Implementation of CSA with Clone-Mutation Mechanism to the JSSP

Yılmaz Atay and Halife Kodaz

***Abstract*—Job Shop scheduling problem (JSSP), which is called NP-Hard, is classified as one of the most difficult problems. Existing methods for the solution of such problems is not enough. Therefore, to solve such problems, some of the artificial intelligence techniques are used. In this study, clonal selection algorithm (CSA), which is one of artificial immune algorithms, is proposed to solve the job shop scheduling problems. In the proposed method the selection of a set of clone and mutation process of selected antibodies were carried out. This method is named after clone-mutation mechanism. The proposed method was applied on some test problems called as Ft06, La01, La03, La04 and La05. Furthermore, the application, in order to understand the effect of the mutation mechanism was executed with values ranging from 0 to 1, and the results are given in the Table VI. The results obtained are compared with the best known makespan values. As a result, the proposed method has been applied successfully in job shop scheduling problems.**

***Index Terms*—Artificial immune systems, artificial intelligent, clonal selection algorithm, job shop scheduling problem, mutation.**

I. Introduction

Optimization problems emerge in the most important moments of our lives. It is very important to solve problems like distribution of health workers in hospitals, nursing scheduling, vehicle routing, aircraft scheduling, schools optimization services and job-machine optimization in factories.

In this study clonal selection algorithm which gives very good results in many different scheduling problems was applied to job shop scheduling problem. As distinct from the classical clonal selection algorithm, specific changes in the mechanism of clone selection and mutation were made according to the problem. Antibodies that have more than a particular threshold have been selected from within the entire population and then these antibodies are included in the set of clone. In the cluster the mutation process was performed to every antibody. As a result of mutation process obtained antibodies subjected to re-selection and this process was continued for all iterations.

As a result, the minimum completion time is calculated for the entire job. Up to date many studies on job shop scheduling has been conducted. The rapid development of artificial intelligence techniques increased success to the application of algorithms to this type of optimization. Some studies conducted so far on the job shop scheduling is given below.

Garey *et al.*, examined the complexity of flow and job shops scheduling. In addition, they stated that in two machine job shop production scheduling; NP-hard is the best solution to make lowest total completion time [1].

Arkin and Silverberg, worked on job shop scheduling problem with constant start and end time. They proposed an algorithm to maximize the number of completed jobs with the same functionality machines. In this way, they attempted to minimize the completion of job by doing maximum work in a certain time [2].

Ogbu and Smith, attempted to reduce the completion time of simulated annealing and banned search techniques. Thus, the completion of all work provided as quickly as possible [3].

Sotskov and Shakhlevich, examined the specific situation which the number of job is 3, in order to determine complexity of job shop scheduling problem that minimizes the maximum completion time. Eventually operations can be divided into parts or not, both within the purview of job shop scheduling problem with three machines and three jobs has proven to be NP-hard [4].

Zhou *et al.*, indicated that scheduling is important for the shop production management and combinatorial optimization. They proposed hybrid genetic algorithm method which takes rules of scheduling into account. They compared the results obtained with the neighborhood search, simulated annealing and the traditional genetic algorithm [5].

Yang *et al*, proposed memetic algorithm based on clonal selection to solve the jobs scheduling problems. They compared the obtained result with different studies [6].

Murugesan *et al.*, in their “Clonal selection algorithm using improved initialization for solving JSSP” named study have proposed a new method for job shop scheduling. The proposed method provides better results and faster convergence than with the conventional method. They compared the results obtained with genetic algorithm [7].

Akhshabi *et al.*, used clonal selection algorithm in flexible job shop scheduling problems. They aimed to minimizing the time for completion of the work. When the results obtained are analyzed, the proposed method has been applied successfully in scheduling problems [8].

Cheng *et al.*, in 2013 they published an article named “A hybrid evolutionary algorithm to solve the job shop scheduling problem”. In this article they have proposed hybrid evolutionary algorithm (HEA) to solve the job shop scheduling problems. Tabu search algorithm is incorporated into the evolutionary process and it has been adapted to the job shop scheduling problem. Their proposed method was applied to commonly used test problems in the literature and has been obtained 90% of the best known solutions [9].

