Development of a Timetabling Software Using Soft-computing Techniques With a Case Study

M.F.M. Sabri, M.H. Husin, S.K. Chai
Faculty of Engineering,
Universiti Malaysia Sarawak
Kota Samarahan, Sarawak, Malaysia
e-mail: msmfaizrizwan@feng.unimas.my

***Abstract*—Timetabling is the assigning of an event to a particular timeslot in a timetable. Timetabling becomes a problem when the assigning task becomes hard to be implied where certain specific requirements need to be fulfilled. Genetic Algorithm (GA) emerge as one automation timetabling method to solve timetabling problem by searching solution in multi-points and the ability to refine the existing solution to a better solution. The timetabling program is specifically designed and developed in MATLAB programming language based on simple GA method to solve the timetabling problem for the case study. The developed timetabling program hardly found a good timetable solution as the solution finding converges at one point. Result shows the simple GA is not applicable in timetabling problem with high constraints and limited free slots.**

Keywords-Timetabling program; soft computing technique; Genetic Algorithm; automation timetabling method

I. Introduction

Timetabling is the assigning of an event with the availability of resources to a particular timeslot. Timetabling becomes a problem when the assigning task becomes hard to imply when specific requirements need to be followed. In a timetabling problem, requirements under consideration can be divided into two groups; i.e., hard constraints and soft constraints. Hard constraints are events or conditions that must be strictly taken into consideration. Soft constraints are the secondary considerations where a timetable which violates the soft constraints will still work but the timetable may not give optimal solution on behalf of an organization goal. A conventional way to solve a timetabling problem is to manually arrange events to be held into timeslots in a timetable. Only important requirements may be considered to reduce the problem complexity in a timetable and minimize the clashes of events in a timetable during the manually arrangement process. It's always time consuming and the timetable arrangement process needs to be redone when the preconditions in constructing a timetable changed. Investigation on the underlying algorithm such as Genetic Algorithm (GA), Simulated Annealing (SA), Tabu search, and random search for the timetable generator software has been made in order to find an efficient or universal algorithm to solve the timetabling problem. Tests show that GA and SA produced very good timetables where the results obtained from SA were slightly better than those using GA but GA was 54% faster than SA [1].

II. Genetic Algoritm (GA)

GA adopts the genes as its basic elements in algorithm and the genes form the chromosome which represents the possible solution for an optimization problem. Unlike classical single point search method, GA can work with a population of possible solutions at a time. GA has the ability to perform multi point searching where a population of solution is created among the population and the best solution is selected. Furthermore, GA can adopt more strategies into its framework in order to find a better solution. GA has been used in optimization based application such as finding optimal solution for a timetabling problem, scheduling the production in manufacturing system, direction finder optimization for positioning antennas and others. Several papers presented in Practical and Theory of Automatic Timetabling (PATAT) International Conference used GA in timetabling and scheduling [2], [3]. GA is also used with other optimization methods, such as the random search [1].

III. GA Based Timetabling

For school teacher/class timetabling problem, a study by Carrasco and Pato where a multi-objective GA is used and software called GAHOR was developed based on the algorithm[2]. GA is derived from the principles of natural genetics and the theory of evolution. GA is realized by the creation of a population of solutions represented by chromosomes. Chromosomes in the population are then going through a process of evolution. Evolution process consists of selection, crossover, and mutation operations. New generations better fit to the environment appear after each evolution process.

Gyori, Petres and Varkonyi-Koczy suggested that timetable is represented in a "set" form[11]. The set representation allows freedom in merging and splitting the space of timetable. The GA does not care for the internal structure of the individuals and for the meaning of data on which it executes the operators. So, the timetable has to be coded before algorithm is applied. For the timetable to be represented in set form, linearization is made both vertically and horizontally for the two dimensional timetable set as

978-1-4244-5586-7/10/$26.00 ©2010 IEEE