Earth Observation Satellite Management

E. BENSANA   bensana@cert.fr
ONERA, Centre de Toulouse,[1] *2, avenue Idouard Belin, BP 4025, 31055 Toulouse cedex 4, France*

M. LEMAîTRE   mlemaitre@cert.fr
ONERA, Centre de Toulouse,[1] *2, avenue Idouard Belin, BP 4025, 31055 Toulouse cedex 4, France*

G. VERFAILLIE   verfaillie@cert.fr
ONERA, Centre de Toulouse,[1] *2, avenue Idouard Belin, BP 4025, 31055 Toulouse cedex 4, France*

Abstract. The daily management of an earth observation satellite is a challenging combinatorial optimization problem. This problem can be roughly stated as follows: given (1) a set of candidate images for the next day, each one associated with a weight reflecting its importance, (2) a set of imperative constraints expressing physical limitations (no overlapping images, sufficient transition times, bounded instantaneous data flow and recording capacity), select a subset of candidates which meets all the constraints and maximizes the sum of the weights of the selected candidates.

It can be easily cast in variants of the *CSP*, *ILP* or *SAT* frameworks. As a benchmark, we propose to the *CONSTRAINTS* community a set of instances, which have been produced from a simulator of the order book of the future satellite *SPOT5*. The fact that only some of them have been optimally solved should make them very attractive.

Keywords: benchmarks, earth observation satellite management, constraint satisfaction, discrete optimization

1. Problem Description

1.1. The SPOT Satellites

The *SPOT* satellites constitute a family of earth optical observation satellites, which are developed by the *CNES*[2] (French *Centre National d'Études Spatiales*) and exploited by the *SPOT Image*[3] company. The first (*SPOT1*) was launched in 1986. The launch of the last (*SPOT5*) is planned for 2002. All of them use a circular, near-polar, sun-synchronous orbit, with about 14 revolutions per day and a cycle of 26 days. They are equipped with *HRV* (*High Resolution Visible*) imaging instruments, with adjustable oblique viewing capability. *SPOT5*, which is the target of the proposed benchmark, will be equipped with three instruments (front, middle, and rear). Mono-images need one of the three instruments. Stereo-images need the front and the rear instruments. If it is possible, image data are directly down-linked to a ground receiving station. If not, they are stored using the on-board recorders and down-linked when the satellite is within range of a receiving station. The *SPOT Image* company receives imaging orders coming from clients all over the world and is in charge of satisfying them as well as possible. Long-term and short-term management systems are used with this aim. The problem we present is the shortest term management problem, which consists in deciding each day which images will be taken the next day and how to take them.

1.2. *Informal Description*

The daily management problem of the *SPOT5* satellite can be informally described as
follows. Given:

– a set *I* of images which could be taken the next day from at least one of the three
  instruments, with respect to the satellite trajectory and to the oblique viewing capability;

– for each image, a positive integer weight expressing its importance and resulting from
  the aggregation of several criteria like the client importance, the demand urgency and
  the meteorological forecasts;

– for each image, a set of ways of taking it: up to three for a mono-image and only one
  for a stereo;

– a set of imperative constraints: non overlapping and sufficient transition time between
  two successive images on the same instrument, limitation of the instantaneous data
  flow through the satellite telemetry resulting from simultaneous images on different
  instruments, limitation of the on-board recording capacity for the images that are not
  directly down-linked.

The problem consists in finding an admissible subset I' of I (imperative constraints met)
which maximizes the sum of the weights of the images in I'.

1.3. *Formal Description*

As the problem looks like a *Multi-Knapsack* problem, the *ILP* framework and variants of
the *CSP* and *SAT* frameworks can be used to represent and solve it. We use here a *CSP*-like
framework, to describe it more formally:

– a variable is associated with each image;

– a weight is associated with each variable;

– a domain is associated with each variable: a subset of {1, 2, 3} for a mono-image, the
  singleton {13} for a stereo-image (1, 2, and 3 corresponding respectively to the use of
  the front, middle, and rear instrument, 13 corresponding to the use of the front and rear
  instruments);

– a set of imperative binary constraints expresses the non overlapping and minimum
  transition time constraints;

– a set of imperative binary or ternary constraints expresses the limitation of the instan-
  taneous data flow through the satellite telemetry;

– an n-ary imperative constraint, involving all the variables that are associated with images
  that must be recorded on-board, expresses the limitation of the on-board recording
  capacity;

– an assignment of the problem variables is said partial if it involves a subset of the variables;

– the weight of a partial assignment is defined as the sum of the weights of the assigned variables;

– a partial assignment is said feasible if and only if it satisfies all the imperative constraints (with the n-ary constraint restricted to the assigned variables);

– the problem consists in finding a partial feasible assignment whose weight is maximum.

