An AI Tool for Planning Satellite Nominal Operations

María Dolores Rodríguez-Moreno [1], Daniel Borrajo [2], Daniel Meziat [1]

[1] Departamento de Automática. Universidad de Alcalá.
Carretera Madrid-Barcelona, Km. 33,600. 28871 Alcalá de Henares (Madrid), Spain.
{mdolores, meziat}@aut.uah.es

[2] Departamento de Informática. Universidad Carlos III de Madrid.
Avda. de la Universidad, 30. 28911 Leganés (Madrid), Spain.
dborrajo@ia.uc3m.es

Abstract

Satellite domains are becoming a fashionable area of research within the AI community due to the complexity of the problems that these domains need to solve. With the current US and European focus on launching satellites for communication, broadcasting, or localization tasks, among others, the automatic control of these machines becomes an important problem. Many new techniques in both planning and scheduling have been discovered and applied successfully but still much work is left to be done for reliable autonomous architectures.

The purpose of the paper is to give an overview of a software already developed that allows to plan and schedule nominal operations to perform in three satellites along the year for a commercial Spanish satellites company. We have used an AI domain independent planner for this task. The tool is currently in the user validation phase.

1 Introduction

Planning and scheduling are two areas with many points in common. Deliberative planners embody powerful techniques for reasoning about actions and their effects. They are good at finding precedences among activities but quite limited at resource or time reasoning. Scheduling systems are good at optimising and assigning time and resources to activities, but they require to know order relations among the activities. A current approach to solve this problem, is to use a scheduler at the output of the planner in order to assign resources and time to each activity [?].

Depending on the problem complexity some domains allow a strict separation between planning and scheduling. But, in other cases, there is an indirect temporal and resource dependency with other states and goals that can not be taken into account if we separate both tasks. The above approach is weak if the scheduler fails to find a solution: expensive and unsatisfactory solutions can be again generated by the planner if it does not receive any feedback from the scheduler.

A more reasonable approach is to integrate planning and scheduling in the same reasoning tool [?, ?]. We have explored this possibility, using a standard non-linear domain independent planner. We present here an AI tool that uses this approach for planning and scheduling maintenance operations over a set of satellites currently in orbit.

The paper is structured as follows. Next section introduces the satellites type of operations that have to be performed during the year. Then, the tool is presented. Finally, future work and conclusions are drawn.

2 Type of satellite operations

HISPASAT is a Spanish multi-mission system in charge of satisfying national communication needs. It also supplies capacity for digital TV in Europe, America and North Africa, TV image, radio and other signals, and special communications for defence purposes. It is the first European satellites system to offer transatlantic capacity that allows simultaneous coverage between South and North America.

Every operation in orbit must have explicit engineering instructions. These instructions provide a guide for technicians to consider the work accomplished. Currently, the engineer group generates this documentation for every satellite and year *by hand* and *on paper*. Later, this documentation is revised, modified and verified.

Three categories of planned operations can be identified, according to the flexibility to schedule them in the HISPASAT domain:

- Operations which are driven by external events and shall start or be completed at a fixed time. Examples are Moon Blindings, Sun Blindings or Eclipses of the Sun by the Earth or by the Moon. We have represented these hard constraints as pre-conditions of the operators.

- Operations that are part of a given strategy and should be performed at specific dates. However, some flexibility exists to modify the strategy. This is the case of Manoeuvres, Localisation Campaigns or Batteries Reconditioning. Although they are not hard constraints, we have coded functions in the pre-conditions to guide the planner for preferring some dates over others having in mind the restrictions imposed in some pre-conditions.

- Operations that should be performed within a wide period of time, and a significant flexibility exists for scheduling them as in Steerable Antenna Maintenance. These are the softest constraints and they are represented as control rules for preferring some dates over others.

In the type of operations that belong to the first category, the external (i.e. eclipses and blindings) and seasonal (i.e. solstice and equinox) events are foreseen several weeks before the year starts. All these events are represented in the initial conditions of the problem. In the case of external events, one needs to include the affected satellite, the start and end times, and for the seasonal events just the start time. Examples of these operations are CONF-ADCS, performed during the sun blinding periods, or SASS DATA, performed in solstices and equinoxes. As goals, we have to define the number of times that the sun blindings occur during the year (four per year) or the possible modes: summer, winter, spring and autumn for SASS DATA. Then the planner generates the appropriate satellite operations (for more details on how the planner solves the problem go to [**?**]).

