IMPROVED CELLULAR MANUFACTURING SYSTEM USING PARTICLE SWARM OPTIMIZATION

Dhurgham Aiham Kadhim
durghamalshakerchi@gmail.com

ABSTRACT

Cellular manufacturing system is the nerve of industry in our days and many studies and researches discussed in this field . In this research row materials need to cross long distance about thousand meters during the manufacturing process to be final product in ALWAZIRIA ELECRICAL MOTOR FACTORY the problem was needed to reduce the routing of row material flow and increase flexibility with saving same operation sequence . First segmented the facility into subsystems called cells and figured the system output .the research goes in two ways first assign the problem and parameter affected on production process secondly apply the particle swarm to the problem after writing the algorithms. The particle swarm technique shows great flexibility in scheduling purposes problem. Also the product will cross about 400 m instead of 1000 m before using PSO after extracted the result which it discussed and recommended to the future researcher .

KEYWORD : manufacturing cell , particle swarm optimization

انظمة التصنيع الخلوية المطورة باستخدام استمثال عناصر السرب
(Particle Swarm Optimization)

ضرغام ايهم كاظم

الخلاصة

تعتبر انظمة التصنيع الخلوية عصب الصناعة في وقتنا الحالي و هناك دراسات و بحوث عديدة في هذا المجال و نوقشت العديد من نظريات الأمثلية في هذا المجال .في هذا البحث المواد الاولية تحتاج لان تقطع مسافة تقدر كيلو متر واحد لتصبح منتج نهائي في معمل الوزيرية لصناعة المحركات الكهربائية فلابد من تقليل مسارات المواد الاولية وزيادة المرونة في خط الانتاج مع الابقاء على نفس تسلسل العمليات . في البداية تم تقسيم المنشأة الى مجاميع فرعية صغيرة و معرفة الطاقة الانتاجية للمعمل ثم اتجه البحث الى تحديد المشكلة و التعبير الرياضي الذي يمثله المعمل و كذلك المعاملات التي يتأثر بها و اما المرحلة الثانية تم تطبيق النظرية على المعمل و كتابة الخوارزمية الخاصة بالتطبيق والتي اثبتت فاعلية في التطبيق و مرونة عالية في تغيير تسلسل العمليات و تم اختزال الوقت و الزمن حيث كان المواد الاولية تقطع ما يقارب 1000 متر قبل التطبيق ثم و صلت الى ما يقارب 400 متر . و تمت مناقشة النتائج و اعطيت بعض التوصيات.

INTRODUCTION

Cellular manufacturing System (CMS) is a process of manufacturing which is a subsection of just-in-time manufacturing and lean manufacturing cover group technology that endorse unravel a manufacturing system into subsidiary systems to do it easier for managing rather than entire manufacturing system. Portions in this system are gathered into families; each one belongs to parts and machines are specified into cells. Part families are classified based on similar machine types needed for production. Therefore, one of the important missions in the design of a CMS is to differentiate related parts (part families) and their correlating machines (machine cells) which is known as cell formation problem. The cell formation problem (CFP) split a set of parts into distinguish part families, so this set of the same family are similar in terms of production requirements, line production , geometry, and so on.   Thereafter, allocated manufacturing cells is rated for each part family which contains all the production requirements of the conformable part family such as machines, tools, manpower, etc. As a consequence, cellular manufacturing system is a suitable solution to the problem of productivity in a batch production circumference which is used in 50-75% of industries . The basic idea in cellular manufacturing system is the fundamentality of similar things that should be done at the same way; this means that in dedicated manufacturing cells comparable manufacturing process and eminent should be particular and grouped C.S. Krishnamoorthy and S. Rajeev1996.

PARTICLE SWARM OPTIMIZATION

The (PSO) algorithm mimic the attitude of finch congregated. Imagine the next scenario; where a set of bees are randomly searching for food in some region and only one piece of food is being founded. Not all bees know the food place , but know how far is which depend on iteration. What's the best strategy to find the food? The best strategy is to be pursued the closest bee to the food. PSO inspired from bee and it used it to find optimum solution to the problems. In PSO, each individual bird or bee represent unique solution in the search space . It named "particle". For each particle has fitness values which are estimated by the function called fitness function to be optimized, and it has its own velocity and position of   the flying of the particles. The particles fly through the problem space by following the current optimum particles. A group of random set of birds initializes the PSO (solutions); and then searches for an optimal by updating generations S. Narayna and Reddy 2012 .

