Innovative Rover Operations Concepts – Autonomous Planner (IRONCAP) – Supporting Rover Operations Planning on Ground

R. Steel[1], A. Hoffman[2] and M. Niézette[3]
VEGA Space GmbH, Europaplatz 5, D-64293 Darmstadt, Germany

A. Cimatti[4], M. Roveri[5]
Fondazione Bruno Kessler – irst, Via Sommarive 18, 38123 Povo (TN), Italy

K. Kapellos[6]
TRASYS, Terhulpsesteenweg 6c, 1560 Hoeilaart, Belgium

and

A. Donati[7], N. Policella[8]
European Space Operations Centre, Robert-Bosch-Strasse 5, D-64293 Darmstadt, Germany

IRONCAP is an ESA study project to explore and define the concepts, techniques and interactions needed to plan and schedule the activities of an interplanetary rover. Its aim is to develop a prototype system to support the science and engineering planning activities of an interplanetary rover using state-of-the-art methods and techniques in planning and scheduling combined with existing and/or developing ground segment systems and technologies. The prototype will support the situational analysis of the rover and facilitate the planning and scheduling of activities/observation for the applicable autonomy levels, supporting the teams in their daily activities. As with any rover mission, situational assessment of the rover has to be performed to establish the context in which the planning of operations can be performed. This is performed on an engineering level and on a science level both with their own goals and objectives. Our paper will discuss this topic further and the issues between the two.

I. Introduction

This paper outlines the aims of the study giving a brief background to why this study is needed, presents the architecture for the prototype detailing the relation and positioning within a ground segment including the expected interactions with the various components of a ground segment, describes the contrast between rover science operations planning and engineering operations planning illustrating the common or conflicting requirements, introduces the planning and scheduling techniques that are used within the prototype, providing details of the two demonstration test cases, and finally summarize the synergies with other ESA projects currently under development. We conclude with a look at the results of the project to date and present the final steps needed to bring the project to completion. Through this study new techniques, concepts and technologies will be explored which will potentially benefit current and future rover missions planned by ESA.

---

[1] Project Manager, Technology Division, robin.steel@vega.de.
[2] Software Engineer, Technology Division, alexander.hoffmann@vega.de.
[3] Practice Leader, Technology Division, marc.niezette@vega.de.
[4] Project Manager, Embedded Systems Unit, cimatti@fbk.eu.
[5] Software Engineer, Embedded Systems Unit, roveri@fkb.eu.
[6] Project Manager, Robotics, Konstantinos.Kapellos@trasys.be.
[7] Technical Officer, HSO-OS, alessandro.donati@esa.int.
[8] Deputy Technical Officer, Future Studies Section, nicola.policella@esa.int..

IRONCAP is an ESA run study project started in January 2011 and being performed by a consortium of three members. The prime contractor VEGA Space GmbH with two partners FBK and TRASYS, each of them providing their own specific expertise to the study; VEGA bringing its expertise in operational ground segments and flying space missions, TRASYS with their extensive knowledge of rover operations & simulation and FBK providing the planning & scheduling experience of model synchronization and planning with uncertainty. This constellation provides for an effective and diverse knowledgebase on which the study will develop, providing a fruitful result.

The key objectives of the study are to:

- Assess and summarize the state-of-the-art concepts and technologies for operations of both orbiting spacecraft and rovers.
- Define advanced concepts for controlling and monitoring rover operations, considering the presence of autonomous planning and execution capabilities in the rover segment. Enabling cutting-edge technology shall be considered during the course of the study since the focus is on future rover missions.
- Identify possible engines and languages to handle the different types of planning data such as occurrences, eventd, activities, and resources.
- Identify optimum ways to synchronize on-board and ground planning processes.

The ultimate result of the study will be the development of a general-purpose proof-of-concept prototype providing a coherent and complete working implementation of an Automated Ground Activity Planning/Scheduling and Validation System for rover operations for the Agency.

A. Background

Before diving into the core of the work it is necessary to first understand some of the drivers for this. Planning of operations within ESA are usually distributed between two entities, one providing the science planning inputs and the other supplying the engineering planning inputs to the overall planning activities of any given spacecraft. Nominally the interactions at a planning level between these two entities, usually with each entity having their own set of tools to perform analysis of previous activities and to facilitate the planning tasks. These tools commonly have different knowledge bases built into them with no real synchronization involved between them. One of the foremost rationales for this study is to investigate and define ways to harmonize these interactions, developing concepts and techniques that are applicable to both teams. Understanding and study of both entities is important to the project to discover the synergies between them.

