IJE TRANSACTIONS C: Aspects..Vol. 28, No. 12, (December 2015) 1756-1764

International Journal of Engineering

Journal Homepage: www.ije.ir

A Particle Swarm Optimization Approach to Joint Location and Scheduling Decisions in a Flexible Job Shop Environment

H. Daliri[a], H. Mokhtari*[b], I. Nakhai Kamalabadi[c]

[a] *Department of Industrial Engineering, Faculty of Engineering, Tarbiat Modares University, Tehran, Iran*
[b] *Department of Industrial Engineering, Faculty of Engineering, University of Kashan, Kashan, Iran*
[c] *Department of Industrial Engineering, Faculty of Engineering, University of Kurdistan, Kurdistan, Iran*

PAPER INFO

Paper history:
Received 07 August 2015
Received in revised form 25 November 2015
Accepted 24 December 2015

Keywords:
Flexible Job Shop Scheduling
Facility Layout
Transportation Time
Particle Swarm Optimization
Taguchi Design of Experiments

ABSTRACT

In traditional scheduling literature, it is generally assumed that the location of facilities are predetermined and fixed in advance. However, these decisions are interrelated and may impact each other significantly. Therefore finding a schedule and facility location has become an important problem as an extension of the well-known scheduling problems. In this research we consider joint decisions on planning of machines' layout and scheduling of jobs on each machine in a Flexible Job Shop environment. The aim is to minimize maximum completion time. The problem is formulated as a mathematical programming model and is solved using an enhanced particle swarm optimization (PSO). Furthermore, parameters of algorithm is optimized by Taguchi statistical tool. A lower bound is also devised for evaluating the obtained results.

doi: *10.5829/idosi.ije.2015.28.12c.08*

1. INTRODUCTION

In recent years, the increase in productive units' competition, decreasing production costs and increasing the productive systems' efficiency have attracted much attention [1-3]. An important issue that affects the production costs is the facilities' layout and scheduling. Finding the appropriate layout decreases the transportation costs. It also increases production. On the other hand, appropriate scheduling will also affect production rate and system efficiency. Facility layout is a decision-making problem that deals with facility location in a productive unit and comprising arrangement of different sections including career centers and the facilities utilized in the production process. Result of studying layout is a diagram of positioning mentioned cases. In addition, Flexible Job Shop problem is a general case of scheduling Job Shop product problem. In Flexible Job Shop problem, sequence of performing operations for different jobs might be different but this sequence is constant for each job. In addition, for performing each operation there is a set of machines, among which one should be selected. Frequently, these two problems are solved separately or with precedence, while they influence each other. Thus, in this paper, Flexible Job Shop schedule and facility layout are investigated simultaneously considering that it is impossible to neglect the transportation time between machines.

As mentioned, transportation time cannot be neglected, regarding the transportation between machines' jobs, these times affect scheduling. On the other hand, determinant factor for transportation time is the distance between machines, which is related to machines' layout. Therefore, it can be concluded that layout affects scheduling. On the other hand, in most cases the determinant factor in finding facility layout is the material flow among different facilities. In the flexible shop problem, since there are several machines to perform every job, the material flow would change depending on the selection of different machines. Thus, scheduling and selection of machines affect the flexible shop problem scheduling. According to transportation

* Corresponding Author's Email: *mokhtari_ie@kashanu.ac.ir* (H. Mokhtari)

Please cite this article as: H. Daliri, H. Mokhtari, I. Nakhai Kamalabadi, A Particle Swarm Optimization Approach to Joint Location and Scheduling Decisions in a Flexible Job Shop Environment, International Journal of Engineering (IJE), TRANSACTIONS C: Aspects..Vol. 28, No. 12, (December 2015) 1756-1764

time between machines, the relationship between these two problems is a sweeping relation as shown in Figure 1. Thus, these two problems are investigated simultaneously in this paper.

As mentioned, in most researches, layout and scheduling problems are investigated separately. Not many researches have studied these problems simultaneously, so it can be considered as a new topic. However, among the few studies conducted in this field, a work by Ranjbar and Razavi [4] can be mentioned; in which, job layout and scheduling problems have been considered simultaneously. Besides presenting a model, they have also solved the problem through a hybrid algorithm based on scatter-search algorithm. Some other works [5,6] have studied job and layout problem simultaneously. Considering transportation time among machines, they have investigated job layout and scheduling simultaneously. They have taken the problem as a multi-objective problem, and minimizing the maximum production time, total transportation cost were the objectives. They solved the problem with Genetic Algorithm and a hybrid approach with Variable-neighborhood-search Algorithm and obtained the set of dominant results.

The rest of this paper is organized as follows. Problem statement and modeling is presented in section 2. In section 3, solution method will be presented and calculation results will come after in section 4. Finally, section 5 concludes the paper and presents a few suggestions for future works.

