Journal of Theoretical and Applied Information Technology
10th October 2015. Vol.80. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MINIMIZE THE MAKESPAN FOR JOB SHOP SCHEDULING PROBLEM USING ARTIFICIAL IMMUNE SYSTEM APPROACH

AHMAD SHAHRIZAL MUHAMAD, [1]SAFAAI DERIS, [2]ZALMIYAH ZAKARIA
[1]Professor, Faculty of Computing, Universiti Teknologi Malaysia
[2]Senior Lecturer, Faculty of Computing, Universiti Teknologi Malaysia
E-mail: [1]safaai@utm.my, [2]zalmiyah@utm.my

ABSTRACT

In the manufacturing industry, scheduling is a process of arranging, controlling and optimizing work and workloads in a production process. This research discussed about job-shop scheduling problem. The main problem in job-shop scheduling is to optimize the usage of machines in order to obtain the shortest time in completing the activities. Several methods have been used to solve job-shop scheduling problems and the method proposed here is artificial intelligence by using the artificial immune system algorithm (AIS). The advantage of this algorithm is fabricated by imitating the natural immune system. The results produced by this method are compared with the best results of the previous research.

Keywords: *Scheduling, Artificial Intelligence, Job-shop Scheduling, Artificial Immune System, Evolutionary Computation*

1. INTRODUCTION

Scheduling can be defined as a process to allocate shared resources over time to competing activities with hard or soft constraints given [1]. In essence, scheduling can be considered as a searching or optimization problem, with the goal of finding the best schedule. Job-shop scheduling problem is one of the well-known hardest combinatorial optimization problem. It consists of a set of n jobs with a number of m machines. In each job lies a series of operations. At a particular time, one machine can only address at most one operation. Preceding operation must be scheduled to complete before the machine can proceed to its succeeding operations in the job series. All operations are required to be completed in a continuous time without any interruption in a certain length on a specific machine. In most industries, job-shop scheduling (JSS) is important because it decides process maps and process capabilities [2]. The ultimate aim of job shop scheduling is to produce a scheduling that minimizes the total time taken to complete all the activities. Figure 1 illustrates the job shop scheduling problem.

The solution of any optimization problem is evaluated by an objective function. Objectives associated with cost, resources and time are minimized. There are several objective functions for job shop scheduling problem, but in this paper we will discuss on how to minimize the *makespan*, also known as maximum completion time and indicates the completion time for the last job to be completed. The process constraints will influence the finding of the best schedule and determine whether its employment will be very easy or very difficult [3]. The *makespan* is important when having a finite number of jobs and is closely related to the throughput objective. When the maximum completion time can be minimized as can as possible, the machine resource can be used to process another job as early as.

	Machine			Processing Time		
	O_1	O_2	O_3	O_1	O_2	O_3
Job_1	1	2	3	4	3	2
Job_2	2	1	3	1	4	4
Job_3	3	2	1	3	2	3
Job_4	2	3	1	3	3	1

Figure 1: Job Shop Scheduling Problem

The solution of any optimization problem is evaluated by an objective function. Objectives associated with cost, resources and time are

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

minimized. There are several objective functions for job shop scheduling problem, but in this paper we will discuss on how to minimize the *makespan*, also known as maximum completion time and indicates the completion time for the last job to be completed. The process constraints will influence the finding of the best schedule and determine whether its employment will be very easy or very difficult [3]. The *makespan* is important when having a finite number of jobs and is closely related to the throughput objective. When the maximum completion time can be minimized as can as possible, the machine resource can be used to process another job as early as.

In business, customer's satisfaction is very important. To ensure that the customers are satisfied with the service provided, all of their requirements must be fulfilled. In the manufacturing industries, producing the product according to time can ensure customer's satisfaction. One of the challenges of the manufacturing industry is to schedule the machine to ensure the product required by customer can be produced according to time or before the due date. Apart from that, the industry must also take into consideration the requirements from several other customers at the same time. Other problems to be considered by manufacturing industry are machine break down and rush order. In order to overcome the problems faced by manufacturing industries, a tool or technique must be developed to assist them in organizing the schedule for their machines.

