ALLOCATING CONFERENCE DELEGATES TO WORKSHOPS: A SPECIAL TIMETABLING PROBLEM

M. SINCLAIR and D.G. ESTERHUYSEN

Department of Industrial Engineering
University of Stellenbosch
Private Bag X5018
7600 STELLENBOSCH

(Refereeing process supervised by A. Pachyannis)

ABSTRACT

The problem of assigning delegates to workshops at a conference can be formulated as a timetabling problem. Such an assignment must take into account the preferences of the delegates, as well as the number of participants each workshop can accommodate. This paper will report on a heuristic solution technique for a special case of such a problem. An implementation of the method for the solution of a real world problem will also be reported.

1. INTRODUCTION

Many examples of timetabling problems and solution techniques for such problems have been reported in the Operations Research literature. Such problems may be the assignment of students to teachers and time windows (De Werra [3,4], De Werra, et al [5], Schreuder [7]), the allocation of classrooms to classes and teachers (Mulvey [10]), examination scheduling (Romero [1], Carter [2], De Werra [3], Anderson and Bernhard [6]) or combinations of the above (De Gans [8], Junginger [9], Tripathy [11,12], Ferland and Roy [13], Phala

[14]). The solution methods proposed for these problems vary from graph theoretical methods (Carter [2], De Werra [3,4]) through mathematical programming techniques (Anderson and Bernhard [6], Mulvey [10], Tripathy [11,12], Ferland and Roy [13]) to heuristic methods (Romero [1], Schreuder [7], De Gans [8], Junginger [9], Phala [14]). One problem that has not received much attention in the literature is the assignment of delegates to workshops at a conference. This paper will report on such a problem, a solution technique for the problem and the performance of this solution technique on a real world example of the problem.

The next section will describe the problem and the practical situation from which this study was drawn. Section three will show a mathematical programming model for the problem. A heuristic solution technique for the problem will be discussed in section four. Section five will be devoted to details of the programming of the solution method and its performance in solving the real world problem. The final section will draw a few conclusions.

2. FORMAL DESCRIPTION OF THE PROBLEM

Suppose some delegates to a conference are interested in attending a number of workshops which are held on various days of the conference. Some of these workshops are repeated on different days. Obviously each delegate would prefer to attend particular workshops on specific days. The goal is to assign delegates to their preferred workshops on the days which suit them best. The difficulty is that the resources available make unrestricted attendance at each workshop impossible.

This problem came to our attention when the Engineering Students Representative Council tried to organize the 1990 Engineering Winter School at the University of Stellenbosch. During this occasion, high school students interested in a

career in engineering attend various workshops organized by companies in the Western Cape. The problem is that the enrolment for a specific workshop may be higher than the number of students the company can handle simultaneously. This gives rise to the problem formalized in the following section. An extra complication is that every student must attend a workshop on each of four days, but may not repeat his/her visit to the same company on different days.

In order to assign students to workshops in an equitable manner, questionnaires were sent to them beforehand, requesting them to indicate their preferences for the different workshops on the different days. In this case, a low number assigned to a workshop represented a high preference for the student. The only other data available was the number of students each company could accommodate on each day.

3. MATHEMATICAL MODEL FOR THE PROBLEM

We use the following notation:

s = index indicating student.
c = index indicating company.
d = index indicating day.

DATA:

N_{cd} = number of students company c can accommodate on day d.

A_{scd} = positive integer, indicating the priority given by student s to company c on day d; for example A_{scd} = 1 if company c is the first choice of student s on day d.

VARIABLE:

x_{scd} = 1 if student s visits company c on day d.
      0 otherwise.

The mathematical programming model can thus be formulated as follows:

$$\text{Minimize } \sum_s \sum_c \sum_d A_{scd}\, x_{scd}$$

Subject to:

1. Each student must visit one and only one company on each day:

$$\sum_c x_{scd} = 1, \text{ for all s and d.}$$

2. The number of students visiting each company on each day may not exceed the number the company can accommodate on that day:

$$\sum_s x_{scd} \le N_{cd}, \text{ for all c and d.}$$

3. No student may visit the same company more than once:

$$\sum_d x_{scd} \le 1, \text{ for all s and c.}$$

4. Binary constraints:

$$x_{scd} = 0 \text{ or } 1, \text{ for all s, c and d.}$$

This is a so-called planar three-dimensional transportation model, which falls into the class of NP-hard problems (Gilbert, et al [16]). Given the size of the problem we faced (4800 binary variables), it was decided to develop a

heuristic solution method for the problem, rather than try to solve it optimally. Such a heuristic solution method will be discussed in the next section.

4. HEURISTIC SOLUTION METHOD

In the development of the heuristic algorithm, extensive use was made of a heuristic developed in Sinclair [15]. The latter heuristic is similar to Vogel's method for transportation problems (see any good book on linear programming). Since the steps of the algorithm are easy to follow, we shall not make any attempt at motivating these steps any further.

