Curriculum Schedule Arrangement Solved by An Improved Immune Optimization Algorithm

Xiao-feng Li
Department of Computer Science and Technology,
ChengDong College of Northeast Agricultural University
Harbin, China
Mberse@126.com

An Shi , Jian-guo Luo
Information Center of National Defense Science and Industry Bureau
Beijing, China
cidexam@163.com

Abstract— **To solve the university timetabling problem (UTP) effectively, a immune algorithm-based solution for UTP was proposed. The mathematical model of UTP was expounded, a framework of immune algorithm was given, and simulation experiments were done to validate algorithm. Experimental result shows that proposed algorithm can solve the UTP effectively, and has the advantage of good application value.**

Keywords-component;university timetabling problem; immune optimization algorithm; matrix coding

I. INTRODUCTION

Curriculum schedule arrangement is much more complicated in the academic management in colleges and universities, for the reason that there're so many departments and courses which are crossed, and curriculum schedule hard to be arranged in an accurate, uniform and efficient way just by manual work due to a serious deficiency in faculties and classrooms. Earlier arrangement is manually executed by experienced academic staff. Such means of curriculum schedule arrangement is less efficient and unable to compromise the conflict teachers are plagued temporally and spatially in classrooms. As computer has become popular and artificial intelligence technology develops, an increasing concern has been received with artificial intelligence algorithm being applied in the construction of informatization in high schools by replacing manual schedule arrangement for courses, which also appeals to scholars both home and abroad their interests in research. The work [1] presents a scheme in course scheduling system based on basic genetic algorithm; in [2], it discusses an approach based on decimal coding algorithm; for the improvement of convergence and efficiency of course arrangement, a method is proposed on the basis of multi-population genetic algorithm in [3]; go further in [4], it's a solution based on particle swarm optimization (PSO); a solution for curriculum schedule arrangement based on particle swarm intelligence and optimization algorithm with immune quantum behavior can be seen in [5] in order to deal with issue in reasonable allocation of resources in course scheduling; and in [6], it proposes a hybrid genetic algorithm against the above-mentioned problem.

Genetic algorithm has a defect in premature convergence. When solving the problem, a basic particle swarm algorithm can't perform well about the discrete optimization problem, easily falling into local optimization and thus a difference appears between optimized result and the desired one. Artificial immune system (IAS) is one of the most recent research findings in the field of artificial intelligence. By using the working mechanism of biological immune system as bionic mechanism, it simulates the natural defense mechanism and immune response process of immune cells to external substances. The system has many advantages, such as noise-forbearing, unsupervised learning, self-organization and memory. It has huge potential in settling multi-objective optimization. As one of the tendencies of the study on artificial immune system, immune method simulates such mechanisms in biological immune system as self- and non-self-recognition, parallel distributed processing (PDP), adaptive adjustment, immune memory and immune tolerance, being widely used in the area of engineering optimization. In this case, this paper designs a new immune optimization algorithm to solve problems with regards to curriculum schedule arrangement in colleges and universities.

II. MATHEMATICAL MODEL FOR COURSE SCHEDULING ISSUES IN COLLEGES AND UNIVERSITIES

Scheduling issues can be considered as the one about resources allocation, that is, some quantitative resources are distributed to every demanding individual under the premise of meeting some restrictions. Its main objective is to arrange classrooms, teachers, classes and courses schedule in the non-conflict time in a week by according to the instructional plan. Many conflicts will appear in courses arrangement and resulting factors are mainly as follows:

Time: calculated by week as school hours in practical scheduling. The class period of each week can't more than seven days, with every day split into morning, afternoon and evening sessions. Each session has its own number of lessons. Say morning session P1, afternoon and evening sessions P2 and P3 respectively. Session is the smallest unit of a lesson, which is normally for two class periods.

Course: each course has its own serial number, name, corresponding department and also the plan for giving lessons, e.g. from which week the work begins, in which week it ends, how many credit hours is required each week, etc.