2. PROBLEM STATEMENT AND MODELING

According to the mentioned cases, this paper studies the flexible job and layout simultaneously, such that there are specified number of machines and locations for locating the machines; on the other hand, there are a number of jobs to be performed flexibly on machines.

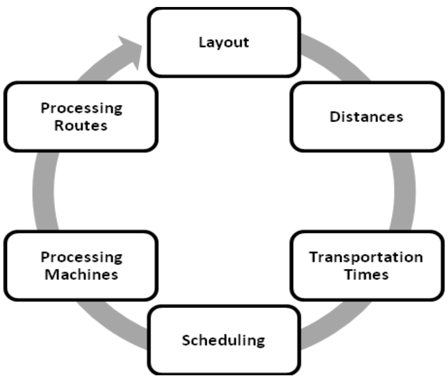

Figure 1. Key factors in interaction between scheduling and layout

Flexibility means that there are several machines to perform each job. Flexible Job Shop problem is analogous to the Job Shop Product problem.

The difference is that in Flexible Job Shop problem unlike the Job Shop Product problem, several machines can perform each job. Thus, this study aims to find the following.

- Finding the machines' location or changing the machines' location
- Selecting the machines for performing jobs
- Defining the order of performing different jobs on machines

In the following, we describe the general assumptions used in modeling the problem. These assumptions are as follows:

- Number of machines and their locations are equal. All machines can sit in all locations.
- Transportation time among machines is not negligible and depends on facilities' distance and facility layout
- Sweeping time among machines is assumed equal and raw material depository location and final product depository location are considered to be fixed.
- At the beginning, all jobs are in the raw material depository. Interrupting the operations is not allowed and a middle depository exists for all machines.
- No machine can perform more than one job in an instant. In addition, there are several machines for some jobs that one should be selected among them.
- Upon the start of a job on a machine, that job will be set on that machine until processing is over. No machine fails during the process.
- Jobs are considered independent of each other and there is no preference among them.
- Material flow is considered the same for all jobs. After each operation, jobs move to the next machine immediately.

In order to check the reciprocity of layout and scheduling in Flexible Job Shop Product, an example with four machines and three jobs are evaluated. Datum of the problem is as follows. Note that for datum related to scheduling, some examples of the work presented in literatures, which are partially flexible have been used. In Table 1, O_{ij} shows ith operation of jth job, p_{ijk} is the time of operation O_{ij}, by kth machine, M_k represents the kth machine. The location of the machines is presented in Figure 2. According to Figure 2 and assuming step transportation and transportation time for two adjacent locations, data regarding transportation time is reported in Table 2. Note that L_v shows the vth location.

According to the described problem, to study the effect of layout on scheduling, several layouts are considered fixed, so optimal scheduling is obtained. Based on these results, impact of layout on scheduling is studied.

TABLE 1. Processing times

Jobs	Operations	M_1	M_2	M_3	M_4
J_1	$O_{1,1}$	2	5	–	1
	$O_{2,1}$	5	4	15	7
	$O_{3,1}$	–	5	–	4
J_2	$O_{1,2}$	2	–	4	7
	$O_{2,2}$	–	2	3	2
	$O_{3,2}$	–	5	4	10
J_3	$O_{1,3}$	1	5	2	4
	$O_{2,3}$	–	1	2	5
	$O_{3,3}$	2	5	4	–
	$O_{4,3}$	4	8	2	1

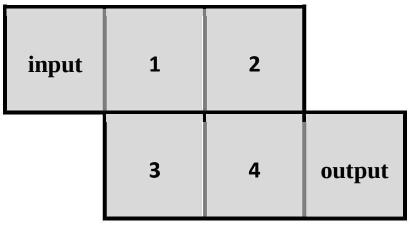

Figure 2. Layout of machines

TABLE 2. Transportation times

	L_1	L_2	L_3	L_4	Input warehouse	Output warehouse
L_1	0	1	1	2	1	3
L_2	1	0	2	1	2	2
L_3	1	2	0	1	2	2
L_4	2	1	1	0	3	1

According to the proposed description, first fixed layout is shown in Figure 3. Note that this layout has been selected randomly.

Taking Figure 3 as the problem's layout, Figure 4 shows the machines performing jobs, sequence of jobs for Flexible Job Shop Product. The Gantt chart and scheduling are also shown in Figure 5.

As shown in Figure 5, completion time of all jobs is equal to 16. In addition, the amount of transportation, which is equal to, total traveled distance for all jobs, is taken as a measure for checking the used layout, which is calculated 14. Then, the problem is studied by changing the layout and the impact of this change on scheduling is evaluated. Considering Figure 6 as the problem's layout, machines performing the jobs and order of performing the jobs would be the directional network of Figure 7.

