Sharing the use of Earth observation satellites

Michel Lemaître and **Gérard Verfaillie**
ONERA, Toulouse, France
{Michel.Lemaitre, Gerard.Verfaillie}@onera.fr

Hélène Fargier and **Jérôme Lang**
IRIT, Toulouse, France
{fargier, lang}@irit.fr

Nicolas Bataille and **Jean-Michel Lachiver**
CNES, Toulouse, France
{Nicolas.Bataille, Jean-Michel.Lachiver}@cnes.fr

Abstract

Because they are more and more costly, space projects are more and more frequently funded by several entities, which share the use of the space system. This sharing must, on the one hand, satisfy as best as possible each entity and, on the other hand, meet as best as possible predefined sharing rules.

In this paper we report studies which have been carried out in the specific context of shared Earth observation satellites. We describe informally the sharing requirements and the sharing principles we adopt. Then, we describe a sharing protocol, based on these principles and on three levels of utility.

Background about agile optical Earth observation satellites

Earth observation satellites are placed on low altitude, circular, quasi polar, heliosynchronous, and phased orbits. They are equipped with optical, radar, or infra-red observation instruments. Degrees of freedom for image acquisition can be provided in various ways. In the special case we consider (that of agile optical Earth observation satellites), they are provided by the satellite attitude control system which is able to move the whole satellite along the roll, pitch, and yaw axes. That allows the instrument to acquire any strip of a fixed width, but of any direction and any length, at the Earth surface on each side of the track of the satellite on this surface (within some limits; see Figure 1). Acquired images are either immediately downloaded, or recorded on board and downloaded when the satellite is within the visibility window of a ground station. Image acquisitions are thus time, energy, and memory consuming. Transitions between image acquisitions are time and energy consuming too.

Observation requests may be submitted at any time by users to a mission control center. With each request are associated five main parameters: (i) a geographical area, which can be either a target (a small circular area, which can be acquired in one pass) or a polygon (a large polygonal area, which must be split into parallel strips and acquired in several passes, possibly from several satellite revolutions), (ii) a validity period, outside of which its acquisition has no utility, (iii) a set of acquisition angular constraints (minimum and maximum roll and pitch acquisition angles), (iv) a type, which can be either mono or stereo, and (v) either a weight or a priority, which expresses its importance from the user's point of view.

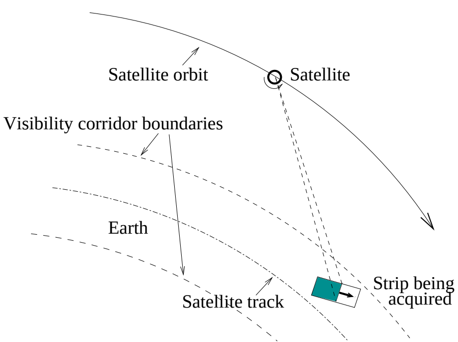

Figure 1: Acquisition of a strip.

Mission management reasons on a specified horizon, which may be either one day or a satellite revolution around the Earth (in fact only the illuminated half-revolution from which image acquisition is possible with an optical instrument). From the set of strips that can be acquired on that horizon, it selects a subset the associated gain of which is as highest as possible and then orders them along time. In other words, it builds a sequence of strip acquisitions which is feasible and of as highest as possible utility (see Figure 2

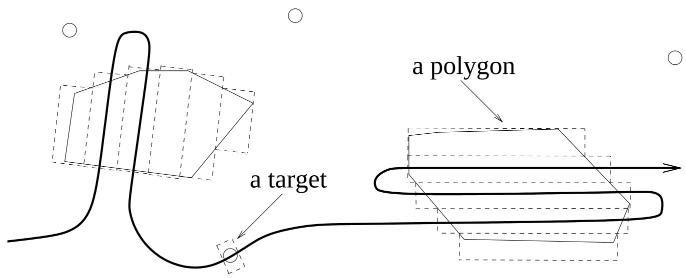

Figure 2: Example of a sequence of strip acquisitions.

for an example of a sequence of strip acquisitions).

