Artificial Immune Algorithms for University Examination Timetabling

Muhammad Rozi Malim
Faculty Info. Technology &
Quantitative Science, UiTM
40450 Shah Alam, Malaysia
rozi@tmsk.uitm.edu.my

Ahamad Tajudin Khader
School of Computer Science
Univ. of Science Malaysia
11800 Penang, Malaysia
tajudin@cs.usm.my

Adli Mustafa
School of Math. Science
Univ. of Science Malaysia
11800 Penang, Malaysia
adli@cs.us

ABSTRACT

The examination timetabling is known to be a highly constrained optimization problem. Metaheuristic approaches (simulated annealing, tabu search, and evolutionary algorithms) have successfully been applied to solve the problem. The artificial immune algorithms, algorithms inspired by the immune systems, have successfully been applied to anomaly detection, pattern recognition, computer security, fault tolerance, dynamic environments, robotics, data mining, optimization and scheduling. This paper presents three artificial immune algorithms for examination timetabling; clonal selection, immune network and negative selection. The main objective is to compare the effectiveness of the algorithms on examination timetabling problems. The experimental results, using benchmark datasets, have shown that all algorithms have successfully produced good examination timetables on all datasets. The clonal selection and negative selection algorithms are more effective than immune network algorithm in producing good quality examination timetables; however, the immune network algorithm runs faster than clonal selection and negative selection. For future work, these three algorithms will be applied to university course timetabling.

KEYWORDS

Artificial Intelligence, Evolutionary Algorithms, Examination Timetabling, Artificial Immune Algorithms.

1. Introduction

The *examination timetabling* is known to be a highly constrained combinatorial optimization problem. *Metaheuristic* approaches such as simulated annealing (SA) [1], tabu search (TS) [6], and evolutionary algorithms (EAs) have successfully been applied to solve the problem. The most popular EAs for timetabling problems are genetic algorithm (GA) [9] and memetic algorithm (MA) [2].

Artificial immune system (AIS), a new branch of Artificial Intelligence [3], is a new intelligent problem-solving technique that being used in optimization and scheduling problems [11]. AISs have been more successful than GA and other methods in pattern recognition, computer security, and dynamic tasks scheduling due to the applicability features of natural immune systems (IS) [3]. Furthermore, the solutions produced by the AIS are observed to be *robust* than solutions produced by a GA [12].

This paper presents *three* artificial immune algorithms for examination timetabling; *clonal selection algorithm* (CSAET), *immune network algorithm* (INAET), and *negative selection algorithm* (NSAET). The main objective is to compare the *effectiveness* of these algorithms in solving examination timetabling problems (ETPs). Twelve benchmark datasets have been used to implement and compare the algorithms. The experimental results have significantly shown the effectiveness of the three algorithms; all algorithms have successfully produced good quality examination timetables with low fitness values in most of the datasets. CSAET and NSAET are more effective than INAET in producing low fitness values for examination timetabling problems; however, INAET runs faster than CSAET and NSAET. The rates of convergence of the three algorithms are approximately equal for most of the datasets.

2. Examination Timetabling Problem

The examination timetabling regards the scheduling for the exams of university courses, avoiding overlap of exams having common students, and spreading the exams for the students as much as possible [6]. Given a set of exams, a set of timeslots, a set of students, and a set of student enrollments to exams, the problem is to assign exams to timeslots subject to a variety of *hard* and *soft* constraints.

Hard constraints must be satisfied in order to produce a *feasible* timetable. The main hard constraints in ETPs are:

i) Every exam in the set must be assigned to exactly one timeslot of the timetable.
ii) No individual should be timetabled to be in two different places at once.
iii) There must be sufficient resources available in each timeslot for all the exams timetabled.

Individual institutions may have their own specialized hard constraints based on their needs and requirements. Any timetable fails to satisfy these constraints is deemed to be *infeasible*.

