



HAL
open science

Modeling and Solving A Crew Assignment Problem in Air Transportation

Farah Zeghal Mansour, Michel Minoux

► **To cite this version:**

Farah Zeghal Mansour, Michel Minoux. Modeling and Solving A Crew Assignment Problem in Air Transportation. European Journal of Operational Research, 2006, 175 (1), pp.187-209. 10.1016/j.ejor.2004.11.028 . hal-01146497

HAL Id: hal-01146497

<https://hal.science/hal-01146497>

Submitted on 6 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Modeling and solving a Crew Assignment Problem in air transportation

F.M. Zeghal^{b,c,*}, M. Minoux^a

^a *Université Paris-6, LIP6, 4 Place Jussieu, 75005 Paris, France*

^b *Ecole Polytechnique de Tunisie, CORG-ROI, BP 743, 2078, La Marsa, Tunisia*

^c *Ecole Nationale d'Ingénieurs de Tunis, Department of Industrial Engineering, BP 37, Le Belvédère, 1002 Tunis, Tunisia*

A typical problem arising in airline crew management consists in optimally assigning the required crew members to each flight segment of a given time period, while complying with a variety of work regulations and collective agreements. This problem called the *Crew Assignment Problem (CAP)* is currently decomposed into two independent sub-problems which are modeled and solved sequentially: (a) the well-known *Crew Pairing Problem* followed by (b) the *Working Schedules Construction Problem*. In the first sub-problem, a set of legal minimum-cost pairings is constructed, covering all the planned flight segments. In the second sub-problem, pairings, rest periods, training periods, annual leaves, etc. are combined to form working schedules which are then assigned to crew members.

In this paper, we present a new approach to the Crew Assignment Problem arising in the context of airline companies operating short and medium haul flights. Contrary to most previously published work on the subject, our approach is not based on the concept of crew-pairings, though it is capable of handling many of the constraints present in crew-pairing-based models. Moreover, contrary to crew-pairing-based approaches, one of its distinctive features is that it formulates and solves the two sub-problems (a) and (b) simultaneously for the technical crew members (pilots and officers) with specific constraints. We show how this problem can be formulated as a large scale integer linear program with a general structure combining different types of constraints and not exclusively partitioning or covering constraints as usually suggested in previous papers. We introduce then, a formulation enhancement phase where we replace a large number of binary exclusion constraints by stronger and less numerous ones: the clique constraints. Using data provided by the Tunisian airline company TunisAir, we demonstrate that thanks to this new formulation, the Crew Assignment Problem can be solved by currently available integer linear programming technology. Finally, we propose an efficient heuristic method based on a rounding strategy embedded in a partial tree search procedure.

* Corresponding author. Address: Ecole Polytechnique de Tunisie, CORG-ROI, BP 743, 2078, La Marsa, Tunisia.
E-mail addresses: farah_zeghal@yahoo.fr (F.M. Zeghal), michel.minoux@lip6.fr (M. Minoux).

The implementation of these methods (both exact and heuristic ones) provides good solutions in reasonable computation times using CPLEX 6.0.2: guaranteed exact solutions are obtained for 60% of the test instances and solutions within 5% of the lower bound for the others.

Keywords: Crew assignment; Combinatorial optimization; Integer linear programming; Heuristic; Air transportation

1. Introduction

For airline companies, the Crew Assignment Problem (CAP) is an economically significant issue in today's highly competitive market. Crew costs constitute one of the largest components of direct operating costs and are only dominated by fixed aircraft costs and fuel consumption costs. Thus, important savings can be generated by solving the Crew Assignment Problem optimally.

This problem is commonly decomposed into two independent sub-problems which are modeled and solved sequentially:

- (a) The well-known *Crew Pairing Problem*: it consists in generating a set of minimal cost crew pairings covering all the planned flight segments. A crew pairing is a sequence of flight segments separated by connections or rest periods, operated by a crew leaving and returning to the same crew home base.
- (b) The *Working Schedules Construction Problem*: it aims at constructing working schedules for crew members by assigning them pairings, resulting from the previous step (a), rest periods, training periods, annual leaves, etc.

The solution to these two sub-problems (a) and (b) must satisfy all the operational constraints deriving from the current regulation and the collective agreements.

The Crew Assignment Problem is very difficult to solve because of the highly combinatorial nature of the two sub-problems (a) and (b), the huge size of the corresponding formulations and the complexity of the regulation rules and the collective agreements that must be taken into account.

The Crew Pairing Problem and the Working Schedules Construction Problem have been extensively studied for several decades. These problems are usually formulated as Set Partitioning or Set Covering problems, where variables correspond to feasible pairings for the first problem and to feasible working schedules for the second problem.

To solve real-life Crew Pairing Problems and Working Schedules Construction Problems, various approaches have been suggested based on column-generation principle (Minoux, 1984; Lavoie et al., 1988; Gamache et al., 1994) which has been subsequently integrated in the "Branch-and-Bound" method leading to the so-called "Branch-and-Price" method (Ribeiro et al., 1989; Desaulniers et al., 1997; Vance et al., 1997; Barnhart and Shenoi, 1998; Gamache et al., 1998; Stojkovic et al., 1998; Lettovsky et al., 2000).

A "Branch-and-Cut" method has been proposed by Hoffman and Padberg (1993) for the Crew Pairing Problem: it consists in integrating a cutting plane strategy in the "Branch-and-Bound" method.

1.1. Contributions

In the present paper, we study the Crew Assignment Problem faced by airline companies operating short and medium haul flights. For this case, we propose a new approach that formulates and solves the two sub-problems (a) and (b) simultaneously for the technical crew members (pilots and officers):

- The Crew Assignment Problem is formulated as a large scale integer linear program with a general structure combining different types of constraints (and not exclusively partitioning or covering constraints as usually suggested in previous papers).
- This formulation is then enhanced by replacing a large number of binary exclusion constraints by stronger and less numerous ones: the clique constraints. A special feature of our approach is that *clique constraints corresponding to all maximal cliques in the constraint graph* are generated. In spite of the large size of this constraint graph, this is made possible by a proper exploitation of the fact that it is *close to an interval graph* (see Section 3.3).
- We show that thanks to this new formulation, the Crew Assignment Problem can be solved by currently available integer linear programming technology (CPLEX 6.0.2).
- We develop a new heuristic method based on a rounding strategy embedded in a partial tree search procedure. As will appear from the computational experiments, this heuristic turns out to be more efficient on the test problems considered than the standard heuristic of CPLEX 6.0.2.

Contrary to most previously published work on the subject, the model presented here is not based on the classical concept of crew-pairing, namely a sequence of flight services carried out by a given crew. Indeed, in our model, the decision variables relate to how an individual crew member is assigned to flight services. The constraints accounting for the impossibility, for a given individual crew member, to carry out two given successive distinct flight services (due e.g. to insufficient rest time, or any other reason) are actually included in our model in the form of binary exclusion (incompatibility) constraints. So, in a specific manner, our model can (does) take into account many of the constraints which contribute to the definition of valid crew pairings in classical crew-pairing-based approaches. On the other hand, our model appears to be more flexible at least in the following precise sense: in crew-pairing-based models, it is implicitly assumed that the same crew (i.e. the same set of technical crew members) has to be assigned to every flight service and thus to every flight forming a crew pairing. By contrast, in our model, if Pilot A and co-pilot B are assigned to e.g. flight service 1, it is possible that Pilot A is assigned together with another co-pilot, say co-pilot C, to a subsequent flight service. *This kind of flexibility is not offered in crew-pairing-based models.*

Of course, the model discussed and solved in the present paper, while certainly representing an example of problems arising in the context of crew management in airline companies, does not pretend to cover all kinds of problems in this class. Indeed, it is well-known that, from one company to the next, many differences can arise, due to distinct priorities in the objective to be optimized, to distinct regulation rules, etc. We do feel however that, beyond the specific application developed here (case of TunisAir), the general methodology proposed, using state-of-the-art integer programming concepts and tools, has a real potential for wider applicability.

1.2. Outline of the paper

The paper is organized as follows. The next section describes the Crew Assignment Problem treated here. Then, Section 3 presents the proposed integer linear programming formulation of the problem. Next, Section 4 discusses the computational results obtained by exact methods to solve 20 real problems provided by the Tunisian airline company TunisAir. Section 5 describes a new heuristic method and its implementation to solve real problems. A comparison between exact and heuristic solution methods is presented in this section too. Finally, we present some possible extensions to the proposed model in Section 6, and some conclusions and perspectives in Section 7.

2. The Crew Assignment Problem

The Crew Assignment Problem can be stated as follows.

