T.C.

YEDİTEPE UNIVERSITY

GRADUATE INSTITUDE OF SCIENCE AND ENGINEERING

A MATHEMATICAL MODEL FOR PLANNING OF CLASS TIMETABLES: AN APPLICATION IN TURKISH MILITARY ACADEMY

by

A. Cem ÜSTÜNEL

Supervisor

Asst. Prof. Dr. Varol GÜNYAŞAR

Submitted to the Graduate Institute of Science and Engineering
in partial fulfillment of the requirements for the degree of
Master of Science
in Systems Engineering

ISTANBUL, 2001

A MATHEMATICAL MODEL FOR PLANNING OF CLASS TIMETABLES:
AN APPLICATION IN TURKISH MILITARY ACADEMY

by

A. Cem ÜSTÜNEL

Approved by:

Asst. Prof. Dr. Varol GÜNYAŞAR ....
(Supervisor)

Assoc. Prof. Dr. Melek DEMİRHAN ....

Prof. Dr. Linet ÖZDAMAR ....

Date of Approval by the Administrative Council of the Institute 04 / 12 / 2001

LIST OF ABBREVIATIONS

CSP	Constraint satisfaction problem
esp.	especially
etc.	et cetera
GAs	Genetic algorithms
i.e.	id est (that is)
i.g.	for example
OR	Operations research
PC	Personal computer
resp.	respectively
SA	Simulated annealing
TMA	Turkish Military Academy
TS	Tabu search
vs.	verse

LIST OF FIGURES

Figure 4.6 Flowchart of the solution process.. 46

LIST OF TABLES

Table 4.1 Unavailable periods of sections occupied by military and physical training and foreign language course....36

Table 4.2 Unavailable periods of sections occupied by the course of leadership.... 36

Table 4.3 Frequencies of courses and the number of teachers previously assigned.... 36

Table 4.4 Unavailable periods of teachers.... 37

Table 4.5 The penalty matrix for cost coefficient values.... 38-39

Table 4.6 The timetable of sections 1-6.... 49

Table 4.7 The timetable of sections 7-12.... 50

Table 4.8 The timetable of sections 13-18.... 51

Table 4.9 The timetable of sections 19-24.... 52

Table 4.10 The timetable of teachers....53-54

ACKNOWLEDGEMENTS

First of all, I am thankful to my thesis advisor Assist. Prof. Varol GÜNYAŞAR for his academic contributions and unwavering support throughout the writing of this thesis, from its inception to its publication.

I would like to express my gratitude to Operations Research Department at Turkish Military Academy, especially to Major Haluk AYGÜNEŞ and Captain Alpaslan ÇETİN, for their invaluable comments, support, and advises throughout my study. I can never forget their support in modelling the problem and obtaining research documents and information. I am also grateful to Planning and Programming Department at Turkish Military Academy for providing me necessary documents regarding the timetabling process.

I also would like to express my gratitude to all my colleagues who were my classmates during the master education.

Very special thanks are for my wife Esra who had the most essential contribution to this study: encouraging me when I got very tired of studying and making me feel her patience and support at all times.

Finally, I am deeply grateful to the Turkish Army for giving me the opportunity to pursue a Master's degree.

ABSTRACT

A timetabling problem of an educational institution involves scheduling a large number of classes, teachers and courses to a number of periods while satisfying a set of constraints of various types. The aim of this study is to find good and satisfactory solutions for the timetabling problem of Turkish Military Academy in shorter time when compared to the manual timetabling process. An integer programming model which has the appropriate facilities for providing valuable help to the scheduler to implement a good timetable of Turkish Military Academy and a solution strategy for this model is presented. The model satisfies the teachers' preferences to a sufficient degree by using suitable objective function coefficients. These coefficients penalize the assignment of teachers to the periods which they do not prefer. They also enable us to assign the part-time teachers at their available periods prior to the full-time ones. In addition, the solution strategy proposed is based on grouping sections and teachers, and on allocating teacher groups to section groups iteratively in such a way that the output data of an iteration will be added to the input data of the next one. This grouping strategy made it possible to decompose the problem into the problems of smaller sizes. The solution found by the implementation of the model in this study satisfies all problem requirements and minimizes the objective function. It is near-optimal within the iteration limits and is found in several hours. The timetable constructed has only one non-preferred period for only one teacher. Besides, the time consumed for this model is very short when compared with the manual timetabling. Thus, it is said to be "a good timetable" both for the teachers and the scheduler.

ÖZET

Bir eğitim kurumunun zaman çizelgeleme problemi, çeşitli tipteki kısıtları sağlayarak büyük miktarlardaki sınıf, öğretmen ve dersleri belli sayıdaki zaman dilimlerine tahsis etmeyi gerektirir. Bu çalışmanın amacı, Kara Harp Okulu' nun ders çizelgeleme problemine elle yapılan işleme kıyasla daha az zamanda daha iyi ve tatmin edici çözümler bulmaktır. Kara arp Okulu' nun ders çizelgelerini ortaya koymak üzere bu işle görevli kişilere değerli bir yardım sağlayacak olan bir tamsayılı programlama modeli ve bu model için bir çözüm yolu sunulmuştur. Model, uygun amaç fonksiyonu katsayıları kullanılarak öğretmen tercihlerini belli bir seviyeye kadar tatmin etmektedir. Bu katsayılar öğretmenlerin tercih etmedikleri zaman dilimlerine atanmalarına ceza puanı vermektedir. Ayrıca, diğer üniversitelerden kısıtlı bir zaman için gelen öğretmenlerin okulun kendi öğretmenlerine nazaran öncelikli olarak onların uygun oldukları zamanlarda atanmalarını da sağlamaktadırlar. Ek olarak, kısımları ve öğretmenleri gruplara ayırarak her bir grubu bir sıra dahilinde birbiriyle eşleştiren ve bir aşamadaki eşleştirme sonuçlarını bir sonrakinin girdi verisi olarak kullanan bir çözüm yolu önerilmiştir. Bu gruplama yöntemi problemi daha küçük ölçeklerdeki alt gruplara ayrıştırmamızı sağlamaktadır. Bu çalışmadaki modelin uygulama sonucu elde edilen çözümü tüm problem gereksinimlerini karşılamakta ve amaç fonksiyonunu en az bir değere indirgemektedir. Sonuç, modele girilen tekrar kısıtlamaları dahilinde en uygun değere yakın bir sonuçtur ve saatlerle ifade edilebilecek bir sürede bulunmuştur. Elde edilen ders çizelgesi sadece bir öğretmen için tek bir tercih edilmeyen ders saati ihtiva etmektedir. Ayrıca, elle yapılan çizelgelemeyegöre bu model için daha azbir zaman harcanmıştır. Bu nedenle, hem öğretmenler, hem de planlayıcı açısından "iyi bir çizelge" olarak nitelenebilir.

1. INTRODUCTION

A timetabling or course scheduling problem consists of fixing a sequence of meetings between teachers and students in a prefixed period of time (typically a week), satisfying a set of constraints of various types.

Among the constraints which have to be taken into account for a typical timetabling problem are:

a. No class, subject, room, and teacher overlaps are allowed,

b. One has to take care of class, subject, room, and teacher availabilities,

c. Teachers may prefer to teach at certain times of the week,

d. Compactness constraints have to be regarded,

e. Frequencies of the subjects,

f. Distribution of the lectures of a subject over the week have to be considered,

g. Rule of consecutive lectures of a given subject have to be taken into account,

h. Pre-assignments of lectures must be possible,

i. Lunch breaks have to be observed, etc.

Note that compactness constraints typically require no free time (other than breaks or lunch breaks) between lessons for the students.

Process of timetabling or course scheduling is a necessary and important activity for all educational institutions including schools, universities, and colleges. The resulting weekly schedules of the timetabling process have a great influence on the job satisfaction of the teachers and the efficiency of the lessons.

It is not an easy task to construct a good timetable, but it must be done every year at every school. The one who is responsible for scheduling the courses has to take account of the characteristics of the school week, preassignments, availability of teachers and classes, frequencies and frequency divisions (period blocks) of a course, and many other constraints in order to obtain a good solution for classes and teachers.

School timetables have been constructed by hand for many years. But, the manual construction of timetables is time consuming and tedious. The timetablers employ several days of work to manually solve the problem, and a satisfactory solution taking account of the availabilities and the preferences of the teachers is rarely achieved. Therefore, there appear complaints from teachers to the manual timetable.

Timetabling and course scheduling models must accommodate the characteristics and regulations of specific education systems. Therefore the problem varies from country to country, and even from one kind of educational institution in a country to the other. Depending on the specifications of the education system, the timetabling problem includes a large variety of problems. There exist numerous timetabling problems differing from each other based on the type of the educational institution involved and distinct constraints. Each institution has different timetabling problem, so the modelling and solution methods will be different.

For large institutions such as Turkish Military Academy (TMA), the timetabling problem becomes more difficult due to the complexities of the Academy' s curriculum structure and the various constraints and objectives. TMA is a military institution where cadets have a four-year academic and military education. Its mission is to educate, train and inspire the Corps of Cadets so that each graduate is a commissioned leader of character committed to the values Honor, Honesty, Affection, Duty, Country; professional growth throughout a career as an officer in the Turkish Army; and a lifetime of selfless service to the nation. Daily activities of the students are carefully regimented and balanced considering the academic, military and physical requirements.

At the time being, the construction of timetables is being performed manually at TMA. Timetablers of TMA employ several days of work to manually solve the problem, and a satisfactory solution taking account of the availabilities and the preferences of the teachers is rarely achieved.

In this study an integer programming model will be constructed for the timetabling problem of Turkish Military Academy (TMA). Since the problem for the TMA' s four years as a whole is too large to deal with, the timetabling problem will be restricted with the second-year cadets. But, the model can be iteratively applicable to the other years by revising the data and making little modifications. Having decided to utilize integer programming for building the timetabling model, a thorough and detailed set of variables has been used in this study for the representation of the problem. This allows for the modelling of a significant number of rules and regulations and provides the indirect satisfaction of a large set of quality rules. The detailed and multidimensional nature of the variables allows for building an efficient and compact model.

The thesis has been organized as follows: In Chapter 2, a literature review is presented to introduce the modelling and solution approaches for several timetabling problems. In Chapter 3; the rules and regulations of the timetabling problem for TMA are outlined, the dimensions of all variables are defined, and a detailed presentation of the model including the objective and the constraints is made. In Chapter 4, a complete formulation of the timetabling problem for the second-year cadets of TMA has been presented and solved. In the last chapter, conclusions and implications for further research have been presented.

2. THE TIMETABLING PROBLEM: A LITERATURE REVIEW

2.1. Definition of the Timetabling Problem

Abramson (1991) states that the school timetabling problem involves scheduling a number of tuples, each consisting a class of students, a teacher, a subject and a room, to a fixed number of time slots. A number of such tuples may be scheduled in the same time slot providing no class, teacher or room appears more than once in the time slot.

After defining teachers, classes, lecture halls, laboratories, pieces of equipment, etc. as "participants", and the collection of participants which have to come together and the number of hours required for it as a "meet"; Schmidt and Ströhlein (1980) state that a timetable is a schedule assigning to all these meets the precise number of hours required, so that these hours are available for all participants of the meets and such that, as a fundamental requirement, none of the participants is scheduled twice in the same hour.

Birbas et al. (1997) state that the timetabling problem involves scheduling a certain number of resources, such as classes, teachers, and courses to a number of time-periods on a daily basis. The valid combinations for these resources are those that avoid conflicts among teachers and classes during all time-periods of the timetable, while obeying all rules and regulations of the school system.

De Werra (1997) suggests that the timetabling problems should be considered as different. One reason for this is that they differ from each other not only by the types of constraints that are to be taken into account, but also by the density of the constraints. The other reason is that the solution methods may be quiet different.

For the above reasons, a considerable attention has been devoted to automated timetabling. During the last thirty years, many contributions related to timetabling have appeared, several applications have been developed and employed, and it will probably continue with the same rate for years. Because the problem varies from country to country, there are several models and algorithms for schools in different countries, such as Australia (Abramson, 1991), Great Britain (Wright, 1996), Spain (Alvarez-Valdes et al., 1996), Switzerland (Chahal and de Werra, 1989), Greece (Birbas et al., 1997).

There is usually no clear distinction made between timetabling and course scheduling. There seems to be confusion on the different topics of the former and the latter. Defining the borderline between both categories as a function of the problem size is really not sufficient. De Werra (1985) states that we are confronted with academic scheduling when a university (or even a school) offers a collection of courses (each one consisting of a given number of lectures) and there is no fixed curriculum: Each student may choose a certain number of courses. His definition of course scheduling problem consists in assigning each lecture to some period of the week in such a way that no student is required to take more than one lecture at a time. He also states that the examination scheduling problem is quiet similar.

Drexl and Salewski (1997) point out the differences between three important problem categories, i.e. school timetabling, academic course scheduling, and other closely related timetabling problems, respectively. In their article, after defining the process of teaching as "job", and the resources, such as classes, teachers, rooms, etc. as "item", they compile the issues which are "must" for any of the three problem categories and name them "necessary issues". Then, they list the optional issues, such as room categories, precedence constraints between jobs, workload constraints, breaks, preferences for rooms and/or hours, compactness constraints, etc. Finally, they categorize the problem with respect to the presence/absence of the optional issues. Their definition of categories is as follows:

a. *School timetabling*: The timetable must satisfy compactness constraints for students and for teachers. In addition the large teaching units have to be distributed over the week in order to get acceptable timetables. Other optional issues might be accommodated in specific situations. Aubin and Ferland (1989)'s work, which they deal with a "large scale" timetabling problem, is an example for that category. Their model includes two main components (the timetabling subproblem and the grouping subproblem) that interact via the objective function. As another example; Abramson (1991) considers pupils, teachers, subjects, and rooms as relevant items and examines the use of simulated annealing to solve this problem.

b. *Academic course scheduling*: In contrast, at universities there do not exist compactness constraints, neither for students nor for professors. In addition, the teaching units per week are not that large that they have to be split into smaller pieces of work. But, a timetable for universities will only be acceptable if and only if it takes care of the professor's preferences for rooms and/or hours. For instance; Tripathy (1980) considers a course scheduling problem, which does not allow to take care of rooms of different sizes, but which considers the professors' preferences for periods.

c. *Other timetabling problems*: No general guidelines can be established. It depends on the specific timetabling problem under consideration which optional issues become necessary. In this category the farmost relevant problem is the examination scheduling problem which has been dealt with by Tripathy (1980) and by Dimopoulou and Miliotis (2001).

Burke and Elliman (1994) give a similar categorization. They point out some minor differences between a school timetabling problem and a university timetabling problem although they may at the first glance appear to be the same problem. One such difference they point out is that in a school it can usually be assumed that classes are roughly the same size and that all rooms in the school are sufficiently large to house any class. However, in a university, classes can range from 5 students to more than 200.

They also point out the differences of a course timetabling problem from an examination timetabling problem. With examination scheduling, "student-conflicting" courses, i.e. courses having a student in common, cannot clash and adjacent exams are to be avoided. With course timetabling, it is desirable that courses do not student-conflict or that the conflicts be kept to a minimum. However, in most situations a solution without some student-confliction does not exist. This simply means that the student has to choose between the conflicting courses. However, with examination timetabling, it is essential that no two exams conflict. Examination scheduling is easier in the sense that an exam is a "one-off" occurrence, whereas a course has to be offered two or three times a week. Another difference they point out is that lecturer-confliction can be ignored in the examination problem, but it is essential that it does not occur in the course problem.

Whatever its category is, timetabling/scheduling problems are problems of time-based planning and combinatorial optimization that tend to be solved with a cooperation of search and heuristic methods, which usually lead to satisfactory but sub-optimal solutions. Fang (1994) gives some reasons for answering the question of "what makes the timetabling/scheduling problems so difficult?":

a. The space of possible solutions for most real problems is too large for search methods to be feasibly applied.

b. Advanced search techniques using various heuristics to prune the search space will not be guaranteed to find an optimal (or near optimal) solution. In other words, it is very difficult to design effective heuristics.

c. Timetabling/scheduling problems are often complicated by the details of a particular timetabling/scheduling task. A general algorithmic approach to a problem may turn out to be inapplicable, because of the certain special constraints required in a particular instance of that problem.

d. Real world timetabling/scheduling problems often involve constraints that cannot be precisely represented or even precisely stated.

2.2. The Class-Teacher Model

Timetabling problem in its simplest form can be defined as a problem of assigning each lecture to some period such that no teacher (resp. no class) is involved more than one lecture at a time. Because of the reasons addressed above for the difficulties of the timetabling problems, many approaches and models have been proposed for dealing with a variety of timetabling problems and it will probably continue with the same rate for years.

In a review article, de Werra (1985) noted that there were two distinct stages to the timetabling process:

a. First, the curricula are defined for each class or for each group of students and one has to assign the various resources (in manpower or in equipment) to the classes.

b. Second, when an agreement has been reached concerning these assignments of resources, then one tries to see whether a workable detailed timetable can be worked out which is compatible with all the previously defined requirements.

Although some work has been done on the assignment problems implicit in the first stage, the traditional timetabling problem is usually identified with the second stage, that of ordering the various classes or groups over a specified period of time (usually a week) subject to stated restrictions on the various resources involved. Indeed most of the approaches described in the OR and computing literature relate to some aspect of this second stage, usually in the context of a specific problem situation. (Johnson, 1993)

A timetabling problem can be suitably modelled in terms of a set of constraints. The chosen method for problem modelling has a great influence on the solution search. Often the success of the search depends directly on the chosen model. Furthermore, the possibility for modelling depends on the chosen constraint solver. (Goltz et al. 1997)

2.2.1. A simple model

A class will consist of a set of students who follow exactly the same program. Let $C = \{c_1, \quad, c_m\}$ be a set of classes and $T = \{t_1, \quad, t_m\}$ a set of teachers. We are given an $m \times n$ requirement matrix $R = (r_{ij})$ where r_{ij} is the number of lectures involving class c_i and teacher t_j. (de Werra, 1985)

It will be assumed that all lectures have the same duration (say one period). Given a set of p periods, the problem is to assign each lecture to some period in such a way that no teacher (resp. no class) is involved in more than one lecture at a time. More precisely, if x_{ijk} is defined to be 1 if class c_i and teacher t_j meet at period k and 0 otherwise, then the problem CT1 below must be solved (de Werra, 1985):

$$\sum_{k=1}^{p} x_{ijk} = r_{ij} \qquad (i = 1,\ldots,m;\ j = 1,\ldots,n), \qquad (2.1)$$

$$\sum_{j=1}^{n} x_{ijk} \leq 1 \qquad (i = 1,\ldots,m;\ k = 1,\ldots,p), \qquad (2.2)$$

$$\sum_{i=1}^{m} x_{ijk} \leq 1 \qquad (j = 1,\ldots,n;\ k = 1,\ldots,p), \qquad (2.3)$$

$$x_{ijk} = 0 \text{ or } 1 \qquad (i = 1,\ldots,m;\ j = 1,\ldots,n;\ k = 1,\ldots,p), \qquad (2.4)$$

Till now, no distinctions between the daily and weekly scheduling problems have been made. In daily problem, one has to assign each lecture to some hour of the day and R represents all lectures that have to be scheduled on one day (de Werra, 1985).

In the weekly scheduling problem each lecture has to be assigned to some day of the week and R represents all lectures that must take place during the week. In such a situation for each class c_i (resp. teacher t_j) we have a positive integer a_i (resp. b_j) representing the maximum number of lectures in which c_i (resp. t_j) may be involved during each one of the p days. The assignment to days is now formulated as follows (x_{ijk} will be the number of lectures involving c_i and t_j, which are assigned to day k): Find values of x_{ijk} satisfying (2.1) and the problem CT2 below (de Werra, 1985):

$$\sum_{j=1}^{n} x_{ijk} \leq a_i \qquad (i = 1,.....,m; \quad k = 1,......p), \tag{2.5}$$

$$\sum_{i=1}^{m} x_{ijk} \leq b_j \qquad (j = 1,.....,m; \quad k = 1,......p), \tag{2.6}$$

$$x_{ijk} \geq 0 \text{ integer}, \quad (i = 1,.....,m; \quad j = 1,......,n; \quad k = 1,......p), \tag{2.7}$$

Clearly, if a_i and b_j are given, then the minimum number of days p needed is given by (de Werra, 1985);

$$p = \max \left(\max_j \left\lceil \sum_{i=1}^{m} r_{ij} / b_j \right\rceil , \max_i \left\lceil \sum_{j=1}^{n} r_{ij} / b_j \right\rceil \right)$$

where $\lceil t \rceil$ is the smallest integer not less than t.

In most cases however, the number of days p is given and one wants to spread the lectures involving the same class c_i and the same teacher t_j as uniformly as possible throughout the p days. The formulation is then the following: Find integer values x_{ijk} satisfying (2.1) and the problem CT3 below (de Werra, 1985):

$$\left\lfloor \sum_{j=1}^{n} r_{ij} / p \right\rfloor \leq \sum_{j=1}^{n} x_{ijk} \leq \left\lceil \sum_{j=1}^{n} r_{ij} / p \right\rceil \qquad (i = 1,.....,m; \quad k = 1,......p), \tag{2.8}$$

$$\left\lfloor \sum_{i=1}^{m} r_{ij} / p \right\rfloor \leq \sum_{i=1}^{m} x_{ijk} \leq \left\lceil \sum_{i=1}^{m} r_{ij} / p \right\rceil \qquad (j = 1,.....,n; \quad k = 1,......p), \tag{2.9}$$

$$\lfloor r_{ij} / p \rfloor \le x_{ijk} \le \lceil r_{ij} / p \rceil \qquad (i = 1,...,m;\ \ j = 1,...,n;\ \ k = 1,...p), \qquad (2.10)$$

Here $\lfloor t \rfloor$ is the largest integer not larger than t. Constraints (2.8) and (2.9) simply express that the daily loads of all classes and of all teachers are perfectly balanced over the p days (de Werra, 1985, p153)

In the next part, the daily problem (x_{ijk} = 0 or 1) will be reconsidered since in practice there are additional requirements that have to be taken into account (de Werra, 1985).

2.2.2. Preassignments

Some pairs c_i, t_j, have to meet at fixed periods k; this means that for some triples i, j, k the value of p_{ijk} will be 1. In CT1, new constraints will be introduced:

$$x_{ijk} \ge p_{ijk} \quad (2.11)$$

where p_{ijk} will be 0 if there is no preassigned meeting of class c_i and teacher t_j at period k (de Werra, 1985).

2.2.3. Unavailabilities

At some periods k a teacher t_j (or a class c_i) may be unavailable. Such constraints may be reduced to preassignments by stating that if t_j is unavailable at period k, there is a meeting of t_j with a dummy class c_i^* at period k and similarly for a class c_i which is unavailable at some period (de Werra, 1985).

Instead of keeping constraints (2.11), a different formulation will be presented for the daily problem with preassignments and unavailabilities. We define $\bar{x}_{ijk}$ = 1 if class c_i and teacher t_j meet at period k for a lecture which is not a preassignment and $\bar{x}_{ijk}$ = 0 otherwise. Then the following constraints CT4 will appear(de Werra, 1985).

$$\sum_{k=1}^{p} \bar{x}_{ijk} = \bar{r}_{ij} \qquad (i = 1,.....,m;\ \ j = 1,......n), \qquad (2.12)$$

$$\sum_{j=1}^{n} \bar{x}_{ijk} \le \bar{b}_{ik} \qquad (i = 1,.....,m;\ \ k = 1,......p), \qquad (2.13)$$

$$\sum_{i=1}^{m} \bar{x}_{ijk} \leq \bar{c}_{jk} \qquad (j = 1,.....,m; \quad k = 1,......p), \qquad (2.14)$$

$$\bar{x}_{ijk} = 0 \text{ or } 1 \qquad (i = 1,...,m; \quad j = 1,...,n; \quad k = 1,...p), \qquad (2.15)$$

where;

$$\bar{r}_{ij} = r_{ij} - \sum_{k=1}^{p} p_{ijk},$$

$\bar{b}_{ik} = 1$ if c_i is available and not preassigned at period k, 0 otherwise,

$\bar{c}_{ik} = 1$ if t_j is available and not preassigned at period k, 0 otherwise.

As it stands, this is a feasibility (or satisficing) problem rather than an optimality problem. The aim is to find a feasible or satisfactory resolution of the stated constraints rather than to optimize a chosen objective function (Johnson, 1993).

2.2.4. Feasibility vs. optimality

Most timetabling problems discussed so far have been formulated as feasibility problems (find a feasible solution) rather than optimization problems (find an optimal solution) (de Werra, 1985).

In fact, both formulations are very similar since finding a feasible solution may be regarded as minimizing a kind of "distance to feasibility" (de Werra, 1985).

2.3. Course Scheduling Models

2.3.1. Basic formulation

The course scheduling problem arises when a university (or even a school) offers a collection of courses (each one consisting of a given number of lectures) and there is no fixed curriculum: Each student may choose a certain number of courses. The problem consists of assigning each lecture to some period of the week in such a way that no students are required to take more than one lecture at a time. The situation is quite similar to the examination scheduling problem. (de Werra, 1985)

For the course scheduling problem, the following graph-theoretical model is used: We associate with each lecture l_a of each course K_b a lecture-node m_{ab}, for each course K_b we introduce edges between all pairs of lecture-nodes in K_b . Also whenever there is a student taking courses K_b and $K_{\bar{b}}$ we introduce an edge between each pair of lecture-nodes m_{ab}, $m_{\bar{a}\bar{b}}$. (de Werra, 1985)

A feasible course schedule in p periods will correspond to a node coloring of the above graph, with p colors: Each node receives some color and no two adjacent nodes are allowed to have the same color. (de Werra, 1985)

Depending on the number p of periods, it will not always be possible to find a schedule in p periods. In fact, deciding whether a graph has a node coloring with p colors is known to be an NP-complete problem when $p \geq 3$ (de Werra, 1985).

