

# Resolution of an Antenna-Satellite assignment problem by means of Integer Linear Programming

Rafael Vázquez<sup>1</sup>, Federico Perea<sup>2</sup>, Jorge Galán Vioque<sup>3</sup>

<sup>1</sup> *Departamento de Ingeniería Aeroespacial y Mecánica de Fluidos. Universidad de Sevilla (Spain). rvazquez1@us.es.*

<sup>2</sup> *Departamento de Estadística e Investigación Operativa Aplicadas y Calidad. Universitat Politècnica de València (Spain). perea@eio.upv.es*

<sup>3</sup> *Departamento de Matemática Aplicada II e Instituto de Matemáticas de la Universidad de Sevilla (Spain). jgv@us.es*

---

## Abstract

Ground station operators have to assign different antennas in their ground stations network to passes of satellites from customers that have requested the use of the network. However, for operators that support a high number of satellites, in many cases these requests yield conflicts (which appear when more than one satellite requests the same time slot on the same antenna). If there are many conflicts, the process of deconflicting (i.e., moving passes to other antennas or sites or cancelling them so that conflicts are avoided) is not very efficient when done manually, due to the large number of interacting requests. Thus, there is a need for an automatic tool that is able to manage the Antenna-Satellite assignment problem for a large number of passes, by considering the problem globally for a given time-frame (for instance, a week). In this paper we propose to address the deconfliction process by means of Integer Linear Programming. Models that take into account the basic deconflicting operations (moving antenna, moving site, shortening, or cancelling), are proposed and tested on real data provided by the company that posed this problem, yielding better results than the solutions obtained by their previous system.

*Keywords:* Deconfliction, Integer Programming, Antenna-Satellite Allocation.

---

## 1. Introduction

Satellite owners need the support of ground networks to be able to upload commands or download gathered data. Increasing data requirements are resulting in a considerable growth of requests of time allocation of satellite passes to ground antennas, particularly those located on strategic geographical locations—for instance, ground stations located close to the poles, which are more frequently accessible to satellite in sun-synchronous orbits (which includes the majority of Earth Observation Satellites, Barret and Curtis (1992)), and thus are experiencing a considerable increase of time allocation requests from customers. While ground networks continue to expand, and build sites in diverse locations throughout the world, the number of satellite customers continues to increase. Also, the appearance of distributed networks of small satellites (Schilling (2009)) could push the capabilities of the networks to the limit. The number of requests is leading to very complex problems of antenna-satellite allocation operations with several peculiarities, which make the assignment planning a cumbersome task if done by hand.

The satellite-antenna assignment problem is often called “Satellite Range Scheduling” (SRS) problem, for which some strategies have already been proposed in the literature. Barbulescu et al. (2002) apply the SRS to the US Air Force Satellite Control Network (AFSCN) employing 100 satellites, 16 antennas, 9 stations, 500 requests per day, producing several schedules per day. The considered objective is the reduction of the number of conflicts (typically 120). The authors observed that genetic algorithms performed better. Barbulescu et al. (2004) analyze the SRS both empirically and formally, proving that the problem is NP-complete. They also provide new algorithms improving those in Barbulescu et al. (2002). Later, Barbulescu et al. (2006a) present the evolution of the problem in the last 10 years in the AFSCN. They analyze possible alternatives to the cost function such as minimizing the sum of overlaps. The same group of authors study more heuristics for the SRS in Barbulescu et al. (2006b), combining several algorithms. Chien et al. (2012) studies the concept of timeline and reviews the generic systems that have been developed. Clement and Johnston (2005) describe the SRS for the Deep Space Network (DSN) with 16 antennas, 20 spacecrafts, four-month time-frames, 1650 passes per week. They generate and repair schedules, describing the problem and heuristics for solving it with emphasis in the re-scheduling problem. Corrao et al. (2012) integrate Genetic Algorithms, Graph Theory and Linear Programming in order to build

conflict-free plans, and apply their approach to a case study obtained from a satellite service company. Lee et al. (2008) study the scheduling of a single geostationary satellite. Marinelli et al. (2011) formulates the problem as an ILP model, which is declared as infeasible. The problem is solved by means of a Lagrangian relaxation. As a case study they apply their approach to the station Galileo. Schmidt and Schilling (2009) consider academic problems consisting of satellites and stations distributed all over the Earth. They maximize redundancy in order to solve possible failures in communication. They solve a toy example with 6 satellites and 4 stations yielding 51 contact windows. Xhafa et al. (2013) use Struggle Genetic Algorithms and STK. Zhang et al. (2014) propose ant-colony algorithms, solving examples with 17 satellites, 11 to 13 antennas yielding around 400 passes. Zufferey et al. (2008) apply graph coloring algorithms over a set of 500 realistic instances.

However, due in part to tradition, and in part to the complexities of the problem, manual elaboration of schedules are still usual procedure for ground networks managers. This problem has also arisen in the context of ground station networks (Schmidt et al. (2008); Bester (2009)) provided by research institutions for small satellites used for academic projects, which usually have some specific needs such as redundancy and flexibility, and therefore require specific algorithms.

