On the Application of Data Mining Technique and Genetic Algorithm to an Automatic Course Scheduling System

Yao-Te Wang
Department of Computer Science and Information Management, Providence University
200 Chung-chi RD., Shalu, Taichung Hsien, Taiwan 433
ytwang@pu.edu.tw

Yu-Hsin Cheng
Department of Information Management
Ling Tung University, Taichung City, Taiwan 408
Southwestern University of Finance and Economics
hsin@mail.ltu.edu.tw

Ting-Cheng Chang
Dean of The College of Information Science
Ling Tung University
1, Lingtung Road, Taichung City, Taiwan 408
tcchang@mail.ltu.edu.tw

S.M.JEN
Department of Information Data Center
Ling Tung University
1, Lingtung Road, Taichung City, Taiwan 408
jen@mail.ltu.edu.tw

***Abstract*—In this study, a data mining technique and a genetic algorithm are applied to an automatic course scheduling system to produce course timetables that best suit students' and teachers' needs. Course scheduling in colleges and universities is a highly complicated task for satisfying multiple constraints. Conventional course scheduling is done mainly from the school's point of view, such that courses and timetables are planned according to the characteristics of individual departments and institutes, with little attention to students' interests and their needs in career development. This study develops a practical automatic course scheduling system based on students' needs, wherein the course scheduling process is divided into two stages. In the first stage, students' needs in course selection are considered and an association among courses selected by students is mined using the association mining technique; while in the second stage, the genetic algorithm is used to arrange the course timetable. More particularly, this study is based on students' willingness in course selection, analyzes the effects of course arrangement in different class periods on students' learning performance, takes into account teachers' preferred schedules, determines the cost function value of each class period, and then applies the genetic algorithm for class period exchange, so as to produce an optimal course timetable. It is shown in the experiment results that the automatic course scheduling system proposed in this study not only can efficiently replace the onerous operation of conventional manual course scheduling, but also produce course timetables that truly fulfill users' needs and increase students' and teachers' satisfaction, thereby providing a win-win-win solution for students, teachers and the school.**

***Keywords*—data mining, association mining, genetic algorithm, automatic course scheduling, course timetable**

I. INTRODUCTION

Customer Relationship Management (CRM) has become an important tool for business organizations to attract customers and gain more advantages in competition. A primary strategy to increase customers' satisfaction is to provide high-quality, customized services. From the perspective of school management, students are customers of the school. Therefore, in order to enhance the service quality of a school, it is imperative to improve the school's learning environment and curricula, wherein the efficiency of a course scheduling and selecting system has been viewed as a crucial indicator of the school's service quality. In the curricular management of colleges and universities, course scheduling is a critical service item. Now that course timetables in a previous school year cannot be copied and used in the next school year, course scheduling must be performed in each semester. Course scheduling in colleges and universities is an NP-complete operation [1]. Under the constraints of limited time, space, faculty and teaching facilities, in addition to the constraints of relevant laws and regulations and personal preferences, the school not only has to provide the required courses, but also has to offer a variety of elective courses. Therefore, the course scheduling problem can be regarded as a Constraint Satisfaction Problem (CSP) [2]. Constraints in course scheduling can be divided into two categories: the hard constraints and the soft constraints. Common hard constraints include: (1) each course is arranged to be taught at a specified time; (2) each teacher or student can be assigned to only one classroom at a time; (3) the classroom assigned must be able to accommodate students taking a corresponding course; (4) demands for special classrooms such as professional classrooms, laboratories or computer classrooms must be considered; and (5) common courses such as general knowledge courses must be arranged at particular time slots. Common soft constraints include: (1) neither teachers nor students would like unconnected time slots in a course timetable; (2) considerations must be given to the field of profession (research fields) of teachers and their preferred schedules; (3) classrooms should be close to department offices and teachers' research rooms; (4) students and teachers attending and leaving a class should be moving in the shortest distance between two classrooms; and (5) classrooms have a limited capacity.