2.3.2. An example for the formulation of course scheduling

In this section, a formulation given by de Werra (1985) that exploits the underlying network structure of the problem presented in an article of Tripathy (1980) will be considered:

Given a collection of q courses $K_1, \ldots, K_q$; course K_i consists of k_i lectures of one period each. The total number of periods is p. The students are divided into r groups $S_1, \ldots, S_r$ such that in each S_l all students take exactly the same courses; l_k is the maximum number of lectures which can be scheduled at period k (this may correspond to the number of available classrooms) (de Werra, 1985).

Then define $y_{ik} = 1$ if a lecture of course K_i is scheduled at period k and $y_{ik} = 0$ otherwise (de Werra, 1985):

$$\text{Max} \sum_{i=1}^{q} \sum_{k=1}^{p} c_{ik}\, y_{ik}, \qquad (2.16)$$

$$\text{s. t.} \sum_{k=1}^{p} y_{ik} = k_i \qquad (i = 1, \ldots, q), \qquad (2.17)$$

$$\sum_{i=1}^{q} y_{ik} \leq l_k \qquad (k = 1, \ldots, p), \qquad (2.18)$$

$$\sum_{i \in S_l} y_{ik} \leq 1 \qquad (l = 1,......,r;\ k = 1,.....,p), \qquad (2.19)$$

$$y_{ik} = 0 \text{ or } 1, \qquad (2.20)$$

The costs c_{ik} occurring in the objective function (2.16) are the measures of the desirability of $y_{ik} = 1$; one can assign a higher value if it is desired to have a lecture of course K_i in period k, or 0 otherwise. Applying a Lagrangean relaxation technique, the constraints (2.19) are incorporated in the objective function below (de Werra, 1985);

$$\text{Max} \sum_{i=1}^{q} \sum_{k=1}^{p} c_{ik}\, y_{ik} + \sum_l \sum_k \lambda_{kl} \left(1 - \sum_{i \in S_l} y_{ik}\right) \qquad (2.21)$$

The constraints are now (2.17), (2.18), (2.20) so that one has a capacitated transportation problem; the values λ_{kl} were obtained with a subgradient optimization method combined with a Branch and Bound procedure (de Werra, 1985).

2.4. Solution Approaches to the Timetabling Problem

Timetabling problems have attracted the continuous interest of researchers mainly because they provide the opportunity of testing combinatorial solution methods in formulations that represent difficult practical problems. For this reason, there has been an extensive study of several timetabling problems throughout the literature and various approaches have been adopted for their solution.

The computer-science and operations research approaches culminated with an extensive review of the literature by Schmidt and Ströhlein (1980). Of their 343 references, 274 were in English, 57 in German, and 12 in various other European languages. Schmidt and Ströhlein made several predictions of future trends that we can look back on after more than a decade of additional experience. They predicted that:

"A major evolution will, therefore, come in the near future. Timetable programs will probably move from remote handling in huge computing centers to minicomputer systems owned by the school and handled directly by the teachers."

This did not happen in the near future but is happening now with microcomputers rather than minicomputers. Schmidt and Ströhlein also predicted that: "... software support by database systems for bookkeeping will grow rapidly". This has turned out to be more true than they perhaps expected. Many modern school timetabling systems do not actually timetable. Instead they provide data input facilities, arrange the data to make timetabling easy, and produce reports; in other words, only the bookkeeping services envisioned by Schmidt and Ströhlein. This greatly eases the often enormous clerical burden on timetablers. (Bezeau, 1993)

Timetabling models and methods known so far are closely related to one of the following approaches:

2.4.1. Random and/or exhaustive search approaches

Most of the early techniques (see Schmidt and Ströhlein, 1980) were based on a simulation of the human way of solving the problem. That is, a partial timetable is filled in, lecture by lecture, until either all lectures have been scheduled or no lecture can be scheduled without violating the constraints (Schaerf, 1996).

The performance of random search methods on timetabling/scheduling problems will typically be a function of what proportion of the space of solutions are actually *good* solutions. This ratio is often very low, because such problems often have very tight constraints. Therefore, looking randomly for a good timetable/schedule is like looking for a needle in a haystack. Classical heuristic search-based techniques often perform adequately on small timetabling/scheduling problems, but in larger problems the size of the search space is such that we can expect classical heuristic search-based techniques to be very time consuming. Classical heuristic search can be seen as very similar to the method used by human experts. In the same way, it can be expected to find local minima often. Exhaustive search is known to be infeasible for NP-Complete problems due to their immense search space. Some of the constraint-satisfaction approaches treat any solution as a good solution; they use no separate measure of quality in addition to the set of hard constraints (Fang, 1994).

Due to vast search space traversed by exhaustive search and the limitations of exact algorithms, the implicitly enumerative methods or more efficient heuristic methods are usually employed to solve timetabling/scheduling problems (Fang, 1994).

2.4.2. Operations research approaches

2.4.2.1. Enumerative search

a. *Mathematical programming* : Mathematical programming is a family of techniques for optimizing a function constrained by independent variables (Fang, 1994).

Linear and integer programming or Lagrangean relaxation are the examples to this approach. Linear programming involves the optimization (maximization or minimization) of a linear objective function subject to linear constraints. It has been widely used in government and industry to solve problems associated with scheduling, transportation, construction, and manufacturing. Objections to using linear programming in timetabling have generally been based on concerns about the size of the problem but these are becoming increasingly irrelevant as computers become more powerful and as computer time becomes less expensive (Bezeau, 1993).

Schmidt and Ströhlein (1980) discussed the "operations research approach" which consisted essentially of linear programming. Like the others, they were not optimistic about the future of linear programming in timetabling because the problems can become large and complex very quickly and can consume large amounts of computer time. This, it turns out, is also true of the other approaches. The consumption of large amounts of computer time is no longer very relevant because costs have gone down by several orders of magnitude and continue to decline. What has become important in its place is the effort required to prepare the input for linear programming if the timetabling problem is large.

Tripathy (1980) has tried Lagrangean relaxation to solve either the course or exam scheduling problem whose formulation has been presented in Section 2.3.2. Birbas et al. (1997) and Dimopoulou and Miliotis (2001) solved the school timetabling and course scheduling problem, respectively using integer programming.

b. *Dynamic programming* : Dynamic programming is an implicit enumerative search method which can be seen as a kind of divide-and-conquer technique. In order to solve a large problem, we can first decompose it into several small independent subproblems. Since it's not known which subproblems should be solved first, all the small subproblems can be solved and kept for later use (Fang, 1994).

c. *Branch and bound* : Branch and Bound search is also an implicit enumerative method. This approach consists of two fundamental procedures. Branching the process of partitioning a large problem into two or more subproblems, and bounding is the process of calculating a lower bound on the optimal solution of a given subproblem. (Fang, 1994). Examples of this approach are Hultberg and Cardoso (1997) and Laporte and Desroches (1986). The former first formulated the teacher assignment problem as a mixed integer program and presented an equivalent alternative formulation of the problem. Then, they outlined a branch and bound algorithm for its solution based on this alternative formulation. The latter used branching and bounding processes at the first phase of their algorithm.

2.4.2.2. Heuristic search

a. *General heuristic rules* : De Werra (1985) states that the only available methods are heuristic procedures, which will try to produce a feasible solution but without guarantee of obtaining one whenever a solution exists. He surveyed in his article many of the graph theoretical and network models, and provided an introductory tutorial on the formulation of simple timetabling problems. He mentioned that generally real timetabling problems couldn't be formulated with these simple models; however, many of the heuristic methods are often derived and adapted from exact methods developed for the simple cases.

De Werra (1985) also states that most of these methods combine in a clever way the use of combinatorial models for some structured subproblems with some more or less intuitive decision rules.

In the case of de Werra (1985)'s daily problem CT1, for instance, one observes that if any index is kept fixed in x_{ijk}, then the determination of the values x_{ijk} can be performed by network flow techniques. For instance, if k is fixed, the problem of scheduling lectures at this period k is simply a matching problem in a bipartite graph; one has to assign classes which are available at period k to teachers who are available at period k. How de Werra (1985) defines the graph (or edge) coloring and network approaches is summarized below:

- *Edge coloring approach: Bipartite multigraphs*

De Werra (1985) states that one of the most preferred approaches is to associate a bipartite multigraph, G = (V, E), with the formulation CT1 where V = (C, T) and E = R. Its vertices are the classes (c_i) and the teachers (t_j) where the vertices are connected by r_{ij} parallel edges. Now, if each period k corresponds to a color then the problem can be restated as follows:

"Find an assignment (a color) among p colors to each edge of G such that no two adjacent edges have the same color; hence $x_{ijk} = 1$ if some edge (c_i , t_j) gets color k."

This procedure includes p iterations (coloring p bipartite graphs) and the necessary and sufficient condition for a feasible solution to exist has been stated by de Werra (1985) as a proportion: There exists a solution to CT1 iff:

$$\sum_{i=1}^{m} r_{ij} \leq p \qquad (j = 1,.....,n),$$

$$\sum_{j=1}^{n} r_{ij} \leq p \qquad (i = 1,.....,m).$$

It means that there is a timetable in p periods if and only if no teacher (resp. no class) is involved in more than p lectures.

Hence it seems natural to try to construct a schedule period after period while trying to schedule as many 'urgent' lectures as possible at each period. Such a method is heuristic since one has no guarantee that after having scheduled lectures at some periods, the remaining problem still admits a solution. At each step a maximum weight matching is constructed with a network flow algorithm. De Werra (1985)

- *Path packing approach: Network flows*

De Werra (1985) introduces network flows as another approach which can perform, for a given class c_i, the construction of a schedule consisting of all lectures involving c_i. He introduces one node k for each period and one node t_j for each teacher; there is an arc (k, t_j) if t_j is available for a lecture with class c_i at period k. He also introduces a node s with arcs (s, k) and a node t with arcs (t_j, t); the capacities $c(t_j, t)$ and the lower bounds $l(t_j, t)$ are r_{ij} . For all previously introduced arcs (x, y) he set $c(x, y) = 1$ and $l(x, y) = 0$. Any feasible integral flow will define a timetable for class c_i.

A schedule for c_i exists if and only if for any subset T of teachers, the number of periods where at least one teacher in T is available is at least as large as the total number of lectures involving class c_i and teachers in T (de Werra, 1985)

The construction of the timetable class after class results in a heuristic procedure. In a similar way one could consider a procedure where the timetable would be constructed for each teacher consecutively. More generally methods have been devised which consist in consecutively assigning lectures to some periods while trying to keep satisfied some necessary conditions of existence of solutions for the remaining problem. (de Werra, 1985)

The process is continued until either all lectures have been assigned or until no more can be assigned and some lectures are left unassigned. At this point, some methods will stop and some others will cancel some of the last assignments and start again at this stage. (de Werra, 1985).

The contribution of Chahal and de Werra (1989) is an example to the network flow approach. Their model is based on a simple network and a timetable is constructed by determining a collection of path packings in appropriate networks.

Another heuristic approach is the global approach that has been proposed by Aubin and Ferland (1989). They formulate each subproblem as an assignment problem and use a procedure that successively modifies the timetable and the grouping until no more improvement of the objective function is obtained.

The solution method in the model of Aubin and Ferland (1989) is a heuristic procedure where, at each iteration, one or two courses (resp. students) are reassigned to different periods of the week. (resp. course sections). Such a model delivers a local (but generally not global) optimum.

Aubin and Ferland (1989) first generate an initial timetable with an assignment of the students to the course sections; then an iterative procedure is used which adjusts the timetable and the grouping successively until no more improvement of the objective function can be obtained. At each iteration two procedures are used:

- given the grouping generating during the preceding iteration, the timetable is modified to reduce the number of conflicts,
- with this new timetable, the grouping of students is modified to reduce the number of conflicts.

Such an iterative descent procedure which Aubin and Ferland (1989) use for handling each of these two subproblems stops when a local minimum is reached.

b. *Simulated annealing (SA)* : Annealing is a thermal process for obtaining low energy states of a solid in a heat bath. The process contains two steps: first, increase the temperature of the heat bath to a maximum value at which the solid melts; second, decrease carefully the temperature of the heat bath until the particles arrange themselves in the ground state of the solid. Basically SA is a local search method, in which one wishes to choose a direction to move. Rather than always choosing the direction of best improvement, which gives steepest ascent hill-climbing, SA initially chooses random or semi random direction but over time comes to prefer any direction of best improvement. Thus the direction selection process is controlled by some sort of temporal parameter, which by analogy with real annealing is usually called 'temperature' (Fang, 1994).

There exist many SA applications in the Operations Research (OR) literature. For example, Dowsland (1990) considers timetabling problems that have no valid solution and therefore need to violate some constraints. She suggests three models to the problem, graph coloring, set partitioning, and simulated annealing (SA), and discusses the advantages and disadvantages of using each of these to find a satisfactory solution.

Abramson (1991) applied simulated annealing (SA) to a school timetabling problem using both sequential and parallel machines.

Wright (2001) reports on experiments with 'subcost-guided' heuristic search procedures, which are incorporated into a simulated annealing method and a threshold acceptance method, for some modified school timetabling problems where several subobjectives are combined into a single overall objective.

c. *Tabu Search (TS)* : Tabu search is a metaheuristic designed for getting a global optimum to a combinatorial optimization problem. Tabu search techniques have been especially designed for avoiding being trapped in local minima and are generally much more powerful than descent methods (Hertz, 1991).

Hertz (1991) produced a tabu search based timetable but only deals with restricted soft constraints.

Schaerf (1996) describes a solution algorithm and its implementation based on tabu search. His algorithm interleaves different types of moves and makes use of an adaptive relaxation of the hard constraints.

Alvarez-Valdes et al. (1996) developed an algorithm with three phases. In the second phase of the algorithm, a tabu search procedure starting from the solution of the first phase obtains a feasible solution to the problem.

Wright (1996) presents a solution method to school timetabling which involves four phases of heuristic search with little or no manual intervention necessary. In the first phase, a form of tabu search is used in which the subcosts guide the search.

d. *Genetic Algorithms (GAs)* : Fang (1994) investigates the application of genetic algorithms (GAs) to timetabling/scheduling problems. According to his definition; genetic algorithms (GAs) are a group of methods that solve problems using algorithms inspired by the processes of neo-Darwinian evolutionary theory. In a GA, the performance of a set candidate solutions to a problem (called 'chromosomes') are evaluated and ordered, then new candidate solutions are produced by selecting candidates as 'parents' and applying mutation or crossover operators which combine bits of two parents to produce one or more children. The new set of candidates is then evaluated, and this cycle continues until an adequate solution is found.

Genetic algorithms provide a way of separating the optimization algorithm from domain knowledge which has led to promising and useful results in the areas of timetabling/scheduling, as well as several other areas. The trick genetic algorithms use is to separate a roughly domain independent and very powerful optimization technique (mutation and crossover operators applied to populations of chromosomes) from the domain-specific aspects of a problem (the evaluation function for the chromosomes).

2.4.3. Artificial intelligence approaches

Among artificial intelligence approaches that have been proposed in the literature for the timetabling/scheduling problems, Fang (1994) lists the following ones. The discussion on these is beyond the scope of this study.

a. Constraint satisfaction problem (CSP)

b. Knowledge-based systems

c. Distributed artificial intelligence

d. Rule-based expert systems

e. Neural networks

3. TIMETABLING PROBLEM AT TURKISH MILITARY ACADEMY (TMA)

Turkish Military Academy (TMA) is a military institution where cadets have a four-year academic and military education. Its mission is to educate, train and inspire the Corps of Cadets so that each graduate is a commissioned leader of character committed to the values Honor, Honesty, Affection, Duty, Country; professional growth throughout a career as an officer in the Turkish Army; and a lifetime of selfless service to the nation.

The education at Turkish Military Academy (TMA) lasts for four years. Daily activities of the students are carefully regimented and balanced by considering the academic, military and physical requirements. Activities of each of these requirements are scheduled by the special committees within their own departments.

In this study, the scheduling of the academic education will be considered.

At TMA, the cadets are divided into 24 groups each year. These 24 groups are allocated into 4 different buildings equally. The cadet groups of the other years' classes are allocated to the buildings in the same manner. So, there are 6 cadet groups from each of the four years in every building.

The weekly timetable of TMA provides for each cadet group a sequence of courses to be taught every day of the week. The program for each year is fixed and all cadet groups at each year, except the fourth year, have the same courses. In other words, there are no elective courses in the first three years. Cadet groups of the fourth year are split and merged again with others according to the military branches of the cadets in order to form new groups for specific branch courses. This is done simultaneously and the number of groups remains the same. For this reason, this situation does not modify the structure of the timetable.

The timetabling problem at TMA is difficult due to the complexities of the Academy' s curriculum structure and the various constraints and objectives. At the time being, the construction of timetables is being performed manually in TMA. Timetablers of TMA work for several days in order to manually solve the problem, but a satisfactory solution taking account of the availabilities and the preferences of the teachers is rarely achieved.

3.1. Definition of some concepts

At this point, some definitions necessary for the problem formulation will be presented:

a. *Section* : Group of cadets which follow the same program. As described above, there is only one academic program for all sections in a year except for the fourth year sections. There are no elective courses offered in the first three years.

b. Class : Group of sections of the same year at TMA. The 24 sections of the same year form one of the four classes of TMA. Classes are named according to the years; for example, group of the first-year sections is called the first class.

c. *Week* : The time interval of five days (Monday, Tuesday, Wednesday, Thursday, Friday) for the scheduling cycle.

d. *Period* : Smallest unit of time in which the teaching activity takes place.

e. *Course* : Subject to be taught (for example Differential Equations). There are 8 courses for the second class cadets.

f. *Lecture* : Teaching unit consisting of a course taught during one period (for example one period of Differential Equations).

g. *Frequency* : The number of lectures that have to be taken (given) by a section (teacher) in a week.

h. *Curriculum* : The program or the list of courses to be taught for each section with their frequencies.

i. *Availability* : A Boolean days-periods matrix with value 'true' corresponding to days and periods when a section or a teacher is available.

j. *Compactness* : The condition that there is no vacancy between lectures, except for ten-minute breaks and the lunch break. Depending on whether compactness requirements are satisfied for classes or teachers, timetable is called 'section-compact' or 'teacher-compact'. If the requirements are satisfied for both, then the timetable is called as 'compact'. In the problem considered here, the timetable is 'section-compact' because the sections are busy at each period with academic courses or with other activities.

3.2. Formulation of the model

The main objective of this study is to develop a mathematical programming model for the timetabling problem at TMA, by considering the preferences of the teachers. In this chapter, an integer programming model for the timetabling problem in Turkish Military Academy (TMA) will be presented. The model will aim at minimizing the total penalty cost of the meetings (assignments) between the teachers of the courses and the sections in all periods. The penalty costs are derived by considering the preference levels of the teachers. By the help of this model, it is aimed at the efforts and the time spent for timetabling to be decreased.

Since the problem for all classes of TMA as a whole is too large to deal with, the timetabling problem will be restricted with only one class. But, the model can be iteratively applicable to the other classes by changing the data and applying little modifications. Having decided to utilize integer programming approach for building the timetabling problem, a thorough and detailed set of variables has been used for the representation of the problem. This allows for the modelling of a significant number of rules and regulations and provides the indirect satisfaction of a large set of quality rules. The detailed and multidimensional nature of the variables allows for building an efficient and compact model.

In building the model; a set of sections, a set of courses where each course has a set of teachers, and a set of days with a set of periods have to be considered. The curriculum of a class at TMA is fixed for a given section and the timetabling problem involves scheduling of the various courses as frequently as needed in a weekly schedule, in accordance with the particular educational system rules of TMA.

The educational system rules to be considered in the model are listed below:

a. Each section gets all courses in its curriculum with the appropriate frequency.

b. Every teacher gives all courses in his curriculum with the appropriate frequency.

c. A combination involving a teacher is not scheduled at a period in which that teacher is not available. There are some part-time teachers coming from the other universities. Their available periods are limited with the ones that they have declared. So, the assignment will be done to these periods. The penalty costs of the periods for part-time teachers are set to zero.

d. The activities other than the academic courses are scheduled first by the responsible departments. Thus, the periods of each section assigned to these activities are regarded as unavailable periods for academic scheduling. As a result, a combination involving a section is not scheduled at a period in which that section is not available. There are 11 periods in a week that are not available for each section and this leaves 19 available periods per week.

e. The teachers have to be assigned to sections in the sense that a given course will be taught to a given section by the same teacher. In other words, a section should take all lectures of a course from the same teacher.

f. There are 6 periods in a day making a total of 30 in a week. The first four of the six periods in the day take place before lunch, and the last two are in the afternoon.

g. The frequencies of the courses in the problem are 1, 2, 3, and 4 periods. For a given section, a two-lecture course must be scheduled at consecutive periods on the same day as a block. Similarly, a four-lecture course has to be given to each section on two different days, and at two consecutive periods on each. Three-lecture courses have an option. They can be given to the sections either a three-consecutive periods block on one day, or as a two-consecutive periods block in a day plus one period on some other day.

h. A two-consecutive periods block of a course cannot be divided by a lunch break. That is, the second period of a block cannot be in the afternoon while the first one takes place before lunch. This rule is invalid for three-period blocks.

i. The maximum number of periods assigned to a teacher must be less than or equal to 20. The remaining periods are left free for the research activities, projects, and other extra duties of the teacher.

j. There are some teachers who attend an academic education program in the other universities and also some teachers who have extra duties rather than teaching a course. The penalty cost for assigning to the periods which they need for their academic education or their extra duties will be ranked as "high". On the other hand, the penalty costs for the periods other than the ones needed for academic education or extra duties should be ranked as "low" or "medium" according to the preferences of the teachers.

The following are assumptions made to simplify the problem and make it solvable:

a. Teachers have already been assigned to the courses and the number of courses they have been assigned to is only one.

b. All the teachers will be assigned to the sections of only one class at TMA. If some teacher is given the courses of different classes, then the periods he is assigned to one class will be unavailable periods of him for the courses of the other classes.

c. It is assumed that there are enough teaching rooms available in total to accommodate all sections. But indeed, there exist inevitable problems caused by the use of special-purpose rooms such as laboratories or military workshops.

d. Some courses such as 'Foreign Language' are scheduled by their responsible departments. These courses will be regarded as 'other activities'.

e. Preferences of the sections, such as the undesired situation of having more than one hard and difficult course on the same day, are neglected.

f. Assignment to sections which are located in different buildings in consecutive periods, except for the ones that have lunch break in between, is not preferable for the teachers. These quality constraints have been neglected in the model, for keeping the model size in reasonable dimensions.

g. It is assumed that the data provided as input to the model will not change during the semester.

3.3. Model

3.3.1. Decision variables

Before attempting any timetabling optimization all the decision variables must be considered. The following sets have been considered as the basic structural elements for the model:

- Section = {Sec1, Sec2,..................., SecI}
- Course = {Course1, Course2, ..., CourseJ}
- Teacher = {Name1, Name2, ..., Name K}
- Day = {Mon., Tue., Wed., Thu., Fri.}
- Period = {Per1, Per2, ..., PerM}

A set of values for these five structural elements that covers every period of a week for each section defines the timetable of a class at TMA. It is assumed that the value sets of the structural elements Section, Course, Teacher, Day, and Period are finite and strictly ordered. This structure is quite generic and can be applied to a number of timetabling and scheduling problems. The number of combinations between the values of the elements and the resulting complexity of the problem depend on the size of the value set of each structural element.

Since there is an assumption that teachers have already been assigned to courses and the number of courses they have been assigned to is only one; the subsets of the element Teacher, formed by grouping the teachers giving the same subjects may represent the element Course. Thus, the number of the elements needed for the formulation of the model can be reduced to four.

The formulation of this reduced structure as a mathematical model is achieved by introducing variables, $y_{a,b}$, $v_{a,b,c}$, and $x_{a,b,c,d}$ with the indices *a, b, c, d.* Each index of the variables corresponds to a different structural element. Thus, the index *a* takes values from the set *Section*, index *b* from the set *Teacher*, index *c* from the set *Day*, and index *d* from the set *Period.* However, since it is necessary for the indices to be natural numbers, the value set of each element is required to correspond to a finite subset of the natural numbers. Using a one-to-one mapping between the value set of each basic structural element and a subset of the natural numbers, the variables $y_{a,b}$, $v_{a,b,c}$, and $x_{a,b,c,d}$ are transformed to the variables $y_{s,t}$, $v_{s,t,d}$, and $x_{s,t,d,p}$ respectively, where

- $s \in S = \{1, 2, \ldots\ldots, ss\}$,
- $t \in T = \{1, 2, \ldots\ldots, tt\}$,

- $d \in D = \{1, 2, \ldots\ldots, dd\}$,
- $p \in P = \{1, 2, \ldots\ldots, pp\}$,

with *ss, tt, dd,* and *pp* corresponding to the cardinalities of the sets *Section, Teacher, Day, Period.*

Note that, the elements of each subset of the set T teaching the same course, let us call $T1$, $T2$, ..., TJ, represent the elements of the set Course = {Course1, Course2, ..., CourseJ} respectively. Thus, the subsets of the set T corresponding to courses can be shown as;

$t \in T1 = \{1, 2, \ldots\ldots, tt1\}$, corresponding to Course1,

$t \in T2 = \{1, 2, \ldots\ldots, tt2\}$, corresponding to Course2,

$\vdots$

$t \in TJ = \{1, 2, \ldots\ldots, ttJ\}$, corresponding to CourseJ,

where $T = T1 \cup T2 \cup \ldots\ldots \cup TJ$.

