Combinatorial Optimization for University Examination Proctor Assignments

Takeshi Koide, *Member, IAENG*

***Abstract*—Our university conducts final examinations at the end of each semester. University personnel is assigned as a proctor to invigilate. The task of assigning proctors is currently performed manually by the university's registrar and it usually requires a few days. In a previous study, a mixed integer programming model was proposed to derive an optimal proctor assignment scheme with satisfying some requirements. The results were acceptable but the model became outdated as a result of changes in the administration of the examinations. This study revised the previous model to derive a more appropriate scheme. The revised model was systemized as a system based on Microsoft Excel and tested by numerical experiments.**

***Index Terms*—combinatorial optimization, examination proctor assignment, mixed integer programming, optimization in university**

I. Introduction

FINAL examinations are administered at the end of each semester in classrooms more spacious than used for regular courses in our university. Faculty and staff members are assigned as examination proctors to invigilate. The assignments are determined manually by the university's registrar, which requires a few days to construct the proctor assignment plan. Manual assigning is troublesome; therefore, an automated system has been desired to replace it.

We previously proposed a mixed integer programming model and constructed a prototype system to derive optimal assignments [1]. The constructed optimization problem was solved using a commercial optimization program.

Proctor assignment is one of the types of timetabling problems, which include course assignments and classroom assignments. Various models and approaches have been proposed for a variety of situations [2–7]. A contest was also held for participants to find efficient algorithms for timetabling problems [8].

Two years ago, our university changed the duration and scheduling of its final examinations, as well as the lengths of interval periods between examinations. Consequently, the constructed system became outdated and was expected to be revised by the registrar.

In this paper, we revised our previous model and constructed a system based on the use of electronic spreadsheets to derive the optimal proctor assignments. Numerical experiments were conducted on actual data to test the model.

II. Preliminaries

A. Target Problem

The target problem is to find the optimal proctor assignments for the final examinations held by our university. Before the assignments are begun by the registrar, faculty members are asked for the favorable type of classrooms for their examinations. Then, classrooms are assigned for the examinations. The classroom assignments are also time-consuming; therefore, in another study, we formulated a mathematical programming approach for this particular task [9].

The required number of proctors for each examination is determined by the number of examinees. A proctor is assigned per forty examinees. When the number of proctors required for an examination is even, equal numbers of faculty and staff members are assigned. When the number is odd, one more faculty member is assigned. The lecturer of a course is necessarily assigned as the chief proctor of that course's examination.

A proctor's task is physically and mentally tiring; therefore, the number of invigilating should be equalized among faculty members and among staff members. Additionally, being assigned to multiple examinations in a single day is not favored by most personnel, especially for 2 consecutive examinations in one day or 2 examinations separated by more than 2 interval periods.

For most university departments, assigning multiple personnel during the same period on the same day is disruptive. Hence, for each department, a maximum number of proctors who can invigilate simultaneously must be determined. Some staff members work at other campuses but must travel to the main campus for proctoring. Thus, they are limited to 2 days per week for proctoring.

There are other conditions to be considered for proctor assignments: (1) both one female and one male proctor are required to monitor the examinees' use of the bathroom; (2) some classrooms are rather spacious and the task of proctoring is relatively more difficult so the number of proctors assigned to such rooms is restricted to one; and (3) personnel unfamiliar with proctoring must be assigned together with experienced personnel.

Our previous model [1] considered the duration of the examinations but the modified model in this paper does not because the changes in the administration of the examinations have rendered the durations irrelevant.

This work was partly supported by MEXT, Japan.

Takeshi Koide is with the Department of Intelligence and Informatics, Konan University, Kobe 658-8501, Japan (corresponding author to provide telephone and fax no.: +81-78-435-2532; e-mail: koide@konan-u.ac.jp).

