Australian Journal of Basic and Applied Sciences, 6(5): 135-145, 2012
ISSN 1991-8178

An Extended Genetic Algorithm for Group Scheduling Cellular Manufacturing System with Regard to Effect of Learning

Vahid Majazi Dalfard

Young Researchers Club, Kerman Branch, Islamic Azad University, Kerman, Iran

Abstract: Group scheduling issue includes two stages of scheduling. At first, sequence of parts in each family-part is specified and then sequence of family-parts is determined. In this article, group scheduling issue was studied with flow shop rate in order to minimize the longest time of the work completion. In traditional group scheduling, work processing time is constant value and independent of work sequence. This hypothesis is not suitable at most times because ability and skill of worker increased by repetition of work, ability and skill of the worker and as result, work processing time deceases. This phenomenon is recognized as learning effect. In this article, a position-based learning model was used in cellular manufacturing system in which processing time of each family –part depends on sequence of that part entrance. Group scheduling problem was modeled with regard to position-based learning effect and by assuming family-part sequence dependent setup time. Two different genetic algorithms and a heuristic method for solving this problem were developed and assessed by experimental issues. Numerical results obtained from assessment of the proposed solutions indicate that the second genetic algorithm is better than other proposed methods in terms of answer quality.

Key words: *Cellular manufacturing system; Group scheduling; Learning effect; Position-based learning effect; Sequence-dependent setup time.*

INTRODUCTION

Cellular manufacturing system is a manufacturing system in which parts similar in terms of form, design or manufacturing method are clustered in a group and manufactured by a group of machinery in a cell. Success of cellular manufacturing system is subject to correct division of cells and accurate planning of work. Therefore, group scheduling has special importance and position in this system. In group scheduling, sequence of parts (works) of each group (family, part) was identified and sequence of groups (family, parts) is specified for entering the cells. In traditional group scheduling problems, parts processing time (works) was assumed as constant and dependent values of sequence of the parts entrance to the cell. This hypothesis was regarded inefficient and ignored because repetition of work increases ability and skill of the worker and decreases work processing time. This phenomenon is recognized as learning effect in review of literature. Learning effect led to decrease of work processing time and will have direct effect on determination of work scheduling sequence. In this article, group scheduling problem was studied with regard to learning effect and setup time dependent on sequence in cellular manufacturing system with flow shop structure of some machines. In this article, Dejong learning model was included and modeled in this group scheduling with flow shop structure of some machines and then the model was solved with use of genetic method and a heuristic method.

At first, scheduling problems were reviewed with regard to learning effect in section 2. In section 3, suitable learning model was selected and a mathematical model was presented for this problem. In section 4, model solutions were developed. In section 5, the sample problems were designed and efficiency of the proposed solutions was assessed. At the end of section 6, final results were mentioned and field of the future research was specified.

Review of literature:

Learning plays important role in manufacturing environments and effects of learning were proved by experimental studies. Learning philosophy is that time and effort spent for completion of an iterative operation decrease with increase of the number of iterations. On the other hand, increase of production rate increases knowledge and skill of operators and work processing time is reduced. Bicker in 1974 entered two hypotheses in scheduling problem and these two hypotheses are found in most of scheduling problems as follows:

- Work processing time is specified in advance.
- Work processing time is independent of sequence of work performance.

In recent years, these two hypotheses were inefficient and ignored due to nonconformity of experimental planning to scheduling theory. In fact, many researchers concluded after study of experimental scheduling and scheduling theory that work processing times are compressible. A different approach for decreasing processing

Corresponding Author: Vahid Majazi Dalfard, Young Researchers Club, Kerman Branch, Islamic Azad University, Kerman, Iran
E-mail: Vmd_60@yahoo.co.nz

time is use of learning concept. In fact, scheduling problem becomes more realistic wit learning effect. Learning effect is important when production environment changes. These changes can include some cases such as entrance of new personnel, machinery with new equipment, change of working process or new production. In literature of the learning which is created due to iteration of similar operations, automatic learning and learning which is created due to performance of managerial activities such as education, change of production process etc is called inductive learning. In scheduling problems, two different approaches were used for automatic learning as follows:

- Position based approach: in this approach, learning is affected by the number of work which was processed so far. This hypothesis is realistic hypothesis in case practical processing of work is machine-based and there is no human interference.
- Total processing time approach: this approach considers time of processing for work which was processed so far. Therefore, this approach applies worker's experience approach which was obtained from production and processing of work.

