Class Scheduling to Maximize Participant Satisfaction

Objective of the Paper:
Darden Graduate School of Business Administration of the University of Virginia decided to try out a new system for course selection by which they would be allowing students to enroll in a course with their preference of time. This was done because in 1989-1990 over one-third of the second year students reported that they could not get into two or more classes they sought and 15 percent could not get into three or more classes. This problem had to be addressed as soon as possible as students had only one year to complete all their electives. This paper by Scott E. Sampson, James R. Freeland and Elliot N. Weiss addresses this problem and gives a solution to this problem.

Problem Description:
The first year students at Darden have no course choices hence they cannot indicate their preferences. The second year program is different. It consists of two semesters (fall and spring). Each semester has two non-overlapping halves. A particular course will be taught either early-week (includes Mondays, Tuesdays and sometimes Wednesdays) or late-week (includes Thursdays, Fridays and some Wednesdays). An early-week course never overlaps with a late-week course. Fifty-eight of the sixty-four courses offered are half semester in length. So the problem can be considered to have eight sub-divisions (2 semesters * 2 semester halves * 2 week halves). Each elective is offered only during one of the eight segments (multiple sections may be offered during that segment). There are 6 full semester courses. These full semester courses are considered as two half-semester courses. If the first half is scheduled then automatically the second half is scheduled. Students were not allowed to enroll in only half of a full-semester course.

The class-scheduling process consists of the following steps:
1) The Dean’s office specifies courses (which will be offered during each of the eight segments) based on projected enrollments and assigns instructors to courses.
2) Course lists are made available to students and contacts are listed so that students can get an idea about the course from people who have already taken them.
3) Students enter rankings for the desired courses, which goes directly into a selection database.
4) Faculties have some requests that need to be considered too. They are taken in as problem constraints (for e.g. a faculty might want to schedule his class before or after lunch so that the students can talk to the guest speaker during lunch).
5) A schedule is developed using a heuristic procedure because after constructing a mathematical model that included both students desires and resource constraints they found that the number of integer variables rendered a general integer programming solution impractical.
6) The resulting class schedule is shown to both the dean and the faculty. If they have a conflict, then the heuristic procedure is repeated.
7) If a professor felt that he had a difficulty teaching the class assigned to him by the schedule, then the heuristic was run again with an additional constraint, which would address that particular professor’s problem. If the result from this happened to affect a lot of students (say 20 students were denied this course because of this added constraint) then the professor had to give in.
8) Final copies of the schedules are handed to students indicating the course schedules in which they are enrolled.

Formulation:
PREF \(_{s,c}\) (the preference ranking student \(s\) specifies for class \(c\))

\[
E_{s,c,t} \begin{cases} 
1 & \text{if student } s \text{ is enrolled in class } c \text{ at time period } t \\
0 & \text{otherwise}
\end{cases}
\]

\[
S_{c,t,r} \begin{cases} 
1 & \text{if an offering class } c \text{ is scheduled at time } t \text{ in a room of type } r \\
0 & \text{otherwise}
\end{cases}
\]

CAPACITY\(_{c}\) (enrollment capacity of a section of course \(c\))

1

Paper Published in INTERFACES Volume25: May – June 1995
Abstract prepared by Sanjeevkumar Uruvandavida

Objective: \( \sum_{s,c,t} (\text{PREF}_{s,c}) E_{s,c,t} \) (To maximize students enrollment preference)

Subject to

1. \( \sum_c E_{s,c,t} \leq 1 \ \forall \ s, t \) (ensures that each student is assigned to only 1 course at each time period)

2. \( \sum_t E_{s,c,t} \leq 1 \ \forall \ s, c \) (ensures student isn’t enrolled for more than 1 section of a particular course)

3. \( \sum_s E_{s,c,t} \leq \sum_r S_{c,t,r} \text{CAPACITY}_c \ \forall \ c \) (serves 2 purposes)

   a) Ensures enrollment only in courses scheduled at a particular time interval b) enrollment within capacity

4. \( \sum_{r,c} S_{c,t,r} = \text{SECTION}_c \ \forall \ c \) (ensures each section of each course is scheduled)

5. \( \sum_{c \in \text{TAUGHTBY}_p} \sum_t S_{c,t,r} \leq 1 \ \forall \ p, t \) (ensures that a professor isn’t assigned 2 classes at the same time)

6. \( S_{c,t,r} \leq \text{ALLOW}_{c,t,r} \ \forall \ c, t, r \) (allows courses to be restricted to certain rooms or time periods or both)

7. \( \sum_c \sum_{t \geq r_1} S_{c,t,r} \leq \sum_r \text{ROOMS}_{t,r} \ \forall \ t, r \) (permits the schedule to meet classroom quantity)

Heuristic Solution Procedure:

The problem was solved by using modified local-search procedure. A feasible schedule was located. Once this was done they tried to reassign a given course section to other time periods. If this was feasible the objective value was computed. If better, then the new schedule was accepted else rejected. They also avoid staying at the current assignment for a long period of time i.e. they tried getting a better schedule by starting with different initial feasible solutions. Sometimes schedule feasibility is ignored to find a solution space with better objective value. The search was terminated after a fixed number of iterations.

Conclusions:

Using this procedure to schedule classes it was seen that less than 5% of the students experienced unmet requests. I feel that this procedure of scheduling classes is very efficient. This model has other applications such as 1) Conference Scheduling (objective: to maximize ability of participants to attend sessions of interest) 2) Multi-event sports competition scheduling. This paper explains a type of problem where it is easier to use a heuristic approach rather than use integer programming. I would say that the authors of this paper have taken care to explain things in a very clear fashion enabling a person with less than average knowledge on optimization understand the concept clearly.