C. Pralet, S. Roussel
(ONERA)

J. Jaubert
(CNES)

J. Queyrel
(ONERA)

E-mail: cedric.pralet@onera.fr

DOI: 10.12762/2020.AL15-05

Planning for Space Telescopes: Survey, Case Studies, and Lessons Learned

In this article, we present planning and scheduling techniques that we developed for optimizing the operations of space telescopes. The latter are satellites whose mission is to observe celestial objects such as planets, exoplanets, stars, or galaxies. After a survey of some existing mission planning tools, we present three case studies that we tackled using a constraint-based optimization and operations research approach, with for each case study the lessons that we learned. Based on these, we propose future work directions related to the development of generic mission planning tools, to the management of uncertain events, and to the definition of a global mission planning concept for several telescopes.

Introduction

Space telescopes (or space observatories) are satellites whose mission is to observe celestial objects such as planets, exoplanets, stars, or galaxies. As expressed in the European Space Agency (ESA) Cosmic Vision program,[1] they are used by the scientists to answer questions concerning the formation of planets, the emergence of life, the way in which the Solar system works, and to understand the physical laws of the Universe and its origins.

To achieve this goal, space telescopes usually embed several instruments such as gamma-ray imagers, gamma-ray detectors, X-ray imagers, optical cameras, photometers, photodetectors, infrared cameras, infrared spectrometers, ultraviolet sensors, etc. All instruments are body-mounted on the satellite for acquisition quality reasons, and to observe a specific target the whole satellite must be pointed at a specific direction, with a need for a high angular precision and for a very stable pointing.

For space telescopes, Artificial Intelligence (AI) can come into play to analyze the set of data produced by the instruments. It also comes into play for the construction of the telescope activity plans. The latter include tasks such as *observation tasks* requested by the scientists, *calibration tasks* used for setting up the instruments, *maneuvers* used for pointing the telescope toward particular directions, and *communication tasks* used to receive telecommands and to send observation data to ground stations. In this context, the plans constructed must be valid according to various constraints related to physical limitations or user requirements. They must also optimize the exploitation of the telescope over a given time period, for various objective functions related to the priority of some targets, the temporal dispersion of groups of observations, or the amount of resources consumed. One reason that motivates the use of automated AI planning and scheduling is that there are usually a lot of candidate tasks (thousands or tens of thousands), and it is not straightforward even for a system expert to manually define valid and efficient activity plans.

Globally, AI planning and scheduling is used both for the *long-term planning* phase, where the activity plans are constructed over long time periods (e.g., one year), and for the *short-term planning* phase, where the plans must be refined and where last-minute observation requests may be received. The latter include Targets of Opportunity (ToOs), corresponding to high-priority targets for which events are detected by other ground or space telescopes. The short-term plans sent to the telescope can also be updated directly on-board if highly relevant events such as Gamma-Ray Bursts (GRBs) are detected by the embedded instruments. It is then useful to be able to reschedule the parts of the observation plan canceled due to the arrival of ToOs and GRBs. It can also be noted that for the short-term planning phase all activities of the telescope are taken into account, but for the long-term planning phase some of them are sometimes not explicitly modeled. This can occur for maneuvers, when their duration is not significant compared to the duration of observations, or for operations that are carried out during specific parts of the orbit where no observation is possible.

In this article, we first give an overview of some mission planning systems available in the literature to manage space telescopes (Section "A survey of some mission planning systems"). After that, we provide feedback on our experience in the field, based on three

[1] https://en.wikipedia.org/wiki/Cosmic_Vision

case studies that we tackled. On the latter point, Section "Planning for INTEGRAL" deals with long-term mission planning for the active ESA INTErnational Gamma-Ray Astrophysics Laboratory (INTEGRAL), Section "Planning for SVOM" describes mission planning for the future French-Chinese (CNES-CNSA) Space Variable Objects Monitor (SVOM), and Section "Planning for ARIEL" presents long-term mission planning for the future ESA Atmospheric Remote-Sensing Infrared Exoplanet Large-survey (ARIEL) mission. We describe a constrained-based scheduling and operations research approach used for these missions, as well as the results obtained. The descriptions provided for these three missions correspond to a global view of the full descriptions available in dedicated papers [35, 34, 38]. Last, Section "Conclusion and future work directions" provides future work directions concerning the development of generic constraint-based optimization tools, the production of robust plans that better anticipate the arrival of ToOs and GRBs, and the definition of a planning system for managing several telescopes.

A survey of some mission planning systems

Nowadays, a little over 100 space telescopes are referenced.[2] This section gives a synthetic view of the mission planning systems developed for some of them, based on the restricted list given in Table 1 composed of the Hubble Space Telescope (HST), X-ray Multi-Mirror (XMM)-Newton, INTEGRAL, Spitzer, Swift, Herschel, the James Webb Space Telescope (JWST), SVOM, the Exoplanet Characterization Observations (EChO) telescope, ARIEL, and the Advanced Telescope for High ENergy Astrophysics (ATHENA).

Telescope	Space agency	Orbit	Launch date	Termination	References
HST	NASA	Low-Earth Orbit (590km)	1990	-	21, 20, 22, 26, 13, 14
XMM-Newton	ESA	highly elliptical orbit around the Earth	1999	-	5
INTEGRAL	ESA	highly elliptical orbit around the Earth	2002	-	35, 24
Spitzer	NASA	heliocentric orbit (372-day period)	2003	-	29
Swift	NASA	Low-Earth Orbit (600km)	2004	-	31
Herschel	ESA	around the L2 Lagrange point	2009	2013	16, 4, 7
JWST	NASA	around the L2 Lagrange point	2021 (planned)	-	37, 15
SVOM	CNES-CNSA	Low-Earth Orbit (625km)	2021 (planned)	-	34, 19
EChO	ESA	around the L2 Lagrange point	canceled	canceled	32, 30, 12
ARIEL	ESA	around the L2 Lagrange point	2028 (planned)	-	38
ATHENA	ESA-JAXA	around the L2 Lagrange point	2031 (planned)	-	18

Table 1 - Analyzed space telescopes and references to their mission planning systems

[2] https://en.wikipedia.org/wiki/List_of_space_telescopes

A previous generic analysis [8] is based on a list of telescopes different from the one considered here. Globally, similar conclusions are derived as regards the mission needs, but we try to give a few more details on the problem components with regard to standard constraint-based optimization concepts. Note also that, for some telescopes such as Planck [11], there is no need for AI planning and scheduling since the successive pointings used by the satellite are given by a predefined pointing law.

Problem features

High-level observation requests

Many telescope missions involve high-level observation requests that cover multiple elementary observation tasks. These tasks can be linked by various kinds of constraints. For instance, some elementary tasks might have to be performed sufficiently close to each other or, more generally, the elementary tasks can be linked by minimum and maximum distance constraints. The latter are temporal constraints of the form $y - x \in [a, b]$, where x and y are two variables representing the start or end times of two tasks, and where a and b are two constants. Moreover, it might be forbidden to interleave the elementary observations of a given request with other elementary observations.

The elementary observations can also be linked by various kinds of preference functions. For instance, a preference for grouping the elementary tasks as much as possible, or a preference for performing them as periodically as possible, might exist.

Task selection

Almost all telescope mission planning problems are over-constrained, meaning that it is not possible to perform all of the candidate tasks, one exception being the long term mission planning for EChO, where the goal is to perform all tasks within a minimum amount of time [32, 30, 12]. Several constraints can be imposed on the selection of tasks, including (1) constraints forcing some mandatory tasks to be performed, (2) constraints linking the performance of several tasks, such as constraints imposing that a specific calibration task should be selected only if one of the observation task requiring this calibration is selected, or (3) constraints specifying that a high-level observation request is selected if and only if a sufficient number of its elementary observation tasks is scheduled.

In many missions, the observation tasks to be performed are fully specified by the users, but in some cases there is a freedom on some parameters. This occurs for HST and JWST, where an observation orientation must be chosen within an orientation range specified by the users, and where relative orientation constraints can impose that the angular distance between the orientations chosen for two observation o_1, o_2 must be within a given range.

Choice of realization windows

Usually, the target associated with an observation task is visible only during certain *visibility windows*, computed by taking into account elements like the positions of the Earth, the Sun, the Moon, and other planets, or the user requirements concerning the earliest and latest times at which observation data must be collected. One issue is then to choose, for each selected observation task, the visibility window(s) within which it is carried out, as in works on scheduling with multiple time windows. The time granularity used for representing the visibility windows can differ from one mission to another: windows expressed in days, revolutions, hours, etc.

Disjunctive or cumulative telescope resource

For the short-term planning phase, the telescope is viewed as a *disjunctive resource*, that is, as a resource that cannot carry out two observation tasks in parallel. Indeed, given that all instruments are body-mounted on the satellite, two observations associated with distinct pointing directions cannot be performed simultaneously.

For the long-term planning phase, the telescope is also modeled as a disjunctive resource most of the time. In some missions, however, it is viewed as a *cumulative resource* that can carry out activities in parallel up to a certain limit. This is the case for HST and JWST, where the long-term planner computes, for each selected observation task, a *soft realization window* that is larger than the actual window required to perform the observation. Then, soft realization windows can overlap up to a certain limit, which leads to a model based on a cumulative resource instead of a disjunctive one. The rationale is that instead of choosing fixed windows in long-term plans, some level of temporal flexibility is kept in prevision of last-minute urgent observation requests.

Knapsack and cardinality constraints

For most of the missions a long-term plan is built, but it is necessary to take into account the fact that the occurrence of events such as GRBs or ToOs leads to new mandatory observation tasks, which induce a temporary interruption of the nominal mission plan. To build long-term plans that are robust to such random events, the usage rate of some time periods is sometimes limited. For instance, for INTEGRAL, a maximum usage rate is imposed over every revolution [35], and for XMM-Newton a maximum usage rate is imposed over groups of 4-5 successive revolutions [5]. In terms of optimization problems, the revolutions or groups of revolutions are kinds of knapsacks whose capacity corresponds to the maximum usage duration, and the observations carried out are items whose size corresponds to their duration. In some cases, cardinality constraints must be satisfied, such as when there is a limit on the number of observations per revolution, to indirectly minimize maneuvers.

