{B_k^2} C_{tk}^2 \\ & + 2 \sum_{k=1}^K \sum_{k'=k+1}^K \frac{C_{tk} C_{tk'}}{B_k B_{k'}} \text{cov}(D_{tk}, D_{tk'}) \end{aligned} \tag{13}$$

$$C(\pi_{dm}) = \sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^M \sum_{l=1}^M \sum_{q=1}^M f_{ij} d_{lq} x_{il} x_{tjq} + \sum_{t=2}^T \sum_{i=1}^M \sum_{l=1}^M \sum_{q=1}^M a_{tilq} x_{(t-1)il} x_{tiq} \tag{14}$$

Since f_{ij} is a random variable with normal distribution, then according to Eq. (14), $C(\pi_{dm})$ is also a normally distributed random variable (Braglia et al. 2005). Using Eq. (14), the expected value and variance of $C(\pi_{dm})$ are given in Eqs. (15) and (16), respectively. According to the assumption (3), demands of a particular part in different time periods are independent. It means that $Cov(D_{tk}, D_{t'k}) = 0$ and thereby in Eq. (16), $Cov(f_{ij}, f_{i'j'}) = 0$, but as earlier mentioned, $Cov(D_{tk}, D_{tk'}) \neq 0$.

$$E(C(\pi_{dm})) = \sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^M E(f_{ij}) \sum_{l=1}^M \sum_{q=1}^M d_{lq} x_{il} x_{tjq} + \sum_{t=2}^T \sum_{i=1}^M \sum_{l=1}^M \sum_{q=1}^M a_{tilq} x_{(t-1)il} x_{tiq} \tag{15}$$

(16), (17), and (18), the new form of the expectation and variance of the total cost are given in Eqs. (19) and (20), respectively. For a given dynamic machine layout π_{dm} if the decision maker considers $U(\pi_{dm}, p)$ as the maximum value (upper bound) of $C(\pi_{dm})$ with the confidence level p , then $U(\pi_{dm}, p)$ given in Eq. (21) can be minimised instead of minimising $C(\pi_{dm})$ (Kulturel-Konak et al. 2004; Moslemipour and Lee 2012; Norman and Smith 2006; Tavakoli-Moghaddam et al. 2007)

$$C_{tk} = C_k(1 + I_r)^t \tag{17}$$

$$a_{tilq} = a_{0ilq}(1 + I_r)^t \tag{18}$$

$$E(C(\pi_{dm})) = \left(\sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^M \sum_{k=1}^K \frac{E(D_{tk})}{B_k} C_k(1 + I_r)^t \sum_{l=1}^M \sum_{q=1}^M d_{lq} x_{il} x_{tjq} + \sum_{t=2}^T \sum_{i=1}^M \sum_{l=1}^M \sum_{q=1}^M a_{tilq} x_{(t-1)il} x_{tiq} \right) \tag{19}$$

$$Var(C(\pi_{dm})) = \sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^M \sum_{k=1}^K \left(\frac{Var(D_{tk})}{B_k^2} C_k^2(1 + I_r)^{2t} + 2 \sum_{k=1}^K \sum_{k'=k+1}^K \frac{C_k C_{k'}}{B_k B_{k'}} (1 + I_r)^{2t} cov(D_{tk}, D_{tk'}) \right) \left(\sum_{l=1}^M \sum_{q=1}^M d_{lq} x_{il} x_{tjq} \right)^2 \tag{20}$$

$$Var(C(\pi_{dm})) = \sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^M Var(f_{ij}) \left(\sum_{l=1}^M \sum_{q=1}^M d_{lq} x_{il} x_{tjq} \right)^2 \tag{16}$$

$$U(\pi_{dm}, p) = E(C(\pi_{dm})) + Z_p \sqrt{Var(C(\pi_{dm}))} \tag{21}$$

The mathematical model to obtain the optimal layout of machines for each period of the SDFLP can be written as follows using Eqs. (19), (20) and (21):

Minimisation of

Since we consider time value of money, C_{tk} and a_{tilq} can be calculated using Eqs. (17) and (18), respectively. In these equations, C_k is the present value of the movement cost for part k , a_{0ilq} is the present value of a_{tilq} and I_r is the interest rate for each period. Using Eqs. (12), (13), (15),

$$OFV_{dm} = \left\{ Z_p \left[\sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^M \sum_{k=1}^K \frac{E(D_{tk})}{B_k} C_k(1 + I_r)^t \sum_{l=1}^M \sum_{q=1}^M d_{lq} x_{il} x_{tjq} + \sqrt{\sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^M \sum_{k=1}^K \left(\frac{Var(D_{tk})}{B_k^2} C_k^2(1 + I_r)^{2t} + 2 \sum_{k=1}^K \sum_{k'=k+1}^K \frac{C_k C_{k'}}{B_k B_{k'}} (1 + I_r)^{2t} cov(D_{tk}, D_{tk'}) \right) \left(\sum_{l=1}^M \sum_{q=1}^M d_{lq} x_{il} x_{tjq} \right)^2} \right] + \sum_{t=2}^T \sum_{i=1}^M \sum_{l=1}^M \sum_{q=1}^M a_{tilq} x_{(t-1)il} x_{tiq} \right\} \tag{22}$$

Subject to: Equations from (2)–(4)

Proposed the CS-SA hybrid algorithm to solve the SDFLP

Due to the fact that SA is a single search algorithm, the solution quality and computational time are affected by the initial solution. In addition, it is a well-known local search technique, which has been used to improve the quality of a known initial solution generated by other algorithms. Therefore, in the proposed hybrid algorithm, to conquer the reliance of SA on initial solution, a population of initial solutions is used rather than one initial solution. The proposed hybrid CS-SA algorithm contains three stages as follows: (1) generating a population of feasible initial solutions (antibodies) randomly, (2) selecting and cloning the antibodies produced in the first stage using the CS algorithm, and (3) improving the cloned antibodies using SA algorithm. The notations used in this section are shown in Table 2 apart from the ones given in Table 1. The three above-mentioned stages are described in the following sub-sections.

