IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

Solving of Lectures Timetabling Problem and Automatic Timetable Generation using Genetic Algorithm

Nashwan Ahmed Al-Majmar[1, 2], Talal Hamid Al-Shfaq [2]

Department of Math's and Computers, Faculty of Science, IBB University, IBB, Yemen[1]

Department of Computers and Information Techniques, UST, IBB Branch, IBB, Yemen [2]

Abstract: The timetabling is a difficult problem, which is an element of the widen field of Scheduling. The scheduling problems belong to NP hard problems and are defined as an allocation problem of resources over time. The classes or lectures timetabling problem is usually solved by hand and taking several days or weeks of repair after feedback from staff and students in departments. This paper proposed an improved algorithm to solve lectures timetabling problem using genetic algorithm (GA), which has enhanced performance especially because the modification of genetic algorithm behavior. This paper also shows implementation details of timetabling software solution, which employs GA for finding an optimal solution of timetabling problem and generating lectures timetables by using real datasets and constraints from the departments at University of Science and Technology, IBB branch (hinted in the paper as college), Yemen. This software was developed by using C# to program genetic algorithm interface and using SQL Server database to store optimal timetables and college information.

Keywords: Genetic algorithm (GA), scheduling, timetabling, chromosome representation, constraints.

I. INTRODUCTION

Scheduling is defined as "allocation of resources over time to perform a collection of tasks, while taking all constraints into account, so that input demands are met in a timely and cost-effective manner"[1]. There are a lot of categories of scheduling inside educational institutions, such as teaching, exam and student scheduling.

This paper has considered lectures timetable construction problems at the college. For this timetabling problem, the events are the courses. Timetabling problem is mainly allocating resources, i.e. lecturers, levels groups of students, halls or rooms, and timeslots (periods) to courses. Every lecturer and group of students can have at most one course at the same time; this requirement can be considered as lecture and group of students constraints. Similarly, room constraints are only satisfied if each room is used for only one course at an one timeslot.

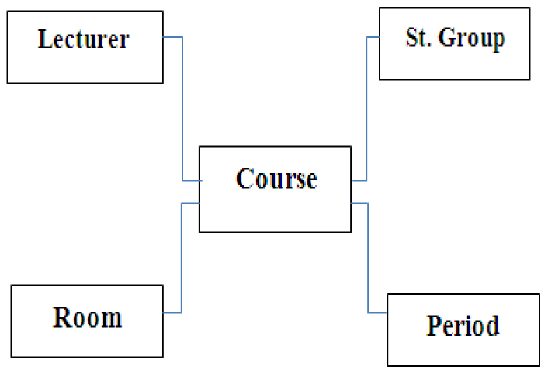


Fig -1: The relationship between timetabling concepts

The main objective of this research is to schedule courses to fully utilize available resources by assigning the course to lecturer at correct timeslot and place to appropriate event. The timetabling constraints are many and varied. In this research, genetic algorithm approach has been applied for solving lecturers timetabling problem. Figure 1 represents a central concept and the relationship between them and other concepts of timetabling problem.

Lectures timetabling problem description has been depicted in detail in section 2. General genetic algorithm, chromosome representation, genetic operators and the evaluation function are presented in section 3. Proposed behavior of genetic algorithm is presented in section 4. GA implementation with description of timetabling software solution including designed database and GA graphic user interface and implementation results are presented in section 5. Finally, section 6 presents conclusions and future works.

II. DESCRIPTION OF LECTURES TIMETABLING PROBLEM

The timetabling is a multi-dimensional assignment problem, in which students and lecturers are assigned to courses and events are assigned to classrooms and timeslots [2].
The events are usually meeting between people at a particular location [3], Educational timetabling is the sub-class of timetabling and the event in this sub-class for example can be: Lecturers in course offered at university

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

(course/class timetable). Scheduling timetable could also be represented like special class of 3D cutting problems and the timetable could be presented as a 3D structure. The dimensions of 3D timetables are: days (x-axis), timeslots (y-axis) and rooms (z-axis). The classes are shown in Figure 2 as cubes, which should be placed in a 3D timetable structure [4].

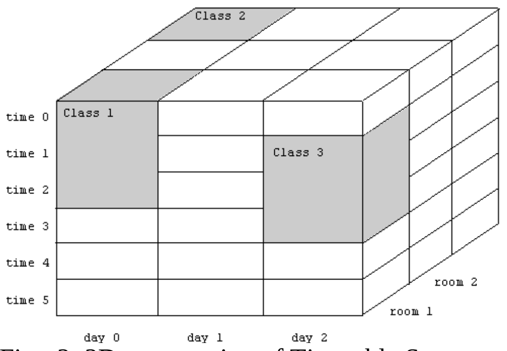


Fig - 2: 3D presentation of Timetable Structure

The scheduling is a process of placing those cubes into a timetable, in the way that no conflicting classes are placed in the same timeslot[5]. In this case lectures timetabling problem involves a scheduling of courses , lecturers, levels groups of students, rooms to a number of time-periods in a week. This paper has constructed a timetable for the two semesters courses of departments at the college. Every specialization has four levels but some new specialization have two or three levels .