ALGORITHM:

1. Arrange the data in the form of a matrix as follows:
    (a) Let the rows represent students, with index s.
    (b) Let the columns represent companies on each day, i.e. with indices (c,d).
    (c) Let the elements in each row s and column (c,d) be A_{scd}.
    (d) Add one row to the bottom of the matrix, with the elements N_{cd}.

2. For each row representing a student, calculate the difference between the best and the second-best preference in the row. Denote these differences by $diff_s$.

3. Identify all rows with $diff_s = \max_s \{diff_s\}$.

Choose one of these rows at random. Denote the index of the chosen row by s^*.

4. Identify the day d^* and company c^* corresponding to the best preference of student s^*. Assign student s^* to his/her best preference.

5. Let A_{s*cd*} = M (a large positive number) for all the companies c accommodating student s^* on day d^*. Also let A_{s*c*d} = M for all days d on which company c^* accommodate student s^*. Let $N_{c*d*} = N_{c*d*} - 1$.

6. Remove all rows s for which all matrix elements are equal to M, and all columns for which $N_{cd} = 0$ from the matrix.

7. Repeat steps 2 to 6 until all students have been assigned to a company on each of the days.

Notice that constraints 1 and 3 of the mathematical model are enforced by the changes made to the matrix elements in steps 5 and 6 of the algorithm. Constraint 2 is enforced by the changes made to N_{cd} in step 5. The choice of assignment in steps 2-4 of the algorithm has as its aim the minimization of the objective function of the mathematical model.

5. SOLUTION OF THE REAL PROBLEM

Two hundred and forty students enrolled for the 1990 Engineering Winter School held at the University of Stellenbosch. The workshops were attended on four days. Ten companies in the Western Cape were prepared to accommodate students in workshops, but only 5 were visited each day.

To do the allocation and scheduling of students by hand turned out to be impossible. The algorithm of the previous section was coded in Turbo PASCAL 4, and the problem solved on an IBM compatible microcomputer. The solution obtained was satisfactory, and was implemented without any problem.

The solution time for the real problem was about 3 minutes on an ordinary IBM compatible XT microcomputer. The input to and output from the program can be improved, but since the user

of the program was also the programmer thereof, there was no need to spend more time on this aspect!

6. CONCLUSIONS

A very difficult practical problem was solved with a simple but effective algorithm. The solution was implemented and proved to be acceptable.

The approach can be used for a variety of other scheduling problems. Extra conditions, such as that no student may attend a certain combination of companies, can be included in the problem, without substantially increasing the complexity of the heuristic.

The following extension to the algorithm could improve the quality of the final solution: In step 3, instead of choosing a row at random from the set with the maximum value of $diff_s$, choose from the subset with the maximum second-best preference. This variation in the algorithm is still to be investigated.

REFERENCES

[1]   ROMERO, B P: Examination scheduling in a large engineering school: A computer-assisted participative procedure. Interfaces 12(2) (1982), 17 - 23.

[2]   CARTER, M W: A survey of practical applications of examination timetabling algorithms. Operations Research 34 (1986), 193 - 202.

[3]   DE WERRA, D: An introduction to timetabling. European Journal of Operational Research 19 (1985), 151 - 162.

[4] DE WERRA, D: Graphs, hypergraphs and timetabling. Methods of Operations Research 49 (1985), 201 - 213.

[5] DE WERRA, D ; PASCHE, C and PETTER, A: Time-tabling problems: Should they be canonical? INFOR 24 (1986), 304 - 308.

[6] ANDERSON, J M and BERNHARD, R H: A university examination-scheduling model to minimize multiple examination days for students. Decision Sciences 12 (1981), 231 - 239.

[7] SCHREUDER, J A M: Timetables in Dutch high schools. In Operational Research '84 (J P Brans, ed.), Elsevier, Amsterdam (1984), 601 - 612.

[8] DE GANS, O B: A computer timetabling system for secondary schools in the Netherlands. European Journal of Operational Research 7 (1981), 175 - 182.

[9] JUNGINGER, W: Timetabling in Germany - A survey. Interfaces 16(4) (1986), 66 - 74.

[10] MULVEY, J M: A classroom/time assignment model. European Journal of Operational Research 9 (1982), 64 - 70.

[11] TRIPATHY, A: A Lagrangean relaxation approach to course timetabling. Journal of the Operational Research Society 31 (1980), 599 - 603.

[12] TRIPATHY, A: School timetabling - A case in large binary integer linear programming. Management Science 30 (1984), 1473 - 1489.

[13] FERLAND, J A and ROY, S: Timetabling problem for university as assignment of activities to resources. Computers and Operations Research 12 (1985), 207 - 218.

[14] PHALA, J M: A university course timetabling problem. ORiON 4(2) (1988), 92 - 102.

http://orion.journals.ac.za/

[15] SINCLAIR, M: The assignment of workers to tasks - An example from an academic department. ORiON 1(2) (1985), 49 - 69.

[16] GILBERT, K C and HOFSTRA, R B: Multidimensional assignment problems. Decision Sciences 19(2) (1988), 306 - 321.