Classroom: each has its own serial number, name and room. In the same period only the lesson of one course is accepted and the capacity of a classroom should prevail the number of students attending the course.

Class: each has its own serial number and name. In the same period, only the lesson of one course is accepted.

Teacher: each has its own number and name. Also in the same period, each teacher can just give one course.

Suppose a school has classes C, teachers G, courses L, classrooms R and time sessions T, this model can be described as:

Class set CS= $\{1,2,...,i,...,I\}$, each member is a class. Each class has respective person $\{k_1, k_2, \cdots, k_c, \cdots, k_C\}$.

Teacher set GS= $\{g_1, g_2, \cdots, g_g \cdots..., g_G\}$, the corresponding number of courses to each one is $\{y_1, y_2, \cdots, y_g \cdots..., y_G\}$.

Course set LS= $\{l_1, l_2, \cdots, l_l, \cdots, l_L\}$. Each course corresponds to a teacher. The corresponding number of class to each course is $\{z_1, z_2, \cdots, z_z, \cdots, z_Z\}$.

Classroom set RS= $\{r_1, r_2, \cdots, r_r, \cdots, r_R\}$. The accommodation of each classroom is $\{x_1, x_2, \cdots, x_r, ..., x_R\}$ persons.

Time set: T= $\{t_1, t_2, \cdots, t_t, \cdots, t_T\}$.

Cartesian product of time and classroom pair is $M = T \bullet R = (t_1, r_1), (t_2, r_2), \cdots, (t_t, r_r), \cdots (t_T, r_R)$. In this regard, course scheduling turns to the problem about one course finding for a suitable pair of time and classroom.

In the same period, no more than one lesson is acceptable in one class, i.e, $\sum_{g=1}^{G}\sum_{l=1}^{L}\sum_{r=1}^{R} c_c g_g l_l r_r t_t \le 1$, where $c = 1, 2, \cdots, C; t = 1, 2, \cdots, T$. So now we can get the expression of in the time *t* and classroom *r*, the lesson *l* for class *c* imparted by teacher *g* as $c_c g_g l_l r_r t_t$ =1, or 0.

In the same period, one teacher can simultaneously give no more than one lesson, i.e. $\sum_{c=1}^{C}\sum_{g=1}^{G}\sum_{l=1}^{L} c_c g_g l_l r_r t_t \le 1$, where, $r = 1, 2, \cdots, R; t = 1, 2, \cdots, T$. Classroom *r* in the time *t*, where the lesson *l* imparted by teacher *g* to class *c* can be expressed as $c_c g_g l_l r_r t_t$ =1, or 0.

The accommodation *x* of allocated classroom *r* should not less than the number *k* in class *c*, $x_r \ge k_c$.

Course schedule arrangement is actually a constrained optimization problem, so any solution (or scheduling solution) meeting the constraint is a feasible solution, which will be a solution set. For the convenience of differentiating advantages and disadvantages of feasible solutions (scheduling solution), this paper designs these soft constraint measurements:

(1) In the light that teaching effect of a course is closely associated with the period of giving lessons, comparatively important courses are prioritized to be arranged to the extent in the time period when good results are fulfilled in the course arrangement;

(2)Cater for instructors' proposals for preferable time and place of instruction;

(3)The week-times arrangement for multi-credit courses should be staggered. In practice, those courses are generally arranged at least every other day in order to make sure good instructional effects;

(4)Availability of resources: an appropriate course schedule arrangement will help save plenty of resources.

III. SCHEDULING SOLUTION BASED ON IMMUNE ALOGORITHM

A. Antibody coding

In immune algorithm, problem is abstractively seen as antigen, and potential candidate solutions as antibbodies. According to what's discussed previously, scheduling problem is virtually the combinatorial optimization in space (classroom), time (the number of lessons and times) and event (a teacher gives a lesson to students in class x), for which decimal coding is fit.

