ADE: AUTONOMOUS DECISION MAKING IN VERY LONG TRAVERSES

Virtual Conference 19–23 October 2020

J. Ocón[1], I. Dragomir[1], A. Coles[2], A. Green[2], L. Kunze[3], R. Marc[4], C. J. Perez[5], T. Germa[6], V. Bissonnette[6], G. Scalise[7], M. Foughali[8], K. Kapellos[9], R. Dominguez[10], F. Cordes[10], G. Paar[11], G. Reina[12], A. Kisdi [13]

[1]*GMV Aerospace and Defence, Isaac Newton 11, Tres Cantos, 28760 Spain, {jocon,idragomir}@gmv.com*
[2]*King's College London, Bush House 30 Aldwych, London, WC2B 4BG, UK, andrew.coles@kcl.ac.uk*
[3]*3Oxford University, Parks Road, Oxford, OX1 3PJ, UK, lars@robots.ox.ac.uk*
[4]*Airbus Defence and Space, Gunnels Wood Road, Stevenage, SG1 2AS, UK, gnc.uk@airbus.com*
[5]*Universidad de Malaga, Dr. Ortiz Ramos, Malaga, 29071 Spain, carlosperez@uma.es*
[6]*Magellium, Rue Ariane 1, Ramonville St Agne, 31520 France, {thierry.germa,vincent.bissonnette}@magellium.fr*
[7]*Thales Alenia Space Italia, Antica di Collegno 253, Torino, 10146 Italy, genny.scalise@thalesaleniaspace.com*
[8]*VERIMAG, 700 Avenue Centrale, St Martin d'Heres, 38041 France, mohammed.foughali@univ-grenoble-alpes.fr*
[9]*Trasys International, Interleuvenlaan 10, Heverlee, 3001 Belgium, konstantinos.kapellos@trasysinternational.com*
[10]*DFKI RIC, Robert-Hooke-Str. 1, Bremen, 28359 Germany, {raul.dominguez,florian.cordes}@dfki.com*
[11]*Joanneum Research, Steyrergasse 17, Graz, 8010 Austria, gerhard.paar@joanneum.at*
[12]*Politecnico di Bari, via Amendola 126, Bari, 70126 Italy, giulio.reina@poliba.it*
[13]*GMV NSL, HQ Building, 77. Thomson Avenue, Oxfordshire OX11 0QG UK akisdi@gmv.com*

ABSTRACT

ADE (Autonomous DEcision making in Very Long Traverses) is an ongoing H2020 project aiming to develop key technologies for future space robotics missions. The focus of the project is to develop an exhaustive autonomous system for planetary rover exploration that can traverse at least 1km/sol, perform and maximize the scientific data collection, take appropriate decisions in nominal and contingency situations and ensure its safety at any moment.

In this paper we describe the system developed and its capabilities, along with the validation approach and testing facilities.

1 INTRODUCTION

ADE (Autonomous DEcision making in Very Long Traverses) [1] is an ongoing European H2020 research project, part of the PERASPERA second call [2]. Its ambition is to design, develop, and test in a representative analogue a fully autonomous rover system, inspired by the Mars Sample Fetching Rover. The ADE system is capable to take all decisions to pursue mission objectives, to increase data collection and overall science, to perform autonomous long traverse surface exploration, to guarantee fast reaction and adapt to unforeseen situations, increasing mission reliability, and guaranteeing optimal exploitation of resources.

The architecture of the ADE demonstrator consists of three high-level subsystems as follows: (i) the rover system capable of performing the considered features, (ii) a ground segment capable of commanding and validating the rover operations, and (iii) testing facilities for executing and validating the system in a Mars analogue environment. This paper presents the ADE design and implementation, along with the validation assets. More specifically, we will describe 1) the challenges and ambitions for the ADE project, 2) the different subsystems/components are the interactions between them and 3) the validation approach, before concluding.