Most of the operations in the second category (and some in the third) are scheduled having in mind the date when a South Manoeuver was performed. They are performed every two weeks at an hour corresponding to the secular drift direction. In the initial conditions we have the satellite affected by the secular drift, and the date and time in the year during which the Manoeuvre is going to be performed. Given that the maneuvers are forbidden during moon or sun blindings they have to be moved ahead twenty-four or forty eight-hours (in case two moon blindings appear in following days) from the secular drift time. The two maneuvers that follow the first one must be also moved ahead the same number of hours.

The rest of the operations in this category start/finish some hours before/after the start/end of the South Maneuver. Some operations cannot be performed if in that week a Maneuver has been scheduled, while others must be performed during Maneuvers. Others just have to be performed N days/weeks/months since the last operation.

3 The tool

In this section an overview of a graphical modelling and validation tool developed for scheduling the nominal operations for in orbit control of HISPASAT's satellites is presented.

Basically the tool consists of the following subsystems:

- **The user subsystem:** is in charge of the control access to the tool and the interaction with the user in order to obtain and manipulate all the data needed for the scheduling.

- **The reasoner subsystem:** once the input data is manipulated, a domain independent planner is in charge of generating the solution to the problem.

- **The generator susbsystem:** is responsible for maintaining coherence between the two possible representations (annual and weekly) obtained by the planner and that the user can modify as wished. This subsystem also allows to convert the solution to a specific format document type, compare two different solutions, or generate an HTML version.

Figure **??** shows a high level view of the tool, and the modules that it comprises.

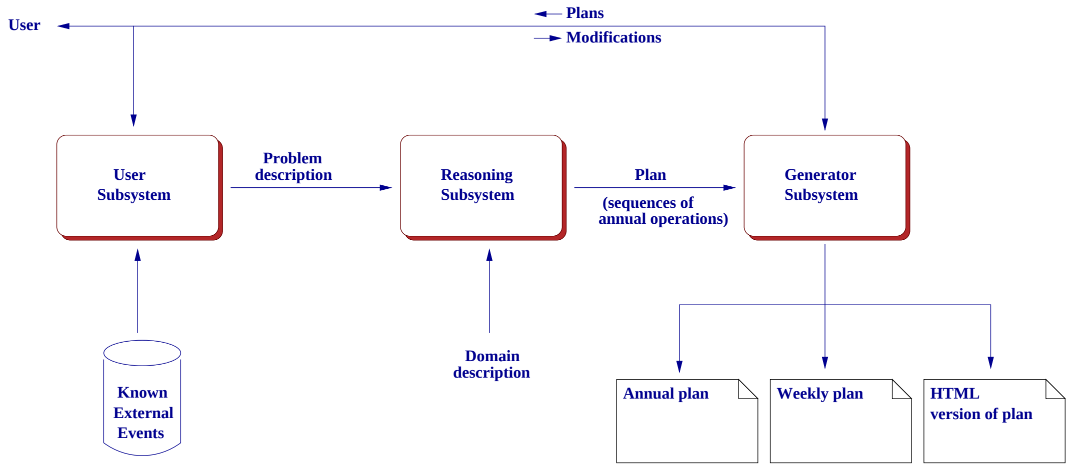

Figure 1: Architecture of the planning tool.

3.1 The User Subsystem

The external events that drive the operations of the first type, are known in advanced by means of specific software that predicts according to some parameters when the events should occur. This software generates the data in a set of ASCII files. Nowadays, the engineers represent this data in an specific document and schedule the operations according to them. Every time a new modification is done, the engineer in charge of it must sign the document. The operations belonging to the third type of operations require to know the last date, so the engineers must find in the last year's document when these operations were performed.