Although many mission management modes are possible (see for example (Verfaillie & Bornschlegl 2000)), this selection and scheduling task is usually performed off-line before execution (either the day i for the day $i+1$, or during the non-illuminated half-revolution for the following illuminated half-revolution). It is also performed on the ground in a mission control center, and semi-automatically under the supervision of human operators.

For more details, see either (Verfaillie & Lemaître 2001) or (Lemaître *et al.* 2002).

Sharing requirements

In the previous section, we made the assumption that there was only one user of the satellite, in fact only one entity (the mission control center) which is assumed to act fairly on behalf of a set of end users. Such an assumption is at the basis of many studies which have been performed about the selection and scheduling of Earth observation activities (see (Bensana *et al.* 1996), (Bensana, Lemaître, & Verfaillie 1999), (Gabrel *et al.* 1997), (Vasquez & Hao 2001), (Vasquez & Hao 2002), (Bornschlegl, David, & Schindler 1998), (Pemberton 2000), (Frank *et al.* 2001), (Dungan *et al.* 2001), (Hall & Magazine 1994), and (Wolfe & Sorensen 2000)).

But more and more space projects are funded by several entities, from the same country or from several countries, for example by several space agencies or by civil and military organizations. When funding the space system, these entities want to have the guarantee that they will be able to use it proportionally to their financial investment.

The problem is no more to maximize only the utility of one entity. It is to maximize the utility of each entity involved in the project and to meet as best as possible sharing rules on which all the entities came to an agreement before using the system.

In the problem we consider (the sharing of an agile optical Earth observation satellite between several entities), the mission management is decomposed into three phases which are activated in sequence: management of the high priority requests, management of the medium priority requests, and management of the remaining ones.

In the first phase we only consider in this paper, the request emitted by an entity is assumed to be a list of unit requests[1] (*i.e.* the associated geographical area of which is a square of a specified dimension) which is ordered according to a decreasing *priority*. This order is interpreted as follows: the associated entity prefers the satisfaction of a unit request of a given priority to the satisfaction of any number of requests of lower priority. For example, if the ordered list of an entity is (r_1, r_2, r_3), this entity prefers that request r_1 be satisfied rather than requests r_2 and r_3 together. Moreover, the number of satisfied unit requests is for each entity limited to a specified number, which depends on its investment in the project and which we refer to as its *rights*.

Sharing principles

In this section and in the following, we call *selection* a subset of the candidate unit requests, which can be scheduled taking into account satellite acquisition constraints.

Arbitration versus negotiation

Sharing procedures (or protocols) can be divided into two main classes:

- the decentralized *game* or *negotiation* procedures, where all the entities together agree with game or negotiation rules, and then act for their own interest by respecting the common rules (see for example (Rosenschein & Zlotkin 1994; Brams & Taylor 1996));
- the centralized *arbitration* procedures, where one central entity, which is assumed to be fair and to act according to principles that have been accepted by all the entities, decides about the sharing (see (Moulin 1988; Young 1994)).

We chose to explore centralized arbitration procedures for four reasons: (i) the entities may want to maintain confidentiality about their own requests and we guess this confidentiality can be guaranteed by a centralized arbitration procedure (the arbitration agent says nothing to an entity about the requests of the others) better than by a decentralized negotiation procedure, which imposes to exchange information; (ii) the number of unit requests which must be dealt with may be high (dozens and possibly hundreds of requests) and a negotiation seems to be in that case very difficult to manage; (iii) the time which is available for the mission management task may be very short and we guess that a decentralized negotiation procedure would be too time consuming; and (iv) if the optimization criteria can be clearly defined and if the associated optimization algorithms are efficient enough, a centralized arbitration procedure is at least as efficient as and often more efficient than a decentralized negotiation procedure.

