ORiON, Vol. 1, No. 2, pp. 49-69

THE ASSIGNMENT OF WORKERS TO TASKS - AN EXAMPLE FROM AN ACADEMIC DEPARTMENT

M SINCLAIR

Department of Applied Mathematics

University of Stellenbosch

7600 STELLENBOSCH

ABSTRACT

A description is given of the application of Operations Research techniques to the problem of assigning lecturers to tasks in an academic department. All the steps in the approach are described - from the collection of the data through the building of a mathematical model to the implementation of the solution. Two approaches to ob= taining a solution are described. Firstly a heuristic method is shown which can be done with pencil and paper. The solution obtained with this method can be used for implementation, or it can be used as a starting solution for the computer package used for the solution of the mathematical model.

1. INTRODUCTION

In any work situation the question of the division of labour is an important and sensitive one. The responsibility for the assignment of workers to jobs is carried by someone in a managerial position. This person (foreman, manager, etc) has the task of making sure that the division of labour is fair and that the workers are satisfied and motivated to do their jobs well.

In this paper one such situation is considered. We look at the situation in an academic department at a university, where the tasks (administration, lectures, etc) for the academic year are to be divided among the members of staff. This situation is similar to various others in the private and public sectors and thus the approach put forward here should be widely applicable. The problem is formulated as a multiple objective mixed integer programming problem. There is nothing new to such an approach to problems in an educational environ= ment. Examples of this are the allocation of space to different facilities in an academic department (Ritzman, Bradford and Jacobs [9]), the allocation of funds to activities (Lee, Van Horn and Brisch [6]) and the allocation of teachers to schools (Lee and Schniederjans [7]).

Various approaches to the problem addressed in this paper have been reported in the literature. Andrew and Collins [1] and Dyer and Mulvey [3] propose linear programming models with pure network struc= tures. Tillett [11], Shih and Sullivan [10] and Breslaw [2] suggest integer programming formulations for the problem. An approach similar to the one described here, can be found in Harwood and Lawless [4] and McClure and Wells [8]. In Harwood and Lawless [4] the number of courses given in a certain period is restricted, instead of the work load (measured in effective hours) as in this paper. McClure and Wells [8] generate a large number of possible work schedules and then use integer programming to find the best one. The most important difference between the present paper and those cited above is that a heuristic method which can be done with pencil and paper is described here, whereas all the other papers describe methods which need a com= puter for the solution of the mathematical model.

The model described in this paper is designed to ensure a good com= promise between a fair division of labour and the satisfaction of the preferences of each member of staff. In order to set up such a model some quantitative measure must be found for each lecturer's preferences, as well as the work load imposed by each task. The next section addresses this problem. The solution of the model is discussed at length. Firstly, a heuristic method is described that can be done with pencil and paper. Then we discuss the solution with the aid of a mathematical programming package on a computer. Finally attention is devoted to ways in which the solution can be implemented to ensure the highest possible degree of support from the staff.

2. OBTAINING DATA FOR THE MODEL

Consider an academic department for which the division of labour must be done for the next academic year. The head of the department is in a difficult position, since he has to take into account the following two goals, which are not completely reconcilable:

(i) Satisfy the preferences of the staff to the highest possible degree;
(ii) Ensure as fair a division of labour as possible.

It is obvious that these two goals are not completely reconcilable. For, if only (ii) is used, for example, it may happen that a lecturer is assigned to a task he would prefer not to do. Some division of labour must be devised to ensure the best possible reconciliation between these two goals. To do this some quantitative measures must be found for:

(1) Each lecturer's preference for each task.
(2) The work load imposed by each task.

Administrative tasks can be distributed evenly among all members of staff, since most of them can easily be broken up into small units. Therefore these tasks will not be considered in what follows. The only remaining issue concerning administrative tasks is that the nature of his job forces the head of the department to do more of them than the rest of the staff. In the Department of Applied Mathematics, University of Stellenbosch, (referred to as the "Department" in what follows) this is compensated for by assigning to the head of the department only two thirds of the normal lecturing load.