LITERATURE REVIEW

Here below some researcher in same field presented to show the application of (PSO) in different cases and compared the results also to see how is the gap is filled in this research . Qianwang Deng 2017 the objective of this research is to minimize the maximal total completion time, the amount of work to be done , a bee evolutionary leading non dominated sorting genetic algorithm (BEG-NSGA-II) for Multi objective FJSP (MO-FJSP) is presented. This research , which adopted through the enhancing process a mechanism goes in two-stage optimization for first one , the (NSGA-II) algorithm with number of iteration firstly used to get population N, in which a bee evolutionary orienting scheme is presented to exploit the solution space extensively. Secondly acquire the Pareto-optimal solutions, the NSGA-II algorithm with GEN set of iteration used again. For second stage, updating of mechanisms is granted to enhance the searching efficiency and avert the convergence coming early. To be more accurate, the population contains three parts, each one changed with the iteration. Also some , numerical work are executed .
Zhang 2009 . Hybrid algorithm constructed on Particle Swarm Optimization (PSO) and Simulated annealing (SA) is presented , to solve Flexible Job Shop Scheduling with multi

objectives total workload maximal machine workload minimize make-span, , machine idle time, & total tardiness. Rescheduling strategy used to mix up workload if one machine breakdown in this algorithm which presented. The hybrid algorithm compile high general search performance of PSO with the powerful to avoid the trap in local minimum of SA . Objective of hybrid multi-PSO (MPSO) and SA algorithm is presented to differentiate the approximation of the Pareto front for Flexible job shop scheduling (FJSSP). Pareto front and overload distance is used to identify the particles fitness. MPSO is significant to global search and SA used to local search. Kamble, S. 2015 this paper works on a special job-shop scheduling problem that handle capabilities of accommodation and different double-dates. And developing the model, which can accord process capacity and different double-dates in verity tasks to choose some to process, and the goal is as far as possible to maximize the ability for throughput . to solve complexity in scheduling problems present , a discrete (PSO) algorithm for testing ability of performance to the novel algorithm, they use some random instances to emulate practical problems. Lian, Zhigang, 2014 In this work , the principle theory of adaptive PSO is figured . Also, he code and select of parameters as well as decipher of Adaptive-PSO are calculated . The maximum of flow time should minimized to maintain the ability of the algorithm, and tested to find optimum solution for scheduling purpose. A great number of testing carried out to the algorithm has best feasibility and good performance in job-shop scheduling . The presented work get close results to minimize total completion time just like Qianwang Deng 2017 by using genetic algorithms and with Zhang 2009 where they use hybrid algorithm to solve job shop scheduling also the existing paper reallocate the resources and machines to solve the operation sequences .The deference's of existing work from other researcher are use the PSO algorithm to reallocate the machine and redistribute the geography of plant according to the constraints supplied to algorithms with saving process sequence and reducing traveling distance to the row material inside factory

METHODOLOGY

In classical cellular manufacturing systems (CMSs) each cell has machines specified to machine with particular jobs or task used to be close to each other to reduce the material handling costs and flow of times. As Consequences, rearrange the facility geography will reduce also process cycle change must be taken. In addition, doing reconfiguration considerably become impractical. Though using cellular system in facility organized jobs and reduces material handling efforts in dynamic environments with unstable demands and unpredictable compositions in implementation of CMSs. Because of reconfiguration of one product-mix may passive in another environment and redesigning required significant inter cell flows must be satisfaction . To minimize inter cell flows, resource persistent leads to higher investment costs and divert in utilization among resources duplicated. To reduce adverse effects of CMSs while keeping the positive one, companies have been encouraged to use artificial or virtual cellular manufacturing systems. Lian & Zhigang 2014.

PROBLEM ASSIGNMENT

Actually , issues concerning with distance that the part crosses it in the plant are done using part-machine incidence matrix, and also its desired to construct cells of machines with predication to "spaces" between them. It is deemed a group of technology type. Throw distributing each part to the group where it induces, manufacturing cells are then adjusted completely. The assumption in can be handled as follow B. H. Ateme-Nguema & T.-M. Dao 2007

$$A = \lambda_{ip},$$ (1)

where $\lambda_{ip} = \begin{cases} 1 \text{ if product } p \text{ visit machine } i \text{ at step } t \\ 0 \quad \text{if not} \end{cases}$

If the product get processed   on machines i , j respectively,

$m_{ip} = \begin{cases} 1 \text{ if } m_{ip} * m_{ip} \neq 0 \text{ and } |m_{ip} - m_{ip}| = 1 \\ 0 \qquad\qquad\qquad\qquad\qquad\quad \text{if not} \end{cases}$

Where m_{ip} represent the constraint, if the part will be processed   sequentially in two stages or not. in general the parts in all steps .

$$C_{ij} = \sum_{p=1}^{p} m_{ip} \text{ where } i = 1 \text{ to } n - 1 \tag{2}$$

And j = 1+I to n

$$z = min \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} C_{ij} d_{ij} \tag{3}$$

Where d_{ij} logically represent ($1 - x_{ij}$)

Which it represent objective function under following constraint .

$$\sum_{i=1}^{k-1} x_{ij} + \sum_{j=k+1}^{n} \leq m - 1 \tag{4}$$

$x_{ij} = \begin{cases} 1 & \text{if the tow process done in same cell } i = \text{to } n-1 \\ 0 & \text{if not } j = i+1 \text{ to } n \end{cases}$

Constraint (4) prevents confusing between machines when there is more than one A. S. Ramkumar 2009

$$\left.\begin{array}{l} x_{ij} + x_{ik} - x_{ki} \leq 1 \text{ where } i = 1 \quad \text{to } n - 1 \\ x_{ij} - x_{ik} - x_{ki} \leq 1 \text{ where } j = i + 1 \text{ to } n - 1 \\ -x_{ij} - x_{ik} + x_{ki} \leq 1 \text{ where } k = j + 1 \text{ to } n - 1 \end{array}\right\} \tag{5}$$