2. JOB SHOP SCHEDULING PROBLEM SOLVING METHOD

Methods of solving job shop scheduling problems include exact methods and approximation methods. The most significant Exact Method used for the Job Shop Scheduling Problem is called the Branch-and-Bound Method [4]. This method was developed by Land and Doig [5], primarily for the purpose of 'optimization of problems which could be formulated as linear programming problems with additional constraints. Although the Exact Method is fairly simple, it could not give optimum solution, especially to resolve scheduling problem. To overcome this problem, an approximation technique is used. There are several type of approximation techniques which include priority dispatch rules, bottleneck based heuristics, opportunistic scheduling and artificial intelligence.

Recently, artificial intelligence becomes a popular technique to solve the problem in scheduling and job shop scheduling. The popular technique in artificial intelligence to solve the scheduling problem is genetic algorithm. Genetic Algorithm (GA), proposed by Holland [6] has been successfully applied to solve many combinatorial optimization problems including scheduling. GA is not only effective to perform a global search, but also flexible to hybridize with other domain-dependent heuristics or local search techniques for solving specific problems.

Although genetic algorithms have proven to be an efficient and powerful problem-solving strategy, they are not a panacea. GAs also has certain limitations such as the language used to specify candidate solutions must be robust. Another limitation is the other parameters of a GA which are the size of the population, the rate of mutation and crossover, the type and strength of selection must be also chosen with care. One of the problem for genetic algorithms, it has a difficulty dealing with problems such as "deceptive" fitness functions, those where the locations of improved points give misleading information about where the global optimum is likely to be found.

To overcome the GA problem, the artificial immune system can be used. Originally, immunity means control of disease especially infectious disease. Cells and responsible molecules create immunity form immune system. Coordination intercellular is reaction with and molecule to entry something foreign substance known as immune reaction or immune response.

Natural immune system has become one of important subject studies recently due to its ability to process huge information. During (the time where) natural immune system become becomes popular, an imitation immune system was introduced. Imitation immune system or known as Artificial Immune System (AIS) is a set of techniques, which its algorithm is fabricated by imitating the natural immune system, so that its behaviour is similar to the natural immune system [7]. These techniques are commonly used in pattern recognition, detection of defects, diagnosis, and others, including optimization.

AIS can be defined as a calculation system based on metaphors for biological immune system. The engineering immune is a meta-synthesis process using information contained in the problem itself to define the solution tool to a given problem and apply it to find the solution for that problem. To develop AIS for engineering, we need to give attention to the following important aspects:

Journal of Theoretical and Applied Information Technology
10th October 2015. Vol.80. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

i. Hybrid structure and algorithm are translated into immune system components;

ii. Algorithm calculation based on immunology principle, distributed processing, clone selection algorithm, and network theory immune;

iii. Immune based on optimization, self-learning, self-organization, artificial life, cognitive models, multi agent system, design and scheduling, pattern recognition and anomaly detection;

iv. The immune tools for engineering.

3. ARTIFICIAL IMMUNE SYSTEM FOR JOB SHOP SCHEDULING

In past research by Hart *et. al* [8] and Hart and Ross [9], have shown that AIS model can be used to solve the scheduling problem in the industrial environment for real world situations which require a schedule maker to make new schedule when there are changes, such as the change in environment and unexpected. In his study, [10] define antigen as "sequence of jobs on a particular machine given a particular scenario" and antibody as "a short sequence of jobs that is common to more than one schedule".

In AIS approach, there are five models that can be used, including Bone Marrow Models, Negative Selection Algorithm, Clonal Selection Algorithm, Somatic Hyper mutation and Immune Network Models. For scheduling problem purpose, the suitable models are Negative Selection Algorithm (NSA) and Clonal Selection Algorithm CSA. In this paper, we will use the CSA approach to solve the JSSP and get a minimize *makespan*. Figure 2 illustrates how the Clonal Selection Principle (CSP) to work.