On the other hand, lecturing work cannot, for educational reasons, be broken up into units smaller than a prescribed minimum. It was found, for example, that too many lecturers assigned to parts of the same course during a semester tend to confuse the students. Therefore it was decided that the smallest unit into which a course would be divided, would be the equivalent of half a semester. This means that the fair division of labour as far as lecturing work is concerned, is much more difficult than was the case with administrative tasks. In what follows, we are thus going to concentrate on the assignment of lecturing tasks to lecturers. Ways of obtaining the measures referred to in (1) and (2) above for these tasks, are described separately.

2.1 Lecturers' Preferences

The only way to obtain lecturers' preferences is to ask them about it. A questionnaire used for this purpose in the Department is shown in Appendix A. To make provision for lecturers' preferences with respect to the distribution of their work load through the academic year, the questionnaire in Appendix B was also distributed.

2.2 Numerical Values for Course Loads

A specific formula is used in the Department to calculate the work load for the courses to be offered, as well as for the supervision of graduate students. The formula uses the expected number of students for the course and the number of lectures, tutorials, practical sessions, etc as input and gives so-called effective hours as output. The principles on which the formula is based, are given in Appendix C.

The data obtained thus can be used to calculate the total work load of each member of staff. Add the total effective hours needed for all courses and all graduate students to obtain a total of TE hours. Suppose the head of the department is assigned two thirds of the lecturing load of other lecturers and suppose one lecturer will be taking study leave for six months. Then the following equation must be solved to obtain the average load (AL) of each lecturer in a department with (say) 10 members of staff:

$$8 \text{ AL} + \frac{2}{3} \text{ AL} + \tfrac{1}{2} \text{ AL} = \text{TE}.$$

In the next section we discuss a method designed to assign a total load as close as possible to AL to each lecturer while trying to satisfy his preferences to the highest possible degree.

3. HEURISTIC FOR DIVISION OF LABOUR

To facilitate the division of labour, the academic year was divided into four quarters of equal length. These quarters do not necessarily coin= cide with the quarters on the official University calendar. A preferred load for each lecturer in each quarter can then be calculated from the answers to question 1 on the questionnaire in Appendix B.

Graduate work is assigned first. This is because Masters- and Doctoral students usually choose their own supervisors. Furthermore, graduate courses can usually be taught by only one lecturer. After the assign= ment of graduate work, each lecturer is left with part of his preferred load in each quarter, which must be used for undergraduate work. The assignment of graduate work is of course the responsibility of the head of the department, who should see to it that a balance is struck between the graduate- and undergraduate work done by each lecturer. The head of the department is also responsible for the assignment (or not) of special requests, such as those mentioned in question 2 of Appendix B.

In the remainder of this section a heuristic method will be described for the assignment of lecturers to the rest of the undergraduate courses. The heuristic was designed for ease of application and is related to Vogel's Approximation Method for the transportation problem (see any good book on linear programming). The main idea is that an assignment is chosen on the basis that if it is *not* made it will have the most undesirable con= sequence.

For each quarter a matrix such as the following is set up:

$$
\begin{array}{c|ccccccc|c}
 & L_1 & L_2 & \cdots & L_j & \cdots & L_n & \\
\hline
C_1 & a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1n} & LC_1 \\
C_2 & a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2n} & LC_2 \\
\cdot & \cdot & \cdot & & \cdot & & \cdot & \cdot \\
C_i & a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{in} & LC_i \\
\cdot & \cdot & \cdot & & \cdot & & \cdot & \cdot \\
C_m & a_{m1} & a_{m2} & \cdots & a_{mj} & \cdots & a_{mn} & LC_m \\
\hline
 & LL_1 & LL_2 & \cdots & LL_j & \cdots & LL_n &
\end{array}
$$

where L_j indicates lecturer j

C_i indicates course i

a_{ij} = the preference weight of lecturer j for course i

LC_i = work load of course i

LL_j = work load of lecturer j in this quarter (after subtraction of graduate work load).