Where (m) Machine at same cell and it also keep correct operations sequences (k) , and for triangular constraints in equation (5) it enforced the integrity of cells.
Adjustment of any revisiting part occurred within a same machine could not take more than zero. Some parameters like volume weighting in the routings must be considered. If the maximum cell size, is fixed it can be handled in quadratic assignment or matrix norm minimization problem. If there are k cells of pre-identify sizes m1, m2,m3,……..…, m_k and y_{ik}, where i=1 to n & and k=1 to k, these binaries variables ensures that the machines m are allocated to cell k and could be connected with (6) in relations as below.

$$x_{ij} = \sum_{k=1}^{k} y_{ik} y_{kj} \tag{6}$$

 Where i = to n-1 , j = i + 1

$$z = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} C_{ij} - max \sum_{i=1}^{n-1} \sum_{j=1+1}^{n} \sum_{k=1}^{k} C_{ij} y_{ik} y_{ik} \tag{7}$$

$$\sum_{k=1}^{k} y_{ik} \quad i = 1 \text{ to } n \tag{8}$$

$$\sum_{i=1}^{n} y_{jk} = m \quad k = 1 \text{ to } k \tag{9}$$

$y_{ik} \in \{0 \, / 1\}$ $i = 1$ to n & k = 1 to k   (10)

   In eq. 10 and 11 for ensuring the machine at same cell and prevent perplexing between machines in same cell or in other cells .

MODULATION PARTICLE SWARM

   Eberhart and Kennedy 1995, propose particle swarm optimization (PSO) algorithm, second type (AI), that is simulated by social manner of bird flocking or school of fish, where they find food together in a specific area. This algorithm is developed and used over years.
   The birds and bees transfer from place to another by coordinate collected using the best and shortest way during the participate between these individuals, This can be obtained by detecting the position .and velocity which may represents a solution in search space . During the iteration, objective function is established to maintain the fitness value of each individual used position coordinate of   as input. Fitness values decide the best position. Each individual flies in the search space with a velocity that is periodically adjusted depend on its own flying strategy.
 For each individual the tracking of own coordinates in the search space is correlated with the best position if it has achieved. This fitness value is also kept which is called (p_{best}). In other side the individuals that are closed attempt to shift away from other if they are too close and then fly towards the average heading of its next individual so they attempt to go towards the average position of closest individual. Their movement looks like path that attracted to the location of the roost so the individuals remembered how to get closer roost as it cleared in figure (1).
The individual adjust itself by doing two main procedures.
   - Global best (g_{best}) means that all particles should be updated
   - Immediately when there is a new good position found.

Neighbor best means individuals immediately shared information with a subset of swarm about best locations "Best" value that is followed   by the particle swarm optimizer is optimal value, are gained by any individual in the neighbors of the swarm. This position named vl_{best} . When a individual gets all of the all swarm as its topological neighbors, the best value is a global best and is named g_{best}. Acceleration of all individuals are weighted in random way with random numbers being created for acceleration (p_{best}) and (l_{best}) positions. If there is a particle in the search space (s^k) so the update of new location can be modeled as below Albert Y. Zomaya 2006

$$v_i^{k+1} = wv_i^k + c_1 rand_1 \, (...) * (pbest_i - s_i^k) * c_2 rand_2 \, (...) * (gbest_i - s_i^k) \qquad (11)$$

where

$$W = w_{max} - [(w_{max} - w_{min})] * iter / iter_{max} \qquad (12)$$

$$s_i^{k+1} = s_i^k + v_i^{k+1} \qquad (13)$$

v_i^k: Individual's velocity at iteration k,

W: function of weight,

c_i: Randomly weight factor,

Rand: fixed value between 0 and 1,

s_i^k: Instant position of individual at iteration k,

$pbest_i$: Best position of individual i,

$gbest_i$: Global position of the swarm.

w_{max}: Initial weight ,w_{min}: final weight ,
max_{iter}: maximum iteration no,

IMPLEMENTATION OF PARTICLE SWARM SYSTEM

To implement (**PSO**) there is some issues need to figure out, like the site plan definition and the movement products. The distance between nodes. In the **PSO** algorithm, solution of the optimization problem is respected to the bee in the search space, which is called individual. Each individual has its own velocity derived to   the direction and range of the flying of the individual are specified and a fitness value that is decided by the optimized function. The individual explore the solution space by scan the all search space. To implement (**PSO**) firstly identifying   the parameters that the system depending on it. The inputs parameters should include the site plan of the plant and the implementation of the thesis, study was carried out at the ministry of industry/ Iraq department of motor assembly. The plant area is about (3674) square meter (90m*42m) and it produce electric motor for air cooler. The total production about (1000) pieces in month and each piece should cross over 1000m. The coded machine in the plant used numbers from 1 to 37 as it showed in Table no. 1 The production process will happen in five phase . each number in each phase represent the code of section or machine group which the part are processed during the life cycle to be final product (electrical motor).

