17th International Conference on Space Operations, Dubai, United Arab Emirates, 6 - 10 March 2023.

Validation approach for the CNES Autonomous Navigation solution accommodated on the ExoMars rover

Michel Delpech[1], Irene Valenzuela[1]

[1]*Centre National d'Etudes Spatiales (CNES), Toulouse, France*

michel.delpech@cnes.fr

irene.valenzuelamolina@cnes.fr

Abstract

The objective of ExoMars mission (ESA) is to land and operate on Mars the Rosalind rover to search for signs of past and recent life. Before the Ukrainian crisis broke out, the rover was to be launched in September 2022 by a Proton rocket and land in March 2023 in the Oxia Planum region. The current perspective is now to fly this mission in 2028 in order to allow the development of a new lander. As regards the rover, designed by Airbus D&S UK (ADS-UK), it will be operated in different modes including Autonomous Navigation (A.N.) to extend the traverse capability up to 70 m / sol.

In 2018, following a CNES / UKSA agreement to cooperate in the field of planetary rover autonomous navigation, a Contract Change Notice was approved by ESA for supporting the implementation of the CNES A.N. solution in complement of the already contracted ADS-UK solution. The presence of two Autonomous Navigation solutions on-board the rover has introduced some design and schedule constraints but brought higher flexibility and robustness due to the different characteristics of the solutions from the functional scope and performance perspective.

Today, the software development and validation for both solutions is achieved and the validation by the prime contractor Thales Alenia Space (TAS-I) has been performed using the Rover Ground Test Model. Specific challenges were faced during the course of the project due to the stringent reliability requirements imposed on the Autonomous Navigation functionality. Even though any abort of the A.N. software has no impact on the mission safety since the rover will stop and wait for operator instructions, it is of paramount importance to guarantee that the commanded path will never put the rover into a configuration that would lead to the loss of the mission or dramatically impair the rover motion capability.

The paper focuses on the validation approach adopted to fulfill the safety requirements while addressing the very specific technical and schedule challenges imposed to the whole software project.

Keywords: planetary exploration, rover autonomy, validation

1. Introduction

The involvement of the CNES robotics team in rover autonomous navigation dates back from the mid 90s with its participation to the Russia led Marsokhod project. CNES contribution comprised the stereo camera system and the Autonomous Navigation (AN) software package which was thoroughly optimised because of the limited computing resources at that time.

AN algorithms were subsequently improved and matured through several years of R&D development and validation effort including field testing in preparation of future European exploration missions. A consolidated patrimony of Autonomous Navigation functionalities (EDRES) was the result of the extend involvement of CNES in these activities.

After the reinsertion of the CNES AN Software in the Exomars project in 2018, CNES work has been mainly focused on adapting the existent AutoNav Software to comply with the project functional needs and architecture that should remain common between the two AutoNav solutions that will be present on-board. Each AutoNav solution runs on the ExoMars co-processor as a specific software, this co-processor includes two additional components: (1) the Standard Interface Layer, (2) The Visual Localization module (VisLoc) that is used during the Locomotion phases:

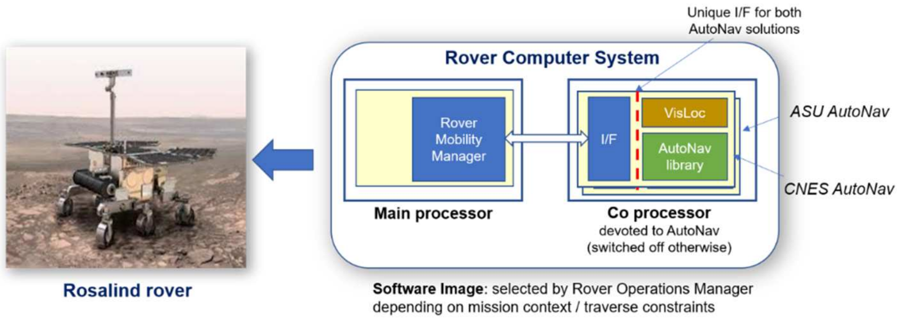

Figure 1 : Overview of the A.N. software accommodation context

The first section will introduce functional and performance requirements imposed to the software. Additionally, the functional scope of the autonomous navigation software and the software architecture will be described.

The next sections detail the software development approach (performed in consecutive steps) and the incremental validation process that is driven partly by the stepwise development and mostly by project schedule constraints.

Then, the specific approach adopted to demonstrate the fulfillment of the safety requirements is detailed.