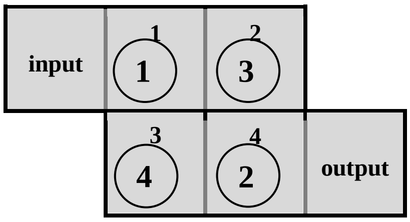

Figure 3. Layout of state A

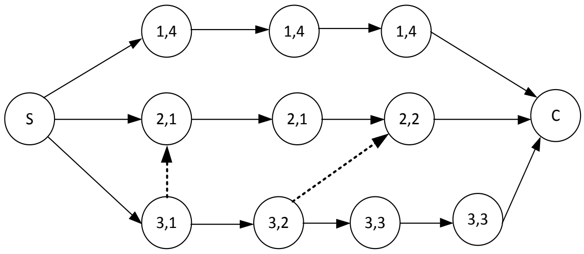

Figure 4. Job sequences in form of disjunctive graph for state A

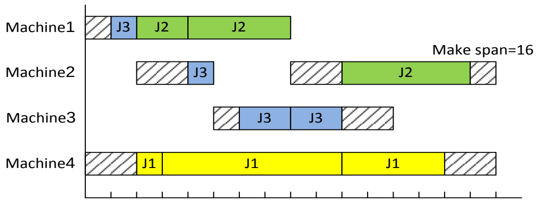

Figure 5. Gantt chart for state A

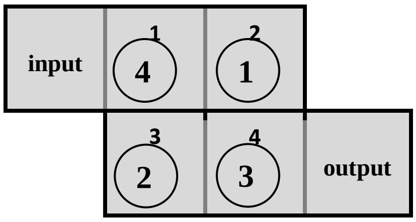

Figure 6. Layout of state B

The Gantt chart of sequence and schedule of jobs is shown in Figure 8.

As shown in Figure 8, completion time of all jobs is equal to 17. The amount of transportation is also, equal to 21. As it comes with the results, layout not only influences the completion time of the job, but also affects the sequence of jobs and the selected machines to perform the job. Interestingly, if schedule obtained in state A is performed on the layout of state B, completion time of jobs will increase to 18. On the other hand, if schedule of state B is performed on the layout of state A, completion time of jobs will increase. Effect of layout on scheduling was studied. As the results obtained for layouts show, changing the schedule and operating machines has a great influence on the machines' layout.

Before formulating the problem, sets, parameters and the used variables are introduced as follows.

Sets:
n: set of machines $\{k, g \in n\}$
L: set of locations $\{l, v \in L\}$
J: set of jobs $\{j, f \in J\}$
I_j: set of operations $\{i, h \in I_j\}$
O_{ij}: ith operation of the jth job
M_{ij}: set of available machines for performing operation O_{ij}
A: set of operations' sequence for different jobs $\{(i,j) \rightarrow (h,j) \in A\}$

Parameters:
T_{lv}: Transportation time between locations l and v.
p_{ijk}: Processing time of operation O_{ij} by k machine.
$p_{ifinal\ j\ k}$: Processing time of last operation by k machine
I_j: Number of jth job's operations

Variables:
C_{ij}: Completion time of operation O_{ij}
C_{max}: Maximum completion time
X_{kl}: Binary variable, it takes 1 if machine k is in place l.
W_{ijt}: Binary variable, it takes 1 if machine t is selected to perform operation O_{ij}.

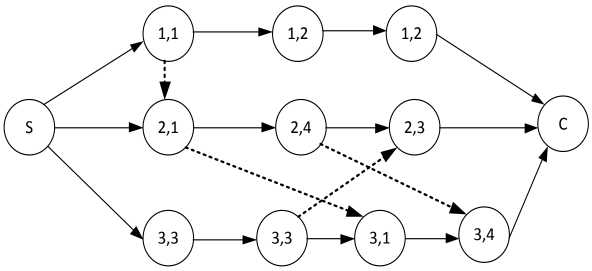

Figure 7. Job sequences in form of disjunctive graph for state B

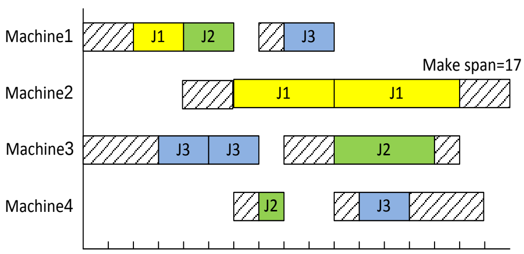

Figure 8. Gantt chart for state A

$$\min \ C_{max} \tag{1}$$

$s.t.$

$$\textstyle\sum_{l=1}^{L} x_{kl} = 1 \quad \forall \ \ k = 1, \dots, n \tag{2}$$

$$\textstyle\sum_{k=1}^{n} x_{kl} = 1 \quad \forall \ \ l = 1, \dots, L \tag{3}$$

$$\textstyle\sum_{k \epsilon m_{ij}} W_{ijk} = 1 \ \ \forall \ \ i, j \tag{4}$$

$$C_{max} \geq C_{final,j} + t_{\infty l} x_{kl} W_{finaljk} \ \ \forall \ j \in J; k, l \tag{5}$$

