Twinkle Dahiya al. International Journal of Recent Research Aspects ISSN: 2349-7688, Vol. 2, Issue 2, June 2015, pp. 165-168

Exam Timetabling Problem Using G.A.

Twinkle Dahiya[1], Er. Kapil[2], Er.Aakash Goyal[3]
[1]M. tech. Student, SITM, Sonepat
[2]H.O.D., CSE deptt., SITM, Sonepat
[3]A.P. CSE deptt., JIET, Jind, India

***Abstract*—Exam scheduling problem is essentially concerned with scheduling a number of exams into a limited number of timeslots or periods. Exam scheduling is a multi-constraint problem. There should be no conflict in the arrangement and it should meet a number of constraints. These constraints vary from Institution to Institution. Exam scheduling problem is NP complete problem and has very high computational complexity. Exam time scheduling is a multi-constrained problem. The main aim is to produce the timetable and to optimize it under given set of constraints. Some of the constraints are used for the timetable generation and some other constraints are used to optimize the generated timetable.**

***Keywords*— Genetic algorithms, mutation, scheduling, optimization, evaluation, generation.**

I. Introduction

The general scheduling problem is described as a process of selecting among alternative plans and assigning resources and time to the set of activities in the plan. Exam Timetabling is a special case of scheduling which concerns every teaching Institution such as schools and universities. Organizations like Universities, Polytechnics and most educational Institutions in general use timetables to schedule classes and lectures, assigning time and places to future events in a way that makes use of the available resources in the most optimal way. Exam Scheduling is a kind of problem in which events (i.e. exams, classes) have to be arranged into a number of time slots, subject to various constraints. Every semester a new timetable is to be produced by taking into account staff available, number of students and number of courses and these require a large amount of work. A poorly designed timetable is inconvenient and also takes a lot of time for generation. There are many important tasks, for which it is very difficult to find a solution, but once we have it, it is easy to check the solution. This fact led to NP-complete problems. NP stands for nondeterministic polynomial and it means that it is possible to "guess" the solution by some nondeterministic algorithm and then check it, both in polynomial time. If we had a machine that can guess, we would be able to find a solution in some reasonable time. Exam scheduling problem belongs to NP complete problem. Exam scheduling problem is typical combinatorial optimization problem. Exam Scheduling has a very high computational complexity and it became more complex with the increase in number of courses, number of exams and number of students. This problem has been studied by numerous researchers due to its NP complete nature [1]. The final purpose of the timetable is to guarantee that all the students can take any exam that they are required to, but also to maintain reasonable use of resources.

Need for Automated Exam Scheduling Different Institutes have different requirements and different constraints. There is large number of variations in the concept of exam timetabling with different Institutions. The difficulty is due to the great complexity of the construction of timetables for exams, due the scheduling size of the examinations and the high number of constraints. To overcome this problem, educational Institutions need an automated system to organize their timetables. The method of using the traditional and manual exam arrangement was time consuming and laborious. Therefore it becomes essential to arrange the exam with the help of modern computerized optimization techniques.

II. STEPS OF G.A.

Genetic algorithm is fairly simple. For each generation it performs some basic operations. A simple genetic algorithm describes the following cycle:

1) Generation of random (n) chromosomes that form the initial population.
2) Assessment of each individual of the population.
3) Verification of the termination criteria.
4) If verify termination criterion - cycle ending.
5) Selection of n/2 pairs of chromosomes for crossover.
6) Reproduction of chromosomes with recombination and mutation.
7) New population of chromosomes called new generation.
8) Go back to step 2.

The cycle is repeated from one generation to other and after so many generations, the probability of getting optimized solution increases. So by varying the population size and number of generations, GA is able to find the best possible solution and can be used on variety of applications.

III. PROPOSED METHODOLOGY FOR OPTIMIZATION USING GA

The step by step implementation of GA to generate best timetable is shown in the given block diagram. First step is to generate initial clash free timetables and then applied genetic operators on the generated timetables. Fitness value is evaluated in each generation and after many generations best fitness value is selected in order to get the best timetable.

