

Artificial Immune System algorithm for multi objective flow shop scheduling problem

M.Vairamuthu, S.Porselvi, DR.A.N.Balaji, J.Rajesh Babu

ME (CAD/CAM), K.L.N college of Engineering, Tamilnadu, India

Assistant professor of mechanical engg, K.L.N college of Engineering., Tamilnadu, India

Professor of mechanical engg, K.L.N college of Engineering, Tamilnadu, India

Associate professor of automobile engg K.L.N college of Engineering, Tamilnadu, India

ABSTRACT— The n-job, m-machine flow shop scheduling problem is one of the most general job scheduling problems. This study deals with the multi objective optimization with the criteria of makespan minimization and minimization of total weighted flow time for the flow shop scheduling problem. The n-job m-machine problem is known to be NP hard problem, several meta heuristics have been applied so far in the literature. Artificial Immune Systems (AIS) are new intelligent problem solving techniques that are being used in scheduling problems. AIS can be defined as computational systems inspired by theoretical immunology, observed immune functions, principles and mechanisms in order to solve problems. In this research, a computational method based on clonal selection principle and affinity maturation mechanisms of the immune response is used. Matlab code was generated to use the algorithm for finding the optimal solution.

KEYWORDS— Artificial Immune Systems, flow shop scheduling problem.

I. INTRODUCTION

In the quick changing market the production scheduling plays a key role in manufacturing systems of an enterprise wanting to maintain competitive position. Due to that competition, developing effective and efficient advanced scheduling technologies is extremely important. The so-called flow shop scheduling problem represents a class of widely studied cases based on ideas derived from production engineering, which currently modeled a lot of manufacturing systems, assembly lines, information service facilities. [1]Ventura et.al, presents linear programming formulations to find optimal starting and completion times for all the jobs in a given sequence considering various types of buffers and sub lots.

Heuristic algorithms that blend linear programming with several pair wise interchange strategies are proposed to find near-optimal solutions for multiple-job, multiple machine lot-streaming flow shop scheduling problems. [2]Buthainah.et.al, Concluded that the Flow Shop problem is tackled using Genetic Algorithm based solution for Traveling Salesman Problem (GATSP) in order to obtain optimal machine sequencing schedule. It must be noted that critical path must always pass through machine with large processing times for all jobs. The proposed algorithm targeted the maximization of the total efficiency of the shop given only the job/machine processing time matrix for machine scheduling, with ease and flexibility of data handling and manipulation. [3]Mostafa.et.al, proposed a parallel genetic algorithm to solve the flow shop Scheduling Problem with regard to being NP-hard, the method of parallel genetic algorithm with the use of MATLAB 7.0 software has been developed and then the quality of the results with their time of calculation is compared with the results obtained from GA(Genetic Algorithm). [4]Pempera.et.al, gives that the flow shop scheduling problem with minimizing two criteria simultaneously: the total completion time (makespan) and the sum of tardiness of jobs. The problem is strongly NP-hard, since for each separate criterion the problem is strongly NP-hard. There is a number of heuristic algorithms to solve the flow shop problem with various single objectives, but usage of those heuristics to multi-criteria flow shop problems is rather limited. They are propose a new idea of the use of simulated annealing method to solve certain multi-criteria problem. Especially, define a new acceptance rules and the mechanism of moving the search in different regions of solution space by using so called drift. [5]Jolai.et.al, proposed a bi-objective no-wait two-stage flexible flow shop is considered. The objectives of minimizing makespan and maximum tardiness of jobs are taken into account. The aim is finding the best approximate of

Artificial Immune System algorithm for multi objective flow shop scheduling problem

Pareto optimal solutions. Three meta-heuristic Pareto-based multi-objective algorithms, called Classic weighted simulated annealing (CWSA), Normalized weighted simulated annealing (NWSA) and Fuzzy simulated annealing (FSA), were proposed. In order to evaluate the performance of these algorithms, 18 problems in different sizes were solved.