An airline company has to operate, within some given time period (one week, one month, etc.), a number of flight segments of various origins and destinations. To operate all these planned flight segments, the company has to assign them the appropriate crew members (number and qualifications), while taking into account a variety of work regulations and collective agreements.

To formulate and solve this problem, the approach proposed here consists of two distinct phases:

- Phase 1. Flight service generation: a flight service is a sequence of flight segments separated by connections, followed and, in some cases, preceded by a rest period. In this phase, a set of flight services is built, covering at least once each of the planned flight segments. All the flight services generated must satisfy a set of regulation rules and collective agreements.
- Phase 2. Crew assignment to flight services: it aims at assigning the required crew members to the flight services previously generated in Phase 1, covering all the flight segments while satisfying the current regulation and the collective agreements.

2.1. Phase 1: Flight service generation

To generate all valid flight services from the given set of planned flight segments, various constraints must be taken into account. These constraints result from the current regulation (briefing time, minimum connection time, maximal duration of a flight service, maximal duration of flying, maximal duration of night flying, minimum number of hours of rest between flight services, etc.) and collective agreements (leaving late for far away destinations, maximum connection time, etc.). Thus, the problem of building flight services from flight segments is highly constrained, and the total number of flight segments which compose a flight service rarely exceeds 3 or 4. As a result, even for the largest real problems, complete enumeration of all the flight services turns out to be possible (Table 2 in Section 4 shows that the number of flight services is of the same order of magnitude as the number of flight segments).

Flight services are composed of consecutive flight segments which can be performed in succession without the need of deadhead flights. Deadheads or deadhead flights are passive flight segments used to transport crew members between stations, either by ground or air transportation. For flight services, since there are no constraints on their starting and ending stations, these may be distinct from crew bases. So, we suppose that it is always possible to reposition the crew members by using deadheads for transporting them from their location (crew base or ending station of a flight service) to the starting station of a flight service or to their crew base.

In order to construct working schedules, we must integrate all the activities of a technical crew member and not only flying. So, we construct a special flight service for each activity other than flying (weekly rest periods, training periods, annual leaves, union meetings, etc.). These flight services don't need rests nor deadheads. Their duration will be estimated by an equivalent number of hours of flight and not their real duration. Besides, there are no constraints on these special flight services (no briefing time, no connection, no maximal duration of a flight service, no maximal duration of a flying, etc.).

We note here that some of these activities other than flying are often preassigned to crew members.

In the following sections, we will concentrate only on flying activities and we will not take into account the other ones. The proposed formulation can integrate easily these activities using the special flight services described above.

2.2. Phase 2: Crew assignment to flight services

After generating all valid flight services covering at least once each flight segment, we must find an individual assignment of the required technical crew members to some of these flight services such that, for each planned flight segment, there is an appropriate technical crew to operate it.

The objective of this phase is therefore to find a feasible minimum cost assignment i.e. satisfying all the regulation rules (crew composition, crew member qualifications, minimum number of days off, annual vacations, training periods, maximum flying time per week and per month, etc.) and the collective agreements (incompatibilities between crew members, maximum number of consecutive working nights, etc.). The cost of an assignment is typically expressed as the total number of additional flight credits, where flight credits are hours of flight and/or the equivalent for activities such as training periods, union meetings, annual leaves, etc.

Technical crew members are divided into three groups called *Colleges: Pilots, Officers and Instructors*. An Instructor is a Pilot who can replace an Officer when needed. A crew is composed of a Pilot (or an Instructor) and an Officer (or an Instructor).

Moreover, each technical crew member is characterised by its fleet qualifications. In fact, a fleet family is a set of aircraft types that share the same crew requirements. Thus, a technical crew member qualified to fly one fleet in a fleet family is qualified to fly all fleets in the family, and for a same fleet family, all qualified technical crew members of a same college are polyvalent. Given that, for security reasons, each technical crew member is allowed to operate on only one fleet family, the Crew Assignment Problem can be treated for each fleet family separately without affecting the solution quality.

To solve the problem of crew assignment to flight services, for any given fleet family, we propose an approach based on formulations in terms of integer linear programs which will be described in the next section.

3. Proposed formulations

In this section, we present a large scale integer linear programming formulation for the Crew Assignment Problem.

3.1. Notation

Let F , indexed by f , represent the set of flight segments to be assigned to one and exactly one crew. Let S , indexed by s , be the set of valid flight services built previously from the given flight segments.

Let $S(f)$ be the set of flight services which include the flight segment f , and let $S(w)$ and $S(m)$ be the sets of flight services for week w , and month m respectively.

Let I , indexed by i , represent the set of instructors, P , indexed by p , represent the set of pilots and O , indexed by o , represent the set of officers.

Let L_S be the list of pairs of flight services which cannot be performed by a same crew. The elements of this list are pairs of flight services (1) having overlapping periods or (2) for which the ending station of the earlier flight service is different from the starting station of the other flight service and the time between the two flight services is too short for including a deadhead.

Let I_{ip} be the set of pairs (instructor, pilot) which are incompatible in the sense that they cannot be part of a same crew. Similarly, let $I_{ii'}$, I_{io} and I_{po} be the sets of incompatible pairs (instructor, instructor), (instructor, officer) and (pilot, officer) respectively. These incompatibilities derive from regulation which

imposes that in a technical crew, there is at most one crew member with a medical restriction, at most one training crew member, etc.

For a given flight service s , let BT_s be the number of hours of flight called *block* time and CT_s be the number of flight *credits*, which are hours of flight and/or the equivalent for activities other than flying (training periods, union meetings, etc.).

Let $T_{\max,w}$, $T_{\max,m}$ be the maximal allowed number of hours of flight per week and per month respectively, and $T_{\text{guarantee}}$ be the number of flight credits corresponding to the minimum duty guarantee cost. In fact, a crew is paid at least some amount, even for flight credits lower than $T_{\text{guarantee}}$. On the other hand, additional costs are paid when the flight credit exceeds $T_{\text{guarantee}}$.

Two types of decision variables are used: 0–1 variables and integer variables. For each flight segment f and each flight service s , three 0–1 variables are defined corresponding to the three technical crew colleges (pilots, officers and instructors):

- $x_{p,f} = 1$ if pilot p is assigned to the flight segment f ; and 0 otherwise,
- $y_{o,f} = 1$ if officer o is assigned to the flight segment f ; and 0 otherwise,
- $z_{i,f} = 1$ if instructor i is assigned to the flight segment f ; and 0 otherwise,
- $X_{p,s} = 1$ if pilot p is assigned to the flight service s ; and 0 otherwise,
- $Y_{o,s} = 1$ if officer o is assigned to the flight service s ; and 0 otherwise,
- $Z_{i,s} = 1$ if instructor i is assigned to the flight service s ; and 0 otherwise.

Let HS_k be the integer variables defined as the number of extra flight credits exceeding $T_{\text{guarantee}}$ worked by the crew member k , $k \in P \cup O \cup I$.

Using this notation, the Crew Assignment Problem may then be formulated as described in the next section.

3.2. A first mathematical model: ILPI

We propose to formulate the Crew Assignment Problem as a large scale integer linear program with a general structure and different types of constraints as follows:

$$(ILP1) \quad \text{Minimize} \quad \sum_{p \in P} HS_p + \sum_{o \in O} HS_o + \sum_{i \in I} HS_i \quad (3.1)$$

subject to:

$$\begin{aligned} x_{p,f} - \sum_{s/s \in S(f)} X_{p,s} &= 0 \quad \forall p \in P, \forall f \in F, \\ y_{o,f} - \sum_{s/s \in S(f)} Y_{o,s} &= 0 \quad \forall o \in O, \forall f \in F, \\ z_{i,f} - \sum_{s/s \in S(f)} Z_{i,s} &= 0 \quad \forall i \in I, \forall f \in F, \end{aligned} \quad (3.2)$$

$$\sum_p x_{p,f} + \sum_o y_{o,f} + \sum_i z_{i,f} = 2 \quad \forall f \in F, \quad (3.3)$$

$$\begin{aligned} \sum_p x_{p,f} &\leq 1 \quad \forall f \in F, \\ \sum_o y_{o,f} &\leq 1 \quad \forall f \in F, \\ \sum_i z_{i,f} &\leq 2 \quad \forall f \in F, \end{aligned} \quad (3.4)$$

$$\begin{aligned}
\sum_p X_{p,s} + \sum_o Y_{o,s} - \sum_i Z_{i,s} &\geq 0 \quad \forall s \in S, \\
\sum_p X_{p,s} - \sum_o Y_{o,s} + \sum_i Z_{i,s} &\geq 0 \quad \forall s \in S, \\
-\sum_p X_{p,s} + \sum_o Y_{o,s} + \sum_i Z_{i,s} &\geq 0 \quad \forall s \in S,
\end{aligned} \tag{3.5}$$

