

An Evolutionary Algorithm Applied for Optimisation of Single Objective (makespan minimisation) Job Shop Scheduling Problems.

K.C. Udaiyakumar, Dr. M.Chandrasekaran.

Abstract— Job shop scheduling problem is well known common production problems which consists of 'n'-Jobs and 'm'-Machine. Many researchers applied *varied* approaches with existing heuristics for small size problems with near optimal solutions. This paper presents the criterion of minimisation of makespan for the different size Job Shop Scheduling problems. The proposed computational method of selective breeding Algorithm is used for finding optimal makespan values of different size problems. This algorithm is tested with 10 benchmark problems. The result shows that the proposed algorithm is a most efficient and effective algorithm which produced better result than other algorithm compared in literature. Hence the proposed algorithm is a good tool for problem solving technique for such kind of scheduling problems.

Index Terms— Benchmark problems, selective breeding algorithm, makespan and minimisation

1 INTRODUCTION

The word scheduling in manufacturing system is used to determine the sequence in which the jobs are to be processed over the production stages, at the same time to determine the start and end time of processing of part or jobs, in order to meet an objective or set of objective, scheduling plays a vital role to increase the productivity and efficiency of the manufacturing system. Because of the complexity this type of problem are known as non-polynomial (NP) hard problems. Scheduling can be classified into three categories, (i) Single machine scheduling, (ii) Flow shop scheduling and (iii) Job shop scheduling. Single machine shop scheduling problems deals with 'n' jobs with same single operation on each of the jobs, the flow shop scheduling problem comprises 'n' jobs with 'm' operation on each of the jobs. In this problem all the jobs have to follow the same process sequences. The job shop scheduling problem comprises 'n' jobs with 'm' machines on each of the jobs whereas process sequence of the jobs will be different from each other.

Job shop scheduling problem is one of the gargantuan combinatorial problems. Scheduling is the allocation of resources over period of time to complete a collection of task. The job shop scheduling problem is comprised by a set of 'm' machines $\{M_1, M_2, \dots, M_n\}$ and collection of 'n' jobs $\{J_1, J_2, \dots, J_n\}$ to be scheduled, where each job must pass through each machine once only. Each job has its own processing order.

Metaheuristic algorithms[5,6] becomes a vital part of con-

ligence and soft computing. In the last three decades, a large number of algorithms were developed such as genetic algorithm, ant colony algorithm, simulated annealing, particle swarm optimisation, differential evaluation, bat algorithm and selective breeding algorithm. In which selective breeding algorithm have appeared and proved great potential in solving tough engineering optimisation. In this work, the minimisation of makespan is considered as objective. In this paper explanation of selective breeding algorithm serves as an efficient tool for solving any kind of job shop scheduling problems.

2 LITERATURE REVIEW

Many optimisation techniques have been applied to solve the job shop scheduling problem. In the beginning traditional method based on mathematical approach or numerical search such as branch and bound technique and Lagrangian approach[2,8] both these methods can assured optimum solution for small size job shop scheduling problems most effectively and efficiently. Whereas for a moderate and large size JSS problems this traditional methods find difficulties and involves high computational time with available resources. In the reason past a larger size of JSS problems have been solved by meta heuristics such as Tabu search[4], simulated annealing[3], artificial immune algorithm[7], selective breeding algorithm[9] and so on. These methods usually follow stochastic steps in their search process.

3 SINGLE OBJECTIVE JOB SHOP SCHEDULING

Single objective Job Shop model consists of minimization of makespan as an objective function which is given in equation 1, Sequence and resource constraints are shown in equations 2,3 and 4 respectively. The makespan means the completion time of all the jobs which are con-

- Author name is currently pursuing masters degree program in electric power engineering in University, Country, PH-01123456789. E-mail: author_name@mail.com
- Co-Author name is currently pursuing masters degree program in electric power engineering in University, Country, PH-01123456789. E-mail: author_name@mail.com
(This information is optional; change it according to your need.)