Another important driver is the prescribed levels of autonomy used to describe the capability of a spacecraft. There are four different levels of autonomy defined in the ECSS-E-70-11 standard for space applications which are:

- E1 - Execution under ground control
- E2 - Execution of pre-planned mission operations on-board
- E3 - Execution of adaptive mission operations on-board
- E4 - Execution of goal-oriented mission operations on-board

As can be expected, for most space rover operations only the E2 to E4 levels are really applicable due to the time delays that are nominally involved and the synchronization of the daylight hours at the rovers location with that of the human controllers on Earth. Moon Rovers may well be able to cope with the small latency to allow for E1 control of a rover, although simulated motion techniques would need to be employed to guide the operator in control.

Currently for most missions the autonomy level E2 is used. These are plans that consist of time-tagged schedules of activities. The on-board controller will execute these schedules and monitor the status of the execution. If some activity fails, the execution will be immediately aborted. This usually results in the controller ensuring that the payload/platform are in a safe state and then waiting for further instructions from the ground.

The next level of autonomy, which is becoming more used in new and already operational missions is the level E3. This level prescribes that the execution of activities on-board can be triggered and driven by events. The controller monitors the status of the system and environment, starting or selecting activities to be performed on the basis of conditions of the on-board telemetry. In this case IRONCAP is required to generate conditional plans that would integrate uncertainty on time, resource consumption, and environment/system state. The controller monitors the execution of the plan together with the environment and system state and triggers the execution of the plan branches on the based on their enabling conditions. If the plan execution fails the controller will react in the same way as at the lower level of autonomy and wait for further instruction from the ground (although non-critical operations can continue to run in the background). This could happen in the case that none of the pre-compiled options in the conditional plan are valid for instance.

At the highest level of autonomy, the E4 level, the on-board controller expects a high-level plan of goals with constraints which is expanded and mapped down to lower level activities by the on-board planning system. The goals and constraints that are passed to the on-board system will generally be expressed at a lower level of abstraction than the goals that are defined as input to IRONCAP. The on-ground planning process is therefore not limited to a process of merging and checking consistency of possible conflicting goals, but also to compile the on-ground timeline to the level of abstraction required by the on-board controller. A mandatory requirement of the project is to be able to demonstrate the IRONCAP concepts using the Goal-Oriented Autonomous Controller (GOAC).

For most ESA missions the E2 level of autonomy is currently the preferred choice, even though this is typically applied to spacecraft and not rover missions. The E3 autonomy level is foreseen for future ESA missions but has already been used in a limited context within some missions in the form of On-Board Control Procedures (OBCP). For the time being the E4 autonomy level has been addressed only at a prototyping level within ESA although plans are already being floated for operational missions.

B. Where does IRONCAP fit in?

IRONCAP studied the concepts needed to define the operations of autonomous Rovers and the systems required on-ground to support these operations. For the ground aspects it complements a parallel study, the GOAC study completed by ESA in October 2011, which aimed at defining and prototyping an on-board autonomous controller capable to support the levels of autonomy up to level E4. The end result provides an integrated on-board goal-oriented re-planning functionality that could be used on future ESA rover missions.

This study project enables ESA to bridge the gap between the science planning and engineering planning for rover operation, combining both concepts into a single prototype tool that can be used in both environments. We provide concepts and techniques that can be used for current and future rover missions as well as providing the necessary integration with the GOAC study by mean of supporting the E2 to E4 levels of autonomy prescribed by the ECSS standard.

In the context of the ground segment, IRONCAP is positioned primarily in the Rovers Operations and Control Centre (ROCC) of a given mission. Its aim to allow the planning and scheduling of rover operations by both the scientific and engineering teams involved in the mission by providing the level of support for the autonomy level available on the rover itself. In support of this it interfaces with the Mission Control System (MCS) obtaining telemetry data in the form of science/housekeeping data and provides the schedules to be uploaded to the spacecraft in the form of time-tagged commanding and/or goal oriented plans. For the planning of communication windows it also interfaces with the ESTACK Management System (EMS) receiving plan view files which contain the station allocations to the rover.

