

Paper:

Deploying Interactive Mission Planning Tools – Experiences and Lessons Learned –

Amedeo Cesta, Gabriella Cortellessa, Simone Fratini,
Angelo Oddi, and Giulio Bernardi

Consiglio Nazionale delle Ricerche, Istituto di Scienze e Tecnologie della Cognizione

Via S. Martino della Battaglia 44, I-00185 Rome, Italy

E-mail: name.surname@istc.cnr.it

[Received May 25, 2011; accepted August 22, 2011]

This article contains a retrospective overview of connected work performed for the European Space Agency (ESA) over a span of 10 years. We have been creating and refining an AI approach to problem solving and have infused a series of deployed planning and scheduling systems which have innovated the agency's mission planning practice. The goal of this paper is to identify strong features of this experience, comment on general lessons learned and offer guidelines for work practice of the future. Specifically, the work considers some key points that have contributed to strengthening the effectiveness of our approach for the development of an end-to-end methodology to field applications: the attention to domain modeling, the constraint-based algorithm synthesis and the relevance of user interaction services.

Keywords: mission ground segment, end-to-end development, timeline-based planning, AI planning and scheduling, constraint-based reasoning

1. Introduction

Our research group has developed a continuous collaboration with the European Space Agency (ESA) over a span of ten years. The topic of this collaboration has been the injection of ideas from the AI planning and scheduling research area to support operations of real missions. As summarized in Fig. 1, we have performed activities of different kinds; in particular, it illustrates the sequence of systems developed for ESA:

- In “product driven” activities we have developed solutions for specific mission problems. For example, in a fruitful collaboration with the Mission Planning Team of the MARS-EXPRESS mission at the European Space Operations Centre (ESOC), we have addressed specific problems connected with data/command management for the remote spacecraft. This effort resulted in two operational prototypes currently in daily use at ESA-ESOC. In particular, MEXAR2 supports the data downlink problem

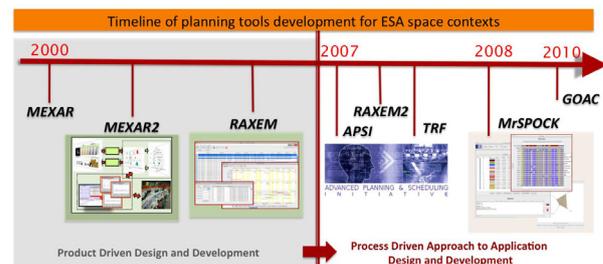


Fig. 1. A summary timeline showing the work for ESA.

of the MARS-EXPRESS memory [1], while RAXEM [2] and its strengthened version RAXEM2 [3], resolve the complementary problem of the uplink of operational commands to the probe. Such an approach inevitably entails a huge implementation effort in terms of development: specification extraction, design, coding, maintenance.

- In “process driven” activities we have attempted to develop general purpose tools for facilitating the design and synthesis of new products. The general pursued idea is the one of improving the “process” of tool development, taking advantage of the state of the art planning and scheduling technology.

In particular, within a larger project that spanned from 2007 to 2009, called the Advanced Planning and Scheduling Initiative (APSI), we have developed the Timeline-based Representation Framework (APSI-TRF) [4]. The aim of APSI is to develop a general framework to improve the cost-effectiveness and flexibility of the Mission Planning Systems (MPS) development. The tool offers a Java platform with primitives to capture the specifications of an application domain and a given problem, thus fostering rapid and fast prototyping. An example of such support is represented by MrSPOCK [5], an operational product for MARS-EXPRESS Long-Term Planning (LTP). A new example of a “process driven” approach is the GOAC project, an ongoing activity for designing a future generation of robotic controllers [6].

This paper aims at presenting a comprehensive picture of these activities and an attempt at drawing general points that emerge from the work done. In revising our collaboration with ESA, we focus also on the lessons learned and the open challenges. An inspiring paper [7] makes a similar effort to gather a number of lessons learned while deploying automated planning and scheduling systems for space applications at the Jet Propulsion Laboratory. Similar to that work, we here report on our experience and try to give an updated vision of the open challenges and achievements of AI based solutions for space contexts. One specific aim we have is to highlight our steps toward the consolidation of a robust methodology for fast development of ground segment mission planning support tools. In particular, we describe the transition from a “product driven” approach to a way of working based on a general open framework, which is “process driven” and hopefully more general and structured.

The paper is organized as follows: Section 2 proposes a brief introduction of all the deployed software (both specific products and enabling software platforms) and creates the context for the critical discussion. Section 3 presents the main lessons learned and suggests some general challenges and issues that are recurrent when designing applications for space contexts. Some conclusive remarks end the paper.

2. Deployed Work

An opportunity for a fertile collaboration with ESA was given by the MARS-EXPRESS probe launched by ESA in June 2003 and prolonged until December 2012. MARS-EXPRESS created the opportunity for providing spectacular images of the Red Planet and other valuable data such as high-resolution photo-geology, mineralogical mapping, mapping of the atmospheric composition, study of the subsurface structure, the global atmospheric circulation, as well as the interaction between the atmosphere and the subsurface and between the atmosphere and the interplanetary medium.

The mission is the first ESA program that uses a rapid prototyping design and implements concepts of re-using modules between different missions to shorten both development time and costs. Given the relatively short design time, the actual operational phase presented some problems that were seriously able to prevent a full exploitation of the mission activities. For example, the on-board memory downlink problem was underestimated before launch and actually impacted the scientific data return. In this scenario our work was commissioned by ESA to solve different subproblems identified within the mission planning management. This effort led to the development of several prototypes that quickly were turned in operational modules given their effectiveness. They are now operating in various parts of the mission planning cycle.

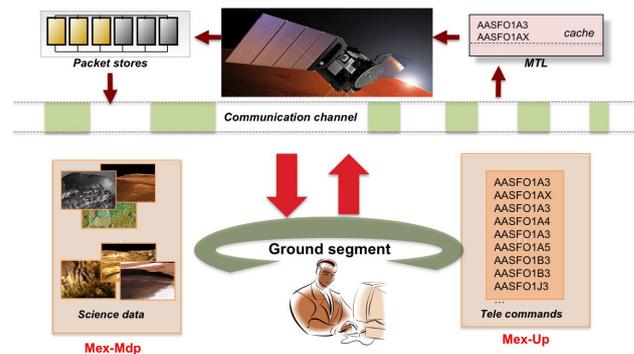


Fig. 2. The MARS-EXPRESS memory downlink and command uplink problems.