B. Notations

Sets:

- Mem — set of university personnel
- Mem_f — set of faculty members, a subset of Mem
- Mem_s — set of staff members, a subset of Mem
- Mem_o — set of members working at other campuses, a subset of Mem
- Mem_k — set of members belonging to department k, a subset of Mem
- Exa — set of examinations
- Dep — set of departments
- Day — set of days allotted to administering final examinations
- Per — set of periods, {1, 2, 3, 4, 5, 6}

Constants and Parameters:

- $Sc_{e,d,p}$ — 1 if examination e is conducted on day d in period p; otherwise, 0.
- $Sc_{e,d}$ — 1 if examination e is conducted on day d; otherwise, 0.
- $Ava_{m,e}$ — 1 if member m can be assigned to a proctor for examination e; otherwise, 0.
- $Ch_{m,e}$ — 1 if member m is the chief proctor for examination e; otherwise, 0.
- $R_{e,type}$ — the required number of faculty members (*type* = f) and staff members (*type* = s) for proctoring examination e
- N^-_m, N^+_m — lower and upper bounds of the number of proctor assignments for member m
- $N^+_{m,d}$ — upper bounds of the number of proctor assignments for member m on day d
- $N^+_{k,d,p}$ — upper bounds of the number of proctor assignments for members belonging to department k on day d in period p
- $Se_{m,type}$ — 1 if member m is *type*; otherwise, 0 where *type* = f (female) or m (male).
- Sp_e — 1 if examination e is conducted in spacious rooms; otherwise, 0.
- E_m — 1 or −1 if member m is experienced at or new to proctoring; otherwise, 0.
- α_i — positive weight for the terms in the objective function (i =1, 2, 3)

Design variables:

- $x_{m,e}$ — 1 if member m is assigned to examination e; otherwise, 0.
- $y_{m,d,p}$ — 1 if member m is assigned to an examination conducted on day d in period p; otherwise, 0.
- $z_{m,d}$ — 1 if member m is assigned to an examination conducted on day d; otherwise, 0.
- s^-_m, s^+_m — slack variables for lower and upper bounds of number of assignments for member m
- $s^+_{m,d}$ — slack variables for upper bound of number of assignments for member m on day d
- $s^+_{k,d,p}$ — slack variables for upper bound of number of assignments for members belonging to department k on day d in period p

III. FORMULATIONS

A mixed integer programming model for the examination proctor assignment (problem EPA) is formulated here.

Problem EPA:

Minimize $\alpha_1\phi_1 + \alpha_2\phi_2 + \alpha_3\phi_3$ (1)

subject to

$$\phi_1 = \sum_{m \in Mem}\left(s^-_m + s^+_m\right) \quad (2)$$

$$\phi_2 = \sum_{m \in Mem}\sum_{d \in Day} s^+_{m,d} \quad (3)$$

$$\phi_3 = \sum_{k \in Dep}\sum_{d \in Day}\sum_{p \in Per} s^+_{k,d,p} \quad (4)$$

$$y_{m,d,p} = \sum_{e \in Exa} Sc_{e,d,p}\; x_{m,e}, \quad \forall m \in Mem, \forall d \in Day, \forall p \in Per \quad (5)$$

$$6z_{m,d} \geq \sum_{e \in Exa} Sc_{e,d}\; x_{m,e}, \forall m \in Mem, \forall d \in Day \quad (6)$$

$$x_{m,e} \leq Ava_{m,e}, \quad \forall m \in Mem, \forall e \in Exa \quad (7)$$

$$\sum_{m \in Mem} Ch_{m,e} x_{m,e} = 1, \quad \forall e \in Exa \quad (8)$$

$$\sum_{m \in Mem_{type}} x_{m,e} = R_{e,type}, \quad \forall e \in Exa, \forall type \in \{\mathrm{f}, \mathrm{s}\} \quad (9)$$

$$N^-_m + s^-_m \leq \sum_{e \in Exa} x_{m,e} \leq N^+_m + s^+_m, \quad \forall m \in Mem \quad (10)$$

$$\sum_{p \in Per} y_{m,d,p} \leq N^+_{m,d} + s^+_{m,d}, \forall m \in Mem, \forall d \in Day \quad (11)$$