Biskup was the first person who studied effect of learning in scheduling problems. He used position based traditional exponential learning model ($P_r, P_r = P \times r^a$, position work processing time rth , r work position, a learning index), in single machine scheduling problem. Work processing time in exponential model depends on position of that work. He showed that this problem can be solved with shortest time rule in order to minimize total work completion time. Moshif showed that some traditional known solutions are not valid for single machine scheduling problem by assuming exponential learning model. He proved that single machine scheduling problem is solved with the shortest time rule in order to minimize the longest work completion and with regard to exponential learning model. This problem can be solved in order to minimize common due date as an allocation problem. Moshif showed in another research that parallel machines problem can be solved in order to minimize total time of work completion and with regard to exponential learning model as an allocation problem. Lee et al studied single machine scheduling problem with regard to traditional learning model and developed a branch and bound algorithm to solve it in order to minimize total time of work completion and maximum tardiness. Lee and Woo entered exponential learning in two machine flow shop problem and assumed that learning is applied separately on each machine. They studied this problem in order to minimize total completion time and presented a branch and bound algorithm. Bachman and Janiac showed that single machine scheduling problem is solved in order to minimize work balanced completion time with the shortest processing time rule. They also showed that this problem is solved in spite of equal processing time for all works with non-ascending ordering. They used traditional exponential learning model in both problems. Javeau et al used exponential learning model in their research. They showed that single machine scheduling problem can be solved with the shortest processing time rule. Hypothesis of consistent weights in this problem indicates that the work which has shortest processing time has more weight. They also proved that single machine problem can be solved with the earliest due date in order to minimize the maximum tardiness and assume consistent due date. Consistent due date in this problem indicates that the work which has shorter processing time has earlier due date. They focused on flow shop problems of two machines and showed that this problem can be solved with the shortest processing time rule on the first machine in order to minimize total work completion time and assume equal processing time for all works on the second machine and to minimize longest work completion time. Chen at al used exponential learning model developed by Lee and Woo. They studied flow shop scheduling problem of two machines in order to minimize total completion time and the maximum tardiness and presented a branch and bound algorithm for solving it. Kolamas and Kiparisis studied problem of flow shop of two machines with exponential learning model developed by Lee and Woo with two different hypotheses. In the first hypothesis, work processing time on the first machine was shorter than their processing time on the second machine. In the second hypothesis, all work processing time on the second machine is weight of work processing time on the first machine. They showed that this problem can be solved with the shortest processing time rule in order to minimize total completion time and with the maximum completion time. Wang and Zia showed that known rule of Johnson doesn't present optimal scheduling for flow shop problem of two machines and with regard to exponential learning model and in order to minimize the longest completion time. Iren and Goner entered traditional exponential learning model in single machine scheduling problem and solved this problem in order to minimize total tardiness with use of metaheuristic algorithm. Ku and Yang combined sequence dependent setup time with effect of position based learning in single machine scheduling problem. They showed that this problem can be solved with the shortest processing time rule in order to minimize total completion time and minimize the maximum completion time. Review of literature shows that learning effect was studied only in single machine scheduling problems or flow shop of two machines. In most of these problems, traditional and developed exponential learning model was used. In this model, work processing time decreases by iterating performance of a work and such decrease is affected by works position. The major problem of exponential model is that machine operation time is not separated from manual operation and learning is effective on total processing time.

Proposed model:

This article studies group scheduling in cellular manufacturing system in shop flow state. One of the gaps of group scheduling problems in cellular manufacturing system is failure to consider learning effect and we can get closer to real conditions. In this section, integer 0 and 1 was used for modeling scheduling problem. In this model, sequence of parts of each family-part was specified in the first stage and then family of the parts is scheduled by assuming fixed sequence of families' parts and specifying real processing time of each part on each machine. Here, work completion time minimization target function was used in this problem. Minimization of this target led to increase of output rate and speed of manufacturing process and finally caused the product to reach the customer in no time. Therefore, in the first stage, scheduling i.e. determination of optimal sequence of parts in each family-part was introduced as target function and maximum completion time was minimized as target function.

3-1- learning model:

In order to use learning effect in scheduling problems, two approaches of position based learning effect and learning effect are used on the basis of total processing time. Position based approach is applicable when work processing is machine oriented. With regard to the fact that manufacturing system is cellular in which machinery plays important role in processing of parts and family of parts so that we can say that most cells are machine oriented , in this research , position based learning effect was sued. By comparing the available position based learning models, Dejong model was selected among the available models due to two important characteristics for group scheduling in cellular manufacturing system as suitable model.

- Separation of manual and machine operations from each other and application of learning effect only on manual work operations
- Easy estimation of model parameters

In this model, there is a parameter called incompressible factor leading to separation of machine operation from manual operation. As result, machine processing time remains constant and only manual operation processing time decreases. General state of Dejong model is as relation 1:

$$P_r = P_0.(M + (1 - M).r^{\alpha} \quad (1)$$

P_r : Time necessary for performing operations in rth position

P_0 : Time necessary for performing operations in normal state (without consideration of work positon)

r : work performance position

α : learning index or learning effect (learning rate logarithm in base 2 which is negative number)

M : Incompressible factor

With increase of the number of operations, time for performing manual operations in model was reduced to 0 and time for performing total operations will have tendency to machine processing time.

$$P_{ifrj} = P_{ifj}.(M_{ifj} + (1 - M_{ifj}).r^{\alpha_f} \quad (2)$$

With regard to similarity in manufacturing process or design of parts of a family-part, we can assume unique for each family-part. In this model, a unique learning index (α_f) was considered for each family-part. Incompressible factor (M) for each part of the family-part was considered to be unique on each one of the machinery.