With the indices given above, we can define the decision variables $y_{s,t}$, $v_{s,t,d}$, and $x_{s,t,d,p}$:

- $y_{s,t}$: A binary variable providing the assignment of teachers to sections, independently from the days and the periods.

$$y_{s,t} = \begin{cases} 1 & \text{if } teacher\ t \text{ is assigned to } section\ s, \\ 0 & \text{otherwise.} \end{cases}$$

- $v_{s,t,d}$: A binary variable indicating the day on which a meet between a teacher and a section takes place.

$$v_{s,t,d} = \begin{cases} 1 & \text{if } section\ s \text{ and } teacher\ t \text{ meet on } day\ d, \\ 0 & \text{otherwise.} \end{cases}$$

- $x_{s,t,d,p}$: A binary variable indicating the period at which a meet between a teacher and a section takes place.

$$x_{s,t,d,p,} = \begin{cases} 1 \text{ if } \textit{section s} \text{ and } \textit{teacher t} \text{ meet at } \textit{period p} \text{ of } \textit{day d,} \\ 0 \text{ otherwise.} \end{cases}$$

In general the total number of each variable set for the timetabling problem equals the product of the cardinalities of the index sets they have. For instance, the total number of variables $x_{s,t,d,p,}$ equals the product of the cardinalities of the sets, *S, T, D, P*.

3.3.2. Constraints

The detailed nature of the variables $y_{s,t,}$ $v_{s,t,d,}$ and $x_{s,t,d,p}$ allows for a convenient way to represent all constraints and sensitivities of the timetabling problem. The model may then be constructed using an integer programming approach, since all variables are binary and take the values of one or zero. For instance, when the value of a variable of $x_{s,t,d,p}$ is equal to 1, then the corresponding combination of *section s, teacher t, day d* and *period p* is included in the final solution. Otherwise, *teacher t* will not be assigned to *section s* at *period p* of *day d.*

Constraint Set 1: At most one teacher can be assigned to *section s* at a time (at *period p* of *day d*). This constraint set also provides that there will be no assignment of *section s* to *period p* of *day d* if it is unavailable. Let $sav_{s,d,p}$ be a Boolean days-periods matrix representing the section availability;

$$\sum_{t \in T} x_{s,t,d,p} \leq sav_{s,d,p} \qquad \forall \ s \in S, d \in D, p \in P \qquad (3.1)$$

where $sav_{s,d,p} = \begin{cases} 1 & \text{if } \textit{section s} \text{ is available at } \textit{period p} \text{ of } \textit{day d,} \\ 0 & \text{if } \textit{section s} \text{ is unavailable because of its activities rather than academic education..} \end{cases}$

Constraint Set 2: Teacher t can be assigned to at most one section at a time. (at *period p* of *day d*). This constraint set also provides that there will be no assignment of *teacher t* to *period p* of *day d* if he is unavailable. Let $tav_{t,d,p}$ be a Boolean days-periods matrix representing the teacher availability;

$$\sum_{s \in S} x_{s,t,d,p} \leq tav_{t,d,p} \qquad \forall\ t \in T,\ d \in D,\ p \in P \qquad (3.2)$$

where $tav_{t,d,p} = \begin{cases} 1 & \text{if } \textit{teacher } t \text{ is available at } \textit{period } p \text{ of } \textit{day } d, \\ 0 & \text{otherwise.} \end{cases}$

Constraint Set 3: This set of constraints ensures that, for each course, at most one teacher will be assigned to *section s.* That is, all lectures of a course will be taught by the same teacher.

$$\left.\begin{aligned} \sum_{t \in T1} y_{s,t} &= 1 \\ \sum_{t \in T2} y_{s,t} &= 1 \\ &\vdots \\ \sum_{t \in TJ} y_{s,t} &= 1 \end{aligned}\right\} \qquad \forall\ s \in S \qquad (3.3)$$

where *T1* is the group of teachers teaching *Course1, T2* is the group of teachers teaching *Course2,* and so on...

Constraint Set 4: This set of constraints ensures that any meet between a section and a teacher will not take place on any day if that teacher is not assigned to that section. Otherwise, there is no limitation.

$$v_{s,t,d} \leq y_{s,t} \qquad \forall\ s \in S,\ t \in T,\ d \in D \qquad (3.4)$$

Constraint Set 5: This set of constraints ensures that any meet between *section s* and *teacher t* will not take place on any period if *teacher t* is not assigned to *section s* on *day d*. Otherwise, there is no limitation.

$$x_{s,t,d,p} \leq v_{s,t,d} \qquad \forall\ s \in S,\ t \in T,\ d \in D,\ p \in P \qquad (3.5)$$

Constraint Set 6: This set of constraints limits the weekly workload for a teacher.

$$\sum_{s \in S} \sum_{d \in D} \sum_{p \in P} x_{s,t,d,p} \leq wl_t \qquad \forall\, t \in T \qquad (3.6)$$

where wl_t is maximum total number of periods that a teacher can be assigned to in a week.

Constraint Set 7: These constraints ensure that the teaching hours of *teacher t* in each section must satisfy the teaching requirements from that teacher (the total number of lectures in a week that a teacher has to give in a section).

$$\sum_{d \in D} \sum_{p \in P} x_{s,t,d,p} = h_t * y_{s,t} \qquad \forall\, s \in S,\ t \in T \qquad (3.7)$$

where h_t is the total number of lectures that *teacher t* has to give in a section.

Constraint Set 8: These sets of constraints ensure that two-lecture and four-lecture courses should be given in two-period blocks to each section. As explained previously, because of the assumption that every teacher can teach only one course in a class, courses are represented by teachers.

Let T_{four} be the combination of teacher groups of four-lecture courses and T_{two} be the combination of teacher groups of two-lecture courses.

a. Constraints that ensure the assignment of *section s* to *teacher t* of a four-lecture course on at most two days:

$$\sum_{d \in D} v_{s,t,d} \leq 2 \qquad \forall\, s \in S,\ t \in T_{four} \qquad (3.8)$$

b. Constraints that ensure the assignment of *section s* to *teacher t* of a two-lecture course on at most one day:

$$\sum_{d \in D} v_{s,t,d} \leq 1 \qquad \forall\, s \in S,\ t \in T_{two} \qquad (3.9)$$

c. Constraints that ensure the assignment of *section s* to *teacher t* of two-lecture or four-lecture courses at two periods at most on *day d*:

$$\sum_{p=P} x_{s,t,d,p} \leq 2 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.10)$$

d. These constraints ensure that the periods of a course block that *teacher t* gives in *section s* be consecutive. At this point, the lunch break is taken into account that it can not take place between two periods of the block. If it is assumed that, for example, there are 6 periods in a day then the relationship between them for the consecutiveness of the lectures is provided by:

$$x_{s,t,d,'1'} - x_{s,t,d,'2'} \leq 0 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.11)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'3'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.12)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'4'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.13)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'5'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.14)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'6'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.15)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'4'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.16)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'5'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.17)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'6'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.18)$$

$$x_{s,t,d,'3'} + x_{s,t,d,'5'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.19)$$

$$x_{s,t,d,'3'} + x_{s,t,d,'6'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.20)$$

$$x_{s,t,d,'4'} + x_{s,t,d,'5'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.21)$$

$$x_{s,t,d,'4'} + x_{s,t,d,'6'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.22)$$

$$x_{s,t,d,'5'} - x_{s,t,d,'6'} = 0 \qquad \forall\, s \in S,\ t \in T_{four}, T_{two},\ d \in D \qquad (3.23)$$

Constraint Set 9: These sets of constraints ensure that three-lecture courses should be given to each section either in three-period blocks on just one day, or in two-period block on a day and one period on another day. As explained previously, because of the assumption that every teacher can teach only one course in a class, courses are represented by teachers.

Let T_{three} be the combination of teacher groups of three-lecture courses.

a. Constraints that ensure the assignment of *section s* to *teacher t* of a three-lecture course on at most two days:

$$\sum_{d \in D} v_{s,t,d} \leq 2 \qquad \forall\, s \in S,\ t \in T_{three} \qquad (3.24)$$

b. These constraints ensure that the periods of a course block that *teacher t* gives in *section s* be consecutive. At this point, the lunch break is taken into account that it can not take place between two periods of a two period block. If the course is given in three-period block, then there are no rules about the presence of lunch break among the periods. If it is assumed that, for example, there are 6 periods in a day then the relationship between them for the consecutiveness of the lectures is provided by:

$$x_{s,t,d,'1'} - x_{s,t,d,'2'} + x_{s,t,d,'3'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{three},\ d \in D \qquad (3.25)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'4'} \leq 1 \qquad \forall\, s \in S\ t \in T_{three},\ d \in D \qquad (3.26)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'5'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{three},\ d \in D \qquad (3.27)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'6'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{three},\ d \in D \qquad (3.28)$$

$$x_{s,t,d,'2'} - x_{s,t,d,'3'} + x_{s,t,d,'4'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{three},\ d \in D \qquad (3.29)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'5'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{three},\ d \in D \qquad (3.30)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'6'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{three},\ d \in D \qquad (3.31)$$

$$x_{s,t,d,'3'} - x_{s,t,d,'4'} + x_{s,t,d,'5'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{three},\ d \in D \qquad (3.32)$$

$$x_{s,t,d,'3'} + x_{s,t,d,'6'} \leq 1 \qquad \forall\, s \in S,\ t \in T_{three},\ d \in D \qquad (3.33)$$

$$x_{s,t,d,'4'} - x_{s,t,d,'5'} + x_{s,t,d,'6'} \leq 1 \qquad \forall s \in S, t \in T_{three}, d \in D \qquad (3.34)$$

$$-x_{s,t,d,'3'} + x_{s,t,d,'4'} + x_{s,t,d,'5'} - x_{s,t,d,'6'} \leq 1 \qquad \forall s \in S, t \in T_{three}, d \in D \qquad (3.35)$$

3.3.3. Objective function

Given that some combinations of the indices *s, t, d,* and *p* are preferable to others, the objective function for the timetabling model is defined as follows:

$$\text{Min} \sum_{s \in S} \sum_{t \in T} \sum_{d \in D} \sum_{p \in P} pm_{t,d,p} * x_{s,t,d,p} \qquad (3.36)$$

where $pm_{t,d,p}$ is a set of penalty cost coefficients. These coefficients are designed to penalize their corresponding variables in order to reflect positive or negative preferences. For example, if it is desired that a lecture of the course of teacher 1 is to be placed at the first period of the first day, then a relatively low value should be assigned to $pm_{1,1,1}$ coefficient.

The determination of the penalty cost coefficients $pm_{t,d,p}$ has been performed as follows:

- The available periods of the part-time teachers coming from the other universities will get no penalty and have a cost coefficient with value '0' in order to provide them a higher priority with respect to full-time teachers.
- The available periods of the full-time teachers will have a penalty cost coefficient with value 'low' if they highly prefer to be assigned at that period.
- The available periods of the full-time teachers will have a penalty cost coefficient with value 'medium' if they moderately prefer to be assigned at that period.
- The available periods of the full-time teachers will have a penalty cost coefficient with value 'high' if they do not prefer to be assigned, or if they have any extra duty at that period, or if they need that time for their own academic education. The penalty cost coefficient with value 'high' prevents but does not prohibit the presence of its corresponding variable in the final solution.

4. IMPLEMENTATION OF THE TIMETABLING MODEL AT TURKISH MILITARY ACADEMY (TMA)

In part three, a model for the timetabling problem of TMA was presented. But the timetabling problem will be restricted with only one class for the reason that the problem for all classes of TMA as a whole is too large to deal with. The model presented in the previous part can be applied iteratively to the other classes by modifying the data and making little modifications.

In this part of the thesis an application of the timetabling model for the second-year cadets of Turkish Military Academy (TMA) is presented. In this presentation the related data sets and model parameters are described first. Then, the model just presented is adopted for the second-class of TMA. The hard and quality rules and the assumptions described in the previous part are valid for this application as well. Later, the timetabling model for the second class of TMA is solved with the presented methodology and the results obtained from the solution process are discussed.

4.1. The Data for the Problem

The Planning and Programming Department of TMA provided the data used for the application of the model. They are mainly based on the real 2000-2001 data for the second class of TMA. The teacher preference matrix is formed after applying little arbitrary changes to the information obtained from an interview with that department.

At TMA, the second class consists of 24 sections (the same with the others) and is staffed with the necessary specialized personnel that will cover the total number of periods in the scheduling cycle. The scheduling cycle is a week and is composed of the days Monday, Tuesday, Wednesday, Thursday, and Friday. Each section is required to attend 30 periods (six on each day) in a scheduling cycle.

The number of courses that each section in the second class has to undertake is 8. Military training, physical training, and some courses such as foreign language, which are scheduled by the other departments, are not included in that number. All the courses are compulsory, i.e. there are no electives. The assignment of the lectures of a course to the days and periods of a week is subject to the constraints described in Part 3.2.

The total number of periods where a section is unavailable because of the other activities, such as military training, physical training, and courses of leadership and foreign language is 11. Table 4.1 and Table 4.2 display the unavailable periods of each section of the second class.

Table 4.1 Unavailable periods of sections occupied by military and physical training and foreign language course.

Sections (s)	Days (d)														
	1			2			3			4			5		
	Periods (p)			Periods (p)			Periods (p)			Periods (p)			Periods (p)		
	1-2	3-4	5-6	1-2	3-4	5-6	1-2	3-4	5-6	1-2	3-4	5-6	1-2	3-4	5-6
1-3				F.L.		P.T					F.L.	M.T			
4-6				F.L.				P.T			F.L.	M.T			
7-9			M.T	F.L.	P.T.						F.L.				
10-12			M.T	F.L.					P.T		F.L.				
13-15			P.T			M.T				F.L.			F.L.		
16-18						M.T				F.L.	P.T		F.L.		
19-21									M.T	F.L.		P.T	F.L.		
22-24		P.T							M.T	F.L.			F.L.		

Note : F.L.:Foreign Language, M.T:Military Training, P.T.:Physical Training

Table 4.2 Unavailable periods of sections occupied by the course of leadership.

Sections	1-4	5-6	7-10	11-12	13-14	15-18	19-22	23-24
Unavailable periods (day.period)	1.1 1.2 1.3	2.4 2.5 2.6	3.1 3.2 3.3	2.4 2.5 2.6	4.4 4.5 4.6	3.4 3.5 3.6	2.1 2.2 2.3	4.4 4.5 4.6

The assignment of courses to the specialized teachers of TMA is achieved by applying the specific rules referring to each teacher's specialty, which is beyond the scope of this study. Every teacher is required to undertake at most 20 teaching hours (periods) for courses related to his/her specialty.

Frequencies of the courses and the number of teachers previously assigned to them are shown in Table 4.3.

Table 4.3 Frequencies of the courses and the number of teachers previously assigned.

Courses	1	2	3	4	5	6	7	8
Frequency	4	2	2	2	2	3	3	1
# of Teachers Assigned	8	3	6	7	5	8	6	3

As explained in Part 3.2., there are some part-time teachers coming from the other universities. Their available periods are limited with the ones that they have declared. The teacher availability matrix for the courses is shown in Table 4.4.

Table 4.4 Unavailable periods of teachers.

X : Unavailable periods		DAYS (d)																													
		1						2						3						4						5					
Course	Teachers	Periods (p)						Periods (p)						Periods (p)						Periods (p)						Periods (p)					
(c)	(t)	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
1	1																														
	2																														
	3																														
	4																														
	5																														
	6	X	X			X	X	X	X			X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	7	X	X	X	X	X	X	X	X			X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X
	8	X	X			X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	X	X	X	X	X
2	9																														
	10																														
	11																														
3	12																														
	13																														
	14																														
	15																														
	16																														
	17																														
4	18																														
	19																														
	20	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X						
	21							X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	22																														
	23																														
	24																														
5	25																														
	26																														
	27																														
	28																														
	29	X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X	X	X
6	30																														
	31																														
	32																														
	33																														
	34																														
	35																														
	36	X	X	X	X	X	X	X				X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	37	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X				X	X
7	38	X	X	X	X	X	X	X	X	X	X	X	X							X	X	X	X	X	X	X	X	X	X	X	X
	39																														
	40																														
	41																														
	42																														
	43																														
8	44																														
	45																														
	46																														

Table 4.5 The penalty cost matrix for cost coefficient values.

X : Unavail -able periods		DAYS (d)																													
		1						2						3						4						5					
		Periods (p)						Periods (p)						Periods (p)						Periods (p)						Periods (p)					
C:	T:	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
1	1	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
	2	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2	4	4
	3	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	5	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	6	X	X	0	0	X	X	X	X	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	7	X	X	X	X	X	X	X	X	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	0	X	X
	8	X	X	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	0	X	X	X	X	X	X	X	X
2	9	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	2	2	2	2	2	2
	10	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	11	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	12	4	4	4	4	4	4	4	4	4	4	4	4	2	2	2	2	2	2	4	4	4	4	4	4	2	2	2	2	2	2
	13	4	4	4	4	4	4	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	14	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	2	2	2	2	2	2	6	6	6	6	6	6
	15	2	2	2	2	2	2	4	4	4	4	4	4	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2
	16	6	6	2	2	2	2	6	6	2	2	2	2	6	6	2	2	2	2	6	6	2	2	2	2	6	6	2	2	2	2
	17	2	2	2	2	2	2	4	4	4	4	4	4	4	4	4	4	4	4	2	2	2	2	2	2	6	6	6	6	6	6
4	18	4	4	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2
	19	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2
	20	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0	0	0
	21	0	0	0	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	22	6	6	6	6	2	2	2	2	2	2	2	2	4	4	4	4	4	4	6	6	6	6	6	6	2	2	2	2	2	2
	23	2	2	2	2	6	6	2	2	2	2	6	6	2	2	2	2	6	6	2	2	2	2	6	6	2	2	2	2	6	6
	24	4	4	2	2	2	2	6	6	6	6	6	6	6	6	6	6	6	6	4	4	4	4	4	4	4	4	2	2	2	2

Table 4.5 (continued)

X : Unavail -able periods		DAYS (d)																													
		1						2						3						4						5					
		Periods (p)						Periods (p)						Periods (p)						Periods (p)						Periods (p)					
C:	T:	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
5	25	2	2	2	2	2	2	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	2	2	2	2	2	2
	26	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4
	27	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	6	6	6
	28	2	2	2	2	2	2	2	2	2	2	2	2	6	6	6	6	6	6	6	6	6	6	6	6	2	2	2	2	2	2
	29	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X
6	30	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	31	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	6	6	6	6	6	6
	32	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2
	33	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	6	6	6	6	6	6
	34	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	6	6	6	6	6	6
	35	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	4	4	6	6	6	2	2	2	2	2	2	6	6	6
	36	X	X	X	X	X	X	X	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	37	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	X	X
7	38	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X
	39	2	2	2	2	2	2	4	4	4	6	6	6	6	6	6	2	2	2	2	2	2	2	2	2	6	6	6	2	2	2
	40	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2
	41	2	2	2	2	2	2	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	42	2	2	2	2	2	2	6	6	6	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	2	2	2	2	2	2
	43	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
8	44	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	45	4	4	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2	4	4	2	2	2	2
	46	4	4	4	4	4	4	4	4	4	4	4	4	2	2	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4

The cost coefficients $pm_{t,d,p}$ in the objective function will take on the associated values from the penalty matrix of Table 4.5 given above. This matrix reflects the preference levels of the teachers for the various periods of the week. The values of the cost coefficients are determined as follows:

- $pm_{t,d,p} = 0$, for the available periods of part-time teachers coming from the other universities in order to provide them a priority with respect to full-time teachers.
- $pm_{t,d,p} = 2$, for highly preferred periods.
- $pm_{t,d,p} = 4$, for moderately preferred periods.
- $pm_{t,d,p} = 6$, for the periods which are not preferred, or the teachers have any extra duty, or they need that time for their own academic education.

The maximum number of periods that are going to be assigned to a teacher is 20. The remaining periods are left free for the research activities, projects, and other extra duties of the teachers.

4.2. The Timetabling Model Formulation

a. Indices

$s \in S = \{1, 2, ..., 24\}$: The set of sections in a class. Each number in the set corresponds to a section.

$t \in T = \{1, 2, ..., 46\}$: The set of all teachers previously assigned to teach the sections of the second class. Each number in the set corresponds to the name of a teacher. Subsets of T are $T1, T2, T3, T4, T5, T6, T7, T8$.

$d \in D = \{1, 2, ..., 5\}$: The set of days whose elements correspond to Monday, Tuesday, Wednesday, Thursday, and Friday respectively.

$p \in P = \{1, 2, ..., 6\}$: The set of periods on a day.

where,

$t \in T1 = \{1, 2, ..., 8\}$ represents Course 1 with a frequency of 4.

$t \in T2 = \{9, 10, 11\}$ represents Course 2 with a frequency of 2.

$t \in T3 = \{12, 13 ...., 17\}$ represents Course 3 with a frequency of 2.

$t \in T4 = \{18, 19, ..., 24\}$ represents Course 4 with a frequency of 2.

$t \in T5 = \{25, 26, ..., 29\}$ represents Course 5 with a frequency of 2.

$t \in T6 = \{30, 31, ..., 37\}$ represents Course 6 with a frequency of 3.

$t \in T7 = \{38, 39, ..., 43\}$ represents Course 7 with a frequency of 3.

$t \in T8 = \{44, 45, 46\}$ represents Course 8 with a frequency of 1.

$T1, T2, ...,T8 \subset T$

b. Parameters

$sav_{s,d,p}$: A Boolean matrix for the availability of *section s* at *period p* of *day d.*

$sav_{s,d,p}$ = 1 if *section s* is available at *period p* of *day d,*
0 otherwise.