State constraints

As mentioned previously, all telescopes considered embed several instruments. The latter can sometimes operate in different states, and setup operations are needed to reach a given state. For instance, in the case of Herschel [4], cool down operations are required to set up the instrument(s) before an observation, and these operations are quite long (several hours) and consume liquid Helium. In this case, to avoid spending too much time and energy for state changes, a (unique) state of the instrument(s) is chosen for each day of operation. Moreover, there exists a minimum number of successive days during which a given state must be maintained. From a scheduling point of view, if setup operations are explicitly modeled, the problem reached shares similarities with Job Shop Scheduling Problems with Sequence-Dependent Setup Times [1]. Another example where cool down operations are explicitly considered is the ATHENA mission [18], where one issue is to plan cryocooler regeneration cycles that allow one of the instruments to be cooled down and observations to be performed for a given duration. One issue is then to define cooling strategies that allow the response time to ToOs to be optimized, in addition to being efficient for carrying out the long-term program.

Resources with consumption and production

In terms of mission constraints, the more complex specifications are probably those of the JWST. Basically, due to its large surface, the JWST is subjected to a significant solar radiation pressure. The perturbation induced on the kinetic momentum is countered using reaction wheels available on board. However, the accumulation of the perturbations can lead to a saturation of the reaction wheels, when their maximum speed is reached. It is then necessary to desaturate these wheels by using some of the fuel available (so-called *momentum dumping* operations). It is also possible to "produce" momentum by performing observations in a direction that allows the wheels to be slowed down. As a result, the mission planner for the JWST must handle a *momentum resource* [37] that is subject to both momentum consumptions and momentum productions.

Optimization criteria

One common point between the mission planning problems associated with the telescopes considered is that there is never a single optimization criterion. Examples of objective functions encountered are:

- maximization of the number of observation requests that are carried out, while potentially taking into account the priority associated with each request;
- maximization of the total usage duration of the telescope for scientific purposes (useful mission time);
- minimization of the cumulated slew required to carry out maneuvers between the successive tasks of the plan;
- minimization of the use of consumable resources, like the fuel, to increase the long-term lifetime of the mission;
- maximization of preference degrees over the realization times of some observations: preference for earliest performance times, preference for grouping as much as possible the elementary observations associated with a single request, preference over the regularity of the performance of periodic observations, user-preference for performing observations during specific parts of the orbits, etc.;
- fair sharing of the satellite among the mission contributors or among the pointing directions;
- minimization of the degree of violation of some *soft* constraints, that should ideally be satisfied but for which partial satisfaction is allowed if needed;
- maximization of the robustness of the plans produced, with regard to the arrival of GRBs and ToOs; one goal here is to provide the users with a good estimation of the time at which the data they requested will be available.

Planning techniques

The mission planning systems developed for the space telescopes analyzed all use incomplete search algorithms, which are able to quickly find good quality solutions but that offer no guarantee with regard to the optimality of the solution produced. The reasons for this are twofold. First, the size of the instances that must be solved precludes the use of systematic techniques guaranteeing that the whole search space is explored. Second, the notion of optimal solution is often hard to define due to the presence of multiple objective functions. As detailed below, two kinds of mission planning algorithms are used in practice, namely (1) greedy search, and (2) local search coupled with metaheuristics (simulated annealing, tabu search, genetic algorithms, etc.).

Greedy search

For space telescope planning, a greedy search consists in (1) choosing at each step an elementary observation task or an observation request, (2) inserting exposure time (*i.e.,* observation time) within the current plan so as to fulfill the observation task or the request selected, (3) iterating this process so as to obtain a plan that is filled as much as possible, given that an activity inserted into the plan at some step can never be removed.

The main parameters of such a greedy search scheme are the *selection heuristic*, used for selecting an observation at each step, and the *insertion heuristic*, used for choosing an insertion position in the current plan. As an illustration, for the Swift telescope, the greedy search scheme implemented in the TAKO planning tool [31] is based on a *static* selection heuristic that successively selects tasks based on fixed priorities, and on an insertion heuristic that inserts each task at the earliest position in which there is a sufficient idle period. For the SVOM mission [34] detailed later in this article, the selection heuristic is dynamic (depending on the content of the current plan), and the insertion heuristic also inserts tasks at the earliest position in which there is a sufficiently idle period, but in this case with the possibility of moving back the tasks of the current plan (manipulation of temporally flexible solutions). In the HST [22] or XMM-Newton [5], the selection strategy considers the observations that are more constrained first, since these observations become increasingly difficult to insert as the search progresses, especially the observations composed of several elementary observations that must be carried out following a specific pattern. It is also possible to consider first the observations that are the least constrained, the intuition being that these observations will be easier to rearrange in case GRBs or ToOs occur.

Greedy decision rules are also used for short-term replanning following the arrival of high-priority requests. In this case, the conflicts between the new mandatory observations and the observations of the current plan are analyzed and resolved based on fast priority-based decision rules.

Local search and metaheuristics

In another direction, several mission planners developed for space telescopes use local search and global optimization methods (or metaheuristics) that allow the exploration of the search space to be diversified and local optima to be escaped from.

On the local search side, the mission planners for the HST and JWST use the *minconflicts* algorithm [28], which starts from an inconsistent plan and reduces step-by-step the number of conflicts between the observations of the plan (iterative conflict resolution approach [22]). For other telescopes like INTEGRAL or XMM-Newton, the local search techniques proposed handle only consistent plans at each step of the search, where tasks can be successively added or removed.

Concerning metaheuristics, various global optimization methods were tested for space telescopes: multi-objective evolutionary algorithms for the JWST [15], stochastic hill-climbing or tabu search for INTEGRAL [35, 24], iterated local search for SVOM [34], and simulated annealing, or restart techniques that allow to diversify the exploration of the search space thanks to the stochastic nature of some decision rules. Some works also use optimization strategies that first build a plan based on a coarse-grain model and then refine this plan based on a detailed model [24].

Generic techniques

For several missions, the low-level constraints of the problem are handled by core planning and scheduling frameworks. One example is the APSI framework [6] initially developed by ISTC-CNR (Rome, Italy) for ESA, and which was used for INTEGRAL [35] and XMM-Newton [5] to determine whether scheduling a small set of observation tasks within a given revolution is feasible. It was also tested for planning the mission of Herschel [7].

A second example is the constraint-based Spike tool [22, 26], developed by the US *Space Telescope Science Institute* (STScI). This tool is used for planning the activities of many ground and space telescopes, including the HST and JWST. Basically, Spike helps to prune the task insertion positions that would lead to dead-ends given the set of temporal constraints of the problem.

A last example is the InCELL library [36, 33], which was used for two of the three telescope mission planning problems detailed in the next sections. Basically, the scheduling layer of InCELL handles an ordering over the tasks of the plan and maintains the earliest and latest start times of these tasks as a consequence of the current ordering. InCELL also allows various optimization criteria to be modeled, whereas for instance on the HST specific work was necessary to formalize the objectives truly optimized by the Spike-based planner after several years of operations [14]. As for common points, both Spike and InCELL are constraint-based, and both can manage at some step temporally inconsistent plans. Compared to APSI, InCELL is dedicated to the implementation of local searches and metaheuristics.

Interactive scheduling

A last point that is common to many space telescope planning systems is the availability of manual and interactive scheduling modes in addition to fully automated search. This aspect is highly relevant for missions where it is hard to aggregate several optimization criteria into a single one, and where it is useful to propose several mission plans to the end-users, with some statistics that help in evaluating the quality of each of the plans proposed.

Uncertainty management

Almost all space telescopes must manage *uncertain events* such as ToOs or GRBs. As seen before, the precise management of uncertainty varies depending on the mission. For example, in INTEGRAL [35], a maximum filling percentage is specified for revolutions located in periods of the year where GRBs are more likely to be observed. In XMM-Newton [5], a maximum filling percentage is imposed over groups of successive revolutions. In Spitzer [29], uncertainty about data volume is handled on-board to avoid the over-filling of the spacecraft mass memory. In the long-term planning phase of the HST, soft realization windows are chosen for observations instead of fixed non-flexible windows [26]. For SVOM, a task reschedulability measure is optimized [34]. On the execution side, in SVOM, GRB events directly erase parts of observation plans, whereas for telescopes such as the HST and JWST, the plans uploaded to the satellites are ordered lists of observations that can be postponed at execution time instead of being erased.

Planning for INTEGRAL

In the following, we describe three case studies that we tackled in the past and show how the combination of incomplete search techniques and constrained-based approaches allowed us to produce good-quality plans. The first realization presented was developed within the context of a study initiated in 2007 by ESA, involving ISTC-CNR (Rome, Italy), VEGA (Darmstadt, Germany), the *Politecnico di Milano* (Milan, Italy), and ONERA. The objective of this study was to explore the use of AI Planning and Scheduling techniques for *ESA* missions [41]. ONERA was in charge of one of the test cases, namely the long-term planning (over one year) of the observation activities of the INTEGRAL space telescope. In the following, we give an overview of the specifications of the planning problem, the constraint-based model defined, the planning algorithm developed, the results obtained, and the lessons drawn from this work.

Mission description

INTEGRAL is an *ESA* mission, managed in cooperation with Russia and the USA, whose goal is to observe gamma-ray emissions from the universe. Starting in 2002 for at least two years, it has now been extended until 2020. As shown in Figure 1, the INTEGRAL satellite is moving on a highly elliptical orbit around the Earth. One orbit corresponds to 72 hours and only about 58 hours of these, out of the Earth radiation belts, are available for observation. Due to the presence of the Sun, the Earth, the Moon, and other planets, a given target is not permanently observable during these 58 hours. The satellite embeds four instruments: a gamma-ray spectrometer named *SPI*, a gamma-ray imager named *IBIS*, an X-ray monitor named *Jem-X*, and an optical monitor camera named *OMC*. These four instruments are fixed on the platform and point in the same direction. The Attitude and Orbit Control System (AOCS) allows the platform (and thus the instruments) to remain pointed in a given direction during an observation, and to move from one direction to another between two successive observations. For the long-term planning phase, the slewing time between two successive observations is indirectly considered by limiting the number of different observations within each orbit. Moreover, in order to keep some time available for opportunistic observations of unexpected events, such as the appearance of new X-ray/gamma-ray sources, only a given percentage of the observation time within each orbit is considered to be available. Constraints related to energy, data recording and downloading are not taken into account at this step.