Some subjects should be allocated 2 time periods like Arabic language and some subjects should be allocated 3 time periods like English language. Each day of the week is divided into 3 periods (of about 2 hours duration). There are six working days per week (the holiday is Friday).

Hence, the set P of periods consists of 18 elements. In academic year 2015-2016 first semester, department of computers and information technique, was the first department in the college to introduce automated lectures timetable. This paper presents the lectures timetable for the college, such that many students and lecturers were happiest with this timetable.

This paper has been described on a real case scenario, first, the algorithm has been trial for first semester of academic year 2015-2016. After the successful trial of the algorithm, it has been implemented in second semester of 2015-2016 academic year. The paper has considered the following case:

At the college there are 5 academic programs, 18 levels and there are about 110 courses (lecture and practical). The practice giving in average of 20 students in one section of the offered course, in each day there are three timeslots available (8^{00}h – 9^{45}h), (10^{00}h-11^{45}h) and (12^{00}h – 13^{45}h), and in each week there are 6 working days.

A. Problem Definition

There are the following participants of general lectures timetable:

L lecturers $l_1, l_2, l_3, \dots, l_L$ and

C courses $c_1, c_2, c_3, \dots, c_C$ and

R rooms $r_1, r_2, r_3, \dots, r_R$ and

D days $d_1, d_2, d_3, \dots, d_D$ and

P periods $p_1, p_2, p_3, \dots, p_P$

L is the set of all lecturers, a course having duration of about two hours and denoted by C, R is the set of rooms available at the college, D is the set of days and P is the set of all periods in a day of duration 2 hours. The same formulation of the problem has been also presented in [6, 7], in which the assignment is a 5 tuple (L, C, R, D, P).

In this research the participants of our timetable is an 8 tuple (SL, G, L, C, T, R, D, P):

SL specialty &Level, G group, L lecturer, C course, T type, R room , D day and P time period.

B. Involved Constraints

In timetabling of lectures problems, a given course has to be assigned to number of periods. Hard constraints, that are considered and satisfied in our situation are many and some of them are:

- No lecturer may have more than lecture in the same period.
- No level and group of students have more than one lecture in the same period.
- No lecturer may have lecture in periods, in which he isn't at the college.
- No two lectures in the same room and same timeslot.
- Not all allocated rooms are same of size and some of them not large enough to hold the group of students.

In order to produce a good quality timetable some of soft constraints will be considered for example:

- The room is suitable for number of students.
- The group of students not studying in the same room more than three times a week.

and other soft constraints will be considered in the future works.

As mentioned earlier the creation of a timetable is a complex problem that is a part of NP-hard problems and timetables have been constructed by hand and then modified each semester and each year. It is known that the time, required for solving this type of problems increases exponentially with the problem size. In order to generate this type of schedules we must choose the right optimization techniques, that produce optimal solutions in an acceptable time depending on size because the usage of many dimensions increases the time needed for creating the schedule. The most efficient and powerful method for minimizing timetabling conflicts and solving this type of problems are genetic algorithms [8,9].

Genetic algorithms are general search and optimization algorithms inspired by processes and normally associated with natural world [10, 11].

III.GENETIC ALGORITHM

Genetic algorithms mimic the process of natural selection and can be used as a technique for solving complex optimization problems which have large spaces. They can be used as techniques for solving complex problems and

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

for searching the large problem spaces. Unlike many heuristic schemes, which have only one optimal solution at any time, Genetic algorithms maintain many individual solutions in the form of population. Individuals (parents) are chosen from the population and are then mated to form a new individual (child). The child is further mutated to introduce diversity into the population [12]. Genetic Algorithms (GAs) were formalized in 1975 by John Holland and have been growing in popularity since, particularly for solving problems with a large irregular search space of possible solutions described in [6]. John Holland's original schema was a method of classifying objects, then selectively "breeding" those objects with each other to produce new objects to be classified stated by Buckles and Petry in [13]. The programs followed a simple pattern of the birth, mating and death of life forms from Darwinian natural selection. The fittest results are selected form the basis of next iteration or generation [14]. Basic operators such as selection, mutation and crossover are applied to get the best results. Figure 4 describes the general operators of genetic algorithm as in [3], the initial population is randomly generated.

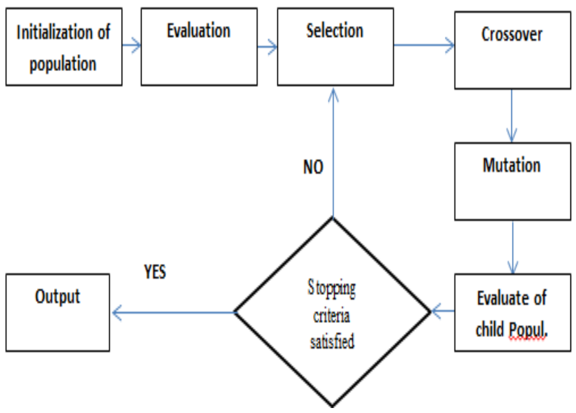


Fig - 4: Operators of a GA

IV. PROPOSED IMPROVING FOR GENETIC ALGORITHM BEHAVIOR

There is some modification that we have applied for GA. The following figure shows a suggested GA behavior:

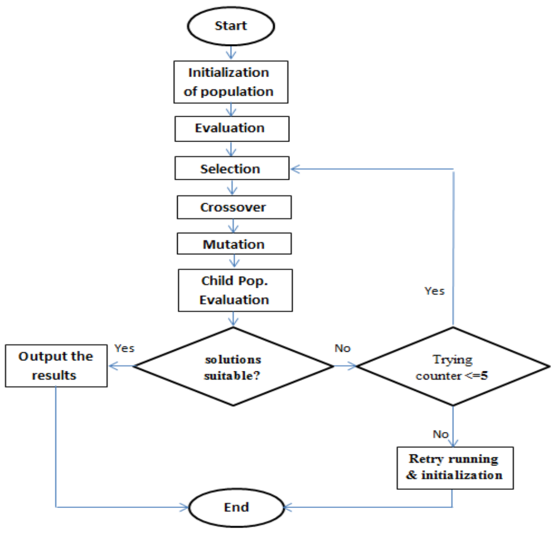


Fig - 5: GA behavior

In this section we will try to describe our suggestions for improving GA operators:

A. Chromosome Representation

Chromosome is a set of parameters which define a proposed solution to our problem, that we trying to solve using genetic algorithm. The chromosome is often represented as a simple string and the fitness of a chromosome depends upon how well that chromosome solves the problem at hand. We have chosen decimal form of chromosome and gens representation as follow:

Every gene in the chromosome contains of the following elements:

SL $_{\text{specialty \&Level}}$	G $_{\text{group}}$	L $_{\text{Lecturer}}$	C $_{\text{course}}$	T $_{\text{type}}$	R $_{\text{room}}$	D $_{\text{day}}$	P $_{\text{time}}$

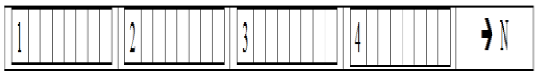
Data about these elements of chromosome we will store in individual tables, Every table contains at least two fields, one of them represents the code of the element (primary key) and the other field represent the rest of the data item (data field) for example the course table has the following form:

Course_ID	Course_Name
1	Computer Fundamentals
2	Arabic Language
3	English language
…	…

As mentioned above the lecture elements are:

SL	**G**	**L**	**C**	**T**	**R**	**D**	**P**

So the chromosome that represents one table of lectures will be as follow:



It is noticed, that the length of the chromosome, which represents a timetable is very long and complicated, let's say that we have 10 lectures for particular specialty and every lecture has 8 elements, the length of the chromosome, which represents one timetable will own length of 80 elements, therefore we need a better way to represent this data.

For solving this problem, we have suggested a method to reduce a size of the chromosome and then a timetable size dramatically. This method has the following steps:

First step: We have divided the elements of the lecture into two groups (fixed and variable):

- fixed elements:

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

SL sp&lev	G group	L lecturer	C cource	T type	R the best room for G

- Variable elements:

D day	P timeslot

So a lecture will contain elements of two group:

Fixed part	SL	G	L	C	T	R	variable part	D	P

For example the following part of data (the speciality & level "IT-1with id 1" group "2 with id 2" Lecturer "Talal with id 12" course "Computer Fundamentals with id 1" with the type "practical" the best room for this group "5") represent the fixed part of data, and (the day "6" and time "3") represent the variable part of data and so a lecture will be encoded as:

1	2	12	1	2	5	6	3

The fixed part of data is:

1	2	12	1	2	5

And the variable part is:

6	3

Second step: we have added a key for fixed part of data (Lecture_ID), that defines this part of data, so the previous fixed part will has for example the value (10) as ID:

10	1	2	12	1	2	5

Therefore the lecture will be represented as three following elements (Lecture_Id, Day, Time period) as:

Lecture_ID	D	P

so the previous lecture (with id 10 and allocated in day 6 and in period 3) will has the following form:

10	6	3

Hence, the size of a gene will be reduced from 8 to 3 elements and then the size of chromosome will be also reduced from 80 to 30 elements.

Third step: for more reducing of the chromosome size we have used hybrid method of lectures representation that uses both numbers and symbols and in which the elements of lecture are parted by (-). For example the previous lecture will be represented as:

10 - 6 – 3

and for example one timetable will be encoded as:

10-6-3	…	N

Thus the size of the chromosome has been reduced significantly.

B. Initialization

The initialization procedure creates a population of feasible solutions at random. For a timetable to be generated, the initial solution will be generated by using three sub-steps:

- Initialize the capacity and the type of rooms
- Initialize groups of students
- Initialize the courses lecturers
- Generate random values of timeslot for each lecture

So a genetic algorithm will start to work with correct initialized timetables.

C. Evaluation

Fitness in biological sense is a quality value which is a measure of the reproductive efficiency of chromosomes presented by Goldberg [15]. In genetic algorithm, fitness is used to allocate reproductive traits to the individuals in the population and thus act as some measure of goodness to be maximized [3].