If $\sum_{j=1}^{n} LL_j < \sum_{i=1}^{m} LC_i$, the last row is multiplied by $\sum_{i=1}^{m} LC_i / \sum_{j=1}^{n} LL_j$ to ensure that the total work load necessary for the quarter can be met by the work load available from the lecturers.

It is clearly possible that a certain lecturer (say j) may be unable to present a certain course (say i) because the load LC_i for that course exceeds the remaining preferred load LL_j of the lecturer by too wide a margin. The question is, by how much may LC_i exceed LL_j before we decide that lecturer j cannot be assigned to course i. No lecturer can expect never to have his preferred load exceeded. However, we would prefer this excess to be as small as possible. Suppose we decide that if

$$LC_i - LL_j > X, \qquad \text{..........} \qquad (A)$$

lecturer j will not be assigned to course i. The following problems arise:

(1) X is too large. Some lecturers will be dissatisfied because their preferred loads are exceeded by too big a margin.

(2) X is too small. In the resulting work schedule some lecturers may have to handle courses for which they had indicated a low preference (high a_{ij}). It may even be impossible to find any feasible work schedule.

In the Department it was found that values of $50 \leq X \leq 70$ give reasonable results. In the algorithm that follows, it is assumed that some cri= terium of the form (A) exists according to which it can be decided whether a certain lecturer can be assigned to a certain course or not. Another reason why a certain lecturer cannot be assigned to a certain course is if the lecturer has already been assigned a course which overlaps with this course on the time table.

Algorithm.

1. Set up the matrix for the quarter. Make sure that $\sum_{j=1}^{n} LL_j = \sum_{i=1}^{m} LC_i$.
2. Cross out a_{ij} if lecturer j cannot present course i.
3. Mark each remaining row of the matrix with (a ; b), where
   (i) a = difference between best preference in the row and the second best preference.
   (ii) b = second best preference in the row.
4. Find all the rows with the maximum value of a. Find among these rows the ones with the maximum value of b. Choose any of the latter rows and assign the course of that row to any lecturer with the best preference in the row. Suppose course i is thus assigned to lecturer j.
5. Delete row i. Decrease LL_j to $LL_j - LC_i$.
6. If all rows have been deleted, stop. Otherwise, go to 2.

This algorithm is repeated for each quarter. The residual of each lecturer's preferred load or the amount by which it is exceeded is carried over to the next quarter (except in the last quarter).

An example of the application of this algorithm to a small problem is shown in appendix D.

4. MATHEMATICAL PROGRAMMING APPROACH

The initial mathematical programming model used for the problem included the assignment of lecturers to courses for the whole academic year. The resulting integer programming problem was so huge that it was im=possible to solve within a reasonable time. When its solution was attempted by using the FMPS package on the SPERRY 1100 computer of the University of Stellenbosch, no solution better than the one obtained by hand with the algorithm in section 3 could be obtained after 3 hours of computing time. Therefore, it was decided to use the same approach as in section 3, that is, the assignment was done for one quarter at a time, carrying over the excess or residual load for each lecturer.

For each quarter we use the following notation:

j = index denoting lecturer

i = index denoting course to be done in present quarter

c_i = work load of course i in present quarter (obtained as described in section 2.2)

ℓ_j = work load which lecturer j would prefer in present quarter (obtained from question 1 in Appendix B)

a_{ij} = preference which lecturer j has indicated for course i (obtained from Appendix A).

w_j = total of all preferences which lecturer j has indicated for courses which he chose to lecture without taking other lecturers into account (obtained from question 3 on Appendix B)

x_{ij} = variable, with value 1 if lecturer j is assigned course i, and value 0 otherwise.