II. Architecture

The architecture of the system, shown in the **Error! Reference source not found.**, takes into account the situational assessment needed by the science planners and the engineering planners and foresees an integrated 3D visualization component. The study has already highlighted the re-use of 3DROV for this purpose, a comprehensive system developed to visualize and simulate rover activities.

IRONCAP will allow the planner to generate plans of different kinds to fulfill the requirements imposed by the three different autonomy control levels, E2 to E4.

During the course of the project we have identified the following four classes of plans derived from the three levels of autonomy that are considered applicable to the project:

- Class A1: This kind of plan is a simple sequence of activities with no conditional branches and with no flexibility on the duration of the activities but allowing for parallel activities. Typically this is called a time tagged schedule within the ground-segment. This kind of plan will be suitable for representing the E2 level of autonomy.
- Class A2: An extension of the class A1 plan but along two directions.
  - Conditions are introduced with only one branch (e.g. "if (battery_level_good) then goto(x,y); experiment(1)"). When the condition does not hold during the execution of this type of plan then the execution is halted, requiring ground-segment intervention/guidance.

    - We also introduce flexibility on the start time and duration of the activities, thus allowing for dependency on activities encoded in relationships between activities.
- Class A3: This plan class further extends A2 by allowing for conditions with multiple branches, providing event-based autonomous operations to be executed on board. For this class the execution of the plan is not necessarily halted if a condition is not met as there may be a recoverable path still available in the execution.
- Class A4: At the highest level of autonomy we derive the class A4 plan. This consists of a set of goals to be uploaded on board to achieve goal oriented mission re-planning. The execution of this type of plan is decided on-board rather than on-ground as is the case in the previous classes of plan. These kinds of plans are compliant with those goal plans that the GOAC system is able to support.

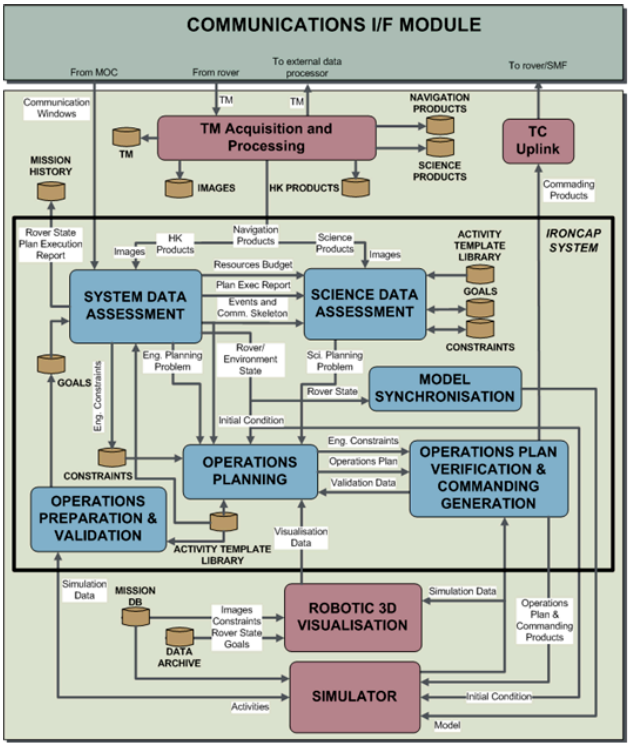

Figure 1. IRONCAP Architectural Overview

From these definitions we can see that plans belonging to class A1 to A3 allows for the execution of more than one activity in parallel. Plans of all four classes are also associated with the set of assumptions used for plan generation which can be monitored during the execution of the plan. Not only are these plans tagged with timing information, but also annotated with other situational checkers, e.g. checking the battery level.

III. Planning and Scheduling

An Operations Planner is responsible for the generation of possible solution plans of a given planning problem if such a solution exists. There may be times when no solution can be found and this is reported back to the user who can then adjust the planning problem or initial constraints based on the reported debugging information. During the refinement of the planning problem several actions can be taken, such as the relaxation/strengthening of planning goals or assumptions, modification of constraints or introduction of additional goal and constraints. The appropriate action to take is governed by the debugging information provided to the operator for analysis

The facilitation of this processing is provided within two components, the Planner & Scheduler and the Planning Problem Refiner. The former is used to determine and generate the actual planning and scheduling activities. In combination with this, the Problem Refiner facilitates the refinement process taking into account the debugging details produced by the Planner & Scheduler and/or validation data from a simulation facility when a solution to the planning problem cannot be found.

