Application of Genetic Algorithm on Job Shop Scheduling Problem to Minimise Makespan

Anshulika[1], L. A. Bewoor[2]

[1]Post Graduate student, Department of Computer Engineering, VIIT-Pune, Pune, India

[2]Assistant Professor, Department of Computer Engineering, VIIT-Pune, Pune, India

Abstract: ***Scheduling of large number of jobs/tasks is a tedious and time taking work. With the increase is demand of products, the manufacturing industries have been facing a lot of trouble in fulfilling those demands while optimizing the production. Job shop scheduling problem (JSSP) is a well known combinatorial optimization problem with NP hard difficulty. Job shop scheduling (JSS) is the efficient allocation of shared resources (M) to competing jobs (J) such that a specific optimization criterion is satisfied. The complexity of JSS is (J!)^M, which makes it NH hard. Various techniques have been used to solve the JSS problem till date. Metaheuristic techniques like Genetic Algorithm (GA) have shown good results and have been proven to be better performers than other techniques.***

Keywords: Job shop scheduling (JSS); Genetic Algorithm (GA); metaheuristic; optimization

1. Introduction

There are mainly two types of Scheduling environments: (1) Flow shop, (2) Job shop. In Flow shop, all the jobs pass through all the machines in the same order whereas, in Job shop, the machine order can be different for each job [2]. Job shop scheduling has been an interesting NP hard research problem since there are no exact algorithms that can solve JSS problem consistently even for small problem size.

The methods that have been used to solve optimization problems are:

A. Exact method:

They provide optimum only for any particular instance of a problem. Dynamic programming and branch ‘n bound are examples of exact algorithms that haven‘t been consistently successful in solving JSS problem and have also been very time consuming.

B. Approximate method

A certain quality of result is provided by approximate method algorithms for any instance of the problem. This implies that we know the distance from this result to the optimum result.

C. Heuristic method:

It provides a ―good enough” result for many instances of the problem. Heuristics have significant advantages such as (1) work well with dynamic problem sizes, (2) less computation time (3) can be easily combined with other methods. But, if the problem size goes on increasing, heuristics fail to give optimal results. The problem size is a major lacking factor for heuristic techniques.

D. Metaheuristic method:

Meta (beyond) heuristic (to find) methods can be applied to almost all combinatorial optimization problems. They are upper-level general methodologies that are used to guide in designing the underlying heuristics. Metaheuristics are broadly classified as constructive approach where each step take presiding output as the input and construct new sequence of output; local search approach that acquire local optimal solution to find global solution; and evolutionary approach evolves more optimal result than previous iteration to improve global solution[3].

2. Job Shop Scheduling

In the general job shop scheduling problem J jobs have to be processed on M machines.

Table I represents the general JSS problem. The general objectives undertaken during researches are to reduce makespan, tardiness, mean flow time; makespan being the main focus of most of the researchers. The minimization of these criteria can result in a near optimum result of the Job shop scheduling problem.

Table 1: Job shop scheduling

J	No. of jobs
M	No. of machines
P_j	Total processing time of job j
P_{jm}	Processing time of job j on machine m
d_j	Due date of job j
C_j	Completion time of job j
$L_j = C_j - d_j$	Lateness of job j
$T_j = \max(L_j)$	Tardiness of job j
$C_{max} = \max(C_j)$	Makespan
$F = \Sigma C_j / J$	Mean flow time

In JSS since the machine and the job sequences can be random so with the increase in number of jobs (J) or machines (M), the possible combinations increase with respect to (J!)^M. above a problem size of 3x3, the complexity becomes NP hard. Table I represents the general JSS problem.

3. Genetic Algorithm

Genetic algorithms belong to the class of evolutionary algorithms that are based on the principles of natural evolution (Fig.1). Evolutionary algorithms are designed with reference to the behavior of a population of individuals. It takes into considerations the biological model of evolution and natural selection [4].

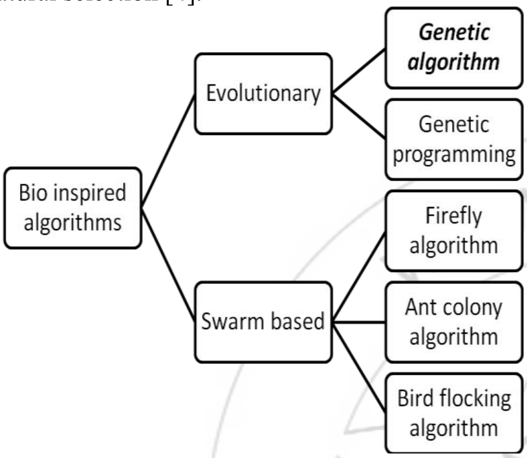

Figure 1: Bio inspired algorithms

In GA a solution to the problem is represented as a genome (Chromosome). A population of solutions is created using GA and then GA operators are applied to the population to reach to the optimal solution.