When planning, it must be taken into account that a given satellite can be supported just by a subset of the available antennas, and during specific time intervals which must be computed by using knowledge of the satellite orbit and the antenna geographical location and allowable positions of Azimuth-Elevation for the antenna (i.e. which region of the sky is accessible for the antenna). Each pass has a default antenna, which is the one requested by the user; this would be considered as the most preferred antenna. Since different users make requests independently of one another, preferred allocations may cause *conflicts*, i.e., time intervals where requests from different passes overlap on the same antenna. Such conflicts can be addressed by a number of alternatives. First, the preferred option would be reallocating some passes to compatible antennas located in the same site; other options would be changing the pass to another site (which could imply a considerable change in the previously allocated time of access if the new site is far away from the original site), shortening the pass (up to a minimum duration) to accommodate several passes, or, if no other options are available, canceling the pass. These operations might be available only for a subset of passes if there is a number of already allocated passes that must be honored (for instance for preferred

clients or previous commitments). It also must be taken into account that some satellites or customers could be of higher priorities than others.

In this paper we propose the use of Integer Linear Programming (ILP) models to solve the problem of deconfliction posed by a ground network operator that manages several sites with dozens of antennas, from now on called “the company”. While the scheduling problem has been reported NP complete (Barbulescu et al. (2004)), using our models we have been able to solve in a reasonable time (less than a minute) real-world instances of the problem for realistic dimensions (thousands of passes over dozens of antennas) and time frames (a week) by using an open source ILP solver. The use of Integer Linear Programming modelling has proven fruitful for other space mission optimization problems, such as the problem of swath acquisition planning for multiple Earth Observation Satellites, see Galán-Vioque et al. (2011).

The remainder of this paper is structured as follows. In Section 2, the problem is formally stated and the notation used throughout the paper is introduced. The different deconfliction objectives, and the resulting models are described in Section 3, formulated as Integer Linear Programming problems. Computational results, taken from real data, are analyzed in Section 4. We finish with some concluding remarks in Section 5.

## 2. Input data

We next show which data are needed on satellites, antennas and passes and which computations are required to formulate the Antenna-Satellite assignment problem.

### 2.1. Input data

The input data of the problem are:

- The time-frame for the planning problem is an interval  $[T_0, T_f]$  given by the initial and final times  $T_0$  and  $T_f$ . At times we might refer to this interval as  $T$  (in our case, usually a week).
- A set  $\mathcal{S}$  of satellites. The orbit of the satellites can be given in any conventional format, for instance given as Two-Line Elements (TLEs) for a certain epoch (which should be close to the time-frame to be able to precisely determine the passes).

- A set  $\mathcal{A} = \{A_1, \dots, A_{n_a}\}$  of antennas, given by their geographical locations. Antennas which are geographically close to each other are considered to be in the same site, whereas antennas located far away from each other are in different sites. For each antenna, we also assume that we know its admissible range of Azimuth-Elevation, which would depend on obstacles and local geography (for instance mountains) and the required minimum elevation above the horizon to avoid atmospheric effects. This is mathematically formulated as the set  $\Omega_a = \{(\text{Az}, \text{El})\}$  of accessible points in the sky given by their azimuths and elevations.
- Other relevant input data include the minimum duration for a pass to be considered valid  $t_{\min_{sa}}$  (which can depend on the satellite and antenna) and the set of compatible antenna-satellite pairs  $\mathcal{C} \subset \mathcal{A} \times \mathcal{S}$ .

## 2.2. Computation of passes

The next step to formulate the problem is to calculate the set of possible passes for all satellites  $\mathcal{S}$  and antennas  $\mathcal{A}$ . For each revolution of satellite  $s \in \mathcal{S}$  over the Earth we obtain a pass  $P$  when there are time intervals of the form  $[t_0, t_1] \subset T$  during which a satellite is accessible for one or more antennas  $a \in \mathcal{A}$ , given that the duration of the accesses,  $t_1 - t_0$ , is greater or equal than the minimum duration  $t_{\min_{sa}}$  and the antenna is compatible with the satellite requirements, i.e.  $(a, s) \in \mathcal{C}$ . We assume that there is an antenna to which the pass is originally assigned; the possible antennas to which the pass can be assigned (different from the original one) are called alternative antennas.

To compute the passes, the first step is to propagate the orbital elements of the satellites during the mission time-frame. This can be done using any of the many possible methods available in the literature, which incorporate more or less accurate models of orbit perturbations (see for instance Vallado and McClain (2007), and references therein). Once the elements are known at all times  $t \in T$ , the vector position  $\vec{r}_s(t)$  in the geographical reference frame (that rotates with the Earth) can be computed (Curtis (2009)), for all  $s \in \mathcal{S}$ . Then, using the antenna geographical coordinates the vector position of the antennas  $\vec{r}_a$  for all  $a \in \mathcal{A}$  can be also computed. Then, by projecting the relative position of the satellite with respect to the antenna,  $\vec{r}_{as}(t) = \vec{r}_s(t) - \vec{r}_a$  on the topocentric frame centered in the respective antenna, one can compute the azimuth and elevation for each compatible antenna-satellite pair,  $(\text{Az}_{as}(t), h_{as}(t))$  for  $(a, s) \in \mathcal{C}$ . Each of the time intervals

in which  $(Az_{as}(t), h_{as}(t)) \in \Omega_a$  for at least the minimum duration  $t_{min,a}$  constitutes an access which will give an alternative to the pass. Satellites will generate a pass only each time the groundtrack passes close to a given antenna (for most locations once or twice a day).