3-2- Model hypotheses:

in this model, the following hypotheses were considered with regard to applied specifications and conditions in cellular manufacturing system:

1- Workgroups and manufacturing cells are pre-specified.
2- Manufactured machinery is accessible and there is failure of operations flow.
3- In case of similar parts, such parts should be placed uniquely in family of the part.
4- Setup time on each machine for each family-part depends on sequence of family's entrance in cell.
5- Anticipatory setup necessary for each family on the machine is independent of the applied parts.
6- Setup time relating to parts of each family (partial setup time) was considered at their production times and was not considered independently.
7- Normal processing time for each part on each machine is defined and partial setup time and machines transportation time were considered in it.
8- Processing of each family of the part can be done only in a cell and there is no intercellular motion.
9- In order to produce each part, a set of operations is necessary each performed by a kind of machine in cell.

10- Each family of the part can have a unique learning index and decrease of parts processing time follows Dejong learning function.

11- In order to decrease setup time, when operations relating to family of the parts start with a machine, operation of another family doesn't start until all operations of the parts are performed by the related machine.

12- The produced cells have flow shop stricture and order of passage of parts through machines in each family is equal.

3-3- proposed mathematical model:

Parameters and symbols used in problem modeling are as follows:

$f = 1,\ldots,PF$ -index relating to family-part number

$i = 1,\ldots,n_f$ -index relating to part number

j= index relating to machine type

$r = 1,\ldots,n_f$ -index relating to position of parts processing position

$K = 1,\ldots,PF$ -index relating to position of family processing

α_f =effect of learning of family-part f

M_{ifj} =ratio of machine operations of part i to family-part f on machine j

S_{gfj} =setup time for family-part f which is processed immediately after family-part g on machine j $S_{gfj} \neq S_{fgj}$

P_{ifj} =normal processing times for part i from family-part f on machine j.

P_{ifrj} =practical processing times for part i from family-part f in position f on machine j

AP_{frj} =practical processing time for part of rth position from family-part f on machine j

C_{rfj} =completion time of a part from family-part which is processed on position r on machine j.

C_{max} =maximum completion time of parts from a family-part

CF_{max} =maximum completion time of parts from a family-part

$CF_{k,r,j}$ =completion time of rth position part from the family which is processed in kth position on jth machine.

The following decision variables were used for modeling in new problem which was considered as variable 0 and 1:

$$Y_{fg}\begin{cases}1, & \text{If family - part f is processed in kth position}\\ 0 & \text{otherwise}\end{cases}$$

$$X_{ifr}\begin{cases}1, & \text{If part I from part i of family - part r is processed in rth position}\\ 0 & \text{otherwise}\end{cases}$$

$$Z_{gf}\begin{cases}1, & \text{If family - part f is processed after family - part g}\\ 0 & \text{otherwise}\end{cases}$$

3-3-1- first stage of group scheduling proposed model

$$Minz \quad z = C_{max} \tag{3}$$

$$C_{rfj} \geq \sum_{i=1}^{n_f} X_{ifr} P_{ifrj} \qquad \text{for } i = 1, j = 1 \tag{4}$$

$$C_{rfj} - C_{r\text{-}1fj} \geq \sum_{i=1}^{n_f} X_{ifr} P_{ifrj} \qquad \text{for } r = 2,\ldots,n_f \quad , j = 1,\ldots,m \tag{5}$$

$$C_{rfj} - C_{rfj\text{-}1} \geq \sum_{i=1}^{n_f} X_{ifr} P_{ifrj} \qquad \text{for } j = 2,\ldots,m \quad , r = 1,\ldots,n_f \tag{6}$$

$$\sum_{r=1}^{n_f} X_{ifr} = 1 \quad \text{for} \quad f = 2,...,PF \;, i = 1,...,n_f \tag{7}$$

$$\sum_{i=1}^{n_f} X_{ifr} = 1 \quad \text{for} \quad f = 2,...,PF \;, i = 1,...,n_f \tag{8}$$

$$C_{max} \geq C_{rfj} \quad \text{for} \quad f = 1,...,PF \;, r = 1,...,n_f, j = 1,...,m \tag{9}$$

$$P_{ifrj} = P_{ifrj}(M_{ifj} + (1 - M_{ifj}).r^{\alpha_f} \quad \text{for} \quad f = 1,...,PF \;, r = 1,...,n_f, j = 1,...,m \;, \alpha_f \;, M_{ifj} \tag{10}$$

Relation 3 shows the first phase target function of model which minimizes the maximum completion time of parts in each family-part. Relation 4 determines a part which should be processed in the first position. Relation 5 ensures that completion time of the present position part is longer than that of the previous part on the same machine in a family. Relation 6 ensures that completion time of a part on machine is longer than that of the same part on the previous machine in a family. Relation 7 shows that any part i from family-part only can be processed on a position in cell. Relation 8 shows that only one part can be processed in each position. Relation 9 shows that the maximum time of family-part f equals to the maximum completion time of position r part on each one of the machines. Relation 10 shows practical processing time of each part from a family-part with regard to normal processing time of that part, family-part learning index, part and incompressibility factor of each part in family –part on each machines.

