World Academy of Science, Engineering and Technology
International Journal of Industrial and Manufacturing Engineering
Vol:6, No:5, 2012

The Design of Self-evolving Artificial Immune System II for Permutation Flow-shop Problem

Meng-Hui Chen, Pei-Chann Chang, Wei-Hsiu Huang

Abstract**—**Artificial Immune System is adopted as a Heuristic Algorithm to solve the combinatorial problems for decades. Nevertheless, many of these applications took advantage of the benefit for applications but seldom proposed approaches for enhancing the efficiency. In this paper, we continue the previous research to develop a Self-evolving Artificial Immune System II via coordinating the T and B cell in Immune System and built a block-based artificial chromosome for speeding up the computation time and better performance for different complexities of problems. Through the design of Plasma cell and clonal selection which are relative the function of the Immune Response. The Immune Response will help the AIS have the global and local searching ability and preventing trapped in local optima. From the experimental result, the significant performance validates the SEAIS II is effective when solving the permutation flows-hop problems.

Keywords**—**Artificial Immune System, Clonal Selection, Immune Response, Permutation Flow-shop Scheduling Problems

I. Introduction

PERMUTATION Flow-shop Scheduling Problems (PFSP) are regarded as one of the combinatorial problems. As the definition of the combinatorial problems, is belong to NP-hard. To solve optimization problems, a common challenge is which an algorithm may be trapped in the local optima of the objective function when the dimension is high and there are numerous local optima. Tsai et al. [1] and Chun et al. [2] has proposed the algorithms for global optimization problems are importance in many different areas which likes modern engineering design and systems operation. Genetic algorithm which is proposed by Holland [3] and Goldberg [4] is a tool based on biological mechanisms and natural selection theory, have paid much attention regarding its potential as an optimization technique for combinational optimization problems and have been successfully applied in many different areas. The main feature of the GAs as an optimization method is their implicit parallelism, which is a result of the evolutionary process. However, there are two major issues in GAs; one is lack of the global search ability and another is the premature convergence. Therefore, numerous of algorithms were proposed for solving the phenomenon. Initially, the improvements in the GAs have been sought in the optimal proportion and adaptation of the main parameters, namely probability of mutation, probability of crossover and population size. Recently, most researchers proposed many GA-based approaches to solve the phenomenon, one of these proposed GA-based algorithm is hybrid Genetic Algorithm and Immune System.

Meng-Huei Chen is with the Department of Information Management, Yuan Ze University, Taoyuan 32026, Taiwan, R.O.C.

Pei-Chann Chang is with the Department of Information Management, Yuan Ze University, Taoyuan 32026, Taiwan, R.O.C.

(Corresponding Author's E-mail: iepchang@saturn.yzu.edu.tw).

Wei-Hsiu Huang is with the Department of Information Management, Yuan Ze University, Taoyuan 32026, Taiwan, R.O.C.

The organisms named antibodies in the immune system which are responsible for protecting the body to against harmful organisms named antigen. Campelo [5] mentioned the immune system is able to detect a huge number of antigens using a fairly limited repertory of gene combinations. To carry out this recognition task, segments of genes are combined to accomplish the specificity of almost all the invader antigens known. A self-recognition task keeps the immune system from attacking itself, because immune cells are capable of recognizing themselves.

II. Literature review

A. Definition of Flow-shop Scheduling Problems

Flow-shops are useful tools in modeling manufacturing processes. A permutation flow-shop is a job processing facility, which consists of several machines and several jobs to be processed on the machines on different machines. In a permutation flow-shop, all jobs follow the same processing order. Our objective is to find a set of compromise solutions so that the makespan is minimized. The flow-shop scheduling problem is a typical assembly line problem where n different jobs have to be processed on m different machines. All jobs are processed on all the machines in the same order. The processing time of the jobs on machines are fixed regardless of the order in which the processing is conducted. The problem is characterized by a matrix $P = (p_{ij})$, $i = 1\ldots n$, $j = 1\ldots m$, of processing time. Each machine processes exactly one job at a time and each job is processed on exactly one machine at a time. The problem then is to find a sequence of jobs of minimizing the makespan which is the completion time of the last job in the sequence on the last machine. If *Ci* denotes the completion time for job *i*, we are trying to minimize max *Ci*. There are many other criteria that can be considered for the purpose of optimization. We refer the reader to Bagchi [14] for a detailed discussion of scheduling using GA. For details of the flow-shop and other scheduling and sequencing problems we refer the reader to Baker. The flow-shop scheduling can be formerly defined as follows: if $p\ (i, j)$ is the processing time for Job i on Machine j, and a job permutation $\{\pi_1, \pi_2, \ldots, \pi_n\}$, where there are n jobs and m machines, accordingly the completion times $C\ (\pi_i, j)$ is calculated as follows which are proposed by Reeves [13]:

$$C\ (\pi_1, 1) = p\ (\pi_1, 1) \quad (1)$$

$$C\ (\pi_i, 1) = C\ (\pi_{i-1}, 1) + p\ (\pi_i, 1),\ for\ i = 2,\ldots, n \quad (2)$$

$$C\ (\pi_1, j) = C\ (\pi_1, j\text{-}1) + p\ (\pi_1, j),\ for\ j = 2,\ldots, m \quad (3)$$

$$C\ (\pi_i, j) = \max\{C\ (\pi_{i-1}, j), C\ (\pi_i, j\text{-}1)\} + p\ (\pi_i, j) \quad (4)$$

$$for\ i = 2,\ldots, n;\ for\ j = 2,\ldots, m$$

The makespan is finally defined as:

$$C\max(\pi) = C(\pi_n, m) \quad (5)$$

Subsequently, the objective is to find a permutation π^* in the set of all permutations $\prod$ so that

$$Cmax(\pi^*) \leqq Cmax(\pi) \ \forall \pi \in \prod \quad (6)$$

A more general flow-shop scheduling problem can be defined by allowing the permutation of jobs to be different on each machine. However, what work has been done to show on the more general flow-shop scheduling problem has tended to small improvement in solution quality over the permutation flow-shop scheduling problems (PFSP) while increasing the complexity of the problem substantially. The size of the solution space increases from $n!$ to $(n!)^m$. Other objective functions for the PFSP also received a lot of attention. For example, the mean flow-time (the time a job spends in the process), or the mean tardiness (assuming some deadline for each job) are to be minimized.

Other real problems from the manufacturing industries such as their jobs may have non-identical release dates, and there may be sequence-dependent setup times, and limited buffer storage between machines and so on. These characteristics of the real world problems will make the problem more complicated to be solved within a reasonable time frame. However, GA approaches provide a more realistic view to the problem. Since it can generate alternatives of sequences (in the evolving process, each chromosome represents a feasible solution to the problem) to the decision maker, a more applicable sequence can be decided to solve the current problem with satisfactory results.

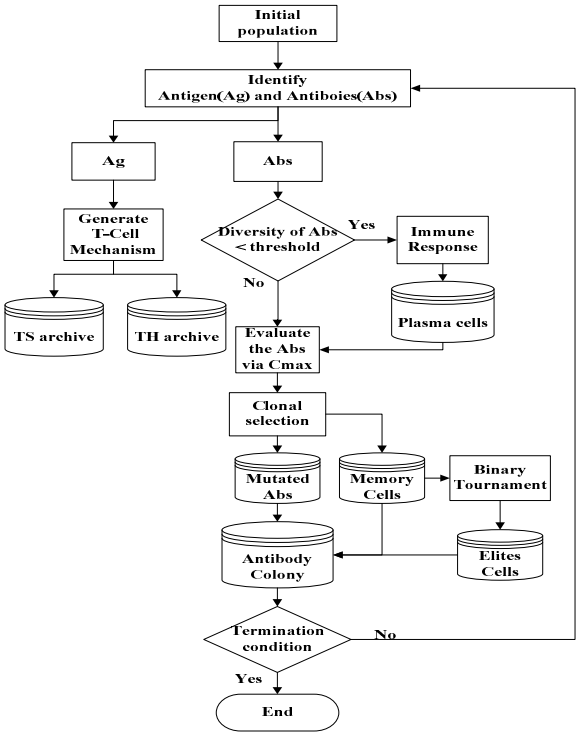
B. AIS relative researches

In the application of computational intelligence, AIS is usually been adopted for solving the combinatorial problems. Chang et al. [6] mentioned HGIA contains two phases. The first part is to take advantage of GA to fast converge. When HGIA trapped in local optima or cannot converge anymore, the pure AIS stage will be active. In the stage of AIS, the B-cell is responsible for continuing evolution. Nevertheless, T-cell is responsible to offer the information and diversified artificial chromosome to enhance the diversity of the population for escaping the local optima. Tan et al. [7] ever propose an algorithm which used the GA and the improved clonal selection to combine as AIS for solving the multiple objective problems (MOPs). The above researches did not have a special design of AIS viewpoint but establish the collaboration model with GA. Therefore, this research aims at developing the AIS mechanism which including the T and B cell. We also discuss the design of Plasma cell and clonal selection which are relative the function of the Immune Response. The design of the Immune Response will help the AIS have the global and local searching ability and preventing trapped in local optima.