To aid the assessment of a valid solution should one be found, additional components are employed, such a 3D visualization tools and simulation tools. These provide graphical feedback to the user allowing for a more thorough assessment of the solution. The level of autonomy required has an influence on the procedure used to produce and analysis the solution. In the case of the autonomy levels E1 to E3 the plan is returned to the operator for further refinement/analysis if a solution is found or processed further to obtain debugging information in the case a solution cannot be found and this is then provided to the operator. For the highest level of autonomy, E4, a slightly difference procedure is used where by a selected subset of goals can be used to find a specific specialized solution. If the selected goal all provide a solution then the operations plan is produce and returned to the operator, otherwise debugging information is obtain, similar to that obtained from the other levels, which the operator can use to de-scope goals or modify the subset selection of goals to be achieved. These different procedures are necessary to cater for the different levels of autonomous behavior rovers support and to successfully, efficiently and effectively perform rover operations.

A. Science & Engineering assessment and planning

One of the most challenging aspects of the project was the understanding, definition and analysis of the various angles that make up the science assessment/planning and the engineering assessment/planning. As with any rover mission, a situational assessment of the location of the rover has to be performed to establish the context in which the planning of operations can be performed. This situational analysis is performed on an engineering level and on a science level both with their own goals and objectives. From an experience point of view, there are few rover missions around to really study the interactions and demands of the science and engineering teams. To this effect some assumptions have had to be made during the study based on experience of non-rover missions to devise possible operational scenarios.

With this in mind, the science assessment is mainly concerned with the evaluation and assessment of what science has been achieved since the last assessment, what exciting new science could be done from what we see now, the science observations already planned to be performed and how to maximize the scientific return to be benefit the community.

In stark contrast to this the engineering assessment is aimed at the state of the space vehicle, constantly checking its health with respect to the last assessment and tweaking parameters to better utilize the platform but in the safest way possible. This would usually involve an evaluation of any energy sources on the space vehicle (i.e. batteries, solar panels, etc.) and their performances, power consumers from the payload/platform, evaluation of all moving/mechanical parts on the vehicle (such as wheel motors, camera arms, internal relays, etc.) noting and reacting to any degradation in performances which could indicate a potential future failure on the spacecraft. This is similar to the assessments made for orbiting satellites.

The result of this is that both assessments provide the goals and objectives for the next planning stage which may or may not conflict with each other. Therefore it is important to cater for both assessments when performing planning operations and provide a mechanism where by the conflicts of interest between the two teams can be resolved and harmonized. The study investigated the collaboration and combination of these two situational assessment analyses with the ultimate aim being to produce a prototype, which will support both necessary approaches, and facilitating interaction between them at the planning domain level.

B. Model Representation & Synchronization Concerns

Within IRONCAP model synchronization is responsible for the update and synchronization of the models used for planning and for plan validation, and for the synchronization of the model used by an external simulator with the model used for reasoning. Formal methods techniques are used for the first kind of synchronization between the planning and plan validation. In particular it can be further decomposed into update of the initial state used for successive plan generation and validation; update of the model used for all the formal reasoning; and finally update of the assumptions used for the plan generation. In the available literature, we were not able to find many articles describing these approaches. However, within the project we are tackling these problems in the following manner. For the synchronization of the initial state we will simulate the plan previously executed and downloaded from the rover. The simulation starts from the previously known state used for the previous plan generation and will be driven not only from the plan, but also from the information coming from the telemetry. Within this phase it is possible also that we discover problems in the model used for simulation, thus the synchronization of the initial state will be tightly integrated with the update of the reasoning model and of the assumptions. For the update of the model and of the assumptions under which to plan, we use techniques developed within the OMC-ARE project for fault-detection and identification. We identify possible faults and/or wrong assumptions by exploiting the telemetry information received, using this information to revise the model (e.g. introduce new faulty-behavior, strengthen the assumptions, etc). Within IRONCAP we exploit and extend techniques defined and used within the OMC-ARE project (OMCARE), and those discussed in (Bozzano et al. 2008) and in (Cimatti, Guiotto, Roveri 2008).