Conventionally, constraints to be considered in course scheduling are mainly focused on those for satisfying the needs of the school and teachers, such as the utilization rate of school resources, regulation by relevant laws, and teachers' manpower and specialties; while factors such as students' preference, interests and learning effects are seldom taken into consideration. Therefore, it is very likely that a student may not be able to take the course he or she is most interested in because of schedule conflicts or because the course is not offered at all, which will be a shame in the subsequent course selection process.

In view of the above, this study takes on the customer-centric philosophy in designing an automated course scheduling system. Firstly, a data mining algorithm for association patterns is designed for mining students' habits and preferences in course selection from historical records. Then, based on the course selection patterns obtained from mining, an algorithm for course scheduling is designed, which leads to the development of an integrative, automatic course scheduling system. This system not only helps increase students' satisfaction of the course scheduling results, but also serves to provide course recommendations to students during the course selection process. Thus, this study adopts the users' point of view and applies data mining techniques to an automatic course scheduling system, which, on one hand, considers students' needs in course selection and on the other hand, takes into account teachers' preferences in their teaching schedules, so that the final course timetables fulfill the expectations of each and every user.

In this study, the automatic course scheduling process is divided into two stages. In the first stage, the association mining technique [3] is used to mine the association among students' preferences in course selection. If the number of students who collectively select certain courses reaches a predetermined threshold value, these courses are identified as having a strong association. In the second stage, automatic course scheduling is performed according to a cost function designed in this study, and Genetic Algorithm (GA) [4] is used for exchanging courses that have been arranged, so as to find the optimal course timetable.

The second section of this paper is a review of relevant studies on course scheduling and data mining techniques. The third section of this paper discloses the framework of the automatic course scheduling system proposed in this study, and discusses the method for mining associated courses and how to apply the genetic algorithm to assigning courses. In the fourth section, real data are collected from a university as the subject of experiment, and the results of course scheduling are analyzed and discussed. The fifth section is the conclusion of this study.

II. LITERATURE REVIEW

This study is targeted at automatic course scheduling systems in colleges and universities and proposes an effective solution. Literature relevant to this study can be divided into the following three categories:

A. Course Scheduling

Gotlieb was the first to perform course scheduling on a computer to simulate manual course scheduling operation [5]. The scheduling is performed by repeatedly searching for time slots that meet the requirements and then assigning courses to the appropriate time slots. Wood proposed a graph coloring technique to solve the timetabling problem [6]. Mulvey used 0-1 integer programming to analyze the course scheduling problem, for the purpose of increasing classroom utilization rates [7]. Glassey and Mizrach integrated more detailed factors into 0-1 integer programming, with an objective function of: the shortest walking distance, the least remaining seats and the lowest idle rate of teaching facilities [8]. Burke et al. proposed the use of an expert system in course scheduling, wherein the experts' feedback regarding the course scheduling results enables the expert system to evolve [9]. Yu and Sung employed a sector-based genetic algorithm to solve the course scheduling problem in universities, wherein a sector to which an event constituted by subject, class, teacher, classroom capacity, department, and type of teaching/learning belongs is matched with a location of the class to produce a course scheduling result with better cost effectiveness [10]. Parthiban et al. studied the course/professor assignment problem and thought the use of mathematical models to solve constrained matching problems has the following shortcomings: quantified data are difficult to deal with; problems are excessively abstracted; problems are hard to describe; and there are problems with combination [11]. They proposed a method for solving the teacher/course matching problem with the Analytical Hierarchy Process. Furthermore, Hertz presented a tabu search method [12] and Thompson and Dowsland disclosed a simulated annealing method [13] for solving the timetabling problem.

In the above-cited studies on course scheduling, a feasible weekly timetable is determined mainly in consideration of the school's administration needs. This study, however, develops an automatic course scheduling system based on students' interests in course selection and previous learning performance. This system can automatically generate course-scheduling results using a minimum amount of school resources, and simultaneously satisfy the needs of students, teachers and the school in course timetables.

B. Association Mining

Association rules can be used to find a hidden yet interesting relationship among a large number of data sets [3]. Support and confidence are the two parameters for determining whether or not the association rule established is significant. The Apriori algorithm is a typical method for mining association rules [14]. However, it is required in the Apriori algorithm to scan the database several times and a great number of candidate itemsets will be generated. In order to improve the efficiency of the Apriori algorithm, many methods have been proposed to reduce the amount of candidate itemsets and the number of database scan [15-18].