$$\begin{aligned}
\sum_{s/s \in S(w)} (BT_s \cdot X_{p,s}) &\leq T_{\max,w} \quad \forall p \in P, \forall w, \\
\sum_{s/s \in S(w)} (BT_s \cdot Y_{o,s}) &\leq T_{\max,w} \quad \forall o \in O, \forall w, \\
\sum_{s/s \in S(w)} (BT_s \cdot Z_{i,s}) &\leq T_{\max,w} \quad \forall i \in I, \forall w,
\end{aligned} \tag{3.6}$$

$$\begin{aligned}
\sum_{s/s \in S(m)} (BT_s \cdot X_{p,s}) &\leq T_{\max,m} \quad \forall p \in P, \forall m, \\
\sum_{s/s \in S(m)} (BT_s \cdot Y_{o,s}) &\leq T_{\max,m} \quad \forall o \in O, \forall m, \\
\sum_{s/s \in S(m)} (BT_s \cdot Z_{i,s}) &\leq T_{\max,m} \quad \forall i \in I, \forall m,
\end{aligned} \tag{3.7}$$

$$\begin{aligned}
X_{p,s} + X_{p,s'} &\leq 1 \quad \forall (s, s') \in L_S, \forall p \in P, \\
Y_{o,s} + Y_{o,s'} &\leq 1 \quad \forall (s, s') \in L_S, \forall o \in O, \\
Z_{i,s} + Z_{i,s'} &\leq 1 \quad \forall (s, s') \in L_S, \forall i \in I,
\end{aligned} \tag{3.8}$$

$$\begin{aligned}
X_{p,s} + Y_{o,s} &\leq 1 \quad \forall (p, o) \in I_{po}, \forall s \in S, \\
Z_{i,s} + X_{p,s} &\leq 1 \quad \forall (i, p) \in I_{ip}, \forall s \in S, \\
Z_{i,s} + Y_{o,s} &\leq 1 \quad \forall (i, o) \in I_{io}, \forall s \in S, \\
Z_{i,s} + Z_{i',s} &\leq 1 \quad \forall (i, i') \in I_{i'i}, \forall s \in S,
\end{aligned} \tag{3.9}$$

$$\begin{aligned}
\sum_{s/s \in S(m)} (CT_s \cdot X_{p,s}) &\leq T_{\text{guarantee}} + \text{HS}_p \quad \forall p \in P, \forall m, \\
\sum_{s/s \in S(m)} (CT_s \cdot Y_{o,s}) &\leq T_{\text{guarantee}} + \text{HS}_o \quad \forall o \in O, \forall m, \\
\sum_{s/s \in S(m)} (CT_s \cdot Z_{i,s}) &\leq T_{\text{guarantee}} + \text{HS}_i \quad \forall i \in I, \forall m,
\end{aligned} \tag{3.10}$$

$$\begin{aligned}
x_{p,f} &\text{ binary} \quad \forall p \in P, \forall f \in F, \\
y_{o,f} &\text{ binary} \quad \forall o \in O, \forall f \in F, \\
z_{i,f} &\text{ binary} \quad \forall i \in I, \forall f \in F,
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
X_{p,s} & \text{ binary } \quad \forall p \in P, \forall s \in S, \\
Y_{o,s} & \text{ binary } \quad \forall o \in O, \forall s \in S, \\
Z_{i,s} & \text{ binary } \quad \forall i \in I, \forall s \in S,
\end{aligned} \tag{3.12}$$

$$\begin{aligned}
HS_p & \geq 0, \text{ integer } \quad \forall p \in P, \\
HS_o & \geq 0, \text{ integer } \quad \forall o \in O, \\
HS_i & \geq 0, \text{ integer } \quad \forall i \in I.
\end{aligned} \tag{3.13}$$

The objective function (3.1) is to minimize the total number of extra flight credits worked by all the crew members in order to operate the planned flight segments of the considered time period.

Constraints (3.2) link flight segment variables and flight service variables. These constraints require that if a crew member is assigned to a flight segment, he/she must be assigned to one and only one of the flight services containing this flight segment.

For each flight segment, constraints (3.3) and (3.4) relate to the composition of the corresponding technical crew. Constraints (3.3) require that each flight segment be operated by exactly two technical crew members while constraints (3.4) impose the assignment of at most one pilot, one officer and two instructors. Thus, constraints (3.3) and (3.4) together lead to the assignment of two technical crew members to two different tasks in each flight segment: a pilot and an officer, an instructor and an officer, a pilot and an instructor or two instructors.

For each flight service, the assignment of the right technical crew is ensured by the constraints on the composition of the technical crew for flight segments, combined with constraints (3.2) and (3.5). Indeed, if a crew member is assigned to a flight segment f , constraints (3.2) require that this crew member be assigned to exactly one of the flight services containing it. Let $s1$ be this flight service. If a second crew member is assigned to the flight segment f under consideration, satisfying the crew composition constraints, relations (3.2) require that this crew member be assigned to exactly one of the flight services containing it. Let $s2$ be this flight service, which can be different from $s1$. To prevent this case, constraints (3.5) impose the choice of the same flight service for a same flight segment so that, crew composition is respected for each assigned flight service.

Regulatory restrictions on the number of hours of flight per week and per month are imposed by constraints (3.6) and (3.7) respectively.

Constraints (3.8) prohibit the assignment of a same crew member to incompatible flight services of the set L_S .

The possible incompatibilities between crew members are taken into account thanks to constraints (3.9) that prevent the assignment of two incompatible crew members to a same flight service.

The number of extra flight credits is determined by constraints (3.10).

Constraints (3.11)–(3.13) are the integrality constraints to be imposed on the variables.

The formulation ILP1 contains a limited number of variables but usually a huge number of constraints. Hence, it would be difficult to solve by standard integer linear programming techniques. However, we show in the following sections how the special structure of the constraint matrix can be exploited to enhance the proposed formulation and then to improve the efficiency of standard integer linear programming techniques.

3.3. An enhanced formulation: ILP2

Before solving the large scale integer linear program ILP1, we introduce a reformulation phase exploiting the particular structure of the constraint matrix. In fact, a large proportion of the constraints in the model is composed of binary exclusion constraints (3.8). A close examination of the structure of the cor-

responding constraint graph shows that it is indeed quite close to an interval graph. A graph $G = (V, E)$ with node set V and edge set E is called an *interval graph* if there is an assignment of an interval of the real line to each $v \in V$ such that $(u, v) \in E$ if and only if the intervals corresponding to u and v intersect. (Golumbic, 1980).

If we consider the graph $G = (V, E)$ where the nodes in V correspond to the valid flight services, and the edges in E represent pairs of incompatible flight services, G is an interval graph if and only if each edge of E links two flight services which must be performed simultaneously during some period of time. In our case, this condition is not exactly satisfied because incompatibility between flight services is not only due to overlap of some period of time but also a few other constraints deriving from the collective agreements. So the exclusion constraint graph is not exactly an interval graph. However it is realized that it is close to the structure of an interval graph in the sense that only a few edges need to be added/deleted to recover the structure of an interval graph.

In view of this, we propose to replace the exclusion constraints (3.8) in the formulation ILP1 by stronger and less numerous ones: the *clique constraints*.

A set $C \subseteq V$ is called a *clique* if each pair of nodes in C is joined by an edge. A *maximal clique* is a clique which is not contained in another clique (Berge, 1970). Thus, the exclusion constraints are satisfied if and only if no more than one node (i.e. flight service) is selected from each maximal clique. This condition is expressed by clique constraints of the form $\sum_{i \in C} x_i \leq 1$ where C is the set of nodes of a maximal clique (Nemhauser and Wolsey, 1988).

In order to be more general, and knowing that an interval graph is chordal (Nemhauser and Wolsey, 1988), we used the efficient algorithms available for enumerating all maximal cliques for chordal graphs as a basis for enumerating maximal cliques in such ‘near-interval graphs’ which are ‘near-chordal graphs’ too. A *chordal* graph is a graph where each cycle with more than 3 edges has a chord.

The idea of replacing the exclusion constraints by the clique constraints is motivated by the well-known fact that the number of maximal cliques in an interval graph is bounded by the number of nodes in the considered graph. Thus, the number of maximal cliques of the exclusion constraint graph of the formulation ILP1 can be expected to be sufficiently small to make it possible to generate them all, and to achieve a substantial reduction on the total number of constraints in the model. This is confirmed by the results shown in Table 3.

To generate the clique constraints, the following three steps are proposed.