-temporary global optimisation algorithm, computational intel-

sidered for scheduling. The Job shop $i(i=1,2,3,\dots,n)$ requires processing by machine $k(k=1,2,\dots,m)$ exactly once in its operation sequence. Let p_{ik} is the processing time of job i on machine k , X_{ik} is the starting time of job i on machine k , q_{ijk} is the indicator which takes on a value of 1 if operation j of job i requires machine k , and zero otherwise. Y_{ihk} is the variable which takes on a value of 1 if job i precedes job h on machine k , and zero otherwise.

The objective function is minimization of makespan

$$\text{Minimize } Z = \sum_{k=1}^m q_{imk}(X_{ik} + p_{ik}) \dots \dots \dots (1)$$

Subject to

a) Sequence constraint

i.e. for a given job i , the $(j+1)$ st operation may not start before the j th operation is completed.

b) Resource constraint

$$X_{hk} - X_{ik} \geq p_{ik} - (H + p_{ik})(1 - Y_{ihk}) \dots \dots \dots (3)$$

$$X_{ik} - X_{hk} \geq p_{hk} - (H + p_{hk})Y_{ihk} \dots \dots \dots (4)$$

Where $(i=1,\dots,n; h=1,\dots,n; k=1,\dots,m)$ where H is a very large positive integer.

3.1 Application of Selective Breeding Algorithm For Job Shop scheduling.

The steps involved in selective breeding algorithm are shown in fig - 1.

This can be explained with numerical illustration of scheduling problem. By considering any one of the bench mark problem..For example

LA16 Benchmark problem [20] which consists
 Number of machines: 10 Number of jobs: 10

Step 1: Generate Initial population (10 job sequences)

Each seed is operated 10 times for achieving one complete job sequence

- 1-10-6-7-5-8-9-3-2-4
- 3-9-1-6-10-8-5-7-2-4
- 3-1-10-8-5-6-9-7-2-4
- 6-5-3-9-7-1-2-8-10-4
- 3-9-2-7-5-10-6-8-1-4
- 3-1-2-7-9-10-8-6-5-4
- 1-4-5-10-8-6-9-3-2-7
- 9-6-1-4-7-3-10-2-8-5
- 9-4-6-8-10-3-7-1-5-2
- 9-10-6-3-7-1-2-4-5-8