As mentioned above, the constraints are divided into two types: hard and soft, and the fitness function of each chromosome is evaluated by defining a penalty value which contributes to the fitness function for each constraint violated.

We have experimented with different settings of weights for the components of the cost function and finally used the following weights:

	Element	Weight
Hard constraints	lecturer_conflict	30
	Room_conflict	20
	Group_conflict	25
	Course_conflict	15
Soft constraints	Not suitable room capacity	5
	Studying in the same room	5

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

Where:
Weight_of_lectureres_conflict= (∑ lecturer_conflict_weights + ∑not_ free_time) /number of lecturers
Weight_of_groups_conflict=∑ conflict_group_ weight /number of groups
Weight_of_rooms_conflict=∑ conflict_room_weights/number of rooms
Weight_of_courses_conflict=∑conflict_cource_weights/number of courses
The evaluation function is fined by a sum of weighted penalty values as:
Chromosome_Evaluation =(∑weights_of_hard_constraints + ∑ weights_of_soft_constraints)\100
The value of evaluation function range is between 0 and 1, and a genetic algorithm aims to find a timetables with zero value of this function.

D. Selection

selection is an operator that makes more copies of better strings in a new population. Reproduction is usually the first operator applied on a population[3]. During each successive generation, a proportion of existing population is selected to breed a new generation. Individual solutions are selected through fitness-based process, where fitter solutions are typically more likely to be selected as presented in [15, 16].

E. Crossover

A crossover operator is used to recombine two strings to get a better string. Once parents have been chosen, breeding itself can then take place. A new creature is produced by selecting, for each gene in the chromosome, an allele from either the mother or the father. The process of combining the genes can be performed in a number of ways.

The simplest method of combination is called single point cross-over stated by Davis [17]. Producing of a child chromosome can be produced more than one time in one cycle. Figure 6 shows that crossover can be happened by more than one gene of parents chromosomes.

As shown the crossover was happened between second and sixth genes of the first parent and the similar genes of the second parent to produce two new individuals

Parent 1	1-5-2	2-4-3	3-6-3	4-2-1	5-2-2	6-5-3	7-1-1

Parent 2	1-3-1	2-6-2	3-2-3	4-5-1	5-4-2	6-1-1	7-2-3

Child 1	1-5-2	2-6-2	3-6-3	4-2-1	5-2-2	6-1-1	7-1-1

Child 2	1-3-1	2-4-3	3-2-3	4-5-1	5-4-2	6-5-3	7-2-3

Fig-6: A crossover example

F. Mutation

Mutation is analogous to biological mutation and its defined as a genetic operator used to Introduce diversity in the population of genetic algorithm. Mutation alters one or more gene values in a chromosome from its initial state by randomly generated values. Therefore GA can get a better solution by using mutation. For example the chromosome before operator of mutation has the form:

Chrom.	1-3-1	2-4-3	3-2-3	4-5-1	5-4-2	6-5-3	7-2-3

And after this operator the chromosome will be as:

Chrom.	1-3-1	2-4-3	3-2-3	4-5-1	5-4-2	6-3-5	7-2-3

In each cycle, the mutation operator can be done for more one gene, in which the values of genes will be altered from their initial values.

V. GENETIC ALGORITHM IMPLEMENTATION

A. Proposed Approach

In order to deal with timetabling issues, we have proposed a system which would mechanically generate timetables for the college. Courses and lectures will be scheduled in accordance with all possible constraints and given inputs and thus a timetable will be generated. Figure 7 shows the general view of software application.
It has been collecting a wide range of data and information from more than one educational institution and it has been viewed on the steps and procedures to create teaching schedules and in particular the procedures followed at the college. the result of that analysis has been showed in the form of entity relationship diagram (ERD) as shown in figure 8.

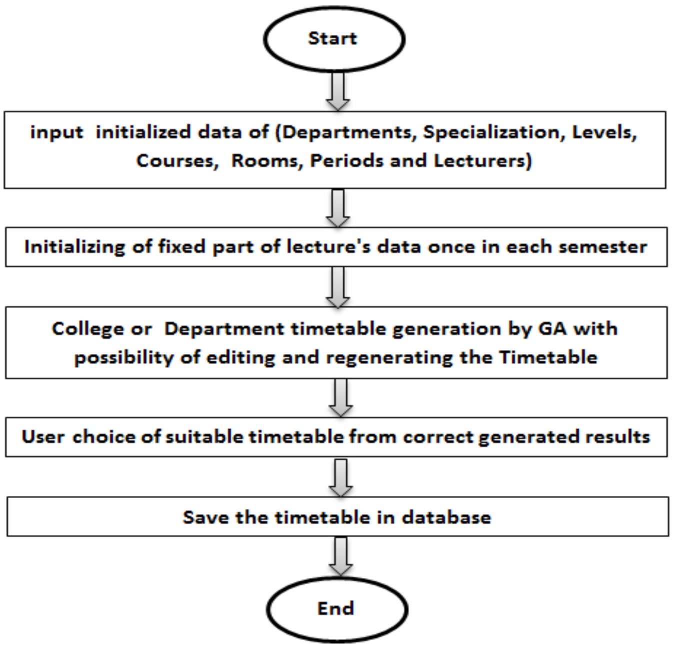

Start
input initialized data of (Departments, Specialization, Levels, Courses, Rooms, Periods and Lecturers)
Initializing of fixed part of lecture's data once in each semester
College or Department timetable generation by GA with possibility of editing and regenerating the Timetable
User choice of suitable timetable from correct generated results
Save the timetable in database
End