2.1. MEXAR and MEXAR2

MEXAR2 is an advanced software tool based on the interactive problem solving architecture deployed to ESA-ESOC during 2004 and which has been operational on a daily basis since beginning of 2005. MEXAR2 is devoted to the support of data dumping activities for MARS-EXPRESS, and it is an operational version of a prototype named MEXAR, synthesized during a previous study, 2001 to mid 2002. Both of the tools solve the MARS-EXPRESS Memory Downlink Problem (MEX-MDP).

2.1.1. The Memory Downlink Problem

The MEX-MDP problem focuses on an important aspect within the overall process needed to manage science data return (see Fig. 2, left part). The space-probe continuously produces a large amount of data resulting from the activities of its payloads and from on-board device monitoring and verification tasks. All of this data is to be transferred to Earth during bounded downlink sessions. On-board data generally requires to be first stored in on-board memory and then transferred to Earth. The on-board memory is subdivided into different banks, called *packet stores*, in which both scientific (*science* from payloads) and spacecraft management data (*housekeeping*) can be stored. Each packet store assigned to science data is managed cyclically, so in the case that new data is produced before the previous is dumped to Earth, older data is overwritten.

The main problem to be solved consists of synthesizing sequences of spacecraft operations (dump plans) that are necessary to deliver the content of the on-board memory during the available downlink windows.

The most important criticality of the problem was the *uncertainty* in the data production. In particular, due to differences in compression algorithms, the amount of data produced by some of the activities is not exactly predictable in advance, creating an occasional need to quickly recompute the short-term plan before uplinking the commands. The previous management of the problem ended up being very time consuming because mission planners were essentially proceeding by “trial and error”

on a textual plan. In addition, the activity of “manual” refinement was possible for a reference period of one or two days but was cognitively very difficult on a larger horizon. Overall this planning process too often produced dump plans not sufficiently reliable with respect to *overwriting* and also entailed a significant amount of time spent on repetitive and tedious low level tasks.

2.1.2. The MEXAR and MEXAR2 Solution

Three years before operations, we had the opportunity to study the memory downlink problem of MARS-EXPRESS. This previous experience produced MEXAR [8], a software demonstrator able to solve a core problem close to the current one and started a trustful interaction with the mission planners. The prototype was never tested on real data because it was not available at that time, but different algorithms were developed based on a CSP approach [9]. In June 2004 when the mission was operational, the daily dump plan synthesis at ESA was so critical, and time consuming for humans, that we received a detailed description of the real data format and have been entrusted with the task of possibly improving the work practice at that time.

Since February 2005 an interactive AI tool, named MEXAR2, is entered in the daily loop of mission practice. An AI based solver finds an initial solution and allows users to influence the solving process by manipulating “modifiers” (input parameters), and preserves users’ responsibility on the choices [1]. In addition, the tool offers to the mission planners a set of additional statistics and information on the current solution and the possibility to obtain and compare different solutions. MEXAR2 has significantly increased the science data return and reduced the mission planning team’s workload by half, compared to the previous method. The tool has also contributed reducing the mission costs thanks to better management of the ground stations’ reservations [10].

2.2. RAXEM and RAXEM2

RAXEM and RAXEM2 are subsequent versions of a software tool deployed to ESA-ESOC designed to help mission planners in managing the uplink of tele-commands for the MARS-EXPRESS (see Fig. 2, right part).

2.2.1. The Memory Uplink Problem

The MARS-EXPRESS spacecraft is not able to plan and execute science operations in a fully autonomous way; hence, its plans arrive from the ground on a continuous basis. A particular on-board memory block, called Mission TimeLine (MTL), is replenished by uploading tele-commands (TCs) from the ground (see Fig. 2, right part).

The spacecraft activities for each month are determined in accordance with the Medium-Term Plan (MTP) for the concerned period (typically four weeks). Based on the given MTP, various *Operations Requests* (OR) are generated. During the daily planning activity, the ORs are converted into *MTL Detailed Agenda Files* (M_{DAF}) by the

Mission Planning System; the M_{DAF} is an expansion of these ORs into time-tagged tele-commands, which are transferred to the spacecraft. On-board the spacecraft the TCs reside in the MTL buffer ordered by execution time. At the specified execution time, each TC will be released and removed from the MTL. Two constraints make this problem hard: the limited bandwidth of the transmission channel and the finite capacity of the on-board memory (MTL), where the commands have to be stored, waiting for the execution.

The memory uplink problem (MEX-UP) consists in producing an uplink plan for the set of M_{DAFs}, considering the available uplink windows, the status of the MTL, and the priority of each M_{DAF}. For each M_{DAF} the plan should retain a backup window whenever possible.

Some critical aspects of the problem revealed the necessity of intelligent support for the uplink management: (a) uplink is an incremental process, so there is a need to *dynamically manage a current plan*; (b) there is always a need to insert additional command sets to upload for some emergency or *unforeseen events*; finally (c) uplink planning is a continuous task that, even being very important for the mission, tends to become routine and, hence, inevitably prone to errors.

2.2.2. RAXEM and RAXEM2 Solution

Both RAXEM and RAXEM2 use an AI constraint-based modeling and solving approach to plan each command file for uplink, retaining a backup window wherever possible, keeping the on-board timeline as full as feasible, and ensuring the safety of the spacecraft at all times. The solver employs a constructive algorithm which provides a good trade-off between speed and optimality of the solution [2, 3]. The user can provide hints to the solver. Planning parameters can be altered for individual M_{DAFs}. The possibility of tuning the solution is provided allowing to perform *what-if* analysis. A key point in RAXEM, inherited by RAXEM2, is to support the continuity of work of mission planners. They are in continuous contact with payload PIs (Principal Investigators) and may receive commands to be uplinked distributed over time including the possibility of having to accommodate new activities on a short notice. As a consequence the tool have been endowed with an interaction layer that supports incremental plan definition and management (see the specific discussion on plan management in [2]).