Notice that oftentimes satellites will have sun-synchronous orbits since these are the most frequently used orbit for Earth Observation Satellites (due to constant lighting properties), which transmit large amounts of data and therefore constitute a large amount of the satellites requesting passes. Given that these satellites have almost-polar low orbits, it is advantageous to locate bases close to the poles of the Earth, since then one would obtain passes on most orbit revolutions (around 13 passes each day).

### 2.3. Additional input data for the passes

Once the passes have been computed, we have sets  $\mathcal{P} = \{P_1, \dots, P_{n_p}\}$ ,  $\mathcal{A} = \{A_1, \dots, A_{n_a}\}$  consisting of  $n_p$  passes and  $n_a$  antennas, respectively. The time interval that each of these passes  $P_i$  covers in a given antenna  $A_k$  is given by the intervals  $[\alpha_{i,k}, \beta_{i,k}]$ . Passes are classified as accepted or free. In the first case, the requested antenna and time slot are considered to be fixed, while for free passes, one is allowed to change the requested conditions (antenna requested, pass length), or even to cancel them in order to improve the conflicts status. The following parameters are additional input data for our problem:

- $F \subset \{1, \dots, n_p\}$  is the set of free passes, i.e., passes which can be modified with respect to the original request.
- $p_{ik}$  : priority of pass  $P_i$  in antenna  $A_k$ .  $p_{ik} < p_{i'k}$  means that  $P_i$  is more preferred than  $P_{i'}$  for antenna  $A_k$ .
- $a_{ik}$  : minimum length of time in which  $P_i$  must be active if antenna  $A_k$  is to get its data (Such length of time includes pre and post-processing times, which depend on the satellite-antenna pair.)
- The binary parameter  $e_{ik}$  takes the value 1 if pass  $P_i$  is originally requested to be assigned to antenna  $A_k$ . We assume that  $\sum_{k=1}^{n_a} e_{ik} = 1$  for every pass  $P_i$ , that is, originally pass  $P_i$  is assigned to one and only one antenna.
- $C_i$  is the set of antennas which have access and are compatible with pass  $P_i$ . The binary parameter  $c_{ik}$  takes value 1 if pass  $P_i$  has access

and is compatible with antenna  $A_k$ , and 0 otherwise. In other words,  $c_{ik} = 1$  if and only if  $k \in C_i$ .

- $(\alpha_{ik}, \beta_{ik})$  is the period of time in which pass  $P_i$  has access to antenna  $A_k$ ,  $k \in C_i$ .

#### 2.4. Computation of time intervals

Once all the passes have been computed and the input data on the passes has been gathered, the formulation of the Antenna-Satellite assignment problem requires the analysis of the different time intervals in which, for a given antenna, possible passes can overlap.

- For each antenna  $A_k$ , we consider the intersections of all possible intervals of time  $(\alpha_{ik}, \beta_{ik})$  of compatible passes. The result is  $n_s$  ( $n_s \leq 2n_p - 1$ ) intervals  $I_{1k}, \dots, I_{n_s k}$ , with lengths  $l_{1k}, \dots, l_{n_s k}$ . The intervals are sorted in such a way that the beginning of interval  $I_{jk}$  is equal to the end of interval  $I_{j-1,k}$ , or larger if there is a “gap” during which no compatible passes exist for the antenna.
- $S_{ik} \subset \{1, 2, \dots, n_s\}$  is the set of indices of the sorted intervals  $\{I_{jk}, j = 1, \dots, n_s\}$ , in which  $P_i$  can be active in antenna  $A_k$ . Taking into account the accepted (fixed) passes, in some intervals, the antenna will already be occupied by these accepted passes, and thus, by construction, for such intervals  $I_{jk}$  we have  $j \notin S_{ik}$ .

We show a simplified example of such a timeline in Figure 1, which considers three passes ( $p_1$ ,  $p_2$  and  $p_3$ ) and two antennas ( $A_1$  and  $A_2$ ). For the sake of simplicity the passes could be located in either antenna with the same start and end times. From the figure we see that the beginning of  $p_1$  in either antenna is  $\alpha_{11} = \alpha_{12} = t_0$ , the ending of  $p_1$  is  $\beta_{11} = \beta_{12} = t_2$ , and similarly for  $p_2$  we have  $\alpha_{21} = \alpha_{22} = t_1$  and  $\beta_{21} = \beta_{22} = t_4$ , and for  $p_3$  we have  $\alpha_{31} = \alpha_{32} = t_3$  and  $\beta_{31} = \beta_{32} = t_5$ . The resulting intervals for both antennas are  $I_{11} = I_{12} = [t_0, t_1]$ ,  $I_{21} = I_{22} = [t_1, t_2]$ ,  $I_{31} = I_{32} = [t_2, t_3]$ ,  $I_{41} = I_{42} = [t_3, t_4]$  and  $I_{51} = I_{52} = [t_4, t_5]$ . Assuming no passes are fixed we have  $S_{11} = S_{12} = \{1, 2\}$ ,  $S_{21} = S_{22} = \{2, 3, 4\}$  and  $S_{31} = S_{32} = \{4, 5\}$ , which means that the first pass spans (in either antenna) the time intervals 1 and 2, the second pass the time intervals 2, 3 and 4, and third pass the time intervals 4 and 5. If Figure 1 represents the originally proposed schedule then we see there is a conflict in antenna 1 between passes 1 and 2, which can