$tav_{t,d,p}$: A Boolean matrix for the availability of *teacher t* at *period p* of *day d.*

$tav_{t,d,p}$ = 1 if *teacher t* is available at *period p* of *day d,*
0 otherwise.

wl_t : Weekly load limit for *teacher t.*

wl_t = {20, 20, ...20} for the first solution stage.

h_t : Frequency of the course that *teacher t* gives. ($t \in T1, T2, T3, T4, T5, T6, T7, T8$)

h_t = {4, 4, ..., 4, 2, 2, ..., 2, 3, 3, ..., 3, 1, 1, ..., 1}

$pm_{t,d,p}$: A penalty cost matrix for the preference of *teacher t* at *period p* of *day d.* (see Table 4.5)

c. Binary variables

$y_{s,t}$: A binary variable providing the assignment of teachers to sections, independently from the days and the periods.

$$y_{s,t} = \begin{cases} 1 & \text{if } \textit{teacher } t \text{ is assigned to } \textit{section } s, \\ 0 & \text{otherwise.} \end{cases}$$

$v_{s,t,d}$: A binary variable indicating the day on which a meet between a teacher and a section takes place.

$$v_{s,t,d} = \begin{cases} 1 & \textit{section } s \text{ and } \textit{teacher } t \text{ meet on } \textit{day } d, \\ 0 & \text{otherwise.} \end{cases}$$

$x_{s,t,d,p}$: A binary variable indicating the period at which a meet between a teacher and a section takes place.

$$x_{s,t,d,p} = \begin{cases} 1 & \textit{section } s \text{ and } \textit{teacher } t \text{ meet at } \textit{period } p \text{ of } \textit{day } d, \\ 0 & \text{otherwise.} \end{cases}$$

d. Model formulation

$$\text{Min} \sum_{s=1}^{24} \sum_{t=1}^{46} \sum_{d=1}^{5} \sum_{p=1}^{6} pm_{t,d,p} * x_{s,t,d,p} \qquad (4.1)$$

Subject to;

$$\sum_{t=1}^{46} x_{s,t,d,p} \leq sav_{s,d,p} \qquad (s=1,...,24;\ d=1,...5;\ p=1,...,6) \qquad (4.2)$$

$$\sum_{s=1}^{24} x_{s,t,d,p} \leq tav_{t,d,p} \qquad (t=1,...,46\ d=1,...5;\ p=1,...,6) \qquad (4.3)$$

$$\sum_{t=1}^{8} y_{s,t} = 1 \qquad (s=1,...,24) \qquad (4.4)$$

$$\sum_{t=9}^{11} y_{s,t} = 1 \qquad (s=1,...,24) \qquad (4.5)$$

$$\sum_{t=12}^{17} y_{s,t} = 1 \qquad (s=1,...,24) \qquad (4.6)$$

$$\sum_{t=18}^{24} y_{s,t} = 1 \qquad (s=1,...,24) \qquad (4.7)$$

$$\sum_{t=25}^{29} y_{s,t} = 1 \qquad (s=1,...,24) \qquad (4.8)$$

$$\sum_{t=30}^{37} y_{s,t} = 1 \qquad (s=1,...,24) \qquad (4.9)$$

$$\sum_{t=38}^{43} y_{s,t} = 1 \qquad (s=1,...,24) \qquad (4.10)$$

$$\sum_{t=44}^{46} y_{s,t} = 1 \qquad (s=1,...,24) \qquad (4.11)$$

$$v_{s,t,d} \leq y_{s,t} \qquad (s=1,...,24;\ t=1,...,46;\ d=1,...,5) \qquad (4.12)$$

$$x_{s,t,d,p} \leq v_{s,t,d} \qquad (s=1,...,24;\ t=1,...,46;\ d=1,...,5;\ p=1,...,6) \qquad (4.13)$$

$$\sum_{s=1}^{24} \sum_{d=1}^{5} \sum_{p=1}^{6} x_{s,t,d,p} \leq wl_t \qquad (t=1,...,46) \qquad (4.14)$$

$$\sum_{d=1}^{5} \sum_{p=1}^{6} x_{s,t,d,p} = h_t * y_{s,t} \qquad (s=1,...,24;\ t=1,...,46) \qquad (4.15)$$

$$\sum_{d=1}^{5} v_{s,t,d} \leq 2 \qquad (s=1,...,24;\ t=1,...,8) \qquad (4.16)$$

$$\sum_{d=1}^{5} v_{s,t,d} \leq 1 \qquad (s=1,...,24;\ t=9,...,29) \qquad (4.17)$$

$$\sum_{p=1}^{6} x_{s,t,d,p} \leq 2 \qquad (s=1,...,24;\ t=1,...,29) \qquad (4.18)$$

$$x_{s,t,d,'1'} - x_{s,t,d,'2'} \leq 0 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.19)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'3'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.20)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'4'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.21)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'5'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.22)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'6'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.23)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'4'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.24)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'5'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.25)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'6'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.26)$$

$$x_{s,t,d,'3'} + x_{s,t,d,'5'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.27)$$

$$x_{s,t,d,'3'} + x_{s,t,d,'6'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.28)$$

$$x_{s,t,d,'4'} + x_{s,t,d,'5'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.29)$$

$$x_{s,t,d,'4'} + x_{s,t,d,'6'} \leq 1 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.30)$$

$$x_{s,t,d,'5'} - x_{s,t,d,'6'} = 0 \qquad (s=1,...,24;\ t=1,...,29;\ d=1,...,5;) \qquad (4.31)$$

$$\sum_{d=1}^{5} v_{s,t,d} \leq 2 \qquad (s=1,...,24;\ t=30,...,43) \qquad (4.32)$$

$$x_{s,t,d,'1'} - x_{s,t,d,'2'} + x_{s,t,d,'3'} \leq 1 \qquad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \qquad (4.33)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'4'} \leq 1 \qquad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \qquad (4.34)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'5'} \leq 1 \qquad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \qquad (4.35)$$

$$x_{s,t,d,'1'} + x_{s,t,d,'6'} \leq 1 \qquad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \qquad (4.36)$$

$$x_{s,t,d,'2'} - x_{s,t,d,'3'} + x_{s,t,d,'4'} \leq 1 \qquad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \qquad (4.37)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'5'} \leq 1 \quad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \quad (4.38)$$

$$x_{s,t,d,'2'} + x_{s,t,d,'6'} \leq 1 \quad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \quad (4.39)$$

$$x_{s,t,d,'3'} - x_{s,t,d,'4'} + x_{s,t,d,'5'} \leq 1 \quad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \quad (4.40)$$

$$x_{s,t,d,'3'} + x_{s,t,d,'6'} \leq 1 \quad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \quad (4.41)$$

$$x_{s,t,d,'4'} - x_{s,t,d,'5'} + x_{s,t,d,'6'} \leq 1 \quad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \quad (4.42)$$

$$-x_{s,t,d,'3'} + x_{s,t,d,'4'} + x_{s,t,d,'5'} - x_{s,t,d,'6'} \leq 1 \quad (s=1,...,24;\ t=30,...,43;\ d=1,...,5;) \quad (4.43)$$

4.3. Solution of the Model

After modelling this problem as a binary integer programming model, an iterative solution heuristic has been applied to the model in order to find a near-optimal solution. The reason is that the size of the model as applied to the second-class of TMA is still too large to be solved on a PC, with 39,745 binary variables and 110,316 constraints.

The solution process proceeds as shown in Figure 4.1. The characteristics of this process is explained below :

a. Grouping

First, group the 24 sections of the class into two parts. Each part contains 12 sections. Then, group the teachers of the courses into two parts. The teachers of the courses which have two-lecture frequency, i.e. teachers of Course 2, 3, 4, and 5, form the first teacher group. The natural numbers corresponding to these teachers are {9, 10, ..., 29}. Similarly, the teachers of the courses which have four, three, and one-lecture frequency, i.e. teachers of Course 1, 6, 7, and 8, form the second teacher group. The natural numbers corresponding to the second group of teachers are {1,.... ,8, 30,.....,46}.

b. Iteration-1

After grouping operations, take 'the <u>first</u> section group' and 'the <u>first</u> teacher group'. Develop the model again with these new sets. The size of the problem is now smaller, with 9,073 binary variables and 28,025 constraints.

Solve the model on a PC by using a Mixed Integer Programming (MIP) solver. In the solution output, the periods where variable $x_{s,t,d,p} = 1$ are the periods that a section and a teacher meet. Take these periods as the input data for the next iterations.

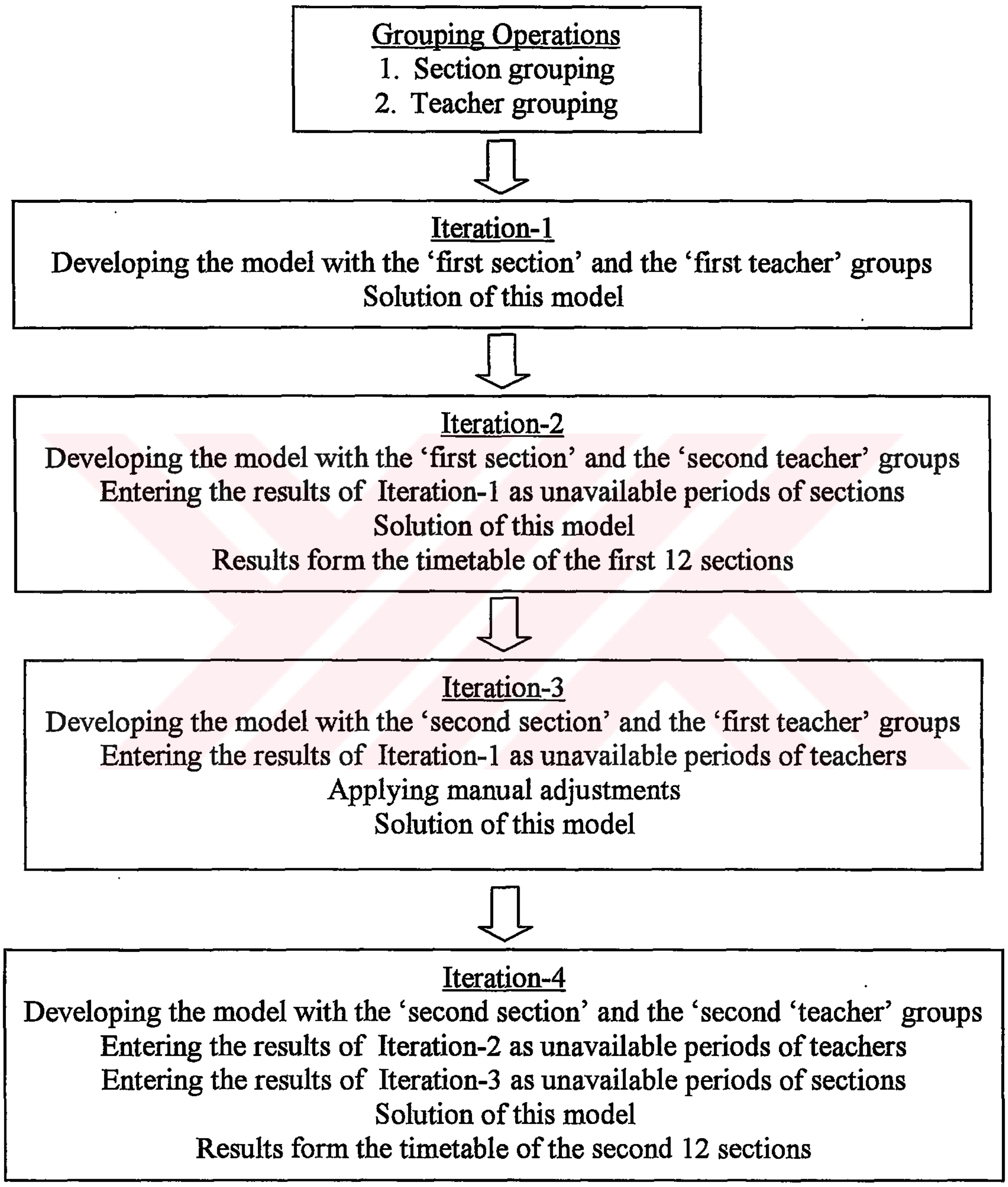

Figure 4.1 Flowchart of the solution process.

c. Iteration-2

Take 'the first section group' and 'the second teacher group' and develop the model again with these new sets. The size of the problem is small enough again, with 10,081 binary variables and 28,209 constraints.

Input the results found in Iteration-1; i.e. the periods where a section and a teacher meet, as unavailable periods of the corresponding 'section'.

Solve the model on a PC by using a Mixed Integer Programming (MIP) solver. In the solution output, the periods where variable $x_{s,t,d,p} = 1$ are the periods that a section and a teacher meet. . Take these periods as the input data for the next iterations.

The results of Iteration-1 and Iteration-2 together form the timetable of the first 12 sections.

d. Iteration-3

Take 'the second section group' and 'the first teacher group' and develop the model again with these new sets. The size of the problem is small enough again, with 9,073 binary variables and 28,025 constraints.

Input the results found in Iteration-1; i.e. the periods where a section and a teacher meet, as unavailable periods of the corresponding 'teacher'. Also, examine the results of Iteration-2 and see whether there are any unassigned periods of any part-time teacher from the second teacher group. If there exist any periods of this type, then set the corresponding periods of any section in the section availability matrix to zero in order to keep them unoccupied for Iteration-4. This adjustment enables us to fully assign the part-time teachers in the second group at their available periods.

After the input and adjustment operations, solve that model on a computer by using a Mixed Integer Programming (MIP) solver for the optimization.

In the solution output, the periods where variable $x_{s,t,d,p} = 1$ are the periods that a section and a teacher meet. Take these periods as the input data for the next iteration.

e. Iteration-4

Take 'the second section group' and 'the second teacher group' and develop the model again with these new sets. The size of the problem is small enough again, with 10,081 binary variables and 28,209 constraints.

Input the results found in Iteration-2; i.e. the periods where a section and a teacher meet, as unavailable periods of the corresponding teacher. Also input the results found in Iteration-3 as unavailable periods of the corresponding section.

Solve this on a PC by using a Mixed Integer Programming (MIP) solver. In the solution output, the periods where variable $x_{s,t,d,p} = 1$ are the periods that a section and a teacher meet. . Take these periods as the input data for the next iterations.

The results of Iteration-3 and Iteration-4 together form the timetable of the second 12 sections.

4.4 The Results of the Solution Process

The iterative solution suggested in Part 4.3 decreases the dimensions of the problem to a solvable size. This procedure is a heuristic, since one has no guarantee that after having scheduled courses at some periods, the remaining problem still admits a solution. When encountered with such a situation, one can solve the problem by making small manual modifications in the problem parameters; i.e. by relaxing the availability and preferences data of the teachers. For the problem considered in this study, we found feasible solutions at each iteration. The solution obtained as a result of four iterations satisfied all of the problem requirements and minimized the objective function. The solution was near-optimal within the iteration limits and was found in several hours.

In Iteration-1, an optimal objective function value of $Z = 160$ has been found in approximately 16 minutes and in nearly 10,000 iterations. In Iteration-2, a near-optimal objective function value of $Z = 244$ has been found (within the iteration limits entered into the solver) in nearly 12 hours and in approximately 1,110,000 iterations.

Having completed the iterations 1 and 2, the teachers were successfully assigned to the first 12 sections. The resulting timetable of the first 12 sections is shown in Table 4.6 and Table 4.7.

Table 4.6 The timetable for the sections 1-6

		DAYS (d)				
SECTIONS	**PERIODS**	**1**	**2**	**3**	**4**	**5**
SECTION-1 (s)	**1**			C: 7 T: 43	C: 4 T: 19	C: 1 T: 3
	2			C: 1 T: 3	C: 4 T: 19	C: 1 T: 3
	3		C: 5 T: 27	C: 1 T: 3		C: 7 T: 43
	4	C: 6 T: 31	C: 5 T: 27	C: 8 T: 45		C: 7 T: 43
	5	C: 6 T: 31		C: 2 T: 11		C: 3 T: 16
	6	C: 6 T: 31		C: 2 T: 11		C: 3 T: 16
SECTION-2 (s)	**1**			C: 5 T: 29	C: 1 T: 2	C: 7 T: 41
	2			C: 5 T: 29	C: 1 T: 2	C: 7 T: 41
	3		C: 2 T: 9	C: 1 T: 2		C: 8 T: 45
	4	C: 7 T: 41	C: 2 T: 9	C: 1 T: 2		C: 6 T: 30
	5	C: 4 T: 21		C: 3 T: 12		C: 6 T: 30
	6	C: 4 T: 21		C: 3 T: 12		C: 6 T: 30
SECTION-3 (s)	**1**			C: 3 T: 13	C: 7 T: 40	C: 7 T: 40
	2			C: 3 T: 13	C: 6 T: 34	C: 7 T: 40
	3		C: 1 T: 7	C: 5 T: 29		C: 1 T: 7
	4	C: 8 T: 44	C: 1 T: 7	C: 5 T: 29		C: 1 T: 7
	5	C: 4 T: 22		C: 6 T: 34		C: 2 T: 11
	6	C: 4 T: 22		C: 6 T: 34		C: 2 T: 11
SECTION-4 (s)	**1**			C: 1 T: 5	C: 3 T: 14	C: 4 T: 20
	2			C: 1 T: 5	C: 3 T: 14	C: 4 T: 20
	3		C: 6 T: 35			C: 1 T: 5
	4	C: 6 T: 35	C: 6 T: 35			C: 1 T: 5
	5	C: 8 T: 45	C: 7 T: 42	C: 5 T: 27		C: 2 T: 9
	6	C: 7 T: 42	C: 7 T: 42	C: 5 T: 27		C: 2 T: 9
SECTION-5 (s)	**1**	C: 4 T: 21		C: 1 T: 4	C: 7 T: 43	C: 5 T: 28
	2	C: 4 T: 21		C: 1 T: 4	C: 7 T: 43	C: 5 T: 28
	3	C: 8 T: 44	C: 7 T: 43			C: 1 T: 4
	4	C: 6 T: 33				C: 1 T: 4
	5	C: 6 T: 33		C: 2 T: 10		C: 3 T: 15
	6	C: 6 T: 33		C: 2 T: 10		C: 3 T: 15
SECTION-6 (s)	**1**	C: 2 T: 11		C: 1 T: 2	C: 5 T: 27	C: 1 T: 2
	2	C: 2 T: 11		C: 1 T: 2	C: 5 T: 27	C: 1 T: 2
	3	C: 3 T: 14	C: 6 T: 34			C: 7 T: 41
	4	C: 3 T: 14				C: 7 T: 41
	5	C: 6 T: 34		C: 7 T: 41		C: 4 T: 18
	6	C: 6 T: 34		C: 8 T: 45		C: 4 T: 18

Note : Dashed cells are occupied with other activities; i.e. they are the unavailable periods of sections.

Table 4.7 The timetable for the sections 7-12

		DAYS (d)				
SECTION	**PERIODS**	**1**	**2**	**3**	**4**	**5**
SECTION-7 (s)	1	C: 1 T: 3			C: 6 T: 30	C: 6 T: 30
	2	C: 1 T: 3			C: 8 T: 44	C: 6 T: 30
	3	C: 4 T: 24				C: 3 T: 12
	4	C: 4 T: 24		C: 7 T: 38		C: 3 T: 12
	5		C: 2 T: 10	C: 7 T: 38	C: 5 T: 27	C: 1 T: 3
	6		C: 2 T: 10	C: 7 T: 38	C: 5 T: 27	C: 1 T: 3
SECTION-8 (s)	1	C: 7 T: 40			C: 6 T: 31	C: 3 T: 15
	2	C: 7 T: 40			C: 6 T: 31	C: 3 T: 15
	3	C: 7 T: 40				C: 1 T: 3
	4	C: 8 T: 45		C: 6 T: 31		C: 1 T: 3
	5		C: 1 T: 3	C: 5 T: 26	C: 2 T: 11	C: 4 T: 20
	6		C: 1 T: 3	C: 5 T: 26	C: 2 T: 11	C: 4 T: 20
SECTION-9 (s)	1	C: 7 T: 43			C: 1 T: 3	C: 7 T: 43
	2	C: 7 T: 43			C: 1 T: 3	C: 6 T: 37
	3	C: 1 T: 3				C: 6 T: 37
	4	C: 1 T: 3		C: 8 T: 44		C: 6 T: 37
	5		C: 2 T: 11	C: 3 T: 13	C: 4 T: 19	C: 5 T: 25
	6		C: 2 T: 11	C: 3 T: 13	C: 4 T: 19	C: 5 T: 25
SECTION-10 (s)	1	C: 1 T: 2			C: 7 T: 41	C: 3 T: 12
	2	C: 1 T: 2			C: 7 T: 41	C: 3 T: 12
	3	C: 4 T: 21	C: 5 T: 26			C: 1 T: 2
	4	C: 4 T: 21	C: 5 T: 26	C: 6 T: 34		C: 1 T: 2
	5		C: 2 T: 9		C: 6 T: 34	C: 7 T: 41
	6		C: 2 T: 9		C: 6 T: 34	C: 8 T: 45
SECTION-11 (s)	1	C: 6 T: 34		C: 2 T: 9	C: 5 T: 26	C: 1 T: 5
	2	C: 6 T: 34		C: 2 T: 9	C: 5 T: 26	C: 1 T: 5
	3	C: 6 T: 34	C: 8 T: 45	C: 1 T: 5		C: 4 T: 20
	4	C: 7 T: 39		C: 1 T: 5		C: 4 T: 20
	5				C: 3 T: 14	C: 7 T: 39
	6				C: 3 T: 14	C: 7 T: 39
SECTION-12 (s)	1	C: 7 T: 42		C: 5 T: 26	C: 1 T: 4	C: 1 T: 4
	2	C: 6 T: 30		C: 5 T: 26	C: 1 T: 4	C: 1 T: 4
	3	C: 6 T: 30	C: 8 T: 44	C: 7 T: 42		C: 4 T: 24
	4	C: 6 T: 30		C: 7 T: 42		C: 4 T: 24
	5				C: 3 T: 17	C: 2 T: 10
	6				C: 3 T: 17	C: 2 T: 10

Note : Dashed cells are occupied with other activities; i.e. they are the unavailable periods of sections.

Table 4.8 The timetable for the sections 13-19

		DAYS (d)				
SECTION	**PERIODS**	**1**	**2**	**3**	**4**	**5**
SECTION-13 (s)	1	C: 6 T: 32	C: 2 T: 10	C: 7 T: 40		
	2	C: 6 T: 32	C: 2 T: 10	C: 8 T: 44		
	3	C: 1 T: 6	C: 1 T: 6	C: 5 T: 26	C: 6 T: 32	C: 7 T: 40
	4	C: 1 T: 6	C: 1 T: 6	C: 5 T: 26		C: 7 T: 40
	5			C: 3 T: 16		C: 4 T: 24
	6			C: 3 T: 16		C: 4 T: 24
SECTION-14 (s)	1	C: 6 T: 33	C: 4 T: 19	C: 6 T: 33		
	2	C: 6 T: 33	C: 4 T: 19	C: 8 T: 46		
	3	C: 2 T: 10	C: 1 T: 2	C: 3 T: 13	C: 7 T: 43	C: 5 T: 28
	4	C: 2 T: 10	C: 1 T: 2	C: 3 T: 13		C: 5 T: 28
	5			C: 7 T: 43		C: 1 T: 2
	6			C: 7 T: 43		C: 1 T: 2
SECTION-15 (s)	1	C: 4 T: 19	C: 5 T: 27	C: 7 T: 41		
	2	C: 4 T: 19	C: 5 T: 27	C: 7 T: 41		
	3	C: 1 T: 8	C: 6 T: 31	C: 6 T: 31	C: 1 T: 8	C: 2 T: 11
	4	C: 1 T: 8	C: 6 T: 31		C: 1 T: 8	C: 2 T: 11
	5				C: 3 T: 16	C: 8 T: 45
	6				C: 3 T: 16	C: 7 T: 41
SECTION-16 (s)	1	C: 1 T: 5	C: 2 T: 9	C: 6 T: 30		
	2	C: 1 T: 5	C: 2 T: 9	C: 6 T: 30		
	3	C: 4 T: 19	C: 1 T: 5	C: 7 T: 43		C: 6 T: 30
	4	C: 4 T: 19	C: 1 T: 5			C: 8 T: 44
	5	C: 5 T: 25			C: 3 T: 15	C: 7 T: 43
	6	C: 5 T: 25			C: 3 T: 15	C: 7 T: 43
SECTION-17 (s)	1	C: 3 T: 17	C: 1 T: 4	C: 8 T: 44		
	2	C: 3 T: 17	C: 1 T: 4	C: 2 T: 11		
	3	C: 7 T: 42	C: 4 T: 22	C: 2 T: 11		C: 7 T: 42
	4	C: 6 T: 32	C: 4 T: 22			C: 7 T: 42
	5	C: 6 T: 32			C: 1 T: 4	C: 5 T: 28
	6	C: 6 T: 32			C: 1 T: 4	C: 5 T: 28
SECTION-18 (s)	1	C: 5 T: 28	C: 8 T: 44	C: 7 T: 38		
	2	C: 5 T: 28	C: 6 T: 36	C: 7 T: 38		
	3	C: 3 T: 17	C: 6 T: 36	C: 7 T: 38		C: 4 T: 23
	4	C: 3 T: 17	C: 6 T: 36			C: 4 T: 23
	5	C: 1 T: 5			C: 2 T: 10	C: 1 T: 5
	6	C: 1 T: 5			C: 2 T: 10	C: 1 T: 5

Note : Dashed cells are occupied with other activities; i.e. they are the unavailable periods of sections.

Table 4.9 The timetable for the sections 19-24

		DAYS (d)				
SECTION	**PERIODS**	**1**	**2**	**3**	**4**	**5**
SECTION-19 (s)	1	C: 6 T: 35		C: 7 T: 42		
	2	C: 6 T: 35		C: 7 T: 42		
	3	C: 6 T: 35		C: 3 T: 16	C: 1 T: 1	C: 2 T: 10
	4	C: 7 T: 42	C: 8 T: 44	C: 3 T: 16	C: 1 T: 1	C: 2 T: 10
	5	C: 1 T: 1	C: 5 T: 26			C: 4 T: 19
	6	C: 1 T: 1	C: 5 T: 26			C: 4 T: 19
SECTION-20 (s)	1	C: 7 T: 39		C: 6 T: 35		
	2	C: 7 T: 39		C: 7 T: 39		
	3	C: 5 T: 28		C: 1 T: 4	C: 2 T: 10	C: 3 T: 15
	4	C: 5 T: 28	C: 8 T: 45	C: 1 T: 4	C: 2 T: 10	C: 3 T: 15
	5	C: 6 T: 35	C: 4 T: 22			C: 1 T: 4
	6	C: 6 T: 35	C: 4 T: 22			C: 1 T: 4
SECTION-21 (s)	1	C: 8 T: 44		C: 6 T: 31		
	2	C: 6 T: 31		C: 6 T: 31		
	3	C: 1 T: 1		C: 1 T: 1	C: 5 T: 27	C: 2 T: 9
	4	C: 1 T: 1	C: 7 T: 40	C: 1 T: 1	C: 5 T: 27	C: 2 T: 9
	5	C: 7 T: 40	C: 4 T: 19			C: 3 T: 13
	6	C: 7 T: 40	C: 4 T: 19			C: 3 T: 13
SECTION-22 (s)	1	C: 4 T: 23		C: 6 T: 34		
	2	C: 4 T: 23		C: 6 T: 34		
	3			C: 6 T: 34	C: 2 T: 11	C: 5 T: 25
	4		C: 7 T: 42	C: 8 T: 46	C: 2 T: 11	C: 5 T: 25
	5	C: 3 T: 16	C: 1 T: 5		C: 1 T: 5	C: 7 T: 42
	6	C: 3 T: 16	C: 1 T: 5		C: 1 T: 5	C: 7 T: 42
SECTION-23 (s)	1	C: 5 T: 25	C: 1 T: 1	C: 1 T: 1		
	2	C: 5 T: 25	C: 1 T: 1	C: 1 T: 1		
	3		C: 6 T: 32	C: 7 T: 41	C: 7 T: 41	C: 6 T: 32
	4		C: 6 T: 32	C: 7 T: 41		C: 8 T: 45
	5	C: 2 T: 11	C: 3 T: 16			C: 4 T: 22
	6	C: 2 T: 11	C: 3 T: 16			C: 4 T: 22
SECTION-24 (s)	1	C: 3 T: 15	C: 2 T: 11	C: 8 T: 46		
	2	C: 3 T: 15	C: 2 T: 11	C: 6 T: 33		
	3		C: 1 T: 4	C: 6 T: 33	C: 7 T: 40	C: 4 T: 18
	4		C: 1 T: 4	C: 6 T: 33		C: 4 T: 18
	5	C: 1 T: 4	C: 5 T: 27			C: 7 T: 40
	6	C: 1 T: 4	C: 5 T: 27			C: 7 T: 40

Note : Dashed cells are occupied with other activities; i.e. they are the unavailable periods of sections.

