

A New Combined Approach for Optimizing Makespan in Job Shop Scheduling Problem

S. Sureshkumar¹, Dr.S.Thiruvankadam²

Assistant Professor, Department of Computer Engineering, P. A. College of Engineering and Technology, Pollachi,
TamilNadu, India¹

Professor and Head, Department of Information Technology, P. A. College of Engineering and Technology, Pollachi,
TamilNadu, India²

ABSTRACT: Scheduling is the most important issue to be solved in the real time environment. One such emerging problem in the scheduling is the job shop scheduling problem, applied in various fields of engineering. Recently many works have been reported to reduce the makespan time in JSP. No Scheduling technique has guaranteed optimality. This paper aims at providing a well optimized scheduling technique, reducing the makespan based on precedence and capacity constraints allowing a machine to process almost one operation at a time. This paper proposed a Genetic algorithm with unordered subsequence Exchange cross-over (USXX) and Hybrid approach called a PSO-GA. The proposed algorithm is examined by using a set of benchmark instances with various sizes. The hybrid approach produced the better computational time with minimum iterations compare to the other existing algorithms.

KEYWORDS: Job shop scheduling, Makespan, Genetic algorithm, Particle swarm Optimization.

I. INTRODUCTION

The JOB-SHOP scheduling problem (JSSP) is a very important practical problem in both fields of production management and combinatorial optimization. Very important to bring out the efficient methods for solving the JSSP have significant effects on profitability and product quality. The JSSP has drawn the attention of researchers for last three decades, but because scheduling problem varies widely according to specific production tasks and most of them are strongly nondeterministic polynomial time hard problems, some test problems of moderate size are still unsolved.

In fact, only small-size instances of the problems can be solved within a reasonable computational time by exact optimization algorithms such as branch and bound [1], [2] and dynamic programming [3], [4], including the benchmark instance 10 x 10 of Fisher and Thompson, which was proposed in 1963 and only solved 20 years later. Problems of dimension 15 x 15 are still considered to be beyond the reach of today's exact methods. By contrast, approximate and heuristic methods make a tradeoff between solution quality and computational cost. These methods mainly include dispatching priority rules [5]-[7], shifting bottleneck approach [8], [9], Lagrangian relaxation [10], [11], and tabu search [12]-[14] and have made considerable achievement. In recently much attention has been devoted to metaheuristics with the emergence of new techniques from the field of artificial intelligence such as genetic algorithm (GA)[15]-[18], simulated annealing [19]-[22], ant colony optimization [23], Particle swarm optimization(PSO) [24], [25], artificial neural network [26]-[28], artificial immune system(AIS) [29].

These metaheuristics can be regarded as problem independent approaches and are well suited to solve complex problems that may be difficult to solve by traditional technique. Moreover, they are capable of producing high-quality solutions with reasonable computational effort. Among the metaheuristics algorithms, GA has been used with increasing frequency to address scheduling problems and may not remain to have much room for improvement. However, the use of PSO and GA for the solution of scheduling problems has been scarce, specifically the PSO. In addition, many research results of the JSSP show that it is difficult to obtain a good enough solution only by single

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

search scheme. Motivated by these perspectives, we propose a novel hybrid intelligent algorithm for the JSSP based on PSO and GA in this paper. Both PSO and GA are evolutionary computation techniques based on swarm intelligence. They exhibit implicit parallelism and contain certain redundancy and historical information of past solutions. Therefore, the proposed hybrid algorithm also effectively exploits the capabilities of distributed and parallel computing of swarm intelligence approaches.

II. OVER VIEW OF JSSP

Job shop Scheduling Problem is a well known constraints satisfaction problem in the field of In a JSP we have a finite set of N jobs, where $N = \{1 \dots n\}$, that have to process in a set M of machines, where $M = \{1 \dots m\}$. Each and every job should divide into a series of m operations O_{ik} , where subscript k indicates the machine M_k on which the operation has to be processed. The technological order of machines i.e. process routing for job is predefined. We consider the static job shop scheduling case every job is a chain of operations and every operation has to be processed on a given machine for a given time. The task is to find the completion time of the very last operation is minimal. The chain order of each job has to be maintained and each machine can only process one job at the same time. Once an operation starts, it must be completed. Two operations of a job cannot be processed at the same time. No more than one job can be handled on a machine at the same time. There is no setup time and idle time. The cost value is not considered. The following additional definitions and notations will help in formulating the problem:

Table 1. The notations of Conceptual Model

| Notations | Description |
|------------|--------------------------------------|
| M | Number of Machines |
| N | Number of Jobs |
| $O_{i,j}$ | j^{th} operation of job i |
| $T_{i,j}$ | Processing time of $O_{i,j}$ |
| $F_{i,j}$ | Finish time of $O_{i,j}$ |
| C_i | Completion time of job i |
| N_i | Number of operation of job i |
| C_{\max} | Maximum completion time of all job |
| O_{ik} | The start time of j on machine k |

Minimization of makespan :

$$f(x) = C_{\max}(O_{jk} + T_{ij}) \quad (1)$$

s.t

$$F_{ij} \leq T_{i,j+1} \quad i = 1, \dots, n_i \quad (2)$$

$$Y_{mij} \leq a_{mij} \quad m = 1, \dots, M \quad (3)$$

Constraint (1) specifies the makespan completion time and (2) indicates the Precedence Constraints among the operation is executed. Constraint (3) indicates that can be assigned to just one machine from among the given machines

A. Example Problem

This problem contains 3 machines (m), 3 jobs (n), Process routing, processing time also given. If the operation sequence is given it will work based On the Precedence constraints and capacity constraints and satisfies the constraints of the problem also. Let we see how it works.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

| | |
|-----------------------------------|---------------------------|
| Permutation Representation | Operation Sequence |
| $J_1 = \{M_3, M_1, M_2\}$ | $M_1 = (J_3, J_2, J_1)$ |
| $J_2 = \{M_2, M_3, M_1\}$ | $M_2 = (J_3, J_2, J_1)$ |
| $J_3 = \{M_2, M_1, M_3\}$ | $M_3 = (J_3, J_2, J_1)$ |

TABLE I. OPERATION PROCESSING TIMES

| JOBS | MACHINES | | |
|----------------|----------------|----------------|----------------|
| | M ₁ | M ₂ | M ₃ |
| J ₁ | 40 | 20 | 55 |
| J ₂ | 30 | 50 | 45 |
| J ₃ | 20 | 40 | 30 |

The above inputs the schedule information is showed in the Fig. 1. And makespan time is calculated using the Gantt chart.

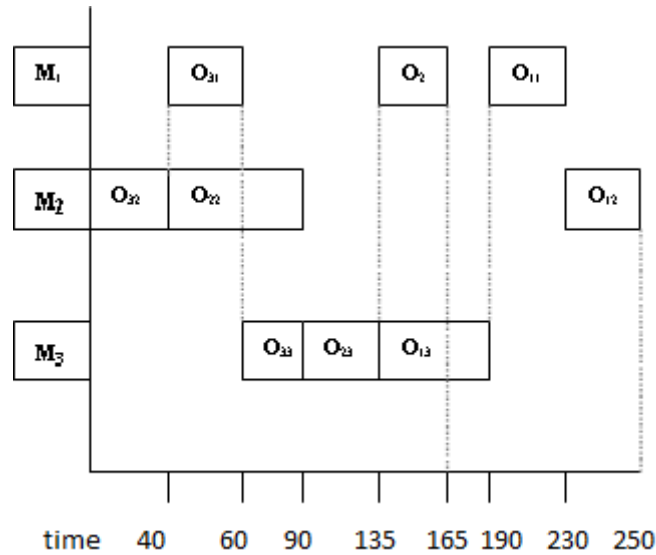


Figure 1. Scheduled Visual information using Gantt chart

For the above operation sequence or schedule and machine allocation or permutation, with operation time the makespan time is 250 calculated through Gantt chart. The scheduled information in Fig 1. Clearly showed satisfies the precedence and capacity constraints. Based on **Table - I** input Gantt chart process is Machine2 (M2) start the first operation for Job3(J₃) with allocated time of 40 minutes but at the same time no process will not allocate on machine M₂. Likewise all the remaining operations are will be executed as per given schedule and allocated time without violation of the constraints.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

III. GENETIC ALGORITHM FOR JSSP

The Genetic Algorithm is a Meta heuristic technique, which may be used to solve maximization optimization problem. The genetic algorithm works based on natural populations evolve according to the principle of natural selection that is survival of the fittest, first clearly stated by Charles Darwin in the Origin of Species. It starts with initial solution called Populations and it is filled with chromosome.

