Using Nested Tables and Mutation in Genetic Algorithms (NTMGA) to Solve Timetabling Problem in Object-Relational Model

Ahmed Sultan Al-Hegami
Associate Professor of
Intelligent Information Systems,
University of Sana'a, Yemen.

Wadee Al-Qubati
Faculty of Computing,
University of Science and
Technology, Sana'a, Yemen.

Ammar Zahary
Assistant Professor,
University of Taiz, Yemen.

ABSTRACT

In this paper, the nested tables feature in relational- object database is applied along with the feature of Mutation in Genetic Algorithms (GA) to solve the problem of timetabling, especially coursework timetables. The advantage of using nested tables is utilized to cancel a lot of conflicts and improve the operations of the Mutation in GA for the ideal solutions of multi-fewest number of generations and a short time has been concluded infer. Our approach conducted by this paper which is called NTMGA has been implemented and applied to timetable problem of Computer Science Department at the University of Science and Technology (UST). Results proved a satisfactory performance of NTMGA approach in terms of the absence of conflicts and a fast release of schedules has been obtained.

General Terms

Genetic Algorithm (GA), NTMGA, Timetabling, Chromosome, Fitness, Mutation , object relational model, nested tables.

Keywords
Genetic Algorithm (GA), NTMGA, Timetabling, Chromosome, Fitness, Mutation , object relational model, nested tables.

1. INTRODUCTION

For as long as the scheduling of study is daunting for those who design it by hand as it is a process of time consuming and must take into account the conditions that must be met in the final tables, such as elimination of the conflict in time and space and other limitations that can arise when designing these tables. For these reasons, a lot of studies and research in the Arab and foreign universities to solve this problem have been conducted in order to create an automated system that does the schedules task. Such studies have adopted various algorithms, to get the schedule free of inconsistencies. Examples of such attempts use the following algorithms to solve the schedule problems.

- Greedy Algorithm
- Genetic Algorithm
- Graph Coloring Algorithm
- Mimetic Algorithm

To solve the conflicts that occur in the timetable, we propose an approach that can be viewed in the Object-Relation database model and a genetic algorithm to address many issues including effective storage structure, indexing, and good processing for information.

Object-relation model has been taken from the quality provided by Object-Relation model and from the flexibility of relation model which provides practical solution enabling the usage of Abstract Data Types (ADT) which known as correlated columns which do not have foreign key which can be integrated at a record of a timetable.

From this type (ADT), we can create what known as "nested table" which can be structured that support the work of Genetic Algorithms when applied to solve the studying timetable problem by improving the operations of mutation and simplifying the equation of fitness, reducing the conflicts in the studying timetable. Subsequently, the performance will be increased by deleting a lot of processing required during the verification of records.

The current study comes to propose a model that depends on a view of executing genetic algorithms depending on nested tables to solve the studying timetable problems.

2. RELATED WORK

As per our reading in the literatures, there are no previous studies taking the same study. For this section, we review some studies that use genetic algorithm to solve scheduling problems.

In [1], the authors' approach is based primarily on the genetic algorithm in a sequence clusters to solve the problem of examination timetables at schools (Examination Timetabling), according to available data for these exams (inputs), and production of the optimal solution of the final table (output). The followed methodology is focusing on the importance of analyzing the items of data, information and knowledge of the mysterious based on methodological steps depending on the terms of the data derived from students and tests, has been the experience of the algorithm on three key variables from a database (decision or matter, the student, test), the comparison of the results of the experiments with the research traditional or manual (theory), where the results showed that the ability to rearrange the locations of the clusters (Clusters Rearrange) arises from the manner in which the high result and display the best methodology in the order to reach the optimal arrangement of the clusters by way of genetic algorithm. Researcher was able to find the best solution to this problem and by the speed of the account and find a good solution close to the realistic solution. Advantages of [1] can be summarized

as the ability to reach the optimal solution or the solution is near it. Also, it is a good solution to the problem of scheduling school exams. On the other hand, there are some shortcomings in [1] as the solution of the paper may not be considered a good solution for big problems like the problem of school schedules, school schedules because the problem is greater than the problem of examination timetables and thus may suffer because of the slow way.