    I.   Production of the stator and it take the following route.
          27-28-14-36-26-37-25-24-23-22-21-20-15-16-17.
    II. Frame and base manufacturing also it take the following route.
          27-34-32-33-35-24-23-22-21-20-15-16-17.
    III. Front and rear cover.
          29-30-1-2-3-20-15-16-17.
    IV. Manufacturing of rotor. 27-13-11-12.
    V. Shaft route. 19-6-5-4-7-8-9-10-12-18-20-16-17.

PARTICLE SWARM SYSTEM ALGORITHIM

Here is some steps to be followed by the algorithms.as the flow chart in figure 2
Step 1. Initialization
Set the number of individual , maximum iteration (I_{max}) and each individual location .
Calculate the distance between cells and store the distance matrix Initialize swarm size.
• Set constants c_1 and c_2.
• Randomly initialize particle position.
• Randomly initialize particle velocity.
• Set counter to 1.

Step 2. Calculate fitness
Calculate the fitness of each particle based on the above distance matrix dist. Every particle utilizes the current position as the present best position (pBest) of its own, and the global best position (gBest) is the best position in current particle swarms.
Evaluate fitness function f_k^i at x_k^i
    • IF $f_k^i < f_{best}^i$ then $f_k^i = f_{best}^i$, $p_k^i = x_k^i$.
    • If stopping condition is satisfied go end and show results.
    • Update particle velocity v_{k+1}^i and position x_{k+1}^i.
    • Increment I . If i<p then increment counter, i = 1.
    • Go to evaluate fitness function.

Step 3. Update p_{best}
 calculate the fitness of individuals by Replacing the p_{best} of the current individuals with instance position of individual if it's better than fitness
Step 4.
Update global position g_{best}
Update the g_{best} based on all particles' p_{best} .
Step 5
Update position of particle Update position of particle for each particle according to equations below.

$$x_i^1 = x_i(t) \oplus ((c1 \text{ X } r1) \ p_i(t)) \oplus ((c2 \text{ X } r2)p_g(t))$$
$$x_i(t+1) = \text{ inject } (x_i^1 v_c)\backslash\backslash\text{reverse } (x_i^1,s,e)$$
$$x_i(t+1) = \eta \text{ X disturb } (x_i \ (t+1),k)$$
$$\eta = \begin{cases} 1 \text{ diversity } < 0.4 \\ o \text{ diversity } \geq 0.4 \end{cases}$$

Step 6.
Termination criterion If
$$iter < Itermax, \text{then go to step 2.}$$
Step 7.
 Output the global best position g_{best}.

PARTICLE SWARM OPTIMIZATION RESULTS.

   After applying PSO to the problem and studying the movement of the particle in the search space and the simulation of individuals it shows that they moves according information sharing during its searching on best solution for best and short route first each individual has its own position and velocity as (a , b and c) in figure 3 which is placed randomly then they share information of each individuals as its mentioned previously each individual represent unique solution for the plant . After that updating the individuals' p_{best} to g_{best} to find best solution which will be done gradually as it cleared in (d , e and f) in figure (3) . Figure 3 shows the way that swarm moves from personal best to global best and find the best solution optimum solution inside the plant .For each individual there is phase represent route which is defer from other but after sharing information about best route (location , position , velocity) . And that was updated instantaneously throw information sharing . the update will choose the best and simplest route to take on for all particle the figures below shows the change of distance according to iteration . Figure (4) shows the iteration of PSO till reach the final position after sharing information which means that the updating of weighted values w_{min} and w_{max} will depend on individuals no. and cycle or number of Iteration used in PSO algorithm as it cleared its start at initial values and reached to maximum value because of no. of solution required in the beginning of comparison between the deferent solution for deferent individual (each individual represent unique solution) then it will be stable after 10 cycle because of the solution will close to the convergence . Also you will find great change in routes of process with taking in consideration the sequences of operation after applying PSO algorithm to the plant as it cleared in figure (5) .
According to the result of work shown that the total distance should the products will travel about more than 800 meter inside the plant in addition there was home and away trips and that can be as multi-cycle before applying artificial work .
  - The initialization scheme of algorithm must be accurate to maintain and state operation sequencing

- After monitoring the results and showing the site plan its significant to take in to account that the total distance that the products should cross about 400 meter and iteration about 1246 (generation) .

- Results discus the behavior of product and life cycle inside the plant and how it will change the road map for changing the way and the procedure of solution as below .

    The sequences of operation as follow .

VI. Production of the stator and it take the following route

    29- 30- 13- 10 -1- 2- 3- 20 .

VII.   Frame and base manufacturing also it take the following route.

    27- 28- 13- 11- 12-.