Manuscript received October 27, 2013; revised December 18, 2013.

The authors are with the Computer Engineering Department, Engineering Faculty, Selçuk University, Konya-Turkey (e-mail: yilmazatay@selcuk.edu.tr, hkodaz@selcuk.edu.tr).

In next sections of this study, job shop scheduling, clonal selection algorithm, clonal selection algorithm with clone-mutation mechanism, proposed method are described respectively in details and the obtained results are evaluated.

II. Job Shop Scheduling Problem

JSSP is the most complex structured in terms of the classification of scheduling problems. Because there is no restriction on the number of process of a specific job and there are many production routes are available to be considered as an alternative. Job shop scheduling has its own processes and processing order and every order to be processed on different machines [10].

To understand the general logic of Job shop scheduling problems Table I-II and Fig. 1 are given below. In Table I the 3×3 job shop problem has been given. In Table II according to the Table I a job example with 9 processes to be solved is given. Fig. 1 shows the solved state of the job given in Table II.

The aim of this study is to develop a method to solve the job shop scheduling problem described above as quickly as possible. So that the total completion time of all operations can be minimized.

TABLE I: 3×3 Job Shop Schedule Type Problem

Jobs	1	1	1	2	2	2	3	3	3
Processes	1	2	3	1	2	3	1	2	3
Machines	3	1	2	1	2	3	3	1	2
Time	4	2	1	10	3	12	2	9	7

TABLE II: Job Shop Problem to be Solved

1	2	3	2	1	3	3	2	1

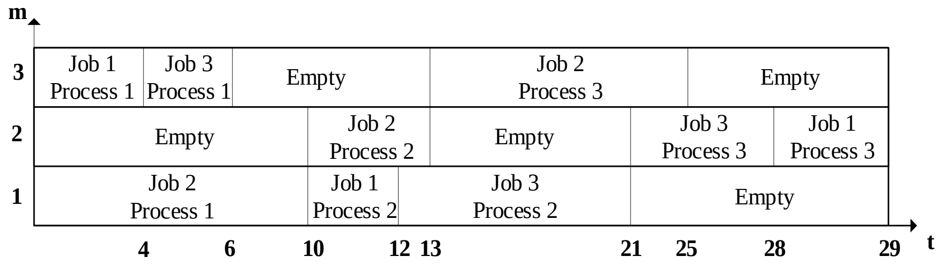

Fig. 1. The solution diagram of 9 processes.

III. Clonal Selection Algorithm

The principle of clonal selection expresses the properties of the immune response which is given by the system against an antigenic stimulation. This principle forms the basis of the algorithm used by the immune system [11]. CSA which takes advantage of the natural immune system uses the basic characteristics of the immune response against an antigenic stimulus.

CSA is based on the principle of the proliferation of only cells which recognize antigens in the system. It means those which recognize would be selected by antigens but those who do not would not be selected. These selected cells are subjected to the process of formation of similarity. This process increases the similarity of cells to antigen. The basic operation related to immunity which is taken into account has been given below [12].

- Storing the memory cells,
- Individual selection and cloning procedure of most stimulated by antigen,
- The elimination of cells not stimulated by antigen,
- Re-selection and mutation of cells to increase the similarity,
- Production of differentiation and participation of different cell in the populations,
- Similarity of cell in proportion to the hyper mutation.