$$C_{ij} - C_{i-1,j} \geq p_{ijg} W_{ijg} + t_{lv} x_{kl} x_{gv} W_{i-1jk} \ \ \forall \ i = 2, \dots, I_i; \ j \in J; \ l, g, k, v \tag{6}$$

$$C_{1j} \geq p_{1jk} W_{1jk} + t_{0l} x_{kl} \quad \forall \ j \in J; k, l = 1, \dots, n; \tag{7}$$

$$[(C_{hf} - C_{ij} - p_{hfk}) W_{hfk} W_{ijk} \geq 0] \ \vee \ [(C_{ij} - C_{hf} - p_{ijk}) W_{hfk} W_{ijk} \geq 0] \quad \forall (i,j); (h,f) \ ; k \in n \tag{8}$$

$$x_{kl} = \{0,1\} \quad \forall \ \ k, l \in 1, \dots, n \tag{9}$$

$$W_{ijk} = \{0,1\} \ \ \forall \ \ i, j \in 1, \dots, n; t \in m_{ij} \tag{10}$$

$$C_{ij} \in N^{+} \quad \forall \ \ i, j \in 1, \dots, n \tag{11}$$

In this model, minimizing the maximum completion time of jobs (Equation (1)) is selected as the objective function of the problem. Using this objective function, completion time of jobs is minimized; on the other hand, since the problem wants to reduce the completion time of jobs, it seeks to reduce the transportation time as well. Thus, it tries to make the machines closer, with more connection rate. As a result, transportation cost is reduced and a better layout is obtained. Model's constraints act as follows, Equation (2) guarantees that all machines are assigned to locations, and each machine is assigned only to one location. Equation (3) guarantees that all locations will be taken and each location is taken by one machine, only. Equation (4) is used to select only one machine for each operation. Equation (5) guarantees that the completion time of last job must be greater than or equal to completion of each job, as well as the time they reach the final depository. Equation (6) is the constraint that determines sequence of performing operations for different jobs such that completion time of each job is equal to or greater than the completion time of previous operation as well as transportation time between two machines. Equation (7) is considered for completion time of the first operation of each job and guarantees that completion time of the first operation is greater than the operating time of it as well as transportation time from the first depository to the machine's location. Since each machine can perform only one job at a time, when two jobs need one machine, one of the jobs should be preceded. This constraint is expressed in the form of Equation (8). Constraints (9) and (10), show the values of decision variables W_{ijt} and X_{kl} which are either 0 or 1. Constraint (11) also guarantees that C_{ij} is positive. In the proposed model, if value of $t_{lv}x_{kl}x_{gv}W_{i-1jk}$ in Equation (6) is equal to δ, since in most cases it takes 0, the model can be relaxed by considering δ to be 0. Thus, in the following a lower bound is proposed for the model in the form of a lemma.

$$\delta = \begin{cases} t_{lv} & x_{kl}, x_{gv}, W_{i-1jk} = 1 \\ 0 & otherwise \end{cases}$$

Lemma 1: if in constraint (6), δ is zero, then the optimal value of C_{max} is the lower band of the problem.

Proof: assume that δ is 0, thus we have:

$$C_{ij} - C_{i-1,j} \geq p_{ijg}W_{ijg}$$

Now, assuming that delta is positive, the two following cases occur:

$$C_{ij} - C_{i-1,j} \geq p_{ijg}W_{ijg} + \delta$$
$$C_{ij} - C_{i-1,j} \leq p_{ijg}W_{ijg} + \delta$$

If the first case occurs, the constraint is established and there is no problem. But if the second case occurs, for the main problem to be possible, the justified region in $C_{ij} - C_{i-1,j} \geq p_{ijg}W_{ijg} + t_{lv}x_{kl}x_{gv}W_{i-1jk}$ should become smaller about δ. It is clear that by increasing δ, distance of Cs will decrease. Anyway, since the justified region decreases, the optimal value of the obtained result would not be better than the main problem's result. Thus, the obtained result can be used as the lower band of the main problem. Interestingly, by relaxing constraint (6), constraints (2) and (3) are also eliminated from the model. Therefore, the model gets much close to the Job Shop model. Thus, the optimal values obtained for the Job Shop problem can be used as the lower band for the proposed model in this research.