Soft constraints are generally more numerous and varied, and far more dependent on the needs of the individual problem than the more obvious hard constraints. The violation of soft constraints should be minimized. The soft constraints define how good a given feasible solution is so that different solutions can be compared and improved via a *fitness* function. The common soft constraints in ETPs are:
i) Spreading exams - students should not have exams in consecutive (adjacent) timeslots.
ii) Time assignment - an exam may need to be scheduled in a specific timeslot.
iii) Time constraints - an exam may need to be scheduled before, after or at the same time as another.
iv) Resource assignment - an exam must be scheduled into a specific room.

The ETP can be seen as consisting of *two* subproblems [1]; assigning exams to timeslots, and assigning exams (with timeslots) to rooms. For real-life situations, these two subproblems can be solved separately.

3. Artificial Immune System and Artificial Immune Algorithms

The 'artificial immune system' is an approach which used the natural IS as a metaphor for solving computational problems, *not* modeling the IS [16]. The main application domains of AIS are anomaly detection, pattern recognition, computer and network security, fault tolerance, dynamic environments, robotics, data mining, optimization, and scheduling.

The 'immune system' can be considered to be a remarkably efficient and powerful information processing system which operates in a highly parallel and distributed manner [10]. It contains a number of features which potentially can be adapted in computer systems; recognition, feature extraction, diversity, learning, memory, distributed detection, self-regulation, threshold mechanism, co-stimulation, dynamic protection, and probabilistic detection. From the perspective of information processing, it is unnecessary to replicate *all* of these aspects of the IS in a computer model, rather they should be used as general guidelines in designing a system.

There are a number of different immune algorithms that can be applied to many domains [4]. These algorithms were inspired by works on theoretical immunology and several processes that occur within the IS. The AISs lead to the development of different techniques, each one mapping a different mechanism of the system. For examples, *Artificial Immune Networks* as proposed by Farmer et al. [7], *Clonal Selection Algorithm* proposed by de Castro and Von Zuben [5], and *Negative Selection Algorithm* introduced by Forrest et al. [8]. Immune network models are suitable to deal with dynamic environments and optimization problems, algorithms based upon the clonal selection principle are adequate to solve optimization and scheduling problems, and the negative selection strategies are successfully applied to anomaly detection.

3.1 Clonal Selection Algorithm for Examination Timetabling (CSAET)

The clonal selection algorithm is inspired by the immunological processes of *clonal selection* and *affinity maturation*. When an antigen is detected, those antibodies that best recognize this antigen will proliferate by cloning. This process is called *clonal selection principle* [5]. The principle explains how the IS 'fights' against an antigen. When a bacterium invades our organism, it starts multiplying and damaging our cells. One form the IS found to cope with this replicating antigen was by replicating the immune cells successful in recognizing and fighting against this antigen. Those cells reproduce themselves asexually in a way proportional to their degree of recognition; the better the antigenic recognition, the higher the number of clones. During the process of cell division (reproduction), individual cells suffer a mutation that allows them to become more adapted to the antigen recognized. The algorithm may be illustrated as a flow diagram in Figure 1.

The main operators in CSAET are *selection*, *cloning*, and *mutation*. A high affinity timetable is randomly selected for cloning using Roulette Wheel and, on average, a number of clones that equal to half of the population size are generated. Almost all clones will be mutated to produce new feasible timetables for the next generation since there is a high mutation rate for each clone. But only new timetables with high affinity will be selected to replace the low affinity timetables in the current population. The reproduction process (selection, cloning and mutation) will be repeated until the stopping criteria are met (maximum number of generations or maximum number of none improvement generations).