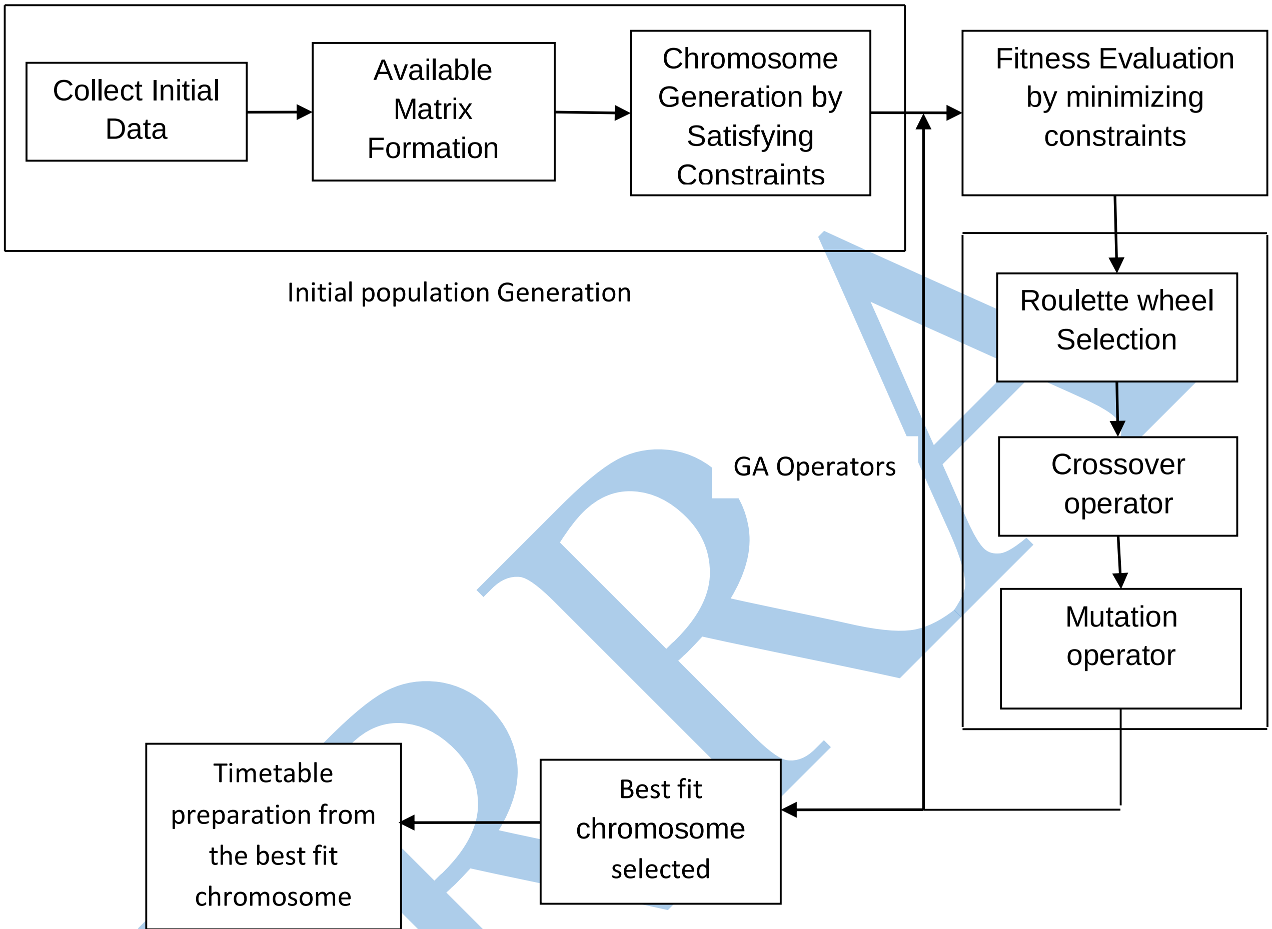

Figure 1: Block Diagram of Exam scheduling optimization using GA

IV. DETAILS OF PROPOSED WORK

A. Initial Population Generation

A population is a collection of timetables. Initial population consists of chromosomes. Each chromosome represents a timetable generated randomly. Before applying genetic algorithm to optimize the timetable generation, clash free timetables are generated which represents the initial population. From these clash free timetables, an optimized timetable is to be produced which satisfy constraints

Chromosome Representation: The encoding of the chromosomes is the first step to solve the problem and a lot of effort must be spent in selecting the suitable and the best encoding techniques because it will provide well-build starting point for the GA and will have great impact on the efficiency and the performance of the GA. To define the suitable solution, we have to search for every possible solution and then evaluate it using the fitness function.