The FP-Growth algorithm [19] is different from the Apriori algorithm in that it is not necessary to generate candidate itemsets, and that the database only has to be scanned twice.

During the first scan, frequent itemsets with a length of 1 are identified. During the second scan, an FP-Tree is constructed according to the frequent 1-itemsets. Then, the FP-Growth algorithm is applied to generate the frequent itemsets from the paths on the FP-Tree.

C. Genetic Algorithm

The concept of genetic algorithm can be traced back to the theory of biological evolution proposed in The Origin of Species by Darwin [20], i.e., natural selection and survival of the fittest. Based on this theory, the genetic algorithm was first proposed by Holland [21], who simulated the mechanism of biological evolution on a computer. The goal of genetic algorithm is to yield a next generation with higher adaptability. Iteration is used in genetic algorithm to find the optimal solution. During the iteration process, a new generation is generated using a fitness function, and the iteration results that do not meet the fitness function are eliminated. The most important task in genetic algorithm is gene exchange in the chromosomes, whereby chromosomes in the next generation are formed.

The genetic algorithm designed in the study uses a cost function for measuring the cost of each class period assignment and exchange during the generating process of a weekly course timetable. A smaller cost function value indicates a better adaptability in evolution; a cost function value approaching zero implies a better course scheduling result.

III. Research Method

In this study, construction of the automatic course scheduling system is divided into two stages. In the first stage, the algorithm for mining association rules is used to establish a table of associated courses selected by students. In the second stage, courses offered are arranged into a weekly course timetable for each class, and the course timetable is optimized using the genetic algorithm. The principle of automatic course scheduling is to first consider the mining results of students' course selection data obtained from the first stage, and then, during the course scheduling process, arranging the courses that most students have collectively selected into different time slots (i.e., at different class periods), so that students have the opportunity to take courses having a strong association.

A. Constructing the Table of Associated Courses

The focus of this stage is to construct the table of associated courses. The FP-Growth mining technique is used in this study to find the association among courses and then save the associated course sets into the table of associated courses, wherein each set of associated courses is given an association ID.

B. Automatic Course Scheduling with Genetic Algorithm

Course scheduling is a task for satisfying multiple constraints, wherein a variety of information must be considered. Whenever a course is arranged into the course timetable, available resources are reduced as a result, making it more difficult to satisfy the constraints. In this study, it is strictly forbidden in an automatic course scheduling process to violate the hard constraints in order not to generate a course timetable that is not feasible. On the contrary, during the course assigning process, violation of the soft constraints has no impact on course scheduling, except that the course scheduling results may be less ideal. The constraints have considerable influence on the order of course assignment.

In this study, the genetic algorithm is applied to course scheduling, and a cost function is designed as a reference in the operation of the genetic algorithm when class periods are being exchanged. If a course is arranged to an ideal class period, the resulting cost function value is smaller; if not, the corresponding cost function value is greater. During course scheduling, class periods with smaller cost function values will be given priority in the assigning process. The cost function is expressed as follows:

Cost function = Cost of constraints + Teacher's preferences + Frequency at which a teacher taught in a particular class period + Student's learning performance

The four factors to be considered in this cost function are: (1) the cost to violate the soft constraints, (2) a teacher's willingness to teach in a particular class period, (3) the teacher's former habit in class period arrangement, and (4) class periods when students had better learning performance. A cost scale is set for each of the four factors, so that when a course is assigned to a class period, the cost of that class period can be obtained. In this study, the cost value for each of the four factors is divided into ten levels on a scale from 0 to 9, wherein a value of 9 is equivalent to a hard constraint, meaning that the class period should not be assigned to a particular course.

The automatic course scheduling method in this study is performed in two stages, as shown in Fig. 1. The first stage includes steps (1) to (6), wherein courses are arranged into the class periods of a weekly course timetable. More particularly, courses which have been selected by students but not yet arranged are sequentially drawn from the database. During the assigning process, strongly associated courses must be assigned to different class periods, the hard constraints must not be violated, and all the courses selected by students must be arranged into the weekly timetable. If a course cannot be arranged into the timetable, then move back and remove the previous course that has been arranged into the timetable and try to assign a class period to the course that has not been successfully assigned. If the present course still cannot be arranged into the timetable, continue to move back and remove a more previous course that has been arranged into the timetable until the present course is successfully arranged. Then, the rest of the unassigned courses are to be arranged into the timetable.

