

# Drawing System of Students for Inter-University Course with Limited Number of Enrollments

Ken'ichi Fujimoto, Hiroshi Murai, Toshihiro Hayashi, Naka Gotoda, and Rihito Yaegashi

**Abstract**—The five national universities in Shikoku area of Japan have inter-university courses through a cooperative e-Learning project. Hundreds of students at the universities may take inter-university courses, because they can study the courses online using e-Learning systems. However, to realize all applications from students may be difficult owing to the capabilities of e-Learning system, computer network, and teachers. Because of that, to restrict the number of enrollments for respective courses is compelled. In this paper, we present a system to draw applicants for inter-university courses with a limited number of enrollments. The proposed system has parameters to cope with the type of courses and students' preference. Our experimental results for a toy problem demonstrate that the proposed system works well.

**Keywords**—Cooperative e-Learning project, inter-university course, limited number of enrollments, drawing system, mixed-integer linear programming, branch and bound method.

## I. INTRODUCTION

In general, students at national universities of Japan apply for courses (subjects) that they want to study before the opening of the courses. Although students should have the right to take courses, in a face-to-face class, the number of enrollments that are students can study the course may be restricted from hardware limitation such as the area of lecture rooms and the number of personal computers (PCs) furnished in a practical room [1]. The limit on the number of enrollments is needed even e-Learning lecture courses owing to the capabilities of teachers, a computer network, and e-Learning servers [2].

In Shikoku area of Japan, five national universities are established: Tokushima University, Naruto University of Education, Kagawa University, Ehime University, and Kochi University. We established the University Consortium for e-Learning, Shikoku Center in 2013, and a project on cooperative university education using e-Learning systems is currently underway [3]. The e-Learning courses provided through the project are opened simultaneously at the five universities. Because of that, an e-Learning course had hundreds of students. From the aforementioned limitations, we need to restrict the number of enrollments for each course.

We have two methods to restrict the number of enrollments. One is to set the upper limit of enrollments for every course, and

another is to set that for every university in a course. In the former case, the shortage of slots for enrollments in a course may occur. On the other hand, the latter setting contains the problem that a student cannot take a course although the course has a vacancy in slots of another university. To handle the trade-off, we need a flexible system to draw applicants who are students apply for inter-university courses.

In this paper, we treat the case of the former setting and present a system to draw applicants for inter-university courses with a limited number of enrollments. The proposed system has parameters corresponding to the type of courses (compulsory or optional) and students' preference. To confirm whether the proposed system works well, we also experiment for a toy problem.

## II. SYSTEM DESCRIPTION

We explain our drawing system briefly. A problem to draw applicants (students) for courses with a limited number of enrollments can be formulated as an optimization problem called mixed-integer linear programming:

$$\begin{aligned}
 &\text{Minimize} && \sum_{i \in M} \sum_{j \in S} -f_{ij} \cdot x_{ij} \\
 &\text{Subject to} && \sum_{i \in M} x_{ij} \leq N_j \\
 &&& \sum_{j \in S} x_{ij} \leq b_i
 \end{aligned} \tag{1}$$

Here,  $M = \{1, 2, \dots, I\}$  represents a set of  $I$  students,  $S = \{1, 2, \dots, J\}$  expresses a set of  $J$  courses,  $N_j$  is the upper limit of enrollments in the  $j$ th course, and  $b_i$  represents the upper limit of courses that the  $i$ th student can take. The parameter  $f_{ij}$  corresponds to the priority of the  $i$ th student's preference for the  $j$ th course and is set to a high value if the  $j$ th course is compulsory for the  $i$ th student or the  $i$ th student's preference for the  $j$ th course is high. We also define  $x_{ij} \in \{0, 1\}$  as a binary variable shows whether the  $i$ th student can take the  $j$ th course, where zero means that the student cannot take the course, and one implies that the student can take it, i.e., this corresponds to drawing results.

This formulation enables us to use a practical method of solving such as a branch and bound method [4], besides, MATLAB that is a powerful tool for numerical calculation has a function "intlinprog" based on a branch and bound method to find a solution to a problem of mixed-integer linear programming.

Manuscript received May 20, 2016.

K. Fujimoto is with Kagawa University, Kagawa 760-8521, Japan

H. Murai is with Kagawa University, Kagawa 760-8521, Japan

T. Hayashi is with Kagawa University, Kagawa 761-0396, Japan

N. Gotoda is with Kagawa University, Kagawa 761-0396, Japan

R. Yaegashi is with Kagawa University, Kagawa 761-0396, Japan

III. PROBLEM DESCRIPTION

To confirm the effectiveness of the proposed system, let us consider the following problem as shown in Table I. This problem consists of three inter-university courses (Course1, Course2, and Course3) opened for five universities (A, B, C, D, and E), and 10 applicants (S1, S2, ..., S10) who are students at the universities. Each university has two, three, two, two, and one applicants, e.g., the students S1 and S2 belong to A University.