In [2], authors proposed an approach for the establishment of a timetable interactively (Interactive Timetable) for use in schools and universities. The interactive timetable enables the user to impact and adjustment in the table at the time of its composition (Real-time Interaction). Authors believe that the ability to see the process of configuring of the table in the Real-time and the possibility of amendment it would be better suited to the process of configuring and modifying the table automatically (full automatic timetabling), which limit the amendment process to the table in case of dissatisfaction from the cell of the table. He establishes the interactive algorithm is that creates this schedule with the possibility of the province and edition by the user, whether at the time of creating the table (real-time) or after you configure this table, taking into consideration not to reconfigure the table (re-scheduling) after a change to the task of tasks, whether deleting or adding or modifying the task, but to continue and follow-up later in the make-table (sub-result), may lead to non-completion, but without the conflicts and this is the good thing about it.

Advantages of [2] can be summarized as high speed access to the best solution, the use of Java language, which is characterized as a Multi-platform, and do not use a database which is an advantage because it speeds up access to the results. On the other hand, there are some shortcomings in [2] as the timetables edited by a user in the Real-time may lead to non-completion and therefore do not reach the optimal solution.

In [3], authors conducted a solution to solve the problem of course timetables of the Department of Computer Science at the University of Science and Technology (UST) using genetic algorithms. Authors adopted a set of criteria or constraints to find a solution using this algorithm with the closest schedules to the ideal solution. However, the solution was limited to only two divisions of each intake. Authors represented the algorithm using the chromosome number 51 for each trait or gene division of the adoption of the following equation:

*Number of available times * number of representation features of one lecture = 3 * 17 = 51*

On that basis, the length of Chromosomes increases according to the number of branches, and the authors used other functions and ready libraries to represent the Chromosome such as Geneticserver.dll In this research, researchers tried to use multiple algorithms and design of optimal Chromosome to include all levels of all departments of the college but not limited to the Division or the level or the Department itself.
Advantages of [3] can be summarized as the use of genetic algorithm, and issuing tables and solving the problem of conflicts. Shortcomings [3] are limiting the application program on the two divisions and one level only, the lack of flexibility in times and hours of

lectures were identified only two hours, and excluding lectures laboratory of the solution.

In [4], authors dealt with a solution to the problem of course timetables of Department of Computer Science at the University of Science and Technology (UST), using genetic algorithms. Authors adopted a set of criteria or constraints to find the closest schedules to the ideal solution. However, the solution is limited to only one section of four levels, as any lecture will be only of two hours and the exclusion of lectures laboratories. Authors represented the algorithm using two-dimensional matrix where each row in the matrix represents the table of one room and all the rows in the matrix represent the entire chromosome. The value of the fitness function was calculated by the adoption of the following equation:

*Optimal value of the appropriate = total number of lectures + Σ (length of the chromosome-1) + ((number of rooms -1) * Number of possible lectures per day * Number of school days).*

This matrix has been utilized to make the Mutation and Selection in order to improve the chromosomes, increasing the appropriate values of the function from generation to generation until reaching the desired solution.

Advantages of [4] can be summarized as the use of genetic algorithm, and issuing tables and solve the problem of conflicts. Shortcomings of [4] are limiting the application of the software on only one section of the four levels only, and the lack of flexibility in times and hours of lectures has been identified only with two hours.

3. THE NTMGA APPROACH

At the time of designing the proposed approach, we have relied on four hypotheses is that the times of lectures ranging from eight in the morning until five in the afternoon, and the lecture could be two hours or three hours, and the halls and laboratories, involving all the specialties section, and that the students in the Study level may be divided into more than a Division according to capacity of the halls for the students.

3.1 The **Mechanism of NTMGA** Approach

The idea of the proposed approach, NTMGA, is based on the database of the University, UST. A collection views have been designed based on some of the summarized information of the lectures and coding for a gathering in the last view. Each column represents a Chromosome and a Mutation is applied to each Chromosome to reach the multiple ideal solutions. Then, each tabulated solution in the nested table composes six columns represent the days of the study week, an additional column to determine the halls, number of ranks by the number of classrooms, and laboratories in the college. The intersection of any row with any column will represent a group object and will be a lecture hall specific for a particular day. Figure 1 shows a view of our approach, MNTGA, that represents the Chromosome of the ideal solution.