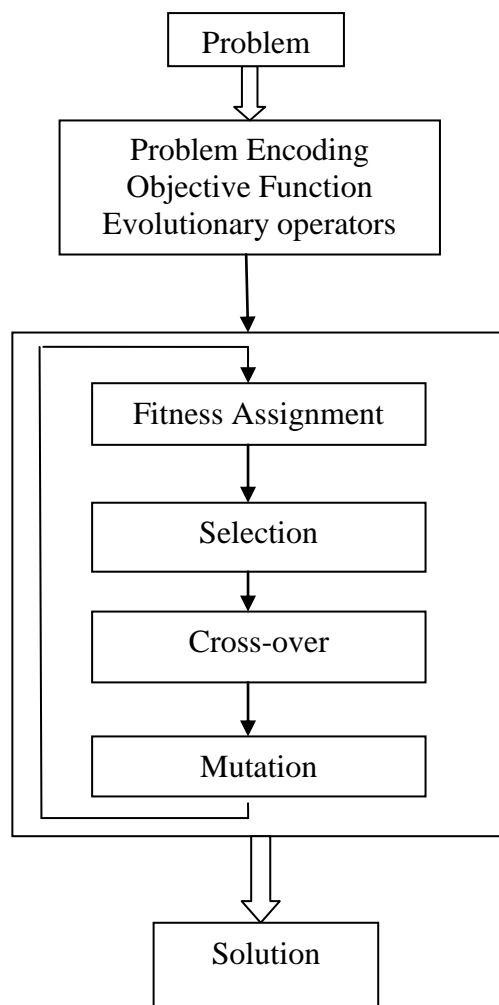


Figure 2. Simple Genetic Algorithm

Each element in chromosome is called gene. Job is represented by each gene in chromosome and the job sequence in a schedule based on the position of the gene. GA uses Crossover and Mutation operation to generate a new Population. A typical Genetic algorithm illustrated in Fig. 2.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

IV. PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization (PSO) is a population based search algorithm inspired by bird flocking and fish schooling originally designed and introduced by Kennedy and Eberhart in 1995. In contrast to evolutionary computation paradigms such as genetic algorithm, a swarm is similar to a population while a particle is similar to an individual through a multidimensional search space in which position of each particle is adjusted to its own experience and the experience of its neighbours.

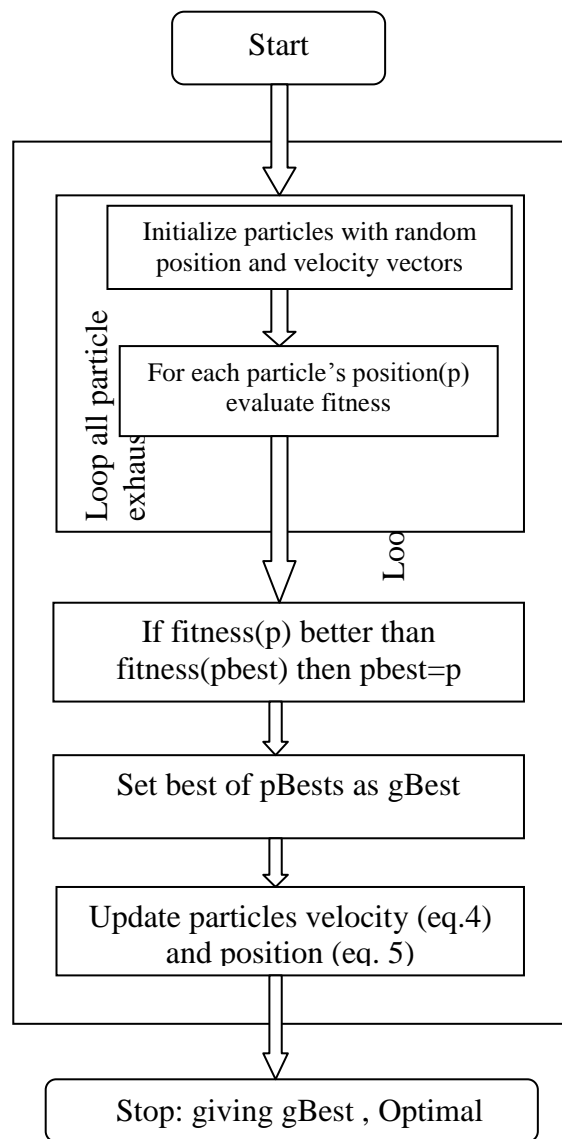


Figure 3. PSO Flow diagram

PSO System combines local search methods through self experience with global search methods through neighbouring experience, attempting to balance exploration and exploitation. The general flow diagram for PSO illustrated in Fig. 3.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

