Thesis Defense

Master of Science in Computer Science

“Using Visualization and Integer Linear Programming for University Class Scheduling”
by Kristi Bushman

Date: July 7, 2021
Time: 12:30 – 2 p.m.
Place: RSVP to receive the Zoom link

Committee:
- Dr. Alexandros Labrinidis, Professor, Department of Computer Science, School of Computing and Information, University of Pittsburgh
- Dr. Panos Chrysanthis, Professor, Department of Computer Science, School of Computing and Information, University of Pittsburgh
- Dr. Kirk Pruhs, Professor, Department of Computer Science, School of Computing and Information, University of Pittsburgh

Abstract:
University class scheduling is the task of assigning a room, instructor, and timeslot to each class in a university schedule. This is a highly-combinatorial task, with multiple constraints and goals. Often, these goals are in competition with each other, meaning improving one metric may deteriorate another. Traditional scheduling workflows have involved the use of spreadsheets and whiteboards to keep track of assignments. These tools may be adequate for small-size schedules, however, as the size and complexity of the schedule increases, these tools become increasingly difficult to use. In those cases, schedules may contain errors or undesirable assignments, such as room conflicts or class conflicts; it is also very difficult to evaluate the quality of the resulting schedule. In this thesis, we introduce a web-based tool that can be used to support the scheduling process. The tool includes multiple ways to visualize a schedule. These visualizations can help the user to quickly identify conflicts or problem areas. Users can make changes to instructor, room, or timeslot assignments and quickly reassess the quality of the resulting schedule. By utilizing the calendar paradigm, making changes to the current schedule is a very intuitive process. Further, we provide the ability for faculty to provide their teaching preferences (in terms of which courses to teach and also which days/times); this information is integrated with the rest of the schedule making it very easy to identify good assignments of instructors to courses. Finally, our tool has an automated scheduling feature, which allows the system to make all scheduling decisions rather than the user. The automated scheduler uses an integer linear programming model to describe the scheduling problem and its constraints. The linear programming model is optimized to reflect multiple scheduling goals that are required by the Computer Science Department of the University of Pittsburgh. We experiment with different ways to use the automated scheduler, specifically with respect to combining and prioritizing different metrics.