Fig-7: The view of software application

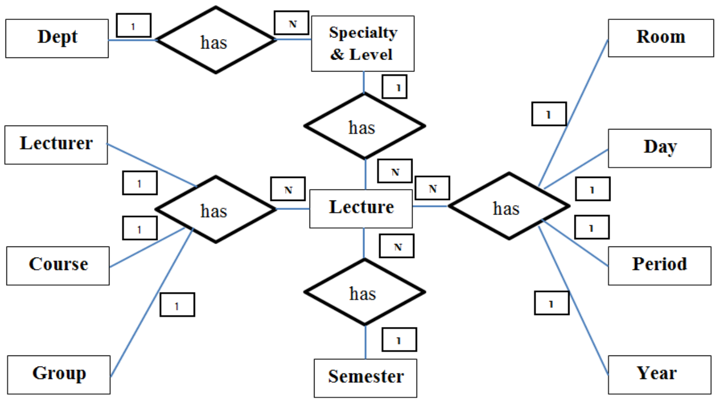


Fig-8: College ERD

This ERD involves the main entities and main relationships between entities in the considered college environment. As shown in the figure, ERD involves information about (departments, specialties, levels, courses, lecturers, rooms, groups, lectures, days, periods, years and semesters) and it was transformed to the appropriate database schema. Database will store optimal results, generated by GA for each semester and for each year. This database also will help in the conservation and archiving information of the institution in order to take advantage of them in the future. The main form of software application and the form of GA implementation are shown in the following figures:

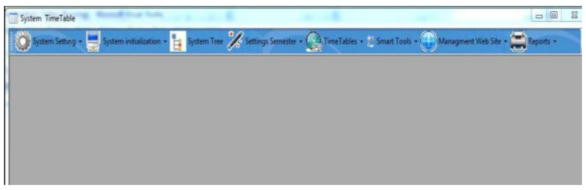
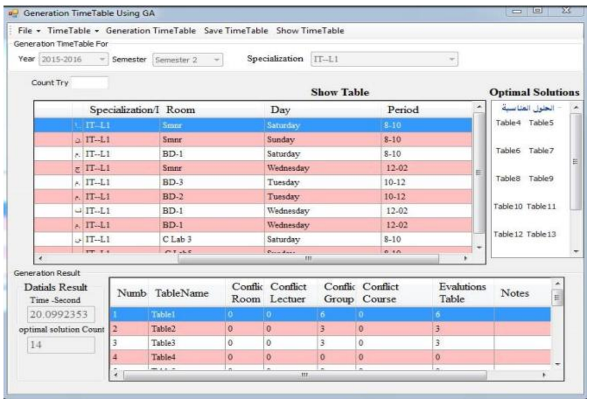


Fig-9: Main form of application



Generation TimeTable Using GA

File TimeTable Generation TimeTable Save TimeTable Show TimeTable

Generation TimeTable For

Year 2015-2016 Semester Semester 2 Specialization IT--L1

Count Try

Show Table

	Specialization/I	Room	Day	Period
١.	IT--L1	Smnr	Saturday	8-10
د	IT--L1	Smnr	Sunday	8-10
م	IT--L1	BD-1	Saturday	8-10
ج	IT--L1	Smnr	Wednesday	12-02
م	IT--L1	BD-3	Tuesday	10-12
م	IT--L1	BD-2	Tuesday	10-12
ب	IT--L1	BD-1	Wednesday	12-02
م	IT--L1	BD-1	Wednesday	12-02
ن	IT--L1	C Lab 3	Saturday	8-10

Optimal Solutions

الحلول المناسبة

Table4	Table5
Table6	Table7
Table8	Table9
Table10	Table11
Table12	Table13

Generation Result

Datials Result

Time -Second: 20.0992353

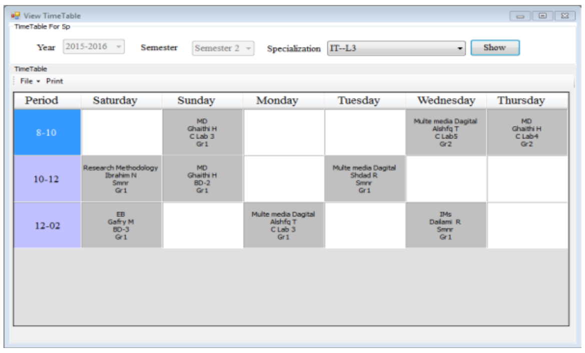
optimal solution Count: 14

Numb	TableName	Conflic Room	Conflict Lectuer	Conflic Group	Conflict Course	Evalutions Table	Notes
1	Table1	0	0	6	0	6	
2	Table2	0	0	3	0	3	
3	Table3	0	0	3	0	3	
4	Table4	0	0	0	0	0	