Any scheduling problem essentially depends upon three important factors namely, job transportation time which includes moving time and idle time, relative importance of a job over another job and breakdown machine time (non-time due any reason). These three factors were separately studied by many researchers. Miyazaki and Nishiyama (1980) had carried out an analysis for minimizing weighted mean flow time in flow-shop scheduling. [6]Chandramouli proposed a heuristic approach for n-job, 3-machine flow-shop scheduling problem involving transportation time, breakdown time and weights of jobs. [7]Pandian and Rajendran improved and simplified the procedure for a constrained flow-shop problem for 3-jobs. [8]Baskar and Anthony gives that new procedure is proposed to obtain a scheduling sequence having optimal or near optimal makespan for a flowshop scheduling problems involving known break down time.

II. PROBLEM DESCRIPTION

Multi-objective optimization is defined as the problem of finding a vector of decision variables that satisfies all constraints and simultaneously optimizes a vector function whose elements represent the objective functions. Mathematically, the multi-objective optimization problem can be formulized as follows:

$$\begin{aligned} \text{Min } f(x) &= \{f_1(x), f_2(x), \dots, f_m(x)\} \\ x &\in X \\ \text{s.t. } g(x) &\leq 0, h(x) = 0, \end{aligned}$$

- The processing of each job has to be continuous.
- That is, once a job is started on the first machine, it must be processed through all machines without any pre-emption and interruption.
- Each machine can handle no more than one job at a time. Each job has to visit each machine exactly once.
- The release time of all jobs is zero. It means all jobs can be processed at time 0.

The first objective considered is to minimize the maximum completion time or makespan (C_{max}). The corresponding fitness function is calculated as follows:

$$\begin{aligned} C_j &= \text{Completion time of job } j \\ \text{Makespan} &= C_{max} = \max(C_j) \\ \min z_1 &= C_{max}. \end{aligned}$$

Another objective considered in this paper is to minimize mean weighted flow time. This objective is calculated as follows:

$$\min z_2 = w_f$$

A mixed integer programming model is given using the following notation.

n	Number of jobs to be scheduled
i	Number of stages ($i = 1, 2$)
m	Number of machines to be processed
K	Number of machines in stage i
$O_{(i,j)}$	Available time for job j in stage i on machine k
x_{jl}	1 if job j locates in position l of the job sequence. 0 otherwise
y_{jk}	1 if job i is processed on machine k. 0 otherwise
B_{ij}	start time of job j in stage i
C_{ij}	completion time of job j in stage i

Objective functions:

$$\begin{aligned} \text{Min } z &= (z_1, z_2) \\ z_1 &= \max(C_j) \\ z_2 &= \max(w_f) \end{aligned}$$

Weighted method:

$$\begin{aligned} \text{Min } Z &= w_1 \times f_1(x) + w_2 \times f_2(x), \\ w_1 + w_2 &= 1, w_1, w_2 \geq 0. \end{aligned}$$

III. PROPOSED METHOD

This section discusses the implementation of an artificial immune system (AIS) algorithm to solve the flow shop problem.

3.1 Introduction to AIS

AIS is an adaptive system inspired by theoretical immunology and observed immune functions, principles, and models, which are applied to complex problem domains [11]. The principle of clonal selection and affinity maturation of the immune response system leads to the development of powerful computational tools to solve combinatorial optimization problems. The human immune system protects our body from the attack of foreign organisms, such as viruses and bacteria. These foreign organisms are called antigens. The molecules, called antibodies, that recognize the presence of an antigen will increase rapidly by cloning. This process is called the clonal selection principle [12].