Key addition in RAXEM2 have been both significant advancements to the user interaction functionalities and new functionalities to automatically generate instructions to the Spacecraft Controller (SPACON), in charge of the uplink, hence arriving to cover the whole mission uplink functional cycle (see [11] for a detailed report). Thanks to RAXEM2, the engineering workload has been reduced from five hours to below one hour. The software was up to expectations with respect to dynamically generating consistent plans for uplink. Plan management has proved to be a very valuable feature to minimize human errors during continuous operations.

2.3. The ApsI Project and the ApsI-Trf

Both the projects described above are examples of great success in introducing AI techniques within ESA mission planning context, and have shown clear advantages in term of performance and users' satisfaction. However, it is worth highlighting how a great effort and amount of time was necessary to both understand the domains and the problems, capturing all the specificity, and to create a model for these domains. The work done for MEXAR2 has been, in some way, useful for the RAXEM2 tool, but in general the development process has been time-consuming and extremely demanding. In addition, both MEXAR2 and RAXEM2 were devoted to solve very specific and isolated problems, while space missions offer many opportunities for relying on AI planning and scheduling to solve problems. This experience suggested to us to operate in a more systematic way trying to identifying commonalities and similarities among the different domains and problems within the space context with the aim of developing a more general and flexible approach that can be applied to different cases.

The general pursued idea is the one of improving the *process* of tool development, taking advantage of state of the art planning and scheduling technology.

The opportunity to investigate in this direction has been provided by the ApsI (Advanced Planning and Scheduling Initiative) project, an ESA initiative to develop an open framework for the flexible support of mission planning systems. The result of this project is a quite general software framework, named ApsI-Trf (where Trf stands for Timeline-based Representation Framework), which provides the basic elements for modeling the relevant entities in the space context [4]. In these contexts the relevant aspects are represented by the ability to deal with *time*, *resources*, description of *operational modes*, and *synchronizations* among events. The ApsI-Trf offers a structured library for managing effectively and efficiently these elements and provides the flexibility to model different domains and problems. It is centered on the concept of *timelines*, which describe features that evolve over time, a concept that is particularly suitable and close to the way of working of human mission planners, thus offering also a good metaphor for managing the interaction with the users.

The ApsI-Trf has been shown at work, and during the ApsI project three separate groups of scientists have rather quickly developed specific innovative applications: (1) first, the problem of Long Term Planning for MARS-EXPRESS has been addressed by our team producing MrSPOCK (described in the next section); (2) second, the problem of science planning in the INTEGRAL mission has been addressed, and a solution proposed by other colleagues [12] has been delivered to ESA that provides a very effective meta-heuristic the problem built on top of the ApsI-Trf services; (3) the third case focused on the support to Long Term Planning of XMM-Newton [13]. These results demonstrated the effectiveness of the software framework in speeding up the application develop-

ment. Furthermore, we are currently working to infuse a planner developed in the ApsI platform as a component for deliberative decisions in the GOAC project [6].

2.4. MrSPOCK

The "MARS-EXPRESS Science Plan Opportunities Coordination Kit," or MrSPOCK, is an application developed to support the collaborative problem solving process between the Science Team and the Operation Team of the MARS-EXPRESS mission. The tool has been developed on top of the ApsI-Trf and combines diversified research aspects from the planning and scheduling area. The leverage offered by the software development platform has allowed us to deliver increasingly accurate and performant versions of the complete system on May, August 2008 and January 2009. During 2009 and 2010 the system has entered a long phase of testing and comparison with current support tools. It is the main tool for Long-Term Planning (LTP) for MARS-EXPRESS starting January 2011.

2.4.1. The MEX-LTP Problem

The two groups of human planners mentioned above iteratively refine a plan containing all activities for the mission. The process starts at the Long-Term Plan (LTP) level – three months of planning horizon – and is gradually refined to obtain fully instantiated activities at the Short-Term Plan (STP) level – one week of planning horizon. This process continuously leads to weekly STPs, which are then further refined every two days to produce final executable plans.

The goal of MrSPOCK has been the generation of a pre-optimized skeleton LTP, which will then be subject to cooperative science team and / operation team refinement (see [5] for more details).

A critical issue in developing the application to produce the MARS-EXPRESS skeleton LTP is represented by the fact that the mission was already running for four year. As a consequence there existed a daily practice, consolidated over time, to be taken into account. Such a practice generated a number of quite specific operational constraints that could not be removed but rather managed by the new tool. Such constraints are difficult to capture for a general purpose solving framework while they are more easily handled in a domain specific solver.

2.4.2. The MrSPOCK Solution

An interesting aspect of MrSPOCK is its hybrid combination of a core constraint-based representation that supports timeline-based planning and scheduling with an optimization genetic algorithm that exploits such representation and an interaction front-end which has multiple features. Apart from the developed prototype, it is worth highlighting here the interesting leverage we obtained with respect to our previous experience in ESA projects due to the use of the ApsI-Trf. This general framework has allowed us to capture an amount of constraints with

a basic domain description language that once parsed by the APSI-TRF creates a basic representation layer. This contrasts with our previous experience in developing the MEXAR2 tool, where a paper and pencil model-based formalization uses timelines to capture problem features but the whole implementation was then done from scratch for the whole application. In MrSPOCK the amount of general purpose modules used in the implemented system is quite high with respect to our previous work. It is also worth mentioning that the development of an application entirely based on a domain independent solver would require the customization of a large amount of specific knowledge in the domain description with a consequent production of a rather cumbersome domain model. Our choice has been to use APSI-TRF for clean modeling purpose while relying on a specific module for driving efficient problem solving (see [5] for a complete description).

