Jonathan Kelaty

Manal Zneit

Course Scheduling Using Constraint Satisfaction

Project Description

As the project title subtly implies, our project will be on course scheduling using constraint satisfaction. Given datasets of certain variables such as the number of professors and the courses they can teach, the possible time slots that a course might fill, the available classrooms to be assigned, and potentially other constraints, we will implement a search algorithm to find an optimal course schedule. We can formulate the problem as a CSP in different ways, for example, we can have our constraints include the workload that is put on individual professors to ensure courses are distributed as equally as possible to avoid overloading an individual professor with too many courses. We will also include a course conflict constraint to ensure that a professor isn't scheduled to teach two courses simultaneously. We might also introduce soft constrains such as preferences for each professor's time slots or having classrooms used on the appropriate floor and building for a course's corresponding department. We might also choose to use heuristics that favor schedules where multiple offerings of the same course are more dispersed throughout the day to ensure students are offered a variety of time slots to accommodate their schedules.

CSCI 350/761 Topics

Constraint Satisfaction Problem with backtracking search: Both team members will be involved in devising the high-level overview of the algorithm and heuristics to be used.

Team Members

Both team members will contribute toward developing the heuristics involved in evaluating how "good" or "bad" a given course schedule is, as well as a high-level overview for the search algorithm.

Jonathan Kelaty: Will implement the search algorithm, as well as tune heuristics and weighting accordingly.

Manal Zneit: Devising datasets to test, as well as normalizing real-world datasets to compare our results vs course schedules that are actually used. Will write the demo which will include a visualization for the search algorithm.

Demo

Once we have finalized our domain, variables and constrains, we may try to create a visualization tool of the search to see how it navigates the search tree. We may also include results of feeding in real-world data sets and comparing the results using weighted scores.

Evaluation

In order to evaluate our project, we can devise large datasets for to feed into our program and try to give an objective evaluation to the results based on certain characteristics that we want to see in course offerings, such as course offerings for day and night timeslots, evenly dispersed workload for the professors, or how frequently we met the professors' preferences. We can also take real-world datasets, such as Hunter's course offerings, and compare the course schedule that our program came up with compared to the one Hunter uses.