2 CHALLENGES AND AMBITION

The mobility range capability of the "flown" surface rovers has been up to date strongly limited to few tens of meters per day [3][4][5]. From a purely technical point of view, this limitation is mostly due to the rover locomotion system and its power storage capabilities from one side, and the other by the lack/reduced skills in terms of autonomous capability to take decision onboard. The result is the impossibility to cover reasonable portions of/or multiples geographical areas of a potential planetary surface, reducing drastically the data returns both in terms of "pure science" and/or potential data collection for in-situ resources analysis and further exploitation.

ADE's main objective is to design, develop and test key technologies suitable to overcome these limitations, performing long traverses while guaranteeing fast reaction, mission reliability and safety, and optimal exploitation of the robot's resources within reasonable costs. ADE also explores new paradigms for telecommanding from ground station, ground truth methods, and soil traversability. More specifically, the project has the following objectives in mind:

1. **To achieve reliable autonomous long-range navigation (up to 1 km in 6 hours)** via short/long term mapping, path planning and motion control over long distances, re-planning traverse paths in case unpredicted events or conflicts occur. This is performed thanks to a complex guidance system, based on experience of previous developments for the Exomars rover mission [4] combined with an efficient and accurate perception and localization system. Such a localization system, based on stereovision, shall have a positioning error lower than 1.5 % with respect to the distance traversed.
2. **To maximize the scientific return along its nominal long traverse journey**. This is done via the autonomous detection of interesting data along the mission path. A scientific detector that uses AI techniques for pattern recognition is responsible for the detection of both known classified classes of targets as well as novel targets. The scientific detection shall be performed in seconds, to allow the rover to change its course dynamically to gather science.
3. **To perform autonomous management of the mission plan**. Based on high-level goals sent from ground, the system can plan on-board, and schedule in real time the low-level commands required to command the rover and achieve the mission's objectives. An on-board planning and scheduling system based on the ERGO framework [6] allows mission goals to be achieved autonomously.
4. **To ensure the safety of the rover under any circumstances**: ADE uses the SBIP toolset [7], which allows the definition and generation of FDIR components to increase the resilience against faults.
5. The system has to reach a **high level of maturity and flexible design**, capable to be instantiated for many different robots, having in mind not only space but also terrestrial applications.
6. The ADE system **is based on previous frameworks** developed in the frame of the first call of the PERASPERA SRC activity [2]. The idea behind this reuse is to enhance these components, and assess their maturity.
7. ADE has also the **objective of a spin-off for ground exploitation**, a secondary use case aimed to develop a robotic system for nuclear plant decommissioning activities.

3 COMPONENTS

As explained in the introduction, the ADE system for the space scenario can be considered a large, complex system, composed of an exploration rover, a ground control system that includes a simulator, and a set of test facilities with dedicated infrastructure. A large team of partners is responsible for the different systems, and a coordinator of the activity is responsible for the integration of all these systems. In the following, we will describe these components in detail.

4 ROVER SYSTEM

4.1 Robotic Platform

The SherpaTT rover (Figure 1) is a four-wheeled mobile robot with an actively articulated suspension system and a robotic arm. The five limbs of the system add up to 26 active Degree of Freedom (DoF) in total, five in each of the four legs, and six DoF in the manipulator's arm.

A standard i7 PC running Linux is used for locomotion and high-level control implementation, this is the so-called On-Board Computer (OBC).

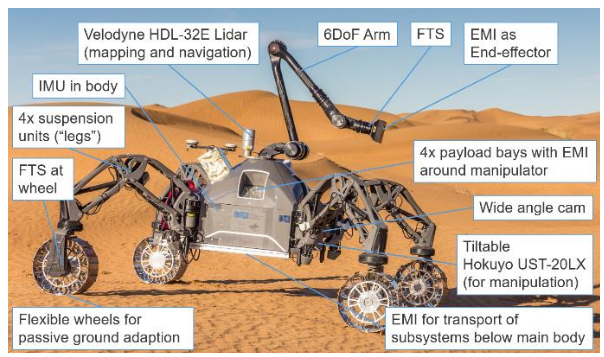

Figure 1: The Sherpa TT robot.