3. Lessons Learned and Challenges

The development experience gained in these years working for ESA has mainly brought a contribution in relation to the synthesis of an approach to the process of system development. In particular, the concepts of an *End-to-End* approach have emerged from the work of design and development. According to our experience the key for successful deployment of AI planning and scheduling technology is the ability to deliver an End-to-End software system that minimizes the impact on the users' work habits while offering a real support to regular difficult activities. To foster a smooth deployment of our solutions within the working environment, we have developed an approach during the years that is depicted in the simple drawing of Fig. 3. The figure shows three main blocks that integrate their work: (a) a domain model able to capture the segment of the real world relevant for the problem to be solved; (b) a problem solver which guide the search process and forces decisions on the domain representation to synthesize a solution to the current problem; (c) a set of user-oriented interaction services that allows end-users to access the domain representation, the solution representation and, usually, a set of parameters associated with the problem solver.

In the following sections, aspects connected to these three blocks are examined in light of the experience gained during our work.

3.1. Modeling Approach

An effective domain model is the core element of a problem solving approach based on AI techniques. In the following two sections, we highlight two complementary issues about the modeling of planning domains: the adopted *planning paradigm* (how planning problems are represented) and the criteria to compare and rank different solutions of a planning problem (how to represent the quality of a solution).

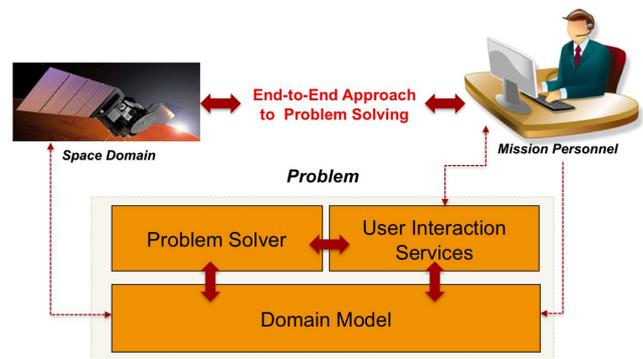


Fig. 3. Supporting the development of an End-to-End approach to problem solving.

3.1.1. The Planning Paradigm

The basic idea for choosing the planning paradigm was to follow the hints from the existing practice in mission planning domains as a main driving factor. There are a number of features that are usual trademark of such domains:

1. The world of interest is described in terms of “events” that happen in time instead of in terms of static properties that can be true or false.
2. The working domain is constituted by physical and natural systems interacting with each other instead of being constituted by a single agent that interacts with the world.
3. These systems are described in terms of “states” they can assume (and consequent constraints on them often are represented as Finite State Automata) instead of in terms of “what they can do and when.”
4. The interactions of these systems are described as causal patterns of temporal synchronizations between the states they can go through instead of in terms of effects that they produce in the surrounding world.
5. The world presents non trivial limited availability of some resources, and the interactions of the systems must take into account the competition for limited resources.
6. The problems are described in terms of system’s desired behaviors and resource optimization instead of being formulated in terms of “which actions will achieve a given status of the world’s static properties.”

The most studied paradigm within the AI planning research community, the so-called “action based” approach [14, 15], usually matches badly with such typical needs of a mission planning domain. It rather presents a number of downsides. The modeling perspective assumes that an executive agent performs *actions* (represented as a collection of action schemata that describes

pre-conditions for action application and the effects of the actions on the world) against a world of *atemporal* properties. The world is usually described in terms of static properties that can be true or false. Furthermore, the paradigm considers a single agent that acts on the world rather than a set of interacting systems. In addition, the world (different space scenarios) usually presents need to describe non trivial limited availability of some resources, and the interactions of the systems must take into account the competition for limited resources. Such features are not naturally represented in “action based” languages [16].

On the contrary, the approach usually referred to as “timeline-based planning” [4, 17, 18] is more suited for the task. According to this approach the planning domain is described in terms of temporal evolutions, called timelines, of sub-systems, and their interactions are modeled as temporal synchronizations between their behaviors. This paradigm is able to capture plans and planning problems in mission planning domains because:

1. Planning problems, surrounding conditions and solutions are all represented by means of timelines. The timeline is a *native* representation primitive in mission planning domains, hence human planners can easily understand and interact with it.
2. Scheduling and resource reasoning can be efficiently applied to timeline-based domains (see [4] for an example).
3. Externally defined temporal events (like flight dynamics, visibility windows, exogenous events and so on) can be naturally represented as timelines.
4. The planning problems that can be addressed concern the achievement of a desired behavior (for one or more subsystems) compliant with the temporal synchronizations and resource allocations that describes correct sub-systems interactions.

In this perspective a timeline-based approach simplifies the elicitation of the domain expert knowledge, the problem formalization and the plan knowledge representation in general. A comprehensive comparison between action-based and timeline-based approach to planning is outside the scope of this paper. We here wanted to enumerate some of the main factors that are the milestones for the prevalent use of timeline based approaches into space applications. This modeling approach, just with attention to different aspects, is the same followed by EUROPA [19], ASPEN [20], and APSI-TRF [4].

3.1.2. Solution Quality

The quality of a solution is one of the most important aspects of a planning support system within the space mission environment. The work from JPL, described in [7], identifies several parameters that contribute to define the quality of a plan. *Efficiency* and *flexibility* are, for instance, two measures that also have a large impact on the

success of the mission. In particular, the first dimension can positively affect the space agencies’ efficiency by reducing costs thanks to an optimized use of the available resources. As for the second aspect, generating *robust* solutions can help handle any failures or unexpected events, as well as support a possible rescheduling need due to emerging problems. The quality of a plan is also connected to the system’s ability to generate solutions that respect both the *soft* and the *hard* constraints of the problem as well as the *preferences* of the mission planners.

Our experience is in line with the above considerations. In particular, in almost all of our applications a common requirement is the need to consider divergent objectives for which it is important to find a good compromise – this is the case of the MEXAR2 and MrSPOCK systems. For this reason the quality of a solution has several aspects that can be viewed as different components of a multi-objective function to be optimized.

As an example, MrSPOCK defines a multi-objective function whose components are mainly three; *Science*: optimizing with respect to this metric entails maximizing the activities of science compared with available opportunities; *Downlink*: by optimizing this component of the objective function the downlink opportunities are maximized and possibly more data is downloaded; *Uplink distribution*: a final important aspect consists in maximizing the opportunity to communicate with the satellite via a uniform distribution of opportunities to uplink, i.e., have opportunities to communicate with the satellite which are numerous and evenly distributed, so as to avoid long windows of non-communication with the satellite. Of course, these goals are conflicting and influence one another. For example, both the activities of science, and some maintenance tasks that must be done regularly contrast with the goal of having numerous and uniformly distributed opportunities for the uplink.