Stage 1: initial solution construction

It is essential to state that in the CSA, antibodies are potential solutions of the problem. In this stage, a population of the potential initial solutions consists of the antibodies are randomly generated and the number of antibodies is defined as the population size. It is assume that the population size to be equivalent of the number of time periods (T) in the SDFLP. In this stage, T initial

Table 2 Notations for CS-SA algorithm

Notation	Description
t	Index for period/ant ($t = 1, 2, \dots, T$)
f_{ijk}	Materials flow for part k between machines i and j in period t
f_{ijk}	Average flow of part k
f_{ij}	Materials flow for all parts between machines i and j
c_{ij}	Total part handling cost between machines i and j
$\pi(i)$	The location containing the machine i
$d_{\pi(i)\pi(u)}$	The distance between the locations where machines i and u are placed
f_{vj}	The flow of parts between machines v and j
T_{in}	Initial temperature in SA
θ	Cooling ratio in SA
P_{el}	The probability of accepting non-improving neighbouring solution in SA

solutions are randomly generated during the two following steps:

Step 1: select T permutations of $1, 2, \dots, M$ as the first row of T matrices

Step 2: to consider each of the permutations for all periods

Each of the T partial solutions (layouts) constructed in Step 1 are considered for all periods in Step 2. By doing so, a population of initial solutions with size T is constructed for the SDFLP. It is necessary to mention that a solution of the SDFLP is represented by a two dimensional matrix where each row represents a period, each column represents a location, and each element represents a machine number. For example, for a dynamic layout problem having three time periods and three facilities including machines 1, 2, and 3, the population of initial solutions can be in the following form:

$$\text{Population of initial solutions} = \{s_1^0, s_2^0, s_3^0\} = \left\{ \begin{bmatrix} 3 & 1 & 2 \\ 3 & 1 & 2 \\ 3 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 2 \\ 1 & 3 & 2 \\ 1 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 3 \\ 2 & 1 & 3 \\ 2 & 1 & 3 \end{bmatrix} \right\}$$

Stage 2: selection and cloning antibodies

As mentioned in section “Clonal selection algorithm”, CS is a population-based algorithm. Therefore, the population of initial solutions constructed in Stage 1 can be easily applied to the CS algorithm. In this stage, only the selection and cloning steps of CS are used to amplify the power of the good initial solutions constructed in the first stage. Therefore, a number of the best initial antibodies are selected and cloned according to their affinity value. The affinity of the initial solution (antibody) s_i^0 is defined as the inverse of its objective function value as given in Eq. (23). In this approach, better antibodies have the higher value of affinity.

$$\text{Affinity}(s_i^0) = \frac{1}{f(s_i^0)} \tag{23}$$

During selection process, n ($n < T$) best antibodies having the highest affinity values are selected for cloning. Actually, the affinity values of the best n antibodies are sorted in descending order. Doing so, for $i < j$, Affinity (s_i^0) > Affinity (s_j^0). Using cloning process, the selected antibodies are cloned according to Eq. (24) extracted from De Castro and Von Zuben (2002) so that the antibody with higher affinity value has higher number of clones.

$$NC_i = \text{round}\left(\frac{\delta * n}{i}\right) \tag{24}$$

where, i denotes the rank of antibody (for the antibody with the highest affinity $i = 1$), NC_i is the number of clones for antibody s_i^0 , δ is a size factor (Here, $\delta = 1$), n is the number

of selected antibodies, and *round* refers to the closest integer. Using this approach, the size of the new population would be the summation of NC_i .

To illustrate the aforementioned procedure, we consider the example of a population of initial solutions given in section 4.1. We assume that the initial solutions are arranged in descending order so that $\text{Affinity}(s_1^0) > \text{Affinity}(s_2^0) > \text{Affinity}(s_3^0)$. It is also assumed that all of the initial solutions are selected (i.e., $n = T = 3$). According to the order of the initial solutions, the rank of s_1^0 , s_2^0 , and s_3^0 (i.e., i in Eq. (24)) will be 1, 2, and 3, respectively. Thus, using Eq. (24), the number of clones (NC_i) for the initial solutions (antibodies) s_1^0 , s_2^0 , and s_3^0 are equals to 3, 2, and 1, respectively. By doing so, the final population of initial solutions (antibodies), which is applied to the SA algorithm in the next stage, is as follows:

$$\left\{ \begin{bmatrix} 3 & 1 & 2 \\ 3 & 1 & 2 \\ 3 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 3 & 1 & 2 \\ 3 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 3 & 1 & 2 \\ 3 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 2 \\ 1 & 3 & 2 \\ 1 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 2 \\ 1 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 3 \\ 2 & 1 & 3 \\ 2 & 1 & 3 \end{bmatrix} \right\}$$

CS also includes mutation and receptor editing processes used for solution improvement and to avoid from getting trapped in local optima, respectively. However, in this stage, there is no need to use the aforementioned processes. Because they are done in the third stage of the hybrid algorithm (i.e., SA algorithm). In the inner loop of SA algorithm, the solution is improved by searching for a better neighbouring solution and accepting non-improving neighbouring solutions prevents the algorithm from getting trapped in local optima.

