Math 480 - Course Project
Creating a schedule for "Chaos Week"

Derek Rhodeham
Justin Zecak
Davis Zvejnieks

1   Description

Jonah Ostroff runs a 6-week math camp for high school students. Between the the first three weeks and the last two weeks is "chaos week." During "chaos week" a new set of classes are voted on by students, and they are enrolled in a week long program containing their top five classes.

In 2015, there were around 30 available topics, which students would vote for the most interesting classes. These classes were narrowed down to 10 classes, which the three instructors could teach. Each student could then vote 0 to 3 on the remaining classes, 0 representing no interest, 3 representing most interest. Each student was then assigned five classes, which they would participate in daily for the week. Students were assigned classes not only by their most preffered class, but also by instructor overrides based on their knowledge and performance of the student. There were 5 time slots available, so 2 classes were taught at the same time.

Jonah would like a schedule for 2016 so that there are no time conflicts between classes an instructor could teach while maximizing the preferences of students. This year the schedule should accommodate 24 students, 4 to 5 instructors, and 15 classes. This would mean three classes will be taught at each of the 5 time slots.

2   Impact

Jonah had created a method searching for a bipartite graph to represent classes which would represent classes that would share a timeslot. This method does not scale easily to finding a 3-matching in a hypergraph. More complexity arises if the schedule were to accomodate 20 classes or 25 classes.

We propose a method that easily scales to larger schedules, thus allowing the instructors easily schedule regardless of the class size.

3   Methods

Use LP relaxation to solve the issue of 3-matching. We can easily adjust the constraints to change from finding 3 simultaneous classes to 4, 5 or more. We can include instructor overrides as a larger constant when considering preference, to avoid having to hand tailor class selection as previously done in 2015.

4   Example

5   References