



Planning Acquisitions for an Ocean Global Surveillance Mission.

G. Verfaillie, X. Olive, C. Pralet, S. Rainjonneau, I. Sebbag

► **To cite this version:**

G. Verfaillie, X. Olive, C. Pralet, S. Rainjonneau, I. Sebbag. Planning Acquisitions for an Ocean Global Surveillance Mission.. International Symposium on Artificial Intelligence, Robotics, and Automation in Space (i-SAIRAS 2012), Sep 2012, TURIN, Italy.

HAL Id: hal-01061393

<https://hal-onera.archives-ouvertes.fr/hal-01061393>

Submitted on 5 Sep 2014

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

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

PLANNING ACQUISITIONS FOR AN OCEAN GLOBAL SURVEILLANCE MISSION

G. Verfaillie¹, X. Olive², C. Pralet¹, S. Rainjonneau², and I. Sebbag³

¹ONERA - The French Aerospace Lab, Toulouse, France

²TAS, Thales Alenia Space, Toulouse, France

³CNES, Toulouse, France

ABSTRACT

In this paper, we present the problem of planning the acquisitions performed by a constellation of radar satellites in order to fulfil as well as possible an ocean global surveillance mission. Then, we describe the local search algorithm, inspired by large neighborhood search techniques and knapsack heuristics, that has been specifically designed and implemented to solve daily planning problem instances. We conclude with experiment reports and directions for further improvements.

Key words: ocean surveillance; radar satellites; planning.

1. INTRODUCTION

In this paper, we report the results of a study that has been performed in 2010-2011 by ONERA and TAS for the French space agency (CNES) on mission planning for the SAMSON mission: an ocean global surveillance mission currently under study at CNES, whose goal is to track ship movements over all the oceans, using a constellation of radar satellites.

In [1], we already analyzed the planning problem and described two simple greedy algorithms that can produce feasible sub-optimal plans. In this paper, we improve on the first analysis and describe a more sophisticated local search algorithm, inspired by large neighborhood search techniques and knapsack heuristics, that can produce still sub-optimal, but better quality plans.

This paper focuses on the problem of planning radar acquisitions: when to set the radar instruments ON in order to collect data. The problem of allocation of downlink windows between satellites and ground stations, allowing acquisition data to be downloaded, is the subject of a companion paper [2]. As to the problem of planning data downloads within the allocated downlink windows, it is assumed to be managed using simple priority rules.

In Section 2, we describe the SAMSON mission, its objectives, as well as the physical system and the mission management system that are considered to fulfill them.

In Section 3, we describe the acquisition planning problem with its decision variables, constraints, and optimization criterion, as well as its various modes. In Section 4, we describe the planning algorithm we have specifically designed and implemented to deal with this problem. Results of experiments are reported in Section 5 and directions for further improvements are pointed out in Section 6.

2. THE SAMSON MISSION

The context of this work is the CNES SAMSON project, currently under study, whose objective is the design and the development of a combined space-ground system able to track ship movements over all the oceans.

2.1. Mission objectives

In fact, the mission objective is twofold: first, to visit every ocean area as often and as regularly as possible, using a wide swath acquisition mode (surveillance mission); second, to observe a small number of areas of special interest at specified times, using a narrow swath acquisition mode (observation mission).

To manage the surveillance mission, all the oceans are split into small 100km x 100km rectangle areas, resulting in about 37,000 areas to be visited every day. Because areas do not have all the same importance (for example, areas covering strategic ship roads may be more important than others), a weight is associated with each of them. For each area, the number of surveillance opportunities per day depends on the area latitude and on the chosen constellation. It is typically about 5 in average.

Areas of special interest are only known every day before planning for the next day. Some of them may arrive at the last minute. Their observation at specified times is compulsory. Their number is typically about 30.

