Implementation of Class Timetabling Using Multi Agents

M.Nandhini
Lecturer & Head , Department of Computer Science
Tagore Arts College, Puducherry-8, India,
mnandhini2005@yahoo.com

S.Kanmani
Prof & Head, Department of Information Technology ,
Pondicherry Engineering College, Puducherry-14, India,
kanmani@pec.edu

Abstract— **Timetabling problem means assigning activities to time slots such that various constraints and restrictions are satisfied. In Artificial Intelligence domain, the techniques viz., Graph coloring, Hill Climbing , Tabu Search, Simulated Annealing , Variable Neighborhood Search etc., have been used in which searching is made faster by applying heuristics . In this paper , Implementation of class timetabling with multi agents by Steepest ascent hill climbing algorithm has been proposed. It uses agents namely, CombinationGenerator which generates the maximum possible combinations for the inputting timetable and MinFinder which finds a combination with minimum evaluation function value for further successive examination . A improved heuristics is applied while generating timetables from random initial solution to the optimal solution to reach the goal at the earliest. Also, the architecture, operations and the interactions between multi agents while implementing class timetabling have been discussed**.

Keywords: Multi Agent; hill climbing; heuristic; constraints; combination ; optimal solution.

I. INTRODUCTION

Timetabling problem means assigning activities to time slots such that various constraints and restrictions are satisfied. Burke, Kingston and de Werra [1] defined timetabling as: a problem with four parameters: a finite set of timeslots ; a finite set of resources; a finite set of meetings ; and a finite set of constraints. The problem is to assign times and resources to the meetings so as to satisfy the constraints as far as possible.

A. *Class Timetabling Problem Description*

We consider a timetable model which is followed in one class/year of a science course of Pondicherry University . There are, Teachers- T {t1,t2,t3,t4,t5}, Subjects S-{s1,s2,s3,s4,s5} and Practical P-{pra1,pra2}. We assume that one subject has to be assigned to one teacher . Each subject in a class has to be scheduled in 5 or 4 distinct time slots in a week and practical have to be in 3 continuous timeslots in the afternoon session of 2 different days so as to satisfy the maximum of 30 hours in a week.

With this, a initial random timetable (TT) has been created as all the subjects and practical to be scheduled in a week of 5 days with 30 hours by satisfying the following hard constraints and a improved heuristics . From the initial solution, the optimal solution has to be obtained by finding a combination satisfying hard constraints , improved heuristics and having the least evaluation function value . Combination is a timetable obtained by altering the entries in the input timetable. Evaluation function value is depending on the soft constraints. This has been implemented using Steepest ascent hill climbing algorithm.

B. *Hard Constraints*

- Lectures of a teacher for a subject must be scheduled at different time slots in a week.
- All timeslots in a week should be scheduled. ie., no empty time slots.

C. *Soft Constraints*

- Maximum number of time slots for a subject in a day could be two.
- There could be a minimum of 2 time slots gap between same subject in a day.
- At least once, each subject should come in the first timeslot of the week.
- A subject should be allotted in minimum of 3 days in a week.
- Maximum in 2 days of a week, a subject can come in the same time slot.
- Adjacent days of a week could not have same subjects in the same timeslots.

The class timetabling problem is described in the mathematical form with objective function (Evaluation Function) and constraints (hard) as follows.
The Evaluation Function is ,

$$Z = \min \sum Ci * W_i \qquad (1)$$

Subject to the constraints of,

$a(i,j) = s \in S$; (i=1,2,3,4,5), (j =1,2,3,4,5,6)

$a(i,j) \neq 0$; (i=1,2,3,4,5), (j =1,2,3,4,5,6)

II. STEEPEST-ASCENT HILL CLIMBING WITH MULTI AGENTS

In our proposed hill climbing framework [3] for class timetabling , to distribute the work involving in , two co-operating agents[6], with different tasks and objectives have been introduced. Those are CombinationGenerator agent which generates the maximum possible combinations for the input timetable. The another one named MinFinder agent , finds a combination with minimum evaluation function (1) value for further successive examination.

A. *Agents Description*

CombinationGenerator agent is designed with the above mentioned hard constraints and with the proposed heuristic operations ; adjacent timeslots could not have the same subjects and teachers workload in a week should be equally distributed ie all teachers should have time slots in both forenoon and afternoon session to the most equally. When a timetable, which is selected as a better one and considered for further classification is assigned to this agent , it generates the various combinations satisfying the constraints and heuristics.

