School Scheduling Optimization Application

The algorithm that we developed is based on Ant Colony Optimization (ACO). ACO is a metaheuristic proposed by Dorigo. The inspiration of ACO is the foraging behavior of real ants. The basic element of ACO is the use of a probabilistic solution construction mechanism based on stigmergy.

ACO has been applied successfully to numerous combinatorial optimization problems including the traveling salesman problem, quadratic assignment problem, scheduling problems, and others.

Before start of the optimization application, the user inputs into the application:

• The total number of students

• The total number of students per each class

• List of classes taken by each student

• The total number of teachers

• The total number of teachers per each class

• A matrix indicating student clashes between events is created.

• List of classes by each teacher

• List of classrooms with number of chairs in each classroom

• Type of class (lectures and seminars are different types of class)

• For each event a list of possible rooms is created, (this is done by analyzing room sizes, features provided by rooms, and also number of students attending each event, and features required by each event).

• For each room it is established how many events may be placed in it.

There are a number of soft constrains in the algorithm and a number of hard constrains as well.

Hard constrains are constrains that make solution generated by an ant infeasible. Suitable solution is the solution that will not violate any hard constraints. Examples of hard constrains are: number of classrooms, number of chairs in each classroom.

The soft constraint aims to increase the quality of the already feasible solution. The soft constrain is run only if the solution is feasible. Examples of soft constrains are: number of teachers, number of students.

When optimization algorithm starts its work, in each iteration of the algorithm, each of the "ants" assigns events into times and classrooms, i.e., the ants choose the timeslot and room for every inputted class event probabilistically.

Once all the ants have constructed their assignment of events into places, a local search routine will be used to further improve the solutions.

Furthermore, one of the main objectives is to minimize a number of days for teachers and for students (i.e., if it is possible that a particular student will be at school only 4 days a week and not 5 – the system will obtain that solution).

In the output of the algorithm, the timetable is represented in the form of an integer matrix.

Then, it will be presented in a user-friendly GUI for user's viewing and printing.