4.2 ADAM

In ADE, the rover is equipped with a separate hardware/software (HW/SW) component, the so-called **Autonomous Decision-making and Action-taking Module** (ADAM). This component provides on-board autonomous functionalities (such as autonomous guidance, mobile manipulation, localization and perception, and mission planning). ADAM uses additional rover avionics consisting of an Ultrascale+ ZU09EG board and an Intel-i7 computer. This hardware is also complemented by one stereo camera mounted on a pan-tilt unit on top of a mast for navigation, a second stereo camera for localization, and an avionics box encompassing the processing elements, sensors, and connectors.

The ADAM SW has been developed following the ERGO architecture [6] and is composed of two main SW layers: a functional layer and an agent.

The ADAM SW design is performed with the TASTE framework [8]. TASTE is a framework used to define and integrate components and their interactions into different executables, and deploy them on the ADAM processors. TASTE automatically generates glue code, as well as the executables, while ensuring important properties of the software such as proper delivery and reception of messages. TASTE is the middleware of ESROCOS [9], a previous PERASPERA project.

4.2.1 Functional Layer

The **Functional Layer,** depicted in Figure 2, interfaces via low-level primitives with the rover, manages sensors and actuators, and provides HW and vehicle abstraction. It also contains SW for autonomous functions that have associated hard real-time constraints, such as Rover Guidance (RG), the localization and perception subsystem, the mobile manipulation subsystem, or the Fault Detection, Isolation and Recovery (FDIR) subsystem. These are explained hereafter.

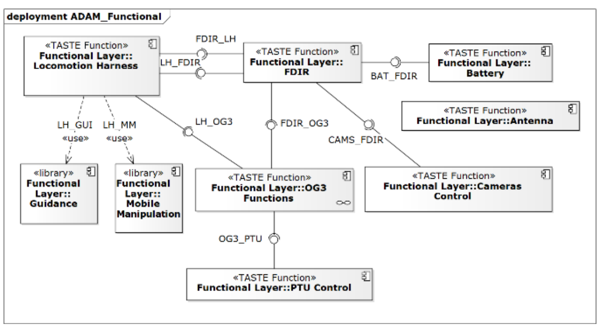

Figure 2: ADAM Functional layer.

The **Perception and Localization system** provides a vision-based localization estimate for a given reference frame by fusing data from the conventional sensors (IMU and odometry) and 2 stereo benches (NavCams and LocCams) providing relative and absolute positioning primitives, and images from stereo cameras. It also creates elevations map of the surrounding area based on a synchronized image pair captured by a calibrated NavCams.

The **Rover Guidance system** (Figure 3) is responsible for the autonomous navigation of ADE. Rover Guidance is in charge of the navigation map, local and global path planning, trajectory control, hazard prevention, and resource estimation to reach long traverses while ensuring rover safety at all times during mission duration. Rover Guidance uses different modes depending on the level of difficulty of the terrain that the rover is traversing [10].

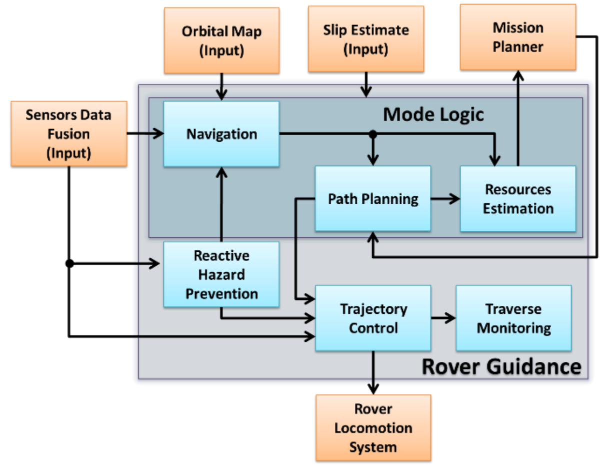

Figure 3. Architectural overview of Rover Guidance, including the external components and their connections.

One of the proposed and implemented modes is “blind drive mode” (L0) in which the rover does fewer safety checks and stops less often. This mode would include slip checks to make sure the rover is not embedding in sand based on Visual Odometry and Wheel Odometry comparison. Ideally, this guidance mode is designed such that the rover would drive quasi continuously stopping just for perception. The proposed mode enables us to traverse flat, completely safe terrains in a much shorter timeframe enabling much longer traverses in a single sol.