3-3-2- the second phase of group scheduling proposed model:

$$Min z \quad z = CF_{max} \tag{11}$$

$$CF_{1,1,j} \geq \sum_{f=1}^{PF} Y_{f,1}(AP_{f,1,j} + S_{ffj}) \quad \text{for} \quad j = 1,...,m \tag{12}$$

$$CF_{1,r,j} \geq CF_{1,r-1,j} + \sum_{f=1}^{PF} Y_{f,1}.AP_{f,r,j} \quad \text{for } r = 2,...,n_f \;, j = 1,...,m \tag{13}$$

$$CF_{1,r,j} \geq CF_{1,r-1,j} + \sum_{f=1}^{PF} Y_{f,1}.AP_{f,r,j} \quad \text{for } r = 1,...,n_f \;, j = 2,...,m \tag{14}$$

$$CF_{k,1,j} \geq CF_{k-1,n_f,j} + \sum_{f=1}^{PF} Y_{f,k}.(AP_{f,1,j}) + \sum_{g=1}^{PF} Z_{gf}.(S_{g,f,j}) \quad \text{for } k = 2,...,PF, j = 1,...,m \tag{15}$$

$$CF_{k,r,j} \geq CF_{k,r,j-1} + \sum_{f=1}^{PF} Y_{f,k}.AP_{f,r,j} \quad \text{for } k = 2,...,PF, r = 1,....n_f \;, j = 1,...,m \tag{16}$$

$$CF_{k,r,j} \geq CF_{k,r-1,j} + \sum_{f=1}^{PF} Y_{f,k}.AP_{f,r,j} \quad \text{for } k = 2,...,PF, r = 2,....n_f \;, j = 1,...,m \tag{17}$$

$$Z_{gf} \geq Y_{g,k-1} + Y_{f,k} - 1 \quad \text{for } f, g = 2,...,PF, g \neq f \tag{18}$$

$$\sum_{k=1}^{PF} Y_{f,k} = 1 \quad \text{for } f = 1,...,PF \tag{19}$$

$$\sum_{f=1}^{PF} Y_{f,k} = 1 \quad \text{for } k = 1,...,PF \tag{20}$$

$$CF_{max} \geq CF_{k,n_f}, m \quad \text{for } k = 1,...,PF \tag{21}$$

Relation 11 shows target function in the second phase of the model which minimizes maximum completion time of family –part. Relation 12 specifies the first family –part for processing in cell. Relations 14 and 13 prevents from time interference of parts from family-part which is processes in the first position so that processing of this part on the previous machine ($j-1$) and processing of the previous part ($r-1$) on machine (j) should be completed in order to process part of rth position on machine j. Relation 15 specifies sequence of the rest of family so that completion time of family-part which enters the cell later is longer than completion time of the family-part which enters cell earlier. In this time limitation, setup time is considered on the basis of entry of the family to the cell. Relations 14 and 13 prevent time interference of parts of each family-part. Relation 18 ensures that setup time enters relation 15 on the basis of sequence of the families. This limitation shows that if family - part g is processed in position $k-1$ and family-part f in position k, family–part f will be processed after family-part g. relation 19 shows that each family-part can only be processed in a position. Relation 20 mentions that a family-part should be processed in each position. Relation 21 shows that the maximum completion time of families equals to the maximum completion time of each family-part.

Proposed mathematical solution:

Group scheduling problem in cellular manufacturing system is of NP-Hard type with regard to setup time dependent on sequence. With regard to the fact that metaheuristic methods have high ability to find good answers, genetic metaheuristic methods were used for solving the problem. These methods act in such a manner that they don't stop in local optimal answer and try to exit local optimal point with use of different strategies while heuristic method gives better answer with higher and more economical efficiency. In this article, a heuristic method and two metaheuristic methods were given for solving the problem.

4-1- heuristic method:

Heuristic methods specify sequence of parts in each family–part and sequence of family's entry to cell. Schaller et al studied metaheuristic methods for group scheduling in cellular manufacturing system with linear flow structure and presented 12 heuristic methods with regard to families sequence dependent setup and in order to minimize the maximum completion time. Results showed that the best heuristic method is combination of CDS (Compbell-Dudek-Smith) (for determining sequence of parts in each family) and modified NEH method (for determining sequence of families to cells). With regard to the fact that applied hypothesis of learning effect was not considered in the first part of heuristic method i.e. CDS in order to determine sequence of parts in each family, therefore, CDS algorithm was revised and corrected and finally CDS algorithm was presented to determine sequence of parts of each family and modified NEH algorithm was used to determine sequence of families in the second part of heuristic method. The above heuristic method was named MCMN.

4-1-1- modified CDS algorithm:

Stages of modified CDS heuristic algorithm are as follows:

Step 1: calculate manual operation processing time for each part i from family f .

$$\mathrm{O_{if}} = \sum_{j=1}^{m} \mathrm{P_{ifj}}.(1 - M_{\mathrm{ifj}}) \quad for \quad i = 1,\dots,n \tag{22}$$

Step 2: order part of each family on the basis of non-descending order of total manual operation processing time and name this sequence SE_0 .