The mathematical model for calculating velocity vector and updating position as follows,

$$V_{i,d} = W * V_{i,d} + C_1 * r_1(P_{i,d} - X_{i,d}) + C_2 * r_2(P_{g,d} - X_{i,d}) \quad - (4)$$

$$X_{i,d} = X_{i,d} + V_{i,d} \quad - (5)$$

Where W is the inertial weight. It is used to control the amount of the previous velocity between the global exploration abilities of the swarm. C_1 and C_2 are two positive constants, and they represent the weight of the acceleration terms that pull each particle toward P_{id} and P_{gd} positions, r_1, r_2 are two random functions in the range $[0,1]$.

For Eq. (4), in traditional PSO, Update of the velocity consists of the following three parts. $W * V_{i,d}$ is referred to as “thrust” part which represents the influence of the last velocity towards the current velocity. $C_1 * r_1(P_{i,d} - X_{i,d})$ is the cognitive part, which represents the self thinking by itself. $C_2 * r_2(P_{g,d} - X_{i,d})$ is the social part, which represents the cooperation among particles.

The traditional PSO resolves the simple optimization concept that individuals are evolved by cooperation and competition among the individuals to accomplish a common goal. In large search process, each particle of the swarm shares the known information globally and benefits from discoveries and previous experiences of all other social group.

V. PROPOSED SCHEME

I. Genetic Algorithm Approach - Unordered subsequence exchange crossover (USXX) and shift change Mutation

A. Genetic Approach

The proposed scheme objective is to solve a job shop scheduling problem to minimize the makespan time. In order to solve a JSSP artificial intelligence technique **genetic algorithm (GA)** is used. The genetic algorithm is a probabilistic Meta heuristic technique, which is used to solve optimization problems. They are based on the genetic process of chromosome. Over many generations, natural population evolves according to the principles of natural selection that is survival of the fittest. It starts with the initial solution called population and it is filled with chromosome. Each element in chromosome is called gene. Job is represented by each gene in chromosome and the job sequence in a schedule based on the position of the gene. In our proposed algorithm **unordered subsequence exchange crossover (USXX)** and **shift change Mutation** is used.

B. Objective function

The main objective of the JSSP is to find a schedule of operations that can minimize the maximum completion time called makespan that is the completed time of carrying total operations out in the schedule for n jobs and m machines. The objective function takes the input as the number of jobs, number of operations, chromosome, operation time sequence and machine sequence of the corresponding operation sequence.

$$C_{max} = \max_{1 \leq i \leq n} C_i \rightarrow \min \quad (6)$$

C. Fitness Function

The objective is to minimize the makespan, so the following fitness function is applied,

$$f(y) = \frac{1}{f(y)} \quad (7)$$

D. Chromosome representation and encoding scheme

The searching space is created in $n \times m$ dimensions space for n jobs on m machines job shop scheduling problem. The problem solution is represented as a chromosome in Genetic Algorithm. The position of a chromosome consists of $n \times m$ dimensions and is represented with $n \times m$ real numbers. Chromosome is initialized based on jobs and machines. Then assign the rank with respect to smaller value in the chromosome and vice versa. Decoding by using the formula,

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

$$(R_i \bmod n) + 1 \tag{8}$$

Where,

R_i is integer number by ranking

n , is number of jobs

From the “Eq. (8)” proposed scheme will obtain the operation sequence. The following Table II shows representation of chromosome and generating the operation sequence.

TABLE II CHROMOSOME AND OPERATION SEQUENCE

| | | | | | | |
|------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| chromosome | 4.5 | 1.3 | 2.3 | 8.2 | 4.1 | 3.5 |
| Give the rank(R_i) | 5 | 1 | 2 | 6 | 4 | 3 |
| $(R_i \bmod n) + 1$ | 3 | 2 | 3 | 1 | 2 | 1 |
| Operation sequence | O ₃₁ | O ₂₁ | O ₃₂ | O ₁₁ | O ₂₂ | O ₁₁ |

From the Table II operation sequence defines 3rd job 1 operation, 2nd job 1st operation, 3rd job 2nd operation, 1st job 1st operation, 2nd job 2nd operation, 1st job 2nd operation. This operation Processed based on Process Routing that is Precedence constraints based. That means allocated jobs are proceed as per planned without changing the job sequence.