VIII.   Front and rear cover.   19-6-5-4-3-7-8-9-10-12-18-21.

IX. Manufacturing of rotor.   27- 31- 34-32-33-35-24 .

    Shaft route.   27-28-14-26-27-25-24-23-22-21-20-15-16-17
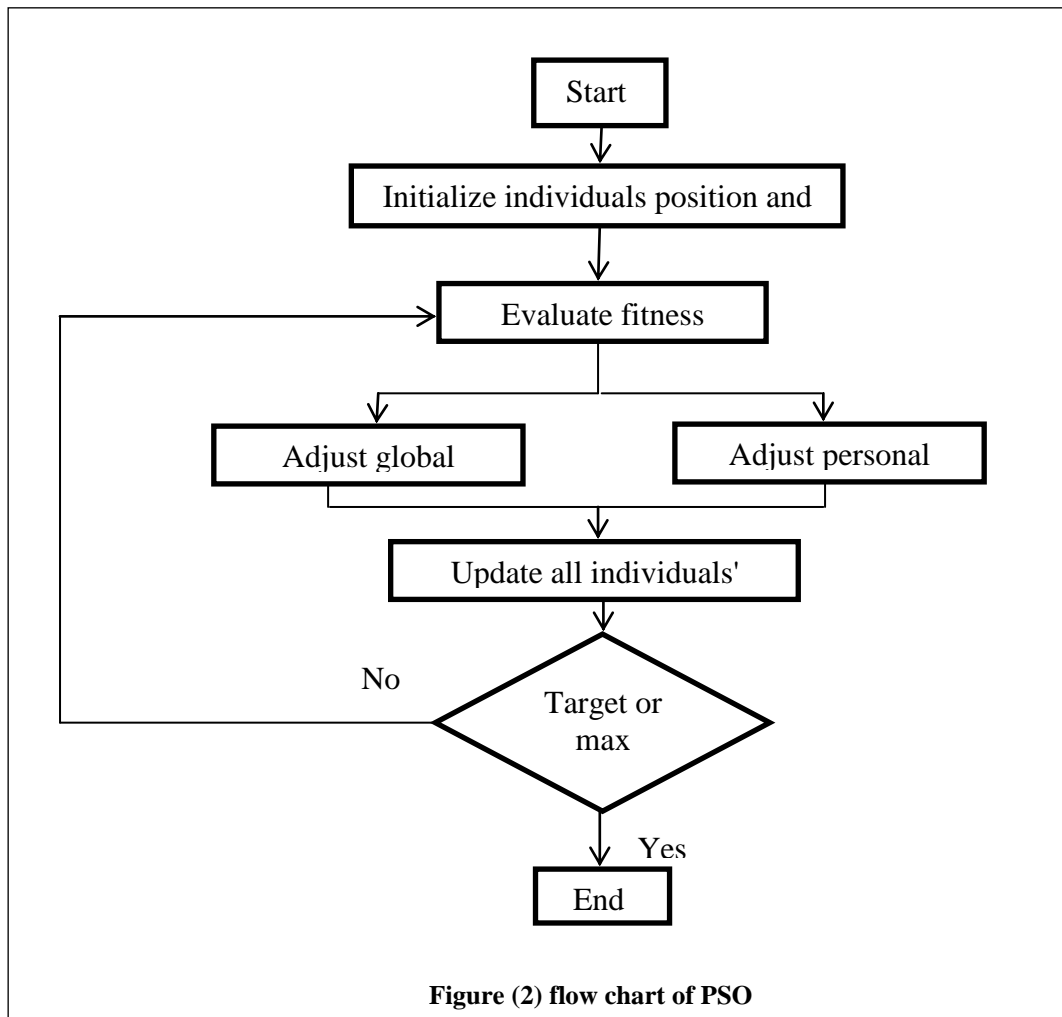

CONCLUSIONS AND FUTURE WORK

The presented result of PSO shows great flexibility in practical and also for the stochastic problem that related to the sequences of operation inside cells . the
randomly weight picked depend on the consuming time during each process . The initial position and weight picked for the first time according to the existing state of the plant . The constant parameter affected on time consuming for each process .
The contributions of this study may be further developed in several ways. Promising directions for future work are discussed in the following, separately for each part of this work .

- Enriched the algorithms with neural network and make a union algorithms works independently   and extract best solution .

- Use GA to build update weight. And bias of neural network online in the plant parameters