Table 4.10 The timetable of teachers

X : Unavailable periods		DAYS (d)																													
		1						2						3						4						5					
		Periods (p)						Periods (p)						Periods (p)						Periods (p)						Periods (p)					
C:	T:	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
1	1			21	21	19	19	23	23					23	23	21	21					19	19								
	2	10	10							14	14			6	6	2	2			2	2					6	6	10	10	14	14
	3	7	7			9	9					8	8		1	1				9	9					1	1	8	8	7	7
	4					24	24	17	17	24	24			5	5	20	20			12	12			17	17	12	12	5	5	20	20
	5	16	16			18	18			16	16	22	22	4	4	11	11							22	22	11	11	4	4	18	18
	6	X	X	13	13	X	X	X	X	13	13	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	7	X	X	X	X	X	X	X	X	3	3	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	3	3	X	X
	8	X	X	15	15	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	15	15	X	X	X	X	X	X	X	X
2	9							16	16	2	2	10	10	11	11													21	21	4	4
	10			14	14			13	13			7	7					5	5			20	20	18	18			19	19	12	12
	11	6	6			23	23	24	24			9	9		17	17		1	1			22	22	8	8			15	15	3	3
3	12																	2	2							10	10	7	7	21	21
	13													3	3	14	14	9	9												
	14			6	6															4	4			11	11						
	15	24	24																					16	16	8	8	20	20	5	5
	16					22	22					23	23			19	19	13	13					15	15					1	1
	17	17	17	18	18																			12	12						
4	18																											24	24	6	6
	19	15	15	16	16			14	14			21	21							1	1			9	9					19	19
	20	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	4	4	11	11	8	8
	21	5	5	10	10	2	2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	22					3	3			17	17	20	20																	23	23
	23	22	22																									18	18		
	24			7	7																							12	12	13	13

Note : The numbers in each cell represents the sections that the teacher on that row is assigned to.

Table 4.10 (continued)

X : Unavail -able periods		DAYS (d)																													
		1						2						3						4						5					
		Periods (p)						Periods (p)						Periods (p)						Periods (p)						Periods (p)					
C:	T:	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
	25	23	23			16	16																					22	22	9	9
	26									10	10	19	19	12	12	13	13	8	8	11	11										
5	27							15	15	1	1	24	24					4	4	6	6	21	21	7	7						
	28	18	18	20	20																					5	5	14	14	17	17
	29	X	X	X	X	X	X	X	X	X	X	X	X	2	2	3	3	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	30		12	12	12									16	16					7						7	7	16	2	2	2
	31		21		1	1	1			15	15			21	21	15	8			8	8										
	32	13	13		17	17	17			23	23											13						23			
	33	14	14		5	5	5							14	24	24	24														
6	34	11	11	11		6	6			6				22	22	22	10	3	3		3			10	10						
	35	19	19	19	4	20	20			4	4			20																	
	36	X	X	X	X	X	X	X	18	18	18	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	37	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	9	9	9	X	X
	38	X	X	X	X	X	X	X	X	X	X	X	X	18	18	18	7	7	7	X	X	X	X	X	X	X	X	X	X	X	X
	39	20	20	20	11										20															11	11
	40	8	8	8		21	21				21			13						3		24				3	3	13	13	24	24
7	41				2									15	15	23	23	6		10	10	23				2	2	6	6	10	15
	42	12		17	19		4				22	4	4	19	19	12	12											17	17	22	22
	43	9	9							5				1		16		14	14	5	5	14				9		11	11	16	16
	44	21		5	3			18		12	19			17	13		9				7								16		
8	45				8	4				11	20						1		6									2	23	15	10
	46													24	14		22														

Note : The numbers in each cell represents the sections that the teacher on that row is assigned to.

When we constructed the teacher timetable after these iterations, it is seen that the available periods of some part-time teachers teaching Course-1 (Teacher-6 and 8) and the first three periods of part-time teacher teaching Course-7 (Teacher-38) remained unassigned. In order to prevent the available periods of sections corresponding to the unassigned periods of these teachers from being occupied at Iteration-3 and be able to assign these teachers to the available periods of the second-group sections, we set the corresponding periods of sections to zero before solving the model of the second section group and the first teacher group at Iteration-3.

When the model is solved in Iteration-3, an optimal objective function value of $Z = 192$ has been found in approximately 15 minutes and in approximately 10,500 iterations.

In Iteration-4, a near-optimal objective function value of $Z = 260$ has been found (within the iteration limits entered into the solver) in nearly 6 hours and in approximately 1,000,000 iterations.

Having completed the iterations 3 and 4, the teachers were successfully assigned to the second 12 sections. The resulting timetable of the second 12 sections is shown in Table 4.8 and Table 4.9.

At the end of the solution procedure, all sections have obtained their timetables without any empty periods.

The model minimized the objective function value. The cost coefficients in the objective function with value 'medium' and 'high' corresponding to moderately preferred and non-preferred periods of teachers respectively, prevented but did not prohibited the presence of their corresponding variables in the final solution. The available periods of part-time teachers with penalty cost coefficient value of '0' in the objective function were fully scheduled in the final solution. There is only one non-preferred period scheduled for only one teacher (Teacher #39 teaching Course-7). Teachers #1 and #2 teaching Course-1, teacher #35 teaching Course-6 and teacher #40 teaching Course-7 were scheduled at 6, 2, 1, and 1 moderately preferred periods respectively. The ratio of non-preferred periods that have been assigned to total non-preferred ones is only 0.95 % and the ratio of moderately preferred periods that have been assigned to total moderately preferred ones is only 4 %.

Therefore, the timetable in the final solution can easily said to be “a good timetable” for the teachers. The resulting timetable of teachers is shown in Table 4.10.

When compared with the manual timetabling which lasts even a month, the time consumed for this model is very short. Thus, this is also “a good timetable” for the timetablers.

5. CONCLUSION AND RECOMMENDATIONS

In this thesis study, an integer programming model and a solution strategy was presented in order obtain good solutions for the timetabling problem of Turkish Military Academy (TMA). Although it was implemented on the sections of the second class at TMA, it is iteratively applicable to the other classes by revising the data and making little modifications.

The proposed model has the appropriate facilities for providing valuable help to the scheduler to implement a good timetable of TMA. It can be easily said that it is "generic" because it addresses most of the items that can appear in many schools: No section and teacher overlaps are allowed; section and teacher availabilities must be taken into account; courses have different number of lectures; the distribution of the lectures of the courses in a week must be done according to the rules; lectures of a course on the same day must be consecutive; lunch breaks have to be observed; etc.

The model satisfies the teacher preferences to a sufficient degree by using suitable objective function coefficients. These coefficients penalize the assignment of teachers to the periods that they do not prefer. These coefficients also enable us to assign the part-time teachers at their available periods prior to the full-time ones.

The penalty cost coefficients in the objective function can take different values according to the teachers' preference intensity. For instance; if some periods are absolutely not preferred with respect to the others, these periods would have penalty cost coefficients with very high values while the others have theirs with very low values. (Example: 0, 20, 30, 150)

In addition, the solution strategy proposed is based on grouping sections and teachers, and on allocating teacher groups to section groups iteratively in such a way that the output data of an iteration will be considered as an additional input data for the next one. This grouping strategy made it possible to decompose the problem into the problems of smaller sizes.

One strong point of the model is the fact that one can easily replace and/or create new rules and constraints. For instance, a goal for not assigning teachers to sections of different locations during the periods before noon and also during the periods in the afternoon may be added to the model. Also, the qualitative conditions that are in conflict can be handled by properly adjusting parameters and/or objective function coefficients. As an example, a part-time teacher inevitably may not be assigned fully at his available periods by the model. If the scheduler wants to assign this teacher fully at his available periods, then he can make little modification on the values of the section availability matrix regarding the periods corresponding to the teacher's available times. Similarly, the scheduler can modify the penalty function values in the objective function in order to achieve better quality and wider acceptance of the results.

The timetable constructed by the implementation of the model in this study has only one non-preferred period for only one teacher. Besides, the time consumed for this model is very short when compared with the manual timetabling. Thus, it is said to be "a good timetable" both for the teachers and the scheduler.

This study creates a starting point to the contributions about the timetabling problem of TMA. It is recommended that, in the future, a solution approach for solving the timetabling problem of TMA at only one stage and with all its classes as a whole have to be improved.

APPENDICES

Appendix A: The Computer Input of the Model for Iteration-1

```
sets
  s /1*12/
  t /9*29/
  t2(t)/9*11/
  t3(t)/12*17/
  t4(t)/18*24/
  t5(t)/25*29/
  t15(t)/9*29/
  d /1*5/
  p /1*6/ ;
scalar
  hb /2/ ;
table sav(s,d,p)
         1   2   3   4   5   6
   1.1   0   0   0   1   1   1
   1.2   0   0   1   1   0   0
   1.3   1   1   1   1   1   1
   1.4   1   1   0   0   0   0
   1.5   1   1   1   1   1   1
   2.1   0   0   0   1   1   1
   2.2   0   0   1   1   0   0
   2.3   1   1   1   1   1   1
   2.4   1   1   0   0   0   0
   2.5   1   1   1   1   1   1
   3.1   0   0   0   1   1   1
   3.2   0   0   1   1   0   0
   3.3   1   1   1   1   1   1
   3.4   1   1   0   0   0   0
   3.5   1   1   1   1   1   1
   4.1   0   0   0   1   1   1
   4.2   0   0   1   1   1   1
   4.3   1   1   0   0   1   1
   4.4   1   1   0   0   0   0
   4.5   1   1   1   1   1   1
   5.1   1   1   1   1   1   1
   5.2   0   0   1   0   0   0
   5.3   1   1   0   0   1   1
   5.4   1   1   0   0   0   0
   5.5   1   1   1   1   1   1
   6.1   1   1   1   1   1   1
   6.2   0   0   1   0   0   0
   6.3   1   1   0   0   1   1
   6.4   1   1   0   0   0   0
   6.5   1   1   1   1   1   1
   7.1   1   1   1   1   0   0
   7.2   0   0   0   0   1   1
   7.3   0   0   0   1   1   1
   7.4   1   1   0   0   1   1
   7.5   1   1   1   1   1   1
   8.1   1   1   1   1   0   0
   8.2   0   0   0   0   1   1
   8.3   0   0   0   1   1   1
   8.4   1   1   0   0   1   1
   8.5   1   1   1   1   1   1
   9.1   1   1   1   1   0   0
   9.2   0   0   0   0   1   1
   9.3   0   0   0   1   1   1
   9.4   1   1   0   0   1   1
   9.5   1   1   1   1   1   1
  10.1   1   1   1   1   0   0
  10.2   0   0   1   1   1   1
  10.3   0   0   0   1   0   0
  10.4   1   1   0   0   1   1
  10.5   1   1   1   1   1   1
  11.1   1   1   1   1   0   0
  11.2   0   0   1   0   0   0
  11.3   1   1   1   1   0   0
  11.4   1   1   0   0   1   1
  11.5   1   1   1   1   1   1
  12.1   1   1   1   1   0   0
  12.2   0   0   1   0   0   0
  12.3   1   1   1   1   0   0
  12.4   1   1   0   0   1   1
  12.5   1   1   1   1   1   1 ;
table tav(t,d,p)
          1   2   3   4   5   6
   9.1    1   1   1   1   1   1
   9.2    1   1   1   1   1   1
   9.3    1   1   1   1   1   1
   9.4    1   1   1   1   1   1
   9.5    1   1   1   1   1   1
```

10.1	1	1	1	1	1	1
10.2	1	1	1	1	1	1
10.3	1	1	1	1	1	1
10.4	1	1	1	1	1	1
10.5	1	1	1	1	1	1
11.1	1	1	1	1	1	1
11.2	1	1	1	1	1	1
11.3	1	1	1	1	1	1
11.4	1	1	1	1	1	1
11.5	1	1	1	1	1	1
12.1	1	1	1	1	1	1
12.2	1	1	1	1	1	1
12.3	1	1	1	1	1	1
12.4	1	1	1	1	1	1
12.5	1	1	1	1	1	1
13.1	1	1	1	1	1	1
13.2	1	1	1	1	1	1
13.3	1	1	1	1	1	1
13.4	1	1	1	1	1	1
13.5	1	1	1	1	1	1
14.1	1	1	1	1	1	1
14.2	1	1	1	1	1	1
14.3	1	1	1	1	1	1
14.4	1	1	1	1	1	1
14.5	1	1	1	1	1	1
15.1	1	1	1	1	1	1
15.2	1	1	1	1	1	1
15.3	1	1	1	1	1	1
15.4	1	1	1	1	1	1
15.5	1	1	1	1	1	1
16.1	1	1	1	1	1	1
16.2	1	1	1	1	1	1
16.3	1	1	1	1	1	1
16.4	1	1	1	1	1	1
16.5	1	1	1	1	1	1
17.1	1	1	1	1	1	1
17.2	1	1	1	1	1	1
17.3	1	1	1	1	1	1
17.4	1	1	1	1	1	1
17.5	1	1	1	1	1	1
18.1	1	1	1	1	1	1
18.2	1	1	1	1	1	1
18.3	1	1	1	1	1	1
18.4	1	1	1	1	1	1
18.5	1	1	1	1	1	1
19.1	1	1	1	1	1	1
19.2	1	1	1	1	1	1
19.3	1	1	1	1	1	1
19.4	1	1	1	1	1	1
19.5	1	1	1	1	1	1
20.1	0	0	0	0	0	0
20.2	0	0	0	0	0	0
20.3	0	0	0	0	0	0
20.4	0	0	0	0	0	0
20.5	1	1	1	1	1	1
21.1	1	1	1	1	1	1
21.2	0	0	0	0	0	0
21.3	0	0	0	0	0	0
21.4	0	0	0	0	0	0
21.5	0	0	0	0	0	0
22.1	1	1	1	1	1	1
22.2	1	1	1	1	1	1
22.3	1	1	1	1	1	1
22.4	1	1	1	1	1	1
22.5	1	1	1	1	1	1
23.1	1	1	1	1	1	1
23.2	1	1	1	1	1	1
23.3	1	1	1	1	1	1
23.4	1	1	1	1	1	1
23.5	1	1	1	1	1	1
24.1	1	1	1	1	1	1
24.2	1	1	1	1	1	1
24.3	1	1	1	1	1	1
24.4	1	1	1	1	1	1
24.5	1	1	1	1	1	1
25.1	1	1	1	1	1	1
25.2	1	1	1	1	1	1
25.3	1	1	1	1	1	1
25.4	1	1	1	1	1	1
25.5	1	1	1	1	1	1
26.1	1	1	1	1	1	1

```
    26.2     1    1    1    1    1    1
    26.3     1    1    1    1    1    1
    26.4     1    1    1    1    1    1
    26.5     1    1    1    1    1    1
    27.1     1    1    1    1    1    1
    27.2     1    1    1    1    1    1
    27.3     1    1    1    1    1    1
    27.4     1    1    1    1    1    1
    27.5     1    1    1    1    1    1
    28.1     1    1    1    1    1    1
    28.2     1    1    1    1    1    1
    28.3     1    1    1    1    1    1
    28.4     1    1    1    1    1    1
    28.5     1    1    1    1    1    1
    29.1     0    0    0    0    0    0
    29.2     0    0    0    0    0    0
    29.3     1    1    1    1    0    0
    29.4     0    0    0    0    0    0
    29.5     0    0    0    0    0    0 ;
table pm(t,d,p)
             1    2    3    4    5    6
    9.1      4    4    4    4    4    4
    9.2      2    2    2    2    2    2
    9.3      2    2    2    2    2    2
    9.4      4    4    4    4    4    4
    9.5      2    2    2    2    2    2
    10.1     2    2    2    2    2    2
    10.2     2    2    2    2    2    2
    10.3     2    2    2    2    2    2
    10.4     2    2    2    2    2    2
    10.5     2    2    2    2    2    2
    11.1     2    2    2    2    2    2
    11.2     2    2    2    2    2    2
    11.3     2    2    2    2    2    2
    11.4     2    2    2    2    2    2
    11.5     2    2    2    2    2    2
    12.1     4    4    4    4    4    4
    12.2     4    4    4    4    4    4
    12.3     2    2    2    2    2    2
    12.4     4    4    4    4    4    4
    12.5     2    2    2    2    2    2
    13.1     4    4    4    4    4    4
    13.2     4    4    4    4    4    4
    13.3     2    2    2    2    2    2
    13.4     2    2    2    2    2    2
    13.5     2    2    2    2    2    2
    14.1     2    2    2    2    2    2
    14.2     2    2    2    2    2    2
    14.3     4    4    4    4    4    4
    14.4     2    2    2    2    2    2
    14.5     6    6    6    6    6    6
    15.1     2    2    2    2    2    2
    15.2     4    4    4    4    4    4
    15.3     4    4    4    4    4    4
    15.4     2    2    2    2    2    2
    15.5     2    2    2    2    2    2
    16.1     6    6    2    2    2    2
    16.2     6    6    2    2    2    2
    16.3     6    6    2    2    2    2
    16.4     6    6    2    2    2    2
    16.5     6    6    2    2    2    2
    17.1     2    2    2    2    2    2
    17.2     4    4    4    4    4    4
    17.3     4    4    4    4    4    4
    17.4     2    2    2    2    2    2
    17.5     6    6    6    6    6    6
    18.1     4    4    2    2    2    2
    18.2     4    4    2    2    2    2
    18.3     4    4    2    2    2    2
    18.4     4    4    2    2    2    2
    18.5     4    4    2    2    2    2
    19.1     2    2    2    2    2    2
    19.2     2    2    2    2    2    2
    19.3     4    4    4    4    4    4
    19.4     2    2    2    2    2    2
    19.5     2    2    2    2    2    2
    20.1     9    9    9    9    9    9
    20.2     9    9    9    9    9    9
    20.3     9    9    9    9    9    9
    20.4     9    9    9    9    9    9
    20.5     0    0    0    0    0    0
```

```
    21.1    0   0   0   0   0   0
    21.2    9   9   9   9   9   9
    21.3    9   9   9   9   9   9
    21.4    9   9   9   9   9   9
    21.5    9   9   9   9   9   9
    22.1    6   6   6   6   2   2
    22.2    2   2   2   2   2   2
    22.3    4   4   4   4   4   4
    22.4    6   6   6   6   6   6
    22.5    2   2   2   2   2   2
    23.1    2   2   2   2   6   6
    23.2    2   2   2   2   6   6
    23.3    2   2   2   2   6   6
    23.4    2   2   2   2   6   6
    23.5    2   2   2   2   6   6
    24.1    4   4   2   2   2   2
    24.2    6   6   6   6   6   6
    24.3    6   6   6   6   6   6
    24.4    4   4   4   4   4   4
    24.5    4   4   2   2   2   2
    25.1    2   2   2   2   2   2
    25.2    4   4   4   4   4   4
    25.3    4   4   4   4   4   4
    25.4    4   4   4   4   4   4
    25.5    2   2   2   2   2   2
    26.1    4   4   4   4   4   4
    26.2    2   2   2   2   2   2
    26.3    2   2   2   2   2   2
    26.4    2   2   2   2   2   2
    26.5    4   4   4   4   4   4
    27.1    4   4   4   4   4   4
    27.2    2   2   2   2   2   2
    27.3    2   2   2   2   2   2
    27.4    2   2   2   2   2   2
    27.5    4   4   4   6   6   6
    28.1    2   2   2   2   2   2
    28.2    2   2   2   2   2   2
    28.3    6   6   6   6   6   6
    28.4    6   6   6   6   6   6
    28.5    2   2   2   2   2   2
    29.1    9   9   9   9   9   9
    29.2    9   9   9   9   9   9
    29.3    0   0   0   0   9   9
    29.4    9   9   9   9   9   9
    29.5    9   9   9   9   9   9   ;
variables
    y(s,t)
    v(s,t,d)
    x(s,t,d,p)
    z ;
binary variables
   y
   v
   x ;
equations
   obj
   secav(s,d,p)
   teaav(t,d,p)
   onetea2(s)
   onetea3(s)
   onetea4(s)
   onetea5(s)
   oneday(s,t,d)
   onedaytw(s,t15)
   oneteatwo(s,t,d,p)
   mostper(t)
   hours2(s,t15)
   twohone(s,t15,d)
   twohtwo(s,t15,d)
   twohthree(s,t15,d)
   twohfour(s,t15,d)
   twohfive(s,t15,d)
   twohsix(s,t15,d)
   twohseven(s,t15,d)
   twoheight(s,t15,d)
   twohnine(s,t15,d)
   twohten(s,t15,d)
   twohele(s,t15,d)
   twohtwe(s,t15,d)
   twohthi(s,t15,d)
   twohfourt(s,t15,d) ;
```

```
obj.. z=e=sum((s,t,d,p),pm(t,d,p)*x(s,t,d,p)) ;
secav(s,d,p).. sum(t,x(s,t,d,p))=l=sav(s,d,p) ;
teaav(t,d,p).. sum(s,x(s,t,d,p))=l=tav(t,d,p) ;
onetea2(s).. sum(t2,y(s,t2))=e=1;
onetea3(s).. sum(t3,y(s,t3))=e=1;
onetea4(s).. sum(t4,y(s,t4))=e=1;
onetea5(s).. sum(t5,y(s,t5))=e=1;
oneday(s,t,d).. v(s,t,d)=l=y(s,t) ;
onedaytw(s,t15).. sum(d,v(s,t15,d))=l=1 ;
oneteatwo(s,t,d,p).. x(s,t,d,p)=l=v(s,t,d) ;
mostper(t).. sum((s,d,p),x(s,t,d,p))=l=20  ;
hours2(s,t15).. sum((d,p),x(s,t15,d,p))=e=hb*y(s,t15) ;
twohone(s,t15,d).. sum(p,x(s,t15,d,p))=l=2;
twohtwo(s,t15,d).. x(s,t15,d,'1')-x(s,t15,d,'2')=l=0 ;
twohthree(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'3')=l=1 ;
twohfour(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'4')=l=1 ;
twohfive(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'5')=l=1 ;
twohsix(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'6')=l=1 ;
twohseven(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'4')=l=1 ;
twoheight(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'5')=l=1 ;
twohnine(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'6')=l=1 ;
twohten(s,t15,d).. x(s,t15,d,'3')+x(s,t15,d,'5')=l=1 ;
twohele(s,t15,d).. x(s,t15,d,'3')+x(s,t15,d,'6')=l=1 ;
twohtwe(s,t15,d).. x(s,t15,d,'4')+x(s,t15,d,'5')=l=1 ;
twohthi(s,t15,d).. x(s,t15,d,'4')+x(s,t15,d,'6')=l=1 ;
twohfourt(s,t15,d).. x(s,t15,d,'5')-x(s,t15,d,'6')=e=0 ;
model esra /all/ ;
OPTION OPTCR=0.0;
OPTION ITERLIM=110000;
OPTION RESLIM=700000;
OPTION SOLPRINT=off;
OPTION LIMCOL=0;
OPTION LIMROW=0;
OPTION EJECT;
OPTION SYSOUT=off;
solve esra using mip minimizing z  ;
display x.l
```

Appendix B: The Computer Input of the Model for Iteration-2

```
sets
  s /1*12/
  t /1,2,3,4,5,6,7,8,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46/
  t1(t)/1*8/
  t6(t)/30*37/
  t7(t)/38*43/
  t8(t)/44*46/
  t15(t)/1*8/
  t67(t)/30*43/
  d /1*5/
  p /1*6/ ;
scalars
  ha /4/
  hc /3/
  hd /1/ ;
table sav(s,d,p)
           1    2    3    4    5    6
   1.1     0    0    0    1    1    1
   1.2     0    0    0    0    0    0
   1.3     1    1    1    1    0    0
   1.4     0    0    0    0    0    0
   1.5     1    1    1    1    0    0
   2.1     0    0    0    1    0    0
   2.2     0    0    0    0    0    0
   2.3     0    0    1    1    0    0
   2.4     1    1    0    0    0    0
   2.5     1    1    1    1    1    1
   3.1     0    0    0    1    0    0
   3.2     0    0    1    1    0    0
   3.3     0    0    0    0    1    1
   3.4     1    1    0    0    0    0
   3.5     1    1    1    1    0    0
   4.1     0    0    0    1    1    1
   4.2     0    0    1    1    1    1
   4.3     1    1    0    0    0    0
   4.4     0    0    0    0    0    0
   4.5     0    0    1    1    0    0
   5.1     0    0    1    1    1    1
   5.2     0    0    1    0    0    0
   5.3     1    1    0    0    0    0
   5.4     1    1    0    0    0    0
   5.5     0    0    1    1    0    0
   6.1     0    0    0    0    1    1
   6.2     0    0    1    0    0    0
   6.3     1    1    0    0    1    1
   6.4     0    0    0    0    0    0
   6.5     1    1    1    1    0    0
   7.1     1    1    0    0    0    0
   7.2     0    0    0    0    0    0
   7.3     0    0    0    1    1    1
   7.4     1    1    0    0    0    0
   7.5     1    1    0    0    1    1
   8.1     1    1    1    1    0    0
   8.2     0    0    0    0    1    1
   8.3     0    0    0    1    0    0
   8.4     1    1    0    0    0    0
   8.5     0    0    1    1    0    0
   9.1     1    1    1    1    0    0
   9.2     0    0    0    0    0    0
   9.3     0    0    0    1    0    0
   9.4     1    1    0    0    0    0
   9.5     1    1    1    1    0    0
  10.1     1    1    0    0    0    0
  10.2     0    0    0    0    0    0
  10.3     0    0    0    1    0    0
  10.4     1    1    0    0    1    1
  10.5     0    0    1    1    1    1
  11.1     1    1    1    1    0    0
  11.2     0    0    1    0    0    0
  11.3     0    0    1    1    0    0
  11.4     0    0    0    0    0    0
  11.5     1    1    0    0    1    1
  12.1     1    1    1    1    0    0
  12.2     0    0    1    0    0    0
  12.3     0    0    1    1    0    0
  12.4     1    1    0    0    0    0
  12.5     1    1    0    0    0    0 ;
table tav(t,d,p)
             1    2    3    4    5    6
   1.1       1    1    1    1    1    1
   1.2       1    1    1    1    1    1
```