Three utility levels

A centralized arbitration procedure shall reason about three levels of utility:

[1] Although it is possible to extend the mechanisms that are presented in this paper to non unit requests (strips of any length, polygons, stereo requests), the mechanisms that allow this are out of the scope of this paper.

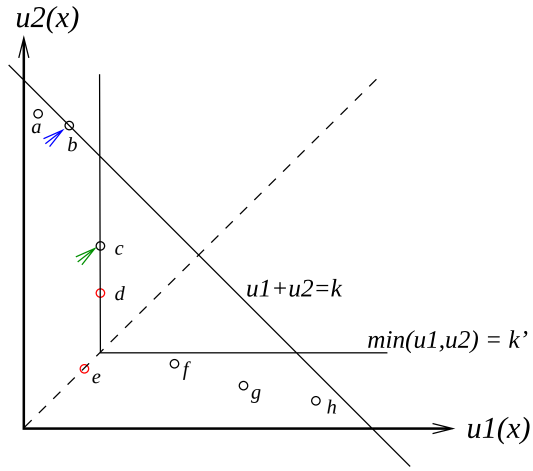

Figure 3: Representation of the solutions of a problem including two utility components u_1 and u_2. The closer a solution is to the diagonal dotted line, the fairer it is.

- the utility of a selection for an entity and a priority level; this utility depends basically on the selection or not of the request this entity has submitted at this priority level;
- the utility of a selection for an entity; this utility results from an aggregation (to be defined) of the utilities of this selection for this entity and all priority levels, *i.e* basically on the selection or not of the requests this entity has submitted;
- finally, the social utility of a selection; this utility results from an aggregation (to be defined) of the utilities of this selection for all the entities.

Efficiency versus equity

The problems we have to deal with are in fact usual multicriteria optimization problems: How to build and how to compare aggregated utilities? How to compare the utilities of a selection for two entities? How to compare the social utilities of two selections?

The first principle with which everyone agrees in multicriteria decision making is that of *efficiency* (or *Pareto* optimality; see for example (Keeney & Raiffa 1976; Ehrgott 2000)): (i) a solution s dominates another solution s' if and only if s is strictly better than s' according to at least one of its components and not worse than s' according to the other components (in Figure 3, solution e is dominated by solution d, which is itself dominated by solution c); (ii) a solution is said efficient (or Pareto optimal) if and only if it is dominated by no other solution (in Figure 3, solutions a, b, c, f, g, and h are all efficient).

We are obviously looking for efficient solutions. The problem is that not all the efficient solutions are fair. Some of them can even be very unfair (see for example Figure 3 where solutions a and h are efficient, but very unfair).

Utilitarism versus Egalitarism

Methods for aggregating individual comparable utilities can be divided into two main families: (i) the *utilitarist* ones where the aggregation operator is the *sum* and (ii) the *egalitarist* ones where it is the *min*. In the first case, one wants to maximize the sum of the individual utilities. In the second case, one wants to maximize the minimum on all the individual utilities. On the example of Figure 3, solution b would be chosen in the first case; either solution c or solution d in the second case.

All the optimal solutions according to the *sum* aggregation operator are efficient, but some of them (possibly all of them) may be very unfair (see for example solution b in Figure 3). Some mechanisms can however counterbalance this phenomenon:

- it is possible to select, among the optimal solutions according to the utilitarist criterion, those that are the fairest ones;
- it is also possible to take advantage of the repetitive nature of the sharing process (performed either each day or for each satellite revolution) and, either to rely on the law of large numbers to obtain fairness, or to compensate any unfair sharing by changes in the data of the following ones (in order to favour entities that have been at a disadvantage during the previous sharings).

On the other hand, the *min* aggregation operator directly favours fairness by maximizing the utility of the less favoured entity. But it can select non efficient solutions, like solution d in the example of Figure 3.