- Rebuild PSO algorithms dependent on genetic algorithm in such way that the genetic restructured the ant and swarm algorithms.
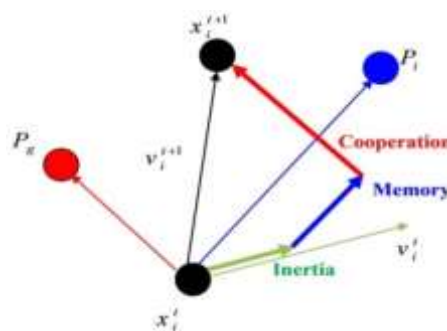
Figure(1) movements of individual according best position

```
                              ┌──────────┐
                              │  Start   │
                              └────┬─────┘
                                   ▼
                    ┌──────────────────────────────┐
                    │ Initialize individuals position and │
                    └──────────────┬───────────────┘
                                   ▼
                         ┌──────────────────┐
      ┌─────────────────►│ Evaluate fitness │
      │                  └────────┬─────────┘
      │               ┌───────────┴───────────┐
      │               ▼                       ▼
      │      ┌──────────────┐         ┌────────────────┐
      │      │ Adjust global│         │ Adjust personal│
      │      └───────┬──────┘         └───────┬────────┘
      │              └───────────┬────────────┘
      │                          ▼
      │               ┌────────────────────────┐
      │               │ Update all individuals'│
      │               └───────────┬────────────┘
      │                           ▼
      │   No                 ╱─────────╲
      └──────────────────── ╱ Target or  ╲
                            ╲   max      ╱
                             ╲─────────╱
                                  │ Yes
                                  ▼
                            ┌──────────┐
                            │   End    │
                            └──────────┘
```
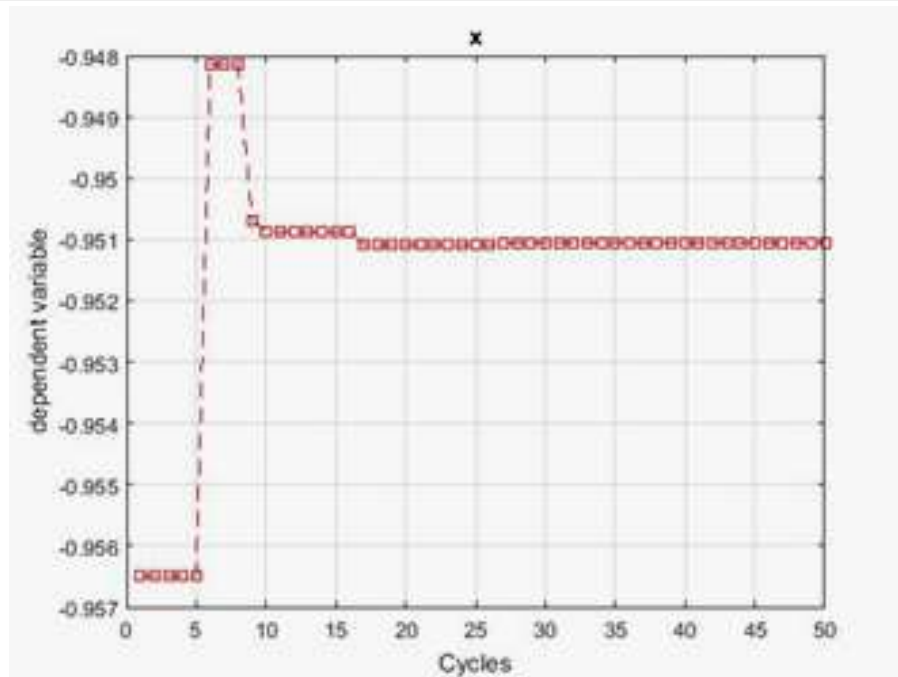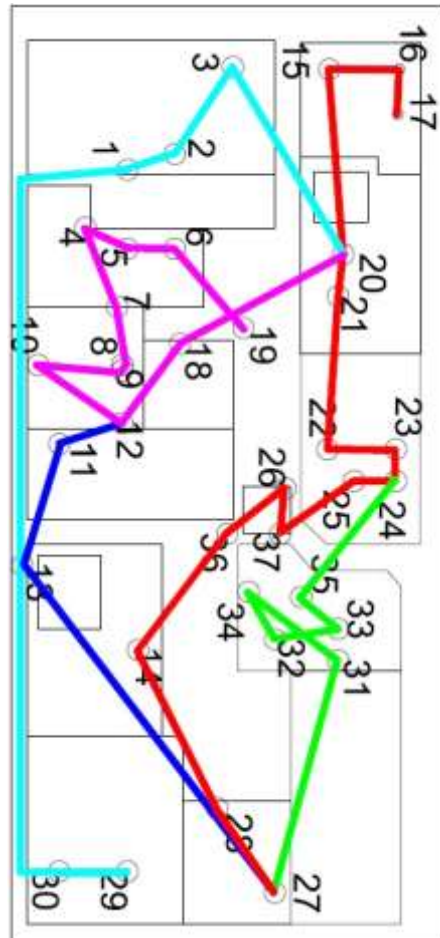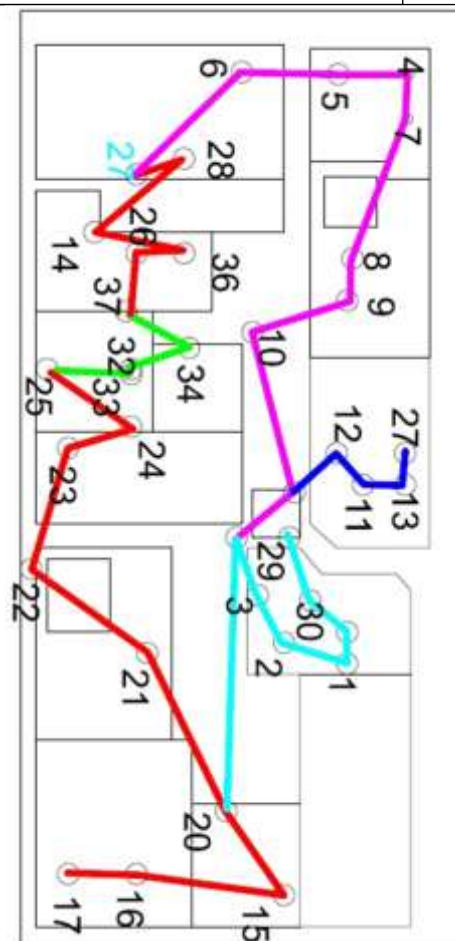
Figure (2) flow chart of PSO
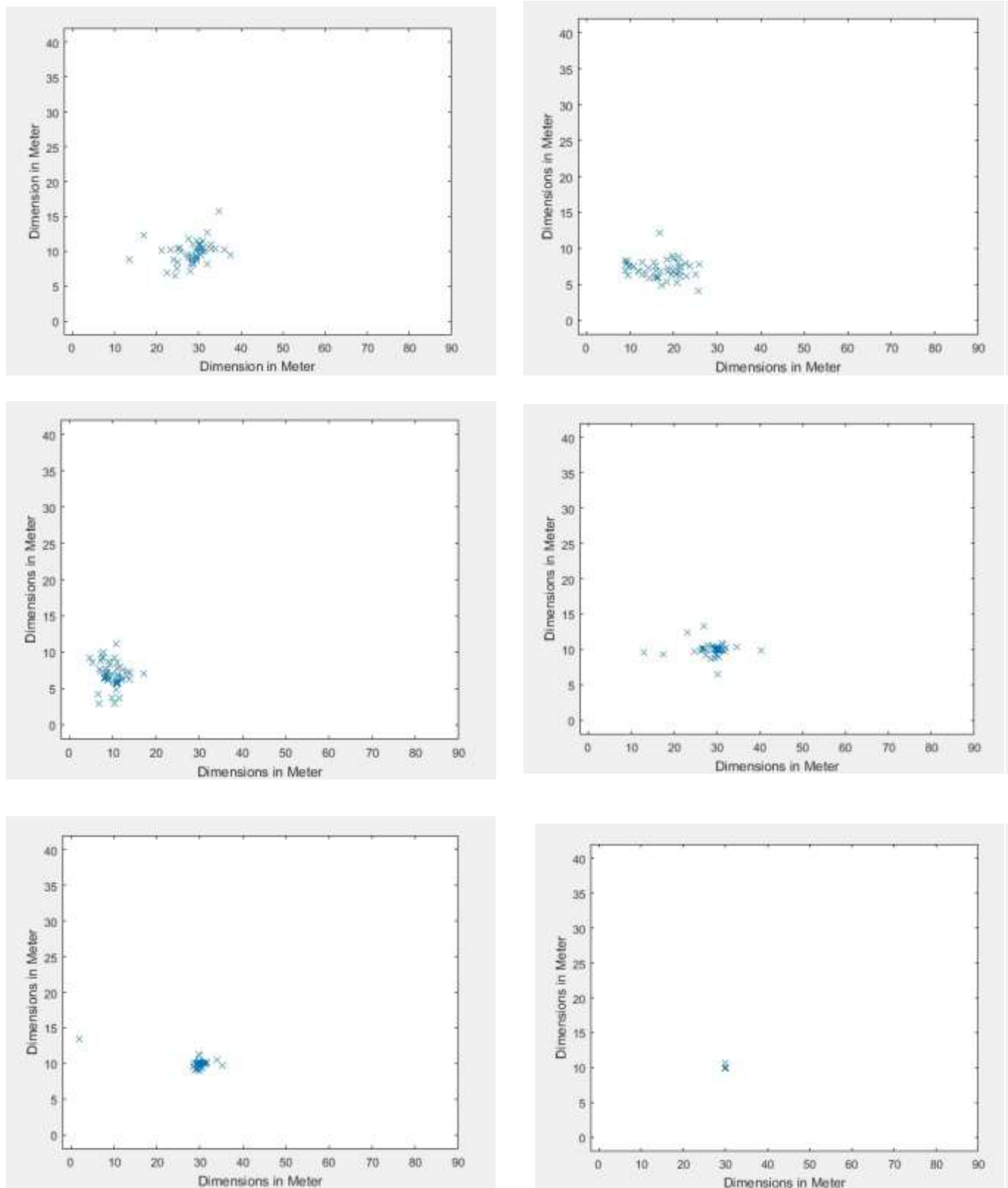
E   F