Step 1. Build the associated constraint graph, i.e. the graph in which:

- the nodes correspond to all flight services,
- the edges set is defined by linking flight service s_i to flight service s_j if they are incompatible. So each edge corresponds to an exclusion constraint expressed between two incompatible flight services of the set L_S .

Step 2. Determine all the maximal cliques of the exclusion constraint graph built in Step 1. To achieve this, the particular structure of chordal graphs (Berge, 1970; Golumbic, 1980; Gondran and Minoux, 1993; Nemhauser and Wolsey, 1988) is used as described in Zeghal (2002). Let C , indexed by c , be the set of the corresponding maximal cliques.

Step 3. Generate the clique constraints. Each maximal clique leads to a clique constraint as follows:

$$\begin{aligned}
 \sum_{s \in c} X_{p,s} &\leq 1 \quad \forall c \in C, \forall p \in P, \\
 \sum_{s \in c} Y_{o,s} &\leq 1 \quad \forall c \in C, \forall o \in O, \\
 \sum_{s \in c} Z_{i,s} &\leq 1 \quad \forall c \in C, \forall i \in I.
 \end{aligned} \tag{3.14}$$

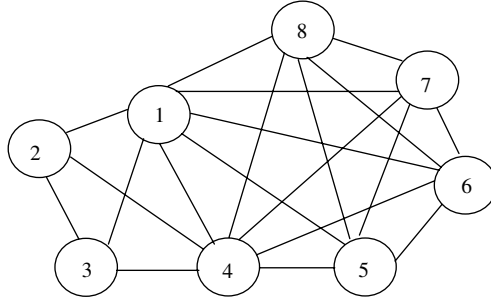


Fig. 1. Constraint graph associated to the example.

The resulting enhanced formulation for the Crew Assignment Problem, denoted ILP2, is obtained by replacing, in the initial formulation ILP1, the exclusion constraints (3.8) by the clique constraints (3.14).

Computational results obtained on a series of real test problems, and reported in Section 4, confirm that formulation ILP2 actually makes the solution process much more efficient.

To illustrate the impact of such an enhancement phase, we consider an example with 8 nodes and 20 edges corresponding to the set of the exclusion constraints. The objective is to find a maximum cardinality stable set i.e. to maximize the number of nodes selected without violating the incompatibility constraints. Fig. 1 represents the associated constraint graph.

The constraint graph of Fig. 1 has 2 maximal cliques: $(1, 2, 3, 4)$ and $(1, 4, 5, 6, 7, 8)$. So, if we apply the enhancement phase, we will generate 2 new clique constraints, which will replace all the original binary exclusion constraints. These clique constraints can be written as follows:

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 &\leq 1, \\ x_1 + x_4 + x_5 + x_6 + x_7 + x_8 &\leq 1, \end{aligned} \tag{3.15}$$

where $\forall i, i = 1, \dots, 8$:

$$x_i = \begin{cases} 1 & \text{if the node } i \text{ is selected,} \\ 0 & \text{otherwise.} \end{cases}$$

We note that the clique constraints are much less numerous than the exclusion constraints.

Besides this, if we relax the integrality constraints, and solve the relaxed problem, we obtain the following optimal objective functions:

- using the binary exclusion constraints: $Z_{\text{relaxed}} = 4$,
- using the clique constraints: $Z_{\text{relaxed}} = 2$.

The comparison of the optimal relaxed solution values to the optimal binary objective function $Z_{\text{binary}} = 2$ shows that the clique constraints are stronger than the binary exclusion constraints.

This example is a typical situation where the clique constraints enhance the formulation by tightening the linear relaxation while reducing the number of constraints.

4. Solving the Crew Assignment Problem by standard integer linear programming techniques

The first step in solving the Crew Assignment Problem is the construction of all valid flight services from the flight segments of the considered time period.

For the considered airline company TunisAir, there is only one base for fleet but three different near bases for crew. So, we suppose that it is always possible to reposition the crew members by using deadheads between different crew bases such that crew members are transported from a crew base corresponding to the end station of a flight service to their crew base or from their crew base to another crew base corresponding to the starting station of a flight service. Consequently, we do not have to introduce more constraints to take into account the crew base of each crew member.

Besides this, special constraints on flight service construction, specific to TunisAir, have been added. These constraints impose to give rests in the crew base, when it is possible. So, for each flight service, if its ending station is not a crew base, required deadheads are added and some constraints (maximal duration of a flight service) are imposed to keep the flight service under consideration valid. The same procedure is applied to flight services needing rest before departure, and leaving from a station which is not a crew base.

If the required deadheads violate the maximal duration constraints, and the various flight segments composing the flight service are covered by at least another flight service, we eliminate the flight service under consideration. Otherwise, we do not add the deadheads and we give rest to the crew members in the ending station of the flight service.

Since a valid flight service is composed of a limited number of flight segments satisfying the set of constraints deriving from regulation and collective agreements, we could easily obtain the set of all valid flight services by developing an enumerative procedure.

The assignment of the required technical crew to the flight services is then solved by standard integer linear programming techniques (CPLEX 6.0.2), using the formulations ILP1 and ILP2 proposed in Section 3.

The computational results obtained on 20 real test problems provided by the Tunisian airline company TunisAir, are displayed in Tables 1–3. No initial solutions were used for any of the test problems.

Table 1
Test problem characteristics

Problems	Fleet family	Time period	Flight segments	Crew members
P.01	A 300	1 month	92	10
P.02	A 300	1 month	182	16
P.03	A 300	1 month	195	18
P.04	A 300	3 weeks	166	18
P.05	A 300	2 weeks	88	16
P.06	B 727	2 weeks	210	34
P.07	A 300	10 days	94	20
P.08	A 300	1 week	59	18
P.09	A 300	1 week	68	20
P.10	A 300	1 week	86	16
P.11	A 320	1 week	155	38
P.12	B 727	1 week	99	30
P.13	B 727	1 week	104	32
P.14	B 727	1 week	121	34
P.15	B 737	1 week	125	32
P.16	B 737	1 week	170	36
P.17	B 737	1 week	104	32
P.18	A 320	5 days	100	38
P.19	B 737	5 days	101	40
P.20	B 737	5 days	83	39

Table 1 displays the following attributes of each problem tested.

- problem name,
- fleet family: we consider 4 fleet families: Airbus 300, Airbus 320, Boeing 727 and Boeing 737,
- time period: it ranges from 5 days to 1 month,
- flight segments: the number of flight segments to be operated within the considered time period,
- crew members: the total number of technical crew members (Pilots, Officers and Instructors) available during the considered time period.

As can be seen from Table 1, the problems involve between 59 and 210 flight segments which have to be operated within different time periods ranging from 5 days to 1 month. The total number of available crew members to be assigned to the flight segments depends on the fleet family and varies between 10 and 40.

Table 2 describes the computational results obtained on the formulation ILP1. The considered real test problems have between 2.270 and 17.000 variables and between 9.808 and 268.316 constraints, with this formulation.

The first part of Table 2 concerns the construction of valid flight services and shows the size of the resulting formulation ILP1. For each problem, we report the number of the valid flight services (Flight services) built from the set of the considered flight segments (Flight segments) and the number of variables (Variables) and constraints (Constraints) of formulation ILP1. It is observed that the number of flight services is of the same order of magnitude as the number of flight segments. This justifies the application of an enumerative procedure to build all the valid flight services.

The middle part of the table focuses on the solution of the linear relaxation (3.1)–(3.10) of the formulation ILP1 and displays the following information:

Table 2
Computational results obtained on formulation ILP1

Problems	Flight segments	Flight services	Variables	Constraints	Z _{LP}	CPU _{LP} (seconds)	Z _{IP}	CPU _{IP} (seconds)	% Gap
P.01	92	134	2270	9808	608	11	968 ^a	83	59.2
P.02	182	238	6736	42924	2904	52	2904 ^a	1531	0
P.03	195	259	8190	53040	2754	80	2754 ^a	2420	0
P.04	166	214	6858	47902	4282	65	4282 ^a	1510	0
P.05	88	111	3200	20698	2582	20	2582 ^a	1009	0
P.06	210	289	17000	191223	5830	1792	–	–	–
P.07	94	123	4360	43114	8724	36	8964 ^a	6852	2.8
P.08	59	80	2520	26546	1844	26	1964	19567	6.5
P.09	68	84	3060	23436	3808	20	3808 ^a	4880	0
P.10	86	116	3248	31700	11956	31	12012 ^a	749	0.5
P.11	155	203	13642	235054	8598	3987	–	–	–
P.12	99	135	7050	76161	2574	75	2949	14452	14.6
P.13	104	143	7936	87401	4120	69	4120 ^a	6153	0
P.14	121	167	9826	99934	3480	104	3520	15142	1.1
P.15	125	174	9600	150461	8578	678	–	–	–
P.16	170	228	14354	268316	9400	4182	–	–	–
P.17	104	142	7968	105290	4650	114	4930	28609	6.0
P.18	100	135	8968	140895	7920	319	–	–	–
P.19	101	135	9480	151498	6782	323	–	–	–
P.20	83	109	7527	105147	3774	91	–	–	–

The symbol “–” indicates that no integer solution could be found after 8 hours of computation.