$$y_{m,d,p} + y_{m,d,p+1} \leq 1, \quad \forall m \in Mem, \forall d \in Day, \forall p \in Per - \{6\} \quad (12)$$

$$y_{m,d,p} + y_{m,d,p+t} \leq 1, \text{ if } p \in Per \text{ and } p+t \in Per \quad \forall m \in Mem, \forall d \in Day, \forall t \geq 3 \quad (13)$$

$$\sum_{m \in Mem_k} y_{m,d,p} \leq N^+_{k,d,p} + s^+_{k,d,p}, \quad \forall k \in Dep, \forall d \in Day, \forall p \in Per \quad (14)$$

$$\sum_{m \in Mem} Se_{type} x_{m,e} \geq 1, \quad \forall e \in Exa, \forall type \in \{\mathrm{f}, \mathrm{m}\} \quad (15)$$

$$\sum_{e \in Exa} Sp_e x_{m,e} \leq 1, \quad \forall m \in Mem \quad (16)$$

$$\sum_{m \in Mem} E_m x_{m,e} \geq 0, \quad \forall e \in Exa \text{ if } R_{e,\mathrm{s}} \geq 1 \quad (17)$$

$$\sum_{d \in Date} z_{m,d} \leq 2, \quad \forall m \in Mem_{\mathrm{o}} \quad (18)$$

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

$$x_{m,e},\, y_{m,d,p},\, z_{m,d} \in \{0,1\},$$
$$\forall m \in Mem,\, \forall e \in Exa,\, \forall d \in Day,\, \forall p \in Per \qquad (19)$$

$$s_m^-,\, s_m^+,\, s_{m,d}^+,\, s_{k,d,p}^+ \in \{0,1\},$$
$$\forall m \in Mem,\, \forall k \in Dept,\, \forall d \in Day,\, \forall p \in Per \qquad (20)$$

The objective function (1) in problem EPA comprises a weighted sum of 3 terms given by (2–4). The 3 terms represent the sums of penalties for violating the hard constraints shown in (10), (11), and (14). The design variable $y_{m,d,p}$ in (5) is defined as a conditional sum of the main design variable $x_{m,e}$. Another design variable $z_{m,d}$ is determined by (6) and is introduced to express the constraints of the number of working days for proctoring simply. Constraint (7) adopts the proctor assignment to university personnel's schedules.

The lecturer of a course is necessarily assigned to its examination as the chief proctor. This condition is expressed by (8). The required number of proctors for each examination is assigned by (9). The respective numbers of required faculty and staff members are determined separately. The total number of assignments for each member is bounded by N^-_m and N^+_m. The hard constraint is relaxed into a soft one by the introduction of the slack variables s^-_m and s^+_m in (10).

The number of assignments in a day is also restricted by (11) and the constraint is also relaxed by the slack variables. Assignments for 2 consecutive examinations and 2 examination periods separated by more than 2 interval periods are prohibited, as represented by (12) and (13), respectively. The number of staff members of a certain department with simultaneous assignments is limited and the constraint is relaxed as shown in (14). Constraints (15) and (16) express the conditions for female and male proctor assignments, as well as for assignments to examinations conducted in spacious rooms, respectively. Constraint (17) prohibits the assignments of more inexperienced proctors than experienced except when only a chief proctor is assigned. Constraint (18) limits the number of days of proctoring for staff members working at other campuses to 2 days. The remaining constraints, (19) and (20), are {0,1} conditions for the design variables.

IV. NUMERICAL EXPERIMENTS

The optimization problem represented by the EPA model was implemented in a Microsoft Excel workbook. Clicking a button on the workbook executes VBA macros which calls a program written in Python to construct the target optimization problem, which was solved by Gurobi Optimizer [10], an optimization solver. This section describes an example of an optimization using data from the final examination sessions for the fall semester of 2018.