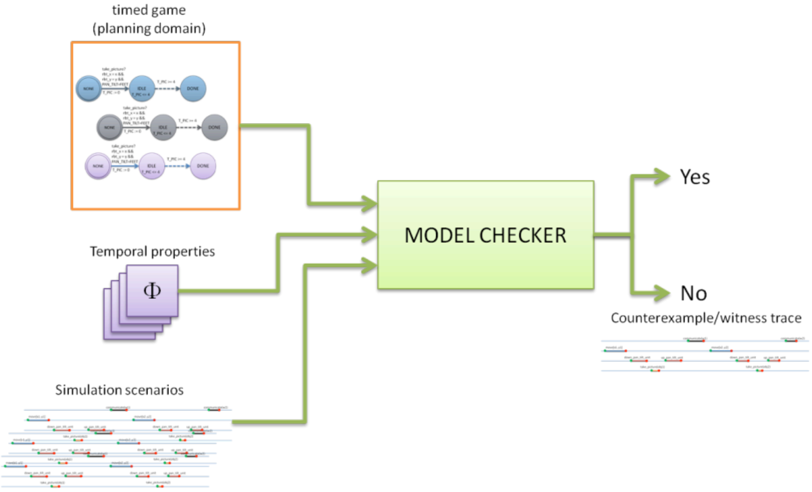

Figure 2. Basic model checking approach

IV. Validation and Verification

For model and plan validation and verification is based on symbolic model checking techniques exploiting Satisfiability Modulo Theory (Audemard, Cimatti, Kornilowicz, Sebastiani 2002), (Bozzano et al. 2005a), (Bozzano et al. 2005b), (Bruttomesso 2008), (Cimatti, Griggio, Sebastiani 2008) and abstraction refinement (Clarke, Kurshan, Veith 2010), (Clarke 2003). As described previously we have formulated 4 types of plan definition (A1-A4) which cover the three autonomy levels (E2-E4) covered within this study project. These 4 plan types cannot be treated in the same way due to their structure/content and hence two algorithms have been devised to cater for them.

Model verification and validation is can be seen as a two sided process, one of validating the domain model against a set of expected behaviors and the other validating the domain model against a set of unexpected behaviors. When validating using the expected behavior approach the operator is verifying that the model is sufficiently defined to allow safe operation of the spacecraft. Validating using the unexpected behavior approach the operator is verifying that the domain model will not allow dangerous situations to occur due to faults in the model. The combination of both of these approaches allows the operator to perform what-if analysis by modifying one or more of the initial conditions, expected or unexpected behaviors, planning domain, planning problem or operations plan. Observing the effect of the changes made allows the operator to determine the validity of the models used and determine if the safety of the spacecraft would be jeopardized.

During plan verification and validation a two-algorithm approach is used. When dealing with plans of type A1 to A3 we use specialized verification algorithms that are based on model checking methods (as illustrated in Figure 2). These algorithms aim to find paths of the operations plan that violate the goals set by the planning problem. If such a path exists then the operation plan is not a solution to the defined planning problem. Counter to this, if no such path exists then the operations plan is validated and is a solution to the planning problem.

It is a different situation when dealing with plans of type A4 because these plans can contain a mixture of sequences, similar to A3 plan types, and goals. An initial processing of the operations plan needs to be performed first to elaborate out the goals into sequences on the plan using the knowledge of the goal planner mechanism. Once this has been achieved the resulting plan can be viewed and treated as a type A3 plan, meaning that the same verification algorithms as used for A1-A3 plan can now be invoked to validate the operations plan. No further processing is performed for the parts of the original A4 operations plan which are already of an A3 class. Instead these are integrated with the results of the goal expansion. The important aspect in this expansion being that it should match the expansion that would occur onboard the actual spacecraft, meaning that the on-ground and onboard models have to be synchronized and the same processing mechanisms have to be available to the goal expansion routines.

C. Reasoning

For planning and scheduling we will exploit hybrid-game approaches based on a mechanism of generate and test (similar to the one of (Brafman, Hoffmann 2004), (Hoffmann, Brafman 2006)) but extended to the hybrid domain case), where we will use satisfiability modulo theory model checking techniques to generate a candidate solution, and then we will check, also with model checking techniques that the candidate solution is a real solution for the considered planning and scheduling problem. Figure 3 illustrates the overall reasoning strategy that is used within the system.