Particle Swarm Optimization Algorithm is an evolutionary computing method based on the results' population. Like Population-base Algorithm, Particle Swarm Optimization Algorithm is a tool that can optimize different types of problems. Eberhart and Kennedy [7] developed this algorithm inspired by the social behavior of a group of birds that are trying to get to an unknown destination [4]. Particle Swarm Optimization Algorithm starts randomly from a group of particles (results) X_{ij}^{t}, and in each step of this algorithm, position of the particles is updated by Equation (12) which is called velocity and the algorithm searches for the optimal result.

$$V_{ij}^{t+1} = w(t+1)V_{ij}^{t} + C_1 r_{1j}^{t}\left(Gbest - X_{ij}^{t}\right) + C_2 r_{2j}^{t}\left(Pbest - X_{ij}^{t}\right)$$

$$X_{ij}^{t+1} = X_{ij}^{t} + V_{\mathrm{ij}}^{\mathrm{t+1}}$$

From Equation (12), velocity vector of each particle is updated according to the particle's velocity in the previous step, in this equation, i shows the particle's number, j shows the cells of each particle. *Pbest* is the best position that the particle has obtained so far and *Gbest* is the best position obtained by all particles, r_{1j}^{t} and r_{2j}^{t} are two random numbers with uniform distribution between (0,1) that are generated independently. Parameters C1 and C2 are the learning coefficients and seek for the impact of *Pbest* and *Gbest* on control process. After updating the particles' velocity, particle's position is updated by Equation (13). Note that $w(t+1)$ is the dynamic inertia weight, which was first proposed by Eberhart and Shi to damp the particles' velocity by increasing the number of iterations and thus increasing the optimal result's convergence accuracy [5]. Its value can be calculated using Equation (14). In the above expression, parameters w_{max} and w_{min} are equal to the initial value and final value of the inertia weight, respectively and i_{max} is equal to the maximum number of iterations in PSO algorithm.

$$w\ (i) = w_{max} - \left(\frac{w_{max} - w_{min}}{i_{max}}\right) i$$

3. 1. Enhanced PSO Algorithm

Different types of PSO have some critical characteristics, which causes the algorithm to fall in the local optimum [6]. One of the bad properties of the PSO algorithm is that if $X_{ij}^{t} = Pbest = Gbest$, and then velocity-updating equation will only depend on V_{ij}^{t}. Therefore, particle's velocity will converge to zero and all particles stop when they reach $Gbest$. Thus, in this study, enhanced PSO (ePSO) algorithm is proposed to solve this problem. In this method, if $Pbest$ changes in any iteration, mutation operator is applied on the particle's position. As a result of this, equation $Pbest - X_{ij}^{t}$ will not become zero, thus particle's velocity will not depend only on V_{ij}^{t}. This operation will not be applied to *Gbest*, because this operation will cause the particles to diverge. In this study, Inversion operator is used. Another property of the PSO algorithm that causes the algorithm to be trapped in local optimization is that if the particles' velocity becomes zero before reaching the optimal point, they will stop. Therefore, if the particles reach the optimal position on time, they can search around the optimal point in time. One of the reasons that causes this event, is that the particles get few information from those particles that have moved well. Thus, an operation crossroad is used in this algorithm for velocity, such that the particle moves based on its velocity after crossroad with probability of $\theta = 0.2$, in addition to its movement based on velocity and better position in considered as the next position of the particle. Crossed velocity is obtained by applying the crossroad operator between velocity of the particle and velocity of the particles that has developed the most improvement. In this study, one-cut point operator has been used.

3. 2. Representation Form The presentation approach used for this problem is that each particle is composed of three layers. Layer 1 includes a sequence of operations, second layer includes the machines that perform the operations and the third layer determines the machines' location. First layer takes integer values smaller or equal to J. It is notable that J shows the number of jobs. In addition, if I_j shows the number of operations of each job, the first layer of each particle has $\sum_{j=1}^{n} I_j$ components, numbers can be repeated, such that numbers 1 show the operations of the first job, numbers 2 show the operations of the second job, and numbers j show the operations of the jth job. Since the numbers show the operations of different jobs, each number should repeat at the number of job's operations. The obtained sequence is a possible sequence. Therefore, the first number one shows the first operation of the job one and this process continues until the job is completed. This process exists for the next jobs.

The second layer of each particle has been considered to determine the machines performing operations and has $\sum_{j=1}^{n} I_j$ components. The results include integers smaller or equal to the number of machines, that shows the machines, which perform the first layer's operation. Since the machines are able to perform different operations, assigned values can be repetitious. On the other hand, since some of the studied examples are partially flexible, in some assignments there is the probability that the problem becomes impossible. Thus, in this study, for the cases that an operation is assigned to the machine that it is not able to perform, a long operation time is considered as a penalty, which increases the objective function's value. Therefore, marginal points of the result's region are searched more.

Note that at the end of the algorithm's run, this part of the results is eliminated. In addition, if as a result of applying velocity, cellular value exceeds the result's region, value of the velocity will not be applied to the intended cell. The third layer of each particle is considered for determining the machines' location and has as many components as the number of machines that can be smaller or equal to the number of locations. Since each location can be assigned only to one machine, components of this layer cannot repeat. This presentation approach is shown in Figure 9. According to the descriptions, sequence of performing operations for example, Figure 9 would be (O_{11}, O_{21}, O_{13}, O_{12}, O_{31}, O_{33}, O_{22}, O_{32}).