Artificial Immune System algorithm for multi objective flow shop scheduling problem

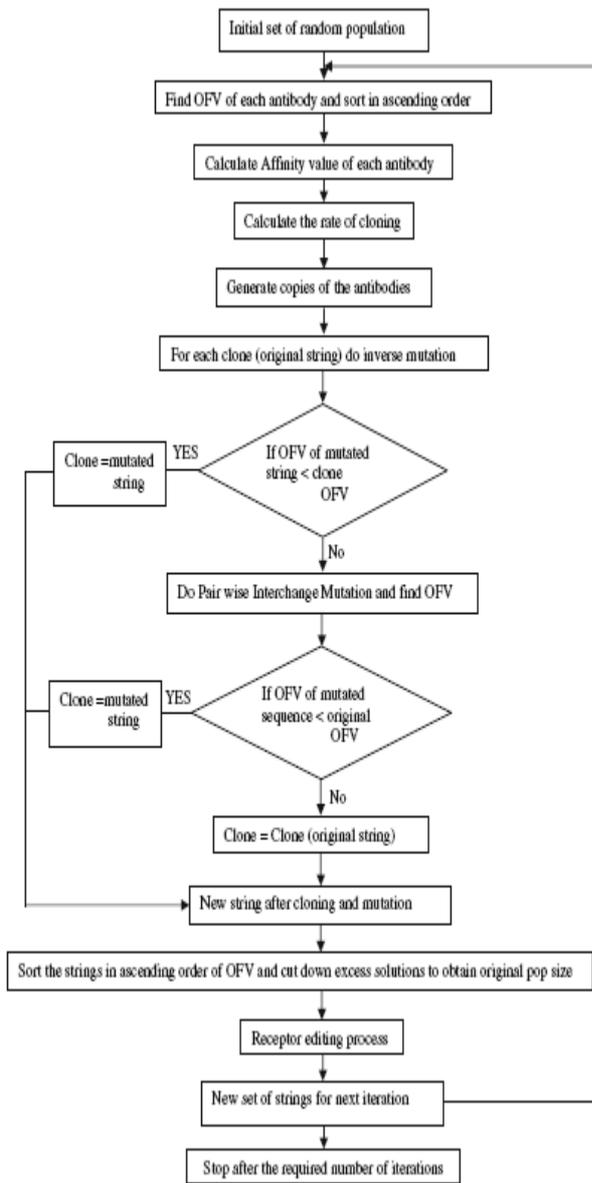


Fig. 1: Flow chart of the proposed artificial immune system (AIS) algorithm

The new cloned cells undergo mutations in order to improve the affinity of the antibodies, which results in antigen neutralization and elimination. These mutations experienced by the clones are proportional to their affinity to the antigen [13]. As a subsequent process to mutation, some percentage of the worst cells is eliminated and the same percentage of new antibodies is introduced into the system. This process is called receptor editing. This process is used to explore new search regions and also to escape from local optima [9]. The recognition and learning capabilities of the natural immune system provides properties such as robustness, dynamism, and adaptability to AIS-based algorithms [14]. These properties of the immune system attract researchers to apply it to other functional areas of engineering. The application of AIS is found in areas such as biological modeling, computer network security and virus detection,

data mining, robotics, scheduling, clustering, and classification.

3.2 The AIS algorithm steps

The general AIS algorithm consists of the following steps:

1. Initialization:
 - Create a random population of individuals p .
2. Affinity evaluation:
 - Measure the affinity of each individual in the population.
3. Clonal selection:
 - Select the n best individuals of the population based on the affinity measure.
4. Clonal expansion:
 - Clone the n best individuals of the population proportional to the rate of cloning. The clone is an identical copy of the original string. (The clone size is an increasing function of the affinity measure.)
5. Affinity maturation:
 - Mutate each clone to generate a matured antibody of the population.
 - Preserve the improved individuals for the next generation.
6. Metadynamics:
 - Replace R individuals with low affinity value with randomly generated new ones. The lower affinity cells have higher probabilities of being replaced. This process introduces diversity into the population.
7. Cycle:
 - Repeat step 2 to step 6 until a certain stopping criteria is met.