If we take into account the way in which the questionnaire in appendix A was structured, as well as goal (i) discussed in section 2, it is clear that each lecturer j, (j = 1, 2, ...) would like to minimize the amount

$$\sum_i a_{ij} x_{ij} \qquad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

This immediately indicates the multiple objective nature of the problem. Furthermore it is obvious that each lecturer j would prefer the fol= lowing equations to hold:

$$\sum_i c_i x_{ij} = \ell_j \qquad \ldots\ldots\ldots\ldots\ldots\ldots (2)$$

(1) and (2) are concerned with the preferences of the lecturers. Equations (2) could be seen as "soft" constraints in the sense that it would be impossible in general to satisfy all of them exactly. Thus the best that can be achieved is to satisfy them as accurately as pos= sible. In mathematical terms this means that the following goal pro= gramming approach to (2) should be adopted:

$$\sum_i c_i x_{ij} + s_j^+ - s_j^- = \ell_j, \text{ for all } j \qquad \ldots\ldots\ldots\ldots\ldots\ldots (3)$$

where s_j^+ is a variable representing the residual of the lecturer's preferred load and s_j^- is a variable representing the amount by which the lecturer's preferred load is exceeded. In order for (3) to represent a best possible alternative for (2), the variables s_j^+ and s_j^- must be minimized in some way.

Further constraints on the problem is of course that each course i must be handled by one and only one lecturer. This can be expressed as:

$$\sum_j x_{ij} = 1 \text{ for all } i \qquad \text{............ (4)}$$

The question of multiple objective functions can be handled in a variety of ways (see for example Zionts [15], Klein and Hannan [5], Villareal, Karman and Zionts [12] and Zanakis [14]). The problem with most of these approaches is that the computational experience reported so far refer to small examples only. As will be shown later on the present model is a large one. The only practical way in which this problem can be solved is by using a goal programming approach (as was the case in Zanakis [14]). This approach seems even more attractive in view of the fact that goal programming has already been employed in (3). In order to replace the expressions (1) by goal constraints, a goal is needed for the total preference of each lecturer in the quarter under discussion. For this we use the numbers w_j defined above:

$$\sum_i a_{ij} x_{ij} + t_j^+ - t_j^- = w_j, \text{ for all } j \qquad \text{.............. (5)}$$

where the variables t_j^+ and t_j^- must be minimized in some way.

Various complicating constraints arise from the practical situation. One example is a rule that a certain pair of courses cannot be assigned to the same lecturer, since they occupy the same time slot on the time=table. Suppose this is true of courses u and v. Then we need extra constraints of the form:

$$x_{uj} + x_{vj} \leq 1, \text{ for all } j \qquad \text{.............. (6)}$$

Another example of complicating constraints concerns the situation where the students enrolled for a specific course are divided into three dif= ferent groups. A lecturer may then choose to handle two of the groups simultaneously, but without being credited with double the load for one group. The reason for this lies in the way the load for each course is calculated (see section 2.2). For example, the credit given for the preparation for the lectures are not doubled when lectures are repeated. In order to illustrate how a situation such as this can be incorporated into our model, let us suppose that a certain course is divided into three groups a, b and c. Without loss of generality we may assume that a and b, but no other combination of two groups, may be handled by the same lecturer. A new 0-1 variable

$$x_{dj} = \begin{cases} 1 \text{ if lecturer } j \text{ handles a and b} \\ 0 \text{ otherwise} \end{cases}$$

is introduced for each lecturer j. Clearly $x_{dj} = 1$ if and only if $x_{aj} = x_{bj} = 1$. To ensure this, we introduce the constraints

$$-x_{aj} \qquad + x_{dj} \leq 0, \text{ for all } j \qquad \ldots\ldots\ldots (7)$$

$$-x_{bj} \qquad + x_{dj} \leq 0, \text{ for all } j \qquad \ldots\ldots\ldots (8)$$

$$x_{aj} + x_{bj} \qquad - x_{dj} \leq 1, \text{ for all } j \qquad \ldots\ldots\ldots (9)$$

No lecturer may be assigned to all three groups, thus

$$x_{cj} \qquad + x_{dj} \leq 1, \text{ for all } j \qquad \ldots\ldots\ldots (10)$$

Finally, to ensure that no other combination of groups can be assigned to the same lecturer, we introduce the constraints

$$x_{aj} \qquad + x_{cj} \qquad \leq 1, \text{ for all } j \qquad \ldots\ldots\ldots (11)$$

$$x_{bj} \qquad + x_{cj} \qquad \leq 1, \text{ for all } j \qquad \ldots\ldots\ldots (12)$$

In order to attach the correct preferences to the double groups the values of a_{dj} in (5) are replaced by $a_{dj} - a_{aj} - a_{bj}$ for all j. It is also necessary to use the difference $c_d - c_a - c_b$ as the load for the double group in (2).