III. Self-evolving Artificial Immune System

Artificial immune system is an algorithm which simulates the immune system, with different evolutionary strategies to identify the unknown virus which is so-called Antigen. The antibodies are regarded as the protectors which are responsible to eliminate the antigen. Therefore, in this paper, the antigen is defined as the best found solution and accepts the challenge from the antibodies. In other words, the winners of the antibodies are defined as the antigens in iterations.

As shown in Fig. 1, when the initial solutions are divided to Antigen and Antibodies, T-cell mechanism starts to analyze the composition information. There will be two cells be generated which are TH and TS. TH is responsible to speed up the identification the evolutionary information of Antigen. TS contains the diversity of the composition.



These two components will act different mechanism which helps the evolution process according to different strategies. The new composited cell with the injection of TH or TS is the cell who are responsible to generate the new Antibodies which with the identification ability and diversity, is so-called Plasma cell in this research.

The definition of B-cell is responsible to collect the effective Antibodies which is also called the Memory cell. B-cell contains two parts which actives the Mutation mechanism when the worse Antibodies are met. In another hand, the Antibodies with the top 10 performance will be considered to store in B-cell for the next Antigen. This research adopts the respective evolutionary strategy to search the different Antibodies for widespread identification of Antigen. Meanwhile, the design of the collaboration of T and B cell helps to identify the Antigens.

The pseudo codes of SEAIS II are described in the following:

1. Initial population();
2. While stopping criterion is not satisfied
3. Define the best antibody from mutated antibodies,

memory cells and elite cells as antigen, and the others as antibodies.
4. Generate T-Cells(antigens) by dominance_matrix as TH, and Generate T-Cells(antigens) by complementary_matrix as TS.
5. If diversity of antibodies < threshold then
   If affinity of antibodies < threshold then
      Immune_response(antibodies, TH);
   Else
      Immune_response(antibodies, TS);
   Endif
   Collect these antibodies to plasma cells archive.
  Endif
6. Sorting antibodies by Cmax and classify them as mutated antibodies, memory cells and elite cells.
7. Collect them as antibody colony.
8. Endwhile

A. Affinity

Affinity chromatography is a method of separating biochemical mixtures and based on a highly specific interaction such as that between antigen and antibody, enzyme and substrate, or receptor and ligand. Chang et al [18] proposed the approach for evaluation of the Affinity.

$$Affinity_{cell} = C_{\max}^{-1} \tag{7}$$

In this paper, we proposed one more approach to evaluate the Affinity which adopts the Hamming distance to evaluate the Affinity of the linkage is represented as follows:

$$Affinity_{linkage} = \text{HammingDistance} / C_{\max} \tag{8}$$

Here, we make an assumption which defines the higher affinity between two cells, the higher identification. According to the index of the Affinity, we can take of advantage to develop the survival strategy for keeping the good Antibodies. The index via the completion time and Hamming Distance not just used to evaluate the similarity between Antibodies, but more specified evaluation the fitness relationship in different stage and the influence to the whole Antibodies.

B. Immune response

Antibodies use swap mutation for evolution strategy in B cells mechanism, and immune response mechanism will help antibodies' evolution under the two situations. One is the evolution inefficient; the definition in our research is according to the equation (9).

$$\theta = \Delta makespan / \Delta generation \; < \; threshold_1 \tag{9}$$

Another is the evolution into the local optimization; the definition in our research is according to the equation (10).It means if antigen is not update for a several generations, the evolution may be in local optimization.

$$N_{non-update\ Ag} > threshold_2 \tag{10}$$

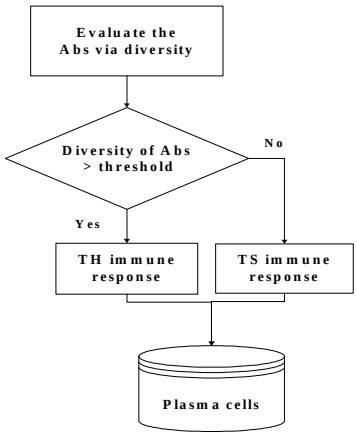


Fig. 2 Architecture of Immune response

When the diversity of antibodies is not enough in immune response process, the immune response mechanism will provide TS cells for antibodies in order to improve the diversity of antibodies, and this will help to escape into local optimization. If antibodies have diversities, it means antibodies don't have enough information of antigens. So, immune response will provide TH cells for antibodies making the convergence with more efficiency.