From the long-term planning point of view, an Announcement of Opportunity (*AO*) is emitted each year, to which scientists answer by posting observation requests over targets of interest. Then, a target allocation committee selects observation requests and assigns to each of them a priority and a realization percentage above which the request is considered to be achieved. In general, a percentage of 100% is not mandatory. The AO used here covers a period from August 2007 to August 2008 and involves 123 orbits and 35 observation requests.

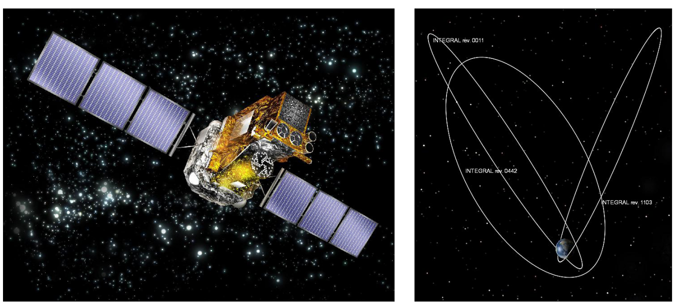


Figure 1 - The INTEGRAL satellite and its highly elliptical orbit (image credit: ESA)

Each high-level observation request r is decomposed into $NEO(r)$ elementary observations corresponding to precise pointings defined by an observation mask. Observations of a given request all have the same duration, and $NEO(r)$ is within the interval [1,1023] in practice, most of the observation requests requiring several hundreds of elementary observations. It is not mandatory and it is often impossible to perform all of these $NEO(r)$ elementary observations within a single orbit of the satellite.

Figure 2 shows an example of a solution plan for an instance involving 5 orbits and 6 observation requests. For request r_1, we have 6 observation windows and 4 observation activities: the first one involving 3 elementary observations in the first window and the other three involving each 2 elementary observations in the last three windows. For request r_2, we have 4 observation windows and 4 observation activities, each involving only one elementary observation in each window. We can observe 2 observation activities in the same window for request r_3 in the last window. We can also observe that no observation activity is associated with requests r_5 and r_6. Finally, we can observe that the duration of an elementary observation depends on the request considered: for example, greater for r_2 than for r_1.

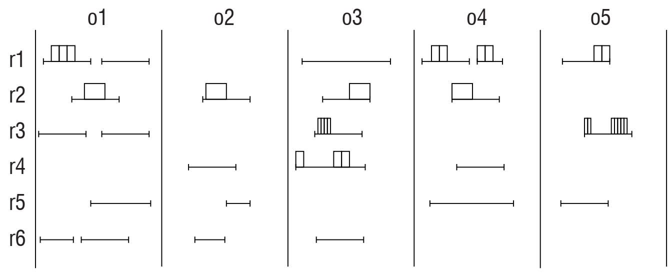


Figure 2 - Example of a solution plan for the long-term planning phase of INTEGRAL

In INTEGRAL, a type is associated with each observation request of a given target. This type specifies the way in which the observations must be performed, and there are four types of requests:

- normal observation requests (NO) which must be split as little as possible and ended as early as possible after they have started;
- no-splitting observation requests (NS), which must not be interleaved with other observation requests;
- periodic observation requests $PE(p,t)$, which must be decomposed into elementary observations performed every p orbits with a tolerance t on the deviation from the period;
- spread observation requests ($SP(n)$), which must be decomposed into n sub-observations to be spread as much as possible over the year.

The long-term planning problem consists in selecting and scheduling over the next observation period, generally of one year duration, the observations associated with the current *AO*, plus the observations selected at the previous *AO* but which were only partially carried out.

The resulting problem is a kind of over-constrained scheduling problem [40, 25] where the objective is to fulfill requests as much and as well as possible, taking request priorities into account and knowing that observations cannot overlap. The long-term plan also serves as an input for regular short-term planning which decides on the detailed activities to be performed by the satellite, taking into account the arrival of new urgent observation requests.

Constraint-based modeling

In this study, following a constraint-based optimization approach, the first step was to formalize the problem by providing a clear definition of the input data, the decision variables, the constraints, and the objective function. The full model is available in [35], and we provide thereafter only its main features.

Input data

We consider a set O of orbits over the planning horizon, with a maximum observation duration allowed for each orbit. We also consider a set R of observation requests, with for each request $r \in R$:

- a type $TY(r)$ among the four possible ones (NO, NS, $PE(p,t)$, $SP(n)$);
- a set $W(r)$ of windows available for carrying out r over the year; each window is included in a single orbit and corresponds to a period where the telescope is outside the Earth radiation belts;
- a weight $WE(r)$ reflecting the priority of the request;
- a number $NEO(r)$ of elementary observations that all have the same duration;
- a percentage above which r is considered to be achieved.

To model the problem, the main difficulty is that the number of *observation activities* used for one request r is not known initially, where an observation activity corresponds to a set of contiguous elementary observations performed for r within a given visibility window $w \in W(r)$. For instance, in Figure 2, there is one observation activity composed of 3 elementary observations in the first visibility window of r_1. In practice, it is not feasible to introduce as many observation activities as the number of elementary observations, which is why the maximum number of observation activities per request is restricted. With each normal observation request $r(TY(r)=NO)$, we systematically associate two possible observation activities per window $w \in W(r)$, and with each special observation request $r(TY(r) \neq NO)$, we associate only one observation activity per window $w \in W(r)$. For the instance we worked on, we ended up with 2731 candidate observation activities. In the sequel, OA denotes the set of candidate observation activities and $OA(r)$ (respectively, $OA(o)$) denotes the set of candidate observation activities associated with request r (respectively, orbit o). We also consider a maximum number of non-empty observation activities per orbit, to limit the overall slewing time.

Variables

The problem is then to choose for each candidate observation activity $oa \in OA$, its starting time $s(oa)$ and the number $neo(oa)$ of elementary observations that it involves.

From this, it is possible to compute the values of other variables that are functionally dependent on the $s(oa)$ and $neo(oa)$ variables. First, the ending time $e(oa)$ of oa can be directly deduced from the fixed duration of each elementary observation. Then, for each request $r \in R$, the total number $neo(r)$ of elementary observations performed for r is defined as $neo(r) = \sum_{oa \in OA(r)} neo(oa)$. This number must not exceed $NEO(r)$, the total number of elementary observations required for r. Last, for each request r, it is possible to compute the sequence of non-empty observation activities associated with r, ordered according to their starting times, and to get a start time $s(r)$ (start time of the first observation activity carried out for r) and an end time $e(r)$ (end time of the last observation activity carried out for r).

Constraints

Several constraints must be satisfied by the variables of the model. Some of them are quite standard in terms of optimization:

- *time windows*: each observation activity must be included in the visibility window w with which it is associated;
- *disjunctive resource constraints*: given that there is a unique telescope resource over which the instruments are body-mounted, observation activities cannot overlap; non overlapping constraints can be expressed separately for each orbit, because windows associated with two distinct orbits are disjoint;
- *knapsack constraints*: for each orbit o, the maximum observation duration within o must not be exceeded;
- *cardinality constraint*: for each orbit o, the maximum number of non-empty observation activities must not be exceeded.

Beside these standard constraints, each specific request type induces side constraints that are less standard for generic optimization methods:

- for each *no-splitting observation request* r, there must be no interleaving between the observation activities associated with r and the observation activities associated with other requests;
- for each *periodic observation request* r, its period must be respected (up to the tolerance allowed) and only elementary observation activities must be used ($neo(oa)=1$);
- for each *spread observation request* r, there is a maximum number of elementary observations for each observation activity associated with r.

Objective functions

In INTEGRAL, the definition of the optimization criteria is not as easy as the definition of the constraints is. After discussion with the end-users, we adopted the following approach. With each request r are associated:

- a *quality of completion* $qc(r) \in [0,1]$, which measures the percentage of completion of r;
- a *quality of realization* $qr(r) \in [0,1]$, which measures to what extent the set of observation activities used for r are consistent with the type of r; the definition of this quality of realization depends on the request type; for normal and no-splitting requests, the quality is higher when r finishes as early as possible after it started (maximum grouping objective); for periodic observation requests, the realization quality is higher when the deviation from the ideal period is lower; for spread observation requests, the realization quality is the mean realization quality of its observation activities;

- an *overall quality* $q(r)$ obtained as a linear combination of the two previous qualities: $q(r)=\alpha \cdot qc(r)+(1-\alpha)\cdot qr(r)$ with $\alpha \in [0,1]$ a parameter that can be adapted by the end-users.

The global criterion q to be maximized is defined as the normalized weighted sum of request qualities:

$$q=\frac{\sum_{r\in R} WE(r)\cdot q(r)}{\sum_{r\in R} WE(r)} \quad (1)$$

Stochastic hill-climbing with restarts

Considering only variables $neo(oa)$ for the real instance to be solved leads to 2731 variables whose domain size is between 2 and 1024. First experiments were made with generic constraint-based optimization tools, which have, in theory, the capacity to find the optimal solution, but the results were not satisfying essentially due to the large size of the search space and to the complexity of the non-standard constraints and criteria. Instead, local search algorithms and meta-heuristics were used in order to produce good quality solutions within limited computation times. The main features of the algorithm developed are the following:

- *local search moves:* The algorithm starts from an empty plan; it maintains a current plan and it modifies it iteratively using two kinds of local moves: either the enlargement of an observation activity oa (by adding to oa as many elementary observations as possible), or the enlargement of an observation activity oa after the removal of another observation activity oa' located in the same orbit; after each step of the algorithm, the consistency of the current plan is maintained, meaning that all of the model constraints are satisfied;
- *restricted neighborhood:* at each step of the algorithm, a small subset of the set of possible local moves is pre-selected, by taking into account the weights associated with the requests; pre-selection is necessary because of the huge number of observation activities and thus of possible local moves; to avoid cycles around local optima, a tabu list of the T previous local moves is maintained, and the local moves included in the tabu list cannot be pre-selected;
- *stochastic hill-climbing*: all of the pre-selected local moves are evaluated by estimating their positive or negative impact on the optimization criterion; one candidate move is then randomly selected among the best ones; the selected local move is effectively applied if the estimated impact is strictly positive and applied with a certain probability if the estimated impact is negative or null, as in simulated annealing [23];
- *restarts*: the algorithm restarts from an empty plan each time a maximum number of local moves without improvement is reached.