Step 3: calculate the maximum completion time in sequence of parts in SE_0.

Step4: put optimal sequence equal to sequence of SE_0 ($SE^* = SE_0$) and put the maximum optimal completion time against the maximum completion time of ($C_{\max}$ (SE^*) $= C_{\max}$ (SE_0)) SE_0 and $\mathrm{K} = 1$.

Step 5: for sequence of SE^* for each part i from family f and with regard to its position r in SE^*, calculate value of a_i and b_i .

$$\mathrm{a_i} = \sum_{j=1}^{m} \mathrm{P_{ifj}}.(M_{\mathrm{ifj}} + (1 - M_{\mathrm{ifj}}))\,.\mathrm{r}^{\alpha_{\mathrm{f}}} \tag{22}$$

$$\mathrm{b_i} = \sum_{j=m-k+1}^{m} \mathrm{P_{ifj}}.(M_{\mathrm{ifj}} + (1 - M_{\mathrm{ifj}}))\,.\mathrm{r}^{\alpha_{\mathrm{f}}} \tag{22}$$

Step 6: for parts of which $a_i \le b_i$, order the parts on the basis of non-descending order of a_i s and call that set u and for the parts of which $a_i > b_i$, order the parts on the basis of non+-ascending order and call that set v. new sequence of SE_t will be as $SE_t = uv$.

Step 7: for new sequence of SE_t , calculate values of a_i and b_i with use of relations 23 and 24 and form new sequence. Repeat this action until another new sequence is not created.

Step 8: calculate the maximum completion time for SE_t s.

Step 9: if Min ($C_{\max}$ (SE_t)) $\le C_{\max}$ (SE^*), then put $SE^* = SE_t$.

Step 10: put $k = k + 1$. If $k < m$, go to the fifth step. Otherwise, accept SE^* as optimal sequence and C_{max} (SE^*) as optimal completion time.

Step 11: If $f < PF$, then, $f = f + 1$ and go to step 1, otherwise, stop.

4-1-2- modified NEH algorithm:

Stages of this algorithm are as follows:

Step 1: calculate the average setup time ($\overline{S}_{fj}$) for each family-part on each machine. Then calculate effective processing time.

$$\overline{S}_{fj} = \frac{\sum_{g=1}^{PF} S_{gfj}}{PF} \qquad (25)$$

$$E_{fj} = \overline{S}_{fj} + \sum_{r=1}^{n_f} P_{rfj} \qquad (25)$$

Step 2: form the primary scheduling $SE_0 = (\mu(1), \ldots, \mu(PF))$ so that for $r \le (PF-1)$, $\sum_{j=1}^{m} E_{\mu(r),j} \ge \sum_{j=1}^{m} E_{\mu(r+1),j}$.

Step 3: put $r = 2$ and $\sigma = (\mu(1), \mu(2))$. For $0 \le q \le r$, $\sigma = \sigma_q \sigma_{r-q}$.

Step 4: obtain $\mu(r+1)$ from SE_0. Produce partial sequences $(r+1)$ with $\sigma_q \mu(r+1) \sigma_{r-q}$ while $0 \le q \le r$ and form set ω. Calculate the maximum completion time for each one of the set members and specify the minimum time. Then put $\sigma = \omega_z$.

$$\omega = \{\omega_1, ..., \omega_{r+1}\} = \{\sigma_{r-q};\ \ 0 \le q \le r\} \qquad (27)$$

$$C_{max}(\omega_z) = \min_{1 \le v \le r+1} \{C_{max}(\omega_v)\} \qquad (28)$$

Step 5: if $r < PF$, put $r = r + 1$ and go to step 4. Otherwise, accept scheduling σ with the maximum completion time C_{max} (σ).

4-2- genetic metaheuristic algorithm:

In this research, we studied two different algorithms on the basis of genetic method for solving the problem. In both algorithms, chromosomes were defined in such a manner that it mentions form of the expected answer. In the first phase of group scheduling, chromosomes show priority of parts in each family-part and in the second phase, it shows priority of family of the parts for entering the cell. Length of chromosome equals to the number of parts in each family-part and in the second phase, it equals to the number of family-part.

4-2-1- first genetic algorithm:

In both algorithm, primary population was created randomly and answers fitness was calculated. With regard to the fact that scheduling problem is of minimization type and goal of the problem is to minimize the maximum work completion time (C_{max}). Therefore, fitness of the obtained answers is reverse of target function introduced in the mathematical model and for this reason; chromosomes which have the highest rate of fitness are identified as the best chromosome. With regard to selected value for elite variable, some chromosomes of which fitness rate is higher is directly transferred to the next generation and suitable answers select the best chromosomes of the available generation with regard to keep rate on the basis of keep rate multiplied by primary population and other unsuitable chromosomes exit from generation . With use of weight roulette wheel method, we select the selected chromosomes for reproduction and reproduction pool is completed by reproducing children from the selected chromosomes with use of LOX crossover operator so that its population equals to crossover rate multiplied by the primary population. In case two equal parents are selected for reproduction of child, the second parent is excluded and the second parent is selected among the kept chromosomes. With regard to the presented mutation rate, we select the chromosomes randomly from reproduction pool and replacement mutation operator is applied on the selected chromosomes and they replace the previous chromosomes. At the end, chromosomes available in reproduction pool are transferred to the future generation with elite chromosome of the current generation and in case of failure to meet stop condition; all phases will be repeated with new generation. Stop condition in both genetic algorithms is the number of generations and 50 generations were considered for both algorithms.