In Figure 1, object one would give all the necessary data for a single lecture and includes the following information:
1. **T** represents the time of the lecture

2. **C** and is scheduled
3. **M** and a lecturer who will deliver the lecture
4. **P** and a specialization that belongs to him to be
5. **L** represents the level at which a student studying
6. **G** represents a group of students (the Division), where students may be divided in the same level to more than one group.
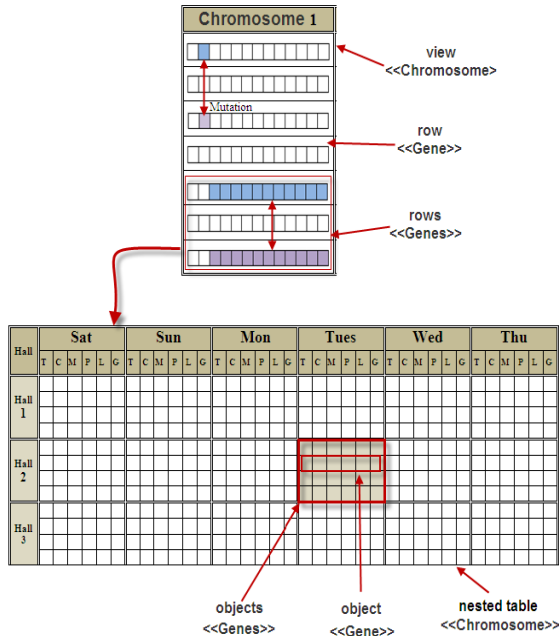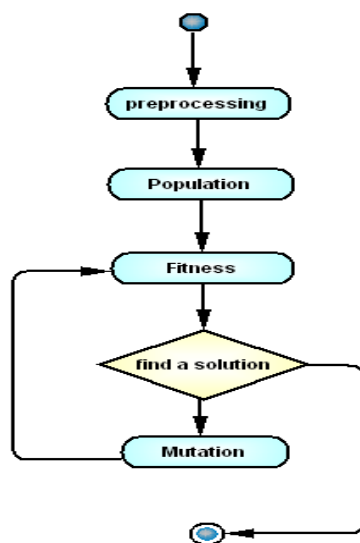

Fig 1: View of the NTMGA Approach

Framework used to implement the previous proposal shown in Figure (2) which shows the general steps that were followed in the implementation of the genetic algorithm.

3.2 Stages of NTMGA Approach

As shown in Figure 2, it is clear that the algorithm of the application consists of the following stages.

Fig 2: Flowchart of NTMGA Approach

3.2.1 Preprocessing Stage

At this stage, the preparation and processing of data on school schedules are made so they could be dealt with as Chromosomes, this has been done through the following steps:

1. Encoding potential times for lectures and knowledge of the conflicts that may occur as shown by Table 1.
2. Knowledge and encoding the number of branches in each level by knowing the number of students in each level and the capacity of the halls.
3. knowledge and encoding the number of laboratories required for each course

Table 1: Encoding all possibilities for the times of Lectures

Conf view	Time	Code
02	8-10	01
01,03,04	8-11	02
02,04,05,06	10-12	03
02,03,05,06,07,08	10-1	04
03,04,06,07,08	11-1	05
03,04,05,07,08,09,10	11-2	06
04,05,06,08,09,10	12-2	07
04,05,06,07,09,10,11,12	12-3	08
06,07,08,10,11,12	1-3	09
06,07,08,09,11,12,13	1-4	10
08,09,10,12,13	2-4	11
08,09,10,11,13	2-5	12
10,11,12	3-5	13

Previous data have been arranged in view proc name consists of one column is a symbol of the lecture consists of 10 characters encodes gives the following information:

a) **C** represents the symbol of course and consists of three characters
b) **M** is a symbol of the teacher and is composed of two characters
c) **P** represents a symbol of specialization, which belongs to him and to be composed of one character (one Character)
d) **L** represents the code-level students and consists of one character
e) **G** and is a symbol of the division level and consists of one character
f) **T** represents the type code of the decided teaching hours (lecture or lab) and consists of one character
g) **H** represents the code number of lecture hours and consists of one character, see Figure 3.

Fig 3: Encoding courses

Taking into account that the process of encoding the ranks of this column is going according to sequential order, where there is encoding for the lectures of two hours, first as follows: the course of the first level , the course of the second level .... etc, for all levels recursively in a particular specialty, and in the same way the rest of the disciplines.

Fig 4: Encoding all lectures

The more finished book a room in a particular day (the room reserved for four lectures a day from 8 am to 5), then go to the second hall, and so then it is filled with lectures three hours with the same serial. This encoding process will ensure that each lecture will have a lab or hall, without specifying time to it.

By the total number of hours (hours per day * 6 = 9 * 6 = 44) and the total number of hours of lectures for all courses that have been encoded in the proc view ,we will determine the number of rows as it is assumed that the total number of hours of lectures for all courses is equal to hours of available time less is filled to the lack of proc view rows with values (0000000000000), which represents the time available for rooms that are not completely booked, this is shown by Figure 4.