1.3	1	1	1	1	1	1
1.4	1	1	1	1	1	1
1.5	1	1	1	1	1	1
2.1	1	1	1	1	1	1
2.2	1	1	1	1	1	1
2.3	1	1	1	1	1	1
2.4	1	1	1	1	1	1
2.5	1	1	1	1	1	1
3.1	1	1	1	1	1	1
3.2	1	1	1	1	1	1
3.3	1	1	1	1	1	1
3.4	1	1	1	1	1	1
3.5	1	1	1	1	1	1
4.1	1	1	1	1	1	1
4.2	1	1	1	1	1	1
4.3	1	1	1	1	1	1
4.4	1	1	1	1	1	1
4.5	1	1	1	1	1	1
5.1	1	1	1	1	1	1
5.2	1	1	1	1	1	1
5.3	1	1	1	1	1	1
5.4	1	1	1	1	1	1
5.5	1	1	1	1	1	1
6.1	0	0	1	1	0	0
6.2	0	0	1	1	0	0
6.3	0	0	0	0	0	0
6.4	0	0	0	0	0	0
6.5	0	0	0	0	0	0
7.1	0	0	0	0	0	0
7.2	0	0	1	1	0	0
7.3	0	0	0	0	0	0
7.4	0	0	0	0	0	0
7.5	0	0	1	1	0	0
8.1	0	0	1	1	0	0
8.2	0	0	0	0	0	0
8.3	0	0	0	0	0	0
8.4	0	0	1	1	0	0
8.5	0	0	0	0	0	0
30.1	1	1	1	1	1	1
30.2	1	1	1	1	1	1
30.3	1	1	1	1	1	1
30.4	1	1	1	1	1	1
30.5	1	1	1	1	1	1
31.1	1	1	1	1	1	1
31.2	1	1	1	1	1	1
31.3	1	1	1	1	1	1
31.4	1	1	1	1	1	1
31.5	1	1	1	1	1	1
32.1	1	1	1	1	1	1
32.2	1	1	1	1	1	1
32.3	1	1	1	1	1	1
32.4	1	1	1	1	1	1
32.5	1	1	1	1	1	1
33.1	1	1	1	1	1	1
33.2	1	1	1	1	1	1
33.3	1	1	1	1	1	1
33.4	1	1	1	1	1	1
33.5	1	1	1	1	1	1
34.1	1	1	1	1	1	1
34.2	1	1	1	1	1	1
34.3	1	1	1	1	1	1
34.4	1	1	1	1	1	1
34.5	1	1	1	1	1	1
35.1	1	1	1	1	1	1
35.2	1	1	1	1	1	1
35.3	1	1	1	1	1	1
35.4	1	1	1	1	1	1
35.5	1	1	1	1	1	1
36.1	0	0	0	0	0	0
36.2	0	1	1	1	0	0
36.3	0	0	0	0	0	0
36.4	0	0	0	0	0	0
36.5	0	0	0	0	0	0
37.1	0	0	0	0	0	0
37.2	0	0	0	0	0	0
37.3	0	0	0	0	0	0
37.4	0	0	0	0	0	0
37.5	0	1	1	1	0	0
38.1	0	0	0	0	0	0
38.2	0	0	0	0	0	0
38.3	1	1	1	1	1	1

```
    38.4     0    0    0    0    0    0
    38.5     0    0    0    0    0    0
    39.1     1    1    1    1    1    1
    39.2     1    1    1    1    1    1
    39.3     1    1    1    1    1    1
    39.4     1    1    1    1    1    1
    39.5     1    1    1    1    1    1
    40.1     1    1    1    1    1    1
    40.2     1    1    1    1    1    1
    40.3     1    1    1    1    1    1
    40.4     1    1    1    1    1    1
    40.5     1    1    1    1    1    1
    41.1     1    1    1    1    1    1
    41.2     1    1    1    1    1    1
    41.3     1    1    1    1    1    1
    41.4     1    1    1    1    1    1
    41.5     1    1    1    1    1    1
    42.1     1    1    1    1    1    1
    42.2     1    1    1    1    1    1
    42.3     1    1    1    1    1    1
    42.4     1    1    1    1    1    1
    42.5     1    1    1    1    1    1
    43.1     1    1    1    1    1    1
    43.2     1    1    1    1    1    1
    43.3     1    1    1    1    1    1
    43.4     1    1    1    1    1    1
    43.5     1    1    1    1    1    1
    44.1     1    1    1    1    1    1
    44.2     1    1    1    1    1    1
    44.3     1    1    1    1    1    1
    44.4     1    1    1    1    1    1
    44.5     1    1    1    1    1    1
    45.1     1    1    1    1    1    1
    45.2     1    1    1    1    1    1
    45.3     1    1    1    1    1    1
    45.4     1    1    1    1    1    1
    45.5     1    1    1    1    1    1
    46.1     1    1    1    1    1    1
    46.2     1    1    1    1    1    1
    46.3     1    1    1    1    1    1
    46.4     1    1    1    1    1    1
    46.5     1    1    1    1    1    1 ;
table pm(t,d,p)
             1    2    3    4    5    6
    1.1      2    2    2    2    2    2
    1.2      2    2    2    2    2    2
    1.3      4    4    4    4    4    4
    1.4      4    4    4    4    4    4
    1.5      4    4    4    4    4    4
    2.1      2    2    2    2    4    4
    2.2      2    2    2    2    4    4
    2.3      2    2    2    2    4    4
    2.4      2    2    2    2    4    4
    2.5      2    2    2    2    4    4
    3.1      2    2    2    2    2    2
    3.2      2    2    2    2    2    2
    3.3      2    2    2    2    2    2
    3.4      2    2    2    2    2    2
    3.5      2    2    2    2    2    2
    4.1      2    2    2    2    2    2
    4.2      2    2    2    2    2    2
    4.3      2    2    2    2    2    2
    4.4      2    2    2    2    2    2
    4.5      2    2    2    2    2    2
    5.1      2    2    2    2    2    2
    5.2      2    2    2    2    2    2
    5.3      2    2    2    2    2    2
    5.4      2    2    2    2    2    2
    5.5      2    2    2    2    2    2
    6.1      9    9    0    0    9    9
    6.2      9    9    0    0    9    9
    6.3      9    9    9    9    9    9
    6.4      9    9    9    9    9    9
    6.5      9    9    9    9    9    9
    7.1      9    9    9    9    9    9
    7.2      9    9    0    0    9    9
    7.3      9    9    9    9    9    9
    7.4      9    9    9    9    9    9
    7.5      9    9    0    0    9    9
    8.1      9    9    0    0    9    9
    8.2      9    9    9    9    9    9
```

8.3	9	9	9	9	9	9
8.4	9	9	0	0	9	9
8.5	9	9	9	9	9	9
30.1	2	2	2	2	2	2
30.2	2	2	2	2	2	2
30.3	2	2	2	2	2	2
30.4	2	2	2	2	2	2
30.5	2	2	2	2	2	2
31.1	2	2	2	2	2	2
31.2	2	2	2	2	2	2
31.3	2	2	2	2	2	2
31.4	2	2	2	2	2	2
31.5	6	6	6	6	6	6
32.1	2	2	2	2	2	2
32.2	2	2	2	2	2	2
32.3	4	4	4	4	4	4
32.4	2	2	2	2	2	2
32.5	2	2	2	2	2	2
33.1	2	2	2	2	2	2
33.2	2	2	2	2	2	2
33.3	2	2	2	2	2	2
33.4	2	2	2	2	2	2
33.5	6	6	6	6	6	6
34.1	2	2	2	2	2	2
34.2	2	2	2	2	2	2
34.3	2	2	2	2	2	2
34.4	2	2	2	2	2	2
34.5	6	6	6	6	6	6
35.1	2	2	2	2	2	2
35.2	2	2	2	2	4	4
35.3	4	4	4	4	4	4
35.4	6	6	6	2	2	2
35.5	2	2	2	6	6	6
36.1	9	9	9	9	9	9
36.2	9	0	0	0	9	9
36.3	9	9	9	9	9	9
36.4	9	9	9	9	9	9
36.5	9	9	9	9	9	9
37.1	9	9	9	9	9	9
37.2	9	9	9	9	9	9
37.3	9	9	9	9	9	9
37.4	9	9	9	9	9	9
37.5	9	0	0	0	9	9
38.1	9	9	9	9	9	9
38.2	9	9	9	9	9	9
38.3	0	0	0	0	0	0
38.4	9	9	9	9	9	9
38.5	9	9	9	9	9	9
39.1	2	2	2	2	2	2
39.2	4	4	4	6	6	6
39.3	6	6	6	2	2	2
39.4	2	2	2	2	2	2
39.5	6	6	6	2	2	2
40.1	2	2	2	2	2	2
40.2	2	2	2	2	2	2
40.3	4	4	4	4	4	4
40.4	2	2	2	2	2	2
40.5	2	2	2	2	2	2
41.1	2	2	2	2	2	2
41.2	4	4	4	4	4	4
41.3	2	2	2	2	2	2
41.4	2	2	2	2	2	2
41.5	2	2	2	2	2	2
42.1	2	2	2	2	2	2
42.2	6	6	6	2	2	2
42.3	2	2	2	2	2	2
42.4	4	4	4	4	4	4
42.5	2	2	2	2	2	2
43.1	2	2	2	2	2	2
43.2	2	2	2	2	2	2
43.3	2	2	2	2	2	2
43.4	2	2	2	2	2	2
43.5	2	2	2	2	2	2
44.1	2	2	2	2	2	2
44.2	2	2	2	2	2	2
44.3	2	2	2	2	2	2
44.4	2	2	2	2	2	2
44.5	2	2	2	2	2	2
45.1	4	4	2	2	2	2
45.2	4	4	2	2	2	2
45.3	4	4	2	2	2	2

```
   45.4    4    4    2    2    2    2
   45.5    4    4    2    2    2    2
   46.1    4    4    4    4    4    4
   46.2    4    4    4    4    4    4
   46.3    2    2    2    2    2    2
   46.4    2    2    2    2    2    2
   46.5    4    4    4    4    4    4   ;
variables
   y(s,t)
   v(s,t,d)
   x(s,t,d,p)
   z ;
binary variables
  y
  v
  x ;
equations
  obj
  secav(s,d,p)
  teaav(t,d,p)
  onetea1(s)
  onetea6(s)
  onetea7(s)
  onetea8(s)
  oneday(s,t,d)
  onedayf(s,t1)
  onedayth(s,t67)
  oneteatwo(s,t,d,p)
  mostper(t)
  hours4(s,t1)
  hours3(s,t67)
  hours1(s,t8)
  twohone(s,t15,d)
  twohtwo(s,t15,d)
  twohthree(s,t15,d)
  twohfour(s,t15,d)
  twohfive(s,t15,d)
  twohsix(s,t15,d)
  twohseven(s,t15,d)
  twoheight(s,t15,d)
  twohnine(s,t15,d)
  twohten(s,t15,d)
  twohele(s,t15,d)
  twohtwe(s,t15,d)
  twohthi(s,t15,d)
  twohfourt(s,t15,d)
  thrhtwo(s,t67,d)
  thrhthr(s,t67,d)
  thrhfour(s,t67,d)
  thrhfive(s,t67,d)
  thrhsix(s,t67,d)
  thrhseven(s,t67,d)
  thrheight(s,t67,d)
  thrhnine(s,t67,d)
  thrhten(s,t67,d)
  thrhele(s,t67,d)
  thrhtwe(s,t67,d)  ;
obj.. z=e=sum((s,t,d,p),pm(t,d,p)*x(s,t,d,p)) ;
secav(s,d,p).. sum(t,x(s,t,d,p))=l=sav(s,d,p) ;
teaav(t,d,p).. sum(s,x(s,t,d,p))=l=tav(t,d,p) ;
onetea1(s).. sum(t1,y(s,t1))=e=1;
onetea6(s).. sum(t6,y(s,t6))=e=1;
onetea7(s).. sum(t7,y(s,t7))=e=1;
onetea8(s).. sum(t8,y(s,t8))=e=1;
oneday(s,t,d).. v(s,t,d)=l=y(s,t) ;
onedayf(s,t1).. sum(d,v(s,t1,d))=l=2 ;
onedayth(s,t67).. sum(d,v(s,t67,d))=l=2 ;
oneteatwo(s,t,d,p).. x(s,t,d,p)=l=v(s,t,d) ;
mostper(t).. sum((s,d,p),x(s,t,d,p))=l=20  ;
hours4(s,t1).. sum((d,p),x(s,t1,d,p))=e=ha*y(s,t1) ;
hours3(s,t67).. sum((d,p),x(s,t67,d,p))=e=hc*y(s,t67) ;
hours1(s,t8).. sum((d,p),x(s,t8,d,p))=e=hd*y(s,t8) ;
twohone(s,t15,d).. sum(p,x(s,t15,d,p))=l=2;
twohtwo(s,t15,d).. x(s,t15,d,'1')-x(s,t15,d,'2')=l=0 ;
twohthree(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'3')=l=1 ;
twohfour(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'4')=l=1 ;
twohfive(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'5')=l=1 ;
twohsix(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'6')=l=1 ;
twohseven(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'4')=l=1 ;
twoheight(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'5')=l=1 ;
twohnine(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'6')=l=1 ;
```

```
twohten(s,t15,d).. x(s,t15,d,'3')+x(s,t15,d,'5')=l=1 ;
twohele(s,t15,d).. x(s,t15,d,'3')+x(s,t15,d,'6')=l=1 ;
twohtwe(s,t15,d).. x(s,t15,d,'4')+x(s,t15,d,'5')=l=1 ;
twohthi(s,t15,d).. x(s,t15,d,'4')+x(s,t15,d,'6')=l=1 ;
twohfourt(s,t15,d).. x(s,t15,d,'5')-x(s,t15,d,'6')=e=0 ;
thrhtwo(s,t67,d).. x(s,t67,d,'1')-x(s,t67,d,'2')+x(s,t67,d,'3')=l=1 ;
thrhthr(s,t67,d).. x(s,t67,d,'1')+x(s,t67,d,'4')=l=1 ;
thrhfour(s,t67,d).. x(s,t67,d,'1')+x(s,t67,d,'5')=l=1 ;
thrhfive(s,t67,d).. x(s,t67,d,'1')+x(s,t67,d,'6')=l=1 ;
thrhsix(s,t67,d).. x(s,t67,d,'2')-x(s,t67,d,'3')+x(s,t67,d,'4')=l=1 ;
thrhseven(s,t67,d).. x(s,t67,d,'2')+x(s,t67,d,'5')=l=1 ;
thrheight(s,t67,d).. x(s,t67,d,'2')+x(s,t67,d,'6')=l=1 ;
thrhnine(s,t67,d).. x(s,t67,d,'3')-x(s,t67,d,'4')+x(s,t67,d,'5')=l=1 ;
thrhten(s,t67,d).. x(s,t67,d,'3')+x(s,t67,d,'6')=l=1 ;
thrhele(s,t67,d).. x(s,t67,d,'4')-x(s,t67,d,'5')+x(s,t67,d,'6')=l=1 ;
thrhtwe(s,t67,d)..
-x(s,t67,d,'3')+x(s,t67,d,'4')+x(s,t67,d,'5')-x(s,t67,d,'6')=l=1 ;
model esra /all/ ;
OPTION OPTCR=0.0;
OPTION ITERLIM=1100000;
OPTION RESLIM=700000;
OPTION SOLPRINT=off;
OPTION LIMCOL=0;
OPTION LIMROW=0;
OPTION EJECT;
OPTION SYSOUT=off;
solve esra using mip minimizing z ;
display x.l
```

Appendix C: The Computer Input of the Model for Iteration-3

```
sets
  s /13*24/
  t /9*29/
  t2(t)/9*11/
  t3(t)/12*17/
  t4(t)/18*24/
  t5(t)/25*29/
  t15(t)/9*29/
  d /1*5/
  p /1*6/ ;
parameter
  m(t)
    /9    12
     10   14
     11   10
     12   14
     13   16
     14   14
     15   16
     16   18
     17   18
     18   18
     19   16
     20   0
     21   0
     22   18
     23   20
     24   16
     25   18
     26   12
     27   12
     28   18
     29   0  /  ;
scalar
  hb /2/ ;
table sav(s,d,p)
          1   2   3   4   5   6
  13.1    1   1   0   0   0   0
  13.2    1   1   0   0   0   0
  13.3    1   1   1   1   1   1
  13.4    0   0   1   0   0   0
  13.5    0   0   1   1   1   1
  14.1    1   1   1   1   0   0
  14.2    1   1   1   1   0   0
  14.3    1   1   1   1   1   1
  14.4    0   0   1   0   0   0
  14.5    0   0   1   1   1   1
  15.1    1   1   0   0   0   0
  15.2    1   1   1   1   0   0
  15.3    1   1   1   0   0   0
  15.4    0   0   0   0   1   1
  15.5    0   0   1   1   1   1
  16.1    1   1   1   1   1   1
  16.2    1   1   1   1   0   0
  16.3    1   1   1   0   0   0
  16.4    0   0   0   0   1   1
  16.5    0   0   1   1   1   1
  17.1    1   1   1   1   1   1
  17.2    1   1   1   1   0   0
  17.3    1   1   1   0   0   0
  17.4    0   0   0   0   1   1
  17.5    0   0   1   1   1   1
  18.1    1   1   1   1   1   1
  18.2    1   1   1   1   0   0
  18.3    1   1   1   0   0   0
  18.4    0   0   0   0   1   1
  18.5    0   0   1   1   1   1
  19.1    1   1   1   1   1   1
  19.2    0   0   0   1   1   1
  19.3    1   1   1   1   0   0
  19.4    0   0   1   1   0   0
  19.5    0   0   1   1   1   1
  20.1    1   1   1   1   1   1
  20.2    0   0   0   1   1   1
  20.3    0   0   0   1   0   0
  20.4    0   0   1   1   0   0
  20.5    0   0   1   1   1   1
  21.1    1   1   1   1   1   1
  21.2    0   0   0   1   1   1
  21.3    1   1   1   1   0   0
  21.4    0   0   1   1   0   0
```

```
     21.5     0    0    1    1    1    1
     22.1     1    1    0    0    1    1
     22.2     0    0    0    1    1    1
     22.3     1    1    1    1    0    0
     22.4     0    0    1    1    1    1
     22.5     0    0    1    1    1    1
     23.1     1    1    0    0    1    1
     23.2     1    1    1    1    1    1
     23.3     1    1    1    1    0    0
     23.4     0    0    1    0    0    0
     23.5     0    0    1    1    1    1
     24.1     1    1    0    0    1    1
     24.2     1    1    1    1    1    1
     24.3     1    1    1    1    0    0
     24.4     0    0    1    0    0    0
     24.5     0    0    1    1    1    1 ;
table tav(t,d,p)
              1    2    3    4    5    6
     9.1      1    1    1    1    1    1
     9.2      1    1    0    0    0    0
     9.3      0    0    1    1    1    1
     9.4      1    1    1    1    1    1
     9.5      1    1    1    1    0    0
     10.1     1    1    1    1    1    1
     10.2     1    1    1    1    0    0
     10.3     1    1    1    1    0    0
     10.4     1    1    1    1    1    1
     10.5     1    1    1    1    0    0
     11.1     0    0    1    1    1    1
     11.2     1    1    1    1    0    0
     11.3     1    1    1    1    0    0
     11.4     1    1    1    1    0    0
     11.5     1    1    1    1    0    0
     12.1     1    1    1    1    1    1
     12.2     1    1    1    1    1    1
     12.3     1    1    1    1    0    0
     12.4     1    1    1    1    1    1
     12.5     0    0    0    0    1    1
     13.1     1    1    1    1    1    1
     13.2     1    1    1    1    1    1
     13.3     0    0    1    1    0    0
     13.4     1    1    1    1    1    1
     13.5     1    1    1    1    1    1
     14.1     1    1    0    0    1    1
     14.2     1    1    1    1    1    1
     14.3     1    1    1    1    1    1
     14.4     0    0    1    1    0    0
     14.5     1    1    1    1    1    1
     15.1     1    1    1    1    1    1
     15.2     1    1    1    1    1    1
     15.3     1    1    1    1    1    1
     15.4     1    1    1    1    1    1
     15.5     0    0    1    1    0    0
     16.1     1    1    1    1    1    1
     16.2     1    1    1    1    1    1
     16.3     1    1    1    1    1    1
     16.4     1    1    1    1    1    1
     16.5     1    1    1    1    0    0
     17.1     1    1    1    1    1    1
     17.2     1    1    1    1    1    1
     17.3     1    1    1    1    1    1
     17.4     1    1    1    1    0    0
     17.5     1    1    1    1    1    1
     18.1     1    1    1    1    1    1
     18.2     1    1    1    1    1    1
     18.3     1    1    1    1    1    1
     18.4     1    1    1    1    1    1
     18.5     1    1    1    1    0    0
     19.1     1    1    1    1    1    1
     19.2     1    1    1    1    1    1
     19.3     1    1    1    1    1    1
     19.4     0    0    1    1    0    0
     19.5     1    1    1    1    1    1
     20.1     0    0    0    0    0    0
     20.2     0    0    0    0    0    0
     20.3     0    0    0    0    0    0
     20.4     0    0    0    0    0    0
     20.5     0    0    0    0    0    0
     21.1     0    0    0    0    0    0
     21.2     0    0    0    0    0    0
     21.3     0    0    0    0    0    0
```

```
      21.4     0    0    0    0    0    0
      21.5     0    0    0    0    0    0
      22.1     1    1    1    1    0    0
      22.2     1    1    1    1    1    1
      22.3     1    1    1    1    1    1
      22.4     1    1    1    1    1    1
      22.5     1    1    1    1    1    1
      23.1     1    1    1    1    1    1
      23.2     1    1    1    1    1    1
      23.3     1    1    1    1    1    1
      23.4     1    1    1    1    1    1
      23.5     1    1    1    1    1    1
      24.1     1    1    0    0    1    1
      24.2     1    1    1    1    1    1
      24.3     1    1    1    1    1    1
      24.4     1    1    1    1    1    1
      24.5     1    1    0    0    1    1
      25.1     1    1    1    1    1    1
      25.2     1    1    1    1    1    1
      25.3     1    1    1    1    1    1
      25.4     1    1    1    1    1    1
      25.5     1    1    1    1    0    0
      26.1     1    1    1    1    1    1
      26.2     1    1    0    0    1    1
      26.3     0    0    1    1    0    0
      26.4     0    0    1    1    1    1
      26.5     1    1    1    1    1    1
      27.1     1    1    1    1    1    1
      27.2     1    1    0    0    1    1
      27.3     1    1    1    1    0    0
      27.4     0    0    1    1    0    0
      27.5     1    1    1    1    1    1
      28.1     1    1    1    1    1    1
      28.2     1    1    1    1    1    1
      28.3     1    1    1    1    1    1
      28.4     1    1    1    1    1    1
      28.5     0    0    1    1    1    1
      29.1     0    0    0    0    0    0
      29.2     0    0    0    0    0    0
      29.3     0    0    0    0    0    0
      29.4     0    0    0    0    0    0
      29.5     0    0    0    0    0    0 ;
table pm(t,d,p)
               1    2    3    4    5    6
      9.1      4    4    4    4    4    4
      9.2      2    2    2    2    2    2
      9.3      2    2    2    2    2    2
      9.4      4    4    4    4    4    4
      9.5      2    2    2    2    2    2
      10.1     2    2    2    2    2    2
      10.2     2    2    2    2    2    2
      10.3     2    2    2    2    2    2
      10.4     2    2    2    2    2    2
      10.5     2    2    2    2    2    2
      11.1     2    2    2    2    2    2
      11.2     2    2    2    2    2    2
      11.3     2    2    2    2    2    2
      11.4     2    2    2    2    2    2
      11.5     2    2    2    2    2    2
      12.1     4    4    4    4    4    4
      12.2     4    4    4    4    4    4
      12.3     2    2    2    2    2    2
      12.4     4    4    4    4    4    4
      12.5     2    2    2    2    2    2
      13.1     4    4    4    4    4    4
      13.2     4    4    4    4    4    4
      13.3     2    2    2    2    2    2
      13.4     2    2    2    2    2    2
      13.5     2    2    2    2    2    2
      14.1     2    2    2    2    2    2
      14.2     2    2    2    2    2    2
      14.3     4    4    4    4    4    4
      14.4     2    2    2    2    2    2
      14.5     6    6    6    6    6    6
      15.1     2    2    2    2    2    2
      15.2     4    4    4    4    4    4
      15.3     4    4    4    4    4    4
      15.4     2    2    2    2    2    2
      15.5     2    2    2    2    2    2
      16.1     6    6    2    2    2    2
      16.2     6    6    2    2    2    2
```

```
    16.3     6    6    2    2    2    2
    16.4     6    6    2    2    2    2
    16.5     6    6    2    2    2    2
    17.1     2    2    2    2    2    2
    17.2     4    4    4    4    4    4
    17.3     4    4    4    4    4    4
    17.4     2    2    2    2    2    2
    17.5     6    6    6    6    6    6
    18.1     4    4    2    2    2    2
    18.2     4    4    2    2    2    2
    18.3     4    4    2    2    2    2
    18.4     4    4    2    2    2    2
    18.5     4    4    2    2    2    2
    19.1     2    2    2    2    2    2
    19.2     2    2    2    2    2    2
    19.3     4    4    4    4    4    4
    19.4     2    2    2    2    2    2
    19.5     2    2    2    2    2    2
    20.1     9    9    9    9    9    9
    20.2     9    9    9    9    9    9
    20.3     9    9    9    9    9    9
    20.4     9    9    9    9    9    9
    20.5     0    0    0    0    0    0
    21.1     0    0    0    0    0    0
    21.2     9    9    9    9    9    9
    21.3     9    9    9    9    9    9
    21.4     9    9    9    9    9    9
    21.5     9    9    9    9    9    9
    22.1     6    6    6    6    2    2
    22.2     2    2    2    2    2    2
    22.3     4    4    4    4    4    4
    22.4     6    6    6    6    6    6
    22.5     2    2    2    2    2    2
    23.1     2    2    2    2    6    6
    23.2     2    2    2    2    6    6
    23.3     2    2    2    2    6    6
    23.4     2    2    2    2    6    6
    23.5     2    2    2    2    6    6
    24.1     4    4    2    2    2    2
    24.2     6    6    6    6    6    6
    24.3     6    6    6    6    6    6
    24.4     4    4    4    4    4    4
    24.5     4    4    2    2    2    2
    25.1     2    2    2    2    2    2
    25.2     4    4    4    4    4    4
    25.3     4    4    4    4    4    4
    25.4     4    4    4    4    4    4
    25.5     2    2    2    2    2    2
    26.1     4    4    4    4    4    4
    26.2     2    2    2    2    2    2
    26.3     2    2    2    2    2    2
    26.4     2    2    2    2    2    2
    26.5     4    4    4    4    4    4
    27.1     4    4    4    4    4    4
    27.2     2    2    2    2    2    2
    27.3     2    2    2    2    2    2
    27.4     2    2    2    2    2    2
    27.5     4    4    4    6    6    6
    28.1     2    2    2    2    2    2
    28.2     2    2    2    2    2    2
    28.3     6    6    6    6    6    6
    28.4     6    6    6    6    6    6
    28.5     2    2    2    2    2    2
    29.1     9    9    9    9    9    9
    29.2     9    9    9    9    9    9
    29.3     0    0    0    0    9    9
    29.4     9    9    9    9    9    9
    29.5     9    9    9    9    9    9  ;
variables
    y(s,t)
    v(s,t,d)
    x(s,t,d,p)
    z ;
binary variables
   y
   v
   x ;
equations
   obj
   secav(s,d,p)
   teaav(t,d,p)
```

```
  onetea2(s)
  onetea3(s)
  onetea4(s)
  onetea5(s)
  oneday(s,t,d)
  onedaytw(s,t15)
  oneteatwo(s,t,d,p)
  mostper(t)
  hours2(s,t15)
  twohone(s,t15,d)
  twohtwo(s,t15,d)
  twohthree(s,t15,d)
  twohfour(s,t15,d)
  twohfive(s,t15,d)
  twohsix(s,t15,d)
  twohseven(s,t15,d)
  twoheight(s,t15,d)
  twohnine(s,t15,d)
  twohten(s,t15,d)
  twohele(s,t15,d)
  twohtwe(s,t15,d)
  twohthi(s,t15,d)
  twohfourt(s,t15,d) ;
obj.. z=e=sum((s,t,d,p),pm(t,d,p)*x(s,t,d,p)) ;
secav(s,d,p).. sum(t,x(s,t,d,p))=l=sav(s,d,p) ;
teaav(t,d,p).. sum(s,x(s,t,d,p))=l=tav(t,d,p) ;
onetea2(s).. sum(t2,y(s,t2))=e=1;
onetea3(s).. sum(t3,y(s,t3))=e=1;
onetea4(s).. sum(t4,y(s,t4))=e=1;
onetea5(s).. sum(t5,y(s,t5))=e=1;
oneday(s,t,d).. v(s,t,d)=l=y(s,t) ;
onedaytw(s,t15).. sum(d,v(s,t15,d))=l=1 ;
oneteatwo(s,t,d,p).. x(s,t,d,p)=l=v(s,t,d) ;
mostper(t).. sum((s,d,p),x(s,t,d,p))=l=m(t)  ;
hours2(s,t15).. sum((d,p),x(s,t15,d,p))=e=hb*y(s,t15) ;
twohone(s,t15,d).. sum(p,x(s,t15,d,p))=l=2;
twohtwo(s,t15,d).. x(s,t15,d,'1')-x(s,t15,d,'2')=l=0 ;
twohthree(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'3')=l=1 ;
twohfour(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'4')=l=1 ;
twohfive(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'5')=l=1 ;
twohsix(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'6')=l=1 ;
twohseven(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'4')=l=1 ;
twoheight(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'5')=l=1 ;
twohnine(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'6')=l=1 ;
twohten(s,t15,d).. x(s,t15,d,'3')+x(s,t15,d,'5')=l=1 ;
twohele(s,t15,d).. x(s,t15,d,'3')+x(s,t15,d,'6')=l=1 ;
twohtwe(s,t15,d).. x(s,t15,d,'4')+x(s,t15,d,'5')=l=1 ;
twohthi(s,t15,d).. x(s,t15,d,'4')+x(s,t15,d,'6')=l=1 ;
twohfourt(s,t15,d).. x(s,t15,d,'5')-x(s,t15,d,'6')=e=0 ;
model esra /all/ ;
OPTION OPTCR=0.0;
OPTION ITERLIM=110000;
OPTION RESLIM=700000;
OPTION SOLPRINT=off;
OPTION LIMCOL=0;
OPTION LIMROW=0;
OPTION EJECT;
OPTION SYSOUT=off;
solve esra using mip minimizing z  ;
display x.l
```

