Discrete Applied Mathematics 159 (2011) 812–825

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

A direct barter model for course add/drop process

Ali Haydar Özer, Can Özturan *

Department of Computer Engineering, Boğaziçi University, 34342, Bebek, Istanbul, Turkey

ARTICLE INFO

Article history:
Received 31 July 2009
Received in revised form 4 January 2011
Accepted 6 January 2011
Available online 5 February 2011

Keywords:
Add drop
Student registration
Timetabling
Course scheduling
Bartering
Barter network
Network flow

ABSTRACT

Even though course timetabling and student scheduling problems have been studied extensively, not much has been done for the optimization of student add/drop requests after the initial registration period. Add/drop registrations are usually processed with a first come first served policy. This, however, can introduce inefficiencies and dead-locks resulting in add/drop requests that are not satisfied even though they can, in fact, be satisfied. We model the course add/drop process as a direct bartering problem in which add/drop requests appear as bids. We formulate the resulting problem as an integer linear program. We show that our problem can be solved polynomially as a minimum cost flow network problem. In our model, we also introduce a two-level weighting system that enables students to express priorities among their requests. We demonstrate improvement in the satisfaction of students over the currently used model and also the fast performance of our algorithms on various test cases based on real-life registration data of our university.

1. Introduction

In universities, course timetabling (CT), student scheduling (SS) and add/drop processes involve the coordination of various resources and entities. CT basically deals with the allocation of time slots and classrooms to courses by taking into consideration issues such as preferences of instructors and classroom locations. Given a timetable, in SS phase, students select courses according to their needs and preferences. Because of course and section quota restrictions or enrollment balancing requirements among the sections, it is not possible to satisfy the needs and preferences of all the students. Therefore, some policy or algorithm needs to be employed in SS phase for the assignment of students to courses and sections. During the add/drop phase, a readjustment of the assignment solution in SS phase basically takes place by the addition, dropping and swapping of courses and/or sections. In the literature, phases CT and SS have been extensively studied (see, for example, surveys [6,7,20,27]). Some approaches tackled either CT or SS exclusively. Some approaches coupled these two phases and solved the combined course timetabling and student scheduling problem. In this paper, our focus will be on the add/drop process. Not much has been done for this phase—we are aware of only one work (that of Graves et al.'s [14]) that addresses the add/drop process. The add/drop process has an important difference from that of CT and SS. A student may have been already assigned to a seat in a course or section from SS phase and he may want to swap (barter) this seat that he owns with another seat owned by other students in another course or section. Hence, one can say that whereas CT and SS phases can be modeled as an assignment problem, for add/drop process bartering is a more appropriate model.

We were motivated to develop a direct barter model for the add/drop process because of some problems we noticed during add/drop periods at our Boğaziçi University. Since 1998, web based online registration system has been used for course registration [5]. Before the beginning of each semester, students are admitted to the system and are allowed to take courses if both prerequisites of the courses are satisfied and the quotas of the courses permit. The system works on a first

* Corresponding author. Tel.: +90 212 359 7225; fax: +90 212 287 2461.
E-mail addresses: ozeralih@boun.edu.tr, ahozer@gmail.com (A.H. Özer), ozturaca@boun.edu.tr (C. Özturan).

0166-218X/$ – see front matter
doi:10.1016/j.dam.2011.01.004

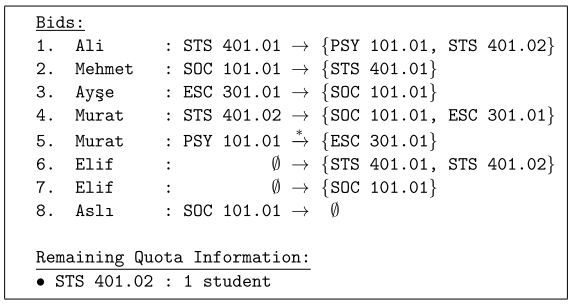

Fig. 1. Example problem for illustrating add, drop, and barter bids.