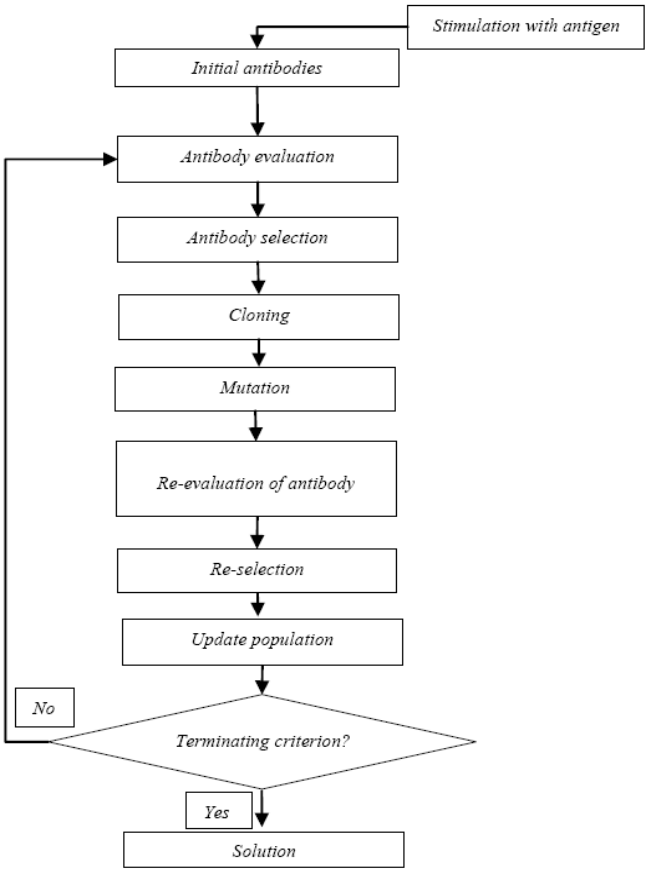

Fig. 2. Clonal selection algorithm flow chart.

IV. CSA with Clone-Mutation Mechanism

The proposed method uses the basic steps of clonal selection algorithm. Besides the classical method a different method of mutation and selection process is proposed. In classic selection and mutation mechanism the entire cluster is subjected to mutation process. In the proposed method selected antibodies are included in the clone set. Proposed mutation mechanism is applied only to antibodies in the clone set. Set of receptor are eliminated from system and replaced by a new produced antibody.

The most important two phases of the proposed method are selection and mutation operations. General descriptions about them are given below.

Creating a set of clones (selection): Cloning ratio value is multiplied by the number of population to obtain cloning number. This number is the sum of antibodies which has the best makespan values in the population. This number is then multiplied by the cloning multiplier value provided as the parameter. Antibodies are constructed in clone set as much as obtained number.

• *Subjecting the clones set to mutation (mutation)*: All antibodies obtained from the clone set undergo mutation. This process has provided diversity to achieve the best solution.

Antibody size which is 36 with a 6×6 antibody is given in Table III. The mutation mechanism for this antibody is given in Table IV.

TABLE III: A 6×6 ANTIBODY SAMPLE

2	**1**	**3**	**6**	**5**	**1**	**1**	**2**	**3**	**4**	**4**	**6**	**3**	**4**	**3**	**5**	**6**	**3**	**5**	**5**	**6**	**2**	**3**	**4**	**5**	**2**	**5**	**1**	**2**	**1**	**6**	**1**	**4**	**6**	**2**	**4**

TABLE IV: 6×6 MUTATION MECHANISM FOR AN ANTIBODY

Location-1	Location-2	Value-1	Value-2	Change-1	Change-2
1	10	2 ↔	4	**4**	**2**
21	13	6 ↔	3	**3**	**6**
5	26	5 ↔	2	**2**	**5**
28	35	1 ↔	2	**2**	**1**
24	16	4 ↔	5	**5**	**4**
31	2	6 ↔	1	**1**	**6**
11	30	4 ↔	1	**1**	**4**
7	3	1 ↔	3	**3**	**1**
33	12	4 ↔	6	**6**	**4**
4	27	6 ↔	5	**5**	**6**
36	8	4 ↔	2	**2**	**4**
17	14	6 ↔	4	**4**	**6**
9	19	3 ↔	5	**5**	**3**
23	22	3 ↔	2	**2**	**3**
25	6	5 ↔	1	**1**	**5**
15	34	3 ↔	6	**6**	**3**
32	29	1 ↔	2	**2**	**1**
20	18	5 ↔	3	**3**	**5**

TABLE V: MUTATED ANTIBODY

4	**6**	**1**	**5**	**2**	**5**	**3**	**4**	**5**	**2**	**1**	**4**	**6**	**6**	**6**	**4**	**4**	**5**	**3**	**3**	**3**	**3**	**2**	**5**	**1**	**5**	**6**	**2**	**1**	**4**	**1**	**2**	**6**	**3**	**1**	**2**

Two positions are replaced according to Table IV. But they should not be the same value. A decimal number is generated randomly to perform this operation. The produced value must be smaller than the mutation as a result of which the mutation process to be performed. After these operations obtained new antibodies are given in Table V. All the steps of proposed mechanism are given below.