be trivially resolved either by moving pass 1 to antenna 2 (solution 1) or by switching antenna between passes 2 and 3 (solution 2), as show in Figure 2.

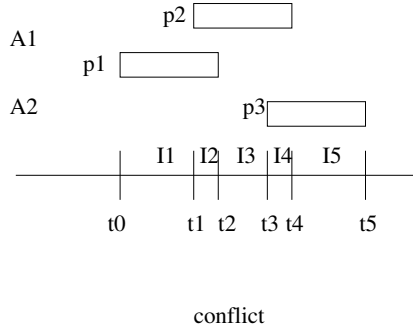


Figure 1: Simple example of construction of time intervals and conflict.

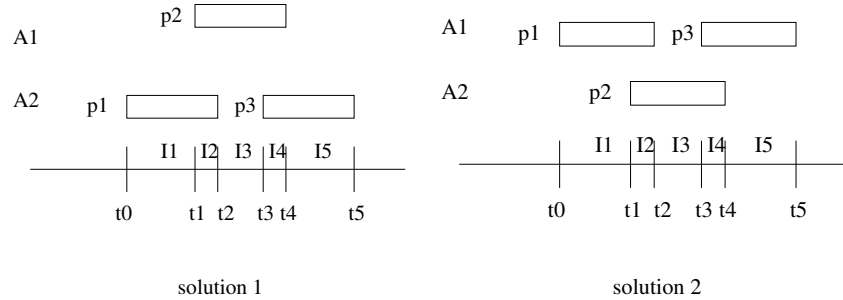


Figure 2: Two possible solutions for the example.

### 3. ILP Models

The models developed here aim at solving conflicts. A conflict is produced if there is an overlap, i.e if over the same time interval, two passes are assigned to the same antenna. Using our notation, this occurs when there exist an antenna  $A_k$  and an interval  $I_{jk}$  such that

$$\sum_{i \in \mathcal{P}: j \in S_{ik}} e_{ik} > 1.$$

When conflicts are found, the company requires that one of the following three deconflicting operations is done:



1. Moving passes to a different antenna (see Section 3.1) which could be in the same site or in another site.
2. Shortening a pass time allocation on the antenna (Section 3.2).
3. Cancellation of passes (Section 3.3).

These operations are written in order of preference, i.e., first, if possible, conflicts should be addressed by moving passes to antennas different to the default ones (and if possible within the same site). Only if conflicts exist after this operation, some passes should be shortened (taking into account the minimum duration of a pass). Still, if some conflicts persist, some passes (beginning with those with lower priority) should be canceled until a feasible solution is found.

In what follows we formulate an integer linear programming (ILP) model that includes all the possible deconflicting operations and at the same time optimizing a performance index directly related with the priority of the different satellites.

### 3.1. Moving passes to a different antenna

We address the problem of re-allocating free passes to antennas so that conflicts disappear. Priorities of the different passes are taken into account.

Our ILP model uses the following binary variables:

- For each  $i \in F$ , for each  $k \in C_i$ , i.e., for each free pass and compatible antenna, define the binary variable  $y_{ik}$  which takes the value 1 if pass  $P_i$  is assigned to antenna  $A_k$  and 0 otherwise.

The constraints of the model would be:

- Every free pass has to be assigned to one and only one antenna.

$$\sum_{k \in C_i} y_{ik} = 1, \forall i \in F. \quad (1)$$

- For a given antenna  $A_k$  and a time interval  $I_{jk}$  available for free passes, i.e., with  $\cup_{i \in F} S_{ik} \neq \emptyset$ , there should be no conflict among the  $n_p$  passes.

$$\sum_{i \in F: j \in S_{ik}, k \in C_i} y_{ik} \leq 1, \forall k, j : \cup_{i \in F} S_{ik} \neq \emptyset. \quad (2)$$

The objective is to keep as many passes allocated to the requested antennas as possible, taking into account the different priorities. We model this as the maximization of the cost index  $J_1$  defined as the sum of priorities of passes that remain assigned to the requested antennas.

$$J_1 = \sum_{i \in F} \sum_{k \in C_i} (p^* - p_{ik} + 1) \xi_{ik} y_{ik}, \quad (3)$$

where  $p^* = \max_i p_i$  and  $\xi_{ik}$  is a weighting function that is defined as

$$\xi_{ik} = \begin{cases} 1, & e_{ik} = 1, \\ 1/2, & e_{ik} = 0, A_k \text{ in the same site,} \\ 1/4, & e_{ik} = 0, A_k \text{ in a different site.} \end{cases} \quad (4)$$

Thus we favor to stay in the initially assigned antenna and penalize changing antenna and site.