- Each chromosome represents a clash free timetable by taking into account all the groups and courses.
- Length of the chromosome is equal to total number of courses.
- Each timetable is generated randomly and consists of strings of random integer values.
- Each gene is assigned with a random number and represents the time slot allocated to the particular course.

$$[C1 = \underbrace{1\ \ 2\ \ 9\ \ 15\ \ 30\ \ldots\ldots\ldots 27\ \ 7}]$$
$$d_1\ \ d_2\ \ d_3\ \ d_4\ \ d_5\ \ldots\ldots\ldots d_{n-1}\ \ d_n$$

Figure 13: Chromosome Representation

Here C1 denotes the first chromosome whose length is 65 and each value represents a time slot on which particular exam is allotted. In which each dn is replaced by subject (examination), whose serial number corresponds to the index of d. Timetables are encoded in a way that there are no clashes between exams of same exam group. Each exam group subjects are given different randomly generated values. If at the same time slot, two exams of same exam group are scheduled, than the value of that time slot is generated again randomly and is checked again for not further clashes with other exams. Available matrix concept is used for clash free timetable generation.

B. Fitness evaluation

The Fitness-Function is the most important component of any GA. It is responsible for determining the value of each chromosome, it allow us to distinguish the good from the bad (which are fitter than others) and in this way guide the GA into better Solution. In order to determine which chromosomes are fitter than the others, each chromosome must be evaluated using the imposed constraints. The objective of this research is to schedule automatically the final exams in a way to meet all constraints.

Various constraints for fitness evaluation are:

1) In one shift, number of exams scheduled should not be greater than three. If any chromosome violates this constraint, a penalty is imposed and it will be multiplied by the corresponding weighting constants to get the fitness value.
2) Gaps between the two exams should be even depending upon the number of subjects present in particular exam group. By this students will get time sufficient time to study. If any chromosome violates this constraint, a penalty is imposed and it will be multiplied by the corresponding weighting constants to get the fitness value.

Each solution, now is associated a numerical value that reflects their adaptation to the environment, or the conditions of constraints. Our aim is to maximize the fitness value. The more the value of fitness, the more chances of it to select for next better generation. The more fit chromosomes have lesser number of penalties, so that we can get the best timetable.

C. Roulette wheel Selection

During reproduction, chromosomes are selected from the combination existed in the population. Roulette wheel selection method is used here for reproduction. Good chromosomes having higher fitness values have high probability to be selected several times in the reproduction process. The higher fitness value has greater chance to be selected for the next generation. The fitter the chromosome, more copies of it is made. The best chromosome gets more copies, the average stay even and the worst die off.

From the above calculated fitness function, each chromosome has its own fitness values which reflects its good or bad characteristics. Roulette wheel is spun and the chromosomes having fitter values are selected for the next generation. The steps involved are:

i) Calculate the fitness value fval (vi) for each chromosome.
ii) Sum up all the fitness values to get the total fitness of population

$$F = \sum_{i=1}^{N} \mathbf{fval}\ (v_i)$$

iii) Calculate the probability, , of a selection of each chromosome ( )
 fval ( )/ F;
 Where i= 1, 2 …………………popsize
iv) Calculate the cumulative probability for each chromosome.

$$q_i = \sum_{j=1}^{i} \mathbf{P_j}$$

v) Generate a real value random number in the range [0 to 1].
vi) If r< , then select the first chromosome, otherwise select the i-th chromosome.

After selecting the parent chromosomes by the reproduction process, Crossover and Mutation operators are applied in order to vary the characteristics and results are checked for getting the optimal timetable.

D. Crossover

Crossover is considered as the primary genetic operator. The crossover operator generally combines genes from two parents to produce a child chromosome without any changes to the values of those genes. Crossover probability ( ) will tell how many of the offspring are exact copied of parents and how many are made from crossover the parents. A number of schemes are available to perform the crossover. Here one point crossover method is used. The following steps describe the crossover method:

i) Generate a random number (float) ‘r’ in the range [0 …… 1].
ii) If r < , select given chromosome pair for crossover.
iii) Generate a random integer number, pos, between 1 and (L-1), where L is the chromosome length.
iv) This generated number is chromosome site.
v) The two chosen chromosomes are replaced by their offspring’s.

Here the crossover is done between groups. If random number generated is more than the crossover probability, than no crossover takes place and the parents are as it is copied to the next generation. Crossover probability is usually kept high for crossover to take place.