Three types of information were entered into the system: examination schedules, examination information, and university personnel's information. There were 6 days of examinations: January 23–29, 2019. There were 6 periods per day. The examination information for each examination contained the date, period, number of examinees, name of the chief proctor, assigned classroom, and the university department administering the examination. The university personnel's information contained the ID number, affiliation, years of employment, schedule, and target number of assignments. The example data had 167 staff members, 220 faculty members, and 909 examinations.

TABLE I
OPTIMIZATION RESULTS

Case	Parameters			Objective Values			CPU Time (s)
	α_1	α_2	α_3	ϕ_1	ϕ_2	ϕ_3	
(a)	1	0	0	1	716	544	6.88
(b)	0	1	0	122	4	544	1.28
(c)	0	0	1	160	722	3	1.26
(d)	1	1	0	1	4	544	7.94
(e)	1	0	1	11	716	15	9.63
(f)	0	1	1	133	4	3	3.24
(g)	1	1	1	11	4	14	12.04
(h)	10	1	1	1	4	26	11.81

The EPA problem was solved by implementing the modified model with several settings of the parameters on a computer with an Intel Core i7 CPU and 16 GB of memory running on Microsoft Windows 10. The results are summarized in Table 1.

Cases (a)–(c) show that the minimum values of ϕ_1, ϕ_2, and ϕ_3 are 1, 4, and 3, respectively. These results mean that constraints (10), (11), and (14) cannot be satisfied without a soft relaxation for the target data. Cases (d) and (f) show that 2 of the 3 functions (ϕ_1, ϕ_2, ϕ_3) are optimized, whereas, in case (e), none are optimized. Case (h) optimizes both ϕ_1 and ϕ_2, then derives a relatively small value for ϕ_3. Computational time was more influenced by emphasized ϕ_1 than by the other functions. The final results were acceptable to the registrar with regard to the quality of the assignments and required computational time.

V. CONCLUSION

This study proposed a mixed integer programming model for solving an examination proctor assignment problem and deriving a proctor assignment plan for our university. The quality of the derived assignments was basically acceptable for practical situations.

The model can be modified with different values and different conditions. More numerical experiments with different types of data are required to further assess the model's performance.

REFERENCES

[1] T. Koide, "Mixed integer programming approach on examination proctor assignment problem," *Procedia Comput. Sci.*, vol. 60, pp. 818–823, 2015.
[2] S. Abdullah and H. Turabieh, "On the use of multi neighbourhood structures within a tabu-based memetic approach to university timetabling problems," *Inf. Sci.*, vol. 191, pp. 146–168, 2012.
[3] G. N. Beligiannis, C. N. Moschopoulos, G. P. Kaperonis, and S. D. Likothanassis, "Applying evolutionary computation to the school timetabling problem: The Greek case," *Comput. & Oper. Res.*, vol. 35, pp. 1265–1280, 2008.
[4] T. Thepphakorn, P. Pongcharoen, and C. Hicks, "An ant colony based timetabling tool," *Int. J. Production Economics*, vol. 149, pp. 131–144, 2014.
[5] C.W. Fong, H. Asmuni, B. McCollum, P. McMullan, and S. A. Omatu, "A new hybrid imperialist swarm-based optimization

algorithm for university timetabling problems," *Inf. Sci.*, vol. 283, pp. 1–21, 2014.
[6] S. Daskalaki and T. Birbas, T. "Efficient solutions for a university timetabling problem through integer programming," *European J. Oper. Res.*, vol. 160, pp. 106–120, 2005.
[7] O. Onouchi, T. Uchigaito, and M. Sasaki, M. "Examination timetabling problem in universities" (in Japanese), in *Abstracts book of ORSJ spring meeting 2013*, pp. 66–67.
[8] International Timetabling Competition 2011. http://www.utwente.nl/ctit/hstt/itc2011/welcome/.
[9] T. Koide and Y. Naba, "A classroom assignment problem in university term examinations," in *Proc. 17th Asia Pacific Industrial Engineering and Management System Conf.*, *APIEMS2016*, Taipei, Taiwan, 5 pages.
[10] Gurobi Optimization. http://www.gurobi.com/.