C. T cell

The probability matrix is updated according to the orders of the Antigen in iterations. T-cell in this paper is responsible to fine the dominate "linkage information" from the matrix and define these as the Gene-Linkage, shown as Fig. 3. In the mechanism of T-cell, TS is composited by TH and TS. TH is defined as the helper cell. Therefore, we assign those Gene-Linkage with dominate information to TH. In other hand, TS is defined the suppressor cell, is applied to enhance the diversity for escaping the local optima.

D. B cell

The principal functions of B cells are to make antibodies against antigens, perform the role of antigen-presenting cells (APCs) and eventually develop into memory B cells after activation by antigen interaction. B cells are an essential component of the adaptive immune system. B-cell is consists of mutated antibodies, memory cells and elites, which are from the mutation strategy, memory cell and the elite cells.

As shown in Fig. 4, the three different types of Antibodies has individual function. Mutated antibodies represent the possibility by taking advantage of Mutation to escape the local optima. Memory cells represent the collections of near iterations. Elites is responsible for storing the top 10 Antigen for the elite strategy.

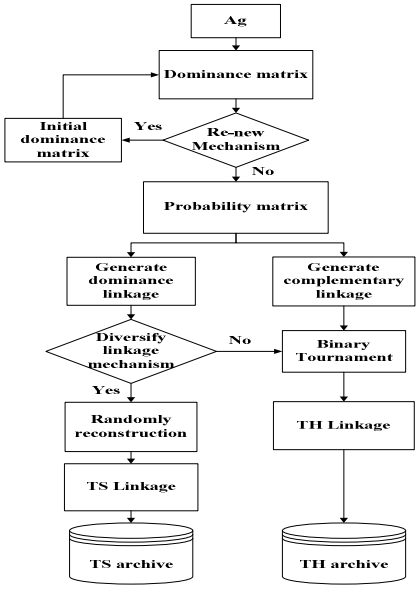


Fig. 3 Architecture of T-Cell

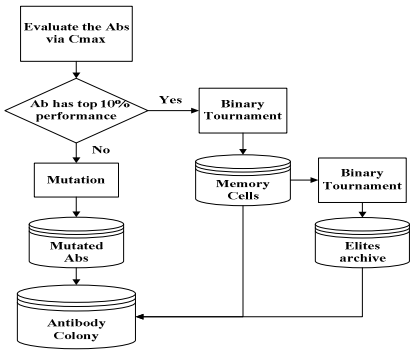


Fig. 4 Architecture of B-Cell

IV. Experimental Results

This research adopted the instances of Reeves and Taillard in OR-Library to validate the performance. Each instance is executed for 30 runs, n represents the job number and m represents the machine number. Table I is given for the experimental date for Taillard's instances, the comparison is based on Zhang et al. [17]. CDPSO-R and CDPSO is with the local search, however, CDPSO-R-A is the standard approach of CDPSO. Due to SEAIS II does not uses the strategy of local search. That is to say, SEAIS II outperform CDPSO-R-A on the eight instances.

TABLE I
Performance Comparison Of Taillard's Instance

			SEAIS II		CDPSO-R-A		CDPSO-R		CDPSO	
Ins.	n,m	opt	Min	Error_rate	Min	Error_rate	Min	Error_rate	Min	Error_rate
ta005	20,5	1235	1243	0.65%	1235	0.00%	1235	0.00%	1235	0.00%
ta010	20,5	1108	1108	0.00%	1108	0.00%	1108	0.00%	1108	0.00%
ta020	20,10	1591	1599	0.50%	1608	1.07%	1608	1.07%	1591	0.00%
ta030	20,20	2178	2185	0.32%	2181	0.14%	2181	0.14%	2178	0.00%
ta050	50,10	3065	3157	3.00%	3120	1.79%	3120	1.79%	3120	1.79%
ta060	50,20	3696	3905	5.65%	3870	4.71%	3870	4.71%	3831	3.65%
ta070	100,5	532	534	0.38	532	0.11	532	0.11	532	0.11
ta080	100,10	5845	5902	0.99%	5903	0.99%	5903	0.99%	5860	0.26%
Avg.				1.44%		2.29%		1.10%		0.73%

Table II shows the performance comparison on Taillard's instances. The comparison standard is based on Chang et al. [18]. From the result, the average error rate is 0.64%, outperforms the other algorithms. From the result in Table I and Table II, SEAIS II has good performance even the stopping criteria are different.