The weekly timetable is not necessarily complete when the courses are successfully arranged into the timetable. During the course assigning process, the cost function value of a class period being assigned may not be the lowest because the class period with the lowest cost function value may have been assigned to another course. Therefore, the next stage of the automatic course-scheduling algorithm aims to optimize the weekly course timetable with the genetic algorithm, so that a most appropriate timetable is generated. In steps (7) through

```
// Stage I, assign courses to course timetable
1.  pick a unassigned course C with the first priority ordered by soft
    constraints;
2.  select non allocated class periods with least cost for C that do
    not conflict with associated courses and do not violate hard
    constraints;
3.  if C cannot be assigned to course timetable then
4.      remove all periods allocated to previous assigned course;
5.  repeat steps (2) to (4) until C is assigned to the course timetable;
6.  repeat steps (1) to (5) until all courses are assigned to the course
    timetable;

// Stage II, use GA algorithm to find optimal solutions
7.  pick the course C having the highest cost function value;
8.  if C can be exchanged with other courses then
9.      exchange the periods of C with those of other courses;
10.     retain the exchanged periods wherein the summation of new
        costs is lowest and less than that before exchanged;
11. pick another course C with the next lower cost function value;
12. repeat steps (8) to (11) until all periods in the course timetable
    cannot be exchanged to reduce total cost further;
```

Figure 1. The automatic course scheduling procudeure

(12), the class periods in the course scheduling results obtained from the first stage are exchanged one after another with the genetic algorithm, so as to find the optimal solution of the weekly course timetable. In order to increase the convergence speed towards the optimal solution, courses are exchanged in such a way that a course occupying a higher cost period exchanges periods with another course, and the cost function values of the periods before and after the exchange are compared. If the cost function value of the new period is lower than that of the old period, the new period is saved. The exchange of periods is repeated until a weekly course timetable with the lowest cost function value is obtained.

IV. EXPERIMENT RESULTS

An actual case is presented below to demonstrate the mining of the associated course table and the course arranging process of the automatic course scheduling algorithm. The experiment data of this study are course selection records gathered from a second year class of the Department of International Business of some University of Science and Technology, for the first semester of the 2006 school year. The records were mined to establish the table of associated courses. Then, course scheduling was conducted on courses offered to the same class for the first semester of the 2007 school year.

To begin with, the elective courses of the Department were sorted by the number of students who had selected them as shown in Table I. The Department offered five elective courses and a total of 180 entries for course selection were recorded. The threshold of minimum support was set at 10%. In the five courses, "Risk Management" did not meet the minimum support threshold and was therefore eliminated. Then, courses selected by each student were sorted by the support of the respective courses, in descending order. Table II shows the course selection record of each student in the Department. In the first entry, for example, student numbered with 95***01 had selected two courses: "International Human Resource Management" and "Global Logistics Management". Then an FP-Tree was constructed with the data in Table II, and the

TABLE I. THE ELECTIVE COURSES SORTED BY STUDENTS' SUPPORT

	YEA	KIDE	CUNAME	SUPCOUNT	SUP	SUPRANK
1	951	4N	International Human Resource Management	47	26.11	5
2	951	4N	Global Logistics Management	43	23.89	4
3	951	4N	International Finance	38	21.11	3
4	951	4N	Development of Cross-Strait Economic and Trade Relations	35	19.44	2
5	951	4N	Risk Management	17	9.44	1

TABLE II. STUDENTS' COURSE SELECTION RECORDS (SORTED BY SUPPORT)