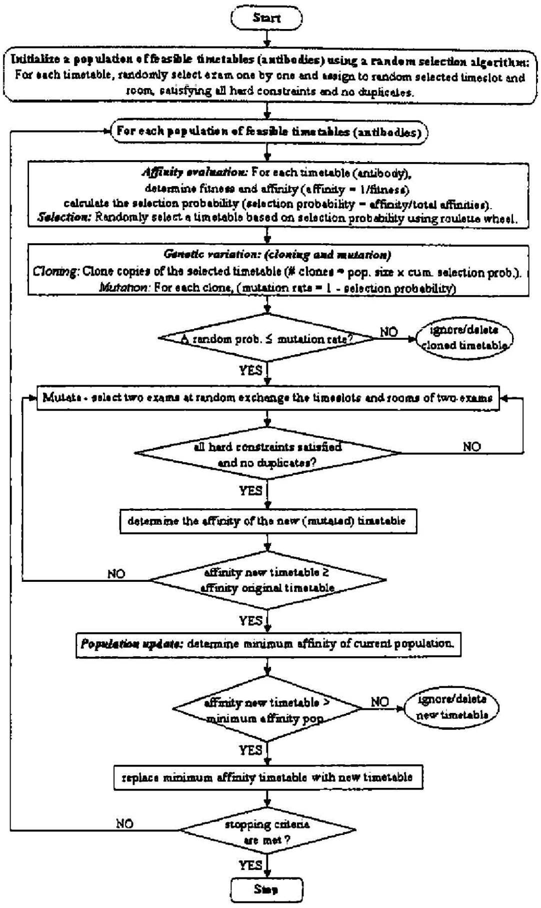

Figure 1: Clonal Selection Algorithm for Examination Timetabling (CSAET)

Malim et al. [14] has implemented CSAET on twelve benchmark datasets (Carter datasets). They have concluded that the algorithm is an effective optimization algorithm, capable of producing good quality examination timetables as good as MA and GA.

3.2 Immune Network Algorithm for Examination Timetabling (INAET)

The immune network algorithm is based on *Jerne's network theory* [13]. According to this theory, immune cells have portions of their receptor molecules that can be recognized by other immune cells in a way similar to the recognition of an invading antigen. This results in a network of recognition between immune cells. When an immune cell recognizes an antigen or another immune cell, it is stimulated. On the other hand, when an immune cell is recognized by another immune cell, it is suppressed. The sum of the stimulation and suppression received by the network cells, plus the stimulation by the recognition of an antigen corresponds to the stimulation level *S* of a cell. The algorithm may be illustrated as a flow diagram in Figure 2.

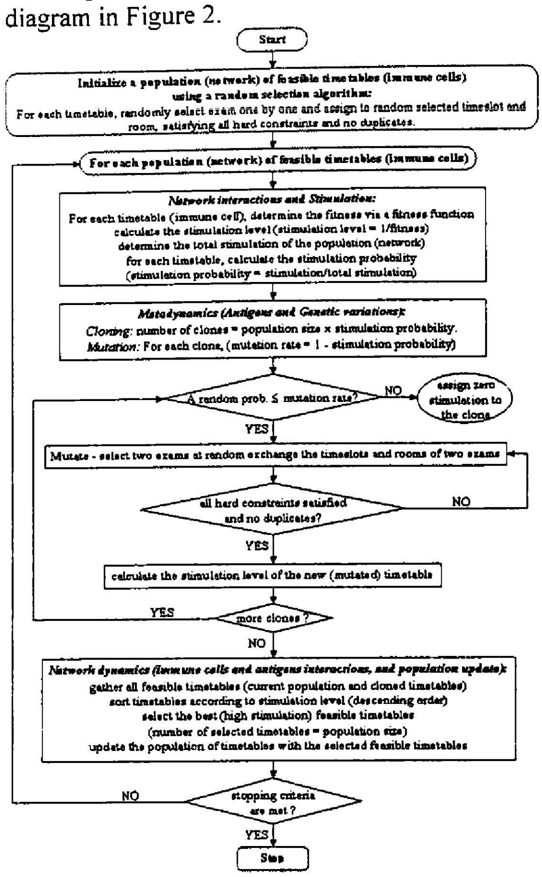

Figure 2: Immune Network Algorithm for Examination Timetabling (INAET)

The main operators in INAET are *cloning* and *mutation*. All timetables are selected for cloning and, on average, only one clone is generated for each timetable. Almost all clones will be mutated since the mutation rate is high for each clone. All feasible timetables, current population and mutated clones, are gathered, but only those with high stimulation will be selected to form a new population for the next generation. The reproduction process (cloning and mutation) will be repeated until the stopping criteria are met (maximum number of generations, or maximum number of none improvement generations).