E. Algorithm Steps

- Step1 : Initialize the number of chromosomes by generating $n \times m$ real numbers for each chromosome.
- Step2 : Assign the operation time sequence and Machine sequence for selected chromosomes.
- Step3 : Find the makespan value for each and every chromosome using the objective function and also find the minimum makespan value among different values.
- Step4 : Select $N/2$ chromosomes using the Roulette- Wheel selection from different chromosomes.
- Step5 : Applying the Unordered subsequence exchange crossover and shift change Mutation to generate the new chromosomes.
- Step6 : Find the makespan values for newly generated chromosomes using the objective function.
- Step7 : Choose the best chromosomes which have the minimum makespan values from newly generated and also from old chromosomes.
- Step8 : Find the minimum makespan value among different chromosomes.
- Step9 : Terminates if the maximum number of iteration is reached or optimal value is obtained.

VI. PROPOSED HYBRID PSO-GA ALGORITHM

The idea of the proposed hybrid algorithm, called PSO-GA. The PSO possesses high search efficiency by combining local search and global search. By reasonably hybridizing these two methodologies, a hybrid PSO-GA algorithm model could be proposed to omit the concrete velocity-displacement updating method in traditional PSO for the JSS problem.

The Proposed hybrid algorithm includes two phases: (1) the initial solutions are randomly generated as per Table II. But assume like chromosome are now particles. (2) The PSO algorithm combined with GA algorithm is run. The General outline of the hybrid algorithm is summarized as follows

- Step 1: Generate initial population. If any of the generated schedules is infeasible, reconstruct them until all the initial solutions are feasible.
- Step 2: Evaluate each particles objective value in the swarm, and compare them, then set the pbest position with a copy of particle itself and gbest position the particle with the lowest fitness in the swarm.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

Step 3: if the termination criterion which is usually a sufficiently good fitness or a specified number of generations are not met, then go to step 4; otherwise go to step 7;

Step 4: Set iteration=iteration+1

Step 5: Apply Unordered Subsequence Exchange cross-over and Shift change Mutation.

Step 6: Convert current particles to a JSSP scheduling solution; then go to step 2.

Step 7: Computational results.

D. Unordered Subsequence Exchange Cross-over

Unordered subsequence exchange crossover creates a new children's even the subsequence of parent 1 is not in the same order in parent 2. the algorithm for USXX is as follows.

Step1: select the two parents from the different chromosomes initialized with the sequence of all operation. Say p1 and p2.

Step2: Generate two children from P1 and P2 name it as C1 and C2.

Step3: select the gene from P1 and same gene selected in P2 but it should be in unordered position in P2

Step4: crossover started from P1 that is P2 unselected genes are moved to the p1 in unselected position. So C1 is generated. Likewise to generate C2 crossover from P2 to P1.

E. Shift change mutation

Shift change mutation is implemented by shifting the same job index in every place into the same direction so that a new sequence is generated.

VI EXPERIMENTAL RESULTS

The Performance of the proposed Genetic algorithm and Hybrid PSO-GA for the JSSP is examined by using some 12 test problems taken from the OR-Library. Numerical experiments are performed in Java Language on a Workstation with Intel(R) Core (TM) i5 2.50GHz Processor and 2 GB memory.

Table IV summarizes the results of the experiments. The content of the table includes the name of the test problem(instance), The size of the problem, the best known value(BKV), The value of the best solution found by using Genetic Algorithm and Hybrid PSO-GA. Table III summarizes the parameters of the GA and PSO-GA.

TABLE III. PARAMETERS OF PROPOSED APPROACH

| PARAMETER NAME | VALUE |
|-----------------------------------|-------|
| SWARM SIZE | 20 |
| NUMBER OF PARTICLES | 100 |
| C ₁ AND C ₂ | 2.1 |
| CROSSOVER PROBABILITY | 0.7 |
| MUTATION PROBABILITY | 0.2 |
| NUMBER OF ITERATIONS | 1500 |

The Results of the proposed algorithms are compared with the six different benchmark results taken from OR-Library. Then the proposed hybrid algorithm is compared with HIA (2008) and PSO (2010).

From the Table IV tabulated results it is shown that for benchmarks LA09, LA11, LA12, LA13 the Hybrid PSO-GA yields better results than GA.