Sequence	1	1	2	3	1	2	2	3	3
Machine	1	1	2	2	4	3	3	1	2
Location	1	3	4	2					

Figure 9. Solution representation form

3. 3. Initialization The first step to run the algorithm is to create an initial population. In this step, initial result is generated randomly. However, in some cases regarding the type of problem, to increase the convergence speed, heuristic methods are also used. In this study, extended results are randomly generated. In addition, in order to establish the initial velocities, Equation (15) has been used:

$$V_{\mathrm{ij}}^{0} = V_{min} + (V_{max} - V_{min}) \times r$$

where r is the random value of uniform distribution between 0 and 1. Among the important parameters used in PSO algorithm, v_{max} and v_{min} bound values of velocity vector.

3. 4. Fitness Function, *Pbest*, *Gbest* According to the previous studies, PSO algorithm shows better performance in maximizing problems naturally. Thus, $1/f$ has been used in this study as the fitness function, because the objective function is a minimizing problem [7]. In this study to increase the algorithm's efficiency for calculating velocity in different layers, for the first and second layers, the same fitness values are used to select $Pbest$ and $Gbest$. But in the third layer, $1/\sum \mathrm{t}$ is used as the selection criterion for $Pbest$ and $Gbest$. Now that the process of solving the problem is determined, we present the computational results.

4. COMPUTATIONAL RESULTS

In this study for coding the problem, Matlab 2010 is used. In addition, Lingo 11 is used to solve the problem accurately. This process is done on a PC with Quad-Core Processor and 4MB RAM.

4. 1. Test Problems To study the efficiency of the proposed algorithms, appropriate sample issues are required to be investigated. Since this problem is proposed for the first time, there is no sample issue created specifically for this problem. Thus appropriate sample issue must be created. As the operations' times are determined, transportation times between machines' locations must also be determined. Thus, transportation times are calculated stepwise. Transportation time between two adjacent locations is considered 10 for Dauzere-Peres problem and so for Fattahi's. for

Brendimart's problem this time is considered 1 time unit.

4. 2. Selecting the Parameters of PSO For selecting the parameters of PSO in this problem, Taguchi testing design method is used. Factors and levels of the initial tests are observed in Table 3.

Appropriate perpendicular arrays for algorithms are considered L18 according to their factors and levels. Suitable levels of factors are reported in Table 4.

4. 3. Numerical Results In this section before comparing the results, to check the validity of the algorithm's operation, some smaller sample issues based on sample issues proposed by Fatahi are used. These problems are solved by Lingo 11 and the obtained results are compared together. Problems are considered $(\boldsymbol{n} \times \boldsymbol{m} \times \boldsymbol{h})$, where n is the number of jobs, m is the number of machines and h is the maximum number of operations.

TABLE 3. Related factors and their levels for PSO

Factor	PSO symbols	
	Levels	Type
$\theta_{min} - \theta_{max}$	3	$A(0)$ – (0.4,0.9)
		$A(1)$ – (0.5,0.8)
		$A(2)$ – (0.3,1)
C_1, C_2	6	$B(0)$ – 1.2
		$B(1)$ – 1.3
		$B(2)$ – 1.4
		$B(3)$ – 1.5
		$B(4)$ – 1.7
		$B(5)$ – 2
Population size	3	$C(0) - \frac{2}{3} \times n$
		$C(1) - n$
		$C(2) - 2 \times n$
$V_{min} - V_{max}$	3	$D(0)$ – (−3,3)
		$D(1)$ – (−3.5,3.5)
		$D(2)$ – (−2.5,2.5)

TABLE 4. Best level of parameters for PSO

Factors	PSO symbols
$\theta_{min} - \theta_{max}$	(0.4,0.9)
C_1, C_2	**1**.7
Population size	$\mathbf{2} \times n$
$V_{min} - V_{max}$	(−2.5,2.5)

TABLE 5. Results of algorithms and LINGO

Problem	Lingo11		PSO		ePSO	
	Obj.	Time	Obj.	Time	Obj.	Time
$(2 \times 2 \times 2)$	137	15	137	1	137	1
$(2 \times 2 \times 3)$	350	120	350	2	350	2
$(3 \times 3 \times 3)$	250	210	250	2	250	2
$(3 \times 4 \times 3)$	307	348	307	2	307	2
$(3 \times 5 \times 3)$	461	614	461	3	461	2
$(3 \times 5 \times 4)$	592	1845	592	4	592	4
$(3 \times 6 \times 5)$	564	4341	564	8	564	9
$(3 \times 7 \times 6)$	594	>7200	573	14	564	18
$(4 \times 7 \times 8)$	–	>7200	804	22	804	27

It can be observed from the results in Table 5 that by increasing the problem's size, accurate method is not able to solve the problem, on the other hand, algorithms obtained optimal results in most cases. Therefore, results of the next instances are reported using Tables 6 and 7.

