Timetabling Competition TTComp 2002: Solver Description

Luca Di Gaspero and Andrea Schaerf

31st March 2003

Our solver is based on the local search paradigm. It is composed by three stages that are interleaved sequentially, and the whole process is repeated as long as the total time is expired. Each stage starts from the final state reached from the previous one, and the first stage starts from a random state.

The search space is represented by two integer-valued vectors of size $|E|$ (where E is the set of events) that store, respectively, the timeslots and the rooms assigned to each event. The neighborhoods employed are described below as modifications of the above vectors.

Stage 1: Hill climbing

The hill climbing procedure draws a random move at each iteration. The move is accepted if it improves or leaves unchanged the value of the cost function.

The neighborhood used is the set-union of the following three neighborhoods:

ChangeTime: changes the timeslot of an event (the room is left unchanged);

ChangeRoom: changes the room of an event (the timeslot is left unchanged);

SwapTime: swaps the timeslots of two events, such that each takes the room of the other.

The random selection is done in two steps: first we select the basic neighborhood, with uniform probability (1/3 each), and then we uniformly draw a random move from the selected neighborhood. Only non-trivial moves are generated and evaluated; that is, we avoid to move an event from a room r to the room r itself (and similar ones).

This stage stops after 300,000 idle iterations (i.e., iterations without an improvement).

The final state of this stage is the last one generated, which has the minimum cost among the visited ones.

Stage 2: Tabu search

The tabu search procedure uses a variable-length short-term tabu mechanism (without any form of long-term prohibition). The neighborhood used is the following:

ChangeBoth: moves an event in a different timeslot and a different room.

Not all the neighbors are evaluated, but only those that satisfy the following conditions:

- the event must have at least 3 students;
- the target room must be free in the target timeslot;
- the move is not tabu, that in our case means that it does not involve the same event of a move in the tabu list.

All moves that satisfy all three conditions are generated and evaluated for the selection of the best move at each iteration.

The stopping condition of this stage is a compound one: the stage stops after a number of idle iterations that is inversely proportional on the density of the instance or when a fixed number of iteration has elapsed. We allow less idle iteration to denser instances in order to allow roughly the same running times for each round. There is also a maximum number of iterations allowed. In particular, we grant the algorithm a total (relative) number of iterations equal to 3 times the number of idle iteration.

The tabu list length varies from 20–30 to 30–40 according to the density of the instance.

The final state of the stage is the one with best cost among the visited states.

Stage 3: Multi-swap shake

This stage performs only one single move, selecting the best neighbor from the following neighborhood:

MultiSwap: all events in a timeslot are move to a different timeslot, and vice versa. Rooms are kept unchanged.

The final state of this stage is the state obtained performing the best move, even if this is a worsening move.