A Heuristic Approach to Grouping and Timetabling of Project Presentations of Student Teams

Can Akkan*, Burçin Bozkaya

Sabancı University,
Faculty of Management,
Orhanlı, Tuzla, İstanbul 34956, Turkey

June 2008

Keywords: timetabling; assignment; diversity; heuristic

1 Introduction

A key requirement of the *Introduction to Management* course at Sabancı University, is the preparation of a business plan. About 40 teams present their business plans. Teams present in groups of 3 or 4, and all members of the teams in a group are required to be in the audience when other teams of their group present. Teams of judges used for different groups may change. MGMT 201 is taken students from different colleges and classes (sophomore, junior, etc).

Three key goals defined the need for the current work. Firstly, as judges inevitably grade presentations relative to each other, we would like to have each group of teams to reflect the academic diversity (college, GPA, class) present in the course. The second goal is to have teams with the same type of business plans (e.g. retail, manufacturing, or services) assigned to the same group. The third goal is related to the timing of the presentations. It is not desirable to schedule all the presentations after 6:30 PM when students do not have any classes. This creates a timetabling problem where only certain day time–slots are feasible for certain teams.

2 MIP Formulation of the Problem

We model characteristics of teams with binary attribute matrices. Three business plan types are used: retail/wholesale trade, pure service firms and non–service production. Then, for

*corresponding author:canakkan@sabanciuniv.edu, ph:+90-216-4839685, fx:+90-216-4839699

each team three academic attributes are defined as binary variables. The first academic attribute is GPA. If a team's median GPA is less than 2.5 it is designated as 'low GPA', otherwise 'high GPA'. The second attribute is college. If the number of engineering students is more than half of the team size we refer to it as an 'engineer–dominated' team. The third attribute is class (or year). We categorize teams based on whether more than half of the students in a team are freshman or sophomore, or not. The notation we use is as follows:
Data:

$\mathcal{T}_i$	:	set of feasible time–slot indices for all members of Team i, $i = 1, \ldots, I$
a_{ik}	:	1 if Team i exhibits academic attribute k, 0 otherwise, $k = 1, \ldots, K$
p_{il}	:	1 if Team i has business plan type l, 0 otherwise, $l = 1, \ldots, L$
n_{min}	:	lower limit on the number of teams per group
n_{max}	:	upper limit on the number of teams per group

Decision Variables:

$$X_{it} = 1 \text{ if team } i \text{ is assigned to time–slot } t, 0 \text{ otherwise}$$
$$Z_t = 1 \text{ if time–slot } t \text{ is used, } 0 \text{ otherwise}$$

We use the Total Absolute Deviation (TAD) measure of Mingers and O'Brien to quantify difference between groups for two reasons (Mingers and O'Brien, 1995). Firstly, it can be formulated as a linear program and secondly, compared to a large selection of alternative objective functions it is reported to perform quite well (see (Baker and Powell, 2002)).

Let,$C_{tk} = \sum_i a_{ik} X_{it}$ and Q_k denote the average number of teams per time–slot exhibiting attribute k, ($Q_k = \sum_i a_{ik}/B$, where $B = \sum_t Z_t$), we measure the academic diversity of a particular time–slot t by

$$D_t = \sum_k |C_{tk} - Z_t Q_k| \tag{1}$$

Since, $\lceil I/n_{max} \rceil \leq B \leq \lceil I/n_{min} \rceil$ and B makes the objective function term (1) non–linear, we assume number of time–slots used is a constant, denoted by b, and use the term $q_k = \sum_i a_{ik}/b$ in the model. Given b, (1) can be linearized as well (Mingers and O'Brien, 1995):

$$C_{tk} + S_{tk}^- - S_{tk}^+ = Z_t q_k \tag{2}$$

Similarity of business plans within a time–slot is measured somewhat differently. Let,

$$C'_{lt} = \sum_i p_{il} X_{it} \tag{3}$$

so that, C'_{lt} denotes the number of teams in time–slot t that have business plan type l. Then, given $N_t = \sum_i X_{it}$, we measure the homogeniety of business plan types in a time–slot by

$$H_t = \sum_l (N_t - C'_{lt}) P_{lt} \tag{4}$$

where, P_{lt} is a decision variable defined as follows:

$$P_{lt} = \begin{cases} 1 & \text{if business plan type } l \text{ is the primary (target) plan type for time–slot } t \\ 0 & \text{otherwise} \end{cases}$$

Expression (4) is linearized as follows.

$$C'_{lt} + R_{lt} = N_t \ \forall l, t \tag{5}$$

$$R_{lt} \leq \bar{R}_t + n_{max}(1 - P_{lt}) \ \forall l, t \tag{6}$$

$$\sum_l P_{lt} = 1 \ \forall t \tag{7}$$

By minimizing $\sum_t \bar{R}_t$ the model sets the business with the largest number of teams as the primary plan of that time–slot, so that $\bar{R}_t$ becomes the number of team which are not the primary–plan teams in that time–slot. Constraints (6) ensure that exactly one plan type is selected as the "target" for each time–slot.