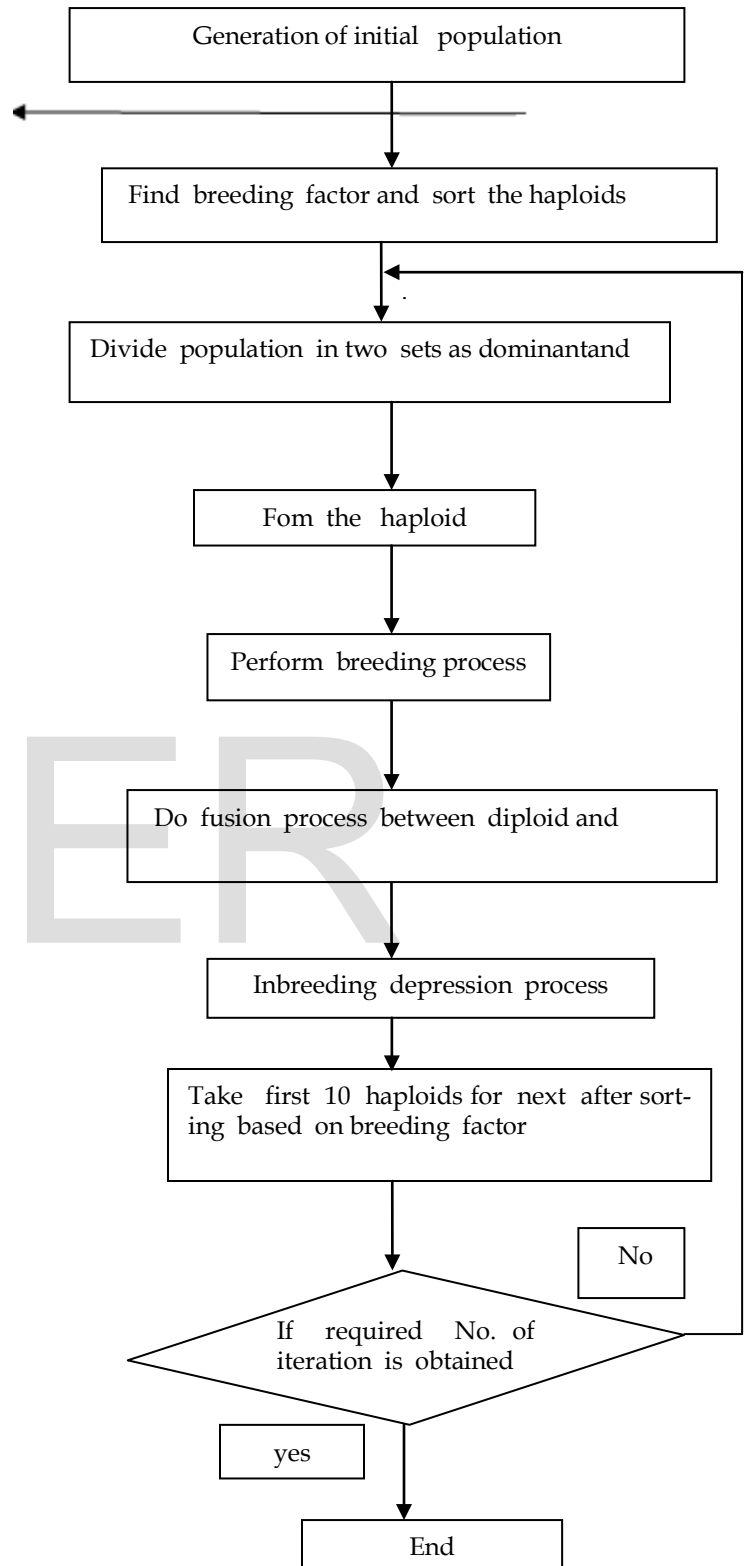


Figure 1 : Flow chart of Selective breeding Algorithm

Step 2: Determination of makespan and breeding factor for each sequence.

Firstly makespan for each sequence is to be calculated to find the breeding factor. Breeding factor is the reciprocal of makespan.

i.e Breeding factor = 1/makespan, then arrange the sequence in descending order based on its breeding factor to form population set.

Step 3: Divide the population set into exactly two sets and form diploid cells

Ten population set is formed which is divided into two equal sets naming Dominant set (R) and recession set (r). In dominant set R contains 5 job sequences namely R1, R2, R3, R4 and R5. In recession set r contains 5 job sequences namely r1, r2, r3, r4 and r5. Then form the diploid set such as {R1r1, R2r2, R3r3, R4r4 and R5r5}.

Step 4: Breeding process

Possibility breeds for one set combination by considering two diploids namely R1r1 and R2r2, and proceed with R2r2 and R3r3 and so on.

Table - 1 Explanation of Breeding process

	R2	r2
R1	R1R2	R1r2
r1	r1R2	r1r2

Finding all the possible breeds (diploid set) R1R2, R1r2, r1R2 and r1r2 similarly form diploid sets for all possible combinations.

Step 5: Fusion process

Fusion process for all the possible diploid combinations are carried out and separated as haploid.

To determine number of fusion points is equal to half of the length of the haploid. Fusion points can be selected randomly. At fusion point interchange position between parents. For example one set of diploid

Before fusion original sequence are

1-10-6-7-5-8-9-3-2-4

3-9-1-6-10-8-5-7-2-4

Fusion points are chosen randomly at **1,3,5,7 and 9.**

After fusion the sequence become

3-10-1-7-10-8-5-3-2-4

1-9-6-6-5-8-9-7-2-4

Step 6: Inbreeding depression process

In selective breeding any particular genes may have tendency of losing some of the other genes from the gene pool altogether which is irreversible, this process is called inbreeding depression. To overcome this, 10% newly generated haploids are to be added in each iteration this paves way for a chance to search solution in the new search space.