MrSPOCK provides the user the possibility of tuning the algorithm parameters so as to favor one or the other component of the objective function. In this way he/she can generate different solutions and select from those available in a database, depending on the needs of the moment. Overall this is a characteristic that we have tried to implement in/for all of our applications. In fact, in almost all delivered systems, we have allowed the user to generate a set of solutions to the problem that were based on different aspects of solution quality, leaving to the human planner the final choice.

3.2. Support for Solving Algorithms

A central role in the *End-to-End* development approach for mission planning tools is played by core algorithms. Mission planning problems can be often modeled as integrated planning and scheduling problems. They involve several complex constraints and many conflicting objectives that users want to accommodate. According to our experience, a rigid algorithm that finds a specific solution is seldom the successful choice. On the contrary, a multi-objective formulation of the problem, connected to

a heuristic approach able to accommodate divergent goals and integrating the user's knowledge, can be the winning choice. The analysis that follows focuses on two different issues for developing interactive mission planning tools: (1) efficient *basic solving algorithms*; (2) integrating *planning* and *execution*. In Section 3.3 we also describe some challenges and lessons learned about algorithms for the user *interaction* process.

3.2.1. Solving Algorithms for Mission Planning

In general, our approach for solving planning and scheduling problems combines heuristic search and constraint reasoning. Specifically, according to our experience, this combination of techniques has emerged as a robust methodology to quickly produce good-quality solutions for a variety of mission planning problems. Despite the advantages of finding *optimal* solutions to scheduling and planning problems, we preferred to use heuristic and incomplete approaches over complete and systematic methods. This is for several reasons. First, generally a complete method cannot scale on large instances. Second, in some cases reaching optimal solutions is meaningless, as in practice we are often dealing with models that are simplifications of the working domain (e.g., this is the case of long-term planning of satellite activities). Third, according to our experience, the use of heuristic strategies allows a smoother integration of the mission planner choices in the search process.

As quickly mentioned in the previous sections, mission planning is a large process involving different stages and interaction among several teams (e.g., Science Working Teams, Mission Analysis Team, Flight Dynamics and Flight Control Teams). In addition, we can identify at least three phases for planning: long, medium and short-term planning, which respectively have durations of several months, one months (or less) and one week (or less). Along these phases planning activities are *incrementally* refined from an horizon of several months to one week or less. In fact, as some activities cannot be predicted one week in advance (e.g., flight dynamic requests), a very short-term plan can be generated each one or two days. The use of an effective *one-shot* solving algorithm is the first step in developing a solution to a mission planning problem, such that, given a set of input parameters, a solution is generated over a single run of the algorithm. No user interaction is allowed during the resolution process and there is the hypothesis that no changes or additions to the problem constraints are allowed at *execution time*. These last two important issues (execution and user interaction) are considered in what follows.

3.2.2. Mission Planning and Execution

One of the challenges for mission planning systems is the generation of plans that are both robust to uncertainty and adaptable to change. In fact, in many practical situations, the effective value of many variables of the plan are only known at execution time, and their values are not under the control of the mission planners. Hence, even if an

initial reference plan can be created via a batch algorithm, its adaptation, which takes into account the real values of the plan variables, can be only obtained via a *dynamic planning* approach, where a reference plan is *continually* updated on the basis of the performed observations on the real world. As suggested in [7], a general dynamic planning approach applies incremental planning techniques in conjunction with a hierarchical approach. In the latter, the long-term planning horizon is planned only at a very abstract level, whereas shorter and shorter planning horizons should be planned in greater detail. The devised general framework in [7] posts several algorithmic challenges and many of them represent open research areas. One challenge, especially at the short-term level, is to guarantee that the generated plan is *controllable*, that is there is a strategy for executing all of the planned activities under the user's control that satisfies all requirements, in all situations involving the uncontrolled events. In the case that the source of uncertainty is only the temporal constraints, and the adopted model is the Simple Temporal Problem, an interesting property is the so-called *dynamic controllability* [21, 22], where it is assumed that each uncertain duration becomes known (is observed) after it has finished, and the property requires a successful strategy that depends only on the past outcomes. About the verification of dynamic controllability the work [23] proposes a model-checking verification tool suitable for verifying flexible temporal plans. Other proposals pursue the representation of conditional/contingent temporal networks or, in general, plans that contains alternatives [24, 25]. When resource constraints are also involved in the plans, steps towards plan controllability are proposed in [26, 27]. Indeed it is worth saying that often to obtain robust solutions with respect to execution it is also the important ability to adapt the one-shot basic solution you designed for a specific problem (see for an example the approach to robustify solutions used in MEXAR2 as described in [28]).

3.3. User Interaction Support

Since the first version of the MEXAR software [8, 29], we pursued the goal of defining interactive decision support systems for solving mission planning problems. The main idea was to integrate human strategic capabilities and automatic problem solving algorithms to find solutions accommodating different and contrasting goals under the control of the mission planners. In fact, despite the use of expressive models and powerful algorithms, there still remains a *gap* between the user desiderata and the set of found solutions. Integrating user knowledge in the search process helps to fill this gap. Over the years we analyzed and applied several techniques to include user's choices, both at the strategic and detailed level. Among the important aspects we can highlight the following ones.

Scenario anticipation This concept is also known as *What-if Analysis* and is based on the idea of obtaining alternative solutions corresponding to different foreseen scenarios and taking into account priorities, preferences and user's experience. More specifically, the idea is to

compare different solutions generated by operating different types of modifications: from the modification of the position of a single activity (e.g., change the scheduled time of a scientific observation) to a radical change of the solution by using for instance a different solving strategy. The use of *incremental algorithms* can improve the efficiency of this kind of analysis. For example, if the modifications are operated only on the temporal constraint and the reference model is the Simple Temporal Problem, then there are available efficient algorithms for operating incremental modifications [30,31] on the reference solution. It is worth saying that due to the complex constraints involved in mission planning problems, even a small modification can induce high computational efforts to regain a feasible situation.