Appendix D: The Computer Input of the Model for Iteration-4

```
sets
  s /13*24/
  t /1,2,3,4,5,6,7,8,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46/
  t1(t)/1*8/
  t6(t)/30*37/
  t7(t)/38*43/
  t8(t)/44*46/
  t15(t)/1*8/
  t67(t)/30*43/
  d /1*5/
  p /1*6/ ;
scalars
  ha /4/
  hc /3/
  hd /1/ ;
parameter
  m(t)
    /1     20
     2     8
     3     4
     4     12
     5     12
     6     4
     7     0
     8     4
     30    11
     31    14
     32    20
     33    17
     34    8
     35    17
     36    3
     37    0
     38    17
     39    17
     40    14
     41    11
     42    14
     43    11
     44    15
     45    13
     46    20 /  ;
table sav(s,d,p)
              1    2    3    4    5    6
  13.1    1    1    1    1    0    0
  13.2    0    0    1    1    0    0
  13.3    1    1    0    0    0    0
  13.4    0    0    1    0    0    0
  13.5    0    0    1    1    0    0
  14.1    1    1    0    0    0    0
  14.2    0    0    1    1    0    0
  14.3    1    1    0    0    1    1
  14.4    0    0    1    0    0    0
  14.5    0    0    0    0    1    1
  15.1    0    0    1    1    0    0
  15.2    0    0    1    1    0    0
  15.3    1    1    1    0    0    0
  15.4    0    0    1    1    0    0
  15.5    0    0    0    0    1    1
  16.1    1    1    0    0    0    0
  16.2    0    0    1    1    0    0
  16.3    1    1    1    0    0    0
  16.4    0    0    0    0    0    0
  16.5    0    0    1    1    1    1
  17.1    0    0    1    1    1    1
  17.2    1    1    0    0    0    0
  17.3    1    0    0    0    0    0
  17.4    0    0    0    0    1    1
  17.5    0    0    1    1    0    0
  18.1    0    0    0    0    1    1
  18.2    1    1    1    1    0    0
  18.3    1    1    1    0    0    0
  18.4    0    0    0    0    0    0
  18.5    0    0    0    0    1    1
  19.1    1    1    1    1    1    1
  19.2    0    0    0    1    0    0
  19.3    1    1    0    0    0    0
  19.4    0    0    1    1    0    0
  19.5    0    0    0    0    0    0
  20.1    1    1    0    0    1    1
  20.2    0    0    0    1    0    0
```

```
         20.3    1    1    1    1    0    0
         20.4    0    0    0    0    0    0
         20.5    0    0    0    0    1    1
         21.1    1    1    1    1    1    1
         21.2    0    0    0    1    0    0
         21.3    1    1    1    1    0    0
         21.4    0    0    0    0    0    0
         21.5    0    0    0    0    0    0
         22.1    0    0    0    0    0    0
         22.2    0    0    0    1    1    1
         22.3    1    1    1    1    0    0
         22.4    0    0    0    0    1    1
         22.5    0    0    0    0    1    1
         23.1    0    0    0    0    0    0
         23.2    1    1    1    1    0    0
         23.3    1    1    1    1    0    0
         23.4    0    0    1    0    0    0
         23.5    0    0    1    1    0    0
         24.1    0    0    0    0    1    1
         24.2    0    0    1    1    0    0
         24.3    1    1    1    1    0    0
         24.4    0    0    1    0    0    0
         24.5    0    0    0    0    1    1 ;
table tav(t,d,p)
                   1    2    3    4    5    6
          1.1    1    1    1    1    1    1
          1.2    1    1    1    1    1    1
          1.3    1    1    1    1    1    1
          1.4    1    1    1    1    1    1
          1.5    1    1    1    1    1    1
          2.1    0    0    1    1    1    1
          2.2    1    1    1    1    1    1
          2.3    0    0    0    0    1    1
          2.4    0    0    1    1    1    1
          2.5    0    0    0    0    1    1
          3.1    0    0    1    1    0    0
          3.2    1    1    1    1    0    0
          3.3    1    0    0    1    1    1
          3.4    0    0    1    1    1    1
          3.5    0    0    0    0    0    0
          4.1    1    1    1    1    1    1
          4.2    1    1    1    1    1    1
          4.3    0    0    1    1    1    1
          4.4    0    0    1    1    1    1
          4.5    0    0    0    0    1    1
          5.1    1    1    1    1    1    1
          5.2    1    1    1    1    1    1
          5.3    0    0    0    0    1    1
          5.4    1    1    1    1    1    1
          5.5    0    0    0    0    1    1
          6.1    0    0    1    1    0    0
          6.2    0    0    1    1    0    0
          6.3    0    0    0    0    0    0
          6.4    0    0    0    0    0    0
          6.5    0    0    0    0    0    0
          7.1    0    0    0    0    0    0
          7.2    0    0    0    0    0    0
          7.3    0    0    0    0    0    0
          7.4    0    0    0    0    0    0
          7.5    0    0    0    0    0    0
          8.1    0    0    1    1    0    0
          8.2    0    0    0    0    0    0
          8.3    0    0    0    0    0    0
          8.4    0    0    1    1    0    0
          8.5    0    0    0    0    0    0
         30.1    1    0    0    0    1    1
         30.2    1    1    1    1    1    1
         30.3    1    1    1    1    1    1
         30.4    0    1    1    1    1    1
         30.5    0    0    1    0    0    0
         31.1    1    1    1    0    0    0
         31.2    1    1    1    1    1    1
         31.3    1    1    1    0    1    1
         31.4    0    0    1    1    1    1
         31.5    1    1    1    1    1    1
         32.1    1    1    1    1    1    1
         32.2    1    1    1    1    1    1
         32.3    1    1    1    1    1    1
         32.4    1    1    1    1    1    1
         32.5    1    1    1    1    1    1
         33.1    1    1    1    0    0    0
```

```
    33.2      1    1    1    1    1    1
    33.3      1    1    1    1    1    1
    33.4      1    1    1    1    1    1
    33.5      1    1    1    1    1    1
    34.1      0    0    0    1    0    0
    34.2      1    1    0    1    1    1
    34.3      1    1    1    0    0    0
    34.4      1    0    1    1    0    0
    34.5      1    1    1    1    1    1
    35.1      1    1    1    0    1    1
    35.2      1    1    0    0    1    1
    35.3      1    1    1    1    1    1
    35.4      1    1    1    1    1    1
    35.5      1    1    1    1    1    1
    36.1      0    0    0    0    0    0
    36.2      0    1    1    1    0    0
    36.3      0    0    0    0    0    0
    36.4      0    0    0    0    0    0
    36.5      0    0    0    0    0    0
    37.1      0    0    0    0    0    0
    37.2      0    0    0    0    0    0
    37.3      0    0    0    0    0    0
    37.4      0    0    0    0    0    0
    37.5      0    0    0    0    0    0
    38.1      0    0    0    0    0    0
    38.2      0    0    0    0    0    0
    38.3      1    1    1    0    0    0
    38.4      0    0    0    0    0    0
    38.5      0    0    0    0    0    0
    39.1      1    1    1    0    1    1
    39.2      1    1    1    1    1    1
    39.3      1    1    1    1    1    1
    39.4      1    1    1    1    1    1
    39.5      1    1    1    1    0    0
    40.1      0    0    0    1    1    1
    40.2      1    1    1    1    1    1
    40.3      1    1    1    1    1    1
    40.4      0    1    1    1    1    1
    40.5      0    0    1    1    1    1
    41.1      1    1    1    0    1    1
    41.2      1    1    1    1    1    1
    41.3      1    1    1    1    0    1
    41.4      0    0    1    1    1    1
    41.5      0    0    0    0    0    1
    42.1      0    1    1    1    1    0
    42.2      1    1    1    1    0    0
    42.3      1    1    0    0    1    1
    42.4      1    1    1    1    1    1
    42.5      1    1    1    1    1    1
    43.1      0    0    1    1    1    1
    43.2      1    1    0    1    1    1
    43.3      0    1    1    1    1    1
    43.4      0    0    1    1    1    1
    43.5      0    1    0    0    1    1
    44.1      1    1    0    0    1    1
    44.2      1    1    0    1    1    1
    44.3      1    1    1    0    1    1
    44.4      1    0    1    1    1    1
    44.5      1    1    1    1    1    1
    45.1      1    1    1    0    0    1
    45.2      1    1    0    1    1    1
    45.3      1    1    1    0    1    0
    45.4      1    1    1    1    1    1
    45.5      1    1    0    1    1    0
    46.1      1    1    1    1    1    1
    46.2      1    1    1    1    1    1
    46.3      1    1    1    1    1    1
    46.4      1    1    1    1    1    1
    46.5      1    1    1    1    1    1 ;
table pm(t,d,p)
               1    2    3    4    5    6
    1.1      2    2    2    2    2    2
    1.2      2    2    2    2    2    2
    1.3      4    4    4    4    4    4
    1.4      4    4    4    4    4    4
    1.5      4    4    4    4    4    4
    2.1      2    2    2    2    4    4
    2.2      2    2    2    2    4    4
    2.3      2    2    2    2    4    4
    2.4      2    2    2    2    4    4
    2.5      2    2    2    2    4    4
```

3.1	2	2	2	2	2	2
3.2	2	2	2	2	2	2
3.3	2	2	2	2	2	2
3.4	2	2	2	2	2	2
3.5	2	2	2	2	2	2
4.1	2	2	2	2	2	2
4.2	2	2	2	2	2	2
4.3	2	2	2	2	2	2
4.4	2	2	2	2	2	2
4.5	2	2	2	2	2	2
5.1	2	2	2	2	2	2
5.2	2	2	2	2	2	2
5.3	2	2	2	2	2	2
5.4	2	2	2	2	2	2
5.5	2	2	2	2	2	2
6.1	9	9	0	0	9	9
6.2	9	9	0	0	9	9
6.3	9	9	9	9	9	9
6.4	9	9	9	9	9	9
6.5	9	9	9	9	9	9
7.1	9	9	9	9	9	9
7.2	9	9	0	0	9	9
7.3	9	9	9	9	9	9
7.4	9	9	9	9	9	9
7.5	9	9	0	0	9	9
8.1	9	9	0	0	9	9
8.2	9	9	9	9	9	9
8.3	9	9	9	9	9	9
8.4	9	9	0	0	9	9
8.5	9	9	9	9	9	9
30.1	2	2	2	2	2	2
30.2	2	2	2	2	2	2
30.3	2	2	2	2	2	2
30.4	2	2	2	2	2	2
30.5	2	2	2	2	2	2
31.1	2	2	2	2	2	2
31.2	2	2	2	2	2	2
31.3	2	2	2	2	2	2
31.4	2	2	2	2	2	2
31.5	6	6	6	6	6	6
32.1	2	2	2	2	2	2
32.2	2	2	2	2	2	2
32.3	4	4	4	4	4	4
32.4	2	2	2	2	2	2
32.5	2	2	2	2	2	2
33.1	2	2	2	2	2	2
33.2	2	2	2	2	2	2
33.3	2	2	2	2	2	2
33.4	2	2	2	2	2	2
33.5	6	6	6	6	6	6
34.1	2	2	2	2	2	2
34.2	2	2	2	2	2	2
34.3	2	2	2	2	2	2
34.4	2	2	2	2	2	2
34.5	6	6	6	6	6	6
35.1	2	2	2	2	2	2
35.2	2	2	2	2	4	4
35.3	4	4	4	4	4	4
35.4	6	6	6	2	2	2
35.5	2	2	2	6	6	6
36.1	9	9	9	9	9	9
36.2	9	0	0	0	9	9
36.3	9	9	9	9	9	9
36.4	9	9	9	9	9	9
36.5	9	9	9	9	9	9
37.1	9	9	9	9	9	9
37.2	9	9	9	9	9	9
37.3	9	9	9	9	9	9
37.4	9	9	9	9	9	9
37.5	9	0	0	0	9	9
38.1	9	9	9	9	9	9
38.2	9	9	9	9	9	9
38.3	0	0	0	0	0	0
38.4	9	9	9	9	9	9
38.5	9	9	9	9	9	9
39.1	2	2	2	2	2	2
39.2	4	4	4	6	6	6
39.3	6	6	6	2	2	2
39.4	2	2	2	2	2	2
39.5	6	6	6	2	2	2
40.1	2	2	2	2	2	2

```
   40.2    2   2   2   2   2   2
   40.3    4   4   4   4   4   4
   40.4    2   2   2   2   2   2
   40.5    2   2   2   2   2   2
   41.1    2   2   2   2   2   2
   41.2    4   4   4   4   4   4
   41.3    2   2   2   2   2   2
   41.4    2   2   2   2   2   2
   41.5    2   2   2   2   2   2
   42.1    2   2   2   2   2   2
   42.2    6   6   6   2   2   2
   42.3    2   2   2   2   2   2
   42.4    4   4   4   4   4   4
   42.5    2   2   2   2   2   2
   43.1    2   2   2   2   2   2
   43.2    2   2   2   2   2   2
   43.3    2   2   2   2   2   2
   43.4    2   2   2   2   2   2
   43.5    2   2   2   2   2   2
   44.1    2   2   2   2   2   2
   44.2    2   2   2   2   2   2
   44.3    2   2   2   2   2   2
   44.4    2   2   2   2   2   2
   44.5    2   2   2   2   2   2
   45.1    4   4   2   2   2   2
   45.2    4   4   2   2   2   2
   45.3    4   4   2   2   2   2
   45.4    4   4   2   2   2   2
   45.5    4   4   2   2   2   2
   46.1    4   4   4   4   4   4
   46.2    4   4   4   4   4   4
   46.3    2   2   2   2   2   2
   46.4    2   2   2   2   2   2
   46.5    4   4   4   4   4   4   ;
variables
   y(s,t)
   v(s,t,d)
   x(s,t,d,p)
   z ;
binary variables
  y
  v
  x ;
equations
  obj
  secav(s,d,p)
  teaav(t,d,p)
  onetea1(s)
  onetea6(s)
  onetea7(s)
  onetea8(s)
  oneday(s,t,d)
  onedayf(s,t1)
  onedayth(s,t67)
  oneteatwo(s,t,d,p)
  mostper(t)
  hours4(s,t1)
  hours3(s,t67)
  hours1(s,t8)
  twohone(s,t15,d)
  twohtwo(s,t15,d)
  twohthree(s,t15,d)
  twohfour(s,t15,d)
  twohfive(s,t15,d)
  twohsix(s,t15,d)
  twohseven(s,t15,d)
  twoheight(s,t15,d)
  twohnine(s,t15,d)
  twohten(s,t15,d)
  twohele(s,t15,d)
  twohtwe(s,t15,d)
  twohthi(s,t15,d)
  twohfourt(s,t15,d)
  thrhtwo(s,t67,d)
  thrhthr(s,t67,d)
  thrhfour(s,t67,d)
  thrhfive(s,t67,d)
  thrhsix(s,t67,d)
  thrhseven(s,t67,d)
  thrheight(s,t67,d)
  thrhnine(s,t67,d)
```

```
  thrhten(s,t67,d)
  thrhele(s,t67,d)
  thrhtwe(s,t67,d) ;
obj.. z=e=sum((s,t,d,p),pm(t,d,p)*x(s,t,d,p)) ;
secav(s,d,p).. sum(t,x(s,t,d,p))=l=sav(s,d,p) ;
teaav(t,d,p).. sum(s,x(s,t,d,p))=l=tav(t,d,p) ;
onetea1(s).. sum(t1,y(s,t1))=e=1;
onetea6(s).. sum(t6,y(s,t6))=e=1;
onetea7(s).. sum(t7,y(s,t7))=e=1;
onetea8(s).. sum(t8,y(s,t8))=e=1;
oneday(s,t,d).. v(s,t,d)=l=y(s,t) ;
onedayf(s,t1).. sum(d,v(s,t1,d))=l=2 ;
onedayth(s,t67).. sum(d,v(s,t67,d))=l=2 ;
oneteatwo(s,t,d,p).. x(s,t,d,p)=l=v(s,t,d) ;
mostper(t).. sum((s,d,p),x(s,t,d,p))=l=m(t)  ;
hours4(s,t1).. sum((d,p),x(s,t1,d,p))=e=ha*y(s,t1) ;
hours3(s,t67).. sum((d,p),x(s,t67,d,p))=e=hc*y(s,t67) ;
hours1(s,t8).. sum((d,p),x(s,t8,d,p))=e=hd*y(s,t8) ;
twohone(s,t15,d).. sum(p,x(s,t15,d,p))=l=2;
twohtwo(s,t15,d).. x(s,t15,d,'1')-x(s,t15,d,'2')=l=0 ;
twohthree(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'3')=l=1 ;
twohfour(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'4')=l=1 ;
twohfive(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'5')=l=1 ;
twohsix(s,t15,d).. x(s,t15,d,'1')+x(s,t15,d,'6')=l=1 ;
twohseven(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'4')=l=1 ;
twoheight(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'5')=l=1 ;
twohnine(s,t15,d).. x(s,t15,d,'2')+x(s,t15,d,'6')=l=1 ;
twohten(s,t15,d).. x(s,t15,d,'3')+x(s,t15,d,'5')=l=1 ;
twohele(s,t15,d).. x(s,t15,d,'3')+x(s,t15,d,'6')=l=1 ;
twohtwe(s,t15,d).. x(s,t15,d,'4')+x(s,t15,d,'5')=l=1 ;
twohthi(s,t15,d).. x(s,t15,d,'4')+x(s,t15,d,'6')=l=1 ;
twohfourt(s,t15,d).. x(s,t15,d,'5')-x(s,t15,d,'6')=e=0 ;
thrhtwo(s,t67,d).. x(s,t67,d,'1')-x(s,t67,d,'2')+x(s,t67,d,'3')=l=1 ;
thrhthr(s,t67,d).. x(s,t67,d,'1')+x(s,t67,d,'4')=l=1 ;
thrhfour(s,t67,d).. x(s,t67,d,'1')+x(s,t67,d,'5')=l=1 ;
thrhfive(s,t67,d).. x(s,t67,d,'1')+x(s,t67,d,'6')=l=1 ;
thrhsix(s,t67,d).. x(s,t67,d,'2')-x(s,t67,d,'3')+x(s,t67,d,'4')=l=1 ;
thrhseven(s,t67,d).. x(s,t67,d,'2')+x(s,t67,d,'5')=l=1 ;
thrheight(s,t67,d).. x(s,t67,d,'2')+x(s,t67,d,'6')=l=1 ;
thrhnine(s,t67,d).. x(s,t67,d,'3')-x(s,t67,d,'4')+x(s,t67,d,'5')=l=1 ;
thrhten(s,t67,d).. x(s,t67,d,'3')+x(s,t67,d,'6')=l=1 ;
thrhele(s,t67,d).. x(s,t67,d,'4')-x(s,t67,d,'5')+x(s,t67,d,'6')=l=1 ;
thrhtwe(s,t67,d)..
-x(s,t67,d,'3')+x(s,t67,d,'4')+x(s,t67,d,'5')-x(s,t67,d,'6')=l=1 ;
model esra /all/ ;
OPTION OPTCR=0.0;
OPTION ITERLIM=1000000;
OPTION RESLIM=800000;
OPTION SOLPRINT=off;
OPTION LIMCOL=0;
OPTION LIMROW=0;
OPTION EJECT;
OPTION SYSOUT=off;
solve esra using mip minimizing z ;
display x.l
```