Each antibody corresponds to a solution and the antibody uses matric coding: $\mathbf{R} = (\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, ..., \boldsymbol{\alpha}_n)^T$ where, column vector $\boldsymbol{\alpha}_j (1 \le j \le M)$ is course schedule arrangement for the jth classroom in a week.

Set a school has two classrooms, ten scheduling sessions every week, i.e. two sessions on every morning and no lessons in the afternoon, two classes, eight courses and eight tutors. One teacher can only give one course. Course schedule arrangement for class 1 and 2 are resepectively seen in Table I and II.

Table I. Course Schedule Arrangement for Class 1

Course No.	Times of giving lesson per week	Teacher's No.
1	2	2
2	2	4
5	3	6
8	2	8

Table II. Course Schedule Arrangement for Class 2

Course No	Times of giving lesson per week	Teacher's No.
3	2	1
4	3	5
6	2	3
7	2	7

The antibody coding corresponding to a scheduling solution is:

$$\mathbf{R} = \begin{pmatrix} (1,1,2) & (2,3,1) \\ (1,2,4) & (2,4,5) \\ (2,6,3) & (1,8,8) \\ (1,5,6) & (2,7,7) \\ (2,3,1) & (1,1,2) \\ (0,0,0) & (2,4,5) \\ (1,8,8) & (2,6,3) \\ (1,5,6) & (2,7,7) \\ (1,2,4) & (2,4,5) \\ (0,0,0) & (1,5,6) \end{pmatrix}$$

where, $R[1,1]=(1,1,2)$ means course 1 in class 1 is given by teacher 2 in classroom 1 in the time period 1 (8:00-9:00AM on Monday); $R[6,1]=(0,0,0)$ means no arrangement for classroom 1 in the session 6 (10:10-11:50AM on Wednesday). In the matrix, each column refers to the course schedule arrangement for a classroom in a week and each row for all classrooms in a time period.

B. Evaluation function for antibody affinity

Course scheduling problem needs to suffice for both hard constraints namely non-conflict in resource allocation and also soft constraints-optimal resource allocation results. Immune algorithm extracts next gengeration population based on the size of antibody affinity. The setting defined by antibody affinity function has a direct effect on the rate of convergence of immune algorithm and if it can find an optimal solution. The idea in this paper for the design of antibody affinity function uses anibody affinity function which combines multi-objective and antibody affinity function due to several soft constraints in the scheduling issue, which implies many optimized objectives.

1) Comparatively important courses are arranged possibly in sessions when teaching results turn out to good. Set $\alpha_i(i=1,2,3,4,5)$, there're five teaching units each day. Instruction in the 1st, 3rd and 5th unit will bring about good effects as $\alpha_i=1(i=1,3,5)$. $\alpha_i=0(i=2,4)$ is marked for bad effects in the 2nd and 4th unit. Use β_j (j=1,2,3,4) for the importance of a course, meaning weight. In this paper, curriculum is classified into optional, basic, professional courses and degree course, whose respective weights are 1, 2, 3 and 4. Hence, optimized objective is:

$$\max(f_1)=\sum(\alpha_i\times\beta_j) \tag{1}$$

2) Instructors' proposals are generally accepted for desired time and place of giving lesson. Base on their professional titles to set χ_i (i=1,2,3,4) indicating teaching assistant (TA), lecturer, assoicate professor and professor, whose values are correspondingly 1, 2, 3 and 4. The will for them to give lesson in a given time is δ_i. δ_i=0, 1, 2, means unwilling, common and willing. And now the optimized objective turns to:

$$\max(f_2)=\sum(\chi_i\times\delta_j) \tag{2}$$

3) Multi-credit courses (n≥4) are scheduled possibly every other day when course arrangement is made, as to guarantee better teaching results. β_i carries the same connotation to (1). Set $\varepsilon_i(i=1,2,3,4)$ as teaching result coefficient of a course which is arranged every *i* day, with corresponding value 1, 2, 3 and 4. This time the optimized objective is:

$$\max(f_3)=\sum(\beta_i\times\varepsilon_j) \tag{3}$$

4) Utilization ratio of resources: a good course schedule arrangement can contributorily save a lot of resources. During the time of a lesson being given, the bigger the ratio of *k* students of class *c* attending a course in a classroom to the capacity *r* of the classroom, the higher the ratio of resource utilization. When the ratio is 1, the classroom accommodates fairly well those students. The optimized objective turns to:

$$\max(f_4)=\sum\frac{k_c}{r_r} \tag{4}$$

To sum up, fitness function for course schedule arrangement as per the summation of each objective's weight is generated:

$$F=\sum_{i=1}^{4}\theta_i\times f_i \tag{5}$$

The value of $\theta_i(i=1,2,3,4)$ can be pre-defined by administrator, representing the importance of each objective's course arrangement. This paper considers 3, 1, 2 and 4.

C. Design of operator

The proposed method uses clone operator and concentration control operator.

Clonal mutation: for w clonal copies generated by antibody $\mathbf{R}_p$, record as $\mathbf{R}_p^{(1)},\mathbf{R}_p^{(2)},...,\mathbf{R}_p^{(w)}$. $\mathbf{R}_p^{(i)'}$ is obtained by interchange of two columns according to the possibility, which are randomly selected by antibody matrix $\mathbf{R}_p$.

Concentration control: compute the concentration of each antibody in antibody population A and clone copy population B. Select *s1* antibodies with lower concentration but a higher affinity and randomly-produced *s2* antibodies to incorporate a new generation population.

D. Framework of immune algorithm

This paper simulates the symbiosis of diversified antibodies and activation of a minority of antibodies in immune response and performs clonal mutation according to the affinity of active antibodies for a concentration control of cloned antibodies in order to keep diversities of population. The algorithm takes the following steps:

Step 1: determine parameters: population size of antibody *pop_size*, memory size of population *mem_size*;

Step 2: initialize antibody's population and memory population. Randomly produce *pop_size* antibodies to constitute initial population A_0, from which extract *mem_size* antibodies to create a memory population;

Step 3: perform a clonal mutation operator on the first *z* antibodies in the *kth* generation population to obtain copied clone population B_k;

Step 4: compute affinity of every antibody in $A_k \cup B_k$ and then place them in a descending order;

Step 5: from $A_k \cup B_k$ select *mem_size* antibodies with a higher affinity and determine if in conformity with constrained conditions. If not, modify them.

Step 6: replace part of antibodies in memory bank *Ab_Membe* with modified antibodies;

Step 7: exert a concentration control operator on population $A_k \cup B_k$ to produce a new generation of population ($k \leftarrow k+1$);

Step 8: if termination condition is sufficed, output the antibody with a highest affinity in memory bank. The operation ends; or turn to step 3.

IV. SIMULATION EXPERIMENT

A. Experimental data

Data used in the experiment are sourced from course schedule (Second semester during academic years 2011-2012) in Harbin Institute of Technology. Related elements are shown in Table III.

B. Parameter settings

If population size *POP_Size* of antibody is too small, big fluctuation of target values won't indicate each optimized objective; if too big, value of each objective converges, but it won't be long and consumes much memory. As shown in Table III, there're 125 classes for this experiment. The population size is chosen as 150 in this paper.

MAXGe stands for the maximum iteration. If MAXGe is too small, each objective won't converge and some objects seem to decrease; if MAXGe is too big, each objective will converge, but it doesn't turn out to be global optimization. The paper selects 1000.

Table III. Distribution of Each Element In The Data of Course Schedule Arrangement

Elements	Quantity
Student	6200
Teacher	387
Class	125
Course	669
Classroom	168
Curriculum plan	669

C. Results

In the test, the method in work [6] (shortly IGA) is used for comparison with the proposed algorithm (i.e. DEGA), which are both operated in roulette method.

The experiment repeats for 10 times. When the population evolves every 100 generations, record the optimal fitness value. Then take the average value of those 10 records of optimal fitness values as the experimental result on fitness value (Fig. 2). Take note of the time for the evolution at every 100 generations and use the mean value of those 10 records as the result on the time (Fig. 1).