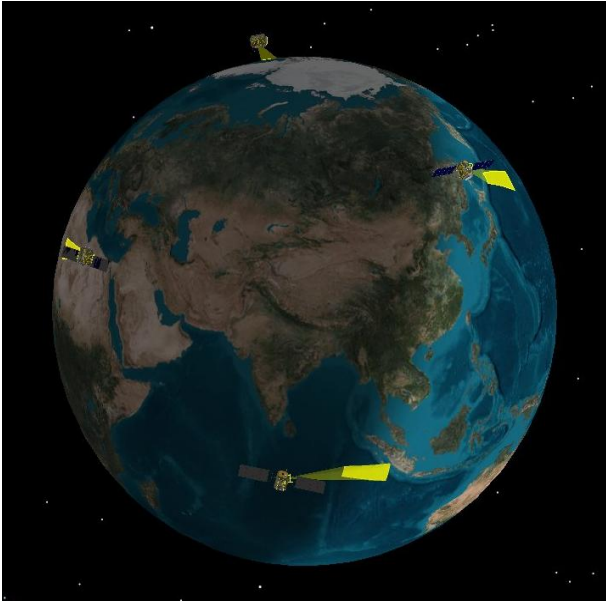


Figure 1. Four satellite constellation.

2.2. Physical system

Constellation To fulfil such a mission, it is planned to use a constellation of 4 to 6 satellites, placed on 2 to 3 orbital planes and on circular, low altitude, quasi-polar orbits. For each satellite, orbiting around Earth takes about one hour and a half.

Instrument modes Each satellite is equipped with a radar instrument usable in four different modes: a wide swath surveillance mode (SURV), two narrow swath high resolution observation modes (HR1 and HR2), and a stand-by mode (SB). Modes are incompatible with each other. Mode transitions are instantaneous. We will consider that the instrument is ON when it is in SURV, HR1, or HR2 mode and is OFF when it is in SB mode.

Each satellite is also equipped with other instruments able to collect signals. Because these instruments are permanently active, they are out of the scope of planning.

Surveillance When the instrument is in SURV mode, it scans a wide strip (about 1000km wide) on Earth surface, on the right of the satellite track, which covers concurrently a number of the areas that result from ocean splitting.

Figure 1 is an artist view of the four satellites with, for each of them, the scanning (in yellow) performed by the radar instrument on the right of the satellite track over a very short time period. Figure 2 shows the scanning performed by the radar instruments of the four satellites over a 45 minute period (a color is associated with each satellite of the constellation). Figure 3 focuses on one satellite

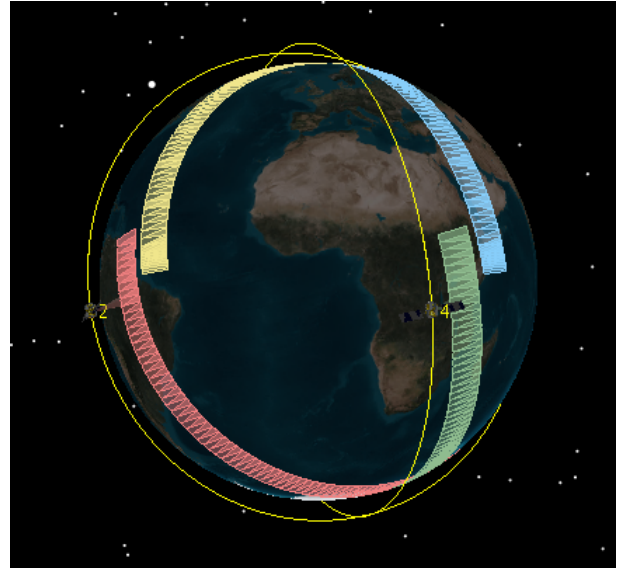


Figure 2. Scanning performed by the radar instruments of the four satellites over a 45 minute period.

and on the ocean splitting (each point on the ocean surface represents the center of an area). Finally, Figure 4 is a schematic view of the ocean splitting, of the satellite track, of the radar instrument swath, of an area, and of its associated acquisition window.

Observation and attitude movements In order to increase the number of observation opportunities in HR1 or HR2 mode, each satellite is able to perform attitude movements in order to observe on the left of the satellite track as well as on the right. No observation and no surveillance is possible during a satellite attitude movement, but the instrument can remain ON (in SURV mode) in order to limit the number of ON/OFFs. However, surveillance is by default performed on the right. As a consequence, in case of observation to be performed on the left, the required attitude movement from the right to the left is performed at the latest time just before observing, and the attitude movement from the left to the right is performed at the earliest time, just after observing, except when there is another observation to be performed on the left and not enough time to move to the right and to move back to the left. In such a case, surveillance can be performed on the left between both observations on the left.