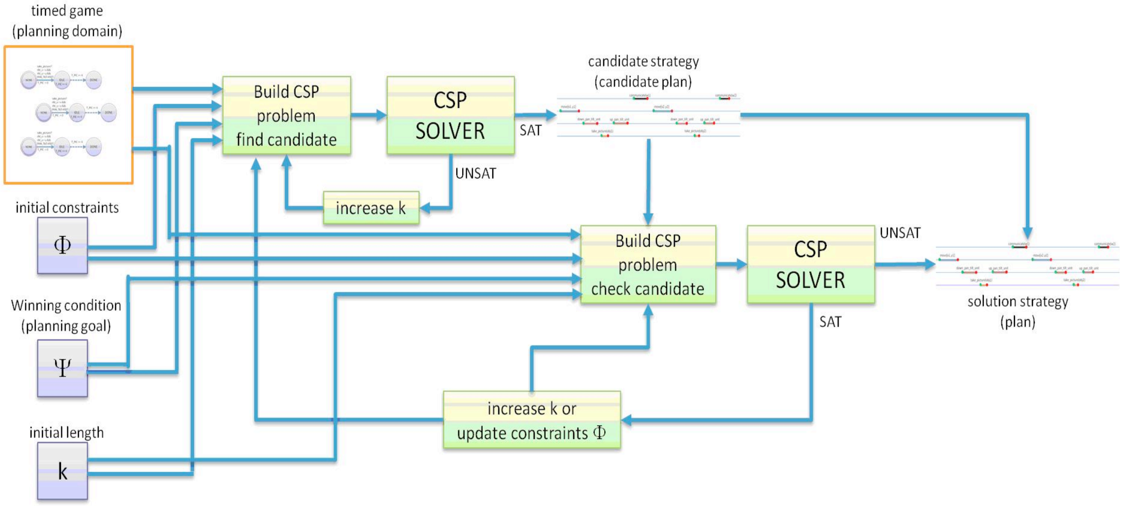

Figure 3. Goal based approach using the time-game principle

V. Re-use of Existing Software/Concepts

As part of the exploration of software that could potentially be re-used for this development, the initial APSI framework (Steel et al. 2009) was considered in the context of this study along with the more recent version of this framework (APSI v2). APSI offers a structured and flexible library for effective modeling and solving of the planning and scheduling problems in different domains. The framework is based on the concept of timelines representing the temporal evolution of the system and of the environment. An initial analysis shows that the framework should be extended to allow expressing non-deterministic effects at the discrete level, specifying uncertainty on the duration of activities, and uncertainty on the resource consumption/production within an activity. Moreover, we see an extension to the framework to model explicitly the sensors (i.e. the observations). As far as a goal language is concerned, the possibility to associate goals with some sort of preference scheme (e.g. mandatory, or nice to achieve if resource allow for) will also be assessed during the course of the study for its applicability.

The main reasoning capabilities of the prototype are provided by the NuSMV framework, which provides symbolic model checking in the form of pure binary reasoning. This is complimented by the MathSAT tool, which is an efficient SMT solver capable of handling conjunction constraints along with arbitrary Boolean combinations of theory atoms supporting a wide range of theories. To complete the process the algorithms of the NuSMT tool will be exploited, providing user-friendly mechanisms and dealing with Hybrid Systems.

For the graphical man machine interface (MMI) side of the prototype, the use of the EGOS user desktop (EUD) has been exploited to provide a common look and feel to that already being used within the ESA ground segment and other projects of the Agency. This environment is based on the Eclipse RCP technologies with tailoring for the EGOS frameworks and tools. This is augmented by the Eclipse Modeling Framework (EMF), Extended Editing Framework (EEF), Log4J and components of the Eclipse Nebula Project.

Graphical representation of the simulated plan and rover activities is provided by the 3DRov environment and is the basis of one of the two demonstration cases. The tool provides the end-to-end simulation for planetary rovers via 3D visualizations. It provides models for the mechanical, electrical and thermal subsystems of the rover along with the planetary environment. Scientific instrument models are also included to allow simulation of science-based scenarios. ESA's SIMSAT 4.1 simulator is the basis of the tool with its models designed to comply with the SMP 2.0 standard.

VI. Conclusion

IRONCAP is setting the ground for the planning and scheduling of operations and activities of future ESA interplanetary Rover missions. It is developing and evolving the concepts required to efficiently and successfully carryout rover operations at the three main levels of autonomy. It provides a prototype, which will bring the science and engineering situational assessments together into a common tool.