Inside the algorithm, the basic scheduling constraints associated with each individual orbit are managed by the core APSI toolbox [6]. Basically, the latter models dynamic systems based on a set of *timelines*, each of which represents the evolution of a component of the system. In APSI, different types of timelines can be used, including timelines modeling state variables or resources, and different kinds of constraints over timelines can be specified, including state and temporal constraints. For INTEGRAL, the specific constraints associated with no-splitting, periodic, and spread requests, as well as the optimization of the plan quality, are managed outside of APSI.

Experimental results

The planning tool implemented can be used to visualize the evolution of the current plan during a search, the best plans found, and statistics on the quality of completion and realization of each observation request. Figure 3 shows some visualizations available. For advanced users, it is also possible to set parameters such as the probability of acceptance of a local move that does not increase the plan quality.

In the one-year instance described before, the algorithm takes in general only some minutes to achieve plans whose quality is close to 0.97 or 0.98, *i.e.*, very close to 1, which is an upper bound on the plan quality. This means that, in the worst case, the best quality obtained is only 2 or 3% below the optimal one.

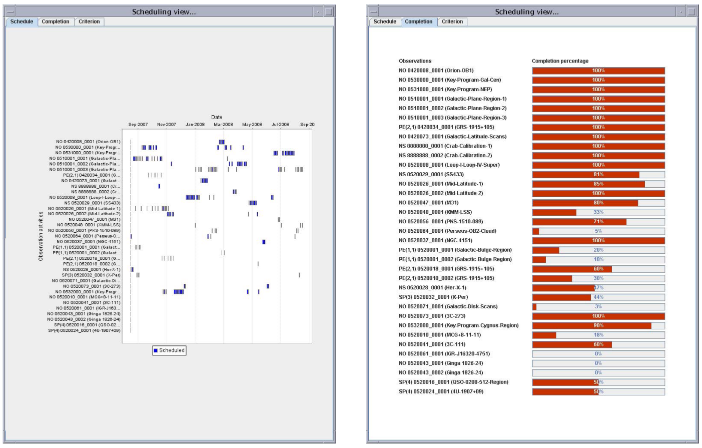


Figure 3 - Left: current plan over the year (one line per request, one blue rectangle per observation activity). Right: current completion percentage of each request

Lessons learned

In this study, it was possible to build an unambiguous constraint-based model for the long-term planning phase of INTEGRAL. The main effort was made on the formal aspects related to the specific request types, on the partition between features modeled as constraints and features modeled as optimization criteria, and on the decomposition of observation requests into observation activities potentially spread over several visibility windows. On the solving side, the core APSI framework was able to manage the basic scheduling constraints, such as the constraints of no overlapping between observation activities within each orbit. Several aspects related to specific request types were, however, not manageable by APSI in its 2009 version, and specific developments were required for managing these specifications outside APSI. Also, a stochastic hill-climbing algorithm with restarts was able to produce very good results in terms of plan quality and computing time, and in the end what previously required some days of manual work now requires only a few minutes of computing. Last, APSI was used as a non-incremental planning and scheduling engine in the sense that the scheduling problem associated with one orbit o is solved from scratch whenever one observation activity is added to o. The number of local moves carried out per second might have been higher by using an incremental solving strategy or a core planner tuned for local search.

Planning for SVOM

We now consider a second mission and show the results obtained by following a similar methodology: definition of a formal constrained-based model, definition of local search and metaheuristics, and development of visualization tools. One difference between SVOM and INTEGRAL is that between the two missions, we developed a core constraint-based optimization library dedicated to local search [36, 33].

Mission description

SVOM is a future Chinese-French space mission, which should be launched in 2021. It is dedicated to the study of the transient universe, which includes the observation of GRBs. The main partners involved are the Chinese Academy of Science, the Chinese and French space agencies (CNSA, CNES), the Shanghai Engineering Center for Microsatellite (SECM), and several Chinese and French science labs. The SVOM mission has a nominal duration of 3 years and an extension phase of 2 years.

As shown in Figure 4, the SVOM telescope carries four instruments: a coded-mask gamma ray imager named ECLAIRs, a gamma-ray spectrometer named GRM, a Micro-channel X-ray Telescope named MXT, and a Visible-band Telescope named VT. It will operate around the Earth at an altitude of 650km and follow a default attitude law called "B1 pointing law" which is roughly anti-solar. It provides an effective observation during the night hemisphere to enhance the follow-up possibilities and also avoids the Galactic plane and some bright X-sources to foster GRBs detection [9].

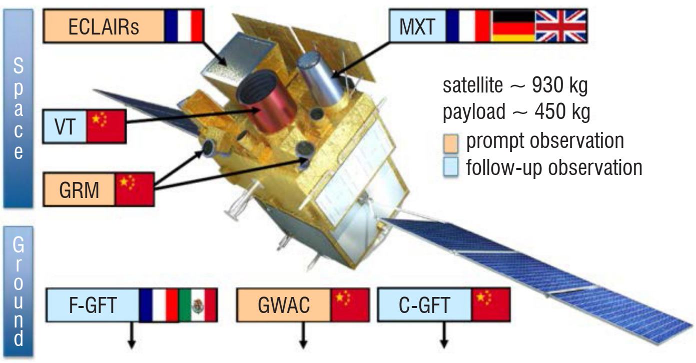


Figure 4 - Instruments of the SVOM space telescope (http://www.svom.fr/en/)

The SVOM scientific program is divided into three categories:

- the *Core Program* (CP), dedicated to GRBs; the latter are detected and managed autonomously on board; with an expected GRB rate of around 60-70 per year; GRBs are unpredictable by nature, and when a GRB is detected on board the current observation plan is interrupted and the satellite remains pointed towards the GRB source for 14 orbits ($\simeq$ 1 day);
- the *Target of Opportunity* (ToO) program, which allows new and urgent observations to be triggered from the ground; ToO observations must be performed by the satellite within 48h for a standard ToO and within 12h for an exceptional ToO (e.g., galactic supernova or gravitational wave alert); the typical observation duration is between 1 orbit and 14 orbits;
- the *General Program* (GP), consisting in observations requested by the scientific community and which are pre-planned one year in advance; the GP targets are classified into three categories: *A-targets* (high priority), *B-targets* (low priority), and *Fill-In Targets* used to provide a default target to the satellite if needed; in the following, we do not consider the fill-in-targets; the duration of a GP request is from 1 orbit up to 5 days; the observation time is allocated to scientific users' requests with a ratio of 60% for the Chinese users' group and 40% for the French users' group; also, the GP allows pointing at sources close (within 10°) to the default B1 attitude pointing law.

Figure 5 gives an idea of the distribution of the mission time among these three programs, both for the nominal mission and for the extended mission. One challenge is then to maximize the GP completion at the end of the year, especially for the A-targets, despite the occurrence of GRBs and ToOs. This challenge must be tackled given that in SVOM the mission planning process works as follows:

- a GP pool of proposals is established once a year after a Call for Obs and a selection process, and a first schedule is computed over the one year span;
- every week the pool of proposals is updated to take into account the GP observations canceled or partially carried out the week before due to the occurrence of GRBs and ToOs; the schedule is fully recomputed up to the end of the year to plan again these missing parts if possible;
- if an observation is not finished by the end of the year, the missing part might be selected to be included in the set of candidate requests for the next year; for each GP observation, it is assumed that at least 95% of the requested exposure time must be fulfilled to obtain usable data.

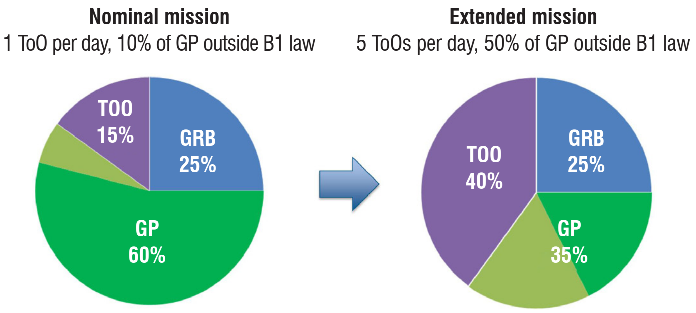


Figure 5 - Distribution of the useful mission time over one year; GP observations are split between those performed outside the B1 pointing law (light green) and those performed around this default law (dark green)

Constraint-based modeling

As for INTEGRAL, we give an overview of the constraint-based model developed for SVOM. See [34] for a full description.

Input data

We consider a set R of observation requests that are candidates for being scheduled over the remaining part of the year, and a set of coarse-grain directions containing B1 (around the default B1 pointing law), HL (High Latitude), and MWC / MWEP / MWWP (Milky Way Center / East Part / West Part). Each request $r \in R$ is defined by several elements:

- a request category among ToO (Target of Opportunity), CAL (Calibration), and GP (General Program), with a priority level

(A or B) and the user requiring the observation ("CH" for Chinese, "FR" for French) for GP requests;
- a direction category in $\{B1, HL, MWC, MWEP, MWWP\}$ and the precise coordinates of the target associated with the request;
- a list $W(r)$ of time windows during which the target associated with r is visible;
- the observation duration associated with r, split between the remaining observation duration for r and the observation duration already elapsed (not null for observations truncated due to GRBs and ToOs).

As an input, we also have, for each user $u \in \{CH, FR\}$ (respectively, for each direction $d \in \{B1, HL, MWC, MWEP, MWWP\}$), the desired satellite usage ratio for user u (respectively, for direction d). Considering usage ratios for directions is useful to make sure that the telescope does not point outside the B1 pointing law too much, which favors the observation of GRBs.

Decision variables

Following the specifications of the mission, we impose that observation requests must be planned in a single block. This means that the observations are non-preemptable at planning time, even if GRBs and ToOs might cause interruptions at execution time. An observation plan is then defined as a sequence $seq = [r_1, \ldots, r_k]$ of successive observation requests planned for the satellite, with for each request $r_i \in seq$ a time window $win(r_i) \in W(r_i)$ chosen for carrying out r_i.

Constraints

Two basic scheduling constraints must be satisfied:
- *time windows*: each observation for a request r must be carried out within the window $win(r)$ chosen for r;
- *no overlap*: the observations successively carried out must not overlap, since all instruments are body-mounted on the telescope.

Such constraints lead to a standard scheduling problem with a unique machine (the telescope) and time windows.