Energy and memory Each instrument mode is characterized by an instantaneous consumption of energy and memory. On-board energy is produced by solar panels when the satellite is not in eclipse until a maximum level that depends on battery capacity. It must never be below a minimum level. On-board available mass memory is released by data downloads when the satellite is within a visibility window of a ground reception station. It must

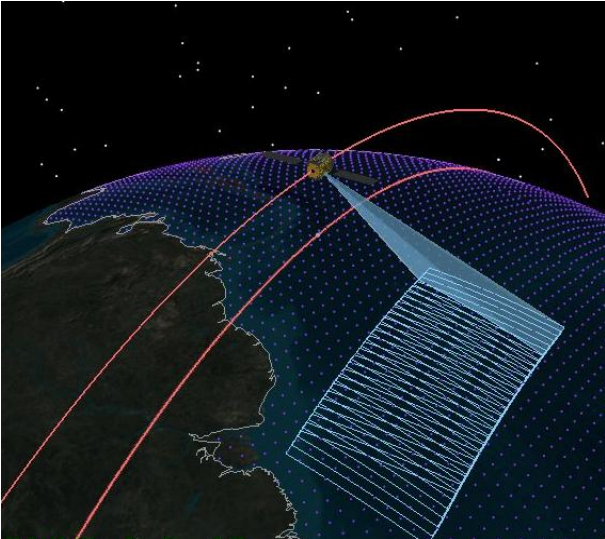


Figure 3. Ocean splitting and scanning performed by the radar instrument of one satellite over a short time period.

never be below 0 (memory overwriting). To be sure to meet these constraints, a maximum consumption of energy and memory per satellite orbit is enforced.

Instrument temperature Instrument temperature evolution depends on the fact that the instrument is ON or OFF and on the fact that the satellite is in eclipse or not. It cannot be below a minimum level and must never be above a maximum level.

Number of ON/OFFs For the sake of long-term reliability, the number of instrument ON/OFFs over the whole planning horizon (typically one day) is limited.

2.3. Management system

The constellation of satellites would be managed from the ground as follows: the requirements in terms of surveillance remain the same day after day; only the requirements in terms of observation change; each day, before planning for the next day, the compulsory observations are known; each one is characterized by a satellite, a starting time, a duration, a side (either right or left), and a mode (either HR1 or HR2); the problem is to organize as well as possible the surveillance mission while satisfying all the physical constraints and guaranteeing the execution of all the compulsory observations; when a executable plan has been built on the ground in the mission center, it is uploaded to all the satellites for execution.

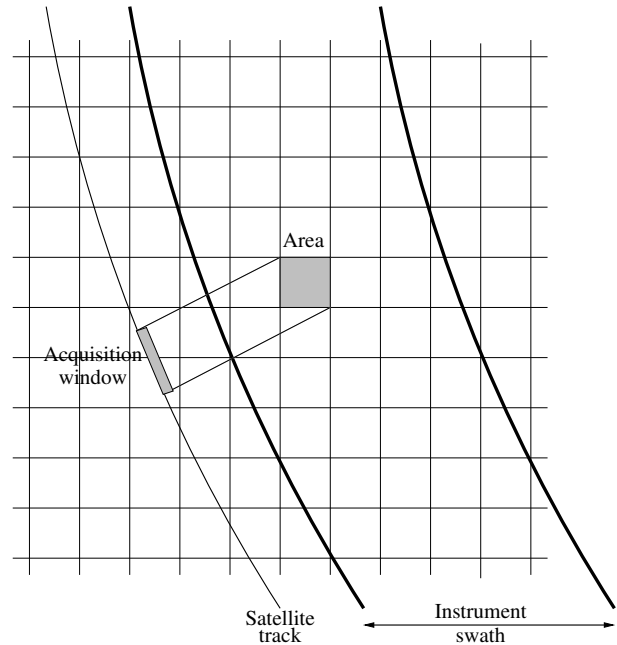


Figure 4. Ocean splitting, satellite track, radar instrument swath, area, and its associated acquisition window.