MinFinder agent is designed with all the above mentioned soft constraints as to find a combination with the least evaluation function value.

Based on the importance of soft constraints , weights are assigned to them and having its cost as either 0 or 1 depending on condition satisfaction factor. With costs and weights of all soft constraints , evaluation function values will be calculated for all combinations which are generated by the CombinationGenerator agent. From these evaluation function values , the least value is selected by the MinFinder agent and its corresponding combination is taken for further climb searching.

B. *Interactions between the Agents*

CombinationGenerator sends all generated combinations to the MinFinder , to find the goal state or better one combination to go to the next level. MinFinder gives the better combination to the CombinationGenerator for generating all combinations [9] and is shown in Figure .1

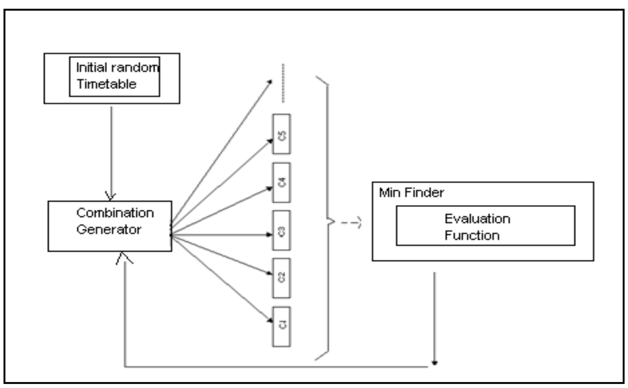

Figure 1. Multi Agents Interaction

C. *Implementation*

To reduce the search space , random timetable is generated by satisfying the above mentioned hard constraints and the proposed heuristic operations . To climb up for the better one, combinations satisfying hard constraints and heuristics are found by replacing each time slot in the input timetable with the remaining values of the subject set S , except for practical time slots. Thus, instead of creating maximum of 96 combinations , combinations which are satisfying the heuristic operations only will be present in the states space. From this set, a combination having minimum evaluation function value has been considered for further steps .This will be repeated till a combination satisfying all soft constraints has been received with the minimum threshold value , -21 or till getting a combination with the best evaluation function value than the earlier steps.

III. FUTURE WORK

With the effective result of this algorithm for class timetabling using agents by decomposing and distributing the main task , we planned to expand this scheduling using intelligent agents for our Pondicherry University Course Timetabling by introducing some more hard, soft constraints and applying heuristics to improve the search process.

IV. CONCLUSION

By applying the heuristics in generating the combinations, the search space has been reduced and gives the optimal at the earliest. By introducing the agents, CombinationGenerator and MinFinder the work has been distributed and the complexity of main task will be reduced.

REFERENCES

[1] E K. Burke, D.de Werra and J.H.Kingston ,"Applications to Timetabling," *In: J.Gross and J.Yellan(eds).* The Handbook of Graph Theory, Chapman Hall/CRC Press, 2004, pp. 445--474.

[2] E. K. Burke and J. P. Newall , "Enhancing Timetable Solutions with Local Search Methods," LNCS, vol. 2740, pp. 195—206,2003.

[3] Wikipedia , Hill Climbing , http://en.wikipedia.org/wiki/Hillclimbing

[4] E. K. Burke, A. J. Eckersley, B. McCollum, S. Petrovic, R. Qu,, "A Case Based Heuristic Selection investigation of Hill Climbing, Simulated Annealing and Tabu Search for Exam Timetabling Problems," University of Nottingham, 2002.

[5] Andrea Schaerf ,"Tabu Search techniques for large High-School Timetabling problem," Netherlands, 1996.

[6] Carla Silva, Rosa Pinto,Jaelson Castro,Patricia Tedesco ,"Requirements for Multi-Agent Systems ," Brasil,2003, pp. 198—212.

[7] Luca Di Gaspero, Stefano Mizzaro and Andrea Schaerf ," A Multi Agent Architecture for Distributed Course Timetabling ," Udine, Italy,2004.

[8] Nunohiro, E., and Mackin, K. J., "Applying Multi-Agent Algorithm to a class scheduling System ," Japan ,2005.

[9] Mihaela Oprea , "MAS_UP-UCT :A Multi-Agent System for University Course Timetable Scheduling ," In International Journal of Computers, Communications & Control Vol. II No. 1, 2007,pp. 94—102.

[10] Irina Tudor,Madalina Carbureanu, "Timetable Planning Using Intelligent Agents," Romania,2008.