Role of explanation Explanation-based reasoning can be used to deduce the *culprit* of a failure, as well a subset of constraints justifying an action of the solver (e.g., a value removal or a bound update). An example of this kind of application is described in [32], where a simple algorithm to compute “user oriented explanation” is presented which employs a filtering step to reduce the constraints involved in a temporal conflict. The remaining constraints are then “interpreted semantically” in order to enable the user to understand the reason for the conflict and possibly resolve it by relaxing some constraints. In [33] a similar attempt to extract meaningful information from constraint-inconsistency detection is presented in the context of planning for space. Specifically, all temporal constraints in the plan are analyzed, and the possible temporal inconsistencies (no-goods) are detected and categorized. Overall the problem of generating automatic explanation is an open challenge as well as an interesting research issue [34,35] the result of which could be profitably applied to support the incremental building of an executable solution, by for instance suggesting to the users possible resolution options or tradeoffs among which to choose.

Continuity of use One last aspect that is worth mentioning is the user’s attitude toward the continuous use of decision support tools within space contexts. On one hand we found that users want to be relieved of their hard work, entrusting the automated algorithm to deal with technical details and difficult calculations. On the other hand, it is also clear that users want solutions that incorporate their experiences, preferences and expectations, and above all they want complete control and supervision over the final choice. Somehow we ended up realizing examples of intelligent systems augmenting work environments where users can have access to a number of collaborative systems that facilitate their regular activities.

Observing back in time the various deployed systems we can see how different issues have been addressed in different systems. In the MEXAR demonstrator the interaction support has been very important to develop user trust toward the AI based solution. As explained in [29], this system was endowed with a quite advanced graphical rep-

resentation that allowed the users to start understanding that we were solving their specific problems. Additional services allowed us to explain the model-based approach showing how it was possible to dynamically modify the problem at hand from the graphical interface instead of rewriting application code. In MEXAR2 [1] the key issue was to preserve previous work practice. To this purpose a number of functionalities have been implemented to preserve both the data cycle from the ongoing mission and the user responsibility over final decisions and integrate his/her strategic guidance into the problem solving. In RAXEM the issue of continuous plan management has been dominant over other services [2]. Supporting the incremental nature of the problem was the key for success. To this purpose, the design and implementation of the interaction functionalities represented the majority of the work, being the problem solving task relatively simple. In MrSPOCK [5] the more relevant problem was to foster the collaboration among different groups of people (science and operations). In this respect the pre-optimized skeleton plans generated by our system was a useful starting point for the negotiation among the two groups. It is worth highlighting the difficulty in putting together the different needs for the interaction services coming from different applicative scenarios. To this purpose the type of interaction support provided by the APSI-TRF is still limited thus entailing that the majority of the work still needs to be done around the specificity of each problem. This obviously identifies an open issue that could be a direction for future research.

4. Final Remarks and Conclusions

In this paper we have tried to analyze our work for ESA over ten years trying to underscore rationale for success factors and get some hints for future developments. We consider it a significant achievement having fielded three different End-to-End interactive problem solvers and having established a collaboration with the ESOC MARS-EXPRESS mission environment which is traditionally quite conservative. Key factors for success have been: (a) the choice of the timeline-based representation that turned out as important for facilitating the elicitation of relevant features and their constraints, (b) to enable hybrid approaches to problem solving that has injected flexibility, also allowing the customization of problem specific biases never easily captured by general solvers, (c) the choice of interactive tools that preserve the role and authority of human users, and, a more subtle point, (d) proposing approaches which were non-disruptive with respect to previous or current mission practice. This last feature is often difficult to describe and very often requires a lot of additional work to be defended but nevertheless plays a key role in technology acceptance.

In the paper we have also introduced the distinction between product vs. process driven approaches. For sure the APSI project has paved the way in a direction which is not only interesting but somehow inevitable for the ESA.

In fact, it may represent a way to cope with the natural conservativeness of an environment which is doing mission operation. Additionally, the fact that MrSPOCK, a product based on the APSI-TRF platform, arrived to be operational creates a success story on top of which to build future developments. Nevertheless it is worth saying that the APSI-TRF is a first step toward the synthesis of software development environments for planning and scheduling systems. It contributes with a number of ideas in an area in which already the notable examples of EUROPA [19] and ASPEN [20] exist. It is worth noting that the current APSI-TRF allows us to integrate constraint-based resource reasoning in a timeline-based development environment in a quite effective way [4]. We should say that the platform in its current shape is a significant step that needs further investment before achieving industrial strength. For sure we can say that it captures several of the modeling features needed in space domains, and it is flexible enough to serve different application domains. Nevertheless to develop effective problem solvers, in particular the hybrid ones that are very often needed in space, the ability and know-how of a research group is still difficult to be customized in a software engineering tool of this type.

Acknowledgements

This work has been sponsored by the European Space Agency under several contracts over the years. The authors would like to thank the Agency for such support. Specific people need to be thanked for their contributions, their continuous assistance and, definitely, for trusting us. In particular we would like to thank Nicola Policella, Erhard Rabenau, Jonathan Schulster, Michel Denis and Alessandro Donati. The work concerning APSI has been supported under ESA ITT 5965, 2006. APSI partners have been VEGA GmbH, ONERA, University of Milan and ISTC-CNR. Comments from journal reviewers have helped us to improve the paper. Authors are indebted to Brad Clement for his very careful reading of a previous version of the paper.