The User Subsystem controls the user access to register who is creating, revising, modifying or verifying the schedule. It also imports files in different formats from other supporting engineer tools, loads the data of the last operations if the tool has been used to generate the previous year schedule, and assists the user in validating the data introduced. This allows to avoid the possible user inconsistencies, such as trying to swap twice the same tank in the same period of the year or perform an operation in an illegal period[1].

3.2 The Reasoner Subsystem

The User Subsystem translates all this information into a suitable format for input to the AI planner. The planner is in charge of all the plan generation, with temporal and resource reasoning. Although in the planner there is not a language that allows an explicit representation of resources or temporal information, it is able to handle the problem thanks to its capability to: represent infinite types (numeric variables); define functions that obtain variables values in preconditions of operators; and use of control rules to prune the search (for more information about how the planner works the reader should refer to [?]).

[1] A legal period in "tank swapping" is a date around equinox.

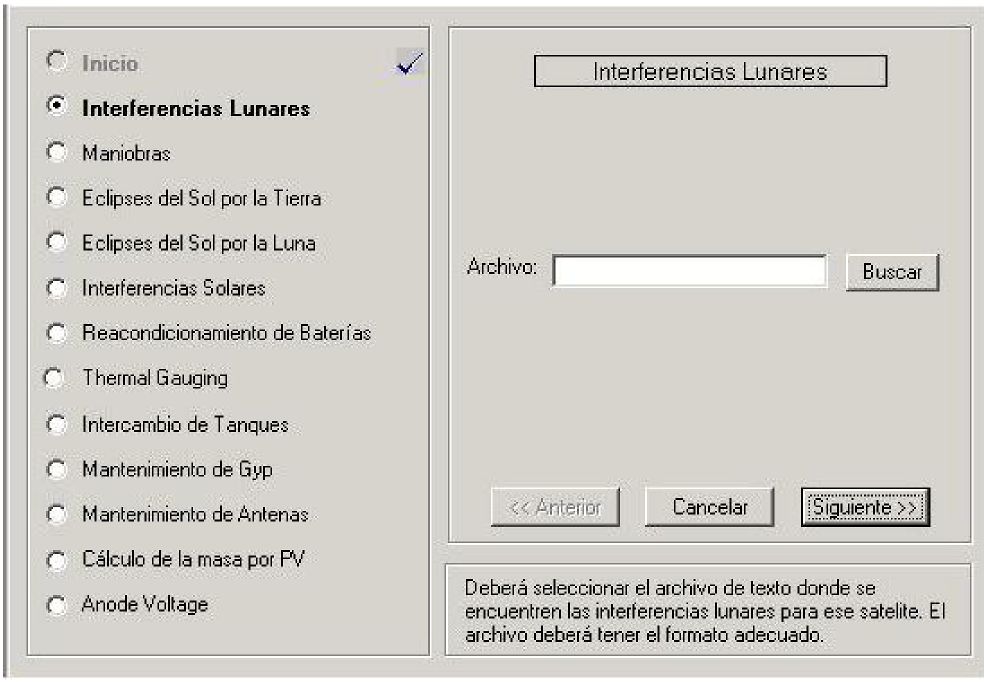

Figure 2: The User Subsystem interface.

3.3 The Generator Subsystem

Currently, once the engineers know every external event and have represented them in a document, the laborious task of scheduling every operation starts. They generate two types of documents:

- A document with gives an overview of all the operations performed each day of the year; and
- A document that represents in more detail each operation duration, type of manoeuvre, the resources, if any, affected in the operation as batteries or tanks, etc.

The weekly representation is generated every week having in mind the annual representation. The problem of maintaining two documents has to do with the incongruencies between them. Also, these documents are generated by the engineers of the headquarters at Arganda (Madrid) and sent to other backup centers in Spain and South America. In case any problem arises at the headquarters, one of those other backup centers must continue performing the scheduled operations.

The Generator Subsystem guarantees the consistency of the two representations allowing to easily modify the results obtained by the planner by just dragging and dropping the symbols in the table of the annual or weekly representations. It also allows to: compare two solutions, showing the differences between them; convert the results to the document format that the engineers use in its daily operation; and generate the solution in HTML for visualisation at the other centers.