3.3 Negative Selection Algorithm for Examination Timetabling (NSAET)

The negative selection algorithm is one of the most widely used techniques in AISs. It is primarily used to detect changes in data behavior patterns by generating detectors in the complementary space. The algorithm is based on the principles of *self-nonself discrimination* [8]. The algorithm was inspired by the *thymic* negative selection process that intrinsic to natural IS, consisting of screening and deleting selfreactive T-cells (T-cells that recognize self cells). The algorithm may be illustrated as a flow diagram in Figure 3.

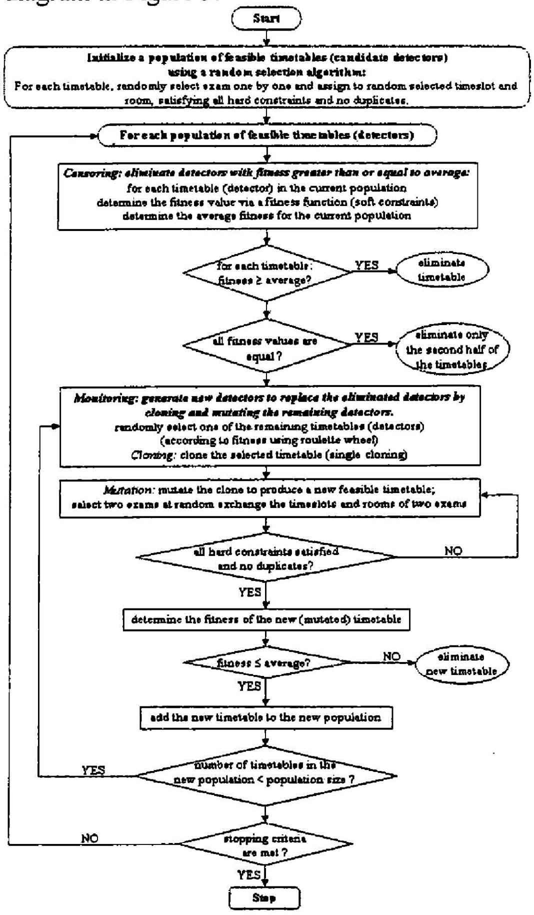

Figure 3: Negative Selection Algorithm for Examination Timetabling (NSAET)

The main operators in NSAET are negative deletion (*censoring*), cloning and mutation (*monitoring*). The timetables with fitness (affinity) greater (less) than or equal to average fitness (affinity) are deleted from the current population. A low fitness timetable is randomly selected from the remaining timetables for cloning and mutation using Roulette Wheel selection method. All clones will be mutated to produce new feasible timetables. For each new (mutated) timetable, if the fitness is less than or equal to average, the timetable will be added to the new population for the next generation; otherwise, it will be deleted. The monitoring process (cloning and mutation) will be repeated until the number of feasible timetables in the new population is equal to population size. The censoring and monitoring processes will be repeated until the stopping criteria are met (maximum number of generations, or maximum number of none improvement generations).

Malim et al. [15] has implemented NSAET on twelve Carter datasets. The experimental results have successfully shown the effectiveness of the algorithm by producing good quality exam timetables, as good as metaheuristic approaches.

4. Benchmark Datasets

The *twelve* Carter examination timetabling datasets used in the implementation of the three artificial immune algorithms are available from ftp://ftp.mie.utoronto.ca/pub/carter/testprob/.