3. PLANNING PROBLEM

As said in the introduction, we focus on the problem of planning observation and surveillance activities and do not consider the problem of allocation of downlink windows to the satellites, as well as the problem of planning data downloads within these windows.

3.1. Data

Planning problem data is the following one:

- the static parameters of the physical system (for example, the duration of a satellite attitude movement or the maximum number of ON/OFFs);
- the planning horizon (for example, the next day);
- for each satellite:
 - its initial state (at the beginning of the planning horizon), including its eclipse status, its observation side, and its instrument mode and temperature;
 - the times of orbit change and of eclipse status change (over the planning horizon);
 - the set of downlink windows with, for each of them, its starting time and its duration;
 - the set of compulsory observations with, for each of them, its starting time, its duration, its side, and its mode;

- the set of surveillance opportunities with, for each of them, its area, its starting time, its duration, and its side.

3.2. Decisions

In classical Earth observation planning problems (see for example [3] for a tutorial and an associated commented bibliography), observations are mutually exclusive: two observations cannot be performed concurrently. In such conditions, it is relevant to associate with each candidate observation o decision variables that represent the fact that o is realized or not and, in case of realization, o 's realization parameters (time, mode ...).

On the contrary, in the planning problem we face, surveillance of different areas can be performed concurrently. In such conditions, it is more relevant to consider decision variables that represent satellite activity (time, mode, side ...).

Let us define an ON interval as an interval over which the instrument is ON, in HR1, HR2, or SURV mode. An ON interval may include attitude movements during which no acquisition is possible, but the instrument is maintained ON in SURV mode. An activity plan for the constellation takes, for each satellite, the form of a set of temporally ordered non overlapping ON intervals.

For each satellite, the number of ON intervals is limited by the maximum number of ON/OFFs over the planning horizon, typically some tens. Moreover the number and the size of the ON intervals is limited due to energy, memory, and temperature limitations. On the other hand, the more numerous and the larger ON intervals are, the better satisfied the surveillance mission is. However, it is useless to maintain the instrument ON when flying over continents and ON intervals must be set in such a way that ocean areas are visited as fairly and regularly as possible: for example, we do not want to have some areas visited many times and others not visited at all.

It must be finally stressed that the domains of value for starting and ending times of ON intervals are large, but discrete and finite, because it is useless to start an ON interval at some time that is not an acquisition starting time and to end it at some time that is not an acquisition ending time. As a consequence the domain of value for starting (resp. ending) times of ON intervals is the set of starting (resp. ending) times of the compulsory observations and surveillance opportunities.

3.3. Constraints

Three sets of constraints must be considered:

- model constraints which enforce that the ON intervals are on each satellite temporally ordered and non overlapping;

- physical constraints which enforce energy, memory, and temperature limitations;
- user constraints which enforce that compulsory observations are all covered by an ON interval.

All these constraints can be checked separately on each satellite. An acquisition plan is said to be consistent if and only if it satisfies all these constraints.

3.4. Optimization criterion

The criterion we propose evaluates the way the surveillance mission is fulfilled.

We associate with each ocean area a three notes, all of them between 0 and 1:

- an acquisition note na_a which measures the number of times a is acquired;
- a regularity note nr_a which measures the regularity of these acquisitions;
- a delivery note nd_a which measures the delay between area acquisitions and data deliveries on the ground.

This allows us to associate with each area a a note n_a which is the weighted sum of these three notes: $n_a = \alpha \cdot na_a + \beta \cdot nr_a + \gamma \cdot nd_a$, with α , β , and γ being parameters to be set, such that $\alpha + \beta + \gamma = 1$.

If W_a denotes the weight of a , the note n of a plan, that is the criterion we want to be optimized, is then the weighted normalized sum of the notes of all areas: $n = (\sum_a W_a \cdot n_a) / (\sum_a W_a)$.

Whereas constraints can be checked separately on each satellite, the criterion must be evaluated globally over all the satellites of the constellation.