2. Instances and Results

2.1. Instances

The benchmark we propose involves 20 instances. These instances have been selected from 498 instances which have been built by a *CNES* simulator of the *SPOT5* order book. They can be down-loaded from ftp://ftp.cert.fr/pub/lemaitre/LVCSP/Pbs/SPOT5.tgz.

The instances whose number is less than 1000 are limited to a single satellite revolution and do not include any recording capacity constraint. The others, whose number is greater than 1000, involve several satellite revolutions and include a recording capacity constraint.

The instances 404, 408, 412, 414, 503, 505, 507, and 509 have been created from the same instance: the instance 414, which is the largest of all the instances without any recording capacity constraint (364 variables and 9744 binary and ternary constraints). To create the instances 404, 408, and 412, some images have been randomly removed. To create the instances 503, 505, 507, and 509, some images have been removed, in order to limit the number of conflicts.

Similarly, the instances 1401, 1403, 1405, 1407, 1502, 1504, and 1506 have been created from the same instance: the instance 1407, which is the largest of all the instances with a recording capacity constraint (1057 variables, 20730 binary and ternary constraints and one n-ary constraint), the instances 1401, 1403, and 1405, by randomly removing some images and the instances 1502, 1504, and 1506, by removing some images, in order to limit the number of conflicts.

The instances 54, 29, 42, 28, and 5 result from a selection among the instances without any recording capacity constraint.

A 21st instance (the instance 8) is here for the sake of format explanation.

2.2. Data File Syntax

One file is associated with each instance. This file is the result of a preprocessing, which computes all the variables with their associated domain and all the binary and ternary imperative constraints with their explicitly defined associated relation: the set of the forbidden tuples. Only the n-ary constraint, associated with the limitation of the recording capacity, remains implicitly defined.

Using a *BNF*-like formalism (something following by an asterisk indicates zero or more, something in square brackets indicates zero or one, and curly brackets are used for grouping), the file syntax can be described as follows:

```
----------------------------------------------------------
file ::=
   variables constraints

variables ::= number-of-variables {variable}*
number-of-variables ::= number \newline
variable ::= variable-ident variable-weight domain-size
             {value-ident recorder-consumption}* \newline
variable-ident ::= number
variable-weight ::= number
domain-size ::= number
value-ident ::= number
recorder-consumption ::= number

constraints ::= explicitly-defined-constraints
                [implicitly-defined-constraints]
explicitly-defined-constraints ::= number-of-constraints
                                   {constraint}*
number-of-constraints ::= number \newline
constraint ::= arity {variable-ident}*
               {forbidden-tuple}* \newline
arity ::= number
forbidden-tuple ::= {value-ident}*
implicitly-defined-constraints ::= recording-capacity
                                   \newline
recording-capacity ::= number
----------------------------------------------------------
```

For example, the file *8.spot* represents the instance 8, a small size instance including 8 variables and 7 constraints, without any recording capacity constraint:

```
-----------------
8
0 1 3 1 0 2 0 3 0
1 1 3 1 0 2 0 3 0
2 1 3 1 0 2 0 3 0
3 1 3 1 0 2 0 3 0
4 2 1 13 0
5 2 1 13 0
6 2 1 13 0
7 2 1 13 0
7
```

```
2 1 0 3 3 2 2 1 1
2 2 0 3 3 2 2 1 1
2 3 0 3 3 2 2 1 1
2 5 4 13 13
2 5 6 13 13
2 2 1 3 3 2 2 1 1
2 3 1 3 3 2 2 1 1
-----------------
```

The first variable (line 2) has the following characteristics:

– its ident is 0;

– its weight equals 1;

– its domain size equals 3;

– its possible values are 1, 2, and 3, all of them without any recorder consumption.

The first constraint (line 11) has the following characteristics:

– its arity equals 2;

– it links the variables 1 and 0;

– the forbidden pairs of values are (3 3) (2 2) (1 1) (disequality constraint).

2.3. *Known Results*

Table 1 shows the main characteristics of the proposed instances and what is known so far about them:

– nb is the instance number;

– n is the number of variables;

– e is the number of binary and ternary constraints;

– w is the weight of the best solution found so far; an asterisk indicates that the optimality of this solution has been proved.