Another mode of operation is local reactive (L1), in which the system relies exclusively on the fixed (navigation) stereo cameras for driving, with a hazard avoidance function in charge of ensuring a safe traverse.

Finally, local guidance (L2) mode allows the rover to traverse difficult terrains. This mode uses the stereo cameras mounted on top of the mast and uses combined images to compose a local DEM of the area that is in front of the rover. Since the terrain is highly cluttered with hazards, relative localization accuracy needs to be high with high spatial frequency updates. This leads to driving at a slower speed.

Generally speaking, the RG system will transition to higher navigation levels according to the orbital map tile category. In higher levels, the rover spends more time on terrain perception, data processing, and drives with slower speeds compared to L1 for example. This is expected to be utilized only for specific scenarios, and the global planner's task is to maximize the time spent in lower modes.

As shown in Figure 3, perception and localization outputs are used by the Rover Guidance system to perform local path planning and hazard prevention. The perception and localization system also computes the slip estimation while driving.

The **Mobile Manipulation** component is a unique feature of ADAM. Once the rover is close to the sample location, this component is activated to perform the last steps. They consist of reaching and scanning the sample area taking into consideration the error committed by the rover during the traverse. The problem is being solved using algorithms dedicated to mobile manipulation. Two main objectives are defined for this component: to reach the sample area avoiding obstacles during the final traverse, and once reached the sample location, perform a scan to analyze the sample composition by using a simulated spectrograph. All these tasks are to be carried out optimizing rover and manipulator motions.

The **Fault Detection, Isolation and Recovery component** is developed and verified within the SBIP Toolbox [7]. SBIP provides an environment to specify and verify formal models using, respectively, the Stochastic Real-Time BIP language SRT-BIP and the BIP Statistical Model Checker BIP-SMC. SBIP features a compiler to generate C++ code from SRT-BIP specifications and an engine to it. An offline environment simulates and verifies the correctness of FDIR under both nominal and faulty behaviors. Once the FDIR component is verified offline, its C++ counterpart is generated, wrapped within the overall TASTE model and executed. To account for the complexity of the FDIR behavior, the FDIR component in ADE encapsulates two FDIR subcomponents in charge of different faults and strategies, and a number of "Input" and "Output" components in charge of, respectively, receiving nominal/faulty messages and sending strategies from/to the involved robotic components. These components use parametrizable queues to enable the asynchrony of communication.

4.2.2 Agent

On top of the functional layer, a software agent is responsible for the autonomous management of the mission by the rover. It performs on-board mission planning, scheduling of existing plans, and scientific detection of interesting events. The agent can be commanded from ground using different levels of autonomy. The levels of autonomy correspond to E1 to E4 ECSS standards, that is, direct telecommanding (E1), time-tagged commanding (E2), event-driven (E3), or goal commanding (E4) [11]. Embedded into the agent we find the following sub-components.

A **Mission Planner**, capable of dynamically elaborating mission plans to achieve high-level goals received from ground. Other goals received from an on-board scientific detector require the plan to be adapted dynamically. In such an over-subscription scenario, it is expected that in most cases the planner will not be able to find a planning solution for all goals requested, and will have to discard some. The on-board planner will receive a list of goals to be achieved and a representation of the current status as a list of observations and will return a list of low-level goals that will be achieved and the associated decomposed plan. The core of this component is the Stellar planner [12]. The Stellar mission planner, first developed in ERGO combines concepts and notions from classical and timeline-based planning. Within ADE, several enhancements to Stellar were performed, namely anytime search (the planner can keep on searching for a better solution even though it already found one), oversubscription planning (it can optimise the combination of goals to be achieved, based on the cost of achieving goals, and their relative scientific value), validating or completing given seed plans (given an existing problem, and a planning solution, the planner can determine the validity of the partial plan, or can complete it if it is necessary). Planning and re-planning are performed in real-time so that the deliberative layer must make a balanced judgement between time spent planning and time spent executing: with the combination of replanning and oversubscription allowing the planner to dynamically adjust which goals are to be achieved, depending on whether execution goes better or worse than anticipated.