These datasets provide a reasonable benchmark problems for comparing different examination timetabling algorithms and methods. The datasets are shown in Table 1.

Institution	No. of Exams	No. of Students	No. of Enrollments	Timeslot Capacity
car-f-92	543	18419	55522	2000
car-s-91	682	16925	56877	1550
ear-f-83	190	1125	8109	350
hec-s-92	81	2823	10632	650
kfu-s-93	461	5349	25113	1955
lse-f-91	381	2726	10918	635
rye-s-93	486	11483	45051	2055
sta-f-83	139	611	5751	465
tre-s-92	261	4360	14901	655
uta-s-92	622	21266	58979	2800
ute-s-92	184	2750	11793	1240
yor-f-83	181	941	6034	300

Table 1: Carter Examination Datasets

Each of the datasets come in two files; *course data file* and *student data file*.

5. Comparing Three Artificial Immune Algorithms on Examination Datasets

The three artificial immune algorithms (CSAET, INAET, and NSAET) have been implemented on the twelve datasets (Carter datasets). The main objective is to compare the *effectiveness* of the three algorithms on ETPs. The following (Table 2) are the experimental results on solving ETPs using the three algorithms. Each algorithm was run on each dataset for *five* trials; the maximum number of generations 500, and the maximum number of none improvement generations 100 were used. The *best fitness*, the *average fitness* and the *average CPU time* (in seconds) for each algorithm on each dataset, based on five trials, have been recorded.

The *number of timeslots* used for all datasets were imposed according to those given by Carter's website. The *fitness value* (soft constraint violations) is the minimum number of students having two exams in adjacent timeslots at generation 500 or less. For the *best fitness*, both CSAET and NSAET have achieved the first position in *five* datasets, while INAET has achieved the first position in only *two* datasets. The best fitness for CSAET have converged to '0' in *two* datasets, INAET in *three* datasets, and NSAET in *one* dataset. For the *average fitness*, both CSAET and NSAET has achieved the first position in *six* datasets, and INAET in only *one* dataset. Finally, for the *average CPU time*, INAET has achieved the first position in *nine* datasets, *two* for CSAET and *one* for NSAET.

Institution	No. of Timeslots	*Fitness Values*		
		CSAET	***INAET***	***NSAET***
		Best / *Average* / *Ave. time*	*Best* / *Average* / *Ave. time*	*Best* / *Average* / *Ave. time*
car-f-92	31	**285**	406	386
		466.6	455.2	**432.8**
		310.6s	**249.8s**	359.4s
car-s-91	40	535	554	**439**
		569.6	582.8	**486.2**
		512.6s	**399.6s**	484s
ear-f-83	24	**17**	65	74
		48	112.8	118.8
		75.4s	**34.8s**	88s
hec-s-92	19	3	**0** (271)	5
		11	**9.8**	14.4
		17.2s	**7.8s**	11.4s
kfu-s-93	20	35	16	**2**
		69.4	32.6	**13.6**
		172.4s	202.2s	240.2s
lse-f-91	18	45	**34**	115
		68.8	82.6	167.2
		132.4s	**120.8s**	147s
rye-s-93	24	**143**	217	180
		240.2	309	327.6
		233.8s	247s	336.4s
sta-f-83	14	0 (196)	0 (185)	**0** (160)
		0	0.4	**0**
		12.2s	11.4s	**9.6s**
tre-s-92	25	**27**	58	56
		36.8	70.2	79.2
		110s	**57.4s**	134s
uta-s-92	32	436	374	**165**
		487.6	436	**244.6**
		343.2s	**307.4s**	387s
ute-s-92	10	**0** (352)	0 (454)	1
		0.4	2.6	9.8
		34.8s	**26.8s**	34.4s
yor-f-83	22	3	24	**1**
		8	33.2	**6.6**
		62.6s	**30.4s**	63.2s