One can observe that, whereas complete methods are able to solve optimally all the instances without any recording capacity constraint (whose number is less than 1000), they are not able to do the same for the instances with a recording capacity constraint (whose number is greater than 1000), except for one (the instance 1502, which is the smallest). This is due to the existence of this global constraint and to the large size of the corresponding instances, which involve several satellite revolutions.

All the provenly optimal results have been obtained, either by using an *ILP* problem formalization and the *CPLEX* commercial software, or by using a *Valued CSP* formalization

Table 1. What is known so far about
the 21 instances.

nb	n	e	w
8	8	7	10*
54	67	204	70*
29	82	380	12032*
42	190	1204	108067*
28	230	4996	56053*
5	309	5312	115*
404	100	610	49*
408	200	2032	3082*
412	300	4048	16102*
414	364	9744	22120*
503	143	492	9096*
505	240	2002	13100*
507	311	5421	15137*
509	348	8276	19125*
1401	488	10476	176056
1403	665	12952	176137
1405	855	17404	176179
1407	1057	20730	176246
1502	209	203	61158*
1504	605	3583	124243
1506	940	14301	168247

[3] and a non-standard *Branch and Bound* algorithm [5]. All the other results, have been obtained by using *Tabu Search* algorithms [1], [4]. The *Constraint Programming ILOG Solver* commercial software has been experimented on smaller instances [2]. *Local Search* algorithms, other than *Tabu Search*, have not been extensively experimented.

2.4. *Some Useful Knowledge*

Because of the distribution of the possible images along each satellite revolution, some instances can be decomposed into independent sub-instances (no constraint between them), which can be solved separately.

   The ordering of the variables in each file is significant: it corresponds to the order of the images according to time along the satellite revolutions. With *Branch and Bound* methods, the best results have been obtained by using this ordering as a static variable ordering.

   This management problem has to be solved daily. In an operational context, an hour is currently considered as a maximum time to decide which images will be taken the next day and how to take them. Although this constraint cannot be considered as imperative in the context of this benchmark and although the distance between research and operational softwares can be very large, it may be important to keep this in mind: a software which would take more than one day to decide would not be very useful.

3.   What and Where to Report

The authors are ready to collect and to report any news about the proposed instances: better solutions, upper bounds, optimality proofs, time necessary to get these results. Use email for that.

   Although results concerning the instances without any recording capacity constraint are still interesting (for example, very short times to solve them optimally), the most interesting results we expect concern the instances with a recording capacity constraint, which have not been solved optimally yet (better solutions, upper bounds, optimality proofs . . .).

Acknowledgements

Notes

1. http://www.cert.fr
2. http://www.cnes.fr
3. http://www.spotimage.fr

References

1. Bensana, E., Verfaillie, G., Agnèse, J. C., Bataille, N., and Blumstein, D. (1996). Exact and Approximate Methods for the Daily Management of an Earth Observation Satellite. *Proc. of the 4th International Symposium on Space Mission Operations and Ground Data Systems (SpaceOps-96)*. Münich, Germany. ftp://ftp.cert.fr/pub/verfaillie/spaceops96.ps.

2. Lemaître, M. and Verfaillie, G. (1997). Daily management of an earth observation satellite: comparison of ILOG Solver with dedicated algorithms for Valued Constraint Satisfaction Problems. *Proc. of the Third ILOG International Users Meeting*. Paris, France. ftp://ftp.cert.fr/pub/verfaillie/ilog97.ps.

3. Schiex, T., Fargier, H., and Verfaillie, G. (1995). Valued Constraint Satisfaction Problems: Hard and Easy Problems. *Proc. of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95)*. Montréal, Canada, pp. 631–637. ftp://ftp.cert.fr/pub/verfaillie/ijcai95.ps.

4. Vasquez, M. and Hao, J. K. (1998). Recherche locale pour la planification journaliére de prises de vue d'un satellite. *Actes du 1er Congrès National sur la Recerche Opérationnelle et l'Aide à la Décision (ROAD-98)*. Paris, France.

5. Verfaillie, G., Lemaître, M., and Schiex, T. (1996). Russian Doll Search for Solving Constraint Optimization Problems. *Proc. of the 13th National Conference on Artificial Intelligence (AAAI-96)*. Portland, OR, USA, pp. 181–187. ftp://ftp.cert.fr/pub/verfaillie/rds-aaai96.ps.