In the example presented in Figure 1, we would have  $e_{11} = e_{21} = e_{32} = 1$ , and  $e_{12} = e_{22} = e_{31} = 0$ . Assume for simplicity that all preferences are the same and equal to 1 and that both antennas are located in the same site. Then we would have 6 binary variables  $y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}$ . Constraint (1), which implies that every pass is assigned only to one (compatible) antenna, would read

$$y_{11} + y_{12} = 1, \quad (5)$$

$$y_{21} + y_{22} = 1, \quad (6)$$

$$y_{31} + y_{32} = 1. \quad (7)$$

Constraint (2), which implies that a solution has no overlaps, is constructed by looking at the potential overlaps interval by interval, and would read

$$y_{11} + y_{21} \leq 1, \quad (8)$$

$$y_{21} + y_{31} \leq 1, \quad (9)$$

$$y_{12} + y_{22} \leq 1, \quad (10)$$

$$y_{22} + y_{32} \leq 1. \quad (11)$$

Finally the cost index (3) becomes

$$J_1 = y_{11} + \frac{y_{12}}{2} + y_{21} + \frac{y_{22}}{2} + \frac{y_{31}}{2} + y_{32}. \quad (12)$$

Thus, maximizing  $J$  subject to constraints (5)–(11) one obtains  $y_{12} = y_{21} = y_{32} = 1$  and  $y_{11} = y_{22} = y_{31} = 0$ , i.e., the solution shown in Figure 2 (left), which represents a conflict-free solution optimizing the number of changes according to (3).

### 3.2. Shortening

If the re-assignment of antennas does not solve all conflicts, the possibility of shortening (for passes that allow it) is considered.

To model this possibility, consider, for each pair of compatible pass-antenna,  $P_i, A_k$ , and for each connected<sup>1</sup> subset of  $S_{ik}^\ell \subset S_{ik}$  that satisfies  $\sum_{j \in S_{ik}^\ell} l_{jk} \geq a_{ik}$ , we define a subpass  $P_{ik}^\ell$  which will span a time interval  $[\delta_{ik}^\ell, \rho_{ik}^\ell]$ . The objective is now to choose one such subpass for each pass  $P_i$  so that the objective function is optimized, and no conflicts arise. This can be modeled as follows.

For each each subpass  $P_{ik}^\ell$  we define the following set of binary variables:

- $y_{i\ell k} = 1$  if subpass  $P_{ik}^\ell$  is selected, and zero otherwise.

The objective is now twofold: to maximize the active time of passes and to keep passes in the antennas they were originally assigned to or at least keep the preference as in Section 3.1. This could be modeled by defining an additional cost index  $J_2 = \sum_{i,\ell,k} (\rho_{ik}^\ell - \delta_{ik}^\ell) y_{i\ell k}$  which counts the active time of passes, to later on maximize a linear combination of  $J_1$  and  $J_2$  defined as:

$$(1 - \gamma)J_1 + \gamma J_2, \quad (13)$$

where parameter  $\gamma \in [0, 1]$  measures the importance given to each of the two objectives ( $\gamma = 0$  means that the only objective is to keep as may passes allocated to the requested antennas as possible,  $\gamma = 1$  means that the only objective is to maximize the active time of passes, any other  $\gamma \in (0, 1)$  maximizes a combination of both objectives).

The constraints of the model are the constraints of Section 3.1 substituting the constraints that include shortened passes by

$$\sum_{\ell,k} y_{i\ell k} = 1, \quad \forall i \quad (14)$$

$$\sum_{i \in F: j \in S_{ik}, k \in C_i} y_{ik} + \sum_{i,\ell: j \in S_{ik}^\ell} y_{i\ell k} \leq 1, \quad \forall k, j \quad (15)$$

Equations (14) force that, for each pass exactly one subpass is assigned to exactly one antenna. Equations (15) impose that for each antenna and each interval, at most one pass or shortened subpass is active.

---

<sup>1</sup>A set of intervals is considered connected if their union forms a unique interval.

In the example presented in Figure 1, we could force the necessity of subpasses if we assume, for instance, that pass 1 is compatible only with antenna 1 and pass 3 is compatible only with antenna 2. Thus the variables  $y_{12}$  and  $y_{31}$  disappear and constraints (5)–(11) are reduced to