^a When exact optimality has been obtained.

- Z_{LP} : the value of the optimal objective function corresponding to the relaxed problem. It is expressed in terms of additional flight credits (in minutes),
- CPU_{LP} (seconds): the computation time, in seconds, needed to obtain this solution (Z_{LP}).

The last part of the table concerns the best integer solutions obtained. It reports:

- Z_{IP} : the value of the objective function corresponding to the best integer solution found. It is expressed in terms of additional flight credits (in minutes). Exact optimal solutions are pointed out by a star “*” while the symbol “-” indicates that no integer solution could be found within 8 hours of computation.
- CPU_{IP} (seconds): the computation time, in seconds, required to find this best integer solution (Z_{IP}).

Finally, we provide the gap (% Gap) between the best integer solution (Z_{IP}) and the optimal linear relaxation solution (Z_{LP}) in percents.

Computational results shown in Table 2 confirm the practical applicability of the ILP1 formulation.

As can be seen in Table 2, the application of the CPLEX solver in MIP mode provides optimal integer solutions to 45% of the problems tested (those having up to 7.936 variables and up to 87.401 constraints are solved exactly in less than 2 hours).

However, in spite of relatively long computation times (up to 8 hours), non-optimal integer solutions are found for only 20% of the problems tested while no integer solution is found for 35% of the instances.

Table 3 displays the computational results obtained on the enhanced formulation ILP2. No initial solutions were used for any of the test problems.

The first part of this table concerns the size of the resulting formulation ILP2. For each problem, we indicate the number of maximal cliques generated for the exclusion constraint graph (Maximal cliques).

Table 3
Computational results obtained on formulation ILP2

Problems	Flight services	Maximal cliques	Variables	Constraints	Z_{LP}	CPU_{LP} (seconds)	Z_{IP}	CPU_{IP} (seconds)	% Gap
P.01	134	43	2270	2098	968	7	968 ^a	49	0
P.02	238	81	6736	6492	2904	29	2904 ^a	573	0
P.03	259	89	8190	7590	2754	33	2754 ^a	2082	0
P.04	214	81	6858	6520	4282	49	4282 ^a	541	0
P.05	111	42	3200	3178	2582	10	2582 ^a	134	0
P.06	289	77	17000	12791	5830	48	-	-	-
P.07	123	41	4360	3914	8964	55	8964 ^a	223	0
P.08	80	27	2520	2336	1844	14	1844 ^a	1365	0
P.09	84	21	3060	2636	3808	9	3808 ^a	3676	0
P.10	116	22	3248	2948	12012	11	12012 ^a	32	0
P.11	203	47	13642	9714	8598	801	8608	14828	0.1
P.12	135	33	7050	5421	2574	27	2574 ^a	23195	0
P.13	143	44	7936	6281	4120	23	4120 ^a	3285	0
P.14	167	35	9826	6672	3480	59	3685	3601	5.9
P.15	174	47	9600	6877	8578	295	9714	8064	13.2
P.16	228	64	14354	10844	9400	657	-	-	-
P.17	142	28	7968	5834	4650	55	4650 ^a	20355	0
P.18	135	26	8968	6185	7920	110	9038	4937	14.1
P.19	135	30	9480	6378	6782	151	7077	24807	4.3
P.20	109	20	7527	5229	3774	24	3869	16888	2.5

The symbol “-” indicates that no integer solution could be found after 8 hours of computation.

^a When exact optimality has been obtained.

We also report the number of variables (Variables) and constraints (Constraints) of the enhanced formulation ILP2.

The middle part of the table is related to the solution of the linear relaxation of the formulation ILP2. It shows:

- Z_{LP} : the value of the optimal objective function corresponding to the relaxed problem. It is expressed in terms of additional flight credits (in minutes),
- CPU_{LP} (seconds): the computation time, in seconds, needed to obtain this solution (Z_{LP}).

The last part of the table describes the best integer solutions obtained. It indicates:

- Z_{IP} : the value of the objective function corresponding to the best integer solution found. It is expressed in terms of additional flight credits (in minutes). Exact optimal solutions are pointed out by a star “*” while the symbol “-” indicates that no integer solution could be found within 8 hours of computation.
- CPU_{IP} (seconds): the computation time, in seconds, required to find this best integer solution (Z_{IP}).

Finally, we provide the gap (% Gap) between the best integer solution (Z_{IP}) and the optimal linear relaxation solution (Z_{LP}) in percents.

Table 3 shows that the exclusion constraint graphs have a very reduced number of maximal cliques, which confirms the validity of our assumption and, to a large extent, explains the improvements in efficiency obtained with ILP2.

As can be seen in Table 3, the enhanced formulation ILP2 has much fewer constraints than the formulation ILP1. Indeed, for the considered real test problems which have between 9.808 and 268.316 constraints with formulation ILP1, there are only between 2.098 and 12.791 constraints with formulation ILP2. This dramatic reduction in the total number of constraints (ranging from 79% to 96%) is due to the replacement of the binary exclusion constraints by the clique constraints: the number of pairs of incompatible flight services is enormous as compared with the number of maximal cliques in the associated constraint graph.

Comparisons between the results obtained with the two formulations ILP1 (see Table 2) and ILP2 (see Table 3) show that formulation ILP2 improves significantly the efficiency of the solution process: as can be seen in Table 3, the application of the CPLEX solver in MIP mode to the enhanced formulation ILP2 gives optimal integer solutions to more problems (60% of the tested problems against 45% with the formulation ILP1) typically in less than 1 h. For the tested problems solved optimally with formulation ILP1, the computation time is much reduced while using formulation ILP2, this reduction can reach 97% in some cases. Furthermore, for 30% of the considered problems, integer but non-optimal solutions are found and thanks to this enhanced formulation, after 8 computation hours, no feasible integer solution could be found for only 10% of the tested problems (against 35% with formulation ILP1).

The main explanatory factors of the improved results using ILP2 as compared with ILP1 are the following:

- the computation times needed to solve the linear relaxations are significantly less with ILP2 as compared with ILP1;
- the lower bounds provided by the linear relaxation of ILP2 appear to be stronger for a number of instances (namely P.01, P.07 and P.10);
- the average number of fractional variables in the optimal solutions to the linear relaxation appear to be significantly reduced with ILP2.

5. Heuristic method

Since finding feasible integer solutions turned out to be difficult for some instances, even with formulation ILP2, when using the standard heuristics imbedded into the MIP solver CPLEX 6.0.2, investigating the possibility of designing improved heuristics was soon recognized as an important issue. We describe in this section a heuristic approach based on a rounding strategy embedded in a partial tree-search procedure.

From the computational results presented in Section 5.2, it will be seen that this heuristic provides feasible integer solutions of better quality in a higher proportion of cases as compared with CPLEX 6.0.2.

5.1. Description of the method

The proposed heuristic method consists in applying the following steps in an iterative way:

- Step 1.* Choose a branching node among the set of *active nodes*.
- Step 2.* For the selected node, choose a separation variable.
- Step 3.* Generate 2 new nodes corresponding to the 2 new problems obtained by fixing the separation variable to the values 0 and 1 respectively.
- Step 4.* Solve the linear relaxation of these 2 new problems, using the primal Simplex method.
- Step 5.* Given the new values of the linear relaxation solutions obtained in Step 4, update the state of the unexplored nodes (active or inactive).
- Step 6.* Round, if some requirements are met, the fractional variables of the linear relaxation solution obtained in Step 4 to the closer integer values 0 or 1.
- Step 7.* Test the feasibility of the integer solution constructed in Step 6.
- Step 8.* If the integer solution constructed in Step 6 is feasible and has a value lower than that of the lowest integer solution encountered, save it as the best integer solution found.

This iterative process continues until:

- an integer solution with a value equal to that of the best linear relaxation solution encountered is found or
- a maximal computation time is reached or
- a maximal number of explored nodes is exceeded.

5.1.1. Node selection

There are essentially two different rules for the node selection: “Breadth First” and “Depth First”. We propose here to use a mixed tree exploration strategy combining these two rules by applying a “Breadth First” rule and an active nodes’ limitation strategy.

The selected “Breadth First” rule consists in choosing, among the set of active nodes, a node which has the best linear relaxation solution’s value. And if many nodes are eligible, the rule consists in selecting the node which is the closest to the root node.