Stage 3: initial solutions improvement

In this stage as the last part of the hybrid algorithm, SA algorithm is used to find the optimal solution of the problem at hand. In the proposed hybrid algorithm, to conquer the reliance of SA on initial solution, a population of cloned initial solutions is used rather than one initial solution. Thus, the cloned initial solutions (antibodies) constructed by the two previous stages, are applied to the SA algorithm according to their order in the population. By doing so, the first initial solution, which has the highest affinity, is the starting initial solution. In general, SA starts with the initial solution s^0 and a high initial temperature T_{in} . As mentioned, the solutions are in the form of matrix and the SA algorithm consists of two loops, including inner loop and outer loop. At each iteration of the inner loop, a neighbouring solution s' for the best current solution s , which is obtained in the previous iteration, is generated

using the local search technique named random exchange method. In this method, first, a row (period) is selected at random. Then, two columns (locations) are randomly selected. Finally, the elements (machines) in the two locations are exchanged. The solution s' is evaluated by the objective function f given in Eq. (2). It is accepted as the current best solution if $f(s') \leq f(s)$. In the case of $f(s') > f(s)$ it is also accepted if $x \in (0, 1) \leq P_{el}$, where x is a randomly generated number and P_{el} is the probability of accepting this non-improving neighbouring solution s' at iteration el of the outer loop. The outer loop starts with a high value of temperature. At each iteration of this loop or in other words, at each temperature, the inner loop is repeated until the system reaches the steady state or thermal equilibrium. In this paper, for solving the SDFLP with M facilities (machines) and T periods, the maximum

number of iterations for the inner loop il_{max} is calculated by Eq. (25).

$$il_{max} = \varphi \times M \times T \quad (25)$$

where, experimentally, $\varphi = 0.5$ for small-sized problems. It is increased proportional to the size of problem. The temperature is gradually reduced by running the outer loop so that its current value in the iteration el of this loop is calculated using Eq. (26), where, $\theta \in (0.80, 0.99)$ is the cooling ratio (here, $\theta = 0.95$), el_{max} denotes the maximum number of iterations of the outer loop, and T_{in} denotes the initial value of the temperature. According to McKendall et al. (2006), the initial temperature is calculated by Eq. (27).

$$T_{el} = T_{in} \theta^{el}; \quad (el = 0, 1, \dots, el_{max}) \quad (26)$$

$$T_{in} = -\frac{0.1f(s^0)}{\ln(0.25)} \quad (27)$$

The probability of accepting the non-improving solution is given in Eq. (28). Using Eq. (28), the value of temperature at iteration el (i.e. T_{el}) can be calculated by Eq. (29). Using Eq. (29), the initial temperature T_{in} and final temperature T_f are given in Eqs. (30) and (31), respectively, where, P_{in} and P_f are the initial and final values of P_{el} . Using Eqs. (30) and (31), Eq. (32) can be written. The maximum number of iterations of the outer loop el_{max} corresponds to T_f can be calculated using Eq. (33) that can be rewritten as Eq. (34). In fact, in Eq. (26), if we put

Fig. 2 The pseudocode of the hybrid CS-SA algorithm

```

procedure Hybrid CS-SA Algorithm
  Initialization of the parameters  $T_{in}, \theta, il_{max}, el_{max}, il = 0, el = 0$ 
  Stage 1: Randomly Generation of  $T$  Initial Solutions ; %  $T$  denotes the number of time periods or population size

  Step 1: Select  $T$  permutations of 1, 2, ...,  $M$  as the first row of  $T$  matrices
  Step 2: To consider each of the permutations for all periods

  Stage 2: CS
  Step 3: Calculate the affinity value of each antibody by Eq. (22)
  Step 4: Sort the affinity values in descending order
  Step 5: Select  $n$  best initial solutions that have higher affinity value
  Step 6: Clone the selected initial solutions according to Eq. (23)

  Stage 3: SA
  for  $j = 1$  : Summation of  $NC_i$ 
   $s = s_i^0$ 
  while outer-loop criterion not satisfied (i.e.  $el \leq el_{max}$  )
  while inner-loop criterion not satisfied (i.e.  $il \leq il_{max}$  )
  if  $f(s) \leq f(s')$  %  $s'$  is a neighbouring solution of  $s$  .
   $s = s'$ 
  end
  if  $f(s) > f(s')$ 
  if randomly generated  $x \in (0, 1) \leq e^{-\frac{f(s)-f(s')}{T_{el}}}$ 
   $s = s'$ 
  end
  end
   $il = il + 1$ 
  end
   $T_{el} = T_{in} \theta^{el}$  %Update Temp  $T_{el}$  ; Cooling ratio  $\theta \in (0.80, 0.99)$ 
   $el = el + 1$ 
  end
   $j = j + 1$ 
  end Hybrid CS-SA Algorithm
  