GA operators are as follows:

a) Crossover

It is a genetic operator that is used to vary the programming of a chromosome. It is analogous to biological reproduction. It takes more than one parent solutions and produces a child solution.

b) Selection

The selection method chooses those individuals from the population that will participate in producing the child solution.

c) Replication

It is the process of producing two identical replicas from the original individual.

d) Mutation

It is the genetic operator that is used to maintain the genetic diversity from one generation to the next. It alters one or more gene values in a chromosome from its original value [2]. The flow chart in figure 2 depicts the GA steps starting from initialization of population and iterating till stopping criteria is met.

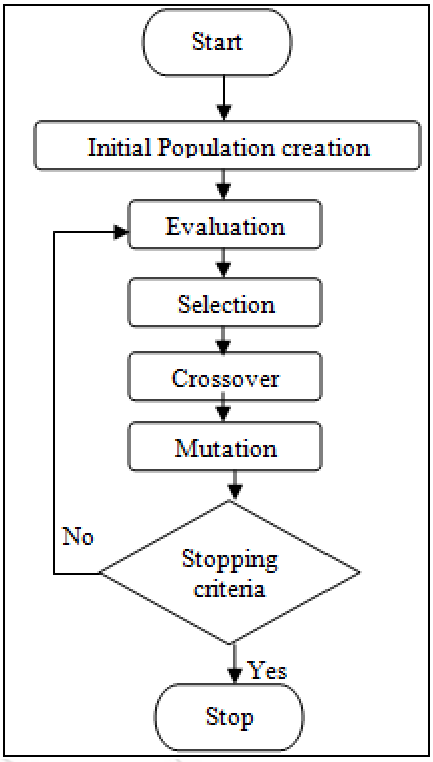

Figure 2: General flow chart for Genetic algorithm

4. Related Work

In the job shop scheduling problem, a finite number of jobs are to be processed by a finite number of machines. Each job consists of a predetermined sequence of operations, which will be processed without interruption for a period of time on each machine. As problem size increases, performance decreases. Therefore metaheuristic techniques are used to find a schedule which minimizes the makespan, total flow time and tardiness.

Shantanu Kolharkar and D.R. Zanwar [2] have mainly focused on the minimization of makespan and prioritization of jobs using due dates and fitness function. They have aimed at the analysis of the JSS problem at Job shop Process Industry, considering static scheduling and due dates as dynamic scheduling. They have also compared the results with and without using GA parameters.

Frank Werner [4] discusses the representation of solution, initial population generation, and application of different GA operators on simple and hybrid shop scheduling problems. He also discusses different GAs and compares their results. He briefly introduces ‗LiSA-A Library of Scheduling Algorithms' developed at Otto-von-Guericke-University Madeburg for shop scheduling problems, which includes a GA.

A case study has been done by Meilinda F. N. Maghfiroh, Agus Darmawan, and Vincent F. Yu in [5]. They have proposed GA with modifications and aim at the minimization of makespan and mean flow time. The maximum number of generation (G) is selected as the stopping criteria. Longest processing time (LPT) rule, shortest processing time (SPT) and first come first serve (FCFS) rules are chosen as the benchmark rules for the experiment. They proposed to induce variation in the choice of stopping criteria.

A hybrid GA with simulated annealing has been studied by A. Tamilarasi and T. Anantha kumar [6]. They consider the minimization of makespan as their objective of the study and show that the result escapes from the local minimums. They have used 21 instances from the OR-Library as benchmarks to test their proposed algorithm.

The makespan minimization problem in static job shop has been studied by Viswanath Kumar Ganesan, Appa Iyer Sivakumar, G. Srinivasan [7]. They have presented a lower bound on the makespan subject to minimum completion time variance (CTV).

$$CTV = \frac{1}{n} \times \sum_{i=1}^{1} (C_i - C_{mean})^2$$

Where, C_i is the completion time of job i, and C_{mean} is the mean completion time.

$$C_{mean} = \frac{1}{n} \times \sum_{i=1}^{n} C_i$$

Their study concluded that the use of backward scheduling heuristic using simulated annealing algorithm performs better than the one using forward dispatch method.

Shyh-Chang Lin, Erik D. Goodman William F. Punch, III [8] have worked with the dynamic job shop scheduling problem. They found a better approach than the priority rule approaches. For the deterministic problems: (1) Single-population GA (SGA) with population size 50, (2) Parallel GA (PGA) in which 25 SGAs with subpopulation sizes of 20 were connected in a 5x5 torus. The migration interval was 50 generations. The number of generations of both versions was 50x (number of jobs). For stochastic problem, number of machine was 5 with jobs arriving continually according to a Poisson process. The process is observed until the completion of 100 jobs. This approach outperformed the priority rules with respect to the machine workload, imbalance of machine workload, and due date tightness.

Takeshi Yamada and Ryohei Nakano [9] had proposed the first important experiment on the application of GA. A bit string representation with the GA operators has been studied by them. For larger-size problems, local search methods with domain specific knowledge were used. They proposed a local search method called multi-step crossover function (MSXF) which outperformed other GA methods.

Khaled Mesghouni, Slim Hammadi, and Pierre Borne [10] have presented their study on the use of Evolutionary algorithm on flexible JSS problem in order to minimize makespan. They have used a problem size of 10x10 and applied parallel machine encoding and paralle job encoding on the problem. A fixed population size with different values of crossover and mutation rates are used in the study.