come first served (FCFS) policy basis and at the beginning of each registration period, a race occurs among students for popular courses. Generally, the quotas of the popular courses are filled within the first few hours of online registration period. After the registration period, the semester begins and during the first week of the semester, the students attend and evaluate their courses. At the end of this week, add/drop period of one week begins and the students are allowed to change their courses and/or sections of their courses. Because of the FCFS basis of the system and the quota restrictions, when a student drops a course, he may not be able to take it again. This situation forces a student who wants to change his course, to first try to add a new course, and then drop the old course. Although this does not pose a problem if the quotas of the courses are not full, it does pose a problem for the popular courses. It is observed in Boğaziçi University student registration system that the current FCFS based system causes deadlock situations, and hence reduces the total satisfaction of students. Although different implementations of FCFS approach exist in different registration systems, all FCFS based systems are prone to the same problem. For instance, in UniTime [21,29], which is an open source enterprise system for automated construction of course timetables and student schedules, when a student wants to add a course which is not available, the student is assigned to the wait-list of that course. Wait-lists are processed automatically in FCFS manner as one seat becomes available for the corresponding course. Therefore, since a student who wants to change his course cannot be sure whether he would be assigned to the new course, he would not want to drop the course he has already assigned until he obtains a seat in the new course. Thus, this would also lead to the same problem.

In order to increase the efficiency of add/drop process compared to the current FCFS based system of our university, a direct barter model for the course add/drop process is proposed. The objective of the model is to increase the total satisfaction of students while preserving fairness among them. For this purpose, along with the usual add and drop requests, this model allows students to barter the courses they want to drop for the courses they want to add. Students express their requests through submitting multiple add, drop and barter bids and in each add and barter bid, they can declare a set of alternative courses to be added. Besides, in this model, they can indicate relative priorities of their bids and the courses they want to register for. For instance, if a student prefers course A over course B, and course B over course C, he just declares $A \succ B \succ C$. Furthermore, students can request the same course or the same set of courses in multiple bids and can also declare restriction sets in which only one course can be added to the schedule.

In this paper, we contribute a formal development of the model. We present a network flow based algorithm that allows us to solve the problems in strongly polynomial time. We also compare the solutions of our model with that of the FCFS approach based on real-world student registration data and present the performance of our algorithms on various tests.

In the next section, we present an example with which we explain our model for the course add/drop process. In Section 3, we formally define and formulate our model using integer programming. Then, in Section 4 we present a minimum cost network flow solution of our problem and in Section 5, we present the experimental results. A review of the related literature is given in Section 6. Finally, the paper is concluded in Section 7.

2. A motivational example and the model

In this section we present an example scenario for add/drop process on which we explain our direct barter model. Assume that during the registration period, students Ali, Mehmet, Ayşe and Aslı have been registered for courses STS 401.01, SOC 101.01, ESC 301.01 and SOC 101.01 respectively. Murat, on the other hand, has been registered for both STS 401.02 and PSY 101.01. Suppose that during the add/drop period, the students declare *add*, *drop*, and *barter bids* as shown in Fig. 1.

Bids 1–5 are examples of a barter bid. In a barter bid, the left hand side of the arrow indicates the course to be dropped and the right hand side indicates the course to be added. A barter bid as the name suggests enforces the student to drop the course on the left hand side if he adds the new course on the right hand side. For instance, in bid 3 Ayşe wants to drop ESC 301.01 if she could add SOC 101.01 to her course list. Bids 6 and 7 are examples of an add bid. An add bid states that the student wants to add the course on the right hand side without dropping any other course. Likewise, a drop bid, e.g. bid 8, states that the student wants to drop the course on the left hand side without adding any other course.

Bids 1, 4 and 6 are different from the others in terms of having a *request set* of more than one course on the right hand side. These bids are called *multi-bids*. A *multi-barter bid* states that the student is indifferent, at least to some degree, to the