3.5. Possible planning modes

As already said, the surveillance mission is always the same and the planning problem instances that must be solved each day only differ in the set of compulsory observations to be performed. In such conditions, at least three planning modes could be considered:

- a normal mode, in which an acquisition plan is built each day from scratch, taking into account the daily compulsory observations;
- a perturbation mode, in which a baseline acquisition plan has been built in advance for each day of the constellation cycle, taking into account no compulsory observation, and this plan is modified by inserting the daily compulsory observations;

- an urgent perturbation mode, in which an acquisition plan has been built, taking into account the daily compulsory observations, and this plan is modified by inserting last minute compulsory observations.

4. PLANNING ALGORITHM

We present the algorithm we designed to solve the acquisition planning problem in the normal planning mode. At the end of this section we will show how it can be slightly modified to work in the other two planning modes: perturbation and urgent perturbation.

The huge size of the search space (large domain of possible value for starting and ending times of ON intervals) prevented us from considering optimal algorithms, based on either tree search or dynamic programming. We already designed simple greedy algorithms, either chronological or non chronological (see [1]). To produce better quality plans, we decided to explore more powerful non chronological local search algorithms.

The choice of local search algorithms is justified by the fact that these algorithms, when sensibly designed and finely tuned, are known to be able to produce quickly high quality solutions to large combinatorial constrained optimization problems [4]. The kind of local search we chose, inspired from large neighbourhood search [5], alternates constructive phases where ON intervals are added to the plan and destructive ones where ON intervals are removed from the plan. In constructive phases, intervals are sequentially added by choosing at each step an interval whose addition has the greatest impact on the current plan. In the opposite direction, in destructive phases, intervals are sequentially removed by choosing at each step an interval whose removal has the smallest impact on the current plan. Inspired from the heuristics that are used to solve knapsack problems [6], the impact of an interval is evaluated by taking into account what it produces in terms of plan quality and what it consumes in terms of resources.

4.1. Several algorithm phases

Figure 5 shows how the different algorithm phases are linked together. Phase 0 is executed first. Then, Phases 1, 2, and 3 are sequentially executed several times, until some stopping condition is met at the end of Phase 2. Finally, Phase 4 is executed. The stopping criterion may be a deadline for plan delivery or an absence of significant plan quality improvement.

4.2. Phase 0

Phase 0 is an initialization phase where all the data necessary for planning is built: satellite initial states; orbit,

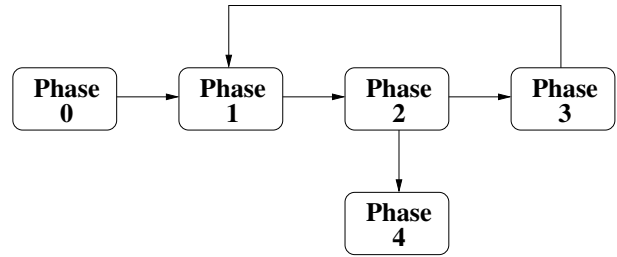


Figure 5. How the different algorithm phases are linked together.

eclipse, and downlink events; compulsory observations and attitude movements; surveillance opportunities ...

4.3. Phase 1

The first time Phase 1 is executed, its input is the set of compulsory observations and its output is a consistent acquisition plan that covers them, if such a plan exists.

To build such a plan, the algorithm builds on each satellite a sequence of ON intervals that covers exactly the compulsory observations and checks whether or not this plan satisfies all the constraints except the maximum number of ON/OFFs. If the maximum number of ON/OFFs is exceeded, it performs a systematic tree search over all the possible ON interval mergings to reduce the number of ON/OFFs, by trying first to merge close intervals. If this search succeeds on each satellite, Phase 1 ends with success. If it does succeed on at least one satellite, the algorithm stops because this means that it is not possible to perform all the compulsory observations and that some of them must be removed.

The next times Phase 1 is executed, its input is the acquisition plan that results from Phase 3. Due to Phase 3, it is guaranteed that this plan satisfies all the constraints except the maximum number of ON/OFFs. The same way, on each satellite, if the maximum number of ON/OFFs is exceeded, the algorithm performs a systematic tree search over all the possible ON interval mergings (see Figure 6). Still due to Phase 3, success is now guaranteed.