The minimization of the variables s_j^+, s_j^-, t_j^+ and t_j^- can be done in two ways - the minisum- and minimax-approaches. According to Widhelm [13] the minimax-approach usually gives the most satisfactory results. However, in the case under discussion the minimax-approach will generate many extra constraints, with the result that an already big problem will grow beyond all practical limits. Therefore we follow Harwood and Lawless [4] and McClure and Wells [8] and employ the minisum-approach.

To make sure that a goal variable does not increase in importance only because it appears in a constraint in which it can assume a large value an artifice from Widhelm [13] was used. All constraints of the form

$$\sum_i a_i x_i + g^+ - g^- = b$$

were replaced by

$$\sum_i a_i x_i + ||\underline{a}||g^+ - ||\underline{a}||g^- = b$$

where the vector $\underline{a} = (a_1, a_2 \ldots)$. This ensures that all goal variables assume values of the same order.

In order to present the model in its final form, we define the vectors

$$\underline{a}_j = (a_{1j}, a_{2j}, \ldots)$$

and $\underline{c} = (c_1, c_2, \ldots)$.

Thus our final model has the following form:

$$\text{Minimize} \sum_j (s_j^+ + s_j^- + t_j^+ + t_j^-)$$

Subject to:

$$\sum_i c_i x_{ij} + ||\underline{c}|| s_j^+ - ||\underline{c}|| s_j^- = \ell_j, \text{ for all } j$$

$$\sum_i a_{ij} x_{ij} + ||\underline{a}_j|| t_j^+ - ||\underline{a}_j|| t_j^- = w_j, \text{ for all } j$$

$$\sum_j x_{ij} = 1, \text{ for all } i$$

Complicating constraints such as (6) - (12)

$x_{ij} = 0$ or 1 for all i and j

$s_j^+, s_j^-, t_j^+, t_j^- \geq 0$ for all j.

5. SOLUTION OF THE MODEL

Attempts were made to solve the model in section 4 by using the data of the Department of Applied Mathematics, University of Stellenbosch. A typical problem would involve 10 lecturers and 15 courses, resulting in a mixed integer programming problem with 170 0-1 variables, 40 continuous variables and 180 constraints. The GAMMA matrix generator was used to generate the problem for solution by the FMPS package on the UNIVAC 1100 computer of the University of Stellenbosch. The group variables option available in this package was used in view of constraints (4). The first attempts were disastrous. It was obvious that some kind of sub-optimal strategy was called for.

Firstly the hand solution method of section 3 was used to generate an initial solution to the problem. From this solution were obtained an upper bound on the objective function, upper bounds on s_j^+, s_j^-, t_j^+, t_j^- and the value of the worst preference a_{ij} assigned to any lecturer. These bounds are not, of course, necessary conditions for the optimal solution of the original problem. They can, however, be regarded as a valid heuristic procedure. The upper bound on the objective function was used as a cuttoff value in the branch-and-bound algorithm employed by FMPS. The upper bounds on the goal variables were imposed on the model, since the more tightly the problem is constrained, the less search effort should be necessary to solve the problem. Still the computing time was much too high.

As a desperate measure the problem was solved as an ordinary linear programming problem without any integrality constraints being imposed. Surprizingly, the linear programming solutions obtained, contained many of the x_{ij}-variables at integer values. In one of the best examples 13 of the 17 groups defined by constraints (4) were already satisfied with integer values for all x_{ij}. The x_{ij} variables equal to 1 were then fixed at those values and the branch-and-bound stage was entered to force the remaining variables to integrality. Vastly improved computational performance was observed. The solutions obtained by this approach were then used to generate new cutoff values and the whole branch-and-bound process restarted. A time limit of 90 minutes was placed on each of these runs. In only two cases better solutions (and only marginally better at that) were obtained within this time limit.