- Set the initial parameters (number of population, number of iterations, cloning rate, cloning multiplier, choose the best rate, mutation rate),
- Generate random antibody to the number of population and initially give a value of 0 to index,
- Increase the number of index by 1 and calculate each antibody's makespan value,
- Sort the antibody with best makespan (at least) value in the population and specify the best antibodies as of clone's number,
- Reproduce the best clones as the number of clone multiplier and create the P(clone) set,
- Specify a value "a" , give 0 to the value and start the loop,
- Apply mutation process according to the requirement "Is produced random decimal smaller than the mutation rate?" to all clones set,
- Add antibody in the population as the number of the best selection from P(clone) and subtract antibody from list as the number of added,
- Create random antibodies as the number of remaining selection number and change the old antibodies in population with those produced,
- Update the entire population, sort from best to worst and specify the best of local antibody,
- "The best local makespan value is better than the best global makespan value?" update the best solution according to the requirement,
- If terminating criteria is satisfied, terminate the process and display the results. Otherwise, continue the loop.

V. EXPERIMENTAL RESULTS

The proposed clone-mutation mechanism using clonal selection algorithm applied to 5 different tests problem, Ft06, La01, La03, La04 and La05. These have been taken from *"http://people.brunel.ac.uk/~mastjjb/jeb/orlib/files/jobshop1.txt"* web site [13], [14]. Additionally, initial parameter value was selected as follow: number of population 100, number of iterations 1000, number of trials 5, cloning ratio 0.6, mutation rate 0.8, cloning multiplier 10 and the best selection ratio 0.7. On the other hand, when the length of every antibody is not the same therefore these values varies from test problem to another test problem. The mutation rate which is one of the parameters has been studied. The application has been tested for mutation rate of parameter values between 0 and 1. The results obtained by different mutation rates are given in Table VI.

Time completion of the works for all problems expresses the makespan. Table VI indicates the best makespan values of all test problems in bold. For all the problems best makespan results usually reaches the mutation rate values between 0.5 and 1.

TABLE VI: TEST RESULTS BY MUTATION RATES

Mutation Rate	Ft06	La01	La03	La04	La05
0,1	**55**	686	667	644	**593**
0,2	**55**	691	675	638	**593**
0,3	**55**	677	671	632	**593**
0,4	**55**	676	651	634	**593**
0,5	**55**	**666**	649	640	**593**
0,6	**55**	678	645	630	**593**
0,7	**55**	675	656	**624**	**593**
0,8	**55**	671	651	637	**593**
0,9	**55**	671	654	639	**593**
1	**55**	**666**	**640**	625	**593**

When we look at Table VI we can see that for Ft06 and La05 the best makespan values were obtained for all the values of mutation rate of test problems. At the same time the best makespan value has been obtained for La01. The best makespan values have been approached for La03 and La04 test problems.

The results obtained and the best known makespan values are given in Table VII.

TABLE VII: BEST MAKESPAN VALUES OBTAINED

CSA with Clone-Mutation Mechanism	Ft06	La01	La03	La04	La05
Best known Makespan Values	55	666	597	590	593
Obtained Makespan Values	55	666	640	624	593

VI. CONCLUSIONS

In this study, the CSA proposed to solve the job shop scheduling problems. In addition to this classical CSA study, a set of clone selection and mutation mechanism is proposed. In this method, the selected antibodies for set of clones were mutated. Instead of antibodies inside the receptor set, randomly generated antibodies were added to system. In this way instead of antibodies far from the best makespan value the new antibodies are produced. This will increase the diversity. Increasing the diversity will prevent the local hang-ups. After working through all the iterations of the proposed method of global best value, it will be labeled as the best makespan value.

After analyzing the results given in Table VII we can see that the value 55 for Ft06, 666 for La01 and 593 for La05 has been obtained. These values are the best known makespan values. The best makespan values have been approached for La03 and La04.

Application ran for every mutation parameter's value between 0 and 1 in clone-mutation mechanism. After examining the results in Table VI, we can see that the mutation rate of 0.5 to 1 is more sensible selection.