3.3 Numerical illustration

The AIS approach has three user-defined parameters, namely, population size (P), the number of iterations (n), and the amount of low-affinity antibodies to be replaced (R). The algorithm has been tested for the following ranges of parameters:

- $P=20, 40, \text{ and } 50$
- $n=500, 1,000, 1,500, \text{ and } 2,000$
- $R=0, 10, \text{ and } 20$

After running the algorithm, better solutions were obtained using the following parameters:

- Population size: 50
- No. of iterations: 2,000
- Amount of low-affinity antibodies to be replaced: 10

Artificial Immune System algorithm for multi objective flow shop scheduling problem

The working of the AIS algorithm is explained through the following numerical illustration:

Step 1 Initialize a random population of strings up to the specified population (P). A string represents a possible solution. For example, the string 6 1 5 9 7 2 3 4 8 10 represents a layout sequence which is to be placed around the loop.

Step 2 Compute the objective function value (OFV) and affinity value for all solutions in the current population. The affinity value is calculated from the following equation:

$$\text{Affinity} = 1/\text{OFV}$$

From the above equation, it is clear that lower the OFV, higher the affinity value.

Step 3 The selection of strings for cloning is done directly proportionally according to their affinity value.

Step 4 The rate of cloning is calculated by using the following relation:

$$\text{Rate of cloning} = \frac{\text{Affinity value of a solution} * \text{population size}}{\text{Total of the affinity values of solutions in the population}}$$

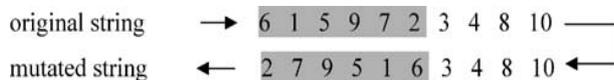
The above procedure gives more clones of strings that have lower OFVs than those with higher OFVs. This also gives rise to a temporary population of clones(C).

Step 5 Each clone inverse and pairwise mutation is carried out as follows:

Step 5a Inverse mutation

Generate a random number between (1, N), where N is the number of machines.

Let 1 and 6 be the randomly selected positions in the string. The inverse mutation in the original string is performed by inverting the sequence of machines between locations 1 and 6, as shown below:

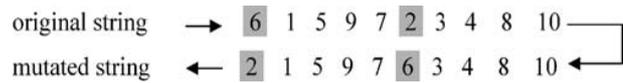


After performing the inverse mutation, if the OFV of the mutated string is less than the original string, then the original string is replaced by the mutated string; else, pairwise mutation is carried out on the original string.

Step 5b Pairwise mutation

Generate a random number between (1, N).

Let 1 and 6 be the randomly selected positions in the string. The pairwise mutation in the original string is performed by interchanging the locations of machines 1 and 6 in the original string, as shown below:



After performing pairwise mutation, if the OFV of the mutated sequence is less than the original sequence, then the original sequence is replaced by the mutated sequence; otherwise, the original sequence is retained.

After the cloning and mutation processes, reselect the improved strings from the temporary population C so as to maintain the original population size P. Now, the original population P is composed of improved strings.

Step 6 Then, R% of the solutions which has the highest OFV in the population is replaced with the same R % of randomly generated solutions.

Step 7 Repeat step 2 to step 6 for the required number of iterations.

IV. RESULTS AND DISCUSSION

The effectiveness of the proposed algorithm is demonstrated and a comparative analysis is presented in result and discussion section. The validity of the proposed model is tested for the problems taken from literature review. It shows the number of parts and the required machine sequence for each part for the test problems. The proposed algorithm is coded in the MATLAB R2009b. All tests were executed on a Intel core i3 processor under a Microsoft Windows 7 (32-bit) operating system. The artificial immune system (AIS) with the best sequence among the minimization of objective function value for the 10 type of problem is shown in table as follows:

Problem no.	No. of jobs, No. Of machines	Optimal sequence	OFV
1	4,3	4-3-2-1	38.4
2	4,4	3-4-2-1	53.9
3	5,3	3-5-2-4-1	45.6
4	5,4	3-2-4-5-1	64.6
5	5,5	2-5-4-3-1	84.5
6	6,3	4-1-6-3-5-2	61.5
7	6,4	3-2-6-4-1-5	73.6
8	6,5	4-1-5-6-2-3	92.8
9	6,6	6-3-2-1-5-4	117.6
10	10,10	2-8-10-1-5-3-7-4-6-9	175.2