A summary of the computational experience with the data of the Department for the 1982 academic year is given in table 1. The computational results for each quarter are presented for the hand method, the method where some variables are fixed at the values they assumed in the linear programming solution and the unrestricted branch-and-bound method (within a time limit of 90 minutes). Particulars are given for each feasible solution found by the branch-and-bound procedure. The iteration count given in column 4 refers to the total number of simplex iterations necessary to generate the given solution.

In view of the fact that only marginally better solutions can be reported in the last row of the table after substantial extra computation, it was decided to accept the solution found after variables with the value 1 in the linear programming solution had been fixed at that value. This solution was then used as a basis for the decision making process described in the next section. This approach seems more acceptable than the one in Harwood and Lawless [4], where the first feasible solution found was accepted as a basis for implementation.

6. IMPLEMENTING THE SOLUTION

The solution found by either the hand method or the mathematical pro= gramming approach remains only a proposal. The head of the department must still make the final decision. Sometimes the head of a department will use the proposal as a basis for making the final decision himself. He has the right to do so and perhaps that will work well for his department.

In the Department of Applied Mathematics, University of Stellenbosch it is believed that enthusiasm and productivity in the implementation of a decision are enhanced if the workers that are responsible for the imple= mentation takes as extensive a part as possible in the decision making

	First Quarter				Second Quarter			
	Objective Value	Branch No	Iteration No	% Improvement on Hand Method	Objective Value	Branch No	Iteration No	% Improvement on Hand Method
Hand Method	1.225				1.6875			
With Upper Bounds and Fixed Variables	1.1699	4	297	4.5	1.4275	54	502	15.4
	1.1694	19	465	4.5	1.3887	548	2223	17.7
	0.9139*	27	625	25.4	1.2637	571	2451	25.1
	*Optimality proved after 1505 iterations.				1.17	599	2587	30.7
					1.1225	645	2757	33.5
					1.12125	1325	4873	33.6
					1.08*	1499	5442	36.0
					*Optimality proved after 8366 iterations			
No fixed variables	No better solution within time limit.				1.0075	2552	7671	40.3

	Third Quarter				Fourth Quarter			
	Objective Value	Branch No	Iteration No	% Improvement on Hand Method	Objective Value	Branch No	Iteration No	% Improvement on Hand Method
Hand Method	1.1325				1.13			
With Upper Bounds and Fixed Variables	1.0757*	552	2352	5.0	1.0415	9	177	7.8
					0.7705*	69	442	31.8
	*Optimality proved after 11610 iterations.				*Optimality proved after 1014 iterations			
No fixed variables	No better solution within time limit.				0.7305*	625	3057	35.4
					*Optimality proved after 6205 iterations			

TABLE 1. Computational results for 1982 data.

process. Therefore, a copy of the proposed work schedule is handed to each lecturer. The staff can then discuss this and exchange courses (in consultation with the head of the Department) before the final schedule is drawn up.

The greatest benefit derived from this exercise was the fact that it generated discussion among members of staff in the initial stages (the completion of the questionnaires) and the final stages (discussion of the final proposal). This discussion impressed upon everyone the com= plexity of the problem. Also, the heuristic method of section 3 was easy to explain and was accepted by the staff. These two factors con= tributed much to the acceptance of the final solution as a genuine im= provement over the usual hit-and-miss method used before.

7. REFERENCES

[1] G.M. ANDREW and R. COLLINS, Matching faculty to courses, *College and University*, 46, 83 - 89 (1971).

[2] J.A. BRESLAW, A linear programming solution to the faculty assign= ment problem, *Socio-Economic Planning Sciences*, 10, 227 - 230 (1976).

[3] J.S. DYER and J.M. MULVEY, An integrated optimization/information system for academic department planning, *Management Science*, 22, 1332 - 1341 (1976).