	YEA	KD	STNO	STNAME	FREQUENT
1	951	4N	95***01	Mr. Lu	International Human Resource Management, Global Logistics Management
2	951	4N	95***02	Ms. Hsu	Global Logistics Management, International Finance
3	951	4N	95***03	Ms. Chang	International Human Resource Management, Global Logistics Management, International Finance
4	951	4N	95***04	Ms. Kuo	International Human Resource Management, International Finance
5	951	4N	95***05	Ms. Hsieh	International Finance
6	951	4N	95***06	Ms. Chou	International Human Resource Management, International Finance
7	951	4N	95***07	Ms. Chang	International Human Resource Management, Global Logistics Management

TABLE III. CHANGES IN THE ORIGINAL CANDIDATE COURSE LIST

	cuname	clsname	teaname	ctms	wtms	etms	gtms
1	Global Logistics Management	2IB3A	Ms. Wei	3	3	0	3
2	International Trade Practice	2IB3A	Ms. Chang	2	3	0	3
3	International Business Management	2IB3A	Ms. Wang	2	3	0	3
4	Principles and Strategies of International Trade	2IB3A	Ms. Li	2	3	0	3
5	Financial Management	2IB3A	Ms. Chang	2	3	0	3
6	International Human Resource Management	2IB3A	Ms. Chen	2	3	0	3
7	International Finance						
8	National Language						
9	Career, Interpersonal Relations and Ethics						

	cuname	clsname	teaname	ctms	wtms	etms	gtms
1	International Trade Practice	2IB3A	Ms. Chang	2	3	0	3
2	International Business Management	2IB3A	Ms. Wang	2	3	0	3
3	Principles and Strategies of International Trade	2IB3A	Ms. Li	2	3	0	3
4	Financial Management	2IB3A	Ms. Chang	2	3	0	3
5	International Human Resource Management	2IB3A	Ms. Chen	2	3	0	3
6	International Finance	2IB3A	Ms. Lin	2	3	0	3
7	National Language	2IB3A	Ms. Chao	2	2	0	2
8	Career, Interpersonal Relations and Ethics	2IB3A	Ms. Yeh	2	2	0	2
9	Global Logistics Management	2IB3A	Ms. Wei	3	3	3	0

FP-Growth algorithm was applied to find the associated courses. Courses with strong association were given the same association ID and saved in the table of associated courses.

And now began the automatic course scheduling algorithm. As shown in Table III, nine courses, from "Global Logistics Management" to "Career, Interpersonal Relations and Ethics", were offered to class 2IB3A. The column heads in Table III are explained below: "cuname" refers to course name, "clsname" refers to class name, "teaname" refers to teacher's name, "ctms" refers to the number of hours a course can be continuously taught, "wtms" refers to the number of hours a course is taught per week, "etms" refers to the number of hours of a course that has been assigned to the weekly timetable, and "gtms" refers to the number of hours of a course yet to be assigned. During the course assigning process, courses were drawn from the database one at a time. The first course in Table III should be taught continuously for three hours, which is longer than the two hours required by the other courses. Therefore, according to the priority order for course assigning in section 3.2 of this paper, "Global Logistics Management" was the first priority candidate course. After "Global Logistics Management" was arranged into the timetable, the order of the candidate course list changed. "International Trade Practice" topped the list because its teacher served concurrently as an administrative head. In the arrangement of "International Trade Practice," the weekly hours of the course were three and yet the hours to be taught continuously were two, which means that only two class periods of the course would be assigned to the course timetable in a single operation and one hour would remain unassigned. The algorithm did not go on to make arrangement for the remaining hour but entered into the next loop, so that the remaining hour joined the candidate course list, which was then sorted again. Since there was only one hour left in "International Trade Practice," which was now the candidate course with the fewest hours, the course was at the bottom of the candidate course list after resorting. The course scheduling process continued to draw courses from the candidate course list for class period assignment until all the candidate courses were arranged into the course timetable.

In the previous step of candidate course assignment, course names were used as key words to search the table of associated courses for strongly associated courses. In Table IV, for example, "Global Logistics Management" has an association ID of 10026, and is shown to be strongly associated with other courses in the table. Class periods which had been occupied by courses with the same association ID should be avoided in the course assigning process. (The numbers in the column "TWEK" each have three digits, wherein the first digit stands for the day of the week from Monday (1) to Sunday (7), and the second and third digits stand for class periods in a day.) Only the remaining class periods were to be assigned according to the priority order.