The clonal selection principle or theory is the algorithm used by the immune system to describe the basic features of an immune response to antigen stimulus. It establishes the idea that only those cells recognize the antigens proliferate, thus being selected against those which do not. The clonal proliferation of the B cells occurs inside the lymph nodes [11] within a special microenvironment named germinal center, in the follicular region of the white pulp, which is rich in antigen presenting cells [12]. When a living body is exposed to an antigen, B cells will respond by producing the antibodies. Basically, B cells clone themselves into as many as is required to fight the infection and clone only those cells which are actually good enough to fight. During the process of clonal proliferation, a hyper-mutation mechanism takes place on the variable region of B cells. The hyper-mutation plays a critical role in creating diverse antibody, increasing affinity and enhancing specificity of antibody.

Based on the CSP working, we come up with the model to solve the JSSP and from this model, the modification was make to minimize the makespan. In this model, the job shop problem is translated into a string, call an antibody and this antibody is generated randomly to get some population of antibodies. All the antibody in the population will go through the clone and mutation process. The best antibody is chosen as the solution before we encode it into the schedule. Figure 3 illustrates the basic model proposed to solve the JSSP.

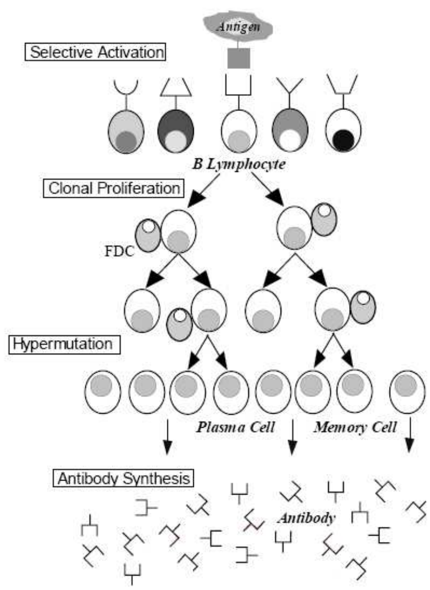

Figure 2: Clonal Selection Principle

In the proposed model, the integer string encoding antibody population is generated in two ways included randomly and by using the libraries. The length of each antibody is equal to m machine multiply by n jobs, in which each job j will appear m times in the antibody. For randomly generated, job j, also known as gene, is generated m times and place randomly in the integer string. Otherwise for those generated from libraries, the libraries contain the components and the component is an integer string. The component will generate randomly and the length of component (integer string) multiply by number of libraries must be equal to length of antibody.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

For the second stage all the antibody in population will be cloned for some number of pre-determine. This entire clone will be mutated to get the antigen using some of mutation types. For the first iteration, the antigen will be assigned to the best solution and for the next iteration the clone (after mutation) will be compared with the current solution to know which one is better, if the clone is better than current solution, that clone will be assign as current solution. The role of mutation type is very important in influencing the final solution. In the proposed model the mutation type used is included:

i. Random Somatic Point Mutation.

ii. Heavy-Light Somatic Point Mutation.

iii. Somatic Recombination.

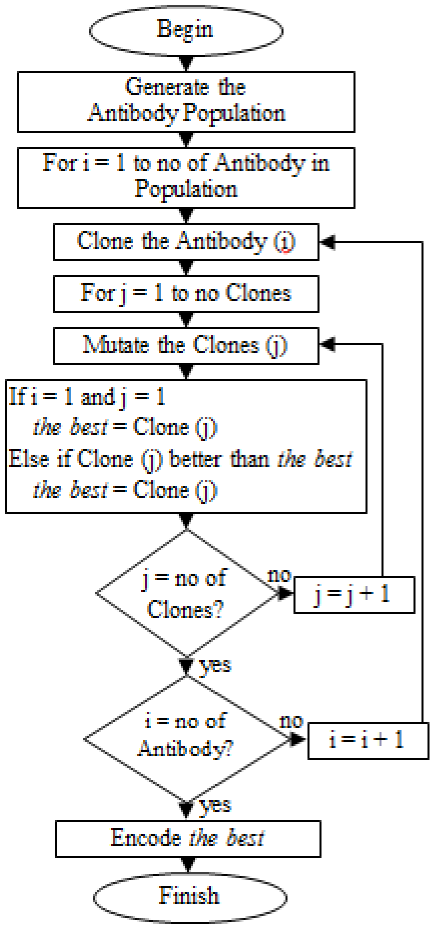

Figure 3: Basic Model Proposed to Solve JSSP using AIS

In Somatic Point Mutation, two genes will be chosen randomly from the antibody and that genes will be swapped. In the antibody each genes are heavy value in chain type except the last genes for each job. Based on the chain type value, Heavy-Light Somatic Point Mutation is implemented, where the two genes will be chosen from antibody with the heavy value and light value, and that genes will be swap [13]. For the Somatic Point Recombination, two couple genes will be chosen randomly and that couple of genes will be swap. In the proposed model, only one of three of the mutation type will be chosen in each iteration [13].