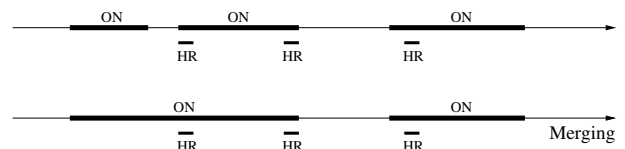


Figure 6. Merging movements in Phase 1.

4.4. Phase 2

The input of Phase 2 is the consistent plan produced by Phase 1. Phase 2 tries to improve on it by adding ON intervals, by extending existing ON intervals, or by merging two successive existing ON intervals, aiming at covering more surveillance opportunities, and thus at improving on plan quality. The output of Phase 2 is a consistent plan that extends the input plan.

To do this, the algorithm works in an incremental way. At each step, it adds an ON interval, extends an existing one, or merges two successive existing ones on one of the satellites (see Figure 7). The chosen added interval (resulting from pure addition, from extension, or from merging) is an interval whose addition is consistent (the resulting plan is consistent) and has the greatest impact on the current plan. Phase 2 ends when no ON interval can be added without violating constraints.

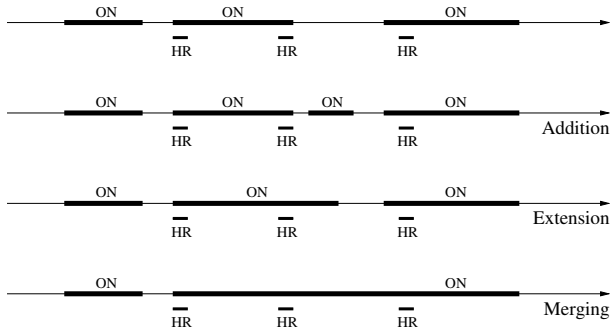


Figure 7. Addition, extension, and merging movements in Phase 2.

The impact I of adding an ON interval $[d, f]$ to the current plan can be evaluated as follows.

Let Δ^+U be the increase in plan quality resulting from adding interval $[d, f]$, $Umoy$ be the mean increase in plan quality resulting from adding any ON interval, and $DUmoy$ be the mean density of increase in plan quality resulting from adding any ON interval. Δ^+U can be computed, whereas $Umoy$ and $DUmoy$ can be evaluated several ways: on the basis of an inconsistent plan covering all the compulsory observations and all the surveillance opportunities that are compatible with them; on the basis of the plan resulting from the previous algorithm execution or, during algorithm execution, on the basis of the plan resulting from the previous execution of Phase 2.

In case of pure addition, we set $I = \Delta^+U - DUmoy \cdot (f - d) - Umoy$ (increase in plan quality, but consumption of duration $f - d$ and of one interval opportunity). In case of extension, we set $I = \Delta^+U - DUmoy \cdot (f - d)$ (increase in plan quality, but consumption of duration $f - d$). In case of merging, we set $I = \Delta^+U - DUmoy \cdot (f - d) + Umoy$ (increase in plan quality, but consumption of duration $f - d$ and release of one interval opportunity). This allows I to measure the impact of adding

an ON interval in terms of plan quality, in terms of duration consumed (to take into account energy, memory, and temperature limitations), and in terms of number of interval opportunities consumed (to take into account the maximum number of ON/OFFs). This criterion favours short intervals with high increase in plan quality and, everything otherwise equal, favours mergings with regard to extensions, and extensions with regard to additions.

Because it may be costly to consider all the possible starting and ending times when choosing the best interval to be added, it is possible to use a greater discretization step ΔT : two successive starting or ending times are considered only if they differ from at least ΔT . It is moreover possible to start the algorithm with a high value of ΔT and to decrease it regularly, at each execution of Phase 2, in order to get finer and finer acquisition plans.

4.5. Phase 3

The input of Phase 3 is the consistent plan produced by Phase 2. Phase 3 reduces it by removing, reducing, or splitting existing ON intervals, without removing compulsory observations. The output of Phase 3 is a plan that satisfies all the constraints except the maximum number of ON/OFFs, due to possible interval splittings.

As in Phase 2, the algorithm works in an incremental way. At each step, it removes, reduces, or splits an ON interval on one of the satellites. In case of reduction or splitting, the change is maximal: this means that it is not possible to remove more without removing compulsory observations (see Figure 8). The chosen removed interval (resulting from pure removal, from reduction, or from splitting) is an interval whose removal has the smallest impact on the current plan. Phase 3 ends when a given percentage P of the removable intervals has been removed.