TABLE II
Performance Comparison Of Taillard's Instance

			AC2GA		HGIA		PSO-Lian		SEAIS II	
Ins.	n,m	opt	Min	Error_rate	Min	Error_rate	Min	Error_rate	Min	Error_rate
ta005	20,5	1235	1235	0.00%	1235	0.00%	1235	0.00%	1235	0.00%
ta010	20,5	1108	1108	0.00%	1108	0.00%	1108	0.00%	1108	0.00%
ta020	20,10	1591	1617	1.63%	1598	0.44%	1617	1.63%	1598	0.44%
ta030	20,20	2178	2196	0.83%	2186	0.37%	2196	0.83%	2186	0.37%
ta050	50,10	3065	3171	3.46%	3111	1.50%	3171	3.46%	3111	1.50%
ta060	50,20	3696	3910	5.79%	3823	3.44%	3910	5.79%	3823	3.44%
ta070	100,5	5322	5324	0.04%	5328	0.11%	5324	0.04%	5328	0.11%
ta080	100,10	5845	5893	0.82%	5848	0.05%	5893	0.82%	5848	0.05%
Avg.				1.57%		0.74%		1.57%		0.74%

Table III shows the result for testing the Reeves's instances. The comparison standard is based on Chang et al. [6]. The average error rate is 0.6%, outperforms the three compared algorithms. From the result, SEAIS II performs the effective searching ability on different problems and complexities.

TABLE III
Performance Comparison Of Reeves's Instance

			SEAIS II		SGA		AC2GA		HGIA	
Ins.	n,m	opt	Min	Error_rate	Min	Error_rate	Min	Error_rate	Min	Error_rate
Rec01	20,5	1247	1249	0.00%	1249	0.16%	1249	0.16%	1247	0.00%
Rec03	20,5	1109	1109	0.00%	1111	0.18%	1109	0.00%	1109	0.00%
Rec05	20,5	1242	1245	0.24%	1245	0.24%	1245	0.24%	1245	0.24%
Rec07	20,10	1566	1566	0.00%	1583	1.09%	1566	0.00%	1566	0.00%
Rec09	20,10	1537	1537	0.00%	1565	1.82%	1537	0.00%	1537	0.00%
Rec11	20,10	1431	1431	0.00%	1456	1.75%	1431	0.00%	1431	0.00%
Rec13	20,15	1930	1932	0.10%	1970	2.07%	1930	0.00%	1932	0.10%
Rec15	20,15	1950	1954	0.00%	1990	2.05%	1951	0.05%	1951	0.05%
Rec17	20,15	1902	1902	0.00%	1960	3.05%	1902	0.00%	1924	1.16%
Rec19	30,10	2093	2099	0.29%	2162	3.30%	2099	0.29%	2103	0.48%
Rec21	30,10	2017	2034	1.44%	2064	2.33%	2021	0.20%	2046	1.44%
Rec23	30,10	2011	2020	0.45%	2075	3.18%	2021	0.50%	2020	0.45%
Rec25	30,15	2513	2525	0.88%	2623	4.38%	2515	0.08%	2530	0.68%
Rec27	30,15	2373	2391	0.25%	2461	3.71%	2387	0.59%	2394	0.88%
Rec29	30,15	2287	2307	0.52%	2392	4.59%	2289	0.09%	2301	0.61%
Rec31	50,10	3045	3075	1.31%	3207	5.32%	3101	1.84%	3077	1.05%
Rec33	50,10	3114	3138	0.51%	3162	1.54%	3131	0.55%	3115	0.03%
Rec35	50,10	3277	3277	0.00%	3280	0.09%	3277	0.00%	3277	0.00%
Rec37	75,20	4951	5075	2.32%	5251	6.06%	5190	4.83%	5084	2.69%
Rec39	75,20	5087	5189	1.65%	5301	4.21%	5285	3.89%	5150	1.24%
Rec41	75,20	4960	5100	2.58%	5263	6.11%	5164	4.11%	5097	2.76%
Avg.				0.60%		2.73%		0.83%		0.66%