The HIA algorithm and PSO algorithm compared with our PSO-GA. The PSO-GA reaches the optimal solution with minimum iterations compared to the HIA and PSO. It is shown in Fig.4 and Fig.5.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

TABLE IV COMPARISONS OF RESULTS BETWEEN PSO-GA AND OTHER APPROACHES

| PROBLEM NAME | JOBS | MACHINES | MAKESPAN | | | | |
|--------------|------|----------|--------------|------|------|-------------------|-----------------------------|
| | | | OPTIMAL TIME | HIA | PSO | GENETIC ALGORITHM | PROPOSED ALGORITHM PSO + GA |
| LA09 | 15 | 5 | 951 | 951 | 951 | 956 | 951 |
| LA10 | 15 | 5 | 958 | 958 | 958 | 958 | 958 |
| LA11 | 20 | 5 | 1222 | 1222 | 1222 | 1267 | 1222 |
| LA12 | 20 | 5 | 1039 | 1039 | 1039 | 1071 | 1039 |
| LA13 | 20 | 5 | 1150 | 1150 | 1150 | 1188 | 1150 |
| LA14 | 20 | 5 | 1292 | 1292 | 1292 | 1292 | 1292 |

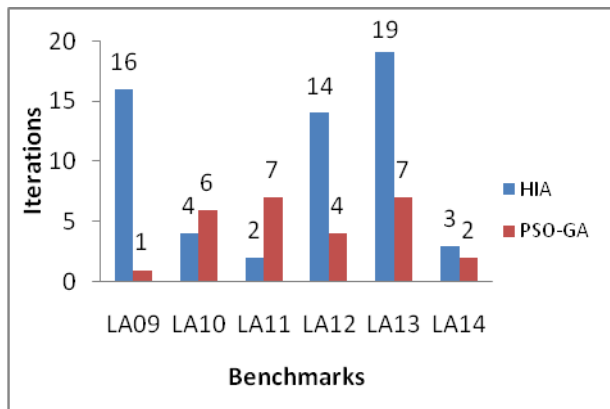


Figure 4. Comparison between HIA with Proposed PSO-GA

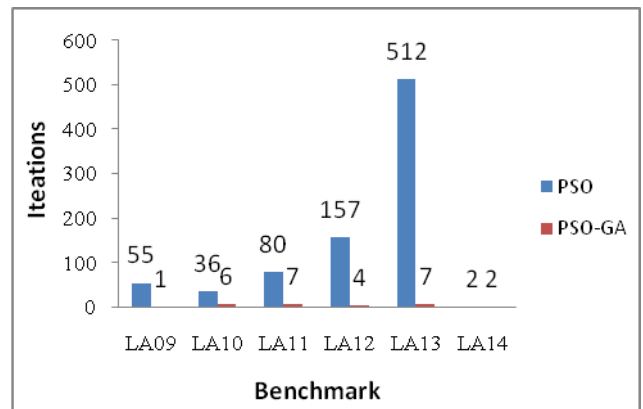


Figure 5. Comparison between PSO with Proposed PSO -GA

VII. CONCLUSION AND FUTURE WORK

In this paper, according to characteristics of static JSSP, a GA with Unordered subsequence exchange cross over and a Hybrid approach called PSO-GA is proposed to solve the JSSP. Two proposed algorithms are compared and evaluated. The best result is obtained by using Hybrid approach.

In future the proposed algorithm can solve a Job Shop Scheduling problem in dynamic environment with multi-criteria objective. Other hybridization heuristic methods are to be used to solve the Job-Shop Scheduling Problem

REFERENCES

- [1] R. Heilmann, "A branch-and-bound procedure for the multi- moderesource-constrained project scheduling problem with minimum and maximum time lags," *Eur. J. Oper. Res.*, vol. 144, no. 2, pp. 348–365, Jan. 2003.
- [2] C. Akkan and S. Karabati, "The two-machine flowshop total completion time problem: Improved lower bounds and a branch-and-bound algorithm," *Eur. J. Oper. Res.*, vol. 159, no. 2, pp. 420–429, Dec. 2004.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