Figure (3) PSO simulation through searching in solution in solution space

Figure(4) PSO Iteration effect on weighted variable

Before applying PSO　　　　　　　　　　　After Applying PSO

\	Table 1: coding of process inside the plan		
code	Process	Code	Process
1	Front and rear cover turning machine.	20	Assembly.
2	Washing front and rear cover machine	21	Immersing insulators.
3	Painting section.	22	Electrical field testing.
4	Shaft grinding machine.	23	Connecting wires and welding.
5	Shaft turning machine.	24	Compounding the frame with stator.
6	Shaft sleeve turning machine.	25	Adding coil to the stator.
7	Blacking the shafts.	26	Stator turning machine.
8	Immersing in acid.	27	Raw materials.
9	Immersing in acid.	28	Lamination punch machine.
10	Washing.	29	Die casting for the front cover.
11	Rotor core turning machine.	30	Die casting for the rear cover.
12	Heating and coupling center.	31	Frame and base punch machine.
13	Rotor die casting.	32	Welding frame and base machine.
14	Annealing center.	33	Turing machine.
15	Painting.	34	Turing frame and base machine.
16	Adding condenser.	35	Quality control.
17	Packaging.	36	Grouping lamination.
18	Quality control center.	37	Compounding the insulator.
19	Shaft raw material store.		