Table 2: Comparing Three Artificial Immune Algorithms on Examination Datasets

Hence, it may be concluded that CSAET and NSAET are equally effective in producing good quality timetables (low fitness) for ETPs, and both are more effective than NSAET. In terms of CPU times, INAET runs faster than CSAET and NSAET. However, the most important factor for a good optimization algorithm is the capability of finding the best or optimum solution in a reasonable time. The results from 12 different examination datasets have significantly shown the effectiveness of the three algorithms. All algorithms have successfully produced good quality examination timetables with low fitness values in most of the datasets.

6. Comparing the Convergence of the Three Algorithms

In this section, the results from the *fifth* trial of the three algorithms are plotted against each other (*generation vs fitness*) for each dataset (Figure 4). The objective is to compare the rates of convergence of the algorithms.

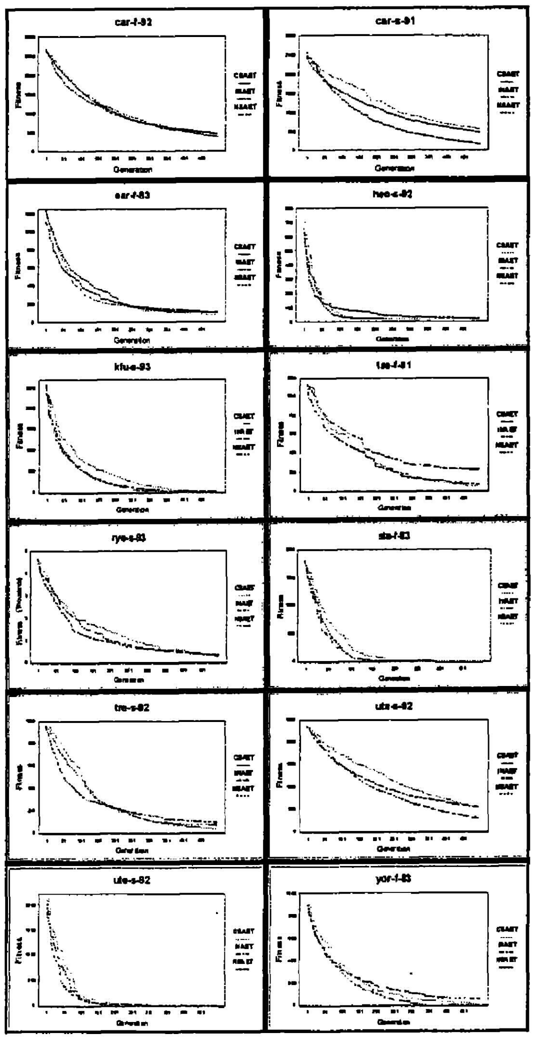

Figure 4: Fitness Values of Three Algorithms plotted against each other for each dataset

From the plots in Figure 4, the rates of convergence of the three algorithms are approximately equal for most datasets. However, for dataset car-s-92, the rate of convergence of INAET is much better than CSAET and NSAET; for dataset lse-f-91, the rates of convergence of CSAET and INAET are both better than NSAET; for dataset uta-s-92, the rate of convergence of NSAET is better than CSAET and INAET; and for dataset yor-f-83, the rates of convergence of CSAET and NSAET are both better than INAET. Hence, it is difficult to conclude which algorithm had shown the best convergence rate.

7. Conclusion

This paper has successfully presented and compared three artificial immune algorithms for university examination timetabling; CSAET, INAET and NSAET. The experimental results using Carter datasets (Table 2) have significantly shown the effectiveness of the three algorithms. All algorithms have successfully produced good quality examination timetables in all datasets. Both CSAET and NSAET are more effective than INAET in producing good quality timetables (low fitness values); however, INAET runs faster than CSAET and NSAET.