In addition, to make sure, each problem is executed four times. Since the objective functions' scale are not the same, by changing the dimensions of the sample issue, mean time of executing algorithms and the relative deviation are used to normalize the outputs and compare the algorithms as follows.

$$SD = \frac{\sum_{i=1}^{n}(f_i - f^*)}{n.f^*}$$

where, f_i is the results from each execution of the algorithm, f^* indicates the best obtained result and n represents the number of executions. The smaller value of SD, indicates better performance of the algorithm. Checking the obtained results in Tables 5 and 6, it can be observed that algorithms used to solve the problem have obtained the best result at least in one execution. Comparing the results with lower bound values, good performance of the algorithms can be proved. While, two criterions, time and relative deviation, are studied to check the supremacy of the algorithms. Thus, diagram of relative deviation and execution time of algorithms are reported. According to Figures 11 and 13, it is observed that for problems with smaller size, relative deviation of different algorithms are close, while continuing the process, it is observed that in most cases ePSO algorithm is separated from PSO algorithm and has shown better performance.

TABLE 6. Results of algorithms and lower bound on Brandimarte's problems

Test Problems		LB	PSO			ePSO			DLB
Instance	$(n \times m \times h)$		**Obj.**	**SD**	**Time**	**Obj.**	**SD**	**Time**	
LMK01	$(7 \times 6 \times 10)$	45	53	0.010	2.5	53	0.011	3.9	0.170
LMK02	$(7 \times 6 \times 10)$	32	37	0.027	5.4	37	0.021	6.1	0.150
LMK03	$(10 \times 8 \times 15)$	402	423	0.001	14.5	423	0.003	28.2	0.052
LMK04	$(10 \times 8 \times 15)$	68	82	0.012	11.3	82	0.001	10.5	0.205
LMK05	$(10 \times 4 \times 15)$	174	195	0.015	22.4	197	0.004	38.4	0.120
LMK06	$(15 \times 15 \times 10)$	59	79	0.021	72.4	76	0.013	98.9	0.160
LMK07	$(5 \times 5 \times 20)$	148	172	0.021	29.9	172	0.023	45.3	0.160
LMK08	$(15 \times 10 \times 20)$	523	564	0.051	120.3	564	0.033	193.5	0.078
LMK09	$(15 \times 10 \times 20)$	307	332	0.033	112.8	332	0.027	148.6	0.081
LMK10	$(15 \times 15 \times 20)$	214	252	0.071	166.2	248	0.054	288.3	0.170

TABLE 7. Results of algorithms and lower bound on Dauzere's problems

Test Problems		LB	PSO			ePSO			DLB
Instance	$(n \times m \times h)$		**Obj.**	**SD**	**Time**	**Obj.**	**SD**	**Time**	
01aL	$(25 \times 5 \times 10)$	2518	2876	0.012	120.3	2844	0.000	132.3	0.14
02aL	$(25 \times 5 \times 10)$	2231	2510	0.000	132.8	2510	0.000	140.5	0.12
03aL	$(25 \times 5 \times 10)$	2229	2456	0.022	106.5	2456	0.012	121.2	0.10
04aL	$(25 \times 5 \times 10)$	2503	2785	0.020	98.2	2770	0.000	106.1	0.11
05aL	$(25 \times 5 \times 10)$	2216	2523	0.014	141.2	2523	0.010	148.9	0.13
06aL	$(25 \times 5 \times 10)$	2196	2456	0.023	99.1	2456	0.022	103.4	0.11
07aL	$(25 \times 8 \times 15)$	2283	2546	0.035	287.9	2546	0.010	321.1	0.11
08aL	$(25 \times 8 \times 15)$	2069	2340	0.030	256.0	2340	0.021	294.5	0.13
09aL	$(25 \times 8 \times 15)$	2066	2364	0.029	271.5	2383	0.041	241.5	0.14
10aL	$(25 \times 8 \times 15)$	2291	2593	0.052	289.2	2577	0.037	325.4	0.13
11aL	$(25 \times 8 \times 15)$	2063	2432	0.023	302.4	2432	0.022	344.4	0.17
12aL	$(25 \times 8 \times 15)$	2030	2421	0.033	311.3	2421	0.021	338.5	0.19
13aL	$(25 \times 10 \times 20)$	2257	2612	0.063	412.3	2612	0.052	443.1	0.15
14aL	$(25 \times 10 \times 20)$	2167	2452	0.072	499.3	2421	0.055	539.9	0.13
15aL	$(25 \times 10 \times 20)$	2165	2415	0.053	512.4	2389	0.050	551.1	0.11
16aL	$(25 \times 10 \times 20)$	2255	2741	0.042	410.4	2741	0.035	462.3	0.21
17aL	$(25 \times 10 \times 20)$	2140	2399	0.064	552.4	2358	0.041	601.5	0.12
18aL	$(25 \times 10 \times 20)$	2127	2409	0.052	533.7	2409	0.043	603.4	0.13