The active nodes’ limitation strategy consists in selecting among the set of the generated and unexplored nodes, a limited number of nodes which will be *active* and thus can be considered for branching: it is the set of active nodes which is updated after each node exploration (Step 5).

For the selection of the active nodes, we choose those having the best evaluations (i.e. values of the linear relaxation solutions) while being the deepest in the tree.

The number of active nodes, denoted by k , is a parameter of the proposed method. It has a great influence on the proposed tree exploration strategy: for small values of k , the tree exploration strategy

converges to a “Depth First” search and for large values of k , the tree exploration strategy converges to a “Breadth First” search. For intermediate values, we have a mixed tree exploration strategy.

5.1.2. Variable selection

For a selected node, the separation variable is chosen among the fractional variables of the corresponding linear relaxation solution. We suggest to select the variable which is the closest to 0.5 in order to maximize the disparity between the two problems that will be generated.

5.1.3. Rounding strategy

For each selected node, if the corresponding linear relaxation solution is not integer, we evaluate the number of its fractional variables contained in the interval $[\alpha; 1 - \alpha]$ where α is a real parameter bounded by 0 and 0.5 strictly. If this number is lower than a maximal value n_{\max} (e.g. $n_{\max} = 10\%$ of the total number of basic variables), all the fractional variables of this solution are rounded to the nearer integer value 0 or 1.

5.2. Computational results

The heuristic method has been applied to the 20 real test problems provided by TunisAir and presented in Table 1 of Section 4. The used formulation is ILP2 for which better solutions are expected.

The parameters of the method are fixed as follows:

- $\alpha = 0.1$.
- $n_{\max} = 10\%$ of the total number of basic variables of the corresponding linear relaxation solution.

For each problem, we have tested different values of the parameter k (2, 5, 10 and 100) and reported only the best results obtained. For more details, see Zeghal (2002).

Table 4 displays the computational results obtained by the heuristic method. No initial solutions were used for any of the test problems.

For each problem, we indicate the number of variables (Variables) and constraints of the two formulations ILP1 and ILP2 (respectively Constraints ILP1 and Constraints ILP2).

Then, we compare the efficiency of the exact and approximate methods by displaying:

1. The following gaps, in percents:
 - $Z1$ (%) exact and $Z2$ (%) exact: the gaps in percents between the best integer solution obtained by solving exactly formulations ILP1 and ILP2 respectively and the best lower bound (i.e. the optimal linear relaxation solution of the formulation ILP2),
 - Z (%) approx: the gap in percents between the best integer solution obtained by the heuristic method applied to the formulation ILP2, and best lower bound (i.e. the optimal linear relaxation solution of the formulation ILP2).
2. The computation time in seconds, needed to find the best integer solution mentioned. So, CPU_1 (seconds) and CPU_2 (seconds) correspond to the computation time required by the exact methods of CPLEX 6.0.2. when applied to the formulations ILP1 and ILP2 respectively. CPU_{app} (seconds) presents the computation time required by the heuristic method applied to the formulation ILP2.

As can be seen in Table 4, the heuristic method achieves a better trade off between solution quality and computational effort: it gives optimal integer solutions in less than 40 minutes to half of the test problems, and approximate integer solutions with good quality (within 5% off the lower bound) in less than 47 minutes for the other problems.

Table 4

Computational results obtained by the heuristic method applied on formulation ILP2

Pbs	Variables	Constraints (ILP1)	Constraints (ILP2)	Z1 (%) exact	Z2 (%) exact	Z (%) approx.	CPU ₁ (seconds)	CPU ₂ (seconds)	CPU _{app} (seconds)
P.01	2270	9808	2098	0	0	0	83	49	53
P.02	6736	42924	6492	0	0	0	1531	573	392
P.03	8190	53040	7590	0	0	0	2420	2082	1537
P.04	6858	47902	6520	0	0	0	1510	541	469
P.05	3200	20698	3178	0	0	0	1009	134	152
P.06	17000	191223	12791	–	–	3.5	–	–	1991
P.07	4360	43114	3914	0	0	0	6852	223	113
P.08	2520	26546	2336	6.5	0	0.3	19567	1365	987
P.09	3060	23436	2636	0	0	0	4880	3676	38
P.10	3248	31700	2948	0	0	0	749	32	15
P.11	13642	235054	9714	–	0.1	3.4	–	14828	2195
P.12	7050	76161	5421	14.6	0	4.9	14452	23195	2799
P.13	7936	87401	6281	0	0	0	6153	3285	2382
P.14	9826	99934	6672	1.1	5.9	3.4	15142	3601	1749
P.15	9600	150461	6877	–	13.2	1.7	–	8064	614
P.16	14354	268316	10844	–	–	2.8	–	–	1375
P.17	7968	105290	5834	6.0	0	0	28609	20355	688
P.18	8968	140895	6185	–	14.1	1.9	–	4937	596
P.19	9480	151498	6378	–	4.3	4.8	–	24807	805
P.20	7527	105147	5229	–	2.5	4.1	–	16888	260

The symbol “–” indicates that no integer solution could be found after 8 hours of computation.

Table 4 shows that the heuristic method solves real problems for which CPLEX 6.0.2. could not find any integer solution within 8 computing hours.

For the instances that exact methods can not solve to optimality, the approximate integer solutions given by the heuristic method are generally closer to the lower bound than those given by the exact methods. However, for some cases (e.g. instance P11), the slight degradation in solution quality is compensated for by a substantial reduction in computation time.

6. Possible extensions

The new approach developed in this paper was tested for the Crew Assignment Problem arising from our study case TunisAir. Thus, the proposed formulations consider only the constraints imposed on this airline company such as the maximal duration of a flight service, the maximal flying time in a flight service, the minimum number of hours of rest between flight services, the incompatibilities between technical crew members, etc. While these constraints involve only flight services and the incompatibilities between pairs of flight services, our formulations can be extended to more common constraints imposed on a sequence of several consecutive flight services such as: the maximum allowed time away from base; and for any specified number of consecutive working days, the maximal allowed flight time, the days off requirements and the maximum number of overnights away from base. We are going to show that such constraints can be incorporated in our formulation by adding a (limited) number of variables and constraints. Clearly the resulting extended formulations may possibly be more difficult to solve exactly, further computational experiments will therefore be needed. This is left for future work.

In the following subsections we get into some detail on how our basic model could be accommodated to handle several types of additional constraints frequently encountered in Crew Assignment Problems.

6.1. The maximal allowed flight time for any set of consecutive working days

A maximal allowed flight time can be imposed on any set of N consecutive working days (not only calendar weeks or months).

To integrate this type of constraints in our model, let T_{\max} be the maximal allowed flight time for this period. For each day $t \in [1, |\Gamma| - N + 1]$ where $|\Gamma|$ is the number of days in the planning period Γ , define:

- $FS(t)$ as the set of flight services included totally or partially in the period $[t, t + N - 1]$,
- $HF_s(t)$ as the number of hours of flight included in this period, for flight service s .

Using the above notation, we propose to replace constraints (3.6) and (3.7) by the following ones:

$$\begin{aligned}
 \sum_{s \in FS(t)} (HF_s(t) \cdot X_{p,s}) &\leq T_{\max} \quad \forall p \in P, \forall t \in [1, |\Gamma| - N + 1], \\
 \sum_{s \in FS(t)} (HF_s(t) \cdot Y_{o,s}) &\leq T_{\max} \quad \forall o \in O, \forall t \in [1, |\Gamma| - N + 1], \\
 \sum_{s \in FS(t)} (HF_s(t) \cdot Z_{i,s}) &\leq T_{\max} \quad \forall i \in I, \forall t \in [1, |\Gamma| - N + 1].
 \end{aligned} \tag{3.16}$$

We notice that these new constraints (3.16) are only slightly more numerous than the constraints (3.6) and (3.7).

6.2. The days off requirements for consecutive working days

The work convention often imposes that each crew member must have a day off, in his/her crew base, after a given number of consecutive working days, say Q .

To integrate this type of constraints in our model, define, for each day t of the planning period Γ , $S(t)$ as the set of flight services included totally or partially in the day t . Let $D(t)$ be the set of flight services starting from stations other than crew base, and included partially or totally in the day t , and similarly, let $A(t)$ be the set of flight services ending at stations other than crew base, and included partially or totally in the day t .

For each day t of the planning period Γ and each crew member k ($k \in P \cup O \cup I$), define the binary variable w_t^k which is equal to 1 if the crew member k is working during day t , and zero otherwise.

We notice that a crew member k is working on day t :

- if he/she is assigned to a flight service $s \in S(t)$, or
- if he/she is not assigned to a flight service $s \in S(t)$, but he/she is out of his/her base.