Finally, test results will be presented and a summary of the current software status with some highlight on the hot issues will be provided in the conclusion.

2. CNES AN Software Description

The AutoNav Software will allow the Rover to perform autonomously traverses up to 70m per sol using a model of the local environment built through stereovision techniques. In order to assure a good quality of the computed local environment, each traverse will consist on consecutive navigation stops.

At each navigation stop the rover reacquires new information about its environment that will be used to ensure the rover safety over the next locomotion step. For Exomars, the maximum distance between two consecutive navigation stops is 2.4m. At each navigation stop, the rover will perform several stereo-image acquisitions rotating its StereoBench Pan and Tilt unit in order to cover a sufficient zone.

2.1. Functional and Performance requirements

The Autonav Software is expected to offer the two following modes of operation:

- **AutoTrav mode**: the ground defines the goal or a list of waypoints and the AutoNav module computes periodically a safe trajectory increment to the goal.
- **CheckPath mode**: the ground defines a trajectory that is to be executed accurately by the Rover. The AutoNav module is in charge of analyzing periodically the terrain traversability along the trajectory and signaling the presence of obstacles.

The techniques for generation and analysis of the traversability maps are common between the two operation modes, the main functional requirements imposed to the software are described in table 1.

Table 1. Functional Requirements

The AutoNav library shall include a functionality that:	
Perception Planning	performs perception planning according to the current context. This function shall return a list of Pan & Tilt angles for the stereo bench steering device (PTU).
Perception Check	checks the exposure of a stereo pair of images provided by the calling application and computes new camera settings if the exposure is not appropriate
Disparity Computation	performs disparity map computation using the current images.
DEM Computation	computes the local DEM using the current disparity map. If a DEM is already available from a prior perception at the same navigation stop, the local DEM is merged.
Map Computation	computes a local Traversability Map (NavMap) associated to the current DEM. If a NavMap is already available from a previous navigation stop, the local NavMap is merged with it.
AutoTrav Mode	computes a Path Sequence taking as input the traverse specification parameters and the current NavMap.
CheckPath Mode	checks the validity of a section of the Path Sequence provided by ground, using the current NavMap.

The safety of the rover is critical for the mission; therefore, special attention is given to the verification of the Safety requirements, described in the table 2:

Table 2. Safety Requirements

The path sequence delivered by the AutoNav library:	
Discontinuities	shall not cross zones where the terrain discontinuities exceed a specified value.
Slope	shall not cross zones where the terrain slope exceed a specified value.
Ground Clearance	shall not cross zones where the ground clearance is smaller than a specified value.
Boogie Angles	shall not cross zones where the boogie angles exceed a specified angular range.
Non allowed Areas	shall not cross zones that have been specified as non-allowed by the ground
Path Planning constraints	shall respect the path planning constraints specified by the ground.

Along with the functional and safety requirements, interfacing, performance, quality and validation and verification requirements are also taken into account; the most impacting ones are summarized here below:

Table 3. Other major Requirements

The AutoNav library shall:	
Configurability	allow the operator to configure the different parameters of the AutoNav.
Observability	generate data of the different states and store to allow monitoring and debugging activities.
Abort	be capable to interrupt its execution and return to a stable state within 6 s.
Restore	record in all relevant information to enable a restart the execution after an abort.
Leon2 Implementation	be validated on a Leon2 processor to provide realistic evaluation for space implementation.
Software Criticality	be compliant with the metrics of a category B flight software

2.2. Description of functional scope

CNES AutoNav Software includes all the needed functionalities to respond to the requirements previously described. The critical part of the system relies on existing algorithms developed in EDRES context and new functionalities were added to answer to some specific needs of the Exomars project.

The AutoNav functional workflow over a cycle in AutoTrav mode is presented in Figure 2. An overview of each functionality is presented in the following sections.