The scheduling and execution of the existing plan will be performed by a **Mission Planner Reactor** running in the agent that dispatches the low-level commands to the functional layer. This component is based on the ERGO framework and has been enhanced to accommodate the improvements in the planning layer.

Finally, a **Scientific Detector**, as part of the agent, maximizes the scientific outcome of planned missions by identifying and assessing potential targets to observe. The scientific detector can process images with multiple channels and uses high-resolution color images as well as thermal infrared images to detect scientific targets of interest. In this work, we consider two kinds of scientific targets:

- **Classified targets (known)**. These targets have been previously identified by experts in the field. That is, there exists training data from previous missions that sufficiently characterizes a set of classes through their features and/or through a set of examples (e.g., image masks).
- **Novel targets (unknown)**. These scientific targets which are potentially interesting have not been characterized in advance of the mission. However, they can be identified as they are significantly different from the data that has been collected in previous missions.

Our approach is based on state-of-the-art machine learning methods, in particular convolutional variational autoencoders (VAE). The scientific detector performs the following tasks:

- **Region proposal generation**. Interesting/salient regions are detected in an input image using a selective search [13]. Selective search over generates region candidates within an image that are then processed in subsequent steps.
- **Novelty detection**. A novelty detector evaluates if a target is known or unknown. To do this, a trained variational autoencoder [14] is used to encode and decode the region proposal. As in [15], the reconstruction error, as well as latent space losses, are used as an indication that the target is novel (unknown). Meaning, that the target is significantly different from the previously observed targets (training data set).
- **Classification**. A set of VAE-based trained models evaluate each region's proposal to determine its class. The classification is based on a semi-supervised approach in which convolutional neural networks (CNN) are trained on different losses (reconstruction loss, latent space loss, and mixed losses) which result from the VAE.
- **Consolidation of results**. Finally, the results of previous steps are consolidated and all known and/or unknown targets are reported in form bounding boxes (including a score and/or a measure of uncertainty). Outputs of the scientific detector are used by the planner to add new goals to observe targets of interest previously identified.

The work for scientific detection consists of an offline process to label the images and train the neural networks, and an online process consisting of developing and testing the scientific detector that will run on-board. For the labelling of the images, specific workflows have been developed under Zooniverse [16].

5 GROUND SEGMENT

Two on-ground components have been developed within ADE: the **Ground Control Station** and the **Rover Simulator**.

5.1 Ground Control Station

Commanding of the ADAM and Rover System/Simulator is done with a **Ground Control Station**. This component allows creating the necessary elements for commanding the system and achieving the target of the planetary scenario: time-tag telecommands, OBCP, plans, and high-level goals. Depending on the autonomy mode, the ground control station can work in the following configurations.

GM1 Interactive mode (E1-E2 levels of autonomy). This ground station mode is aimed to operate the Rover System in E1 (direct telecommanding), and E2 (time-tagged commands). This mode allows the operator to request the execution of individual robotic tasks, acquiring and assessing, online, housekeeping telemetry and products monitoring the evolution of the execution of the telecommands.

GM2 Semi-autonomous mode (E3). In this mode the operator can compose a plan on ground to satisfy a set of high-level goals. An on-ground planner (more precisely, an instance of Stellar running on ground, the same planner used on-board in E4) will help the operator to generate a low-level sequence of activities. Also, the operator is able to introduce changes to the mission plan. The planner is then used to validate the modified plan locally, or to re-generate a new plan to be uploaded to the robotic system. The generated plan can be tested beforehand using the Rover Simulator, and transmitted later on to the on-board controller for execution. This plan is sent as a sequence of timelines transitions across time. The proposed scheme follows the mixed-initiative concept developed in [17]. This mode of operation corresponds to the mode used in previous/ongoing missions, such as the MER rovers, or Exomars.