References:

- [1] A. Cesta, G. Cortellessa, S. Fratini, A. Oddi, and N. Policella, "An Innovative Product for Space Mission Planning – an a posteriori evaluation," ICAPS-07. Proc. of the 17th Int. Conf. on Automated Planning and Scheduling, 2007.
- [2] A. Cesta, G. Cortellessa, M. Denis, A. Donati, S. Fratini, A. Oddi, N. Policella, E. Rabenau, and J. Schulster, "Continuous Plan Management Support for Space Missions: the RAXEM Case," PAIS/ECAI-08. Prestigious Applications on Intelligent Systems, Proc. of the 18th European Conference on Artificial Intelligence, pp. 703-707, 2008.
- [3] G. Bernardi, A. Cesta, and G. Cortellessa, "Deploying RAXEM2: Planning Improvements in Daily Work Practice," SPARK-09. Scheduling and Planning Applications workshop at ICAPS, Thessaloniki, Greece, 2009.
- [4] A. Cesta and S. Fratini, "The Timeline Representation Framework as a Planning and Scheduling Software Development Environment," 27th Workshop of the UK Planning and Scheduling Special Interest Group, pp. 17-24, Edinburgh, UK, 2008.
- [5] A. Cesta, G. Cortellessa, S. Fratini, and A. Oddi, "MrSPOCK – Steps in Developing an End-to-End Space Application," Computational Intelligence, Vol.27, No.1, pp. 83-102, 2011.
- [6] A. Ceballos, S. Bensalem, A. Cesta, L. De Silva, S. Fratini, F. Ingrand, J. Ocon, A. Orlandini, F. Py, K. Rajan, R. Rasconi, and M. Van Winnendael, "A Goal-Oriented Autonomous Controller for Space Exploration," ASTRA 2011, 11th Symposium on Advanced Space Technologies in Robotics and Automation, 2011.
- [7] S. Chien, T. Estlin, F. Fisher, and G. Rabideau, "Lessons Learned Deploying Space Planning Systems," Applied Artificial Intelligence, Vol.15, No.2, pp. 161-187, 2001.
- [8] A. Cesta, G. Cortellessa, A. Oddi, and N. Policella, "A CSP-Based Interactive Decision Aid for Space Mission Planning," AI*TA-03, Vol.2829 of LNAI, pp. 511-522, 2003.
- [9] A. Oddi, N. Policella, A. Cesta, and G. Cortellessa, "Constraint-Based Random Search for Solving Spacecraft Downlink Scheduling Problems," G. Kendall, E. Burke, S. Petrovic, and M. Gendreau (Eds.), Multidisciplinary Scheduling: Theory and Applications, pp. 133-162, Springer, 2005.
- [10] A. Cesta, G. Cortellessa, M. Denis, A. Donati, S. Fratini, A. Oddi, N. Policella, E. Rabenau, and J. Schulster, "MEXAR2: AI Solves Mission Planner Problems," IEEE Intelligent Systems, Vol.22, No.4, 2007.
- [11] E. Rabenau, A. Donati, M. Denis, N. Policella, A. Cesta, G. Cortellessa, A. Oddi, S. Fratini, and G. Bernardi, "The Raxem Tool on Mars Express – Uplink Planning Optimisation and Scheduling Using AI Constraint Resolution," IWPS-09. Proc. of the 6th Int. Workshop on Planning and Scheduling for Space, 2009.
- [12] C. Pralet and G. Verfaillie, "AIMS: A Tool for Long-term Planning of the ESA INTEGRAL Mission," IWPS-09. Proc. of the 6th Int. Workshop on Planning and Scheduling for Space, Pasadena, CA., 2009.
- [13] M. Lavagna and F. Castellini, "Advanced Planning and Scheduling Initiative's XMAS tool: AI for automatic scheduling of XMM-Newton long term plan," IWPS-09. Proc. of the 6th Int. Workshop on Planning and Scheduling for Space, Pasadena, CA., 2009.
- [14] R. Fikes and N. Nilsson, "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," Artificial Intelligence, Vol.2, No.3-4, pp. 189-208, 1971.
- [15] M. Fox and D. Long, "PDDL 2.1: An Extension to PDDL for Expressing Temporal Planning Domains," J. of Artificial Intelligence Research, Vol.20, pp. 61-124, 2003.
- [16] J. Frank, K. Golden, and A. Jonsson, "The Loyal Opposition Comments on Plan Domain Description Languages," Proc. of the PDDL Workshop at ICAPS-03, pp. 39-48, 2003.
- [17] J. Frank and A. Jonsson, "Constraint Based Attribute and Interval Planning," Constraints, Vol.8, No.4, pp. 339-364, 2003.
- [18] N. Muscettola, "HSTS: Integrating Planning and Scheduling," M. Zweben and M. S. Fox (Eds.), Intelligent Scheduling, Morgan Kaufmann, 1994.
- [19] Europa, "Europa Software Distribution Web Site," 2008. <http://code.google.com/p/europa-pso/wiki/EuropaWiki>
- [20] S. Chien, G. Rabideau, R. Knight, R. Sherwood, B. Engelhardt, D. Mutz, T. Estlin, B. Smith, F. Fisher, T. Barrett, G. Stebbins, and D. Tran, "ASPEN – Automated Planning and Scheduling for Space Mission Operations," Proc. of SpaceOps'00, 2000.
- [21] P. Morris and N. Muscettola, "Temporal Dynamic Controllability Revisited," AAAI, pp. 1193-1198, 2005.
- [22] P. Morris, N. Muscettola, and T. Vidal, "Dynamic Control Of Plans With Temporal Uncertainty," IJCAI, pp. 494-502, 2001.
- [23] A. Cesta, A. Finzi, S. Fratini, A. Orlandini, and E. Tronci, "Analyzing Flexible Timeline-based Plans," In ECAI-10, Vol.215, pp. 471-476, IOS Press, 2010.
- [24] P. Conrad, J. Shah, and B. Williams, "Flexible Execution of Plans with Choice," ICAPS, 2009.
- [25] I. Tsamardinos, T. Vidal, and M. E. Pollack, "CTP: A New Constraint-Based Formalism for Conditional, Temporal Planning," Constraints, Vol.8, No.4, pp. 365-388, 2003.
- [26] N. Policella, A. Cesta, A. Oddi, and S. F. Smith, "Solve-and-Robustify. Synthesizing Partial Order Schedule by Chaining," J. Scheduling, Vol.12, No.3, pp. 299-314, 2009.
- [27] R. Rasconi, A. Cesta, and N. Policella, "Validating Scheduling Approaches Against Executional Uncertainty," J. of Intelligent Manufacturing, Vol.21, No.1, pp. 49-64, 2010.
- [28] A. Oddi and N. Policella, "Improving Robustness of Spacecraft Downlink Schedules," IEEE Trans. on Systems, Man, and Cybernetics – Part C Applications and Reviews, Vol.37, No.5, 2007.
- [29] G. Cortellessa, A. Cesta, A. Oddi, and N. Policella, "User Interaction with an Automated Solver: The Case of a Mission Planner," PsychNology Journal, Vol.2, No.1, pp. 140-162, 2004.
- [30] A. Cesta and A. Oddi, "Gaining Efficiency and Flexibility in the Simple Temporal Problem," TIME, 1996.
- [31] L. Plancken, M. d. Weerdt, and R. v. d. Krogt, "P3C: A New Algorithm for the Simple Temporal Problem," ICAPS, pp. 256-263, 2008.
- [32] S. Smith, G. Cortellessa, D. Hildum, and C. Ohler, "Using a scheduling domain ontology to compute user-oriented explanations," L. Castillo, D. Borrajo, M. Salido, and A. Oddi (Eds.), Planning, Scheduling, and Constraint Satisfaction: From Theory to Practice, IOS Press, 2004.