Objective functions

From a scheduling perspective, the main challenge in SVOM is actually to be able to manage multiple objectives for the construction of good quality plans. The model considers six sets of objective functions listed below:
- $nObs(p)$, which measures the number of GP observations carried out for priority p; this number must be maximized, with a strict preference for priority A;
- $duPrio(p)$, which measures the total observation duration for priority p; this duration must be maximized, with a strict preference for priority A;
- $duUser(u)$, which measures the cumulated observation duration for user u; this duration must respect the desired usage ratio as much as possible;
- $duDir(d)$, which measures the cumulated observation duration for direction d; this duration must respect the desired usage ratio as much as possible;
- $slew$, which measures the cumulated slew induced by the chosen observation plan, given the coordinates of the successive targets to which the telescope must be pointed; this cumulated slew must be minimized;
- $reschedulability(r)$, which measures the *reschedulability* of each request r; this metric corresponds to the amount of time available for rescheduling r (or parts of r) in case of interruptions due to the occurrence of GRBs and ToOs; for a request r in the observation sequence, this reschedulability index is maximum when the realization of r starts at the beginning of the earliest realization window of r (potentially many possibilities to reschedule r in this case), and it is equal to 0 when the realization of r ends at the end of its latest realization window (no opportunity to reschedule r in this case); ideally, the mean reschedulability must be maximized and its standard deviation must be minimized, to achieve a fair distribution of reschedulability between observation requests.

As shown in the next section, some choices were made at the level of the planning algorithms to establish an order between these objective functions.

Greedy search and iterated local search

We now describe the components of the planner developed for constructing plans every week over the rest of the year.

Core constraint-based reasoning engine

As mentioned before, the generic InCELL library is used to handle the constraint-based model of the mission. Basically, InCELL implements the Constraint-Based Local Search paradigm [17]. It allows us to incrementally evaluate the impact of additions or removals of observations on all constraints and objectives of the model, and to use predefined primitives for implementing local searches and meta-heuristics.

Greedy search

The SVOM planner starts from a plan that contains only the set of regular calibrations placed in fixed windows and the set of mandatory ToOs known at planning time. The algorithm then tries to insert GP-observations one by one into this plan. This search phase is greedy in the sense that once an observation is inserted into the plan, it is never removed. As discussed previously, in order to fully instantiate such a search procedure, two main parameters must be set, namely a *selection heuristic* to select a candidate observation at each step and an *insertion heuristic* to determine an insertion position into the current plan.

The selection heuristic defined for SVOM promotes: (1) the fair sharing of the telescope among the users (relatively to the ideal usage ratios defined in the input data), (2) the fair sharing of the telescope among the categories of directions (again, relatively to the ideal usage ratios defined in the input data), and (3) the realization of the highest priority requests. A fixed lexicographic ordering is used to combine these three aspects and obtain at each step a set of candidate requests. One candidate in this set is chosen based on a portfolio of possible decision rules, such as (1) the selection of one request that is the most constrained in terms of available time windows, (2) the selection of one request whose duration is minimum, (3) the random selection of one request, or (4) the selection of one request whose observation has already been started in the

Algorithm	CPU time (sec)	Priority A		Priority B		user ratios	slew (deg)
		nObs	reschedulability in days	nObs	reschedulability in days		
greedy	21.4	190/193	**115.5(70.2)**	250/408	40.0(38.3)	0.41/0.59	23483
ILS (perturb 0.1)	300	**192**/193	99.1(74.7)	**261**/408	41.2(43.0)	**0.40/0.60**	24338
slew optimization	300	**192**/193	98.3(73.1)	**261**/408	**42.9(44.8)**	**0.40/0.60**	**13219**

Table 2 - Results obtained in a one-year scenario including more than 600 observation requests (Intel i5-520 1.2GHz 4GBRAM processor, time limit of 5 minutes); for the reschedulability, we give the mean value and the standard deviation in parentheses

past, to favor the completion of observations interrupted by GRBs or ToOs. For the insertion heuristic, several strategies were tested, including (1) insertion at the earliest feasible position that exploits an idle period of the telescope; (2) insertion at a position that maximizes the mean reschedulability of GP requests of priority A; and (3) insertion at a position that minimizes the cumulated slew. For these insertion heuristics, moving back observations of the current plan is allowed, and only insertion positions leading to a feasible plan are considered.

Iterated local search

Given that the heuristics used in greedy search are imperfect, a more efficient planning strategy was developed. The latter uses the *Iterated Local Search* (ILS) metaheuristic [27]. It starts from the solution produced by the greedy search process, and then iterates two search phases until a maximum CPU time is reached:

- a *perturbation phase*, during which x% of the observations are removed from the current plan with x a parameter to be set; the observations removed are chosen using a uniform random distribution;
- a *reoptimization phase*, during which the solution obtained after the perturbation phase is reoptimized; this phase reuses the greedy search scheme to fill the plan again; to diversify search, the heuristic used for filling the plan is chosen randomly among the portfolio of decision rules.

Throughout the iterations, the best plan found is systematically recorded, based on a lexicographic ordering of the optimization criteria.

Post-processing: slew optimization

For SVOM, the cumulated slew can actually be considered as less important than the other objectives because the pointing of the telescope will be perturbed anyway due to the commitment to carry out GRBs and ToOs. This is why the slew objective is considered only at the last optimization step.

To achieve an observation sequence $seq = [r_1, \ldots, r_n]$ that optimizes the cumulated slew, local search techniques developed for routing problems with time windows are used [39]. The corresponding techniques are *or-opt moves* [2], which try to better position a block of k successive observations inside the observation sequence, and *2-opt moves* [10], which consider k successive observations and try to perform them in the reverse order. Local moves are carried out while improvements are made, i.e., until a locally optimal cumulated slew is reached. Then, an ILS search scheme is used, with a perturbation phase that randomly updates the ordering of some observations and a slew reoptimization phase based on or-opt and 2-opt moves again. This mechanism is applied until the maximum CPU time allowed is reached.

Experimental results

Experiments were performed on several data sets to evaluate the performance of the algorithms for the initial planning phase, when the full-year schedule must be synthesized. Table 2 gives the results obtained on one data set involving approximately 600 GP requests. In this scenario, greedy search delivers good quality solutions in a few seconds. Experiments on the different selection and insertion heuristics indicate that the heuristic that fills the earliest idle periods is a good compromise between the computation times and the plan quality. Also, ILS leads to better results in terms of number of observations performed and in terms of observation duration, especially for GP requests of priority B. Using ILS can however penalize the reschedulability criterion a bit. In 5 minutes, ILS performs up to 100 plan perturbation-reoptimization steps. Last, the slew optimization phase has a strong impact on the cumulated slew (50% reduction), while having little impact on the reschedulability objective.

We also simulated the dynamic behavior of the planning system by calling the planner every week to schedule the remaining part of the year. The objective was to determine whether GP requests were completed despite the random arrival of ToOs and GRBs. We conducted the experiments on a set R composed of 426 GP-real-life requests. For each set of scheduling parameters, 20 one-year simulations were completed, each time with random occurrence dates for ToOs and GRBs. For these 20 simulations, Table 3 shows the mean number n_{GP} (respectively, n_{GP-A}) of GP requests (respectively, GP requests of priority A) that are completed up to 95% at least at the end of the year, and the mean observation time dedicated to these requests (columns t_{GP} and t_{GP-A}).

The results show that the ILS phase, which starts here from the solution found by the greedy search, improves the performance obtained after one year of mission time.

	n_{GP}	t_{GP} (days)	n_{GP-A}	t_{GP-A} (days)
greedy	206	142	177	116
ILS - perturb 10%	**218**	**160**	**186**	**132**

Table 3 - Number of observations and observation duration after a one-year simulation

Short-term planning

The techniques presented so far concern the long-term planning phase of SVOM. For the short-term planning phase, another approach is defined to deal with the arrival of ToOs [19]. The main idea is that when a ToO request is received, there is a need to construct a ToO observation plan over a one-day span. For this, the relevant sky areas are decomposed into tiles, and the main task of the short-term planner is to select a subset of the tiles and define the order in which the selected tiles are observed, given constraints on stabilization times, visibility windows, and maximum number of tiles per orbit, and given the likelihood that a given tile contains the source sought. Basically, this tile-sequencing process is a chronological greedy algorithm that iteratively inserts tile observation activities at the end of the current plan. At each step, the algorithm selects either a tile that has the maximum likelihood of containing the source (static selection heuristic), or a tile that maximizes a value depending both on the source presence likelihood and on the last tile of the plan (dynamic selection heuristic).

Lessons learned

On the modeling side, the mission constraints for SVOM are very simple but the optimization criteria required a bit more effort to be derived. On the implementation side, one lesson is that it was very useful to have a generic tool for managing the core constraint-based model. It allowed the lexicographic ordering between the different objective functions to be easily updated during the project. It also allowed various search parameters to be tested. Also, to optimize the slew, the use of standard Operations Research techniques was very beneficial.

Last, even if there is no theoretical guarantee on the stability of the plans produced for GP observations, some settings used favor replanning observations that have already been started or that are more time-constrained. This is why the approach manages to complete observations up to 95% at the end of the year. More precisely, the uncertainty about GRBs and ToOs is handled through two main mechanisms: (1) regular replanning each week during the year, which allows parts of GP observations aborted because of high priority events to be programmed again, and (2) optimization of the reschedulability metric. Given that GRB events will cover around 25% of the useful mission time, and given that ToOs will cover between 15% and 40% of the time, more proactive planning strategies have been sought. In particular, we started to develop a planning process in which, in a first phase, all observation durations are scaled proportionally to the percentage of useful mission time covered by random events. Returning to nominal durations, opportunistic observations can then be added to avoid under-using the telescope. This process has the potential to produce more stable plans, since the plan in which all observation durations are scaled could serve as a reference to be followed, as much as possible, over the year.

Planning for ARIEL

Mission description

We present in this section the ARIEL mission, which is the fourth medium-class mission within the ESA Cosmic Vision science program, with a launch planned in 2028. The total lifetime duration is four years and could be extended for two additional years.

In astronomy, a *transit*, or *occultation*, is the phenomenon when a planet passes directly in front of or behind its host star from the satellite point of view. ARIEL will analyze the atmospheres of around 1000 planets (warm and hot transiting gas giants, Neptunes and super-Earths) orbiting around a range of host star types, using transit and occultation spectroscopy in the $\sim 2-8\ \mu$m spectral range and broad-band photometry in the optical to determine their chemical composition and physical conditions. The results will help scientists better understand planet formation, putting our own Solar System in context.