GM3: Goal commanding mode. This is the most advanced mode, in which the ground control station will let the rover elaborate its mission plan on-board. Supported by a dedicated view (Figure 4), it consists of:

- Specifying the robotic system operations in terms of high-level goals (via the man-machine interface).
- Generating the plan to be uploaded to the robotic system, in the form of a file containing only the high-level goals to be achieved, together with an associated time window for them.

- Transmitting the plan to the on-board controller for execution.
- Importing, after completion of operations, the downlinked data (e.g., telemetry, images).
- Providing the necessary means to assess the downloaded data to plan the next goals to be sent.

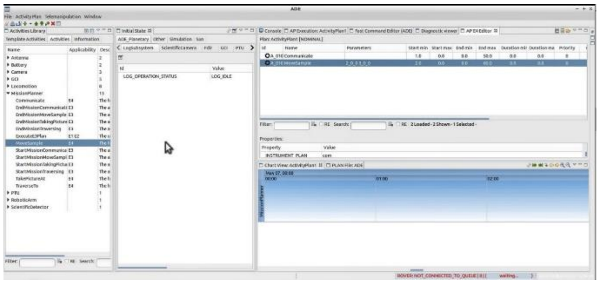

Figure 4: View of GCS goal commanding mode.

5.2 Rover Simulator

The verification and validation (V&V) approach to be used in ADE has as a key component **the Rover System Simulator** (RSS, Figure 5). Two components form the rover simulator:

- A first component is a rover's simulator ADAM instance, running on ground, which mimics the behavior of the on-board ADAM component running in the rover.
- The second component is the SherpaTT simulator (RSS-Sherpa), which reproduces the interaction of the SherpaTT rover with its environment. It has been extended to the needs of ADE in terms of sensors, avionics, and functionalities, such that the simulations are representative of the rover system.

Figure 5: SherpaTT simulator, while executing a testing session

The RSS as a whole has a double role: first, during the ADE development phase, the RSS substitutes the real robotic system allowing to validate and support the integration of new components and algorithms of the robot controller, the integration of the interfaces between the GCS and the robotic system and to rehearse the operations in a virtual environment. Secondly, during the ADE operations, the RSS allows validating the command products generated by the GCS to be uploaded to the rover, to ensure the feasibility of the plan generated by the GCS with the available resources, considering the rover configuration and the encountered environment. The simulator also allows us to generate simulated data to be compared with the downloaded ones for assessing the behavior of the rover and to generate reports for operations awareness.

6 TESTING FACILITIES, GROUND TRUTH, TRAVERSABILITY

The testing approach that has been followed in ADE consists of the following steps: first, unitary testing is conducted by each partner/subsystem provider. Once this has been achieved, iterative integration of the different components is performed. The rationale behind this iterative process is to integrate the individually developed components within the system as soon as possible, to identify and mitigate errors before the field testing. Then, virtual tests are conducted with the rover simulator. Once the rover simulator tests are considered satisfactory, preliminary tests are to be executed. At the moment in which this paper is being written, preliminary tests are planned for October 2020.

6.1 Preliminary Tests

The preliminary testing has been preceded by an integration of the different components in the Rover Simulator in the previous steps, for the same debugging purposes. Two separate preliminary test campaigns are planned to identify and correct as many errors as possible before the field testing.

6.2 Field Tests

The field testing is foreseen in Canary Islands, during December 2020 with a secondary, back-up date for January. Previous visits to the island allowed to gather relevant data (images for the scientific detectors, 2cm resolution DEMs, and a reconstructed 3D model), these data being used to verify the fulfilment of the requirements during the rover's traverses.

For the nuclear scenario, field testing is foreseen at GMV facilities during the same timeframe. In the nuclear scenario, a different rover is used, but it shares most of the components used in the space use case.

6.3 Ground Truth Data

A compilation of local ground truth areas (dozens of meters in square) has been possible by applying local photogrammetry from a low-flying drone or handheld image acquisition, supported by dGPS reference points sparsely distributed in the scene for high accuracy geocoding. The 3D reconstruction into a DEM grid and/or a watertight mesh was performed using COTS SW (e.g. Reality Capture). The result is an oriented image bundle and a DEM and Ortho image couple with a resolution close to the input image resolution. Table 1 summarizes the ground truth data sets.