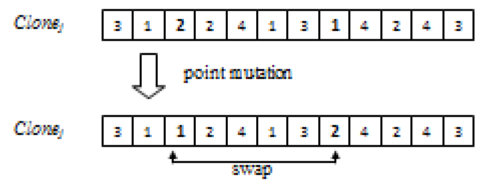

Figure 4: Random Somatic Point Mutation

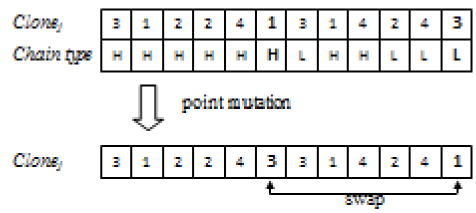

Figure 5: Heavy-Light Somatic Point Mutation

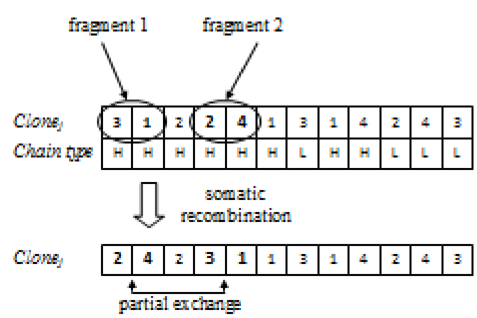

Figure 6: Somatic Recombination

4. EXPERIMENTAL RESULTS

Using the proposed model, the experiment is run using different size of job shop problems. For experiment purpose, Java technology was used to transfer an algorithm into computer code. Java technology was chosen because it supports the object oriented programming and easier to make any changes. The experiment was run by using computing grid system facilities at Universiti Teknologi Malaysia. To ensure the validity of the result from this experiment, the three validation methods were used, there are:

i. Compared with the previous research and with the best known result. Table 1 shows the comparison of the initial result between the existing results for previous research. From the comparison some of

Journal of Theoretical and Applied Information Technology
10th October 2015. Vol.80. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the problem gives the same result with the best known of result and the others give the worst result from the best known of result.

ii. Manual calculation based on integer string, machine list and time list from the result. Figure 6 shows the example of integer string, machine list and time list.

iii. Pick a random result and transfer it into a gantt chart manually by using Microsoft Excel software.

Integer String : 4-10-3-3-6-9-5-10-4-2-8-7-1-8-6-4-7-10-5-9-3-1-2-10-8-9-2-6-4-7-10-3-5-1-9-4-2-7-8-5-9-3-1-6-1-5-8-6-2-7

Machine List : 2-5-4-5-2-4-1-4-1-1-3-4-2-1-3-5-5-3-4-2-2-1-4-2-2-5-5-5-3-2-1-3-3-5-1-4-3-3-4-2-3-1-4-1-3-5-5-4-2-1

Time List : 77.0-77.0-39.0-98.0-54.0-17.0-83.0-79.0-55.0-21.0-38.0-69.0-21.0-60.0-43.0-79.0-77.0-43.0-34.0-49.0-42.0-53.0-52.0-75.0-41.0-25.0-16.0-79.0-66.0-87.0-96.0-31.0-64.0-95.0-44.0-77.0-26.0-87.0-24.0-19.0-98.0-12.0-55.0-92.0-34.0-37.0-83.0-62.0-71.0-93.0