We now assume that the numbers of enrollments for respective courses ( $N_1, N_2$ , and  $N_3$  in (1)), which are restricted to reduce teachers' loads, are set to 5, 7, and 5 as shown in parentheses at the rightmost column. We also assume that Course1 and Course3 are compulsory for the students at universities A and E, respectively; in the other universities, the three courses are optional.

The number at the rows of Course1, Course2, and Course3 corresponds to the value of  $f_{ij}$  in (1). Here, we set it to a high value for a compulsory course and a low value for an optional course. For example, the  $f_{ij}$  values of the students S1 and S2 for Course1 and that of S10 for Course3 were set to three. On the other hand, when a student applied for an optional course, we set the value of the corresponding  $f_{ij}$  to one, otherwise, zero. For example, since the student S4 applied for only Course2, the  $f_{ij}$  value for Course1, Course2, and Course3 were set to zero, one, and zero, respectively.

The number of applicants for each course is described as the values without parentheses the rightmost column and can be calculated as the total number of non-zero values in each row. The values at the bottom row represent the upper limit of courses that each student can take, i.e., they correspond to the value of  $b_i$  in (1). Although the value for a student was calculated as the total number of courses that the student applied for, we restricted its maximum to two. This is because we need to avoid the results that a student cannot take any course. For example, the value for the student S3 is two in spite of the fact that the student applied for the three courses.

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

Table II shows the drawing results using the aforementioned solver (*intlinprog* function in MATLAB). The value at the columns of S1, S2, ..., and S10 corresponds to that of  $x_{ij}$ .

We obtained the results that the students S1, S2, and S10 can take their compulsory course, and the other students can also take some of courses they applied for. In addition, the number of students for each course was within the upper limit of enrollments ( $N_1, N_2$ , and  $N_3$ ), and the number of courses for each student was also within its upper limit ( $b_i$ ) described in the bottom row of Table I. Therefore, the proposed system lead to the good results for the students such that the two conditions in (1) were satisfied.

V. CONCLUSION

In this paper, we showed that a problem to draw students for courses with a limited number of enrollments can be formulated as that of mixed-integer linear programming. We also proposed a flexible drawing system in which course types and students' preference were taken into account as the system parameters. Our experimental results demonstrated that the proposed system works well so that students' preference and prescribed conditions can be satisfied.

REFERENCES

- [1] T. Inaba, "An Attempt for Improvement in Mass Lecture at KOBE University –An Algorithm for Changing Class-room by Lot Which Make Maximizing Student's Request–," *Bulletin of the Faculty of Human Development, Kobe University*, vol. 13, pp. 41–46, March 2006 (in Japanese).
- [2] H. Murai, K. Ura, N. Suehiro, R. Yaegashi, Y. Imai, K. Saisho, and T. Hayashi, "A Study of Load-Balancing Strategy Based on Students' Action on University Cooperative e-Learning", in *Proc. 16th IEEE/ACIS Int. Conf. Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, 2015, pp.587–590. <http://dx.doi.org/10.1109/snmpd.2015.7176265>
- [3] Cooperative University e-Learning Project Based on e-Knowledge for the Shikoku Area (in Japanese), Available: <http://chipla-e.itc.kagawa-u.ac.jp/index.html>
- [4] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica*, vol.28, pp. 497–520, July 1960. <http://dx.doi.org/10.2307/1910129>

TABLE I: EXAMPLE TO BE SOLVED.

Univ. and student Course	A University		B University			C University		D University		E University	No. of applicants
	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	
Course1	3	3	1	0	1	1	1	0	1	0	7 (5)
Course2	1	0	1	1	1	1	0	1	0	1	7 (7)
Course3	0	1	1	0	1	0	1	1	1	3	7 (5)
Upper limit of reg isterable courses	2	2	2	1	2	2	2	2	2	2	

TABLE II: EXPERIMENTAL RESULTS WITH THE PROPOSED DRAWING SYSTEM.

Univ. and Student Course	A University		B University			C University		D University		E University	No. of elected students
	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	
Course1	1	1	1	0	1	0	0	0	1	0	5
Course2	1	0	1	1	1	1	0	1	0	1	7
Course3	0	1	0	0	0	0	1	1	1	1	5