The third goal is modeled by assigning a weight, w_t, to each time–slot t and incorporating w_t in the objective function as follows:

$$Z = \sum_t w_t\left(\sum_k (S^-_{tk} + S^+_{tk}) + \bar{R}_t\right) \tag{8}$$

Thus the mixed integer linear programming formulation of the problem is as follows:

$$\min Z$$

subject to

$$\sum_{t \in T_i} X_{it} = 1 \ \ \forall i \tag{9}$$

$$\sum_i X_{it} \leq n_{max} Z_t \ \ \forall t \tag{10}$$

$$\sum_i X_{it} \geq n_{min} Z_t \ \ \forall t \tag{11}$$

$$X_{it} = 0 \ \ \forall i, t \notin T_i \tag{12}$$

$$N_t = \sum_i X_{it} \ \ \forall t \tag{13}$$

$$C_{tk} = \sum_i a_{ik} X_{it} \ \ \forall t, k \tag{14}$$

$$C'_{lt} = \sum_i p_{il} X_{it} \ \ \forall t, l \tag{15}$$

$$\sum_l P_{lt} = Z_t \ \ \forall t \tag{16}$$

$$\sum_t Z_t = b \tag{17}$$

$$C'_{lt} + R_{lt} = N_t \ \ \forall l, t \tag{18}$$

$$R_{lt} \leq \bar{R}_t + n_{max}(1 - P_{lt}) \ \ \forall l, t \tag{19}$$

$$
\begin{aligned}
C_{tk} + S^-_{tk} - S^+_{tk} &= Z_t q_k \;\; \forall k, t \qquad (20)\\
S^-_{tk}, S^+_{tk} \in \mathbf{R}^+ \;\; \forall t, k\\
R_{lt} \in \mathbf{R}^+ \;\; \forall l, t\\
\bar{R}_t \in \mathbf{R}^+ \;\; \forall t
\end{aligned}
$$

3 A Meta-Heuristic Approach

Our heuristic methodology is based on Tabu Search principles. At the first stage of the algorithm, we construct a starting solution x_0 by first identifying b time-slots and an anchor team for each time slot. We randomly select an unassigned team g and make it the anchor of an available time-slot. We then iteratively assign remaining teams to one of the b time-slots in a cheapest insertion manner.

Once a starting solution is constructed, we attempt to improve it for a pre-defined number of iterations. We use three neighborhood definitions N_1, N_2 and N_3. At each iteration, we consider every member of these neighborhoods as a candidate move, and evaluate the incremental cost of each move. We allow non-improving moves in order to avoid local optima, and we maintain two tabu lists to complement the search. The first tabu list is for N_1 and N_2, and keeps a history of moves at the team level. The second tabu list is for N_3, and keeps a history of moves at the time-slot level. Tabu tenures for both lists are randomly drawn from a range of values.

The TABUCORE algorithm stops when either a) total of S iterations are executed, or b) total of S_{noimp} iterations are executed without improvement in objective function. The TABUCORE algorithm can be used to generate good solutions relatively fast. We further improve the heuristic by implementing Probabilistic Diversification and Intensification (PDI) methodology (Rochat and Taillard (1995)). The idea is to continuously maintain a pool of good solution components, and use these components to create hopefully better solutions.

In our PDI implementation, TABU-PDI, we run the TABUCORE algorithm R_1 times to generate an initial set of solutions, which we use to populate a set $\mathcal{T}$, the pool of assigned time-slots. Next, the PDI algorithm is run for a pre-defined number of iterations, R_2, and any time-slot assignments belonging to better solutions found along the way replace inferior ones in the pool. The best solution found is reported at the end.

4 Computational Experiments

We generated a random data set with $I = 40$, $n_{max} = 4$, $n_{min} = 3$, $K = 4$, $L = 3$. w_t for each day time–slot was 1, and the weights for half of the evening time–slots was 1.25 and the other half was 2.

The number of teams available at a day time–slot was generated using a frequency distribution F=$\{(f, p)\}$, where f is the fraction of teams available at any given day time–slot and p is the corresponding probability. Two distributions were used, F_1 and F_2 with $E(F_1) = 0.345$ and $E(F_2) = 0.533$.

Results for different numbers of day time-slots (D) and evening time–slots (E) are given in Table 1. For each setting, 20 random instances were generated. All the instances were solved by CPLEX mixed-integer programming solver with OPL Studio 5.5. Each instance was solved for possible values of total number of time-slots used, b (10 through 13). CPLEX was given a time–limit of 3600 seconds for each b. These preliminary results show that as $E(F)$ increases CPLEX run-times increase significantly, resulting in more failures in finding optimum solutions. Tests of the tabu search algorithm are in progress.

	F_1				F_2			
	Run-time (sec.)				Run-time (sec.)			
(D,E)	Min	Ave	Max	Nbr Opt	Min	Ave	Max	Nbr Opt
(15,8)	37.9	2032.2	7062.3	15	15.9	7015.1	11621.6	3
(17,6)	45.3	1125.1	7724.2	19	52.0	6605.7	14402.3	3
(15,6)	45.9	888.7	4367.7	19	15.1	4799.8	11486.2	10
(17,4)	14.3	542.5	3786.6	20	14.2	5698.5	14405.9	9
(13,6)	18.2	296.9	1496.9	20	18.7	2552.5	8527.9	13
(15,4)	9.9	334.4	1071.5	20	14.9	4040.9	10802.7	12

Table 1: Performance of CPLEX

References

K. R. Baker and S. G. Powell. Methods for assigning students to groups: a study of alternative objective functions. *Journal of the Operational Research Society*, 53:397–404, 2002.

J. Mingers and F. A. O'Brien. Creating student groups with similar characteristics: a heuristic approach. *Omega*, 23:313–321, 1995.

Y. Rochat and E.D Taillard. Probabilistic diversification and intensification in local search for vehicle routing. *Journal of Heuristics*, 1:147–167, 1995.