Scene	Sensor	Size	Resolution	Notes
Area 4.4 medium quality	ESA UAV	1000 * 960m	49.5mm	Overall region to cover surrounding hills precisely registered to dGPS points
Area 4.4 high quality	ESA UAV	751 * 680m	28 mm	Same as above, Focus on Test Area 4.4
Rover Path medium quality	Hand-held SLR	120 * 70m	20mm	~150m long possible rover trajectory; sunlight
Rover Path highest quality	Hand-held SLR	25 * 23m	2mm	Portion of medium quality path; sunlight

Table 1: Ground truth for navigation and 3D reconstruction

6.4 Traversability Assessment

A traversability analysis of the terrain and the subsequent selection of a danger-free route are prerequisites to increasing the degree of autonomy for planetary rovers to navigate safely in unstructured environments and accomplish their tasks in long-range and long-duration applications. The importance of sensing hazards was highlighted in April 2005, when the Mars Exploration Rover Opportunity became embedded in a dune of loosely packed drift material [18]. The terrain geometry was not hazardous; however, the high compressibility of the loose drift material caused the wheels to sink deeply into the surface, and the combination of the drift's low internal friction and the motion resistance due to sinkage prevented the rover from producing sufficient thrust to travel up the slope. Opportunity's progress was delayed for more than a month while engineers worked to extricate it. A similar embedding event experienced by the Spirit rover in 2010 led to the end of its mobility operations [19].

Among the main challenges that are being investigated in ADE are:

- *Soil unevenness estimation*. This functionality relies on exteroceptive stereo-based sensing to characterize from a distance the ground geometry through the definition of two roughness parameters that are obtained from power spectral analysis of the reconstructed surface profile [20]. These parameters represent the exteroceptive signature of a given terrain patch.
- *Soil modelling and classification*. This task is performed using proprioceptive features to build a terrain classifier. First, a reduced optimal feature space is built. Then, a linear Support Vector Machine (SVM) algorithm is trained to classify incoming terrain patches. The functionality has been tested on data gathered by SherpaTT in previous campaigns [21].

7 CONCLUSIONS

ADE is designed and developed having a set of objectives in mind. Its main objective is to address the current challenges that planetary rover exploration has. ADE is a complex system-of-systems, in which each component is designed to fulfill a specific purpose for reaching the project's objectives.

A first capability provided by ADE is the possibility to command the rover using different levels of autonomy. This capability has involved developments not only at the space segment, but at the ground segment as well. ADE ground station allows commanding in different modes depending on the level of autonomy:

- *Direct telecommanding or time-tagged commanding* can be used when the intervention of the operators is needed.
- *Semi-autonomous* is the current state of the art for planetary exploration rovers. The rover simulator can be used to verify the execution of the plan before unlinking it.
- In contrast, *autonomous* mode is a new mode of operations, beyond the state of the art, in which the burden of elaborating the detailed mission plan is assigned to the on-board system.

All these features are possible in ADE thanks to a combination of assets: the ground control system, the planner and the simulator on ground, and the autonomy framework, the on-board planner, running on the rover.

The second capability in ADE is to allow long traverses. To overcome the limitations existing in current missions due to the lack of autonomy, a novel autonomous guidance system has been developed. This guidance system is complemented by a perception and localization system that provides the fused data needed by the rover guidance system. Also, technology for mobile manipulation allows the combined mobility of the rover platform and the robotic arm. The system is capable to perform opportunistic science by using a combination of the on-board planner and scientific detector. This increases scientific return, allowing a quick response to unexpected events.

Moreover, the ADAM SW component, the most important SW component on-board, uses novel techniques (model-driven design and development, provided by TASTE, formal verification techniques provided by BIP).

ADE is completed with components for Ground Truth as well as a Soil Traversability analysis.

All these capabilities will be demonstrated in a Mars analog scenario within the next months during the field tests. Similar tests will be conducted for the terrestrial use case, using different hardware equipment

Finally, ADE puts in action all the knowledge and SW assets inherited from previous PERASPERA SRC projects (ESROCOS, ERGO, INFUSE [22], and I3DS [23]), demonstrating its validity for its future use in space.