This study is important because of the implementation of CSA, which uses clone-mutation mechanism, in JSSP. Amendment to the current algorithm has provided many benefits in terms of obtaining the best makespan values.

REFERENCES

[1] M. R. Garey, D. S. Johnson, and R. Sethi, "The complexity of flowshop and jobshop scheduling," *Mathematics of Operations Research*, vol. 1, no. 2, pp. 117–129, 1976.

[2] E. M. Arkin and E. B. Silverberg, "Scheduling jobs with fixed start and end times," *Discrete Applied Mathematics*, vol. 18, pp. 1- 8, 1987.

[3] F. A. Ogbu and D. K. Smith, "The application of the simulated annealing algorithm to the solution of the n/m/Cmax flowshop problem," *Computers & Operations Research*, vol. 17, pp. 243-442, 1990.

[4] Y. N. Sotskov and N. V. Shakhlevich, "NP-hardness of shop-scheduling problems with three jobs," *Discrete Applied Mathematics*, vol. 59, pp. 237-266, 1995.

[5] H. Zhou, Y. Feng, and L. Han, "The hybrid heuristic genetic algorithm for job shop scheduling," *Computers & Industrial Engineering*, vol. 40, issue 3, pp. 191-200, 2001.

[6] J. H. Yang, L. Sun, H. P. Lee, Y. Qian, and Y. C. Liang, "Clonal selection based memetic algorithm for job shop scheduling problems," *Journal of Bionic Engineering*, vol. 5, no. 2, pp. 111–119, 2008.

[7] R. Murugesan, K. S. Balan, and V. N. Kumar, "Clonal selection algorithm using improved initialization for solving JSSP," in *Proc. IEEE International Conference on Communication Control and Computing Technologies*, 2010, pp. 470-475.

[8] M. Akhshabi, M. Akhshabi, and J. Khalatbari, "Solving flexible job-shop scheduling problem using clonal selection algorithm," *Indian Journal of Science and Technology*, vol. 4, no. 10, pp. 1248–1251, 2011.

[9] T. C. E. Cheng, B. Peng, and Z. Lü, "A hybrid evolutionary algorithm to solve the job shop scheduling problem," *Annals of Operations Research*, February 2013.

[10] S. Biroğul, "Genetik algoritma yaklaşımıyla atölye çizelgeleme," M.S. thesis, Institute of Science, Gazi University, Ankara, Turkey, 2005.

[11] L. N. de Castro and F. J. Von Zuben, *Artificial Immune Systems: Part I–Basic Theory and Applications*, Universidade Estadual de Campinas, Tech. Rep, December 1999.

[12] S. Parmaksızoğlu, "Yapay bağışıklık algoritması kullanılarak lineer sistemlerin kimliklendirilmesi," M.S. thesis, Institute of Science, Erciyes University, Kayseri, Turkey, 2005.

[13] H. Fisher and G. L. Thompson, "Probabilistic learning combinations of local job-shop scheduling rules," *Industrial Scheduling*, Prentice Hall, Englewood Cliffs, New Jersey, pp. 225-251, 1963.

[14] S. Lawrence, "Resource constrained project scheduling: an experimental investigation of heuristic scheduling techniques (Supplement)," Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1984.

Yılmaz Atay was born in Adana on May 18, 1987. He graduated from Computer Engineering Department of Selçuk University with B.Sc. in 2010. After graduation, he worked in private sector for a short amount of time. He achieved his M. Sc. in Computer Engineering Department of the same University in 2012. Now he is a research assistant of Computer Engineering at Selçuk University besides working on his Ph.D. degree. His research interests are Clonal Selection, Optimization, Artificial Intelligent Algorithm, Machine Vision and Embedded Systems.

Halife Kodaz was born in Sivas on April 15, 1977. He graduated from Computer Engineering Department of Selçuk University with B.Sc. degree in 1999, from Computer Engineering Department of Selçuk University with M.Sc. degree in 2002 and from Electrical-Electronics Engineering Department of Selçuk University with Ph.D. degree in 2008. He works as an assistant professor at Selçuk University. His research interests are artificial immune systems and machine learning algorithms.