3.2.2 Preparing the Population

At this stage, the number of chromosomes is not determined and the creation of each chromosome that has been as follows: Added encoding time and day lecture and reassign code lecture randomly taking into account that you are scheduling for the first day at random and then for a second and then third and so on until the sixth day so that the remaining free time in recent days and is a summary of these procedures in Chrom view as in Figure 5.

Fig 5: Preparing the Population

Total of the resulting rows from a column in Chrom view, representing Chromosomes one and the same previous steps is the establishment of more than one column (Chromosome)
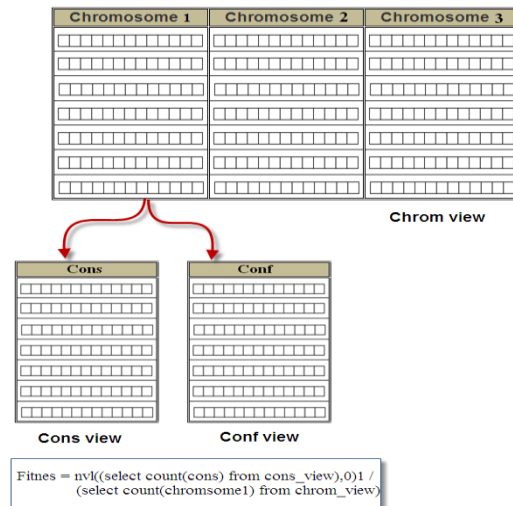
represents all Chromosomes initial solution to the problem of tables.

3.2.3 Fitness Function

At this stage, the solution has been tested through the initial function of the Fitness to make sure the quality of the solution is evaluated with the following conflicts:

a) contradiction in the teacher (the teacher taught more than a lecture at the same time)

b) a conflict of people (more than a lecture taught at the same time one of Division)

They are verified as part of a function test Chrom view to sort the rows with conflicts in the cons view and the rows that do not bear conflicts in conf view and so that the function of the Fitness equal to the number of records in the cons view dividing the number of records in the Chrom would be worth a function of the Fitness between zero and one The closer of 1 are close to the optimal solution (see the Figure 6.

$$Fitnes = nvl((select\ count(cons)\ from\ cons\_view),0)1\ / \\ (select\ count(chromsome1)\ from\ chrom\_view)$$

Fig 6: Fitness Function

3.2.4 performing the Mutation

This stage is performed using the following two methods:

1. Change the chromosome to resolve conflicts of the branches ,where we are switching rows with these values of free classes (0000000000000), which represents the times of the rooms that are not reserved
2. Change the chromosome of the teacher to resolve conflicts through the switch of the rows.

The process of testing the function of the Fitness to see the quality of the solution in the case that the value of the function did not reach the one true (and there are conflicts) is a process of Mutation again, and so this process will continue until you reach an ideal solution as shown by Figure 7.

When you access the ideal solution, it is displayed depending on the nested table composed of 6 columns represent days and the last column represents the halls and laboratories, where lectures are combined for each room in a row as shown by Figure 8.
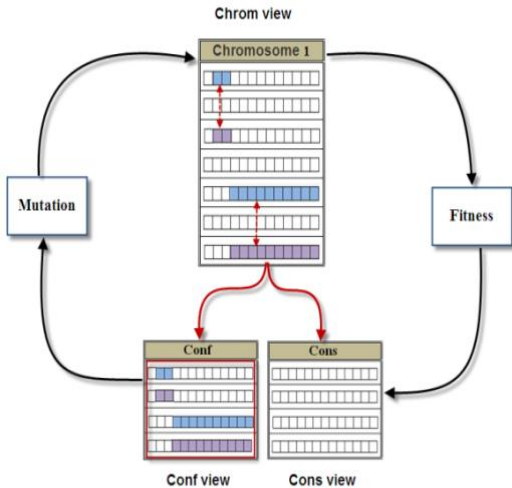
Fig 7: Performing the Mutation

4. EXPERIMENTAL STUDY

NTMGA approach has been implemented and applied to the timetable problem of Department of Computer Science and Information Technology in UST. The department holds four disciplines represented in (information technology, software engineering, computer networks, Programming Diploma). The genetic algorithms method has been applied to ten Chromosomes and the Mutation has been made on each one of them. The results were analyzed based on the version of the ten different copies of studying schedules in quick time and the accuracy of a full (100%) and without any conflicts. The following example shows the version of the timetables for the four disciplines:

1) Initially, preprocessing to prepare lectures and encodings. Figure 9 shows the encoding for all the lectures of the previous disciplines.
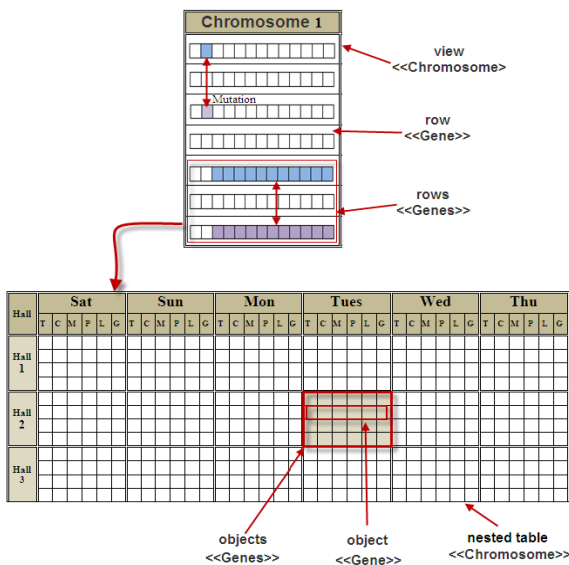
Fig 8: Using NTMGA to represent the Chromosome of the optimal solution

Fig 9: Encoding lectures

After the completion of encoding lectures, preparation has been made to the Population Figure (10) shows the encoding time and day of the lecture and appended it to the lecture code for ten Chromosomes.

Fig 10: Preparing the Population

2) Mutation process was carried out for all tables and the tables which completely free of conflicts have been obtained.

Fig 11: Nested Table Represents the Chromosome of the Optimal Solution

29

5. CONCLUSION

This paper presented a new approach called NTMGA which is developed to solve the problem of timetabling with the object-relational database taking into account the advantage of Nested Tables and Mutation in genetic algorithms. NTMGA has been implemented and applied to create the course timetables of Computer Science Department in UST to find out the ideal solution. The performance of applying NTGA was remarkable as it reduces the reduction of conflicts to zero, and the time of finding out the ideal solution has been minimized. As a result, NTMGA can improve the randomness of initial solutions from one generation to another to get various ideal solutions in a fast and easy way. As a future work, the development of NTMGA in this paper can be improved and dealt with the different kinds of timetabling and achieve flexibility in dealing with restrictions on scheduling such as determination of setting times by teachers, attendance times, lectures hours, room allocation, and any other restrictions. This might be achieved by appropriate adjustments in the chromosome encoding and improve the process of Mutation and Selection of appropriate representation in the nested table.

6. REFERENCES

[1] T. Moller and R. Partak, "Interactive Timetable", Department of Theoretical Computer Science, Charles University, 2001 .

[2] M. Mitchell , "An introduction to Genetic Algorithms", The MIT Press (Massachusetts Institute of Technology), 1998.

[3] M. Khedr, A-K. Abbadi, A. Al-Omari, M. Daud, kyphotic, "Using genetic algorithm to solve the problem of scheduling the school", Journal of Science and Technology (JST), Issue No. 9, 2005.

[4] A.Al-Tehami, A. Al-Salehi, and A. Zahary, "Application of genetic algorithms in the problem solving of course timetables, Case Study: timetables of Faculty of Science and Engineering, University of Science and Technology, Yemen", Proceedings of The International Arab Conference on Information Technology (ACIT), ACIT 2009, 14-16 December 2009, Sana'a, Yemen.

[5] P. Brown, "Object-Relational Database Development", Informix Press, 2001.

[6] R. Elmasri and S. B. Navathe "Fundamentals of database systems", 2nd edition, Redwood City, California, The Benjamin/Cummings Publishing Company, ISBN: 0-8053-1753-8, 1994.

[7] J. Palinski, "Oracle SQL & PL/SQL Handbook", (2002), ISBN: 978-0-201-75294-6.

[8] R. G. G. Cattell, "Object Database Standard: ODMG-93", Morgan Kaufmann Publishers, San Francisco, California, ISBN: 1-558-60302-6, 1994.

[9] L. Bambrick, "Lectures in Timetabling Using Genetic Algorithms", Department of Electrical and Computer Engineering, The University of Queensland, 1997.

[10] E .Burke, D. Elliman and R Weare, "A Genetic Algorithm Based University timetabling System", Department of Computer Science, University of Nottingham, 1995.

[11] L. Haan ,"Mastering Oracle SQL and SQL*Plus ",ISBN : 1-59059-448-7, 2005.