[4] G.B. HARWOOD and R.W. LAWLESS, Optimizing organizational goals in assigning faculty teaching schedules, *Decision Sciences*, 6, 513 - 524 (1975).

[5] D. KLEIN and E. HANNAN, An algorithm for the multiple objective in= teger linear programming problem, *European Journal of Operational Research*, 9, 378 - 385 (1982).

[6] S.M. LEE, J. VAN HORN and H. BRISCH, A multiple criteria analysis model for academic policies, priorities and budgetary constraints, in *Multiple Criteria Decision Making : Theory and Applications* (G. Fandel and T. Gal, eds.), Springer-Verlag, Berlin (1980), 218 - 237.

[7] S.M. LEE and M.J. SCHNIEDERJANS, A multicriteria assignment problem: A goal programming approach, *Interfaces*, 13, 78 - 81 (1983).

[8] R.H. McCLURE and C.E. WELLS, A mathematical programming model for faculty course assignments, *Decision Sciences*, 15, 409 - 420 (1984).

[9] L. RITZMAN, J. BRADFORD and R. JACOBS, A multiple objective approach to space planning for academic facilities, *Management Science*, 25, 895 - 906 (1979).

[10] W. SHIH and J.A. SULLIVAN, Dynamic course scheduling for college faculty via zero-one programming, *Decision Sciences*, 8, 711 - 721 (1977).

[11] P.I. TILLETT, An operations research approach to the assignment of teachers to courses, *Socio-Economic Planning Sciences*, 9, 101 - 104 (1975).

[12] M.H. VILLAREAL, M.H. KARWAN and S. ZIONTS, An interactive branch and bound procedure for multicriteria integer linear programming, in *Multiple Criteria Decision Making : Theory and Applications* (G.Fandel and T. Gal, eds.) Springer-Verlag, Berlin (1980), 448 - 467.

[13] W.B. WIDHELM, Extensions of goal programming, *Omega*, 9, 212 - 214 (1981).

[14] S.H. ZANAKIS, A staff to job assignment (partitioning) problem with multiple objectives, *Computers and Operations Research*, 10, 357 - 364 (1983).

[15] S. ZIONTS, A survey of multiple criteria integer programming methods, *Discrete Optimization II (Annals of Discrete Mathematics*, 5, 389 - 398 (1979)).

APPENDIX A

NAME: ____________________________________

Please give your preference for each course in each quarter. Do this by numbering the courses in each quarter, starting with 1 (highest preference.)

COURSES	First quarter	Second quarter	Third quarter	Fourth quarter
Applied Mathematics 114	△	△		
Applied Mathematics 144			△	△
Applied Mathematics 244			△	△
Applied Mathematics 312	△	△		
Applied Mathematics 322	△	△		
Applied Mathematics 342			△	△
Applied Mathematics 352			△	△
Applied Mathematics 362			△	△
Applied Mathematics B124	△	△		
Applied Mathematics B154			△	△
Applied Mathematics B224	△	△		
Applied Mathematics B254			△	△
Engineering Mathematics 254			△	△
Theory of Machines A244			△	△
Numerical Methods 314	△	△		
Operations Research A314	△	△		
Electrical Circuits 214	△	△		
Applied Mathematics B334	△	△		
Vibrations and Noise 414	△	△		
TWO1	△			
TWO2	△	△		
TWO3		△		
Quantitative Methods GK05				△

APPENDIX B

1. Indicate, by using a percentage, what fraction of your total lecturing load you would prefer to do in each quarter.

1	2	3	4
□	□	□	□

2. Please indicate which courses you feel you must lecture. Reasons for this could be: You have just completed new notes for the course, you were involved in the planning of a new course and you want to be involved in the implementation thereof, etc.

----

3. Please give the specific work load you would choose for yourself, taking into account only the mean work load for this year, as well as the preferences indicated by you in 1 and 2 above.

First quarter	Second quarter	Third quarter	Fourth quarter

The total work load for this is: ------------------------------

APPENDIX C

Academic work is measured in terms of the number of hours spent on it during the academic year. The basis for the calculation of the work load generated by each course or each graduate student is as follows:

UNDERGRADUATE.