- [3] T. Lorigeon, "A dynamic programming algorithm for scheduling jobs in a two-machine open shop with an availability constraint," *J. Oper. Res. Soc.*, vol. 53, no. 11, pp. 1239–1246, Nov. 2002.
- [4] C. N. Potts and L. N. Van Wassenhove, "Dynamic programming and decomposition approaches for the single machine total tardiness problem," *Eur. J. Oper. Res.*, vol. 32, no. 3, pp. 405–414, Dec. 1987.
- [5] Y. B. Canbolat and E. Gundogar, "Fuzzy priority rule for job shop scheduling," *J. Intell. Manuf.*, vol. 15, no. 4, pp. 527–533, Aug. 2004.
- [6] R. Klein, "Bidirectional planning: Improving priority rule-based heuristics for scheduling resource-constrained projects," *Eur. J. Oper. Res.*, vol. 127, no. 3, pp. 619–638, Dec. 2000.
- [7] J. B. Jensen, P. R. Philipoom, and M. K. Malhotra, "Evaluation of scheduling rules with commensurate customer priorities in job shops," *J. Oper. Manage.*, vol. 13, no. 3, pp. 213–228, Oct. 1995.
- [8] W. Q. Huang and A. H. Yin, "An improved shifting bottleneck procedure for the job shop scheduling problem," *Comput. Oper. Res.*, vol. 31, no. 12, pp. 2093–2110, Oct. 2004.
- [9] F. Pezzella and E. Merelli, "A tabu search method guided by shifting bottleneck for the job shop scheduling problem," *Eur. J. Oper. Res.*, vol. 120, no. 2, pp. 297–310, Jan. 2000.
- [10] H. X. Chen and P. B. Luh, "An alternative framework to Lagrangian relaxation approach for job shop scheduling," *Eur. J. Oper. Res.*, vol. 149, no. 3, pp. 499–512, Sep. 2003.
- [11] C. A. Kaskavelis and M. C. Caramanis, "Efficient Lagrangian relaxation algorithms for industry size job-shop scheduling problems," *IIE Trans.*, vol. 30, no. 11, pp. 1085–1097, Nov. 1998.
- [12] F. Geyik and I. H. Cedimoglu, "The strategies and parameters of tabu search for job-shop scheduling," *J. Intell. Manuf.*, vol. 15, no. 4, pp. 439–448, Aug. 2004.
- [13] J. P. Watson, J. C. Beck, and A. E. Howe, "Problem difficulty for tabu search in job-shop scheduling," *Artif. Intell.*, vol. 143, no. 2, pp. 189–217, Feb. 2003.
- [14] S. G. Ponnambalam, P. Aravindan, and S. V. Rajesh, "A tabu search algorithm for job shop scheduling," *Int. J. Adv. Manuf. Technol.*, vol. 16, no. 10, pp. 765–771, Oct. 2000.
- [15] T. K. Liu, J. T. Tsai, and J. H. Chou, "Improved genetic algorithm for the job-shop scheduling problem," *Int. J. Adv. Manuf. Technol.*, vol. 27, no. 9/10, pp. 1021–1029, Feb. 2006.
- [16] S. Hajri, N. Liouane, S. Hammadi, and P. Borne, "A controlled genetic algorithm by fuzzy logic and belief functions for job-shop scheduling," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 5, pp. 812–818, Oct. 2000.
- [17] X. Gang and Z. M. Wu, "Deadlock-free scheduling strategy for automated production cell," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 34, no. 1, pp. 113–122, Jan. 2004.
- [18] K. S. Amirthagadeswaran and V. P. Arunachalam, "Improved solutions for job shop scheduling problems through genetic algorithm with a different method of schedule deduction," *Int. J. Adv. Manuf. Technol.*, vol. 28, no. 5/6, pp. 532–540, Mar. 2006.
- [19] C. Low, J. Y. Yeh, and K. I. Huang, "A robust simulated annealing heuristic for flow shop scheduling problems," *Int. J. Adv. Manuf. Technol.*, vol. 23, no. 9/10, pp. 762–767, May 2004.
- [20] T. Y. Wang and K. B. Wu, "A revised simulated annealing algorithm for obtaining the minimum total tardiness in job shop scheduling problems," *Int. J. Syst. Sci.*, vol. 31, no. 4, pp. 537–542, Apr. 2000.
- [21] M. Kolonko, "Some new results on simulated annealing applied to the job shop scheduling problem," *Eur. J. Oper. Res.*, vol. 113, no. 1, pp. 123–136, Feb. 1999.
- [22] R. K. Suresh and K. M. Mohanasundaram, "Pareto archived simulated annealing for job shop scheduling with multiple objectives," *Int. J. Adv. Manuf. Technol.*, vol. 29, no. 1/2, pp. 184–196, Jan. 2006.
- [23] J. Zhang, X. M. Hu, X. Tan, J. H. Zhong, and Q. Huang, "Implementation of an ant colony optimization technique for job shop scheduling problem," *Trans. Inst. Meas. Control*, vol. 28, no. 1, pp. 93–108, Jan. 2006.
- [24] Z. G. Lian, B. Jiao, and X. S. Gu, "A similar particle swarm optimization algorithm for job-shop scheduling to minimize makespan," *Appl. Math. Comput.*, vol. 183, no. 2, pp. 1008–1017, Dec. 2006.
- [25] W. J. Xia and Z. M. Wu, "A hybrid particle swarm optimization approach for the job-shop scheduling problem," *Int. J. Adv. Manuf. Technol.*, vol. 29, no. 3/4, pp. 360–366, Jun. 2006.
- [26] H. B. Yu and W. Liang, "Neural network and genetic algorithm-based hybrid approach to expanded job-shop scheduling," *Comput. Ind. Eng.*, vol. 39, no. 3/4, pp. 337–356, Apr. 2001.
- [27] S. Yang and D. Wang, "A new adaptive neural network and heuristics hybrid approach for job-shop scheduling," *Comput. Oper. Res.*, vol. 28, no. 10, pp. 955–971, Sep. 2001.
- [28] S. Yang and D. Wang, "Constraint satisfaction adaptive neural network and heuristics combined approaches for generalized job-shop scheduling," *IEEE Trans. Neural Netw.*, vol. 11, no. 2, pp. 474–486, Mar. 2000.
- [29] Jianchao Tang, Guoji Zhang and Binbin Lin, Bixi Zhang, "A Hybrid Algorithm for Flexible Job Shop Scheduling Problem," published by Elsevier Ltd. Doi:10.1016/j.proeng.2011.08.689.
- [30] Jianguo Jiang, Mingxing Wen, Kaige Ma, Junzheng Li, "Hybrid Genetic Algorithm for Flexible Job Shop Scheduling with Multi-objective," *Journal of Information & Computational Science.*, 8: 11 (2011), 2197-2205.
- [31] A. Tamilarasi and T. Anantha Kumar, "An enhanced genetic algorithm with simulated annealing for job shop scheduling," *international journal of Engineering, Science and Technology*, vol. 2, no. 1, 2010, pp. 144-151.
- [32] R. Thamilselvan and balasubramanie, "Analysis of Various Alternate crossover Strategies for Genetic Algorithm to Solve Job Shop Scheduling Problems," *European Journal of Scientific Research*, vol. 64, no. 4, pp. 538-554, 2011.
- [33] Sureka P and Dr.S.Sumathi, "PSO and ACO based approach for solving combinatorial Fuzzy Job Shop Scheduling Problem," *int. j. Comp. Tech. Appl.*, Vol 2(1), 112-120.
- [34] A. Jamili, M.A. Shafia and R. Tavakkoli-Moghaddam, "A Hybrid algorithm based on particle swarm optimization and simulated annealing for a periodic job shop scheduling problem," *int. J. Adv Manuf. Technol.*, DOI: 10, 1007/s00170-010-2932-8, 2010, springer-verlag London Limited.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