Fig-10: GA implementation form

B. Implementation Results

Because the software possibility of requiring many verity information and generating many types of timetables, we will try to show only some important generated timetables, for example the following figures show some of timetables:



View TimeTable

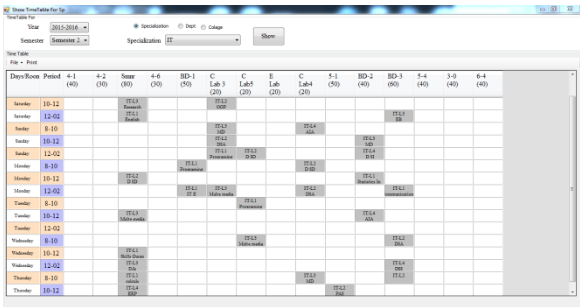
TimeTable For Sp

Year: 2015-2016 | Semester: Semester 2 | Specialization: IT--L3

Period	Saturday	Sunday	Monday	Tuesday	Wednesday	Thursday
8-10		MD Ghaithi H C Lab 3 Gr1			Multe media Dagital Alshfq T C Lab5 Gr2	MD Ghaithi H C Lab4 Gr2
10-12	Research Methodology Ibrahim N Smnr Gr1	MD Ghaithi H BD-2 Gr1		Multe media Dagital Shdad R Smnr Gr1		
12-02	EB Gafry M BD-3 Gr1		Multe media Dagital Alshfq T C Lab 3 Gr1		IMs Dailami R Smnr Gr1	

Fig-11: level timetable

The figure shows the generated timetable for ^{3rd} level – IT.



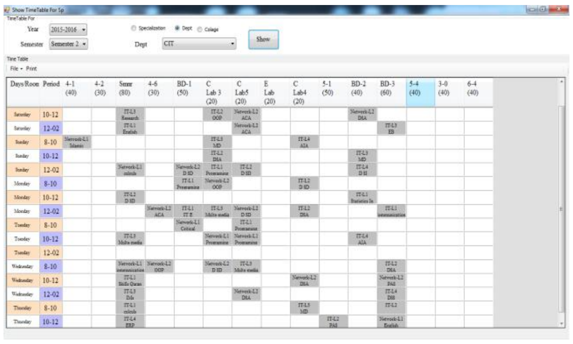
Show TimeTable For Sp

Year: 2015-2016 | Semester: Semester 2 | (● Specialization ○ Dept ○ Colage) | Specialization: IT

Days/Room	Period	4-1 (40)	4-2 (30)	Smnr (80)	4-6 (30)	BD-1 (50)	C Lab 3 (20)	C Lab5 (20)	E Lab (20)	C Lab4 (20)	5-1 (50)	BD-2 (40)	BD-3 (60)	5-4 (40)	3-0 (40)	6-4 (40)
Saturday	10-12			IT-L3 Research			IT-L2 OOP									
Saturday	12-02			IT-L1 English									IT-L3 EB			
Sunday	8-10						IT-L3 MD			IT-L4 AIA						
Sunday	10-12						IT-L2 DSA					IT-L3 MD				
Sunday	12-02						IT-L1 Programming	IT-L2 D SD				IT-L4 D SI				
Monday	8-10					IT-L1 Programming				IT-L2 D SD						
Monday	10-12			IT-L2 D SD								IT-L1 Statistics In				
Monday	12-02					IT-L1 IT E	IT-L3 Multe media			IT-L2 DSA			IT-L1 communication			
Tuesday	8-10							IT-L1 Programming								
Tuesday	10-12			IT-L3 Multe media								IT-L4 AIA				
Tuesday	12-02															
Wednesday	8-10							IT-L3 Multe media					IT-L2 DSA			
Wednesday	10-12			IT-L1 Skills Quran												
Wednesday	12-02			IT-L3 IMs									IT-L4 DSS			
Thursday	8-10			IT-L1 calculs						IT-L3 MD			IT-L2			
Thursday	10-12			IT-L4 ERP							IT-L2 PAS					

Fig-12: IT program timetable

The figure shows the generated timetable for IT specialization.