$$y_{11} = 1, \quad (16)$$

$$y_{21} + y_{22} = 1, \quad (17)$$

$$y_{32} = 1, \quad (18)$$

$$y_{11} + y_{21} \leq 1, \quad (19)$$

$$y_{22} + y_{32} \leq 1. \quad (20)$$

Notice that this results in an infeasible problem; it is not possible to use any of the solutions pictured in Figure 2 as there is no antenna to which we can move pass 2 without overlapping with other passes. Thus we need to include the possibility of shortening. To simplify, assume also that passes 1 and 3 cannot be shortened. Thus one considers shortening pass 2. The possible subpasses would span the following time intervals:  $[t_1, t_4]$ ,  $[t_1, t_3]$ ,  $[t_1, t_2]$ ,  $[t_2, t_4]$ ,  $[t_2, t_3]$ ,  $[t_3, t_4]$ . Assuming that, of these,  $[t_1, t_2]$  and  $[t_3, t_4]$  are too short to be considered, we would end up with four potential subpasses for each antenna, namely  $P_{21}^1, P_{21}^2, P_{21}^3, P_{21}^4$  and  $P_{22}^1, P_{22}^2, P_{22}^3, P_{22}^4$ , and the corresponding binary variables  $y_{211}, y_{221}, y_{231}, y_{241}$  and  $y_{212}, y_{222}, y_{232}, y_{242}$ . The possible subpasses are shown in Figure 3 (only for antenna 1 to avoid cluttering the figure). The

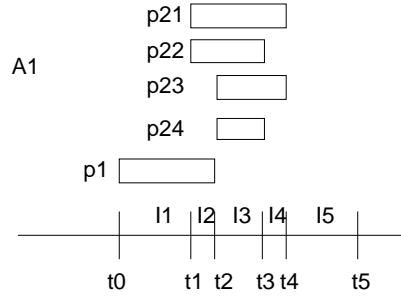


Figure 3: Construction of subpasses (only shown for pass 2 in antenna 1).

respective intervals would be  $[\delta_{21}^1, \rho_{21}^1] = [t_1, t_4]$ ,  $[\delta_{21}^2, \rho_{21}^2] = [t_1, t_3]$ ,  $[\delta_{21}^3, \rho_{21}^3] = [t_1, t_2]$ ,  $[\delta_{21}^4, \rho_{21}^4] = [t_2, t_4]$ , and equally  $[\delta_{22}^1, \rho_{22}^1] = [t_1, t_4]$ ,  $[\delta_{22}^2, \rho_{22}^2] = [t_1, t_3]$ ,

$[\delta_{22}^3, \rho_{22}^3] = [t_2, t_4]$ ,  $[\delta_{22}^4, \rho_{22}^4] = [t_2, t_3]$ . The new set of constraints would be

$$y_{11} = 1, \quad (21)$$

$$\begin{aligned} & y_{211} + y_{221} + y_{231} + y_{241} \\ & + y_{212} + y_{222} + y_{232} + y_{242} = 1, \end{aligned} \quad (22)$$

$$y_{32} = 1, \quad (23)$$

$$y_{11} + y_{211} + y_{221} \leq 1, \quad (24)$$

$$y_{211} + y_{221} + y_{231} + y_{241} \leq 1, \quad (25)$$

$$y_{211} + y_{231} + y_{241} \leq 1, \quad (26)$$

$$y_{211} + y_{231} \leq 1, \quad (27)$$

$$y_{212} + y_{222} \leq 1, \quad (28)$$

$$y_{212} + y_{222} + y_{232} + y_{242} \leq 1, \quad (29)$$

$$y_{212} + y_{232} + y_{242} \leq 1, \quad (30)$$

$$y_{32} + y_{212} + y_{232} \leq 1. \quad (31)$$

The cost function now becomes

$$\begin{aligned} J = & y_{11} + y_{32} + (\gamma(t_4 - t_1) + 1 - \gamma)y_{211} \\ & + (\gamma(t_3 - t_1) + 1 - \gamma)y_{221} + (\gamma(t_4 - t_2) + 1 - \gamma)y_{231} \\ & + (\gamma(t_3 - t_2) + 1 - \gamma)y_{241} + (\gamma(t_4 - t_1) + \frac{1 - \gamma}{2})y_{212} \\ & + (\gamma(t_3 - t_1) + \frac{1 - \gamma}{2})y_{222} + (\gamma(t_4 - t_2) + \frac{1 - \gamma}{2})y_{232} \\ & + (\gamma(t_3 - t_2) + \frac{1 - \gamma}{2})y_{242}. \end{aligned} \quad (32)$$

The solution would depend on the particular values of  $\gamma$  and the times. If one chooses  $\gamma = 1/2$  and the time intervals are to scale in Figure 1, then the solution is to choose subpass  $P_{231}$  as shown in Figure 4.

### 3.3. Cancelling

If it is not possible to resolve all conflicts, the pass(es) with lowest priority should be cancelled. For this situation we use the same variables as in Section 3.1 and Section 3.2. To find a feasible solution, we allow that passes are assigned either to one antenna or none:

$$\sum_{k \in C_i} y_{ik} \leq 1, \forall i \in F, \quad (33)$$

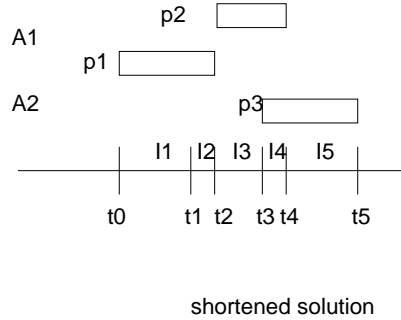


Figure 4: Simple example of a shortened solution.

and similarly subpasses can be rejected as well

$$\sum_{\ell,k} y_{i\ell k} \leq 1, \forall i. \quad (34)$$

The remaining constraints and cost functions are as in Section 3.1 and Section 3.2.