```

$T_{el} = T_f$ and $el = el_{max}$ then Eq. (33) is obtained. The Eq. (33) can be rearranged as Eq. (34). The final form of Eq. (34) can be written as Eq. (35) using Eqs. (30) and (31).

$$P_{el} = e^{-\frac{f(s)-f(s')}{T_{el}}} \tag{28}$$

$$T_{el} = \frac{f(s) - f(s')}{\ln(P_{el})} \tag{29}$$

$$T_{in} = \frac{f(s) - f(s')}{\ln(P_{in})} \tag{30}$$

$$T_f = \frac{f(s) - f(s')}{\ln(P_f)} \tag{31}$$

$$T_f = T_{in} \left(\ln \frac{P_{in}}{P_f} \right) \tag{32}$$

$$T_f = T_{in} \theta^{el_{max}} \tag{33}$$

$$el_{max} = \log_{\theta} \left(\frac{T_f}{T_{in}} \right) \tag{34}$$

$$el_{max} = \log_{\theta} \left(\ln \frac{P_{in}}{P_f} \right) \tag{35}$$

The probability of accepting the non-improving solution P_{el} is high at the initial stages of the algorithm ($P_{in} \approx 1$) and it is reduced gradually while the algorithm is repeated so that it approaches to a very small value at the final stages of the algorithm ($P_f \approx 0$). Finally, the Pseudocode of proposed hybrid is shown in Fig. 2.

Qualitative analysis of the CS-SA algorithm

In this section, the performance of the proposed hybrid algorithm is investigated from both of the solution quality and computational time points of view in a qualitative

manner. In other words, it will be explained that how the good initial solutions generated by CS lead to improve both of the solution quality and the computational time of SA algorithm. First, the effect of the randomly generated initial solutions improved by CS algorithms on the computational time of the SA is investigated. To this end, the maximum number of iterations of the outer and the inner loops of the SA is considered.

To estimate the maximum number of iterations of the outer loop (el_{\max}) of the SA used in the proposed hybrid algorithm, we assume that, $P_{\text{in}} = 0.95$, $P_f = 10^{-15}$ and $\theta = 0.95$ (Baykasoglyu and Gindy 2001). Doing so, it is calculated as $el_{\max} \approx 127$ using Eq. (35). According to Eq. (30), in the proposed algorithm, starting SA algorithm with a better initial solution s^0 generated by CS leads to lower value of the cost function $f(s^0)$ and in turn causes to decrease the initial temperature T_{in} . Referring to Eq. (34), decrease in the initial temperature results in decrease in the maximum number of the outer loop iterations. Therefore, the maximum number of iterations of the outer loop of the SA used in the hybrid algorithm is less than the above-mentioned estimated value (i.e., $el_{\max} \approx 127$). Starting the inner loop of the SA with a good initial solution drives the algorithm to search for a better solution in the neighbouring area of the good solution. Doing so, the local search technique used in the inner loop of SA is reached to a frozen (equilibrium) state very fast. When a frozen state is reached, the chance of finding new better solutions is quite small (Heragu 1997). In other words, there is no improvement in the solution quality after several iterations. Therefore, using the good initial solutions the maximum number of iterations of the inner loop (il_{\max}) is also reduced.

Due to reduction in both of the el_{\max} and the il_{\max} , the SA algorithm converges to a global optimal solution faster than the case in which it starts with only one known or randomly generated initial solution. As a result, the computational time elapsed by the SA algorithm is decreased by starting with a population of good initial solutions constructed by CS algorithm.

The good initial solutions constructed by CS algorithm also result in improvement in the quality of the final solution obtained by the SA algorithm. As mentioned, in SA algorithm, the quality of solution is improved by searching for a neighbouring solution using a local search technique. This technique is amplified by starting the SA algorithm with a good initial solution. Because, by doing so, the local search technique forces the algorithm to search in the neighbouring area of the good solution to hope to find a better solution. The quality of the neighbourhood solution affects the success of the local search method. Actually, this process behaves like the intensification

strategy, which is used in other metaheuristics, for instance, in the TS algorithm. The intensification strategy drives the algorithm to search for a better solution in the neighbouring area of the good solution. As a result, amplification of the local search technique by starting with good initial solution leads to improve the quality of the final solution of SA algorithm. The proposed hybrid algorithm is generic that can be used to solve any forms of the mathematical models of the FLP, which can be different by considering some critical constraints such as using unequal-sized machines, the shop floor with different dimensions and shapes. Since initial solutions constructed by the CS algorithm are improved using a local search technique such as random pair-wise exchange method used in the SA algorithm, they are subject to randomly change at each iteration of the SA method. Therefore, the initial solutions are not affected by some constraint such as the case that two facilities can not be placed consecutively, and budget constraint. Instead, such constraints should be considered in the third stage (i.e. SA algorithm) of the hybrid algorithm.