So, to express these two possibilities (a) and (b), and to take into account the constraints on the day off, we propose to add the following inequalities:

$$\begin{aligned}
 \text{(a)} \quad \sum_{s \in S(t)} X_{p,s} &\leq 2w_t^p \quad \forall p \in P, \forall t \in \Gamma, \\
 \sum_{s \in S(t)} Y_{o,s} &\leq 2w_t^o \quad \forall o \in O, \forall t \in \Gamma, \\
 \sum_{s \in S(t)} Z_{i,s} &\leq 2w_t^i \quad \forall i \in I, \forall t \in \Gamma,
 \end{aligned}$$

$$\begin{aligned}
\text{(b)} \quad & \sum_{s \in A(t-1)} X_{p,s} + \sum_{s \in D(t+1)} X_{p,s} - 3 \sum_{s \in S(t)} X_{p,s} \leq 3d_t^p + 1 \quad \forall p \in P, \forall t \in [2, |\Gamma| - 1], \\
& \sum_{s \in A(t-1)} Y_{o,s} + \sum_{s \in D(t+1)} Y_{o,s} - 3 \sum_{s \in S(t)} Y_{o,s} \leq 3d_t^o + 1 \quad \forall o \in O, \forall t \in [2, |\Gamma| - 1], \\
& \sum_{s \in A(t-1)} Z_{i,s} + \sum_{s \in D(t+1)} Z_{i,s} - 3 \sum_{s \in S(t)} Z_{i,s} \leq 3d_t^i + 1 \quad \forall i \in I, \forall t \in [2, |\Gamma| - 1],
\end{aligned} \tag{3.17}$$

$$\begin{aligned}
& \sum_{j \in [t, t+Q]} w_j^p \leq Q \quad \forall p \in P, \forall t \in [1, |\Gamma| - Q], \\
& \sum_{j \in [t, t+Q]} w_j^o \leq Q \quad \forall o \in O, \forall t \in [1, |\Gamma| - Q], \\
& \sum_{j \in [t, t+Q]} w_j^i \leq Q \quad \forall i \in I, \forall t \in [1, |\Gamma| - Q].
\end{aligned} \tag{3.18}$$

Constraints (3.17) ensure that variables w_t^k takes the value 1 if the crew member k is working on day t . Assuming that a crew member can be assigned to at most 2 consecutive flight services included partially or totally in a given day t (this is typical), in constraints (3.17), the quantities $\sum_{s \in S(t)} X_{p,s}$, $\sum_{s \in S(t)} Y_{o,s}$ and $\sum_{s \in S(t)} Z_{i,s}$ are less than or equal to 2; this explains the use of the constants appearing on the right-hand side of constraints (a) and in constraints (b).

In constraints (3.17), the inequalities (b) force variables w_t^k to take the value 1 if on day t , crew member k :

- had a flight service ending at a place different from his/her crew base on the previous day (first term),
- has a flight service starting from a place different from his/her crew base on the following day (second term),
- does not work on this day t (third term).

Inequalities (3.18) ensure that a crew member can not work more than Q consecutive days.

The constraints on assigning days off can thus be integrated in our model by adding some variables and constraints: their number is proportional to (number of crew members) * (number of days in the planning period).

6.3. The maximum allowed time away from base

The work convention often stipulates that the time away from base must be less than or equal to a given number of consecutive days, say TA_{\max} .

Let $D(t)$ (respectively $A(t)$) be the set of flight services starting from (ending at) places other than crew base, and included partially or totally in the day t . For each day t of the planning period Γ and each crew member k ($k \in P \cup O \cup I$), define the binary variable q_t^k which is equal to 1 if the crew member k is away from his/her base in day t , and zero otherwise.

We notice that a crew member k is away from his/her base in day t :

- if he/she is assigned to a flight service $s \in D(t) \cup A(t)$, or
- if he/she is in a place other than his/her crew base, without being assigned to any flight service (i.e. he/she had a flight service ending at this place on the previous day, he/she does not work on day t and he/she has a flight service starting from a place other than his/her crew base on the following day).

So, to express these two possibilities (a) and (b), and to take into account the constraints on the time away from base, we propose to add the following inequalities:

$$\begin{aligned}
\text{(a)} \quad & \sum_{s \in D(t) \cup A(t)} X_{p,s} \leq 2q_t^p \quad \forall p \in P, \forall t \in \Gamma, \\
& \sum_{s \in D(t) \cup A(t)} Y_{o,s} \leq 2q_t^o \quad \forall o \in O, \forall t \in \Gamma, \\
& \sum_{s \in D(t) \cup A(t)} Z_{i,s} \leq 2q_t^i, \quad \forall i \in I, \forall t \in \Gamma, \\
\text{(b)} \quad & \sum_{s \in A(t-1)} X_{p,s} + \sum_{s \in D(t+1)} X_{p,s} - 3 \sum_{s \in D(t) \cup A(t)} X_{p,s} \leq 3q_t^p + 1 \quad \forall p \in P, \forall t \in [2, |\Gamma| - 1], \\
& \sum_{s \in A(t-1)} Y_{o,s} + \sum_{s \in D(t+1)} Y_{o,s} - 3 \sum_{s \in D(t) \cup A(t)} Y_{o,s} \leq 3q_t^o + 1 \quad \forall o \in O, \forall t \in [2, |\Gamma| - 1], \\
& \sum_{s \in A(t-1)} Z_{i,s} + \sum_{s \in D(t+1)} Z_{i,s} - 3 \sum_{s \in D(t) \cup A(t)} Z_{i,s} \leq 3q_t^i + 1 \quad \forall i \in I, \forall t \in [2, |\Gamma| - 1],
\end{aligned} \tag{3.19}$$

$$\begin{aligned}
& \sum_{j \in [t, t + \text{TA}_{\max}]} q_j^p \leq \text{TA}_{\max} \quad \forall p \in P, \forall t \in [1, |\Gamma| - \text{TA}_{\max}], \\
& \sum_{j \in [t, t + \text{TA}_{\max}]} q_j^o \leq \text{TA}_{\max} \quad \forall o \in O, \forall t \in [1, |\Gamma| - \text{TA}_{\max}], \\
& \sum_{j \in [t, t + \text{TA}_{\max}]} q_j^i \leq \text{TA}_{\max} \quad \forall i \in I, \forall t \in [1, |\Gamma| - \text{TA}_{\max}].
\end{aligned} \tag{3.20}$$

Constraints (3.19) ensure that variables q_t^k are taking the right values to express whether the crew member k is away from base or not. Again, assuming that a crew member can be assigned to at most two consecutive flight services included partially or totally in a given day t , in constraints (3.19), the quantities $\sum_{s \in D(t) \cup A(t)} X_{p,s}$, $\sum_{s \in D(t) \cup A(t)} Y_{o,s}$ and $\sum_{s \in D(t) \cup A(t)} Z_{i,s}$ can take only values in $\{0, 1, 2\}$; this explains the use of the constants appearing on the right-hand side of constraints (a) and in constraints (b).

In constraints (3.19), the inequalities (b) force variables q_t^k to take the value 1 if on day t , crew member k :

- had a flight service ending at a place different from his/her crew base on the previous day (first term),
- has a flight service starting from a place different from his/her crew base on the following day (second term),
- is not working on this day t (third term).

Inequalities (3.20) express the constraints on maximum allowed time away from base.

For some airline companies, this type of constraints is expressed differently: exceeding the maximum allowed time away from base (TA_{\max}) is not prohibited but only penalized. To formulate that, let Φ^k be the penalty associated with each extra day that a crew member k spends away from base. Then, for each day t of the planning period Γ and each crew member k , define binary variable δ_t^k which is equal to 1 if the crew member k has to spend an extra day away from his/her base in the period $[t, t + \text{TA}_{\max}]$ and zero otherwise.

The objective function (3.1) and the constraints (3.20) become respectively:

$$\begin{aligned} \text{Minimize } & \sum_{p \in P} \text{HS}_p + \sum_{o \in O} \text{HS}_o + \sum_{i \in I} \text{HS}_i + \sum_{t \in [0, |\Gamma| - \text{TA}_{\max}]} \sum_{p \in P} \Phi^p \delta_t^p + \sum_{t \in [0, |\Gamma| - \text{TA}_{\max}]} \sum_{o \in O} \Phi^o \delta_t^o \\ & + \sum_{t \in [0, |\Gamma| - \text{TA}_{\max}]} \sum_{i \in I} \Phi^i \delta_t^i \end{aligned} \quad (3.21)$$

$$\begin{aligned} \sum_{j \in [t, t + \text{TA}_{\max}]} q_j^p &\leq \text{TA}_{\max} + \delta_t^p \quad \forall p \in P, \forall t \in [0, |\Gamma| - \text{TA}_{\max}], \\ \sum_{j \in [t, t + \text{TA}_{\max}]} q_j^o &\leq \text{TA}_{\max} + \delta_t^o \quad \forall o \in O, \forall t \in [0, |\Gamma| - \text{TA}_{\max}], \\ \sum_{j \in [t, t + \text{TA}_{\max}]} q_j^i &\leq \text{TA}_{\max} + \delta_t^i \quad \forall i \in I, \forall t \in [0, |\Gamma| - \text{TA}_{\max}]. \end{aligned} \quad (3.22)$$

The constraints on maximum allowed time away from base can thus be integrated in our model by adding some variables and constraints: their number is proportional to (number of crew members) * (number of days in the planning period).