When all the courses were arranged into the course timetable, the genetic algorithm was applied to optimize the timetable. Firstly, courses in the class course timetable were sorted by the cost function values of class periods occupied by the respective courses, in descending order. Then, the genetic algorithm was run in the following manner. The course with the highest cost function value in the class exchanged class periods with other courses, then the cost function values before and after the exchange were compared, and the arrangement with the lower cost function value was kept in the timetable. This procedure was repeated until no more lower cost function values occurred. In Table V, for example, numbers in the column "Original Cost Function" represent the sum of cost of all the class periods occupied by a particular course in the weekly course timetable. As shown in Table V, the course with the highest cost is "Financial Management," and courses which have been successfully arranged in the first run of the course scheduling process have original cost function values of zero.

After a gene exchange was performed in the genetic algorithm, assignment costs in the weekly course timetable were changed. In Table V, for example, when the class period of "Financial Management" was exchanged with the that of another course, the new cost function values of all the courses were recorded in the column "Cost Function After Genetic Algorithm with the First Course", wherein the first number of the number pair represents the new cost function value of "Financial Management" and the second number represents the new cost function value of the course in that row after class period exchange with "Financial Management." For example, the number pair for "International Trade Practice" is {15, 9}, meaning that, after the exchange, the cost function value of "Financial Management" was lowered from 18 to 15, and the cost function value of "International Trade Practice" was reduced from 15 to 9. This suggests that a class period exchange between these two courses generated a better course scheduling result. During the exchanging process of the genetic algorithm, only courses whose new cost function values were lower than the original cost function values exchanged class periods The procedure of course period exchange and comparison was repeated until the optimal solution was obtained.

TABLE IV. ASSOCIATED COURSES WHICH HAD BEEN ARRANGED INTO THE WEEKLY TIMETABLE

	ASSOID	CUNAME1	CUNAME2	TWEK
1	10026	Global Logistics Management	Risk Management	101
2	10026	Global Logistics Management	Risk Management	102
3	10026	Global Logistics Management	Risk Management	103
4	10026	Global Logistics Management	Development of Cross-Strait Economic and Trade Relations	203
5	10026	Global Logistics Management	Development of Cross-Strait Economic and Trade Relations	204
6	10026	Global Logistics Management	International Human Resource Management	405
7	10026	Global Logistics Management	International Human Resource Management	406
8	10026	Global Logistics Management	International Human Resource Management	407

TABLE V. ORIGINAL COST FUNCTION VALUES AND COST FUNCTION VALUES AFTER COURSE EXCHANGE

Course No.	Course Name	Original Cost Function	Cost Function After Genetic Algorithm with the First Course
4N31131	Financial Management	18	
4N31132	International Trade Practice	15	{15,9}
4N31135	International Human Resource Management	6	{18,6}
4N31136	International Finance	5	{18,5}
4N31101	International Business Management	2	{18,2}
4N31102	Principles and Strategies of International Trade	0	{18,0}
4N31708	National Language	0	{18,0}
4N31800	Career, Interpersonal Relations and Ethics	0	{18,0}
4N31108	Global Logistics	0	{18,0}

V. CONCLUSION

For a school to reinforce competitiveness, it is essential to strengthen its teaching and administrative services. The service quality of curricular management, in particular, is a major indicator for students' and teachers' satisfaction. This study has the following contributions in curricular management:

1) Chances for students to have schedule conflicts can be decreased and the time used in course selection can be reduced. Furthermore, students' willingness to make course selection can be satisfied.

2) Course scheduling is done in consideration of teachers' preferences in class periods to increase teachers' satisfaction at course timetables.

3) Fewer schedule conflicts lead to a decreased number of courses offered by the school, so that demands for teachers, classrooms and other facilities can be lowered.

4) Students are allowed to take courses according to their will while teachers are given the opportunity to teach those who are really interested. A better interaction between teachers and students helps intensify teachers' enthusiasm in teaching and upgrade students learning effects.