Future extensions/usage of IRONCAP could include the assistance of human space flight and surface operations on foreign worlds through autonomous robotic control by means of goal oriented control. Further functionality could include the addition of plug-ins to enhance the communications between the teams and team members by means of twitter or RSS like interfaces reporting to mobile devices.

These exciting aspects will need to be studied further to establish their feasibility and real world applicability outside the scope of this work. The project is still under development although approaching a conclusion with an end of the current project planned for November 2012.

Appendix A
Acronym List

APSI	Advanced Planning and Scheduling Initiative
BDD	Binary Decision Diagrams
CFDP	CCSDS File Delivery Protocol
EEF	Extended Editing Framework , Extended Editing Framework
EMF	Eclipse Modelling Framework
EMS	ESTRACK Management System
CGI	Ground Control Interface
GOAC	Goal-Oriented Autonomous Controller
LTP	Long-Term Plan
MCS	Mission Control System
MMI4EXPL	MMI for Exploration Missions
OBCP	On Board Control Procedures
OPS	Operations Preparation System
OSGi	Open Services Gateway initiative
PUS	Packet Utilization Standard
RCP	Rich Client Platform
ROCC	Rover Operations Control Centre
SMT	Satisfiability Modulo Theories
SWT	Standard Widget Toolkit
XMI	XML Metadata Interchange

Appendix B
Glossary

Action	An action is the most elementary brick in the on-board planning. An Action is defined as the parameterized specification of: • A data-flow in continuous time, which has an invariant structure along the whole duration of the Action. • A state based behavior associated to the satisfaction of preconditions, post-conditions and the detection of exceptions during the execution of the Action An Action involves only one Subsystem (e. g: Motion of the Mast Head, WISDOM checkout, etc.)
Activity	It is used as a term indicating either an Action or a Task or a sequence of actions and/or tasks. Due to the difference in content and the way they are managed by the mission planning, they are usually classified in: • Engineering activities: related to the maintenance of the rover and communications with an orbiter or the control centre. They used to be mandatory (i.e. they have to be executed with precedence over any other activity). • Science activities: these are the ones defining the scientific experiments. Its actual execution will depend on the availability of resources and the fulfillment of certain preconditions (such as be close to the target, light conditions, certain attitude).
Camera (in the context of the 3D View)	Cameras are elements of the 3D scene and placed accordingly based on the images/products metadata. These cameras may be physically located on a rover, a robotic arm or a Lander.
Display	A RCP internal window inside the Eclipse or EUD framework.
Engineering Team	The Engineering Team represents the rover's operators and engineers responsible for controlling the rover from the Rover Operation Control Centre in such a way that the mission scientific objectives can be achieved within the mission life time. Thereby one of the main responsibilities of the Engineering Team is maintaining the rover in a safe and operational condition during the whole mission life time and possibly beyond it.
Footprint (in the context of the 3D View)	The footprint indicates the area covered by a potential photograph, not a view from inside the camera.
GIS	Information System that integrates, stores, edits, analyses, shares, and displays geographic information for informing decision making. GIS applications are tools that allow users to create interactive queries (user-created searches), analyze spatial information, edit data, maps, and present the results of all these operations
Initial Conditions	A set of variables used by a particular process, e.g. planning or simulation, which describes the rover's and its environment state.