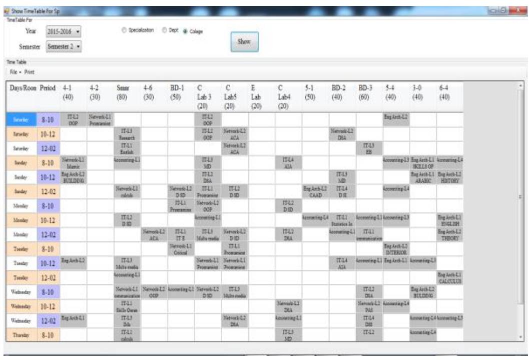
Show TimeTable For Sp

Year: 2015-2016 | Semester: Semester 2 | (○ Specialization ● Dept ○ Colage) | Dept: CIT

Days/Room	Period	4-1 (40)	4-2 (30)	Smnr (80)	4-6 (30)	BD-1 (50)	C Lab 3 (20)	C Lab5 (20)	E Lab (20)	C Lab4 (20)	5-1 (50)	BD-2 (40)	BD-3 (60)	5-4 (40)	3-0 (40)	6-4 (40)
Saturday	10-12			IT-L3 Research			IT-L2 OOP	Network-L2 ACA				Network-L2 DSA				
Saturday	12-02			IT-L1 English				Network-L2 ACA					IT-L3 EB			
Sunday	8-10	Network-L1 Islamic					IT-L3 MD			IT-L4 AIA						
Sunday	10-12						IT-L2 DSA					IT-L3 MD				
Sunday	12-02			Network-L1 calculs		Network-L2 D SD	IT-L1 Programming	IT-L2 D SD				IT-L4 D SI				
Monday	8-10					IT-L1 Programming	Network-L2 OOP			IT-L2 D SD						
Monday	10-12			IT-L2 D SD								IT-L1 Statistics In				
Monday	12-02				Network-L2 ACA	IT-L1 IT E	IT-L3 Multe media	Network-L2 D SD		IT-L2 DSA			IT-L1 communication			
Tuesday	8-10					Network-L1 Critical		IT-L1 Programming								
Tuesday	10-12			IT-L3 Multe media			Network-L1 Programming	Network-L1 Programming				IT-L4 AIA				
Tuesday	12-02															
Wednesday	8-10			Network-L1 communication	Network-L2 OOP		Network-L2 D SD	IT-L3 Multe media					IT-L2 DSA			
Wednesday	10-12			IT-L1 Skills Quran						Network-L2 DSA			Network-L2 PAS			
Wednesday	12-02			IT-L3 IMs				Network-L2 DSA					IT-L4 DSS			
Thursday	8-10			IT-L1 calculs						IT-L3 MD			IT-L2			
Thursday	10-12			IT-L4 ERP							IT-L2 PAS		Network-L1 English			

Fig-13: CIT department timetable

The figure shows the generated timetable for CIT department.



Show TimeTable For Sp

Year: 2015-2016 | Semester: Semester 2 | (○ Specialization ○ Dept ● Colage)

Days/Room	Period	4-1 (40)	4-2 (30)	Smnr (80)	4-6 (30)	BD-1 (50)	C Lab 3 (20)	C Lab5 (20)	E Lab (20)	C Lab4 (20)	5-1 (50)	BD-2 (40)	BD-3 (60)	5-4 (40)	3-0 (40)	6-4 (40)
Saturday	8-10	IT-L2 OOP	Network-L1 Programming				IT-L2 OOP							Eng Arch-L2		
Saturday	10-12			IT-L3 Research			IT-L2 OOP	Network-L2 ACA				Network-L2 DSA				
Saturday	12-02			IT-L1 English				Network-L2 ACA					IT-L3 EB			
Sunday	8-10	Network-L1 Islamic		Accounting-L1			IT-L3 MD			IT-L4 AIA				Accounting-L3	Eng Arch-L1 SKILLS OF	Accounting-L4
Sunday	10-12	Eng Arch-L2 BUILDING					IT-L2 DSA					IT-L3 MD			Eng Arch-L1 ARABIC	Eng Arch-L2 HISTORY
Sunday	12-02			Network-L1 calculs		Network-L2 D SD	IT-L1 Programming	IT-L2 D SD			Eng Arch-L2 CAAD	IT-L4 D SI		Accounting-L4		
Monday	8-10					IT-L1 Programming	Network-L2 OOP			IT-L2 D SD						
Monday	10-12			IT-L2 D SD			Accounting-L1				Accounting-L4	IT-L1 Statistics In	Accounting-L1	Accounting-L3		Eng Arch-L1 ENGLISH
Monday	12-02				Network-L2 ACA	IT-L1 IT E	IT-L3 Multe media	Network-L2 D SD		IT-L2 DSA		Accounting-L1	IT-L1 communication			Eng Arch-L2 THEORY
Tuesday	8-10					Network-L1 Critical		IT-L1 Programming						Eng Arch-L2 INTERIOR		
Tuesday	10-12	Eng Arch-L2		IT-L3 Multe media			Network-L1 Programming	Network-L1 Programming				IT-L4 AIA	Accounting-L1	Eng Arch-L1	Accounting-L3	
Tuesday	12-02			Accounting-L1												Eng Arch-L1 CALCULUS
Wednesday	8-10			Network-L1 communication	Network-L2 OOP	Accounting-L1	Network-L2 D SD	IT-L3 Multe media					IT-L2 DSA		Eng Arch-L2 BUILDING	
Wednesday	10-12			IT-L1 Skills Quran						Network-L2 DSA			Network-L2 PAS	Accounting-L4		
Wednesday	12-02	Eng Arch-L1		IT-L3 IMs				Network-L2 DSA		Accounting-L1			IT-L4 DSS		Accounting-L4	Accounting-L3
Thursday	8-10			IT-L1 calculs						IT-L3 MD			IT-L2		Accounting-L4	

Fig-14: College timetable

The figure shows the generated timetable for the college.