4 Conclusions and Future Work

In this paper, we have presented our work on developing a tool that fully integrate planning and scheduling for the nominal operations to perform in three satellites along the year for a commercial satellite company. Before the software development, the engineer group generated by hand and in paper the operations that have to be done along the week and year. There were many incongruencies between the two types of representation used (weekly and annual) and the document modification was a tedious task.

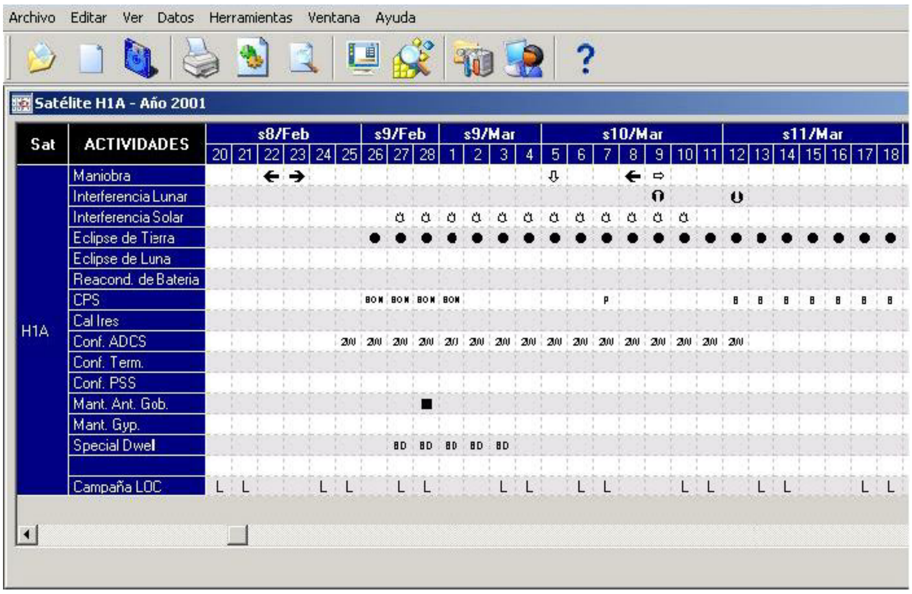

Figure 3: The Generator Subsystem interface.

The tool saves a lot of time to the user due to its capability of importing files of any type, representing the results in a table that allows easily to modify them by just drag and drop, generate more than one solution, see the differences between any two solutions, and generate the result in its internal format or in HTML. The tool is currently under the step of user validation.

We want to explore in the future the possibility of adding more satellites allowing to define/delete new operations. This would allow us to study the scalability of the approach of dealing within the planner with planning, temporal and resource reasoning.

It would be also interesting to integrate a monitoring module to execute the scheduled operations obtained by the planner and to re-plan in case of a failure.

Acknowledgements

We want to thank all the HISPASAT Engineering Team for their help and collaboration shown during the developing time of this project, especially Arseliano Vega, Pedro Luis Molinero and Jose Luis Novillo. This work was partially funded by the CICYT project TAP1999-0535-C02-02.

References

[1] Ghallab, M., and Laruelle, H. Representation and control in ixtet, a temporal planner. In *Proceedings of the 2nd International Conference on AI Planning Systems* (1994).

[2] Muscettola, N. HSTS: Integrating Planning and Scheduling. In *Intelligent Scheduling, ed. M. Zweben and M.S. Fox. pp 169-212. Morgan Kaufmann* (1994).

[3] Muscettola N. and Smith B. On-Board Planning for New Millenium Deep Space One Autonomy. *Procs. of IEEE Aerospace Conference, Snowmass, CO.* (1997).

[4] Rodríguez-Moreno, M. D., Borrajo, D., and Meziat, D. An artificial intelligence planner for satellites operations. In *ESA Workshop on On-Board Autonomy* (AG Noordwijk (The Netherlands), October 2001), European Space Agency, pp. 233–240.

[5] Veloso, M., Carbonell, J., Pérez, A., Borrajo, D., Fink, E., and Blythe, J. Integrating planning and learning: The prodigy architecture. *Journal of Experimental and Theoretical AI 7* (1995), 81–120.