Transit and occultation spectroscopy methods, whereby the signal from the star and planet are differentiated using precise knowledge of the planetary ephemerides, allow atmospheric signals from the planet to be measured. Figure 6 illustrates the orbital lightcurve of the transiting exoplanet HAT-P-7b as observed by Kepler [3], which the methods adopted by ARIEL are based on.

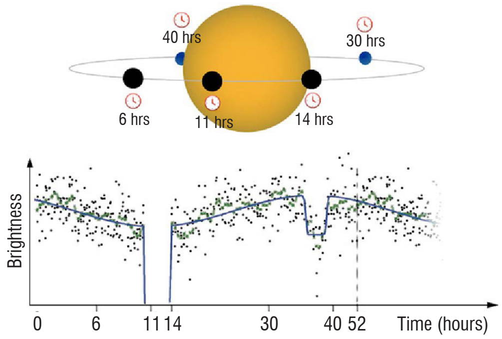


Figure 6 - Orbital lightcurve of exoplanet HAT-P-7b [3]

Observation objectives

ARIEL will visit a large and well-defined set of a few hundred targets. Repeated visits are required to build up the Signal to Noise Ratio (SNR) of individual target spectra. Most of the targets will require between one and a few tens of transit/occultation observations, depending on the brightness and spectral type of the host star and planetary radius and temperature. The maximum duration of a visit to a target system will be less than 10 hours. The time between successive transit/occultation observations will depend on the orbital period and could be as little as a fraction of a day to as long as a few days, with the exception of highly eccentric orbit planets.

Every targeted planet is associated with one or more of the following scientific objectives:

- Basic survey objective (denoted by *Survey*): it requires a minimal number of transit or occultation observations in order to assess the scientific interest of the planet and its main coarse characteristics. It basically addresses all of the targeted planets.
- Deep survey objective: it requires additional transit or occultation observations in order to obtain a better SNR and to achieve a detailed characterization of the planet and its atmosphere. It addresses a large sub-sample of the set of targeted planets.
- Benchmark objective: it requires even more transit or occultation observations to reach the best possible SNR allowing a very detailed knowledge of the chemistry and dynamics of the planet. It addresses a few tens of planets orbiting very bright stars.

According to the definition of these objectives, the global observing strategy that is promoted is to prioritize, at the beginning of the lifetime, the observations for planets with either a benchmark objective first or a basic survey objective, and to postpone the observations accounting only for a *Deep* objective after a period of about 1 or 1.5 years from the beginning of the mission.

Payload calibration needs and operational constraints

During the mission, it is assumed that the operational needs are as follows:

- one housekeeping sequence during 4 hours every 28 days, with a tolerance of +/- 2 days;
- one long calibration sequence during 6 hours every 30 days, with a tolerance of +/- 10 days;
- one short calibration sequence during 1 hour every 36h, with a tolerance of +/- 12h;
- some specific targets may require a short calibration sequence to be performed just before/after the transit observation.

All of the calibration sequences use dedicated targets to point to numerous wide spread predefined G-stars. The mission data downlinks are assumed to not conflict with the observations.

Mission planning challenges and constraints

A mission planning process is required to establish the observation schedule from a set of requests R initially selected and classified by a scientific board. It provides, in particular, the allocations for the three objectives and the types and numbers of observations needed. The mission planning process must take into account the observing strategy promoted for ARIEL and all of the relevant system constraints.

Note that at the time when the study on ARIEL was performed, it was still a mission candidate. Consequently, the goal of the study presented was the evaluation of the global scientific impact of the mission in the following way:

- the various objectives could be achieved for the highest number of targeted planets. In particular, the *Benchmark* objective could be met for all of the planets concerned and a high number of *Survey* objectives could be met at the beginning of the lifetime (500 surveys done the first year);
- a sufficient part of the useful mission time was devoted to scientific observations (the goal is of 80%) avoiding long time periods without observation activities;
- the regular calibration and housekeeping tasks could be fulfilled most of the time.

We describe here the main outcomes of this study [38]: a constraint-based model, two algorithmic approaches, experimental results, and lessons learned.

Constraint-based modeling

Input data

We consider the following input data:

- *mission dates* – start and end dates of the mission, and desired dates before which the *Benchmark*, *Survey* and *Deep* objectives should be performed;
- *satellite features* – slew speed and duration to wait after each slew in order to be thermally and mechanically stable;
- *operational tasks* – each operational task can either be a short calibration, a long calibration or a housekeeping task. Each type has an associated duration, period and flexibility;
- *calibration stars* – stars (name, right ascension and declination) that are the targets to point to for calibration tasks;
- *exoplanets* – each planet is characterized by a name, its orbiting star, its period and its coordinates (right ascension and declination). Candidate transit and occultation events for each planet can be pre-computed along with their start and end dates. The duration of each event is also given as an input;
- *scientific requests* – description of all of the planet observations that can be carried out during the mission. Each scientific request targets one planet and can focus either on its transit events or occultation events.

The numbers of observations required to achieve each objective (among *Survey*, *Deep*, or *Benchmark*) are also given as an input. A request is called a *Survey* request if and only if its number of *Deep* and *Benchmark* observations is equal to 0. We define *Deep* and *Benchmarks* requests along the same lines.

Orthogonally to the objective dimension, requests can also be partitioned into:

- a set of *simple requests*, for which there is no required task before and after observations of events;
- a set of *requests with calibration*, for which each observation must be preceded and/or followed by a short calibration that is performed on the closest G-star to the planet; such requests can be considered as simple requests with a longer duration, for which the pointing target can be either the planet or the G-star depending on the calibration requirement;
- a set of *requests with additional observations*, for which a specific task must be performed before and/or after each observation of an event.

Variables

There are two classes of discrete decision variables:

1. Boolean variables for deciding which observation candidates are in the final plan; the start and end dates of these observations are fixed (dates associated with transit and occultation events) and therefore do not require any decision;
2. Boolean variables for deciding which operational tasks are in the final plan and integer variables for their start and end dates.

Constraints

We consider the following constraints:

- *no overlap*: the tasks performed by the telescope should not overlap. This takes into account the slewing duration between the pointing targets of tasks and the stabilization duration;
- *requests with additional observations*: for a request with additional observations, a candidate observation is part of the final plan if and only if the observation just before and/or just after is also part of the final plan;
- *operational tasks periodicity*: operational tasks must be performed periodically, with a given flexibility as specified by their types;
- *deep observation release date*: observations that allow the objective of *Deep* requests to be achieved cannot be made before a fixed date.

Objective functions

We first list the different elementary criteria taken into account for the mission and then describe two combinations considered.

- $crit\_nReq$ - maximize the number of completed requests;
- $crit\_nB$ (respectively, $crit\_nD$ and $crit\_nS$) - maximize the number of completed *Benchmark* (respectively, *Deep* and *Survey*) requests;
- $crit\_nB\_d$ (respectively, $crit\_nS\_d$) - maximize the number of *Benchmark* requests (respectively, *Survey* objectives) completed before the desired date;
- $crit\_nOp$ - maximize the number of operational tasks carried out;
- $crit\_nHk$ (respectively, $crit\_nLc$ and $crit\_nSc$) - maximize the number of housekeeping (respectively, long and short calibrations) carried out during the mission.

All of the criteria above do not have the same weight, because of the mission observation objectives. Following these mission priorities, we considered two main criteria:

- *Upper bound criterion*. This criterion does not take into account the request types, but rather focuses on the maximizations of the number of completed requests and the number of operational tasks in the plan, thus giving a clue of what could be an upper bound for the planning of ARIEL regarding the number of completed requests. Formally, the upper bound criteria $crit_{UB}$ is defined by a vector $[crit\_nReq, crit\_nOp]$ that is optimized lexicographically.
- *Criterion with request types*. This criterion, denoted as $crit_{Type}$, takes into account the type of requests and the date before which *Benchmark* requests should be completed. Formally, $crit_{Type}$ is vector $[crit\_nB\_d,\ crit\_nB,\ crit\_nD,\ crit\_nS,\ crit\_nHk,\ crit\_nLc,\ crit\_nSc]$ and is optimized lexicographically.

Greedy search and local search

We developed two different approaches. The first one is greedy-based. The second one uses a Constraint Based Local Search paradigm.

Note that a third approach could be adapted from the one used for EChO [30], which is an earlier version of the ARIEL problem in which there was no distinction between *Survey*, *Deep* and *Benchmark* requests. EChO used a two-phase strategy, where first scientific requests are planned using genetic algorithms, and then as many operational tasks as possible are inserted to fill in the gaps in the plan. We did not experiment with this approach on the ARIEL benchmarks.

Greedy approach

Many greedy-based algorithms were developed for solving the ARIEL planning problem. We describe here a hierarchical greedy algorithm that gives the best results, as described in Section "Experimental results".

Starting from an empty plan, the algorithm selects a candidate task and tries to insert it into the plan. More precisely, all requests and candidate observations are first labeled as *unprocessed*. While there is an unprocessed request, one request r is selected by considering first *Benchmark* requests, then *Deep* and *Survey* ones and with a tie-break favoring those with the least flexibility in further observations. Then, the first unprocessed observation for this request is inserted into the final plan if and only if its insertion does not violate any constraint. The observation is marked as *processed*. If the objective is achieved or if all candidate observations of r have been processed, then r is also labeled as processed. If the *Survey* objective of r is not achieved, then all of the observations inserted for r are removed. When all requests have been considered, a similar procedure is followed for operational tasks.

This algorithm tends to maximize the number of completed requests, while taking into account the priority of the tasks. The dates before which benchmark requests and survey objectives should be completed are also considered, since the inserted task is always the first one chronologically.

In order to increase the cumulated duration of activity of the satellite, two procedures have been defined. First, we try to insert into the plan as many observations as possible but only for requests that have at least achieved their *Survey* objective. The second procedure extends the time during which calibration associated with scientific events are carried out.