- [33] J. Bresina and P. Morris, "Explanations and Recommendations for Temporal Inconsistencies," IWPSS-06, Space Telescope Science Institute, Baltimore, MD, USA, October 22-25, 2006.
- [34] J. Amilhastre, H. Fargier, and P. Marquis, "Consistency Restoration and Explanations in Dynamic CSPs – Application to Configuration," Artificial Intelligence, Vol.135, No.2002, pp. 199-234, 2002.
- [35] N. Jussien and V. Barichard, "The PaLM System: Explanation-based Constraint Programming," Proc. of TRICS: Techniques foR Implementing Constraint programming Systems, a post-conference workshop of CP 2000, pp. 118-133, Singapore, 2000.



Name:
Amedeo Cesta

Affiliation:
Senior Research Scientist and Group Lead, CNR, Consiglio Nazionale delle Ricerche, Istituto di Scienze e Tecnologie della Cognizione

Address:
Via S. Martino della Battaglia 44, I-00185 Rome, Italy

Brief Biographical History:
1983 M.Sc. in Electronic Engineering, University of Rome "La Sapienza"
1992 Ph.D. in Computer Science, University of Rome "La Sapienza"
1991- Research Scientist in Artificial Intelligence at CNR (since 2001, Senior research scientist)
1997- Founded and currently coordinates activities of the Planning and Scheduling Technology Laboratory (PST) at CNR-ISTC

Main Works:

- multi-agent systems, intelligent human-computer interaction, planning & scheduling and the synthesis of innovative decision support systems
- In the period between 2000 and 2011 he led several projects supported by the European Space Agency (ESA) that produced advanced prototypical demonstrators (MEXAR, GOAC) and operational tools and environments (MEXAR2, RAXEM, RAXEM2, APSI, MrSPOCK).



Name:
Gabriella Cortellesa

Affiliation:
Research Scientist, CNR, Consiglio Nazionale delle Ricerche, Istituto di Scienze e Tecnologie della Cognizione

Address:
Via S. Martino della Battaglia 44, I-00185 Rome, Italy

Brief Biographical History:
2001 M.Sc. in Computer Science Engineering, University of Rome "La Sapienza"
2005 Ph.D. in Cognitive Psychology, University of Rome "La Sapienza"
2002- Research Assistant, CNR, Consiglio Nazionale delle Ricerche
2008- Research Scientist, CNR, Consiglio Nazionale delle Ricerche

Main Works:

- mixed-initiative problem solving, decision support systems, experimental methods for evaluating intelligent system features, user-centered design, evaluation methods for human-robot interaction
- She has a significant experience in applying AI to space projects having been part of the project development team for the MEXAR, MEXAR2, RAXEM, RAXEM2 and MrSPOCK tools for the European Space Agency.



Name:
Simone Fratini

Affiliation:
Research Scientist, CNR, Consiglio Nazionale delle Ricerche, Istituto di Scienze e Tecnologie della Cognizione

Address:
Via S. Martino della Battaglia 44, I-00185 Rome, Italy

Brief Biographical History:
2002 M.Sc. in Computer Science Eng., University of Rome "La Sapienza"
2006 Ph.D. in Computer Science Eng., University of Rome "La Sapienza"
2003- Research Assistant, CNR, Consiglio Nazionale delle Ricerche
2008- Research Scientist, CNR, Consiglio Nazionale delle Ricerche

Main Works:

- integration of planning and scheduling, AI software architectures, expressive domain modeling languages, knowledge engineering
- He has been part of the project development team for the MEXAR2, RAXEM and MrSPOCK tools and as been the main designer and developer of the APSI TRF (the Timeline-based Representation Framework).



Name:
Angelo Oddi

Affiliation:
Research Scientist, CNR, Consiglio Nazionale delle Ricerche, Istituto di Scienze e Tecnologie della Cognizione

Address:
Via S. Martino della Battaglia 44, I-00185 Rome, Italy

Brief Biographical History:
1993 M.Sc. in Electronic Engineering, University of Rome "La Sapienza"
1997 Ph.D. in Medical Computer Science, Univ. of Rome "La Sapienza"
1997- Research Assistant, CNR, Consiglio Nazionale delle Ricerche
2001- Research Scientist, CNR, Consiglio Nazionale delle Ricerche

Main Works:

- application of AI for scheduling, automated planning and constraint reasoning, particularly, several algorithms for temporal reasoning and both local search and randomized approaches for schedule optimization
- He has developed the MEXAR and MEXAR2 systems acting also as the technical director in the MEXAR2 and RAXEM projects.



Name:
Giulio Bernardi

Affiliation:
Research Programmer, CNR, Consiglio Nazionale delle Ricerche, Istituto di Scienze e Tecnologie della Cognizione

Address:
Via S. Martino della Battaglia 44, I-00185 Rome, Italy

Brief Biographical History:
2009 B.Sc. in Computer Science Engineering, University of "Roma Tre"
2009- M.Sc. Course, University of "Roma Tre"

Main Works:

- He developed RAXEM2, a software tool that assists users in delivering commands to the ESA's Mars Express space probe.
- He is also involved in developing interactive systems for other research projects of the Planning and Scheduling Technology Laboratory at CNR.