REFERENCES

Chen, Z. L. (2004). Simultaneous job scheduling and resource allocation on parallel machines. Annals of Operations Research, 129(1), 135-153.

Cheng, C. H., Goh, C. H., & Lee, A. (1996). Solving the generalized machine assignment problem in group technology. Journal of the Operational Research Society, 47(6), 794-802.

Cheng, M. Y., & Lien, L. C. (2012). A hybrid AI-based particle bee algorithm for facility layout optimization. Engineering with Computers, 28(1), 57-69..

Deng, Q., Gong, G., Gong, X., Zhang, L., Liu, W., & Ren, Q. (2017). A Bee Evolutionary Guiding Nondominated Sorting Genetic Algorithm II for Multiobjective Flexible Job-Shop Scheduling. Computational intelligence and neuroscience, 2017.

El-Baz, M. A. (2004). A genetic algorithm for facility layout problems of different manufacturing environments. Computers & Industrial Engineering, 47(2), 233-246..

Hamedi, M., Ismail, N., Esmaeilian, G., & Ariffin, M. (2012). Virtual cellular manufacturing system based on resource element approach and analyzing its performance over different basic layouts. International Journal of Industrial Engineering Computations, 3(2), 265-276.

Hu, G. H., Chen, Y. P., Zhou, Z. D., & Fang, H. C. (2007). A genetic algorithm for the inter-cell layout and material handling system design. The International Journal of Advanced Manufacturing Technology, 34(11), 1153-1163.

Jannat, S., Khaled, A. A., & Paul, S. K. (2010, January). Optimal solution for multi-objective facility layout problem using genetic algorithm. In Proc. 2010 Intl. Conference on Industrial Engineering and Operations Management.

Jithavech, Id, and Krishna Kumar Krishnan. "A simulation-based approach for risk assessment of facility layout designs under stochastic product demands." The International Journal of Advanced Manufacturing Technology 49, no. 1-4 (2010): 27-40.

Kamble, S. V., Mane, S. U., & Umbarkar, A. J. (2015). Hybrid multi-objective particle swarm optimization for flexible job shop scheduling problem. International Journal of Intelligent Systems and Applications, 7(4), 54.

Krishnamoorthy, C. S., & Rajeev, S. (1996). Artificial intelligence and expert systems for engineers (Vol. 11). CRC press..

Krishnan, K. K., Mirzaei, S., Venkatasamy, V., & Pillai, V. M. (2012). A comprehensive approach to facility layout design and cell formation. The International Journal of Advanced Manufacturing Technology, 59(5-8), 737-753.

Lian, Z., Lin, W., Gao, Y., & Jiao, B. (2014). A discrete particle swarm optimization algorithm for job-shop scheduling problem to maximizing production. Int. J. Innov. Comput. Inf. Control, 10(2), 729-740.

Liu, L., Shu, Z., Hu, X., Hu, X., & Cai, H. (2011). Resource allocation and network evolution considering economics and robustness in manufacturing grid. The International Journal of Advanced Manufacturing Technology, 57(1), 393-410.

Narayanaswamy, R. (2002). STRATEGIC LAYOUT PLANNING AND SIMULATION FOR LEAN MANUFACTURING A LAYOPTTM TUTORIAL. Production Modeling Corporation.

Ngampak, N., & Phruksaphanrat, B. (2011, March). Cellular manufacturing layout design and selection: A case study of electronic manufacturing service plant. In Proceedings of the International Multi Conference of Engineers and Computer Scientists.

Nori, V. S., & Sarker, B. R. (1997). Reducing work-in-process movement for multiple products in one-dimensional layout problems. Journal of the Operational Research Society, 48(4), 412-422.

Prakash, A., Tiwari, M. K., & Shankar, R. (2008). Optimal job sequence determination and operation machine allocation in flexible manufacturing systems: an approach using adaptive hierarchical ant colony algorithm. Journal of Intelligent Manufacturing, 19(2), 161-173.

Ravichandran, K. S., Rao, K. C. S., & Saravanan, R. (2002). The role of fuzzy and genetic algorithms in part family formation and sequence optimisation for flexible manufacturing systems. The International Journal of Advanced Manufacturing Technology, 19(12), 879-888.

Reddy, S. N., Varaprasad, V., & Veeranna, V. (2012). OPTIMIZATION OF MULTI-OBJECTVE FACILITY LAYOUT USING NON-TRADITIONAL OPTIMIZATION TECHNIQUE. International Journal of Engineering Science and Technology, 4(2).

Yalaoui, N., Mahdi, H., Amodeo, L., & Yalaoui, F. (2011). A new approach for workshop design. Journal of Intelligent Manufacturing, 22(6), 933-951

Yang, T., & Peters, B. A. (1997). A spine layout design method for semiconductor fabrication facilities containing automated material-handling systems. International Journal of Operations & Production Management, 17(5), 490-501.

Zhang, G., Shao, X., Li, P., & Gao, L. (2009). An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem. Computers & Industrial Engineering, 56(4), 1309-1318.