Fortunately, it can be refined using the *leximin* operator, which orders according to an increasing utility each of the utility vectors to be compared and then compares the ordered vectors using a lexicographic order (the order that is used in a dictionary).

For example, if we have two utility vectors $u = (4, 1, 2)$ and $v = (1, 3, 1)$, they are first ordered according to an increasing utility of their components, resulting in the vectors $u^* = (1, 2, 4)$ and $v^* = (1, 1, 3)$. The lexicographic comparison between u^* and v^* shows that u must be preferred to v (equality according to the first least favoured component, advantage for u^* according to the second component). On the example of Figure 3, solution c would be selected using this operator.

The main advantage of the *leximin* operator is that all the optimal solutions according to this operator are both optimal according to the *min* operator and efficient. This is why we chose this approach (an egalitarist approach with a *leximin* operator) for dealing with the sharing of an agile optical Earth observation satellite.

Note that we already explored an utilitarist approach in (Lemaître, Verfaillie, & Bataille 1999) and that there exist families of operators, such as the *generalized mean* operators (see (Moulin 1988), chapter 2) or the more recent *ordered weighted averaging* operators (see (Yager 1988)), which offer a *continuum* between utilitarism and egalitarism and could be profitably explored too.

A sharing protocol

The sharing protocol we designed can be described using the three levels of utility, introduced in section 1. Let n be the number of entities. Let m be the maximum number of requests each entity can submit. For each entity i, let m_i be the number of requests that entity i has submitted ($m_i \leq m$). We assume that these requests are numbered from 1 to m_i according to a decreasing priority. For each entity i too, let d_i be its rights (*i.e* the maximum number of its requests that can be selected). Let x be any selection.

Utility of a selection for an entity and a priority level

Basically, the utility of a selection x for an entity i and a priority level p (noted $u_{ip}(x)$) is equal to 1 if the request that entity i has submitted at priority level p has been selected in x (noted $s_{ip}(x) = 1$) and 0 if not. But to take into account the fact that the rights are not the same for all the entities and that these entities can submit as many requests as they want within the limit of m, $u_{ip}(x)$ is refined as follows: it is equal to 1 if and only if one of the following three conditions holds: (i) a request has been submitted by entity i at priority level p ($p \leq m_i$) and it has been selected ($s_{ip}(x) = 1$); (ii) the rights of entity i have been exhausted at priority level p ($\sum_{p'<p} s_{ip'}(x) = d_i$); (iii) no request has been submitted by entity i at priority level p ($p > m_i$).

Utility of a selection for an entity

The utility of a selection x for an entity i (noted $u_i(x)$) is simply the vector of size m of the utilities of selection x for entity i and all the priority levels.

For example, lets assume three entities ($n = 3$), the rights of which are all equal ($d_1 = d_2 = d_3 = 2$) and which can submit at most three requests each ($m = 3$). Lets assume that entities 1 and 2 have submitted three requests and that entity 3 has submitted only two requests ($m_1 = m_2 = 3$, $m_3 = 2$). Let us assume that, in selection x, the first and the third requests from entity 1 have been selected ($s_{11}(x) = s_{13}(x) = 1$, $s_{12}(x) = 0$), that the first two requests of entity 2 have been selected ($s_{21}(x) = s_{22}(x) = 1$, $s_{23} = 0$), and that only the first request of entity 3 has been selected ($s_{31}(x) = 1$, $s_{32}(x) = 0$). The utility vector of this selection for entity 1 is $u_1(x) = (1,0,1)$ (the first and the third request have been selected, but the second one not). For entity 2 it is $u_2(x) = (1,1,1)$ (the first two requests have been selected, the third one not, but the rights of this entity have been exhausted by the first two selections). For entity 3 it is $u_3(x) = (1,0,1)$ (the first request has been selected, the second one not, but no third request has been submitted by this entity).

These vectors can be compared, using a lexicographic order. In our example, the utilities of this selection for entities 1 and 3 are equal and both worse than the one of entity 2.