E. Mutation

After the crossover is performed and before the child released into the wild there is a chance that it will undergo mutation. The chance of this occurring is referred to as mutation rate. The purpose of mutation is to inject noise. The mutation rate is usually defined as constant probability which says how often parts of chromosome (genes) will be mutated. A very high mutation rate will make the GA ineffective because the GA will change to random search. On the other hand, a very low mutation rate or no mutation at all makes the GA meeting convergence point among all solutions in the population. The mutation scheme will detect whether the mutation that is going to occur will contribute to the overall fitness of the chromosome in accordance with the involved constraints. The following steps describe mutation process.

i) Generate a random number q between 1 and 40.
ii) Check q lies in which group, that group will be selected for mutation.
iii) Generate a random number (float) 'r' in the range 0 to 1.
iv) If r< , mutation will take place. is mutation probability.
v) Now generate another random number between 1 and 40 and check whether this lies in that selected group or not.
vi) If not than the value of q is replaced by the newly generated random shift. If it lies than a new number is again generated[5,6]

V. V. RESULT & ANALYSIS

Initial Data for Timetable The construction of timetables for examinations requires the prior knowledge of some initial information taken for any Institution. It is necessary to collect data like the courses available, number of exams, number of time slots, number of exam groups etc. We will use MATLAB for simulating our work. MATLAB is widely used in all areas of applied mathematics, in education and research at universities, and in the industry.[7] MATLAB stands for Matrix Laboratory and the software is built up around vectors and matrices. Initial population is generated and genetic algorithm is applied to optimize the timetable. The process is repeated over many generations. Some of the generated parent chromosomes which represent the clash free timetable with their fitness values for generations. Growth graph includes the variation of mean fitness value and total fitness value from generation 1 to generation 150. The following graph shows the comparison of generation with Total fitness values. Horizontal axis shows Generation number with gap of 50. Vertical axis shows Total fitness value with gap of 5.
The following graph shows the comparison of generation with total fitness values. We observe that at various instants the value increases and at some instants mean fitness value decreases. From generation 1 to generation 180, the mean fitness value increases from 13.9546 to 26.1058 i.e. 45% increase.

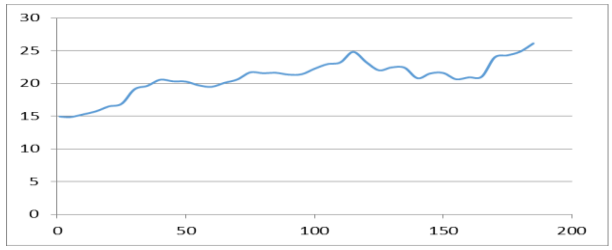

Figure 2: Generation vs. Totalfit

VI. CONCLUSION

GAs effectively demonstrated an ability to solve complex optimisation problems. Notably, this served to provide a very thorough introduction to the techniques employed and incorporated by Genetic Algorithms. Exam timetabling problem is an evolutionary problem, genetic algorithms would be a good method to solve this problem, and to find the best solution or timetable is related on setting the and best operators and parameters usage and configurations. Having finishing the work, we find that the topic Genetic Algorithms in general and the Genetic Algorithms in Timetabling Problem in particular under study demonstrated an ability to solve complex optimization problems. The optimal settings and configurations found have demonstrated their accurateness and suitability to produce better quality timetable representing very good solution to the problem being addressed.

VII. REFERENCES

[1] Schaerf, et. al., "A survey of automated timetabling", Artificial intelligence review,1999, 13(2), 87-127.
[2] Le Xiao, et. al., "An improved ant colony algorithm and its application in exam arrangement", 978- 1-4244- 5874, 2010 IEEE.
[3] José Joaquim Moreira1, "A System for automatic construction of Exam timetable using Genetic Algorithms", 2008, vol VI, Polytechnicalstudies review.
[4] Oluwasefunmi T. Arogundade, et. al., A Genetic Algorithm Approach for a Real-World Universit Examination Timetabling Problem", International Journal of Computer Applications (0975 – 8887), Volume 12– No.5, December 2010.
[5] Randy L. Haupt, et. al., "Practical genetic algorithms", 2004 by John Wiley & Sons, Inc., ISBN 0-471-45565-2.
[6] Omar Al Jadaan,et. al., "Improved Selection Operator For GA", Journal of Theoretical and Applied Information Technology, 2005 - 2008 JATIT.
[7] Y. J. CAO, et.al., "Teaching genetic algorithm using MATLAB", Int. J. Elect.Enging.Educ., Vol. 36, pp. 139–153, Manchester U.P, 1999.