Step 7: Sort the final sequence based on breeding factor

After completion of 100 iteration, the result is 10-5-9-3-4-6-8-2-1-7 and its optimum makespan value is 945

4 EXPERIMENTAL RESULTS AND ANALYSIS

The selective breeding algorithm is applied using C language

on PC Pentium4 2.4 Mhz. The maximum number of iteration have been chosen to 100 *n, where n represents number of jobs. This algorithm has been tested for ten problem instances of various sizes collected from the OR library[1]. Two instances denoted as (TA01 and TA02) and one instance from ORB2 and ABZ5 and remaining six instances of Lawrance (LA01, LA06, LA11, LA16, LA20 and LA25). From the table 2 and Fig. 2&3. It is observed that solutions obtained by the selective breeding algorithm is compared with previous researcher's solution. Out of ten solutions, nine solutions exactly matched with optimum value and one solution is near to optimal. The relative error was calculated for all the problem instances using the formula

Percentage relative error = 100* (upper bound value-lower bound value)/lower bound value.

The mean relative error is almost zero percent in selective breeding algorithm when compared to artificial immune system (AIS) [7] shown in table

Table 2. Comparison of SBA results with Artificial immune system

Problem	Jobs (n)	Machines (m)	Optimal value (makespan)	Makespan by SBA	% RE by SBA	Makespan by AIS	% RE by AIS
LA01	10	5	666	666	0	666	0
LA06	15	5	932	932	0	932	0
LA11	20	5	1222	1222	0	1222	0
LA16	10	10	945	945	0	945	0
LA20	10	10	902	902	0	902	0
LA25	15	10	977	978	0.1	977	0
ORB2	10	10	888	888	0	891	0.34
ORB5	10	10	887	887	0	888	0.11
ABZ5	10	10	1234	1234	0	1234	0
ABZ7	20	15	661	661	0	666	1.52
Percentage mean relative error (% MRE)					0.01		0.197