Social utility of a selection

The social utility of a selection x (noted $u(x)$) is simply the vector of size n of the utilities of selection x for all the entities: a vector of vectors. On the example of the previous section, it is $u(x) = ((1,0,1),(1,1,1),(1,0,1))$.

To be compared using the *leximin* operator, these vectors must be ordered according to the order on utilities of a selection for an entity (defined in the previous section), starting from the lowest utilities (the least favoured entities first). For example, once ordered, the vector $u(x)$ becomes $u^*(x) = ((1,0,1),(1,0,1),(1,1,1))$.

Let us assume that we have to compare two candidate selections x and x' the associated utility vectors of which are respectively $u(x) = ((1,0,1),\ (1,1,1),(1,0,1))$ and $u(x') = ((1,1,0),(1,0,1),(1,1,0))$. Let us observe that, with the first selection, the first and the third entities are the least favoured, whereas, with the second one, it is the second entity.

Both vectors are ordered, resulting in $u^*(x) = ((1,0,1),(1,0,1),(1,1,1))$ and $u^*(x') = ((1,0,1),(1,1,0),(1,1,0))$. By comparing $u^*(x)$ and $u^*(x')$, using a lexicographic order, we conclude that x' must be preferred to x (equality on the first least favoured entity, but advantage for x' on the second least favoured entity: $(1,0,1)$ for x and $(1,1,0)$ for x'.

Maximizing the social utility

The objective of the mission control center can thus be defined as searching for a feasible selection the social utility of which is maximum or as high as possible.

Conclusion and Perspectives

We have defined a basic protocol which enforces a fair sharing between self-interested entities of an Earth observation satellite. Its main advantages are (i) that it is based on a clear and consistent egalitarist approach, (ii) that it guarantees the quality of each sharing, and (iii) that no compensation mechanism between successive sharings is *a priori* necessary.

It has been already extended to the other management phases (management of the medium and low priority requests) for which the sharing requirements are not exactly the same. It is now necessary (i) to adapt it to all the specific characteristics of the management of agile optical Earth observation satellites and besides of constellations of such satellites, (ii) to define and implement efficient constrained optimization algorithms based on the *leximin* aggregation operator, and (iii) to experiment them on realistic scenarii and to analyze their results in terms of sharing and efficiency.

On the other hand, it should be noted that this study and its main results can be profitably reused to deal with the problem of the sharing of any resource between self-interested entities, in or out of the space domain (space laboratory, interplanetary space probe, either space or ground telescope, telecommunication network, computing resource ...).

However, an important aspect of the sharing problem remains incompletely and imperfectly solved: Can these sharing protocols resist against manipulations from some entities? Can they consequently discourage entities from trying to manipulate them?

We started trying to answer these questions, by defining what a manipulation is. A *manipulation* can be defined as a modification by an entity of its requests (going from genuine to non genuine ones) with the aim of increasing its own utility. That may be an adding or a removal of requests, as well as a change in their respective priorities. It is then necessary to be more precise about the context in which a manipulation may occur: precise or imprecise knowledge of all the mechanisms of the selection algorithm; complete, partial, probabilistic, or null knowledge of the requests that will be submitted by the other entities. It is also necessary to define what a profitable manipulation is: absolutely certain gain, absolutely certain absence of loss, positive expected gain. It is finally necessary to establish whether the proposed protocols can be manipulated or not, according to the possible contexts and the chosen definitions. Maybe the subject of other communications . . .

References

Bensana, E.; Verfaillie, G.; Agnèse, J.; Bataille, N.; and Blumstein, D. 1996. Exact and Approximate Methods for the Daily Management of an Earth Observation Satellite. In *Proc. of the 4th International Symposium on Space Mission Operations and Ground Data Systems (SpaceOps-96)*.