Liang Sun, Xiaochun Cheng, Yanchun Liang [11] have proposed a clonal selection based hyper mutation and a life span extended strategy [12]. An adaptive penalty function is used that helps the algorithm to search in both feasible and infeasible regions of the solution space. 23 benchmark instances were taken from the OR-library for simulation.

The problem of deteriorating jobs (processing time is an increasing function of starting time) has been studied by Gur Mosheiov [13]. It has been proved that JSS problem is NP even with 2 machines. The proof has been shown by reduction to the Subset Product problem [14, 15].

5. Conclusion

A survey on the application of GA on JSS problem has been presented in this paper. To improve the optimality of the solution, the traditional metaheuristic algorithms should be combined with each other. Metaheuristic techniques are used to find a schedule which minimizes the makespan, total flow time and tardiness.

As a future work, different combination of metaheuristic techniques can be used to solve combinatorial optimization problems.

6. Acknowledgment

The support and guidance of my guide Prof. Mrs. L.A. Bewoor is highly appreciated.

References

[1] P.K. Gupta, D.S. Hira, *"Operation Research"*, pp. 404-406,S. Chand,1983.
[2] Shantanu Kolharkar, Prof. D. R. Zanwar, ―Scheduling in Job Shop Process Industry", *IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE),2013.*
[3] M. Widmer, *"Metaheuristics and Scheduling"*, pp. 33–68, Jan. 2008.
[4] Werner, Frank. "A survey of genetic algorithms for shop scheduling problems." P. Siarry: Heuristics: *Theory and Applications, Nova Science Publishers (2013): 161-222.*
[5] Meilinda F. N. Maghfiroh, Agus Darmawan, Vincent F. Yu, ―Genetic Algorithm for Job Shop Scheduling Problem: A Case Study", *International Journal of Innovation, Management and Technology, 2013.*
[6] Tamilarasi, T. Anantha kumar, ―An enhanced genetic algorithm with simulated annealing for job-shop scheduling", *International Journal of Engineering, Science and Technology, 2010.*
[7] Ganesan, Viswanath Kumar, Appa Iyer Sivakumar, and G. Srinivasan. "Hierarchical minimization of completion time variance and makespan in jobshops." *Computers & Operations Research* 33.5 (2006): 1345-1367.
[8] Lin, Shyh-Chang, Erik D. Goodman, and William F. Punch III. "A Genetic Algorithm Approach to Dynamic Job Shop Scheduling Problem." *ICGA*. 1997.
[9] Yamada, Takeshi, and Ryohei Nakano. "Genetic algorithms for job-shop scheduling problems." *Proceedings of the Modern Heuristics for Decision Support*. 1997.

[10] Mesghouni, Khaled, Slim Hammadi, and Pierre Borne. "Evolutionary algorithms for job-shop scheduling." *International Journal of Applied Mathematics and Computer Science* 14.1 (2004): 91-104.

[11] Liang Sun, Xiaochun Cheng, Yanchun Liang, ―Solving Job Shop Scheduling Problem Using Genetic Algorithm with Penalty Function", *International Journal of Intelligent Information Processing Volume 1, Number 2,* December 2010.

[12] Timmis, Jon. "Artificial immune systems—today and tomorrow." *Natural computing* 6.1 (2007): 1-18.

[13] G. Mosheiov, *"Complexity analysis of job-shop scheduling with deteriorating jobs,"* Elsevier Science, Discrete Applied Mathematics , vol. 117, pp. 195–209, 2002.x995gw.

[14] Garey, Michael R., and David S. Johnson. *Computers and intractability*. Vol. 29. wh freeman, 2002.

[15] E. Horowitz, S. Sahani, ―*Fundamentals of computer algorithm*", Galgotia, 2006, pp.536-539.

[16] M. Emin Aydin, Terence C. Fogarty, ―Teams of autonomous agents for job-shop scheduling problems: An experimental study", *Journal of Intelligent Manufacturing, 2004.*

[17] F. Zhao, J. Tang, and J. Wang, ―An improved particle swarm optimization with decline disturbance index (DDPSO) for multi-objective job-shop scheduling problem," *Comput. Oper. Res.*, vol. 45, pp. 38–50, May 2014.

[18] N. Shahsavari-Pour and B. Ghasemishabankareh, ―A novel hybrid meta-heuristic algorithm for solving multi objective flexible job shop scheduling," *J. Manuf. Syst.*, vol. 32, no. 4, pp. 771–780, Oct. 2013.

[19] J. Li, Q. Pan, and Y.-C. Liang, ―An effective hybrid tabu search algorithm for multi-objective flexible job-shop scheduling problems", *Comput. Ind. Eng.*, vol. 59, no. 4, pp. 647–662, Nov. 2010.

[20] Jose M. Framinana, Marcelo S. Naganob*," Evaluating the performance for makespan minimisation in no-wait flowshop sequencing",* J of Materials Processing Tech. Vol. 197 ,F eb. 2008.