Figure 6: Integer String, Machine List and Time List

Table 1: The Result Comparison

No.	Problem	Size	Initial Result	The Best Know
1	abz5	10 X 10	1239	1234
2	abz6	10 X 10	948	943
3	abz7	20 X 15	747	665
4	abz9	20 X 15	786	686
5	**ft06**	**6 X 6**	**55**	**55**
6	ft10	10 X 10	976	930
7	ft20	20 X 5	1246	1165
8	**la01**	**10 X 5**	**666**	**666**
9	la02	10 X 5	669	655
10	la03	10 X 5	617	597
11	la04	10 X 5	607	590
12	**la05**	**10 X 5**	**593**	**593**
13	**la06**	**15 X 5**	**926**	**926**
14	**la07**	**15 X 5**	**890**	**890**
15	**la08**	**15 X 5**	**863**	**863**
16	**la09**	**15 X 5**	**951**	**951**
17	**la10**	**15 X 5**	**958**	**958**
18	**la11**	**20 X 5**	**1222**	**1222**
19	**la12**	**20 X 5**	**1039**	**1039**
20	**la13**	**20 X 5**	**1150**	**1150**
21	**la14**	**20 X 5**	**1292**	**1292**
22	la15	20 X 5	1217	1207
23	la16	10 X 10	982	945
24	la17	10 X 10	793	784
25	la18	10 X 10	865	848
26	la19	10 X 10	868	842
27	la20	10 X 10	922	902
28	la21	15 X 10	1137	1046
29	la22	15 X 10	1001	927
30	la23	15 X 10	1064	1032

5. CONCLUSION

From the experiment, it is evident that the proposed model can solve the JSSP. The main problem in this proposed model is that, some results are still poor from the best known result and the time to produce a result is too long for some job shop problems. However, some modifications must be made on the proposed model to improve the model especially to produce the same or better result than the best known result and to ensure the model can give the solution as fast as possible. For further study on improvement of the model, our suggestion is to give more choice for the mutation type. We also plan to make more than one level of clone process, meaning, each clone for antibody will be cloned again after the mutation process to implement the next mutation process. In this case the clone will have a level of clone process as a pre-determine number.

REFRENCES:

[1] K.R. Baker, *Introduction to Sequencing and Scheduling*, New York: John Wiley, 1974.

[2] V. Roshanaei, B. Naderi, F. Jolai and M. Khalili, A Variable Neighborhood Search for Job Shop

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Schedulin with Set-up Times to Minimize Makespan, *Future generation Computer Systems*, Vol. 25, 2009, 654-661.

[3] M. Pinedo, *Scheduling: Theory, Algorithms and Systems*, New Jersey: Prentice-Hall, Englewood Cliffs, 2002.

[4] A.S. Muhamad and S. Deris, An Artificial Immune System for Solving Production Scheduling Problems: A Review, *Artificial Intelligent Review*, Vol. 39, No. 2, 2013, 97 - 108.

[5] A.H. Land and A.G. Doig, An Automatic Method of Solving Discrete Programming Problems, *Econometricca*, Vol. 28, No. 3, 1960, pp. 497 – 520.

[6] J.H. Holland, Genetic algorithms. Scientific American, July, 1992, 114–116.

[7] E. Hart and J. Timmis, Application areas of AIS: the past, the present and the future, *Applied Soft Computing*, Vol. 8, 2008, 191-201.

[8] E. Hart, P. Ross, and J. Nelson, Producing Robust Schedules Via An Artificial Immune System, *Proceeding of the ICEC '98, IEEE Press*, 1998, 464-469.

[9] E. Hart, and P. Ross, An Immune System Approach to Scheduling in Changing Environments, *Proceeding of the GECCO 1999*, W. Banzhaf, J. Daida, A.E. Eiben, M.H. Garzon, V. Honavar, M. Jakiela and R.E. Smith (Eds.), Morgan Kaufmann, 1999, 1559-1565.

[10] E. Hart and P. Ross, The Evolution and Analysis of a Potential Antibody Library for Job-Shop Scheduling, *New Ideas in Optimisation*, D. Corne, M. Dorigo and F. Glover (Eds.), McGraw-Hill, London, 1999, 185-202.

[11] I.L. Weissman and M.D. Cooper, How the Immune System Dvelops, *Scientific American*, Vol. 269, No. 3, 1993, 33-40.

[12] D. Tarlinton, Germinal Centers: Form and Function, *Current Operation in Immune*, Vol. 10, 1998, 245-251.

[13] C.H. Chueh, *An Immune Algorithm for Engineering Optimization*, Tatung University, 2004, Ph.D. Thesis.