Preparation for and giving of ordinary lecture of 40 minutes:
3 hours (This includes the time for the lecture session itself.)

Setting of and presence at one tutorial for B Sc : 4 hours.
Setting of and presence at one practical class for B Ing : 4,5 hours.
Setting of one test paper : 8 hours.
Setting of one examination paper : 10 hours.
Marking of test or examination answer paper : 20 minutes per paper.
Writing of notes for new course : 2 hours per lecture session.

GRADUATE.

Hons- and M-courses : 5 hours per lecture session of 1 hour.
Setting and marking of examination paper for full graduate course:
20 hours.
Supervision of M Sc thesis : 120 hours.
Supervision of Ph D thesis : 240 hours.

APPENDIX D

EXAMPLE OF USE OF ALGORITHM IN SECTION 3.

Consider the following problem:

	L1	L2	L3	L4	
C1	5	3	1	3	142
C2	3	5	4	2	137
C3	7	6	3	1	82
C4	2	2	2	7	121
C5	6	1	5	6	120
C6	1	4	6	5	37
C7	4	7	7	4	63
	208	131	178	170	

Note that 208 + 131 + 178 + 170 = 687 < 702 = 142 + 137 + 82 + 121 + 120 + 37 + 63. Thus the last row is multiplied by (702/687), resulting in the first matrix given below. An encircled number means the lecturer in that column is assigned to the course in that row. The inequality $LC_i - LL_j > 50$ was used as a criterium to decide whether lecturer j should be assigned to course i.

MATRIX 1

	L1	L2	L3	L4	
C1	5	~~3~~	1	3	142(2;3)
C2	3	~~5~~	4	2	137(1;3)
C3	7	~~6~~	3	1	82(2;3)
C4	2	~~2~~	2	7	121(0;2)
~~C5~~	~~6~~	~~1~~	~~5~~	~~6~~	~~120~~(4;5)←
C6	1	4	6	5	37(3;4)
C7	4	7	7	4	63(0;4)
	213	~~134~~ 14	182	174	

MATRIX 2

	L1	L2	L3	L4	
C1	5	-	1	3	142(2;3)
C2	3	-	4	2	137(1;3)
C3	7	-	3	1	82(2;3)
C4	2	-	2	7	121(0;2)
~~C6~~	~~1~~	~~4~~	~~6~~	~~5~~	~~37~~(3;4)←
C7	4	7	7	4	63(0;4)
	~~213~~ 176	14	182	174	

MATRIX 3

	L1	L2	L3	L4		
~~C1~~	~~5~~	~~-~~	~~1~~	~~3~~	~~142~~	(2;3)←
C2	3	-	~~4~~	2	137	(1;3)
C3	7	-	3	1	82	(2;3)
C4	2	-	~~2~~	7	121	(0;2)
C7	4	7	7	4	63	(0;4)
	176	14	~~182~~ 40	174		

MATRIX 4

	L1	L2	L3	L4	
C2	~~3~~	-	-	2	137(1;3)
C3	7	-	3	1	82(2;3)
~~C4~~	~~2~~	~~-~~	~~-~~	~~7~~	~~121~~(5;7)←
C7	4	7	7	4	63(0;4)
	~~176~~ 55	14	40	174	

MATRIX 5

	L1	L2	L3	L4		
~~C2~~	~~-~~	~~-~~	~~-~~	~~2~~	~~137~~	(∞; ∞) ←
C3	7	-	3	1	82	(2;3)
C7	4	7	7	4	63	(0;4)
	55	14	40	~~174~~ 37		

MATRIX 6

	L1	L2	L3	L4	
~~C3~~	~~7~~	~~-~~	~~3~~	~~1~~	~~82~~(2;3)
C7	4	7	7	~~4~~	63(0;4)
	~~55~~ -8	14	40	~~37~~ -45	

Final assignment : L1 → C4, C6 and C7

L2 → C5

L3 → C1

L4 → C2 and C3