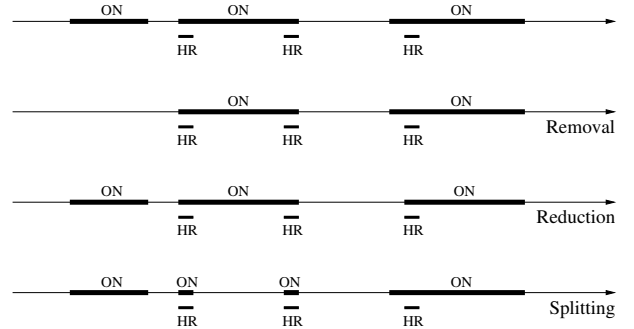


Figure 8. Removal, reduction, and splitting movements in Phase 3.

As in Phase 2, the impact I of removing an ON interval $[d, f]$ from the current plan can be evaluated as follows.

Let Δ^-U be the decrease in plan quality resulting from removing interval $[d, f]$. In case of pure removal, we set $I = \Delta^-U - DUmoy \cdot (f - d) - Umoy$ (decrease in plan

quality, but release of duration $f - d$ and of one interval opportunity). In case of reduction, we set $I = \Delta^-U - DUmoy \cdot (f - d)$ (decrease in plan quality, but release of duration $f - d$). In case of splitting, we set $I = \Delta^-U - DUmoy \cdot (f - d) + Umoy$ (decrease in plan quality, but release of duration $f - d$ and consumption of one interval opportunity). This criterion favours long intervals with small decrease in plan quality and, everything otherwise equal, favours removals with regard to reductions, and reductions with regard to splittings.

It must be stressed that the impact of removing an ON interval may differ from the impact of its previous addition because the background may differ: the impact of an ON interval may increase or decrease due to the other ON intervals present in the current plan. As a consequence, the intervals that are removed in Phase 3 are not necessarily the last ones that have been added in Phase 2.

4.6. Phase 4

The input of Phase 4 is the consistent acquisition plan produced by the last execution of Phase 2. Phase 4 adds download activities to this plan. To do this, it uses FIFO-based decision rules, with priority to observation data with regard to surveillance data.

4.7. Possible algorithm improvements

In local search, a key point is to be able to perform any local move very quickly in order to be able to perform as many moves as possible and thus to cover the search space as well as possible. To do this, the basic algorithmic scheme can be improved along several directions:

- in Phase 2, several rules can be defined to rule out inconsistent interval additions, that are intervals whose addition results in inconsistent plans;
- in Phase 2 too, the impact of an interval addition on the constraints and on the criterion can be incrementally computed; for example, evolution profiles of resources such as instrument temperature can be incrementally updated; the same way, in Phase 3, the impact of an interval removal on the criterion can be incrementally computed.

4.8. Perturbation planning modes

To deal with the perturbation and urgent perturbation planning modes, we propose to use the same algorithm, only slightly modified. In this modified algorithm, Phases 0 and 1 remain unchanged, taking into account all the compulsory observations (the previous and the new ones). The only modification concerns the first execution

of Phase 2, referred to as Phase 2.0. In this phase, the acquisition plan that has been previously built is reused as much as possible by considering the ON intervals present in the previous plan as the only candidate intervals for addition. The next executions of Phase 2, as well as Phases 3 and 4, remain unchanged (see Figure 9).

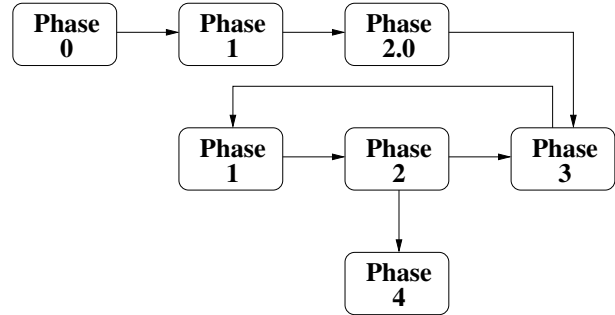


Figure 9. How the different algorithm phases are linked together in the perturbation and urgent perturbation planning modes.