The plots of the fifth-trial results of all datasets have shown that the rates of convergence of the three algorithms are approximately equal for most of the datasets. The plots have significantly shown that the three algorithms are effective optimization algorithms.

All algorithms show great promise in the area of educational timetabling, particularly in its ability to consider, solve, and optimize the wide variety of different examination timetabling problems. The algorithms can handle the hard constraints and soft constraints very well. These algorithms may be accepted as new members of EAs for solving timetabling problems. Each algorithm has all the steps involved in an EA (reproduction and genetic variation, affinity evaluation, and selection).

For future work, the three artificial immune algorithms will be applied to university *course* timetabling.

8. References

[1] Anh, D.T. and Hoa, L.K., Combining Constraint Programming and Simulated Annealing on University Exam Timetabling, *Proceedings International Conference RIVF'04*, Feb 2004, pp.205-210.

[2] Burke, E.K., Newell, J.P., and Weare, R.F. A memetic algorithm for university exam timetabling. *Proc. of the First Inter. Conf. on Practice and Theory of Automated Timetabling* (ICPTAT 1995), pp:496-503.

[3] Dasgupta, D., Ji. Z., and González, F., Artificial Immune System (AIS) Research in the Last Five Years, *Proceedings of the Inter. Conf. on Evolutionary Computation* (CEC2003), Canberra, Australia, Dec 2003.

[4] de Castro, L.N., Immune, Swarm, and Evolutionary Algorithms, Part I: Basic Models, *Proceeding of the ICONIP, Workshop on Artificial Immune Systems*, Singapore, Nov 2002, Vol.3, pp.1464-1468.

[5] de Castro, L.N. and Von Zuben, F.J., The Clonal Selection Algorithm with Engineering Applications, *Proceedings of GECCO'OO (Workshop)*, 2000, pp.36-37.

[6] Di Gaspero, L. and Schaerf, A., "Tabu search techniques for examination timetabling", *Lecture Notes in Computer Science 2079*, Springer, 2001, pp.104-117.

[7] Farmer, J.D., Packard, N.H. and Perelson, A.S., "The immune system, adaptation, and machine learning", *Physica. 22D*, 1986, pp.187-204.

[8] Forrest, S., Perelson, A.S., Allen, L. and Cherukuri, R., Self-nonself discrimination in a computer, *Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy*, 1994, pp:202-212..

[9] Gambhava, N., and Sanghani, G. University Examination Timetabling using Genetic Algorithm. *2nd International Conference on Applied Artificial Intelligence*, 2003.

[10] Hart, Emma, "Immunology as a Metaphor for Computational Information Processing: Fact or Fiction?", *PhD Thesis*, University of Edinburgh, 2002.

[11] Hart, E. and Ross, P., An Immune System Approach to Scheduling in Changing Environments, *Proceedings of the GECCO '99*, pp.1559-1566.

[12] Hart, E., Ross, P. and Nelson, J., Producing Robust Schedules Via An Artificial Immune System, *Proceedings of the ICEC'98*, 1998.

[13] Jerne, N.K., "Towards a Network Theory of the Immune System", *Ann. Immunol.* (Inst. Pasteur) 125C, 1974, pp.373-389.

[14] Malim, M.R., Khader, A.T., and Mustafa, A. A Clonal Selection Algorithm for University Examination Timetabling. *Proc. of the First International Symposium on Bio-Inspired Computing (BIC'05)*, 5-7 Sept. 2005, UTM, Johor Bahru.

[15] Malim, M.R., Khader, A.T., and Mustafa, A. A Negative Selection Algorithm for University Examination Timetabling. *Proc. of the Inter. Conference on Quantitative Sciences and Its Applications (ICOQSIA 2005)*, 6-8 Dec. 2005, UUM, Penang.

[16] Timmis, Jonathan, An Introduction to Artificial Immune Systems, *Research Notes*, ES2001, Cambridge, Dec 2001, University of Kent at Canterbury, England, UK.