In the example presented in Figure 1, we could force the necessity of cancelling if we assume, as before, that pass 1 is compatible only with antenna 1 and pass 3 is compatible only with antenna 2, but now instead of the possibility of shortening we only allow cancellations. Thus constraints (5)–(11) are reduced to

$$y_{11} \leq 1, \quad (35)$$

$$y_{21} + y_{22} \leq 1, \quad (36)$$

$$y_{32} \leq 1, \quad (37)$$

$$y_{11} + y_{21} \leq 1, \quad (38)$$

$$y_{22} + y_{32} \leq 1. \quad (39)$$

The cost index (12) remains the same. The obvious solution is to cancel either pass 1 or pass 2; since we have assumed they have the same priority, both solutions are equally valid. If not, one would cancel the lowest priority pass.

#### 4. Computational results

This section aims at showing how our procedures are able to handle realistic instances, with real schedules provided by the company in seconds.

More specifically, we present the results on 10 instances in which about 3000 passes (a typical quantity of requests for a busy week) were to be managed, see Table 1. We should point out that in some cases the data correspond to very challenging examples with a high number of satellites and sites involved in the conflicts.

Columns in this table represent:

- “Passes”: number of passes considered.
- “Conflicts”: number of conflicts arising in the corresponding instance.
- “Cancelations”: represent the total number of canceled passes.
- Column “Shortenings” represent the number of passes that were shortened.
- “Movements” represents the number of movements done in each instance (between parenthesis those movements done to other sites).
- Columns “Variables” and “Constraints” represent the number of variables and constraints in the corresponding ILP problem, respectively.
- Column “Time” gives the necessary computational time to solve the corresponding instance, in seconds.

These instances were built from real schedules for one week of requests. To perform the deconflicting operations and obtain an optimal deconflicted schedule, the following computational setup was used:

- The calculation of alternative passes from the antenna locations (and their admissible range of Azimuth-Elevation) and the satellite orbital elements (given as TLEs) was performed by SaVoir software ([www.taitussoftware.com](http://www.taitussoftware.com)), a powerful Satellite Planning and Mission Analysis tool.
- Given all passes and their possible alternatives, we developed (in C++) a software tool that efficiently computes all time intervals and possible subpasses following the steps of Section 2.4 and subsequently computed the problem constraints and coefficients of the cost function according to Section 3.

- To solve the resulting ILP problem, the free-software package Lpsolve solver was used. We have used version 5.5.2.0 for windows 32 bits(<http://lpsolve.sourceforge.net/5.5/>).
- All experiments were run on a laptop, Intel Core i7 2 GHz and 4 GB of RAM memory, O.S. Windows 7 Professional 32 bits.

The following straightforward conclusions to these results can be noted :

- On average, 2605.2 passes were analyzed in each instance with 287.9 conflicts. The corresponding problems had 9154.7 variables and 9013.7 constraints. The average computational time required to solve these problems was 64.8 seconds.
- Only 3.13% of passes were canceled.
- Only 0.02% of passes were shortened (note however that many of the passes did not allow for shortening).
- 19.32% of passes were moved to other antennas (0.38% to other site).

From the results cast in table 1 we can affirm that the algorithm proposed in this project provides a *real time* (around a minute) optimal solution for problems of considerable size (up to 4000 passes) that typically correspond to a full week of operation.

The computation time does not correlate directly with the size or the number of conflicts but depends more on the complexity of the conflicts.

The operation constraints and priorities have been efficiently integrated in the modeling and different adjustments can be achieved by tuning the cost weights according to the specifications of the passes.

## 5. Conclusions

In this paper we have introduced Integer Linear Programming models to efficiently manage the scheduling of passes for a multi-antenna, multi-site ground network serving numerous customers. A dramatic growth in the number of requests has rendered manual scheduling planning virtually unfeasible. The aim of our methods is to solve conflicts in the best possible way while respecting preferred assignments and priorities. A conflict appears when, over



the same time interval, two or more passes are scheduled to the same antenna. Several possibilities can be applied to solve such conflicts: movement of passes to other antennas (possibly located on other sites), shortening of acquisitions or cancelation of passes. We have modeled these problems using a basic tool of operations research: Integer Linear Programming.

Our models have been tested over a number of realistic instances provided by the ground network operator, which was previously scheduling the passes manually. Conversations with the company representatives let us know that the performance of our procedures exceeded the operator's expectations in terms of speed and quality of solutions (few number of movements, even fewer number of cancelations) with respect to their previous manual system.

Among future possible refinements, we could mention the inclusion of additional objectives, such as fairness criteria (penalizing multiple cancellations for the same customer) or the development of advanced tools such as adaptive online scheduling (which would imply a scheduler running online with capabilities such as including last-minute requests for passes as they come, or immediately adapting to dynamic constraints, for instance, antenna failures).