5. EXPERIMENTS

Experiments have been carried out on scenarios produced by CNES. The planning horizon was of one day. Some algorithm parameters were set after some trials: 250 seconds for ΔT (the discretization step used for the choice of starting or ending times of added ON intervals in Phase 2), 10% for P (the percentage of ON intervals removed in Phase 3), updating of $Umoy$ and $DUmoy$ (the mean increase and the mean density of increase in plan quality that results from adding any ON interval) after each execution of Phase 2.

From these experiments, it results that, with the current implementation:

- Phase 0 takes about 2 minutes;
- Phase 1 takes about 1 minute, but may take more than 10 minutes in rare cases, when backtracks in tree search are required;
- Phase 2 takes less than 1 minute, except for the first execution which may take more time;
- Phase 3 takes less than 1 minute.

It has been also observed an improvement in plan quality after each execution of Phase 2. The number of such executions depends on the computing power and time available for mission planning.

6. CONCLUSION

The first output of this study performed by ONERA and TAS is that planning is possible and compatible with the available time for such an ocean global surveillance mission, in spite of the huge dimension of the solution space to be explored. The second output is that local search is a sensible algorithmic option, with first good quality results and important improvement margins.

It must be stressed that the algorithmic scheme that has been developed depends on the structure of plans (sets of temporally ordered non overlapping intervals) and on the constraints to be satisfied, but is completely independent of the form of the criterion to be optimized. The only requirement is to be able to compute quantities such as Δ^+U or Δ^-U . This allows the final users to choose any optimization criterion.

Beyond the algorithmic improvements pointed out in Section 4.7, two longer term research directions would deserve to be explored:

- the use of machine learning mechanisms to set automatically the various algorithm parameters [7, 8, 9], either for all the instances, or for each class of instances, or for each individual instance, in order to get quickly efficient parameter settings;
- the development of generic parameterizable algorithms able to solve efficiently planning problems that involve intervals, temporal constraints, complex resource constraints, and complex optimization criteria, in order to avoid developing specific algorithms for each particular planning problem.

REFERENCES

- [1] C. Pralet, G. Verfaillie, X. Olive, S. Rainjonneau, and I. Sebbag. Planning for an Ocean Global Surveillance Mission. In *Proc. of the 7th International Workshop on Planning and Scheduling for Space (IWSPSS-11)*, Darmstadt, Germany, 2011.
- [2] C. Pralet, G. Verfaillie, X. Olive, S. Rainjonneau, and I. Sebbag. Allocation of Downlink Windows for a Constellation of Satellites. In *Proc. of the 11th International Symposium on Artificial Intelligence, Robotics, and Automation for Space (i-SAIRAS-12)*, Turin, Italy, 2012.
- [3] G. Verfaillie and M. Lemaître. Planning Activities for Earth Watching and Observation Satellites and Constellations: From Off-line Ground Planning to On-line On-board Planning. 2006. Tutorial given at ICAPS-06 (16th International Conference on Automated Planning and Scheduling, Cumbria, UK), www.plg.inf.uc3m.es/icaps06/preprints/i06-tu5-allpapers.pdf.
- [4] E. Aarts and J. Lenstra, editors. *Local Search in Combinatorial Optimization*. John Wiley & Sons, 1997.
- [5] P. Shaw. Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems. In *Proc. of the 4th International Conference on Principles and Practice of Constraint Programming (CP-98)*, pages 417–431, Pisa, Italia, 1998.
- [6] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack Problems*. Springer, 2004.
- [7] F. Hutter, H. Hoos, K. Leyton-Brown, and T. Stütze. ParamILS: An Automatic Algorithm Configuration Framework. *Journal of Artificial Intelligence Research*, 36:267–306, 2009.
- [8] S. Kadioglu, Y. Malitsky, M. Sellmann, and K. Tierney. ISAC - Instance-Specific Algorithm Configuration. In *Proc. of the 19th European Conference on Artificial Intelligence (ECAI-10)*, pages 751–756, Lisbon, Portugal, 2010.
- [9] H. Hoos. Programming by Optimization. *Communications of the ACM*, 55(2):70–80, 2012.