Bensana, E.; Lemaître, M.; and Verfaillie, G. 1999. Earth Observation Satellite Management. *Constraints : An International Journal* 4(3):293–299.

Bornschlegl, E.; David, P.; and Schindler, H. 1998. On-Board Software and Operation for Proba Small Satellite Autonomy. In *Proc. of the International Conference on Data Systems in Aerospace (DASIA-98)*.

Brams, S., and Taylor, A. 1996. *Fair Division: from cake-cutting to dispute resolution*. Cambridge University Press.

Dungan, J.; Frank, J.; Jónsson, A.; Morris, R.; and Smith, D. 2001. Advances in Planning and Scheduling of Remote Sensing Instruments for Fleets of Earth Orbiting Satellites. In *Proc. of the Earth Science Technology Conference (ESTC-02)*.

Ehrgott, M. 2000. *Multicriteria Optimization*. Number 491 in Lecture Notes in Economics and Mathematical Systems. Springer.

Frank, J.; Jónsson, A.; Morris, R.; and Smith, D. 2001. Planning and Scheduling for Fleets of Earth Observing Satellites. In *Proc. of the 6th International Symposium on Artificial Intelligence, Robotics, and Automation for Space (i-SAIRAS-01)*.

Gabrel, V.; Moulet, A.; Murat, C.; and Paschos, V. 1997. A New Single Model and Derived Algorithms for the Satellite Shot Planning Problem Using Graph Theory Concepts. *Annals of Operations Research* 69:115–134.

Hall, N., and Magazine, M. 1994. Maximizing the Value of a Space Mission. *European Journal of Operational Research* 78:224–241.

Keeney, R., and Raiffa, H. 1976. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. John Wiley and Sons.

Lemaître, M.; Verfaillie, G.; Jouhaud, F.; Lachiver, J.-M.; and Bataille, N. 2002. Selecting and Scheduling Observations of Agile Satellites. *to appear in Aerospace Science and Technology*.

Lemaître, M.; Verfaillie, G.; and Bataille, N. 1999. Exploiting a Common Property Resource under a Fairness Constraint: a Case Study. In *Proc. of the 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, 206–211.

Moulin, H. 1988. *Axioms of Cooperative Decision Making*. Cambridge University Press.

Pemberton, J. 2000. Towards scheduling over-constrained remote-sensing satellites. In *Proc. of the Second NASA International Workshop on Planning and Scheduling for Space*, 84–89.

Rosenschein, J., and Zlotkin, G. 1994. *Rules of Encounter : Designing Conventions for Automated Negotiation among Computers*. M.I.T. Press.

Vasquez, M., and Hao, J. 2001. A Logic-constrained Knapsack Formulation and a Tabu Algorithm for the Daily Photograph Scheduling of an Earth Observation Satellite. *Journal of Computational Optimization and Applications* 20(2):137–157.

Vasquez, M., and Hao, J. 2002. Upper Bounds for the SPOT5 Daily Photograph Scheduling Problem. *To appear in the Journal of Combinatorial Optimization*.

Verfaillie, G., and Bornschlegl, E. 2000. Designing and Evaluating an On-line On-board Autonomous Earth Observation Satellite Scheduling System. In *Proc. of the Second NASA International Workshop on Planning and Scheduling for Space*, 122–127.

Verfaillie, G., and Lemaître, M. 2001. Selecting and Scheduling Observations for Agile Satellites: Some Lessons from the Constraint Reasoning Community Point of View. In *Proc. of the 7th International Conference on Principles and Practice of Constraint Programming (CP-01)*, 670–684.

Wolfe, W., and Sorensen, S. 2000. Three Scheduling Algorithms applied to the Earth Observing Systems Domain. *Management Science* 46(1):148–168.

Yager, R. 1988. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision Making. *IEEE Transactions on Systems, Man, and Cybernetics* 18:183–190.

Young, H. 1994. *Equity in Theory and Practice*. Princeton University Press.