4-2-2- Second genetic algorithm:

An ideal algorithm should be able to keep high variety degree during transferring from one generation to another generation and unwanted mutations of operators cause deviation of algorithm in section which better answers are traceable. For this purpose, two diversity and intensification operators and one local search method were used. Diversity operator systematically creates the answers which have not been produced so far and expands search domain in answer space. In this operator, frequency of each work in each position is calculated among all reproduction pool answers and the work which was rarely in a special position is identified and a varied answer is produced by putting them together. In case the work which has the lowest frequency is selected in the previous positions during creating an answer, another work which has the lowest frequency is selected and in case such work is not available, this position will remain empty. At the end, the operator determines the works which were not allocated so far and allocate them to the empty positions randomly. Intensification operator has effect on the elite answers and causes to produce answers on the basis of elite answers leading to production of local optimal answers. Stages of this operator execution are similar to that of diversity operator with this difference that the works which have the highest frequency in each position are put beside each other. In genetic algorithm, local search algorithm was used to improve the best answers obtained from each algorithm generation. This method is applied on the best answer of each generation and improves it if possible. This search starts with the first work in the best answer and continues by displacing the adjacent works of the right hand and replaces the previous answer n case of improving target function.

Numerical tests:

In this article, sample problems are designed and produced and then solutions are assessed.

5-1- sample problems:

In this article, sample problems are classified into three small, medium and large classes in group scheduling problem and the number of families was produced in small problems from uniform distribution with range (2,10) , in medium problems from uniform distribution with range (11,20) and large problems from uniform distribution with range (21,30). Parts in each family were produced from uniform distribution with range (2, 15). In order to produce setup time according to research done by Schaller et al, uniform distribution with range (2, 15) was used. Processing time of each part on each machine was produced from uniform distribution with range (5, 25) according to research done by Zegardi et al. In this research, three rates of learning 7, 0,8,0,9, and 0 were randomly used in order to use learning rate according to research done by Lee et al. The number of Machinery for each cell was 10 (problems of 2 to 10 family-parts) for small problems, 20(problems of 11 to 20 family-parts) for medium problems and 30(problems of 21to 30 family-parts) for medium problems. Percentage of machine operations of each part was produced on each machine on the basis of conditions of the problem from uniform distribution with range (0.5, 0.9).

5-2- Assessed methods:

In order to assess efficiency of each solution, computer program of each method was written with use of MATLAB7 software and sample problems were solved on a Pentium 4 computer with Intel processor with processing speed of 3.2 gigahertz and external memory (RAM) of 1024 megabytes. In order to find the suitable value of essential parameters of genetic algorithm i.e. primary population, the number of elite, keep rate and mutation rate , 13 different scenarios were designed for the first genetic algorithm from primary population rate 30,60 and 100 , crossover rate 0.7, 0.8, 0.9, keep rate 0.6 and 0.7 , mutation rate of 0.05, 0.02, 0.01. With regard to the fact that the second genetic algorithm has two diversity and intensification operators, it has two parameters of diversity rate i.e. percentage of answers produced from diversity operator and intensification rate i.e. percentage of answers produced from intensification operator. With regard to the above parameters, diversity rate 0.05 and 0.1 and intensity rate 0.05 , 0.1 and 14 different scenarios were designed for the second genetic algorithm for each scenario , 10 different problems were solved and each problem was executed for 10 times. Average obtained results were regarded as final results to decrease error of algorithms random function. Scenarios results equality hypothesis in the first genetic algorithm and second genetic algorithm was tested with use of randomized blocks design test and scenarios results equality hypothesis was rejected in the first genetic and second genetic algorithm. In order to disclose real differences between scenarios, Duncan multiple range tests was used in each one of the algorithms and suitable scenario was selected for each algorithm. Value of parameters is described in table 1 in selected scenario for the first and second genetic algorithm.