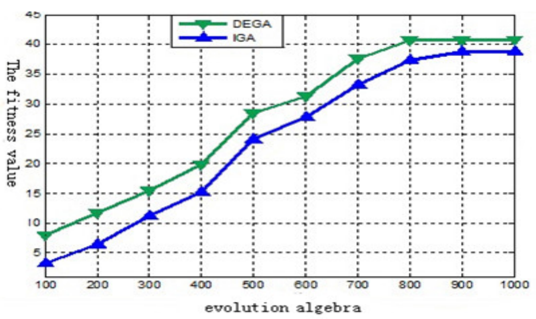

Fig. 1 Comparison of fitness values

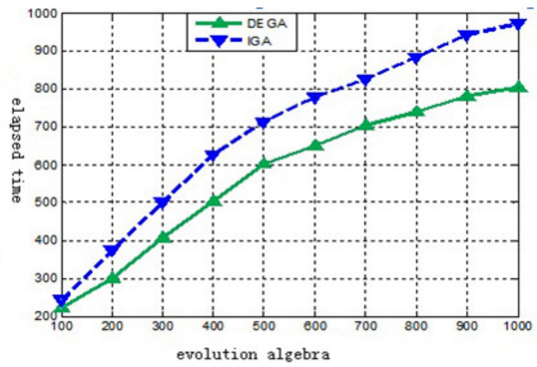

Fig. 2 Comparison of elapsed time

From the above two diagrams, it's found that the proposed approach in this paper is superior to that in [6], in terms of either the average value or time of 10 computed optimal fitness every 100 generations, meaning that the proposed solution in the paper will help achieve a better overall result in the course scheduling problem.

From Table V, we note that after the proposed method is applied, improvement is made in terms of days per week for students to take main courses, intervals for taking a same course, or even the average lessons for students to have.

As seen from table VI. , after the proposed method is used, overall utilization of classroom resources is much better, rather less amount of courses missed in the arrangement. Teachers are generally satisfied and courses conflicts are not likely to happen.

Table V. Comparison of Course Arrangement (1)

Solution	Days per week for students to take main courses	Intervals for taking a same course	Average lessons to take everyday
Solutions in literatures e	2.0—2.5	1—1.4 days	4—8
Proposed solution	2.4—2.6	1.2 — 1.5 days	4—6

Table VI. Comparison of Course Arrangement (2)

Solution	Rate of overall utilization of classroom resources	Amo unt of missing courses	Overall satisfaction of teachers	Course conflicting possibility
IGA	86%	17	84%	15%
DEGA	95%	3	95%	2%

V. CONCLUSION

After probing into problems in course arrangement, this paper establishes its mathematical model and proposes the framework of solution based on immune optimization algorithm. According to features with course arrangement, it introduces immune algorithm and designs various steps for the improved approach. Simulation experiment shows that the proposed method solves well issues on course schedule arrangement and is very useful in practical application.

REFERENCES

[1] Tan yong,wang ling. Course timetabling system based on genetic algorithm. computer application,PP.108-111,2002

[2] Tengzi,Denghuiwen. Based on the genetic algorithm in the course scheduling system design and Implementation. computer application, PP.199-203,2007

[3] Pengfuming,wuzhijian.Based on multi-population genetic algorithm in course scheduling method .Computer engineering and design,PP. 4877-4881,2010

[4] Zhanghuawei,yangkai, Particle swarm optimization algorithm based on the research of course scheduling .Journal of Zhengzhou University of Light Industry,PP.49-52,2010

[5] Wu xia,Jiangyuming.The use of immune particle swarm optimization algorithm to solve scheduling problem. Computer engineering and design,PP. 4877-4881,2010

[6] N. Pillay, W. Banzhaf. An informed genetic algorithm for the examination timetabling problem.Applied Soft Computing, PP.457-467,2011

Published by Atlantis Press, Paris, France.