## Acknowledgments

The authors acknowledge the cooperation of Taitus Software ([www.taitussoftware.com](http://www.taitussoftware.com)) and its team and in particular its founder and CEO (Felipe Martín Crespo), which introduced this problem to us and provided integration with its orbital mechanics visual software SaVoir. We also acknowledge the cooperation of Kongsberg Satellite Services AS (KSAT). JGV acknowledges financial support through grants MTM2012-31821 and P12-FQM-1658.

## References

- Barbulescu, L., Howe, A., Whitley, D., 2006a. Afsn scheduling: How the problem and solution have evolved. *Mathematical and Computer Modelling* 43, 1023–1037.
- Barbulescu, L., Howe, A. E., Watson, J. P., Whitley, L. D., 2002. Satellite range scheduling: A comparison of genetic, heuristic and local search. *Lecture Notes in Computer Science* 2439, 611–620.
- Barbulescu, L., Howe, A. E., Whitley, L. D., Roberts, M., 2006b. Understanding algorithm performance on an oversubscribed scheduling application. *Journal of Artificial Intelligence Research* 27, 577–615.

- Barbulescu, L., Watson, J.-P., Whitley, L., Howe, A., 2004. Scheduling space-ground communications for the air force satellite control network. *Journal of Scheduling* 7, 7–34.
- Barret, E., Curtis, D., 1992. *Introduction to Environmental Remote Sensing*, 3rd Edition. Springer.
- Bester, M., 2009. Automated multi-mission scheduling and control center operations at uc berkeley. In: *IEEE Aerospace conference*. pp. 1–12.
- Chien, S. A., Johnston, M., Frank, J., Giuliano, M., Kavelaars, A., Lenzen, C., Policella, N., June 2012. In: *The 12th International Conference on Space Operations*. Stockholm, Sweden.
- Clement, B., Johnston, M. D., 2005. The deep space network scheduling problem. In: Press, A. (Ed.), *IAAI*. Pittsburgh, PA, USA.
- Corrao, G., Falone, R., Gambi, E., Spinsante, S., 2012. Ground station activity planning through a multi-algorithm optimisation approach. In: *IEEE First AESS European Conference on Satellite Telecommunications (ES-TEL)*. Rome, Italy.
- Curtis, H. D., 2009. *Orbital Mechanics for Engineering Students*, 2nd Edition. Butterworth-Heinemann.
- Galán-Vioque, J., Vázquez, R., Carrizosa, E., Vera, C., Perea, F., Martín, F., 2011. Towards a visual tool for swath acquisition planning in multiple-mission eoss. In: *IWPSS 2011 Workshop Proceedings*. Darmstad, Germany.
- Lee, S., Jung, W. C., Kim, J.-H., 2008. Task scheduling algorithm for the communication, ocean, and meteorological satellite. *ETRI Journal* 30 (1), 1–12.
- Marinelli, F., Nocella, S., Rossi, F., Smriglio, S., 2011. A lagrangian heuristic for satellite range scheduling with resource constraints. *Computers & Operations Research* 38, 1572–1583.
- Schilling, K., 2009. Earth observation by distributed networks of small satellites. In: *International Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering (ICICI-BME)*.

- Schmidt, M., Rybysc, M., Schilling, K., 2008. A scheduling system for small ground station networks. In: SpaceOps 2008 Conference hosted by ESA and EUMETSAT in association with AIAA.
- Schmidt, M., Schilling, K., July 2009. A scheduling system with redundant scheduling capabilities. In: Third IEEE International Conference on Space Mission Challenges for Information Technology. Pasadena, California, USA.
- Vallado, D., McClain, W., 2007. Fundamentals of Astrodynamics and Applications, 3rd Edition. Microcosm Press/Springer.
- Xhafa, F., Herrero, X., Barolli, A., Barolli, L., 2013. Evaluation of struggle strategy in genetic algorithms for ground stations scheduling problem. *Journal of Computer and System Sciences* 79, 1086–1100.
- Zhang, Z., Zhang, N., Feng, Z., 2014. Multi-satellite control resource scheduling based on ant colony optimization. *Expert Systems with Applications* 41, 2816–2823.
- Zufferey, N., Amstutz, P., Giaccari, P., 2008. Graph colouring approaches for a satellite range scheduling problem. *Journal of Scheduling* 11, 263–277.

Passes	Conflicts	Cancellations	Shortenings	Movements total (other site)	Variables	Constraints	Time (s)
3356	537	116	1	839 (13)	8090	9322	64
3066	196	75	3	703 (5)	11219	12157	61
3356	231	94	0	517 (6)	12465	12245	73
3566	306	114	1	561 (11)	12360	12714	73
3470	253	100	0	557 (17)	12016	12788	80
3408	196	91	0	478 (5)	12289	12025	71
1573	68	0	0	487 (7)	7143	4467	64
384	33	21	0	101 (0)	1247	1075	22
1586	6	2	0	479 (2)	7892	4964	63
2287	1053	250	4	312 (35)	6820	8380	77

Table 1: Deconflicting results on 10 real scheduling problem instances.