Table 1: parameters of selected scenarios for algorithms

Selected scenario parameters	First genetic algorithm	Second genetic algorithm
Primary population	100	100
Keep rate	0.6	0.6
Crossover rate	0.7	0.7
Elite rate	30	30
Mutation rate	0.01	0
Intensity rate	-	0.05
Diversity rate	-	0.05

After determination of suitable parameters for genetic algorithms and design of sample problems, 10 problems of 2 to 10 family-parts, 10 problems of 11 to 20 family-parts and 10 problems of 21 to 30 family – parts were solved in order to compare the presented solutions and each problem was executed for 10 times in order to avoid any random error and the best answer was included in each algorithm as the algorithm answer. In order to compare methods, two Indices of average algorithm execution time and percentage of answer error were used. In order to calculate error rate, relation 29 was used in which BR was the best response obtained from solution of the problem by the presented algorithms and RA was result obtained from the algorithm solution.

$$RG = \frac{RA - BR}{BR} \quad (29)$$

5-3-work results:

Results obtained from solution of problems 2 to 10 of part families show that average error rate is 0.33% in the first genetic algorithm, 0.01% in the second genetic algorithm and 3.16% in the MCMN heuristic method and average time of execution is 16,5 in the first genetic algorithm and 21.2 in the second genetic algorithm and 0.1 seconds in MCMN heuristic method. Therefore, the second genetic algorithm gives better answers in problems 2 to 10 of part families.

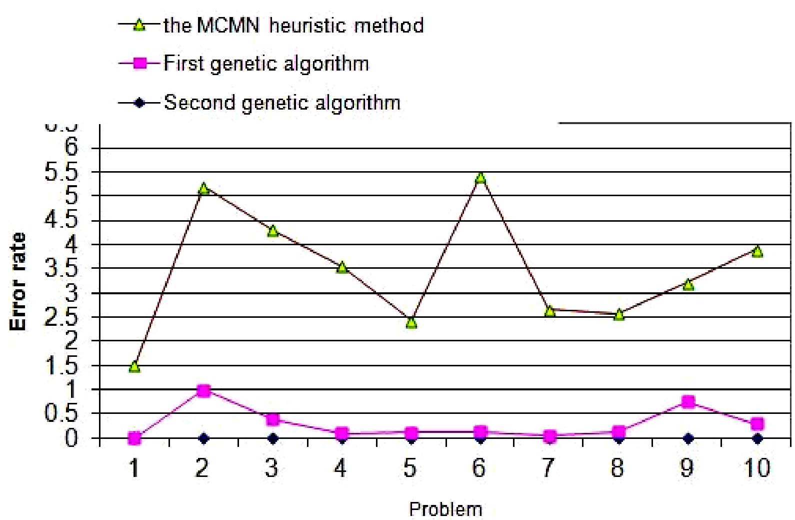

Fig. 1: diagram of algorithms error rate diagram in problems 2 to 10 of part families

Results obtained from solution of problems 11 to 20 of part families show that average error rate is 0.4% in the first genetic algorithm, 0.0% in the second genetic algorithm and 3.25% in the MCMN heuristic method and average time of execution is 46 in the first genetic algorithm and 63.8 in the second genetic algorithm and 0.27 seconds in MCMN heuristic method. Therefore, the second genetic algorithm gives better responses in problems 11 to 20 of part families.

Results obtained from solution of problems 21 to 30 of part families show that average error rate is 0.7% in the first genetic algorithm, 0.0% in the second genetic algorithm and 2.99% in the MCMN heuristic method and average time of execution is 64 in the first genetic algorithm and 82 in the second genetic algorithm and 0.67 seconds in MCMN heuristic method. Therefore, the second genetic algorithm gives better responses in problems 21 to 30 of part families.

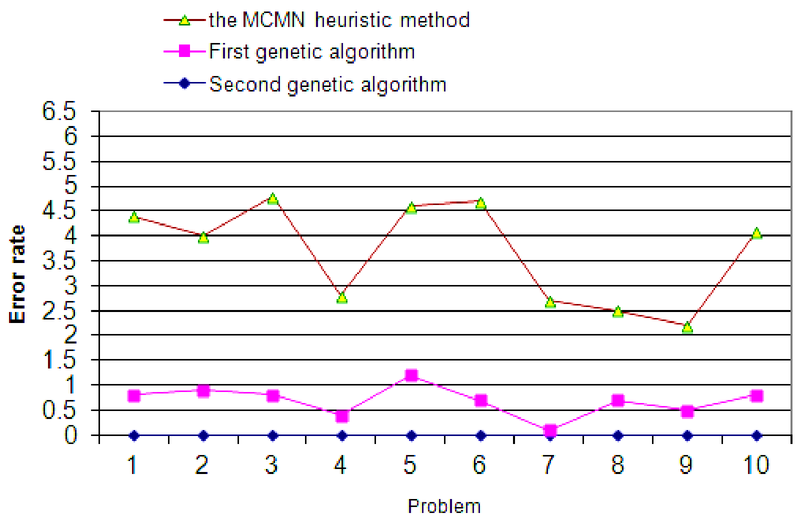

Fig. 2: diagram of algorithms error rate diagram in problems 11 to 20 of part families