Min-conflicts approach

The second approach is a Constraint Based Local Search (CBLS) approach [17]. It is particularly suited for handling constraint programming problems with large benchmarks, especially because various parts of the search space can be explored in a short time. The algorithm built on top of that approach is based on a min-conflicts algorithm [28]. It starts with an initial plan in which all requests are randomly fulfilled and constraints are all satisfied except for the no-overlap ones. Then, the objective of the algorithm is to remove all conflicts due to overlaps and then optimize the criteria $crit$ that can be equal to $crit_{UB}$ or $crit_{Type}$. The steps are as follows:

1. while there exists an overlap conflict or the criteria *can theoretically be improved*, we select an observation o_{old} in the plan;
2. we select a candidate observation o_{new} of the same request that, if inserted into the plan in place of o_{old}, either decreases the number of overlaps, or improves the value of $crit$ without deteriorating the number of overlaps. The insertion of o_{new} must also satisfy all constraints except for the no-overlap ones. A random tie-breaking is used to choose between candidate observations that improve the criteria in the same way;
3. we remove o_{old} from the plan and insert o_{new} instead;
4. we insert all housekeeping tasks in their corresponding temporal interval as early as possible if and only if they do not overlap with already inserted observations. We then proceed the same way for long and short calibrations.

These steps are repeated until the criteria reach the maximum theoretical value or until a maximum number of iterations is reached. If overlap conflicts still exist, an observation from the plan is chosen and removed. The algorithm starts again from Step 1 and the procedure is repeated.

Note that $crit\_nReq$ cannot be improved if its value is the number of requests. Likewise, $crit\_nOp$ cannot be improved when its value is the number of operational tasks. These are the maximum theoretical values of criteria.

Experimental results

In this section, we present experiments carried out on a set of data provided by scientists working on the design of ARIEL.

Scenario

The benchmark that we worked on includes 710 planets and 728 corresponding requests. 61 are *Benchmark*, 240 are *Deep* and 427 are *Survey* requests. 709 requests have a *Survey* objective. A short calibration is required before and after each observation of an event for 34 requests. The benchmark does not contain requests with additional observations.

There are 128439 candidate observations and 4345 observations are required to complete all of the requests. The corresponding duration is equal to 3.52 years, which is longer than the mission duration.

Implementation

The greedy approach was implemented with Scilab on a Intel Core i3 processor with 2GB of RAM. The CBLS experiments were run on a four-Xeon 2.80GHz processors with 8GB of RAM. We implemented the algorithm in Java on top of library InCELL [36].

Results

Results of the experiments are illustrated in Figure 7 and Table 4. For analyzing these results, we consider the various different objectives of this study.

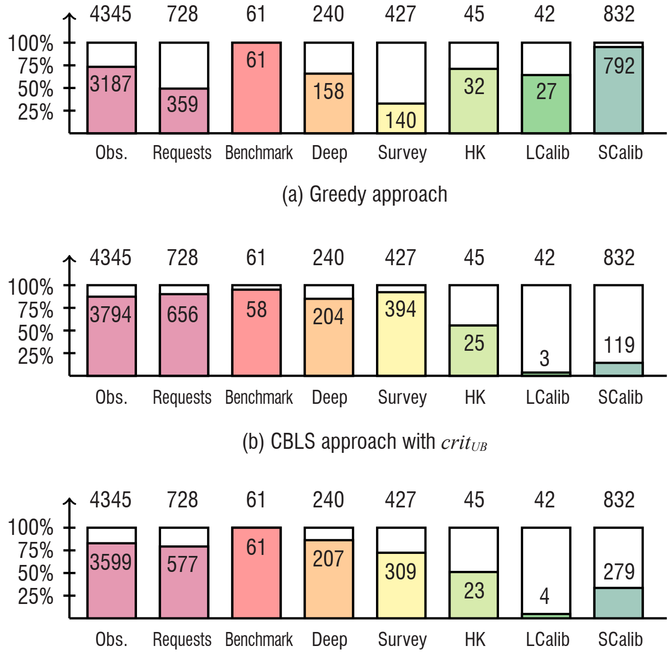
- *Number of completed requests* - As expected, $crit\_nReq$ is maximized with the CBLS approach along with the upper-bound criteria. When the types are taken into account, the best approach is CBLS with the criterion $crit_{Type}$.
- *Cumulated duration of activity* - All approaches generate a plan in which the satellite is active for more than 80% of the mission duration.
- *500 Survey objectives during the first year* - The best result is obtained by the CBLS approach with the criterion $crit_{Type}$. Better results might be hard to obtain, since the plan is saturated during the first year because of *Benchmark* requests.
- *Operational tasks* - All approaches have rather low results for that objective. The best approach is the greedy one: given that there is less time dedicated to observations, there is more time for operational tasks. If the insertion of operational tasks were to be a constraint instead of a preference, the overall results would really be lower. For instance, in this case the number of completed requests falls to 502 for CBLS with $crit_{Type}$ (44 *Benchmark*, 178 *Deep* and 280 *Survey*), and the cumulated activity duration of the satellite represents 74% of the mission duration. Given that this objective has a lower priority, it is not really achieved by the proposed approaches.

Lessons learned

The planning problem associated with the ARIEL mission stresses several challenges. First, it combines both an allocation problem for scientific observations and a scheduling problem for operational tasks, which implies the use of different types of decision variables. Second, there are several non-classical criteria, such as deadlines. Then, it requires requests covering multiple observations to be dealt with, also known as *linked observations*. Fourth, the number of requests can also be quite challenging. For the studied benchmark, there are more than 3 million potential conflicts between candidate observations. Finally, as the design of the mission was still on-going, there was a real need for a generic model and implementation.

Given that the ARIEL mission has been selected, the next step is to define more precisely the operational constraints on the system and make them high-priority. Moreover, the scheduler should be modified in order to dynamically integrate new sets of requests.

From a technical point of view, the initial plan produced maximizes the criteria but violates some constraints. While this outperforms the greedy approach along with several heuristics, some other approaches should also be investigated, such as CBLS starting from an empty plan or Iterated Local Search as mentioned earlier.



(a) Greedy approach

(b) CBLS approach with $crit_{UB}$

(c) CBLS approach with $crit_{Type}$

Figure 7 - Features of the plan generated by each approach. The first column contains the number of observations inserted into the plan, the second one contains the number of completed requests, and the three next columns detail how requests are satisfied per type. The last three columns contain the number of operational tasks that are part of the plan

Approach	*Activity dur.*	*Scientific activity dur.*	*Survey obj. 1st year*
Greedy	80%	72%	411
CBLS - $crit_{UB}$	89%	83%	295
CBLS - $crit_{Type}$	83%	77%	467

Table 4 - For each approach, the percentage of the mission duration during which the telescope is active, the percentage of the mission duration dedicated to scientific activities and the number of requests with *Survey* objective completed during the first year of the mission among the 709

Conclusion and future work directions

This article presented three telescope mission planning systems developed in the past, giving for each a description of the model, the algorithms, the results, and some lessons learned. Based on this experience, we believe that it may be relevant to address the three following challenges in the future.

A generic tool for space telescope planning

Given that many space telescopes share similarities in terms of planning, it would be useful to develop a generic mission manager. At the moment, we have a generic constraint-based optimization tool (InCELL) that allows us to quickly define new mission specific planners. To gain in genericity, we could try to define a generic telescope mission planning tool on top of InCELL, as done by the *Space Telescope Science Institute* (STScI) with Spike [22, 26]. In such a generic tool, we could have a domain specific language, a set of predefined decision rules (e.g., selection heuristics and insertion heuristics for greedy search), and a set of predefined local search and metaheuristics. On this basis, it would be possible to more easily compare several search algorithms (stochastic hill-climbing, iterated local search, min-conflicts, etc.). Also, the precise settings of the search parameters could be optimized by machine learning techniques.

Uncertainty management

Another challenge concerns the way in which the uncertainty is managed for space telescope missions. As seen previously, the degree of uncertainty is quite high for space telescopes, due to random events like ToOs or GRBs. The current practice is that in each mission specification, there is an indirect way of dealing with such events: limitation of the planned observation time during some periods, optimization of reschedulability measures, computation of flexible realization windows, etc. On this point, there is a need to compare the different approaches proposed in the literature. It is likely that it could be relevant to exploit a coarse-grain model of these random events and let the mission planner optimize the plans by using explicit stability measures in addition to the other performance measures. By doing so, the behavior of the telescope would be more predictable for the end-users. From an algorithmic point of view, the approach could be to search for an easily reschedulable backbone plan, and to add opportunistic observations when the occurrence rate of random events is lower than expected. Moreover, to manage uncertainty, it could be useful to develop on-board autonomy concepts, not only to automatically trigger follow-up observations when relevant events are detected, but also to abort observation activities when some conditions are not met.

Planning for several telescopes

Lastly, nowadays there is a significant number of space telescopes. Even if they do not embed the exact same instruments, it could be interesting to globally optimize their activities given a set of candidate targets, that is, to have a kind of centralized telescope planning tool or a kind of distributed planning engine with automated negotiation steps, at least at the level of each space agency to manage its "constellation" of space telescopes. Several reasons could motivate this choice. First, scientists might want to post an observation request r requiring several telescopes embedding complementary instruments. In this case, to maximize the scientific return, there would be a need to coordinate the decisions of the telescope mission planning centers, so that the elementary observations associated with r are either all selected or not selected, and so that these observations are carried out during similar periods of the year if needed. Another reason would be to share the use of the telescopes among the possible GRBs and ToOs. As an example, if a single random event is detected, it is not an issue to point all telescopes to the corresponding target. However, when several random events occur simultaneously, there could be some level of coordination between the mission centers to share the usage of the ground and space observatories because, in the end, all telescopes share common long-term goals ■

References

[1] A. ALLAHVERDI, C. T. NG, T. C. E. CHENG, M.Y. KOVALYOV - *A Survey of Scheduling Problems with Setup Times or Costs.* European Journal of Operational Research, 187(3):985-1032, 2008.

[2] G. BABIN, S. DENEAULT, G. LAPORTE - *Improvements to the or-opt Heuristic for the Symmetric Traveling Salesman Problem.* Journal of the Operational Research Society, 58:402-407, 2007.

[3] W. J. BORUCKI, D. KOCH, J. JENKINS, D. SASSELOV, R. GILLILAND, N. BATALHA, D.W. LATHAM, D. CALDWELL, G. BASRI, T. BROWN, *et al.* - *Kepler's Optical Phase Curve of the Exoplanet HAT-P-7b.* Science, 325(5941):709-709, 2009.

[4] J. BRUMFITT, P. GÓMEZ-ALVAREZ, A. VILLACORTA, P. GARCÍA-LARIO, R. LORENTE, L. O'ROURKE - *Herschel Mission Planning Software*. Proc. of the 7th International Workshop on Planning and Scheduling for Space (IWPSS'11), 2011.