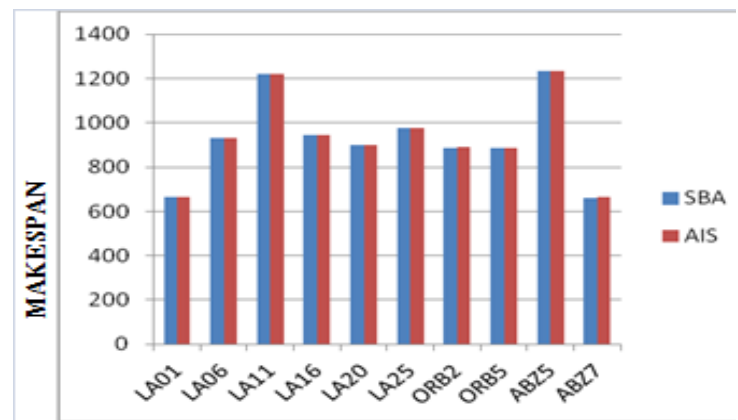


Figure 1. Comparison of SBA results (Makespan) with AIS

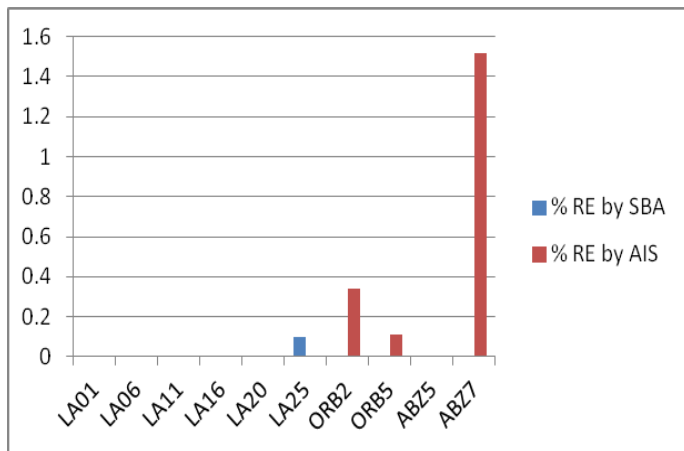


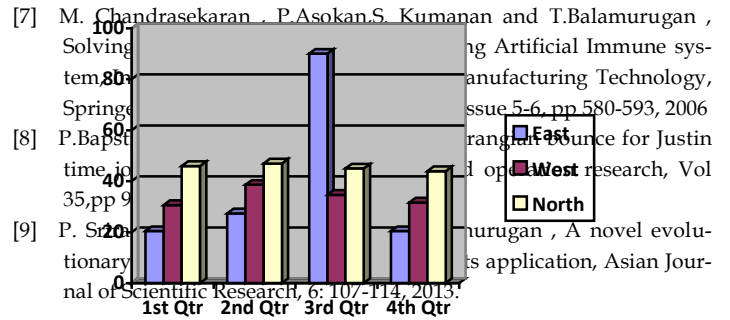
Figure 2. Comparison of % Relative Error SBA with AIS

5 CONCLUSION

Thus, evolutionary metaheuristic has gained popularity owing to the ability of dealing with a non linear global optimisation problems. In this paper the proposed selective breeding algorithm has been applied for solving Job Shop Scheduling problems with the objective of makespan minimisation, this algorithm when tested on . From this, it is observed that, this algorithm found optimal results for almost all the tested problem instances(ten benchmark problem) proves to be effective and efficient. From this, it is observed that, this algorithm found optimal results for almost all the tested problem instances. There is no doubt this algorithm will be applied in solving more challenging problem in near future. Further more it is observed that theoretical understandings are still lagging behind. In fact there is a huge gap between theoretical approach and practical application. Most of the metaheuristic algorithm require good modification in order to solve combinatorial optimization properly. Therefore, it is observed from the literature, most of the metaheuristic research has focused on small scale problems, it will be more helpful if the researcher's focus on large scale- real world applications.

References

- [1] J.E.Beasley, OR library,distributing test problems by electronic mail, Journal of the operation research society, Vol 41, pp1069-1072, 1990
- [2] Christos A.kaskavelis and Michael C.Caramanis, Efficient Lagrangian relaxation algorithms for industry size job shop scheduling programs, IIE transactions, Vol 30, pp 1085-1097, 1998
- [3] S.G.Ponnambalam., N.Jawahar and P.Aravindan, A simulated Annealing algorithm for job shop scheduling, production planning and control, vol 10(8), pp 767-776, 1999
- [4] S.G.Ponnambalam, P.Aravindan and S.V.Rajesh, A Tabu search algorithm for job shop scheduling
- [5] S.Binato, W.J.Hery, Loewenstern and M.G.C. Resende, A Grasp for job shop scheduling, AT & T Labs research Technical report: 00.6.1, 2000.
- [6]Khaled Mesghouni, Silm Hammadi and Pierre Borne, Evolutionary algorithms for job shop scheduling, International journal of application of mathematics and computer science, Vol 14, pp 91-103, 2004



- [7] M. Chandrasekaran, P.Asokan,S. Kumanan and T.Balamurugan, Solving Job Shop Scheduling Problem Using Artificial Immune system, International Journal of Manufacturing Technology, Issue 5-6, pp 580-593, 2006
- [8] P.Bapst, A novel evolutionary algorithm for Just in time job shop scheduling, International Journal of research, Vol 35, pp 9
- [9] P. Srinivasan, A novel evolutionary algorithm for job shop scheduling, Asian Journal of Scientific Research, 6: 107-114, 2013.