Computational results and discussion

In this section, to validate the proposed model and also to evaluate the performance of the proposed hybrid CS-SA algorithm, design of experiment and benchmark methods are used as explained in the two following sub-sections:

Design of experiment

In this section, design of experiment method is utilised to verify and validate the proposed model and the hybrid CS-SA algorithm, which is proposed for solving the SDFLP. To this end, a large number (say, 1000) of randomly generated test problems are applied to the model and solved using the the hybrid CS-SA algorithm. A personal computer with Intel 2.10 GHZ CPU and 3 GB RAM is used to run the hybrid and SA algorithms, which are programmed in Matlab. Since there is no historical data on expectation and variance of demands in each period, design of experiment method is used to evaluate the performance of the proposed CS-SA hybrid algorithm. To this end, 1000 randomly generated test problems are applied to the dynamic machine layout design model given in Eq. (22) and solved using the proposed hybrid CS-SA and the SA algorithms. Actually, the performance of the SA algorithm starting with a population of initial solutions, which is generated by the CS algorithm, is compared with the performance of the SA algorithm starting with a single known initial solution. Each test problem has the following input data: Expectation and variance of parts demand (E and V) are randomly

generated with uniform distribution so that $E \in (1000, 10000)$ and $V \in (1000, 3000)$. The distance matrix is given in Table 3. The number of machines and the number of periods are twelve and ten ($M = 12, T = 10$), respectively. The facility rearrangement cost (a_{ilq}) is equal to 1000. In Eq. (2), number of parts (k), transfer batch size (B_k), and part movement cost (C_{ik}) are set to one.

According to Freund (1992), the $100*(1 - \alpha)\%$ confidence interval for difference between means of two populations is calculated as Eq.(36), where, n_1 and n_2 are sample size, \bar{x}_1 and \bar{x}_2 are sample means, σ_1^2 and σ_2^2 are sample variances, and $z_{\alpha/2}$ is standard normal Z value so that

$$\Pr\left(-z_{\alpha/2} \leq Z \leq z_{\alpha/2}\right) = 1 - \alpha.$$

$$(\bar{x}_1 - \bar{x}_2) - z_{\alpha/2} \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}} < \mu_1 - \mu_2 < (\bar{x}_1 - \bar{x}_2) + z_{\alpha/2} \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}} \tag{36}$$

$$\bar{x} = \frac{\sum_{i=1}^n x}{n} \tag{37}$$

Table 3 Distance between facility locations

From	To											
	1	2	3	4	5	6	7	8	9	10	11	12
1	0	10	20	30	40	50	70	60	50	40	30	20
2	10	0	10	20	30	40	60	50	40	30	20	30
3	20	10	0	10	20	30	50	40	30	20	30	40
4	30	20	10	0	10	20	40	30	20	30	40	50
5	40	30	20	10	0	10	30	20	30	40	50	60
6	50	40	30	20	10	0	20	30	40	50	60	70
7	70	60	50	40	30	20	0	10	20	30	40	50
8	60	50	40	30	20	30	10	0	10	20	30	40
9	50	40	30	20	30	40	20	10	0	10	20	30
10	40	30	20	30	40	50	30	20	10	0	10	20
11	30	20	30	40	50	60	40	30	20	10	0	10
12	20	30	40	50	60	70	50	40	30	20	10	0

Table 4 A comparison between the hybrid and SA algorithms

95% confidence interval for	CS-SA hybrid algorithm		SA algorithm	
$[-2641300, -2064100]$	\bar{x}_{Hybrid}	σ_{Hybrid}^2	\bar{x}_{SA}	σ_{SA}^2
	2.8803e + 008	1.0261e + 013	2.9039e + 008	1.1425e + 013
Computational time (sec.)	15.8		16.7	

$$\sigma^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 \tag{38}$$

After solving each test problem, the optimal value of total cost (objective function) is obtained as a sample of a population. Actually, two populations can be considered as follows. The first population includes optimal cost values obtained by the proposed hybrid algorithm and the second one is the population of optimal cost values obtained by the SA algorithm with a known initial solution. Using Eqs. (36)–(38) the sample means (\bar{x}_{Hybrid} and \bar{x}_{SA}) and sample variances (σ_{Hybrid}^2 and σ_{SA}^2) of the optimal cost values along with the calculated 95% confidence interval for difference between means of the populations are shown in Table 4. According to the computational results, both of the lower and upper bounds of the confidence interval are negative, and thereby $\mu_{Hybrid} < \mu_{SA}$. The average computation time elapsed for each problem is also displayed in Table 4. According to the results, the proposed hybrid algorithm has better performance than the SA algorithm from both the solution quality and computation time standpoints. Besides, this section indicates the capability of the proposed model and the hybrid algorithm to cope with the stochastic environment of facility layout problems.

Benchmark method

Since there is no historical data on the expectation, variance, and covariance of demands in each period, the proposed model is tested in a deterministic environment by comparing with previous approaches as benchmark. To this end, a 50% percentile level ($p = 50\%$) equivalent of $z_p = 0$ is applied to the model. By doing so, the second term of the proposed model containing variance and covariance of part demands is ignored, thereby there is no need to data on variance and covariance of part demands in each period. In this case, demand of parts in each period, which is known in deterministic case, is regarded as the expectation of part demands in the proposed model. Data set used for testing the model are taken from Yaman et al. (1993). The problem includes nine machines and five periods. Data on machine sequence and part demand in different periods are given in Tables 5 and 6, respectively. Rectangular layout configuration is considered so that a location grid of 3×3 is used as facilities locations. Part

Table 5 Data on machine sequence

Part no.	Machine sequence
1	1 → 3→5 → 7→2 → 7→9
2	1 → 4→2 → 5→6 → 8→9
3	1 → 5→7 → 8→5 → 6→2 → 9
4	1 → 2→4 → 6→7 → 8→2 → 3→9
5	1 → 7→6 → 4→2 → 8→3 → 5→6 → 9

Table 6 Part demand in different periods

Part no.	Period				
	1	2	3	4	5
1	10	35	90	40	55
2	30	50	25	65	20
3	45	15	40	70	15
4	70	80	55	90	85
5	85	60	70	20	30

movement cost, and batch size are set to ten and one, respectively.