Layer/Overlay	In the context of a GIS application. The different types of products defined are grouped spatially in what are called layers. While using the GIS application, the user will filter the available data by means of selecting layers to be displayed.
Long-Term Plan	The Long-Term Plan is a data structure aimed at • Storing data which validity goes beyond the range of a single tactical planning session and • Providing resource budget estimation to the Science Team for the tactical planning session. The resource budget estimation gives the Science Team a sense of how many observations they can request in the next tactical planning session.
Operations Plan	Operations Plan is the data structure which specifies either explicitly or via goals (which have to be planned on-board) the operation procedures of the rover.
Science Team	The Science Team in the IRONCAP context represents a scientific institution responsible for specification and definition of the scientific experiments performed by the rover. Note that there might be more than one science teams involved in the Operations Planning process.
Target	Geographical location (i.e. an x, y, z coordinate) representing the actual place where an experiment has taken place, or from where an image has been taken.
Task	A logical and temporal composition of Actions and other Tasks including logic for making checks and decisions. It is formally defined in its most complete form as: • A set of pre-conditions which need to be fulfilled before the main body of the Task starts. • A main body, (nominal execution of the Task), composed of Actions, Tasks and conditions which fulfils the goal of the Task. • A set of post-conditions that induce the end of the Task. • A set of reaction rules to process every exception by a recovery handling body (this is a way to provide optional activities). • A pre-defined behavior for the logical co-ordination of the previous items: the main body of the Task is activated after satisfaction of the pre-conditions, and normally ends when the post-conditions are satisfied. If an exception occurs, this nominal execution is aborted and replaced by the specified recovery body. A Task is executed across two or more subsystems and as such will include presumably decision logic.
View	The contents of a Display excluding the toolbars and actions.
Waypoint	Geographical location (i.e. an x, y, z coordinate) part of a trajectory (either real or simulated)
Workspace	A set of configurations to be saved and loaded that will define the

present elements shown related with the application (e.g. Displays, Views, Properties, Parameters, etc.)

References

Engelmore, R., and Morgan, A. eds. 1986. *Blackboard Systems*. Reading, Mass.: Addison-Wesley.

Brafman, R. I., Hoffmann, J., “Conformant Planning via Heuristic Forward Search: A New Approach”, in: Proceedings of the *14th International Conference on Automated Planning and Scheduling*, Whistler, Canada, June 2004.

Hoffmann, J., Brafman, R. I.: *Conformant planning via heuristic forward search: A new approach*. Artif. Intell. 170(6-7): 507-541 (2006)

Audemard, G., Cimatti, A., Kornilowicz, A., Sebastiani, R., *Bounded Model Checking for Timed Systems*. FORTE 2002: 243-259.

Bozzano, M., Bruttomesso, R., Cimatti, A., Junttila, T., van Rossum, P., Schulz, S., Sebastiani, R., *An Incremental and Layered Procedure for the Satisfiability of Linear Arithmetic Logic*. TACAS 2005a: 317-333.

Bozzano, M., Bruttomesso, R., Cimatti, A., Junttila, T., van Rossum, P., Schulz, S., Sebastiani, R., *The MathSAT 3 System*. CADE 2005b: 315-321.

Bruttomesso, R., Cimatti, A., Franzen, A., Griggio, A., Sebastiani, R., *The MathSAT 4SMT Solver*. CAV 2008: 299-303.

Cimatti, A., Griggio, A., Sebastiani, R., *Efficient Interpolant Generation in Satisfiability Modulo Theories*. TACAS 2008: 397-412.

Clarke, E. M., Kurshan, R. P., Veith, H.,: *The Localization Reduction and Counter-example Guided Abstraction Refinement*. Essays in Memory of A. Pnueli 2010: 61-71

Clarke, E. M., Fehnker, A., Han, Z., Krogh, B. H., Stursberg, O., Theobald, M.,: *Verification of Hybrid Systems Based on Counterexample-Guided Abstraction Refinement*. TACAS 2003: 192-207

OMCARE: On Board Model Checking Autonomous Reasoning Engine. https://es.fbk.eu/projects/esa_omc-are/

Bozzano, M., Cimatti, A., Guiotto, A., Martelli, A., Roveri, M., Tchaltsev, A., Yushtein, Y., On-Board Autonomy via Symbolic Model Based Reasoning. In 10th ESA Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA'2008). 11 - 13 November 2008, ESA/ESTEC, Noordwijk, The Netherlands.

Cimatti, A., Guiotto, A., Roveri, M., On Board Model Checking for Space Applications. In ESA Workshop on Avionics Data, Control and Software Systems (ADCSS). 29 - 31 October 2008, ESA/ESTEC, Noordwijk, The Netherlands.

Steel, R.; Niezette, M.; Cesta, A.; Fratini, S.; Oddi, A.; Cortellessa, G.; Rasconi, R.; Verfaille, G.; Pralet, C.; Lavagna, M.; Brambilla, A.; Castellini, F.; Donati, A.; and Policella, N. 2009. Advanced Planning and Scheduling Initiative: MrSPOCK AIMS for XMAS in the Space Domain. Submitted to the 6th International Workshop on Planning and Scheduling for Space, IWPSS-09, July 19th -July 21st 2009, Pasadena, CA.