Appendix E: The Computer Output of the Model for Iteration-1

```
GAMS 2.50.094  DOS Extended/C                    11/19/01 01:01:18  PAGE      1
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
C o m p i l a t i o n


COMPILATION TIME     =        0.020 SECONDS     0.1 Mb      WAT-50-094
GAMS 2.50.094  DOS Extended/C                    11/19/01 01:01:18  PAGE      8
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
Model Statistics    SOLVE ESRA USING MIP FROM LINE 360


MODEL STATISTICS

BLOCKS OF EQUATIONS        26     SINGLE EQUATIONS      28024
BLOCKS OF VARIABLES         4     SINGLE VARIABLES       9073
NON ZERO ELEMENTS       97333     DISCRETE VARIABLES     9072


GENERATION TIME      =        1.000 SECONDS     4.4 Mb      WAT-50-094


EXECUTION TIME       =        1.000 SECONDS     4.4 Mb      WAT-50-094
GAMS 2.50.094  DOS Extended/C                    11/19/01 01:01:18  PAGE      9
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m


               S O L V E      S U M M A R Y

     MODEL   ESRA                OBJECTIVE  Z
     TYPE    MIP                 DIRECTION  MINIMIZE
     SOLVER  XA                  FROM LINE  360

**** SOLVER STATUS     1 NORMAL COMPLETION
**** MODEL STATUS      1 OPTIMAL
**** OBJECTIVE VALUE              160.0000

 RESOURCE USAGE, LIMIT       1226.130    700000.000
 ITERATION COUNT, LIMIT      9955        110000

 *************************************************************
 *  XA Professional Linear Programming System
 *     Copyright 1991,92,93,94,95,96 by Sunset Software Technology
 *     All Rights Reserved Worldwide.
 *     Phone 818-441-1565    FAX 818-441-1567
 *************************************************************
Tolerances  (OPTCA) 0    (OPTCR) 0
*** End of XA Messages ********************************


**** REPORT SUMMARY :          0     NONOPT
                               0 INFEASIBLE
                               0  UNBOUNDED
GAMS 2.50.094  DOS Extended/C                    11/19/01 01:01:18  PAGE     10
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
E x e c u t i o n


----     361 VARIABLE  X.L
```

INDEX 1 = 1

	1	2	3	4	5	6
11.3					1.000	1.000
16.5					1.000	1.000
19.4	1.000	1.000				
27.2			1.000	1.000		

INDEX 1 = 2

	1	2	3	4	5	6
9 .2			1.000	1.000		
12.3					1.000	1.000
21.1					1.000	1.000
29.3	1.000	1.000				

INDEX 1 = 3

	1	2	3	4	5	6
11.5					1.000	1.000
13.3	1.000	1.000				
22.1					1.000	1.000
29.3			1.000	1.000		

INDEX 1 = 4

	1	2	5	6
9 .5			1.000	1.000
14.4	1.000	1.000		
20.5	1.000	1.000		
27.3			1.000	1.000

INDEX 1 = 5

	1	2	5	6
10.3			1.000	1.000
15.5			1.000	1.000
21.1	1.000	1.000		
28.5	1.000	1.000		

INDEX 1 = 6

	1	2	3	4	5	6
11.1	1.000	1.000				
14.1			1.000	1.000		

GAMS 2.50.094 DOS Extended/C 11/19/01 01:01:18 PAGE 11
G e n e r a l A l g e b r a i c M o d e l i n g S y s t e m
E x e c u t i o n

361 VARIABLE X.L

INDEX 1 = 6

	1	2	3	4	5	6
18.5					1.000	1.000
27.4	1.000	1.000				

INDEX 1 = 7

	3	4	5	6
10.2			1.000	1.000
12.5	1.000	1.000		
24.1	1.000	1.000		
27.4			1.000	1.000

INDEX 1 = 8

	1	2	5	6
11.4			1.000	1.000
15.5	1.000	1.000		
20.5			1.000	1.000
26.3			1.000	1.000

INDEX 1 = 9

	5	6
11.2	1.000	1.000
13.3	1.000	1.000
19.4	1.000	1.000
25.5	1.000	1.000

INDEX 1 = 10

	1	2	3	4	5	6
9 .2					1.000	1.000
12.5	1.000	1.000				
21.1			1.000	1.000		

	1	2	3	4	5	6
26.2			1.000	1.000		

INDEX 1 = 11

	1	2	3	4	5	6
9 .3	1.000	1.000				
14.4					1.000	1.000
20.5			1.000	1.000		
26.4	1.000	1.000				

361 VARIABLE X.L

INDEX 1 = 12

	1	2	3	4	5	6
10.5					1.000	1.000
17.4					1.000	1.000
24.5			1.000	1.000		
26.3	1.000	1.000				

EXECUTION TIME = 0.090 SECONDS 2.8 Mb WAT-50-094

USER: Nijaz Bajgoric (PC system #1) G981007:1547AS-WAT
Bogazici University DC1736

**** FILE SUMMARY

INPUT C:\GAMS\ESRA1.C
OUTPUT C:\GAMS\ESRA1.LST

Appendix F: The Computer Output of the Model for Iteration-2

```
GAMS 2.50.094  DOS Extended/C                       11/19/01 02:00:05  PAGE      1
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
C o m p i l a t i o n


COMPILATION TIME     =        0.020 SECONDS      0.1 Mb      WAT-50-094
GAMS 2.50.094  DOS Extended/C                       11/19/01 02:00:05  PAGE     10
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
Model Statistics    SOLVE ESRA USING MIP FROM LINE 431


MODEL STATISTICS

BLOCKS OF EQUATIONS        40     SINGLE EQUATIONS     28208
BLOCKS OF VARIABLES         4     SINGLE VARIABLES     10801
NON ZERO ELEMENTS      106513     DISCRETE VARIABLES   10800


GENERATION TIME      =        0.870 SECONDS      5.0 Mb      WAT-50-094


EXECUTION TIME       =        0.880 SECONDS      5.0 Mb      WAT-50-094
GAMS 2.50.094  DOS Extended/C                       11/19/01 02:00:05  PAGE     11
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m


               S O L V E      S U M M A R Y

     MODEL   ESRA                OBJECTIVE  Z
     TYPE    MIP                 DIRECTION  MINIMIZE
     SOLVER  XA                  FROM LINE  431

**** SOLVER STATUS     2 ITERATION INTERRUPT
**** MODEL STATUS      8 INTEGER SOLUTION
**** OBJECTIVE VALUE              244.0000

 RESOURCE USAGE, LIMIT       42807.160    700000.000
 ITERATION COUNT, LIMIT    1100365       1100000

 ***************************************************************
 *  XA Professional Linear Programming System
 *     Copyright 1991,92,93,94,95,96 by Sunset Software Technology
 *     All Rights Reserved Worldwide.
 *     Phone 818-441-1565   FAX 818-441-1567
 ***************************************************************

Note 1. XA terminated for other reasons than
        reaching an optimal solution.

Tolerances  (OPTCA) 0    (OPTCR) 0
*** End of XA Messages ******************************


**** REPORT SUMMARY :          0     NONOPT
                               0 INFEASIBLE
                               0  UNBOUNDED
GAMS 2.50.094  DOS Extended/C                       11/19/01 02:00:05  PAGE     12
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
E x e c u t i o n


----     432 VARIABLE  X.L

INDEX 1 = 1

                1            2            3            4            5            6

3 .3                     1.000        1.000
3 .5        1.000        1.000
31.1                                               1.000        1.000        1.000
43.3        1.000
43.5                                  1.000        1.000
45.3                                               1.000

INDEX 1 = 2

                1            2            3            4            5            6
```

	1	2	3	4	5	6
2 .3			1.000	1.000		
2 .4	1.000	1.000				
30.5				1.000	1.000	1.000
41.1				1.000		
41.5	1.000	1.000				
45.5			1.000			

INDEX 1 = 3

	1	2	3	4	5	6
7 .2			1.000	1.000		
7 .5			1.000	1.000		
34.3					1.000	1.000
34.4		1.000				
40.4	1.000					
40.5	1.000	1.000				
44.1				1.000		

INDEX 1 = 4

	1	2	3	4	5	6
5 .3	1.000	1.000				
5 .5			1.000	1.000		
35.1				1.000		
35.2			1.000	1.000		
42.1						1.000
42.2					1.000	1.000
45.1					1.000	

GAMS 2.50.094 DOS Extended/C 11/19/01 02:00:05 PAGE 13
G e n e r a l A l g e b r a i c M o d e l i n g S y s t e m
E x e c u t i o n

432 VARIABLE X.L

INDEX 1 = 5

	1	2	3	4	5	6
4 .3	1.000	1.000				
4 .5			1.000	1.000		
33.1				1.000	1.000	1.000
43.2			1.000			
43.4	1.000	1.000				
44.1			1.000			

INDEX 1 = 6

	1	2	3	4	5	6
2 .3	1.000	1.000				
2 .5	1.000	1.000				
34.1					1.000	1.000
34.2			1.000			
41.3					1.000	
41.5			1.000	1.000		
45.3						1.000

INDEX 1 = 7

	1	2	4	5	6
3 .1	1.000	1.000			
3 .5				1.000	1.000
30.4	1.000				
30.5	1.000	1.000			
38.3			1.000	1.000	1.000
44.4		1.000			

INDEX 1 = 8

	1	2	3	4	5	6
3 .2					1.000	1.000
3 .5			1.000	1.000		
31.3				1.000		
31.4	1.000	1.000				
40.1	1.000	1.000	1.000			

```
45.1                                              1.000

INDEX 1 = 9

               1           2           3           4

3 .1                                  1.000       1.000
3 .4        1.000       1.000
□GAMS 2.50.094  DOS Extended/C                  11/19/01 02:00:05  PAGE     14
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
E x e c u t i o n


        432 VARIABLE  X.L

INDEX 1 = 9

               1           2           3           4

37.5                    1.000       1.000       1.000
43.1        1.000       1.000
43.5        1.000
44.3                                              1.000

INDEX 1 = 10

               1           2           3           4           5           6

2 .1        1.000       1.000
2 .5                                1.000       1.000
34.3                                            1.000
34.4                                                        1.000       1.000
41.4        1.000       1.000
41.5                                                        1.000
45.5                                                                    1.000

INDEX 1 = 11

               1           2           3           4           5           6

5 .3                                1.000       1.000
5 .5        1.000       1.000
34.1        1.000       1.000       1.000
39.1                                            1.000
39.5                                                        1.000       1.000
45.2                                1.000

INDEX 1 = 12

               1           2           3           4

4 .4        1.000       1.000
4 .5        1.000       1.000
30.1                    1.000       1.000       1.000
42.1        1.000
42.3                                1.000       1.000
44.2                                1.000


EXECUTION TIME       =       0.100 SECONDS     2.9 Mb      WAT-50-094
□GAMS 2.50.094  DOS Extended/C                  11/19/01 02:00:05  PAGE     15
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m


USER: Nijaz Bajgoric (PC system #1)                 G981007:1547AS-WAT
      Bogazici University                                       DC1736

**** FILE SUMMARY

INPUT      C:\GAMS\ESRA1G.C
OUTPUT     C:\GAMS\ESRA1G.LST
```

Appendix G: The Computer Output of the Model for Iteration-3

```
□GAMS 2.50.094  DOS Extended/C                      11/23/01 17:16:14  PAGE       1
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
C o m p i l a t i o n


COMPILATION TIME     =        0.020 SECONDS     0.1 Mb      WAT-50-094
□GAMS 2.50.094  DOS Extended/C                      11/23/01 17:16:14  PAGE       9
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
Model Statistics    SOLVE ESRA USING MIP FROM LINE 383



MODEL STATISTICS

BLOCKS OF EQUATIONS      26     SINGLE EQUATIONS    28024
BLOCKS OF VARIABLES       4     SINGLE VARIABLES     9073
NON ZERO ELEMENTS     97333     DISCRETE VARIABLES   9072


GENERATION TIME      =        0.910 SECONDS     4.4 Mb      WAT-50-094


EXECUTION TIME       =        0.910 SECONDS     4.4 Mb      WAT-50-094
□GAMS 2.50.094  DOS Extended/C                      11/23/01 17:16:14  PAGE      10
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m


               S O L V E      S U M M A R Y

     MODEL   ESRA                OBJECTIVE  Z
     TYPE    MIP                 DIRECTION  MINIMIZE
     SOLVER  XA                  FROM LINE  383

**** SOLVER STATUS     1 NORMAL COMPLETION
**** MODEL STATUS      1 OPTIMAL
**** OBJECTIVE VALUE              192.0000

 RESOURCE USAGE, LIMIT        1040.000    700000.000
 ITERATION COUNT, LIMIT      10530        110000

 *************************************************************
 *  XA Professional Linear Programming System
 *     Copyright 1991,92,93,94,95,96 by Sunset Software Technology
 *     All Rights Reserved Worldwide.
 *     Phone 818-441-1565   FAX 818-441-1567
 *************************************************************
Tolerances  (OPTCA) 0    (OPTCR) 0
*** End of XA Messages ******************************


**** REPORT SUMMARY :          0     NONOPT
                               0 INFEASIBLE
                               0  UNBOUNDED
□GAMS 2.50.094  DOS Extended/C                      11/23/01 17:16:14  PAGE      11
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
E x e c u t i o n


----     384 VARIABLE  X.L

INDEX 1 = 13

               1           2           3           4           5           6

16.3                                                       1.000       1.000
24.5                                                       1.000       1.000
10.2       1.000       1.000
26.3                               1.000       1.000

INDEX 1 = 14

               1           2           3           4

13.3                               1.000       1.000
19.2       1.000       1.000
10.1                               1.000       1.000
28.5                               1.000       1.000
```

INDEX 1 = 15

	1	2	3	4	5	6
16.4					1.000	1.000
19.1	1.000	1.000				
11.5			1.000	1.000		
27.2	1.000	1.000				

INDEX 1 = 16

	1	2	3	4	5	6
15.4					1.000	1.000
19.1			1.000	1.000		
9 .2	1.000	1.000				
25.1					1.000	1.000

INDEX 1 = 17

	1	2	3	4	5	6
17.1	1.000	1.000				
22.2			1.000	1.000		
11.3		1.000	1.000			
28.5					1.000	1.000

INDEX 1 = 18

	1	2	3	4	5	6
17.1			1.000	1.000		
23.5			1.000	1.000		

GAMS 2.50.094 DOS Extended/C 11/23/01 17:16:14 PAGE 12
G e n e r a l A l g e b r a i c M o d e l i n g S y s t e m
E x e c u t i o n

384 VARIABLE X.L

INDEX 1 = 18

	1	2	3	4	5	6
10.4					1.000	1.000
28.1	1.000	1.000				

INDEX 1 = 19

	3	4	5	6
16.3	1.000	1.000		
19.5			1.000	1.000
10.5	1.000	1.000		
26.2			1.000	1.000

INDEX 1 = 20

	3	4	5	6
15.5	1.000	1.000		
22.2			1.000	1.000
10.4	1.000	1.000		
28.1	1.000	1.000		

INDEX 1 = 21

	3	4	5	6
19.2			1.000	1.000
9 .5	1.000	1.000		
12.5			1.000	1.000
27.4	1.000	1.000		

INDEX 1 = 22

	1	2	3	4	5	6
16.1					1.000	1.000
23.1	1.000	1.000				
11.4			1.000	1.000		

```
25.5                                  1.000       1.000

INDEX 1 = 23

                1           2           5           6

16.2                                  1.000       1.000
22.5                                  1.000       1.000
11.1                                  1.000       1.000
25.1         1.000       1.000
GAMS 2.50.094  DOS Extended/C                  11/23/01 17:16:14   PAGE      13
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
E x e c u t i o n


        384 VARIABLE  X.L

INDEX 1 = 24

                1           2           3           4           5           6

15.1         1.000       1.000
18.5                                  1.000       1.000
11.2         1.000       1.000
27.2                                                          1.000       1.000


EXECUTION TIME       =        0.100 SECONDS     2.7 Mb      WAT-50-094


USER: Nijaz Bajgoric (PC system #1)                  G981007:1547AS-WAT
      Bogazici University                                        DC1736

**** FILE SUMMARY

INPUT       C:\GAMS\ESRAMC.C
OUTPUT      C:\GAMS\ESRAMC.LST
```

Appendix H: The Computer Output of the Model for Iteration-4

```
GAMS 2.50.094  DOS Extended/C                      11/23/01 19:31:09  PAGE       1
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
C o m p i l a t i o n


COMPILATION TIME     =        0.020 SECONDS      0.1 Mb      WAT-50-094
GAMS 2.50.094  DOS Extended/C                      11/23/01 19:31:09  PAGE      10
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
Model Statistics    SOLVE ESRA USING MIP FROM LINE 458


MODEL STATISTICS

BLOCKS OF EQUATIONS       40     SINGLE EQUATIONS     28208
BLOCKS OF VARIABLES        4     SINGLE VARIABLES     10801
NON ZERO ELEMENTS     106513     DISCRETE VARIABLES   10800


GENERATION TIME      =        0.770 SECONDS      5.0 Mb      WAT-50-094


EXECUTION TIME       =        0.780 SECONDS      5.0 Mb      WAT-50-094
GAMS 2.50.094  DOS Extended/C                      11/23/01 19:31:09  PAGE      11
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m


               S O L V E      S U M M A R Y

     MODEL   ESRA                OBJECTIVE  Z
     TYPE    MIP                 DIRECTION  MINIMIZE
     SOLVER  XA                  FROM LINE  458

**** SOLVER STATUS     2 ITERATION INTERRUPT
**** MODEL STATUS      8 INTEGER SOLUTION
**** OBJECTIVE VALUE              260.0000

 RESOURCE USAGE, LIMIT      21638.280    800000.000
 ITERATION COUNT, LIMIT   1001239        1000000

 ************************************************************
 *  XA Professional Linear Programming System
 *     Copyright 1991,92,93,94,95,96 by Sunset Software Technology
 *     All Rights Reserved Worldwide.
 *     Phone 818-441-1565   FAX 818-441-1567
 ************************************************************

Note 1. XA terminated for other reasons than
        reaching an optimal solution.

Tolerances  (OPTCA) 0    (OPTCR) 0
*** End of XA Messages ******************************


**** REPORT SUMMARY :        0     NONOPT
                             0 INFEASIBLE
                             0  UNBOUNDED
GAMS 2.50.094  DOS Extended/C                      11/23/01 19:31:09  PAGE      12
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
E x e c u t i o n


----    459 VARIABLE  X.L

INDEX 1 = 13

              1           2           3           4

6 .1                               1.000       1.000
6 .2                               1.000       1.000
32.1       1.000       1.000
32.4                               1.000
40.3       1.000
40.5                               1.000       1.000
44.3                   1.000

INDEX 1 = 14
```

	1	2	3	4	5	6
2 .2			1.000	1.000		
2 .5					1.000	1.000
33.1	1.000	1.000				
33.3	1.000					
43.3					1.000	1.000
43.4			1.000			
46.3		1.000				

INDEX 1 = 15

	1	2	3	4	5	6
8 .1			1.000	1.000		
8 .4			1.000	1.000		
31.2			1.000	1.000		
31.3			1.000			
41.3	1.000	1.000				
41.5						1.000
45.5					1.000	

INDEX 1 = 16

	1	2	3	4	5	6
5 .1	1.000	1.000				
5 .2			1.000	1.000		
30.3	1.000	1.000				
30.5			1.000			
43.3			1.000			
43.5					1.000	1.000
44.5				1.000		

459 VARIABLE X.L

INDEX 1 = 17

	1	2	3	4	5	6
4 .2	1.000	1.000				
4 .4					1.000	1.000
32.1				1.000	1.000	1.000
42.1			1.000			
42.5			1.000	1.000		
44.3	1.000					

INDEX 1 = 18

	1	2	3	4	5	6
5 .1					1.000	1.000
5 .5					1.000	1.000
36.2		1.000	1.000	1.000		
38.3	1.000	1.000	1.000			
44.2	1.000					

INDEX 1 = 19

	1	2	3	4	5	6
1 .1					1.000	1.000
1 .4			1.000	1.000		
35.1	1.000	1.000	1.000			
42.1				1.000		
42.3	1.000	1.000				
44.2				1.000		

INDEX 1 = 20

	1	2	3	4	5	6
4 .3			1.000	1.000		
4 .5					1.000	1.000
35.1					1.000	1.000
35.3	1.000					
39.1	1.000	1.000				

	1	2	3	4	5	6
39.3		1.000				
45.2				1.000		

INDEX 1 = 21

	1	2	3	4	5	6
1 .1			1.000	1.000		
1 .3			1.000	1.000		
31.1		1.000				

459 VARIABLE X.L

INDEX 1 = 21

	1	2	3	4	5	6
31.3	1.000	1.000				
40.1					1.000	1.000
40.2				1.000		
44.1	1.000					

INDEX 1 = 22

	1	2	3	4	5	6
5 .2					1.000	1.000
5 .4					1.000	1.000
34.3	1.000	1.000	1.000			
42.2				1.000		
42.5					1.000	1.000
46.3				1.000		

INDEX 1 = 23

	1	2	3	4
1 .2	1.000	1.000		
1 .3	1.000	1.000		
32.2			1.000	1.000
32.5			1.000	
41.3			1.000	1.000
41.4			1.000	
45.5				1.000

INDEX 1 = 24

	1	2	3	4	5	6
4 .1					1.000	1.000
4 .2			1.000	1.000		
33.3		1.000	1.000	1.000		
40.4			1.000			
40.5					1.000	1.000
46.3	1.000					

EXECUTION TIME = 0.100 SECONDS 2.9 Mb WAT-50-094

USER: Nijaz Bajgoric (PC system #1) G981007:1547AS-WAT
Bogazici University DC1736

**** FILE SUMMARY

INPUT C:\GAMS\ESRAMD.C
OUTPUT C:\GAMS\ESRAMD.LST

REFERENCES

Abramson, D., (1991). "Constructing school timetables using simulated annealing: Sequential and parallel algorithms". *Management Science.*, Vol. 37, No. 1, pp. 98-113.

Alvarez-Valdes, R. et al. (1996). "Constructing good solutions for the Spanish school timetabling problem". *Journal of the Operational Research Society,* 47, pp. 1203-1215.

Aubin, J. and Ferland J. A., (1989). "A large scale timetabling problem". *Computers and Operations Research.,* Vol. 16, No. 1, pp. 67-77.

Bezeau, L. M., (1993). "Timetabling an academic department with linear programming". Available on site *http://www.unb.ca/centres/nbcea/lptimtab.html.*

Birbas, T. et al. (1997). "Timetabling for Greek high schools". *Journal of the Operational Research Society,* 48, pp. 1191-2000.

Burke, E. K. and Elliman, D. G., (1994). "A university timetabling system based on graph colouring and constraint manipulation". *Journal of Research on Computing in Education,* Vol. 27, No. 1, pp. 1-18.

Chahal, N. and Werra, D. de, (1989). "An interactive system for constructing timetables on a PC". *European Journal of Operational Research,* 40, pp. 32-37.

Dimopoulou, M. and Miliotis P., (2001). "Implementation of a university course and examination timetabling system". *European Journal of Operational Research,* 130, pp. 202-213.

Dowsland, K. A., (1990). "A timetabling problem in which clashes are inevitable". *Journal of the Operational Research Society,* Vol. 41, No. 10, pp. 907-918.

Drexl, A. and Salewski F., (1997). "Distribution requirements and compactness constraints in school timetabling". *European Journal of Operational Research,* 102, pp. 193-214.

Fang, H. L., (1994). "Genetic Algorithms in Timetabling and Scheduling". Available on site *http://citeseer.nj.nec.com/fang94genetic.html.*

Goltz, H. J. et al. (1997). "Constraint - based timetabling for universities". Available on site *http://citeseer.nj.nec.com/124105.html.*

Hertz, A., (1991). "Tabu search for large scale timetabling problems". *European Journal of Operational Research,* 54, pp. 39-47.

Hultberg, T. H. and Cardoso, D. M., (1997). "The teacher assignment problem: A special case of the fixed charge transportation problem". *European Journal of Operational Research,* 101, pp. 463-473.

Johnson, D, (1993). "A database approach to course timetabling". *Journal of the Operational Research Society,* Vol. 44, pp. 425-433.

Laporte, G. and Desroches S., (1986). "The problem of assigning students to course sections in a large engineering school". *Computers and Operations Research.,* Vol. 13, No. 4, pp. 387-394.

Schaerf, A., (1996). "Tabu search techniques for large high-school timetabling problems". Available on site *http://citeseer.nj.nec.com/schaerf96tabu.html.*

Schmidt, G. and Ströhlein, T., (1980). "timetable construction – an annotated bibliography". *The Computer Journal,* Vol. 23, No. 4, pp. 307-316.

Tripathy, A, (1980). "A Lagrangian relaxation approach to course timetabling". *Journal of the Operational Research Society,* 47, pp. 347-357.

Werra, D. de, (1985). "An introduction to timetabling". *European Journal of Operational Research,* 19, pp. 151-162.

Werra, D. de, (1997). "The combinatorics of timetabling". *European Journal of Operational Research,* 96, pp. 504-513.

Wright, M., (1996). "School timetabling using Heuristic Search". *Journal of the Operational Research Society,* Vol. 31, pp. 599-603.

Wright, M., (2001). "Subcost-guided search – Experiments with timetabling problems". *Journal of Heuristics,* 7, pp. 599-603.

CV OF THE AUTHOR

A. CEM ÜSTÜNEL

Personal Details:

Date of Birth	: February 05, 1969
Place of Birth	: Kütahya
Marital Status	: Married

Education:

High School	: 1983-1987	Kuleli Military High School
Bachelor's degree	: 1987-1991	Turkish Military Academy
Master's degree	: 2000-2001	Yeditepe University Graduate Institute of Science and Engineering, Systems Engineering Department

Duty Adress:

1991-Continued Turkish Armed Forces