The Fralon algorithm to the Post Enrolment Course Timetabling

Abstract. Course Timetabling is a well known NP hard problem. Here is presented an algorithm named Fralon, which was developed for Course timetabling problem of Patat track2. Fralon finds a feasible solution using Simulated Annealing (SA). Then SA is again applied in order to improve the initial solution. The final solution is improved by a Tabu Search. This algorithm designed to find feasible solutions.

1 Introduction

Fralon is a hybrid algorithm of Simulated Annealing (SA) and Tabu Search (TS). Fralon has two parts: one for reach a Feasible Solution (FS) and other to improve it. The first part applied a heuristic based on SA with Extra Timeslots (this heuristic is named SAET) to find a FS, if a FS is not found by SAET, then a hybrid SA with TS and without extra timeslots is executed until a FS is found or the limit of time is reached. In the second part, Fralon improves the FS previously founded; this hybrid SA-TS is applied in a sequential way. SA is executed until the thermical equilibrium is achieved, and then TS is applied is local search. This SA-TS sequence is applied until the best FS solution is founded or the limit time is finished.

2 Algorithm Description

After read the input file, every event is placed in a room and timeslot sequentially, the first events is placed in the first room and first timeslot, the second event is placed in the second room and first timeslot, the third event is placed in the third room and first timeslot, until all the rooms of the firs timeslot are occupied, then the second timeslot is open, and the next event is placed in the first room and second timeslot, and so on. Once all the events are placed in a unique combination of room and timeslot, the events violating at least a hard constraint are detected, and are placed in extra timeslots in such a way that they do not violate any of the first three hard constraints. The two last hard constraints are ignored for the events which are placed in the extra timeslots. SA is then executed searching to minimize the number of extra timeslots. This algorithm is called Simulated Annealing Extra Timeslots (SAET).

2.1 Simulated Annealing Extra Timeslots (SAET)

In SAET, the objective function is the number of students in the extra timeslots, if a student takes three events placed in the extra timeslots he count by three violations. The temperature is the number of events minus 45 (the number of normally periods). The SAET is tuned using a Markov approach published in [1]. The maximum length of the Markov chain is set to 10000. In the cycle of metropolis, a random event is chosen, then a feasible neighborhood is calculated (only by making the movements that do not introduce any violation in the hard constraints for the normal periods or they do not introduce any violation in the first three hard constraints for the next periods). SAET is stopped until the temperature cycle is frozen (the temperature is equal to 0.01) or a FS is founded. A FS found by SAET will be the input of a SA that search to find the best FS (this SA is called SAO). If SAET does not find a FS, its best solution will be the input of another SA which again will try to find a FS (this SA is named SAF).

2.2 Simulated Annealing to find a Feasible solution (SAF)

Whether SAF is called when SAET did not find a FS, then extra periods are compressed by a function called Compressed_Solution (CS). The function CS only reallocates the events that are into the extras periods to the spaces of the first 45 periods, although this means to violate the hard constrains. The temper initial of SAF is set to 470 * number of students + number of events. The cooling scheme of SAF is geometric. The alpha is 0.99 and the SAF stop when the temperature reach 0.01 or a feasible solution has been found. In the cycle of metropolis, a random event is choose, and then its neighborhood feasible is calculated (only the moves that do not introduce a new violation of the hard constraints). If at end of a metropolis cycle, the global solution is not improved. The best solution of SAF will be the input of a Tabu Search to find a feasible solution (TSF), it seeks in all neighborhood seeking improve the objective function. The best solution of TSF is return to SAF and so on, until a feasible solution is found or the time limit is reached. If a feasible solution is found, this will be the input of Simulated Annealing to Optimize the solution (SAO)

2.3 Simulated Annealing to Optimize the solution (SAO)

The temper initial of SAO is set to 470 * number of students + number of events. The cooling scheme of SAO is geometric. The alpha is 0.99 and the system is frozen when the temperature reach 0.01, ten metropolis cycles in a row give the same solution or the time limit is reached. For count the cycles with the same solution the temperature will be less or equal to –8 * Maximum capacity of rooms / log (0.95). The SAET tunes the markov chain using a scheme published in [1]. The maximum length of the markov chain is set to 10000. In the cycle of metropolis, a random event is choused, and then its neighborhood feasible is calculated (only the moves that do not introduce a violation of the hard constraints). The output of SAO is the input of a Tabu Search to Optimize the solution (TSO).

2.4 Tabu Search Optimize the solution (TSO)

This implementation of TS seeks in all neighborhood seeking improve the objective function. For the Tabu algorithm proposed here, two neighborhoods called N1 and N2 are implemented:

- N1 neighborhood: For a given solution, two steps to get a neighbor of it should be done. These steps are: a) Two events are interchanged and then b) A movement of an event to an available place is done.
- N2 neighborhood: A neighbor is found using the two steps of N1 neighborhood and then applying the next step: c) Interchange all the events of two periods k_1 and k_2 where $k_1 \neq k_2$.

Both of these neighborhoods take care of do not violate any hard constraint, and the entire neighborhood is considered. If exists a movement that improve the best solution found is made, else if exists a movement without the status tabu in the events considered, the move is made. Otherwise, a random movement of the event with less frequently moved and without status tabu is made. In all the movements realized, the events are put in the tabu list. Every once an event is move, the frequency of event is incremented by one.

The tenure in the tabu list is a random number b $\in Z^+$ between a [20, 40).

In the Tabu algorithm proposed her, the stop criterion is as follows: a) The optimum solution (zero hard constraints violated and zero soft constraints violated) is found or b) Maximum number of iterations without improve the last best solution is reach. (100 iterations were used in this work). or c) The time limit is reached.

Finally, if any solution feasible is found, all the events implied in any hard constraints all set in room minus one and period minus one.

References

1. Sanvicente-Sanchez, H. and Frausto, J. (2004). Method to Establish the Cooling Scheme in Simulated Annealing Like Algorithms. International Conference, Assis, Italy. ICCSA (2004) LNCS Vol. 3095. 755-763.