- [35] Hong-Wei Ge, Liang Sun, Yan-Chun Liang, and Feng Qian, "An Effective PSO and AIS-Based Hybrid Intelligent Algorithm for Job-Shop Scheduling," IEEE transactions on systems, man, and cybernetics—part a: systems and humans, vol. 38, no. 2, march 2008.
- [36] Guohui Zhang, Xinyu Shao, Peigen Li and Lian Gao, "An Effective hybrid particle swarm optimization algorithm for multi-objective flexible job shop scheduling problem," computers and industrial Engineering 56, 2009.
- [37] Shruti Kapoor, Swathi Singh, Shelly Chikara, Shivangi Garg and Viji Singh, "Ways and Means of Applying Genetic Algorithms for Job Shop Scheduling," International Journal of Computational Intelligence Research, vol. 8, pp. 27-33, 2012.
- [38] Ye Li, Yan Chen, "Hybrid Algorithm Approach To Job Shop Scheduling Problem," Global Journal of Computer Science and Technology. Vol. 10, issue 8, 2010.
- [39] Thongchai Pratchayaborirak, Voratas Kachitvichyanukul, "A two-stage PSO algorithm for job shop scheduling problem" international journal of Management Science and Engineering management, 6(2): 84-93, 2011.
- [40] Zohreh Davarzani, Mohammed-R Akbarzadeh-T and Nima Khairdoost, "Multiobjective Artificial Immune Algorithm for Flexible Job Shop Scheduling Problem," International journal of Hybrid Technology, vol. 5 no. 3, july. 2012.