VI. CONCLUSION

This paper has concentrated on solving of lectures timetabling problem using genetic algorithm. The research has tried to show that genetic algorithm is a powerful method for solving timetabling problem especially with some suggested improvements. The initial timetabling problem with large number of binary variables has been significantly reduced to the acceptable size by grouping of several binary variables into one gene value in the chromosome. We have used C# to develop software application with SQL SERVER database to store and archive timetables data for future using. This model has used real datasets to test the effectiveness and functionality of the method. This software model is very useful, because it can produce varied types of timetables and inside it can be found a good combination between artificial intelligence and software engineering. The future work of this research will be trying to improve genetic approach techniques for solving real-world university teaching timetabling problems.

REFERENCES

[1] Fetaji M., Fetaji B., "Usability testing and evaluation of a mobile software solution: A case study" – IEEE conference, ITI, Dubrovnik, Croatia, 23-26 June 2008, pp. 501-506.

[2] Hossam Faris, Alaa Sheta, Ahmed Tobal. A parallel genetic algorithm for solving timetabling problem. ICGST-AIML Journal, ISSN: 1687-4846, Volume 8, Issue II 2008.

[3] Sujit Kumar Jha, "Exam Timetabling Problem Using Genetic Algorithm", International Journal of Research in Engineering and Technology, Vol. 03 Issue 05, 2014, pp 649-654.

[4] V. Mornar, "Algorithms for some classes of cutting stock problems", PhD dissertation (in Croatian), Electrotechnical faculty, University of Zagreb, Zagreb, 1990.

[5] Branimir Sigl, Marin Golub, Vedran Mornar, "Solving Timetable Scheduling Problem by Using Genetic Algorithms", Faculty of Electrical Engineering and Computing, University of Zagreb Unska 3, 10000 Zagreb, Croatia

[6] Rawat S.S., Rajamani L.,"A Timetable Prediction for Technical Education System using Genetic Algorithm," Journal of Theoretical and Applied Information Technology, Vol. 13(1), 2010, pp. 59-64.

[7] Jain A., Jain S., Chande P.K., "Formulation of Genetic Algorithm to generate good quality course timetabling," International Journal of Innovation Management and Technology, Vol. 1(3), 2010, pp. 248-251.

[8] Majlinda Fetaji, Bekim Fetaji, Mirlinda Ebibi, "Using Genetic Algorithm For Solving Time Tabling Multidimensional Issues and its Performance Testing", TEM Journal, Vol.1, Number 3, 2012, pp. 161-166.

[9] Burke E.K.. Petrovic S, "Recent research directions in automated timetabling," European Journal of Operational Research, Vol. 140(2), 2002, pp. 266-280 .

[10] Mitchell Tom, "Machine Learning", McGraw Hill, 1997.

[11] Berry .M.J.A, Linoff G., " Data Mining Techniques", Wiley Computer Publishing, 1997.

[12] D. Abramson, J.Abela,"A Parallel Genetic Algorithm for Solving the School Timetabling Problem", 15 Australian Computer Science Conference, Hobart, Feb 1992.

[13] Buckles B.P., Petry F., Genetic Algorithms, The IEEE Computer Society Press, 1992.

[14] Dipesh Mittal, Hiral Doshi, Mohammed Sunasra, Renuka Nagpure, "Automatic Timetable Generation using Genetic Algorithm", IJARCCE, Vol. 4, Issue 2, February 2015, pp. 245-248.

[15] Goldberg D.E., Genetic Algorithms in search, Optimization, and Machine Learning, Addison-Wesley Professional, 1989.

[16] Corne D., Fang H.L., Mellish C., "Solving the Modular Exam Scheduling Problem with Genetic algorithms," Proceeding of the

6th International Conference on Industrial and Engineering Applications of artificial intelligence and expert systems, 1993, pp. 370-373.
[17] Davis L., Handbook of Genetic Algorithms, New York, Van Nostrand Reinhold, 1991.

BIOGRAPHIES

Nashwan Ahmed Al-Majmar received his B.S. degree in Computer Systems Engineering and Informatics, in 2003, the M.S. degree in Computer Systems Engineering and Informatics, in 2006 from Saint-Petersburg Electro-technical University "LETI", Saint-Petersburg, Russia, and the Ph.D. degree in methods and systems of information protection and security from Saint-Petersburg State University of Information Technologies, Mechanics and Optics, Saint-Petersburg, Russia, in 2010. He is an assistant professor with Department of Mathematics and computer science, IBB University, IBB, Yemen and he is also working with CIT department at University of Science and Technology "UST", IBB branch, Yemen. His research interests include "information protection and security" ,"software development" and "AI applications".

Talal Hamid Al-Shfaq received his B.S. degree in IT, in 2015 from UST, Yemen. He works now at CIT department, University of Science and Technology, IBB branch, Yemen. His research interests include software development and AI applications.