Yaman et al. (1993)'s problem is applied to the proposed model. The proposed CS-SA hybrid algorithm is used to solve the model. To compare the performance of the proposed model with the previous approaches, which deal with the static environment, facility relocating cost is set to zero. Madhusudan-Pillai et al. (2011) proposed a SA algorithm to solve their robust layout design model, which is termed MP-SA hereafter. Another heuristic method was also developed by Irappa and Madhusudan (2008) to solve their model, which is denoted by I-M hereafter. The results of the proposed hybrid algorithm and the results of different approaches in literature are shown in Table 7. The findings contain the MHC of the optimal layout in each period and the total MHC over the entire planning horizon. On comparison, better performance of the proposed CS-SA hybrid algorithm is concluded so that it leads to 2.1% improvement with respect to the best previous approach (i.e. MP-SA). Besides, the approaches in

literature except for the I-M and MP-SA methods, can be used only for problems having maximum nine machines, whereas the proposed approach including the dynamic model and the hybrid algorithm can be applied to any size of the problem. Finally, this section shows the capability of the proposed model and the hybrid algorithm to deal with the deterministic situation of facility layout problems.

Conclusion

In this paper, a novel hybrid CS-SA algorithm along with a new QAP-based mathematical model to design a dynamic facility layout was proposed to solve the SDFLP. Solving the proposed model leads to design of an optimal facility layout in each period of the multi-period time planning horizon. In the proposed algorithm, a population of randomly generated feasible initial solutions is improved using the CS algorithm. In fact, in the proposed hybrid algorithm, the SA algorithm starts with a population of good initial solutions rather than a single initial solution. The performance of the proposed model and the hybrid algorithm was evaluated using design of experiment (a large number of randomly generated test problems) and benchmark (data from the literature) methods with the following conclusions: (1) on the basis of computational results obtained by the design of experiment method, the proposed hybrid algorithm has outstanding performance in comparison with the SA algorithm starting with a single initial solution from both solution quality and computational time standpoints; (2) according to findings of benchmark method, better performance of the proposed CS-SA hybrid algorithm is concluded so that it leads to 2.1% improvement in total cost with respect to the best previous approach; (3) in spite of some approaches in literature, the proposed hybrid algorithm can be applied to any size of the problem; (4) in the proposed dynamic layout design model, the relative contribution of the expectation and variance of demands on the total cost can be controlled using a decision maker's defined percentile p value; (5) in practice, the proposed model can be applied to both of the stochastic and

Table 7 MHC of the proposed model and previous approaches

Approach	Period					Planning horizon
	1	2	3	4	5	
MP-SA (2011)*	27800	26400	29500	30200	22000	135900
I-M (2008)	29100	27200	31200	30200	22350	140050
Chan et al. (2004)	28500	27900	31600	31650	22750	142400
Tang and Abdel-Malek (1996)	28200	29800	32000	31000	23550	144550
Yaman et al. (1993)	34700	33500	35700	40650	29750	174300
Proposed hybrid algorithm**	27100	26200	28900	29050	21800	133050

*Computational time = 29625 s

**Computational time = 33 s



the deterministic environments of manufacturing systems. Regarding the limitations of this research, since the proposed model has been developed based on QAP formulation, it can be applied only to layout problems having equal-sized facilities such as Raytheon Aircraft Company (Krishnan et al. 2006) and Vought Aerospace Company in Dallas, Texas (Groover 2008) as two real cases. Finally, the following works can be taken into consideration in the future researches:

- To propose a more effective hybrid meta-heuristic approach by combining the SA algorithm with other intelligent approaches to improve the quality of the solution and the computational time.
- To apply the population of initial solutions constructed by the CS, to other algorithms such as GA, TS, and PSO approaches.
- To consider some constraints such as unequal-sized machines, adding and removing machines in different periods, closeness ratio, aisles, routing flexibility, and budget constraint for total cost.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Abtahi A-R, Bijari A (2017) A novel hybrid meta-heuristic technique applied to the wellknown benchmark optimization problems. *J Ind Eng Int*. doi:10.1007/s40092-016-0170-x,13,93-105
- Ashtiani B, Aryanezhad MB, Farhang Moghaddam B (2007) Multi-start simulated annealing for dynamic plant layout problem. *J Ind Eng Int* 3(4):44–50
- Azadeh S, Haghighi SM, Asadzadeh SM (2014) A novel algorithm for layout optimization of injection process with random demands and sequence dependent setup times. *J Manuf Syst* 33:287–302
- Balakrishnan J, Jacobs FR, Venkataramanan MA (1992) Solutions for the constrained dynamic facility layout problem. *Eur J Oper Res* 57:280–286
- Bashiri M, Karimi H (2012) Effective heuristics and meta-heuristics for the quadratic assignment problem with tuned parameters and analytical comparisons. *J Ind Eng Int*. doi:10.1186/2251-712X-8-6
- Baykasoglyu A, Gindy NNZ (2001) A simulated annealing algorithm for dynamic layout problem. *Comput Oper Res* 28:1403–1426
- Braglia M, Zaroni S, Zavanella L (2005) Robust versus stable layout design in stochastic environments. *Prod Plan Control* 16(1):71–80
- Chan WM, Chan CY, Kwong CK (2004) Development of the MAIN algorithm for a cellular manufacturing machine layout. *Int J Prod Res* 42(1):51–65
- Chen CH, Jing CJ (2005) A hybrid ant colony system for vehicle routing problem with time windows. *Eastern Asia Soc Transp Stud* 6:2822–2836
- De Castro LN, Von Zuben FJ (2002) Learning and optimization using the clonal selection principle. *IEEE Trans Evol Comput* 6(3):239–251
- Derakhshan Asl A, Kuan KY (2015) Solving unequal-area static and dynamic facility layout problems using modified particle swarm optimization. *J Intell Manuf*. doi:10.1007/s10845-015-1053-5
- Dong M, Wu C, Hou F (2009) Shortest path based simulated annealing algorithm for dynamic facility layout problem under dynamic business environment. *Expert Syst Appl* 36:11221–11232
- Fazlelahi FZ, Pournader M, Gharakhani M, Sadjadi SJ (2015) A robust approach to design a single facility layout plan in dynamic manufacturing environments using a permutation-based genetic algorithm. *J Eng Manuf*. doi:10.1177/0954405415615728
- Forghani K, Mohammadi M, Ghezavati V (2013) Designing robust layout in cellular manufacturing systems with uncertain demands. *Int J Ind Eng Comput* 4(2):215–226
- Freund J (1992) *Mathematical statistics*, 5th edn. Prentice-Hall, Upper Saddle River
- Groover MP (2008) *Automation, production systems, and computer-integrated manufacturing*. Pearson Education Inc, New Jersey
- Hasani A, Soltani R, Eskandarpour M (2015) A hybrid meta-heuristic for the dynamic layout problem with transportation system design. *Int J Eng Trans B Appl* 28(8):1175–1185
- Heragu SS (1997) *Facilities design*, 1st edn. PWS Publishing Company, Boston
- Hitchings GG (1970) Control, redundancy, and change in layout systems. *AIIE Trans* 2(3):253–262
- Irappa BH, Madhusudan PV (2008) Development of a heuristic for layout formation and design of robust layout under dynamic demand. In: *Proceedings of the international conference on digital factory, ICDF 2008*, August 11–13, 1398–1405
- Khosravian-Ghadikolaei Y, Shahanaghi K (2013) Multi-floor dynamic facility layout: a simulated annealing-based solution. *Int J Oper Res* 16(4):375–389
- Koopmans TC, Beckman M (1957) Assignment problems and the location of economic activities. *Econometric* 25:53–76
- Krishnan KK, Cheraghi SH, Nayak CN (2006) Dynamic from-between chart: a new tool for solving dynamic facility layout problems. *Int J Ind Syst Eng* 1(1/2):182–200
- Krishnan KK, Cheraghi SH, Nayak CN (2008) Facility layout design for multiple production scenarios in a dynamic environment. *Int J Ind Syst Eng* 3(2):105–133
- Kulturel-Konak S, Konak A (2015) A large-scale hybrid simulated annealing algorithm for cyclic facility layout problems. *Eng Optimization* 47(7):963–978
- Kulturel-Konak S, Smith AE, Norman BA (2004) Layout optimization considering production uncertainty and routing flexibility. *Int J Prod Res* 42(21):4475–4493
- Lee TS, Moslemipour G, Ting TO, Rilling D (2012) A novel hybrid ACO/SA approach to solve stochastic dynamic facility layout problem (SDFLP). *Commun Comput Inf Sci Spec Issue Emerg Intell Comput Technol Appl* 304:100–108
- Li L, Li C, Ma H, Tang Y (2015) An optimization method for the remanufacturing dynamic facility layout problem with uncertainties. *Discrete Dyn Nat Soc*. doi:10.1155/2015/685408
- Madhusudan-Pillai V, Irappa-Basappa H, Krishna KK (2011) Design of robust layout for dynamic plant layout problems. *Comput Ind Eng* 61:813–823
- Matai R (2015) Solving multi-objective facility layout problem by modified simulated annealing. *Appl Math Comput* 261:302–311
- McKendall AR, Shang J, Kuppusamy S (2006) Simulated annealing heuristics for the dynamic facility layout problem. *Comput Oper Res* 33:2431–2444
- Montreuil B, Laforge A (1992) Dynamic layout design given a scenario tree of probable futures. *Eur J Oper Res* 63(271):286