6.4. The maximum number of overnights away from base for consecutive working days

Another type of constraints that can be imposed by the work convention is the maximum total number of nights (denoted by TO_{\max}) spent by a crew member away from his/her base for any R consecutive working days.

For each day t of the planning period Γ ($t \in [1, |\Gamma| - 1]$) and each crew member k ($k \in P \cup O \cup I$), define the binary variable h_t^k which is equal to 1 if the crew member k will spend the night ($t, t + 1$) away from his/her base, and zero otherwise.

Using the same notation as in Section 6.3, this type of constraints can be taken into account by adding the following inequalities as well as constraints (3.19):

$$\begin{aligned} q_t^p + q_{t+1}^p - 1 &\leq h_t^p \quad \forall p \in P, \forall t \in [1, |\Gamma| - 1], \\ q_t^o + q_{t+1}^o - 1 &\leq h_t^o \quad \forall o \in O, \forall t \in [1, |\Gamma| - 1], \\ q_t^i + q_{t+1}^i - 1 &\leq h_t^i \quad \forall i \in I, \forall t \in [1, |\Gamma| - 1]. \end{aligned} \quad (3.23)$$

$$\begin{aligned} \sum_{j \in [t, t + R - 1]} h_j^p &\leq \text{TO}_{\max} \quad \forall p \in P, \forall t \in [1, |\Gamma| - R], \\ \sum_{j \in [t, t + R - 1]} h_j^o &\leq \text{TO}_{\max} \quad \forall o \in O, \forall t \in [1, |\Gamma| - R], \\ \sum_{j \in [t, t + R - 1]} h_j^i &\leq \text{TO}_{\max} \quad \forall i \in I, \forall t \in [1, |\Gamma| - R]. \end{aligned} \quad (3.24)$$

Constraints (3.23) ensure that variables h_t^k are taking the right values to express whether the crew member k is away from base or not for the night ($t, t + 1$). For each crew member, the inequalities (3.24) express the constraints on the total number of overnights away from base for any R consecutive working days.

This formulation can also take into account the cost c_t^k of an overnight in day t , for crew member k . In this case, the objective function (3.1) will have additional cost terms and will be expressed as follows:

$$\text{Minimize } \sum_{p \in P} \text{HS}_p + \sum_{o \in O} \text{HS}_o + \sum_{i \in I} \text{HS}_i + \sum_{t \in \Gamma} \sum_{p \in P} c_t^p h_t^p + \sum_{t \in \Gamma} \sum_{o \in O} c_t^o h_t^o + \sum_{t \in \Gamma} \sum_{i \in I} c_t^i h_t^i. \quad (3.25)$$

7. Conclusion

In this paper, we have studied the Crew Assignment Problem arising in the context of airline companies operating short and medium haul flights and having specific constraints. This kind of problem is usually decomposed, in the literature, into two independent problems, formulated and solved sequentially: (a) the Crew Pairing Problem and (b) the Working Schedules Construction Problem.

To the best of our knowledge, this paper represents the first attempt to solve these two problems *simultaneously* while complying with specific regulation rules and collective agreements. It presents a new formulation of the Crew Assignment Problem as a large scale integer linear program with a general structure.

The suggested enhancement of this formulation by replacing a large number of binary exclusion constraints by stronger and less numerous ones (the clique constraints) has led to an important improvement in computation time and quality of integer solutions. In fact, the computational results carried out on 20 real test problems have shown that standard integer linear programming techniques solved optimally the majority of the problems and gave good approximate integer solutions for others.

In order to improve the efficiency of the proposed exact methods, a heuristic method was also designed. Based on a rounding strategy embedded in a partial tree search procedure, this method has solved optimally half of the tested problems, in less than 40 minutes and given approximate integer solutions with good quality (within 5% off the lower bound) for the remaining problems in less than 47 minutes using CPLEX 6.0.2.

In spite of the huge size of the considered integer linear programs (up to 17,000 variables and 268,316 constraints with the first formulation and up to 17,000 variables and 12,791 constraints with the enhanced formulation), our approach has enabled us to find optimal or good approximate solutions within reasonable computation times.

We notice that the model discussed and solved here considers a specific application through which a general methodology has been developed, using state-of-the-art integer programming concepts and tools.

This model does not pretend to be readily applicable to all types of Crew Assignment Problems in all airline companies, because of the huge variety of regulation rules and collective agreements. However, as illustrated by the various extensions discussed in Section 6, we do feel that the proposed methodology has a real potential for wider applicability.

Among the possible future research directions suggested by the present work, it might be worth investigating more extensive use of polyhedral results in order to further improve the enhanced formulation (ILP2) proposed here. In fact, we have concentrated our efforts on how to handle the binary exclusion constraints while other constraints in the model might equally deserve further investigations.

Acknowledgments

We are grateful to Mr. Fathallah from the Tunisian airline company TunisAir for his fruitful help in the description of the problem treated here and for providing real data, which allowed us to conduct our computational tests and prove the computational efficiency and the practical applicability of our approach. We also thank the referees for their valuable comments which lead to an improved revised version of the paper.

References

- Barnhart, C., Shenoi, R.G., 1998. An approximate model and Solution approach for the long-haul crew pairing problem. *Transportation Science* 32, 221–231.
- Berge, C., 1970. *Graphes et Hypergraphes*, DUNOD, Paris.

- Desaulniers, G., Desrosiers, J., Dumas, Y., Marc, S., Rioux, B., Solomon, M., Soumis, F., 1997. Crew pairing at Air France. *European Journal of Operational Research* 97, 245–259.
- Gamache, M., Soumis, F., Marquis, G., Desrosiers, J., 1994. A column generation approach for large scale aricrew rostering problems. *Les Cahiers du GERAD*, G-94-20, Ecole des Hautes Etudes Commerciales, Montréal, Canada.
- Gamache, M., Soumis, F., Villeneuve, D., Desrosiers, J., Gélinas, E., 1998. The preferential bidding system at Air Canada. *Transportation Science* 32, 246–255.
- Golumbic, M.C., 1980. *Algorithmic Graph Theory and Perfect Graphs*. Academic Press.
- Gondran, M., Minoux, M., 1995. *Graphes et Algorithmes*. Eyrolles, Paris.
- Hoffman, K.L., Padberg, M., 1993. Solving airline crew scheduling problems by branch-and-cut. *Management Science* 39, 657–682.
- Lavoie, S., Minoux, M., Odier, E., 1988. A new approach for crew-pairing problems by column generation and application to air transport. *European Journal of Operational Research* 35, 45–58.
- Lettovsky, L., Johnson, E.L., Nemhauser, G.L., 2000. Airline crew recovery. *Transportation Science* 34, 337–348.
- Minoux, M., 1984. Column generation techniques in combinatorial optimization, a new application to crew-pairing problems. In: *Proceedings XXIVth AGIFORS Symposium*, Strasbourg, France, September 1984.
- Nemhauser, G.L., Wolsey, L.A., 1988. *Integer and Combinatorial Optimization*. John Wiley, Chichester.
- Ribeiro, C.C., Minoux, M., Penna, M.C., 1989. An optimal column-generation-with-ranking algorithm for very large scale set partitioning problems in traffic assignment. *European Journal of Operational Research* 41, 232–239.
- Stojkovic, M., Soumis, F., Desrosiers, J., 1998. The operational airline crew scheduling problem. *Transportation Science* 32, 232–245.
- Vance, P.H., Atamtürk, A., Barnhart, C., Gelman, E., Johnson, E.L., Krishna, A., Mahidhara, D., Nemhauser, G.L., Rebello, R., 1997. A heuristic Branch-and-Price approach for the airline crew pairing problem. Technical Report LEC-97-06, Georgia Institute of Technology.
- Zeghal, F.M., 2002. *Résolution de programmes linéaires en nombres entiers de grandes tailles et application à un problème d'affectation en transport aérien*. Ph.D. Dissertation, Université Paris 6, France.