[5] F. CASTELLINI, M. R. LAVAGNA - *Advanced Planning and Scheduling Initiative XMAS Tool: AI for Automatic Scheduling of XMM-Newton Long Term Plan.* Proc. of the 6th International Workshop on Planning and Scheduling for Space (IWPSS'09), 2009.

[6] A. CESTA, S. FRATINI - *The Timeline Representation Framework as a Planning and Scheduling Software Development Environment.* Proc. of the 27th Workshop of the UK Planning and Scheduling Special Interest Group (PlanSIG'08), 2008.

[7] A. CESTA, S. FRATINI, A. DONATI, H. OLIVEIRA, N. POLICELLA - *Rapid Prototyping of Planning and Scheduling Tools.* Proc. of the 3rd IEEE International Conference on Space Mission Challenges for Information Technology, 2009.

[8] J. COLOMÉ, P. COLOMER, J. GUARDIA, I. RIBAS, J. CAMPRECIÓS, T. COIFFARD, L. GESA, F. MARTINEZ, F. RODLER - *Research on Schedulers for Astronomical Observatories.* Proc. of SPIE, 2012.

[9] B. CORDIER, F. DESCLAUX, J. FOLIARD, S. SCHANNE - *SVOM Pointing Strategy: How to Optimize the Redshift Measurements?* AIP conference proceedings vol 1000, 2008.

[10] G. A. CROES - *A Method for Solving Traveling Salesman Problems.* Operations Research, 6:791-812, 1958.

[11] X. DUPAC, J. TAUBER - *Scanning Strategy for Mapping the Cosmic Microwave Background Anisotropies with Planck.* Astronomy and Astrophysics, 2005.

[12] A. GARCIA-PIQUER, I. RIBAS, AND J. COLOMÉ - *Artificial Intelligence for the EChO Mission Planning Tool.* Experimental Astronomy, 40(2-3):671-694, 2015.

[13] M. GIULIANO - *Achieving Stable Observing Schedules in an Unstable World.* Proc. of the 7th Conference on Astronomical Data Analysis Software and Systems (ADASS'97), 1997.

[14] M. GIULIANO - *The Mystery of the Missing Requirements: Optimization Metrics for Spike Long Range Planning.* Proc. of the 8th International Workshop on Planning and Scheduling for Space (IWPSS'13), 2013.

[15] M. E. GIULIANO, M. D. JOHNSTON - *Multi-Objective Evolutionary Algorithms for Scheduling the James Webb Space Telescope.* Proc. of the 18th International Conference on Automated Planning and Scheduling (ICAPS'08), 2008.

[16] P. GÓMEZ-ALVAREZ, J. BRUMFITT, R. LORENTE, P. GARCÍA-LARIO - *HILTS: The Herschel Inspector and Long-Term Scheduler.* Proc. of the 20th annual conference on Astronomical Data Analysis Software and Systems, 2011.

[17] P. VAN HENTENRYCK, L. MICHEL - *Constraint-Based Local Search.* The MIT Press, 2005.

[18] J. JAUBERT, V. DEBOUT - *Mission Planning and Scheduling Activities Conducted at French Space Agency (CNES) for the ATHENA Mission.* Proc. of the 11th International Workshop on Planning and Scheduling for Space (IWPSS'19), 2019.

[19] J. JAUBERT, V. DEBOUT - *Mission Planning and Scheduling Activities Conducted at French Space Agency (CNES) for the SVOM Mission.* Proc. of the 11th International Workshop on Planning and Scheduling for Space (IWPSS'19), 2019.

[20] M. D. JOHNSTON - *Spike: AI Scheduling for Hubble Space Telescope after 18 Months of Orbital Operations.* Proc. of the 6th Conference on Artificial Intelligence Applications, 1992.

[21] M. D. JOHNSTON, G.E. MILLER - *Artificial Intelligence Scheduling for NASA's Hubble Space Telescope.* Proc. of the 5th Conference on AI Systems, 1990.

[22] M. D. JOHNSTON, G.E. MILLER - *Spike: Intelligent Scheduling of Hubble Space Telescope Observations.* Intelligent Scheduling, pages 391-422. Morgan Kaufmann Publishers, 1994.

[23] S. KIRKPATRICK, C. GELATT, M. VECCHI - *Optimization by Simulated Annealing.* Science, 220, 1983.

[24] M. KITCHING, N. POLICELLA - *A Local Search Solution for the INTEGRAL Long Term Planning.* Proc. of the 7th International Workshop on Planning and Scheduling for Space (IWPSS'11), 2011.

[25] L. KRAMER, L. BARBULESCU, S. SMITH - *Understanding Performance Tradeoffs in Algorithms for Solving Oversubscribed Scheduling.* Proc. of the 22nd National Conference on Artificial Intelligence (AAAI-07), pages 1019-1024, 2007.

[26] L. A. KRAMER, M. E. GIULIANO - *Reasoning about and Scheduling Linked HST Observations with Spike.* Proc. of the 1st International Workshop on Planning and Scheduling for Space (IWPSS'97), 1997.

[27] H. R. LOURENÇO, O. C. MARTIN, T. STÜTZLE - *Handbook of Metaheuristics.* Chapter Iterated Local Search: Framework and Applications, pages 363-397. Springer, 2010.

[28] S. MINTON, M. D. JOHNSTON, A. B. PHILIPS, P. LAIRD - *Minimizing Conflicts: a Heuristic Repair Method for Constraint Satisfaction and Scheduling Problems.* Artificial Intelligence, 58(1-3):161-205, 1992.

[29] D. S. MITTMAN, R. HAWKINS - *Scheduling Spitzer: The SIRPASS Story.* Proc. of the 8th International Workshop on Planning and Scheduling for Space (IWPSS'13), 2013.

[30] J. C. MORALES, J.-P. BEAULIEU, V. COUDÉ DU FORESTO, M. OLLIVIER, I. ORTEGA CASTELLO, R. CLÉDASSOU, J. JAUBERT, P. VAN-TROOSTENBERGHE, R. VARLEY, I. P. WALDMANN, *et al. - Scheduling the EChO Survey with Known Exoplanets.* Experimental Astronomy, 40(2-3):655-670, 2015.

[31] OMITRON. Swift TAKO user guide, 2003.

[32] I. ORTEGA CASTELLÓ - *Means for Analysing the Operational Capabilities of Mission Programming.* Master's thesis, ISAE, September 2013.

[33] C. PRALET - *An Incomplete Constraint-Based System for Scheduling with Renewable Resources.* Proc. of the 23rd Conference on Principles and Practice of Constraint Programming (CP'17), 2017.

[34] C. PRALET, S. ROUSSEL, J. JAUBERT, J. QUEYREL - *Observation Planning for the SVOM Space Telescope.* Proc. of the 10th International Workshop on Planning and Scheduling for Space (IWPSS'17), 2017.

[35] C. PRALET, G. VERFAILLIE - *AIMS: A Tool for Long-Term Planning of the ESA INTEGRAL Mission.* Proc. of the 6th International Workshop on Planning and Scheduling for Space (IWPSS'09), 2009.

[36] C. PRALET, G. VERFAILLIE - *Dynamic Online Planning and Scheduling using a Static Invariant-Based Evaluation Model.* Proc. of the 23rd International Conference on Automated Planning and Scheduling (ICAPS'13), 2013.

[37] R. RAGER, M. GIULIANO - *Evaluating Scheduling Strategies for JWST Momentum Management.* Proc. of the 5th International Workshop on Planning and Scheduling for Space (IWPSS'06), 2006.

[38] S. ROUSSEL, C. PRALET, J. JAUBERT, J. QUEYREL, B. DUONG - *Planning the Observation of Exoplanets: the ARIEL Mission.* Proc. of the 10th International Workshop on Planning and Scheduling for Space (IWPSS'17), 2017.

[39] M. W. P. SAVELSBERGH - *Local Search in Routing Problems with Time Windows.* Annals of Operations Research, 4(1):285-305, 1985.

[40] D. SMITH - *Choosing Objectives in Over-Subscription Planning.* Proc. of the 14th International Conference on Automated Planning and Scheduling (ICAPS'04), pages 393-401, 2004.

[41] R. STEEL, M. NIEZETTE, A. CESTA, S. FRATINI, A. ODDI, G. CORTELLESSA, R. RASCONI, G. VERFAILLIE, C. PRALET, M. LAVAGNA, A. BRAMBILLA, F. CASTELLINI, A. DONATI, N. POLICELLA - *Advanced Planning and Scheduling Initiative: MrSpock AIMS for XMAS in the Space Domain.* Proc. of the IJCAI-09 Workshop on Artificial Intelligence in Space, 2009.

Cédric Pralet graduated from an engineering school specializing in the field of aeronautics and space in 2003 and achieved his PhD in Computer Science in 2006. Since then, he has been working at ONERA and obtained his authorization to supervise research ("Habilitation à Diriger les Recherches", HDR) in 2015. His research interests concern constraint-based optimization and automated planning and scheduling, applied to problems in the aerospace field.

Stéphanie Roussel graduated from an engineering school specializing in the field of aeronautics and space in 2007 and achieved her PhD in Computer Science in 2010. From 2010 to 2014, she worked as a research engineer at the CRIL (Centre de Recherche en Informatique de Lens). Since 2015, she has been working at ONERA, mainly on constraint-based optimization, knowledge representation and reasoning, and their application in the fields of aeronautics and aerospace.

Jean Jaubert graduated from engineering schools specializing in the fields of mechanics, aeronautics and space in 1989. Later, he joined the Centre National d'Etudes Spatiales (CNES), the French Space Agency, to work on flight dynamics and satellite guidance and control issues within the framework of space projects for Earth Observation applications. Then, he specialized in the design and implementation of mission planning and programming functions for various space missions in the fields of Earth Observation, Science and Astronomy.

Julien Queyrel received his spacecraft mechanics engineering diploma from ISAE-SUPAÉRO and his MSc in astrophysics from the University of Toulouse in 2007. He completed his studies in 2010 when he obtained his PhD in astrophysics. Afterwards, he worked as a subcontractor for the French space agency (CNES) on electronic intelligence and mission programming. Since 2018, he has been working at ONERA as a research engineer on the use of numerical weather forecast models to predict microwave propagation through the troposphere.