5. CONCLUSIONS

Scheduling and layout problems have always been the focus of major attentions in industry. Finding the optimal layout and scheduling plays an important role in increasing the efficiency and productivity. Nowadays, due to extensive competition among different organizations, reducing costs and increasing productivity is an important issue for survival and profitability of the organizations. In this study, by considering the transportation time between machines, facilities' layout and scheduling for Job Shop Problems are investigated simultaneously. Criterion of the objective function is minimizing the maximum completion time of jobs, which is calculated in the form of layout and scheduling. Due to difficulty of the proposed models, heuristic PSO methods and an enhanced method based on PSO were used. Taguchi testing design method optimized parameters of the used algorithms. Then, computational results from different methods were reported and compared with each other; ePSO showed better performance in terms of quality and PSO showed a better performance in terms of execution time. Solving the proposed model in this study with other algorithms and studying scheduling and layout simultaneously with other scenarios could be considered for future works.

6. REFERENCES

1. Mokhtari, H. and Abadi, I.N.K., "Scheduling with an outsourcing option on both manufacturer and subcontractors", ***Computers & Operations Research***, Vol. 40, No. 5, (2013), 1234-1242.
2. Dauzere-Peres, S. and Paulli, J., "An integrated approach for modeling and solving the general multiprocessor job-shop scheduling problem using tabu search", ***Annals of Operations Research***, Vol. 70, (1997), 281-306.
3. Mokhtari, H., Mozdgir, A. and Kamal Abadi, I.N., "A reliability/availability approach to joint production and maintenance scheduling with multiple preventive maintenance services", ***International Journal of Production Research***, Vol. 50, No. 20, (2012), 5906-5925.
4. Ranjbar, M. and Razavi, M.N., "A hybrid metaheuristic for concurrent layout and scheduling problem in a job shop environment", ***The International Journal of Advanced Manufacturing Technology***, Vol. 62, No. 9-12, (2012), 1249-1260.
5. Ripon, K.S.N., Glette, K., Hovin, M. and Torresen, J., Job shop scheduling with transportation delays and layout planning in manufacturing systems: A multi-objective evolutionary approach, in Autonomous and intelligent systems. Springer (2012),. 209-219.
6. Ripon, K.S.N. and Torresen, J., "Integrated job shop scheduling and layout planning: A hybrid evolutionary method for optimizing multiple objectives", ***Evolving Systems***, Vol. 5, No. 2, (2014), 121-132.
7. Eberhart, R.C. and Kennedy, J., "A new optimizer using particle swarm theory", in Proceedings of the sixth international symposium on micro machine and human science, New York, NY. Vol. 1, (1995), 39-43.

A Particle Swarm Optimization Approach to Joint Location and Scheduling Decisions in a Flexible Job Shop Environment

H. Daliri[a], H. Mokhtari[b], I. Nakhai Kamalabadi[c]

[a] Department of Industrial Engineering, Faculty of Engineering, Tarbiat Modares University, Tehran, Iran
[b] Department of Industrial Engineering, Faculty of Engineering, University of Kashan, Kashan, Iran
[c] Department of Industrial Engineering, Faculty of Engineering, University of Kurdistan, Kurdistan, Iran

PAPER INFO

Paper history:
Received 07 August 2015
Received in revised form 25 November 2015
Accepted 24 December 2015

Keywords:
Flexible Job Shop Scheduling
Facility Layout
Transportation Time
Particle Swarm Optimization
Taguchi Design of Experiments

چکیده

در ادبیات سنتی زمانبندی، عموماً فرض می‌شود که مکان تسهیلات (ماشین‌آلات) از پیش مشخص و ثابت هستند. درحالیکه این تصمیمات مرتبط با هم بوده و بر هم تاثیر متقابل موثری دارند. بنابراین یافتن یک زمانبندی و جایابی بصورت همزمان برای تسهیلات به عنوان توسعه‌ای بر مسایل زمانبندی متداول جزء یکی از مسایل مهم شهرت یافته است. در این پژوهش ما یافتن چیدمان و زمانبندی بهینه بصورت همزمان در یک سیستم زمانبندی کارگاهی منعطف را در نظر گرفته‌ایم. هدف از مساله کمینه‌سازی حداکثر زمان تکمیل است. مساله بصورت یک مدل برنامه‌ریزی ریاضی مدلسازی می‌شود و با استفاده از روش بهبود یافته‌ی بهینه سازی گروه ذرات حل می‌شود. همچنین برای یافتن پارامترهای الگوریتم از طراحی آزمایش‌های تاگوچی استفاده شده است. همچنین به منظور بررسی عملکرد یک کران پایین ریاضی نیز توسعه داده شده و مورد استفاده قرار گرفته است.

doi:*10.5829/idosi.ije.2015.28.12c.08*