V. CONCLUSION AND FUTURE WORK

The major contribution of this work has been the implementation of a best sequence depends upon minimization of the objective values of the flow shop scheduling problems. The artificial immune system (AIS)-based algorithm is implemented to solve the loop flexible flow shop problem and corresponding sequence are introduced. The performance of the proposed algorithm has been validated with the results available in

the literature. The obtained results are found to be better than the results of the literature, which proved that AIS is a robust tool for solving flow shop problems. Also, AIS has been shown to provide significant improvements in the results for large sized problems. In future, the proposed procedure can be integrated with simulation software in order to study the operational difficulties and performance measures of the flexible flow shop scheduling problems.

REFERENCES

- [1] Suk-Hun Yoon, Jose A. Ventura, "Minimizing the mean weighted absolute deviation from due dates in lot-streaming flow shop scheduling", *Computers & Operations Research*, 29, (2002) 1301-1315
- [2] Buthainah F Al-Dulaimi and Hamza A.Ali, "A Novel Genetic Algorithm Approach for Solving Flow Shop Problem", *IICSNS International Journal of Computer Science and Network Security*, VOL.8 No.9, September 2008.
- [3] Mostafa Akhshabi, Javad Haddadnia, Mohammad Akhshabi, "Solving flow shop scheduling problem using a parallel genetic Algorithm", *Procedia Technology* 1 (2012) 351 – 355.
- [4] Pempera Jaroslaw, Smutnicki Czeslaw, Zelazny Dominik, "Optimizing bicriteria flow shop scheduling problem by simulated annealing algorithm", *Procedia Computer Science* 18 (2013) 936 – 945.
- [5] F. Jolai, H. Asefi, M. Rabiee, P. Ramezani , "Bi-objective simulated annealing approaches for no-wait two-stage flexible flow shop scheduling problem" *Scientia Iranica E* (2013) 20 (3), 861–872.
- [6] A. B. Chandramouli, "Heuristic approach for n-job, 3-machine flow shop scheduling problem involving transportation time, break down time and weights of jobs" *Mathematical and Computational Applications*, Vol. 10, No. 2, pp. 301-305, 2005.
- [7] P. Pandian and P. Rajendran, "Solving Constrained Flow-Shop Scheduling Problems with Three Machines" *Int. J. Contemp. Math. Sciences*, Vol. 5, 2010, no. 19, 921 - 929
- [8] Baskar A, Dr.Anthony Xavier M, "A simple model to optimize general flow-shop scheduling problems with known breakdown time and weights of jobs" *procedia engineering*, 38, 2012, 191-196.
- [9] Orhan Engin, Alper Doyen, "A new approach to solve hybrid flow shop scheduling problems by artificial immune system" *Future Generation Computer Systems* 20 (2004) 1083–1095
- [10] J. Timmis, A. Hone, T. Stibor, E. Clark, "Theoretical advances in artificial immune systems" *Theoretical Computer Science* 403 (2008) 11–32
- [11] R. M. Satheesh Kumar & P. Asokan & S. Kumanan, "Artificial immune system-based algorithm for the unidirectional loop layout problem in a flexible manufacturing system" *Int J Adv Manuf Technol* (2009) 40:553–565.
- [12] Wael F. Abd El-Wahed, Elsayed M. Zaki, Adel M. El-Refae'y "Artificial immune system based neural networks for solving multi-objective programming problems" *Egyptian Informatics Journal* (2010) 11, 59–65.
- [13] Mahmoud Moustafa El-Sherbiny, "Alternate mutation based artificial immune algorithm for step fixed charge transportation problem" *Egyptian Informatics Journal* (2012) 13, 123–134.
- [14] Tsui-Ping Chunga, Ching-Jong Liao, "An immunoglobulin-based artificial immune system for solving the hybrid flow shop problem" *Applied Soft Computing* 13 (2013) 3729–3736. *E. Std.* 802.11, 1997.