The shortcomings of conventional course selecting processes have resulted in certain undesired phenomena, such as: students' grades are unsatisfactory, class attendance is low, attention in class is poor, etc. The present course selecting systems only analyze the results of course selection, which may not reflect students' original willingness in course selection. By using these data alone, the school is unlikely to plan courses and timetables that can fulfill students' needs. Therefore, the objective of this study is to apply data mining techniques to automatic course scheduling, so that in a first stage, students' willingness in course selection is mined, and in a second stage, a genetic algorithm is applied to perform automatic course scheduling based on the mining results from the first stage in combination with other constraints. The course scheduling results thus obtained not only conform to the school's administrative considerations but also, more importantly, satisfy the needs of students and teachers in course timetables. Therefore, the automatic course scheduling system proposed in this study is capable of improving the interaction among students, teachers and the school, creating a good relationship among the three parties.

REFERENCES

[1] Burke, E. K., and J. P. Newall(1999), A multistage evolutionary algorithm for the timetable problem, *IEEE Transactions on Evolutionary Computation*, 3(1), 63-74.

[2] Marriott, K. and P. J. Stuckey(1998), *Programming with Constraints: An Introduction*, The MIT Press, Cambridge, Massachusetts.

[3] Agrawal, R., T. Imielinski, and A. Swami(1993), Mining Association Rules between Sets of Items in Large Databases, *ACM-SIGMOD international conference*, 207-216.

[4] Goldberg, D. E.(1989), *Genetic algorithms in search, optimization and machine learning*, MA: Addison-Wesley.

[5] Gotlieb, C. C.(1962), The Construction of Class-Teacher Timetables, in *Proceedings of IFIP Congress*, 73-77.

[6] Wood, D. C.(1969), A Technique for Colouring a Graph Application to Large Scale Timetabling Problem, *The Computer Journal*, 12, 317-319.

[7] Mulvey, J. M.(1982), A Classroom/Time Assignment Model, *European Journal of Operational Research*, 9(2), 151-162.

[8] Glassey, C. R. and M. Mizrach(1986), A Decision System for Assigning Classes to Rooms, *Interfaces*, 16(5), 92-100.

[9] Burke, E. K., D.G. Elliman and R.F. Weare(1994), A genetic algorithm for university timetabling, in *Proceedings of the AISB Workshop on Evolutionary Computing*.

[10] Yu, E. and Sung, K.-S.(2002), A Generic Algorithm for a University Weekly Courses Timetabling Problem, *International Transactions in Operational Research*, 9, 703-717.

[11] Parthiban, P., K. Ganesh, S. Narayanan, and R. Dhanalakshmi(2004), Preferences Based Decision-making Model (PDM) for Faculty Course Assignment Problem, *IEEE International Engineering Management Conference*, 1338-1341.

[12] Hertz, A.(1991), Tabu search for large scale timetabling problems, *European Journal of Operational Research*, 54, 39-47.

[13] Thompson, J. M. and K. A. Dowsland(1996), General cooling schedules for a simulated annealing based timetabling system, *Lecture Notes in Computer Science*, 1153, 345-364.

[14] Agrawal, R. and R. Srikant(1994), Fast algorithms for mining association rules, in *Proceedings of the International Conference on Very Large Data Bases*, 478-499.

[15] Brin, S., R. Motwani, J. D.Ullman, and S. Tsur(1997), Dynamic Itemset Counting and Implication Rules for Market Basket Data, *ACM-SIGMOD international conference*, 255-264.

[16] Park, J. S., M. S. Chen and P. S. Yu(1995), An effective hash-based algorithm for mining association rules, *ACM SIGMOD international conference*, 175-186.

[17] Savasere, A., E. Omiecinski and S. Navathe(1995), An Efficient Algorithm for Mining Association Rules in Large Databases, in *Proceedings of the 21st VLDB Conference*, 432-444.

[18] Toivonen, H.(1996), Sampling Large Databases for Association Rules, in *Proceedings of the 22nd VLDB Conference*, 134-145.

[19] Han, J., J. Pei, and Y. Yin(2000), Mining Frequent Patterns without Candidate Generation, *ACM SIGMOD international conference*, 1-12.

[20] Darwin, C. (1859), *The Origin of Species*.

[21] Holland, J.(1975), *Adaptation in natural and artificial system*, Ann Arbor, MI: The University of Michigan Press.