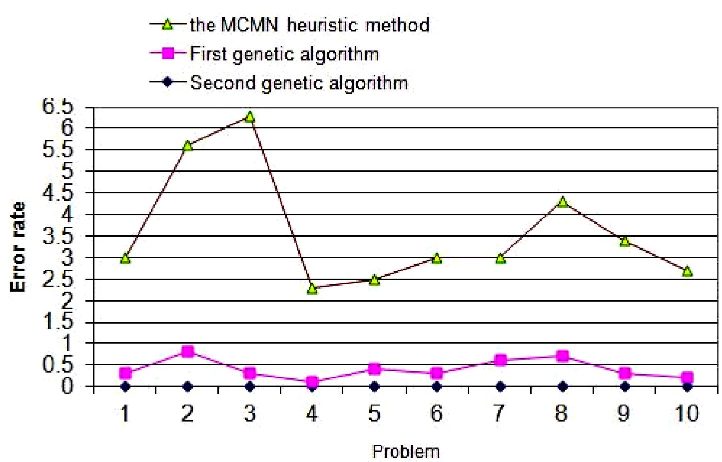

Fig. 3: diagram of algorithms error rate diagram in problems 21 to 30 of part families

As specified from results of small, medium and large problems solutions, the second genetic algorithm has lower error rate than other methods have while MCMN heuristic method has the maximum error rate and the minimum time of execution in comparison to other methods.

6-conclusion:

In this research, position based learning effect was used in group scheduling problem in cellular manufacturing system. Dejong learning model was selected for group scheduling problem and this problem was modeled by assuming structure of flow shop of some machines and in order to minimize the maximum completion time. In order to solve the modeled problem, a heuristic method and two genetic algorithms were developed and efficiency of the proposed solutions was assessed by producing sample problems. Numerical results obtained from sample problem solution showed that the second genetic algorithm has better function in terms of response quality and heuristic method had better function in terms of solution time. In this problem, it was assumed that all parts of a family-part can be processed in a cell. In some cases, we can't produce all parts in one cell. Therefore, parts are transferred from one cell to another cell in order to perform special operations. With regard to intercellular movements, we can develop the above model. In this problem, structure of cells was assumed as flow shop. We can develop the above problem for flexible manufacturing system in the future research.

REFERENCE

Irani, S.I., 1999. Handbook of Cellular Manufacturing Systems, John Wily & Sons Ltd.

Biskup, D., 2008. "A State-of-the-Art Review on Scheduling with Learning Effects", European Journal of Operational Research, 188(2): 315-329.

Biskup, D., 1999."Single-Machine Scheduling with Learning Considerations", European Journal of Operational Research, 115: 173-178.

Mosheiov, G., 2001a. "Scheduling Problems with a Learning Effect", European Journal of Operational Research, 132: 687-693.

Mosheiov, G., 2001b. "Parallel Machine Scheduling with a Learning Effect", Journal of the Operational Research Society, 52: 1-5.

Lee, W., C. Wu, H. Sung, 2004. "A Bi-Criterion Single- Machine Scheduling Problem with Learning Considerations", Acta Informatica, 40: 303-315.

Lee W., & C. Wu, 2004. "Minimizing total completion time in a two-machine flowshop with a learning effect", International Journal of Production Economics, 88: 85-93.

Bachman, A., A. Janiak, 2004. "Scheduling Jobs with Position-Dependent Processing Times", Journal of the Operational Research Society, 55: 257- 264.

Zhao, C.-L., Q.-L. Zhang, H.-Y. Tang, 2004. "Machine Scheduling Problems with Learning Effects", Dynamics of Continuous, discrete and Impulsive Systems, Series A: Mathematical Analysis, 11: 741-750.

Chen, P., C.-C. Wu, W.-C. Lee, 2006. "A bi-Criteria Two- Machine Flow Shop Scheduling Problem with a Learning Effect", Journal of the Operational Research Society, 57: 1113-1125.

Koulamas, C., G.J. Kyparisis, 2007. "Single-Machine and Two-Machine Flowshop Scheduling with General Learning Function", European Journal of Operational Research, 178: 402-407.

Wang, J., Z. Xia, 2005. "Flow-Shop Scheduling with a Learning Effect", Journal of the Operational Research Society, 56: 1325-1330.

Eren, T., E. Guner, 2007. "Minimizing Total Tardiness in a Scheduling Problem with a Learning Effect", Applied Mathematical Modelling, 31: 1351-1361.

Kuo, W., D. Yang, 2007. "Single-Machine Scheduling with Past-Sequence-Dependent Setup Times and Learning Effects", Information Processing Letters, 102: 22-26.

Nanda, R., G.L. Alder, 1982. Learning Curves: Theory and Application, Atlanta: Industrial Engineering & Management Press.

Schaller, J.E., J.N.D. Gupta, A.J. Vakharia, 2000. "Scheduling a Flowline Manufacturing Cell with Sequence Dependent Family Setup Times", European Journal of Operational Research, 125(2): 324-339.

Logendran, R., N. Salmasi, C. Sriskandarajah, 2006. "Two- Machine Group Scheduling Problems in Discrete Parts Manufacturing with Sequence-Dependent Setups",Computers & Operations Research, 33: 158-180.

Zegordi, S.H., K. Itoh, T. Enkawa, 1995. "A Knowledge Smulated Annealing Scheme for the Early/Tardy Flow Shop Scheduling Problem", International Journal of Production Research, 33(5): 1449-1466.