- Moslemipour G, Lee TS (2012) Intelligent design of a dynamic machine layout in uncertain environment of flexible manufacturing systems. *J Intell Manuf* 23(5):1849–1860
- Moslemipour G, Lee TS, Rilling D (2012) A review of intelligent approaches for designing dynamic and robust layouts in flexible manufacturing systems. *Int J Adv Manuf Tech* 60:11–27
- Nematian J (2014) A robust single row facility layout problem with fuzzy random variables. *Int J Adv Manuf Technol* 72:255–267
- Norman BA, Smith AE (2006) A continuous approach to considering uncertainty in facility design. *Comput Oper Res* 33:1760–1775
- Palekar US, Batta R, Bosch RM, Elhence S (1992) Modeling uncertainties in plant layout problems. *Eur J Oper Res* 63:347–359
- Palubeckis G (2015) Fast simulated annealing for single-row equidistant facility layout. *Appl Math Comput* 263:287–301
- Palubeckis G (2016) Single row facility layout using multi-start simulated annealing. *Comput Ind Eng*. doi:[10.1016/j.cie.2016.09.026](https://doi.org/10.1016/j.cie.2016.09.026)
- Pourvaziri H, Naderi B (2014) A hybrid multi-population genetic algorithm for the dynamic facility layout problem. *Appl Soft Comput* 24:457–469
- Ram DJ, Sreenivas TH, Subramaniam KG (1996) Parallel simulated annealing algorithms. *J Parallel Distrib Comput* 37:207–212
- Rezazadeh H, Ghazanfari M, Saidi-Mehrabad M, Sadjadi SJ (2009) An extended discrete particle swarm optimization algorithm for the dynamic facility layout problem. *J Zhejiang Univ Sci A* 10(4):520–529
- Sahin R, Ertogral K, Turkbey O (2010) A simulated annealing heuristic for the dynamic layout problem with budget constraint. *Comput Ind Eng*. doi:[10.1016/j.cie.2010.1004.1013](https://doi.org/10.1016/j.cie.2010.1004.1013)
- Sahni S, Gonzalez T (1976) P-complete approximation problem. *J ACM* 23(3):555–565
- Satheesh Kumar RM, Asokan P, Kumanan S (2009) Artificial immune system-based algorithm for the unidirectional loop layout problem in a flexible manufacturing system. *Int J Adv Manuf Technol* 40:553–565
- Shafiq F, Defersha FM, Moussa SE (2015) A mathematical model for the design of distributed layout by considering production planning and system reconfiguration over multiple time periods. *J Ind Eng Int*. doi:[10.1007/s40092-015-0102-1,11,283-295](https://doi.org/10.1007/s40092-015-0102-1,11,283-295)
- Shirazi H, Kia R, Javadian N, Tavakkoli-Moghaddam R (2014) An archived multi-objective simulated annealing for a dynamic cellular manufacturing system. *J Ind Eng Int* 10:58. doi:[10.1007/s40092-014-0058-6](https://doi.org/10.1007/s40092-014-0058-6)
- Shumsky RA, Zhang F (2009) Dynamic capacity management with substitution. *Oper Res* 57(3):671–684
- Tajbakhsh A, Eshghi K, Shamsi A (2009) A hybrid PSO-SA algorithm for the traveling tournament problem. 978-1-4244-4136-5/09/\$25.00 ©2009 IEEE, 512–518
- Tang C, Abdel-Malek LL (1996) A framework for hierarchical interactive generation of cellular layout. *Int J Prod Res* 34(8):2133–2162
- Tavakkoli-Moghaddam R, Javadian N, Javadi B, Safaei N (2007) Design of a facility layout problem in cellular manufacturing systems with stochastic demands. *Appl Math Comput* 184:721–728
- Tayal A, Singh SP (2016) Integrating big data analytic and hybrid firefly-chaotic simulated annealing approach for facility layout problem. *Ann Oper Res*. doi:[10.1007/s10479-016-2237-x](https://doi.org/10.1007/s10479-016-2237-x)
- Tayal A, Singh SP (2017) Integrated SA-DEA-TOPSIS-based solution approach for multi-objective stochastic dynamic facility layout problem. *Int J Bus Syst Res* 11(1/2):82–100
- Tompkins JA, White JA, Bozer YA, Tanchoco JMA (2003) *Facilities planning*. Wiley, New York
- Ulutas BH, Islier A (2009) A clonal selection algorithm for dynamic facility layout problems. *J Manuf Syst* 28:123–131
- Ulutas B, Islier AA (2015) Dynamic facility layout problem in footwear industry. *J Manuf Syst* 36:55–61
- Ulutas B, Kulturel-Konak S (2012) An artificial immune system based algorithm to solve unequal area facility layout problem. *Expert Syst Appl* 39(5):5384–5395
- Ulutas B, Kulturel-Konak S (2013) Assessing hypermutation operators of a clonal selection algorithm for the unequal area facility layout problem. *Eng Optim* 45(3):375–395
- Vafaiezhad M, Kia R, Shahnazari-Shahrezaei P (2016) Robust optimization of a mathematical model to design a dynamic cell formation problem considering labor utilization. *J Ind Eng Int*. doi:[10.1007/s40092-015-0127-5,12,45-60](https://doi.org/10.1007/s40092-015-0127-5,12,45-60)
- Vinoba V, Indhumathi A (2015) A study on quadratic assignment problem and its applications. *Int J Math Arch* 6(4):5–9
- Vitayasak S, Pongcharoen P, Chris Hicks C (2016) A tool for solving stochastic dynamic facility layout problems with stochastic demand using either a Genetic Algorithm or modified Backtracking Search Algorithm. *J Prod Econ, Int*. doi:[10.1016/j.jjpe.2016.03.019](https://doi.org/10.1016/j.jjpe.2016.03.019)
- Yaman A, Gethin DT, Clarke MJ (1993) An effective sorting method for facility layout construction. *Int J Prod Res* 31(2):413–427
- Zhao Y, Wallace SW (2014) Integrated facility layout design and flow assignment problem under uncertainty. *Inform J Comput* 26(4):798–808
- Zhao Y, Wallace SW (2015) A heuristic for the single-product capacitated facility layout problem with random demand. *EURO J Transp Logist* 4:379–398. doi:[10.1007/s13676-13014-10052-13676](https://doi.org/10.1007/s13676-13014-10052-13676)