Acknowledgement

We would like to thank the European Commission and the members of the PERASPERA Programme Support Activity (ESA as coordinator, ASI, CDTI, CNES, DLR and UKSA) for their support and guidance in the ADE activity. We also want to thank Robert Deen and Gary Doran, from JPL, and Admin Wedler, from DLR, for their support as part of the ADE Advisory board. This project receives funding from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement No 821988.

References

[1] https://www.h2020-ade.eu/

[2] http://www.h2020-peraspera.eu/

[3] https://mars.nasa.gov/mars2020/

[4] http://exploration.esa.int/mars/

[5] JPL. Mars Exploration Rovers. Available at: http://marsrovers.jpl.nasa.gov/home/index.html.

[6] Ocón, J., et al (2018). The ERGO framework and its use in planetary/orbital scenarios, *Proceedings of the 69th International Astronautical Congress (IAC)*, Bremen, Germany, 1-5 Oct 2018

[7] Mediouni, B.L., Nouri, A., Bozga, M., Dellabani, M., Legay, A., Bensalem, S. (2018). SBIP 2.0: Statistical model checking of stochastic real-time systems, *Proc. of International Symposium on Automated Technology for Verification and Analysis (ATVA)*, p. 536-542.

[8] Perrotin, M., Conquet, E., Dissaux, P., Tsiodras, T., & Hugues, J. (2010). The TASTE Toolset: turning human designed heterogeneous systems into computer built homogeneous software. *Embedded Real Time Software and Systems (ERTS2)*.

[9] http://www.h2020-esrocos.eu/

[10] Marc, R., Weclewski, P., Lachat, D. (2018). Autonomous Multi-Mode Rover Navigation for Long-Range Planetary Exploration Using Orbital and Locally Perceived Data, *69th International Astronautical Congress (IAC)*, Bremen, Germany, 1-5 Oct 2018

[11] ECSS Secretariat. (ESA/ESTEC), "ECSS-E-70-11 Space Segment Operability" (August, 2005). Nordjwick, The Netherlands.

[12] Cashmore, M., Coles, A.I., Cserna, B., Karpas, E., Magazzeni, D., Ruml, W. (2018) Situated Planning for Execution Under Temporal Constraints, *Proceedings of the AAAI 2018 Spring Symposium on Integrating Representation, Reasoning, Learning, and Execution for Goal Directed Autonomy*.

[13] Uijlings, J.R.R. , van de Sande, K.E.A., Gevers, T., Smeulders, A.W.M. (2013). Selective Search for Object Recognition, *International Journal of Computer Vision*.

[14] Kingma, D. P., Welling, M. (2013). Auto-encoding variational bayes, *arXiv preprint*, arXiv:1312.6114

[15] Sintini, L., Kunze, L. (2020). Unsupervised and Semi-Supervised Novelty Detection using Variational Autoencoders in Opportunistic Science Missions, *Proceedings of the British Machine Vision Conference (BMVC)*.

[16] https://www.zooniverse.org/

[17] Bresina, J., Jonsson, A., Morris, P., Rajan, K. (2005). Mixed-initiative planning in MAPGEN: capabilities and shortcomings, *Proceedings of the workshop on mixed-initiative planning and scheduling, ICAPS*. Monterey, CA, 2005.

[18] Grossman, L. (2010). Atom & cosmos: Spirit stuck, but in good spot. *Science News*, 177(5), 7.

[19] Cowen, R. (2005). Opportunity Rolls out of Purgatory. *Science News*, 167(26), 413

[20] Reina, G., Leanza, A., Milella, A. (2020). Mind the ground: a power spectral density-based estimator for all-terrain rovers. *Measurement* 151, 107136.

[21] Dimastrogiovanni, M., Cordes F., Reina, G., (2020). Terrain Estimation for Planetary Exploration Robots. *Applied Sciences*, 10 (17), 6044.

[22] https://www.h2020-infuse.eu/

[23] https://i3ds-h2020.eu/