V. Conclusion

In the mechanism of SEAIS II, B-cell polarizes different Antibodies. In different evolutionary strategies, the Immune Response offers different co-evolutionary effectiveness according to different evolutionary requirements. From the experimental result, SEAIS II is validated has good convergence speed in small instances. Furthermore, SEAIS II has good performance to escape the local optima in the large instances. The future research will focus on the linkage of T-cell and enhance the effectiveness of the produced Plasma cell from the combination of T and B cells.

References

[1] J. T. Tsai, W. H. Ho, T. K. Liu, and J. H. Chou, "Improved immune algorithm for global numerical optimization and job-shop scheduling problems," Applied Mathematics and Computation, Vol. 194, pp. 406-424, Dec. 2007.

[2] J. S. Chun, H. K. Jung and S. Y. Hahn, "A Study on Comparison of Optimization Performances between Immune Algorithm and other Heuristic Algorithms," IEEE Transactions on Magnetics, vol. 34, No. 5, Sept. 1998.

[3] J. H. Holland, "Genetic Algorithms and the Optimal Allocation of Trials," SIAM J. Comput, vol. 2, pp. 88-105, 1973.

[4] D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning (Book style). Boston, MA: Addison-Wesley, 1989.

[5] F. Campelo, F. G. Guimar˜aes, and H. Igarashi, "Overview of Artificial Immune Systems for Multi-objective Optimization," Lecture Notes in Computer Science, vol. 4403 , pp. 937-951, 2007.

[6] P. C. Chang, W. H. Huang, and C. J. Ting, "A hybrid genetic-immune algorithm with improved lifespan and elite antigen for flow-shop scheduling problems," International Journal of Production Research, vol. 49, pp. 937-951, Sept. 2007.

[7] K. C. Tan, C. K. Goh, A. A. Mamun, and E. Z. Ei, "An evolutionary artificial immune system for multi-objective optimization," European Journal of Operational Research, vol. 187, pp. 371-392, Jun. 2008.

[8] T. Bagchi, Multiobjective Scheduling by Genetic Algorithms (Book style). New York, 1999.

[9] C. R. Reeves, "A Genetic Algorithm for Flowshop Sequencing," Computers and Operations Research, vol. 5, pp. 5-13, Jan. 1995.

[10] J. D. Farmer, N. Packard, and A. Perelson, "The immune system, adaptation and machine learning," Physica D, vol. 2, pp. 187-204, Oct.-Nov. 1986.

[11] H. Bersini, and F. J. Varela, "Hints for adaptive problem solving gleaned from immune networks," Parallel Problem Solving from Nature, vol. 496, pp. 343-354, 1991.

[12] J. O. Kephart, "A biologically inspired immune system for computers," Proceedings of Artificial Life IV: The Fourth International Workshop on the Synthesis and Simulation of Living Systems, MIT Press. pp. 130–139, 1994.

[13] D. Dasgupta, Artificial Immune Systems and Their Applications (Book style). Berlin : Springer-Verlag, Jan. 1999.

[14] L. N. de Castro and F. J. Von Zuben, Artificial Immune Systems: Part I -Basic Theory and Applications (Book style). Brazil, 1999.

[15] V. Cutello and G. Nicosia, "An Immunological Approach to Combinatorial Optimization Problems," Lecture Notes in Computer Science, vol. 2527, pp. 361-370, 2002.

[16] V. Cutello, G. Nicosia, M. Pavone, and J. Timmis, "An Immune Algorithm for Protein Structure Prediction on Lattice Models," IEEE Transactions on Evolutionary Computation, vol. 11 , pp. 101-117, 2007.

[17] J. Zhang, C. Zhang, and S. Liang, "The circular discrete particle swarm optimization algorithm for flow shop scheduling problem," Expert Systems with Applications, vol. 37, pp. 5827-5834, 2010.

[18] P. C. Chang, W. H. Huang, and C. J. Ting, "Self-evolving Artificial Immune System via Developing T and B Cell for Permutation Flow-shop Scheduling Problems," Proceedings of World Academy of Science, Engineering and Technology, vol. 65, pp. 822-827, May 2010.