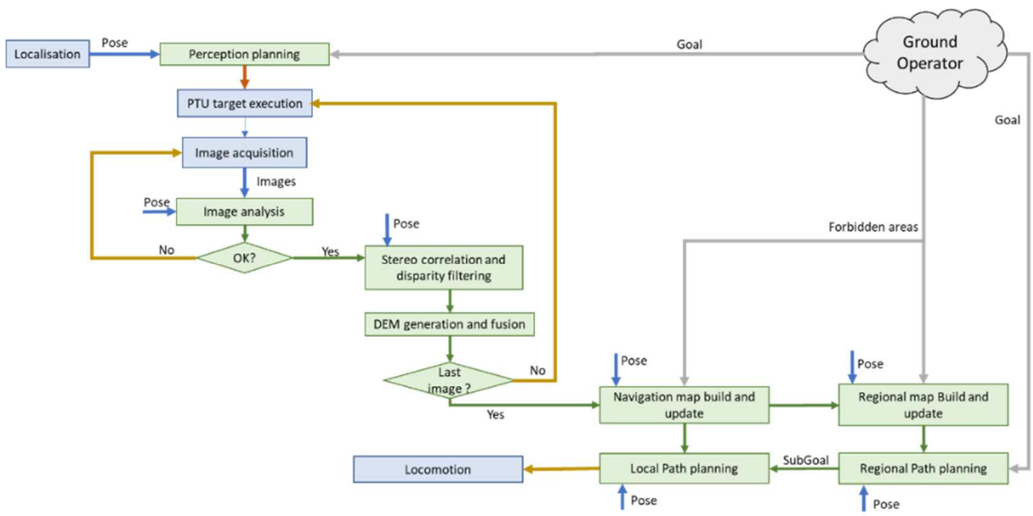

Figure 2 : Functional behaviour of the AutoNav over a cycle in AutoTrav mode

2.2.1. Perception Planning

This functionality provides a series of PTU angles for the perceptions to be acquired at a given navigation stop. In a nominal situation, three acquisitions are commanded for each Navigation Stop.

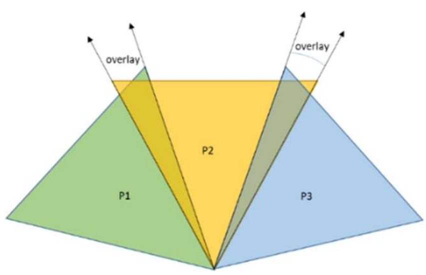

Figure 3 : Definition of a perception campaign containing three acquisitions

2.2.2. Image Analysis

This functionality assess whether the luminance level of both images is appropriate to achieve stereo correlation and determines, if needed, the new camera settings (exposure time and gain) to be applied in a new acquisition.

Some regions like the rover parts and the sky are excluded on the analysis to focus the assessment of the luminance level on the part of the terrain located within a few meters of the rover.

To determine if the image is properly exposed, underexposed or overexposed the histogram characteristics are compared with respect to the luminance bounds using a simple model.

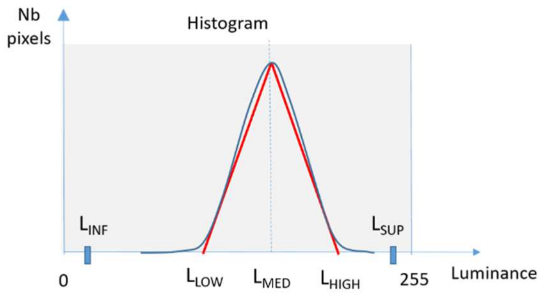

Figure 4 : Histogram characteristics

2.2.3. Disparity Map Computation

The stereo-correlation technique is applied to image of gradients that are computed from the input stereo-images. A two-stages correlation method is applied. This improves the computational efficiency with respect to a classic single-stage method and reduce the probability of erroneous matching.

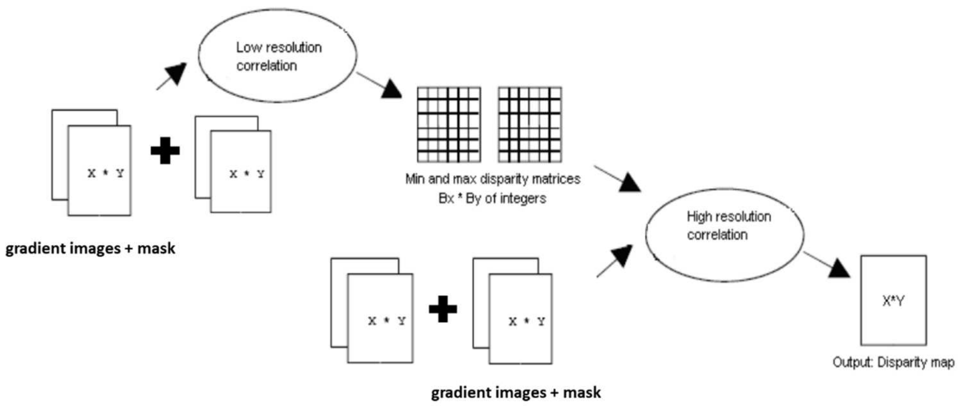

Figure 5 : Two-stages stereo-correlation

The output is a disparity map whose values are directly correlated to each pixel so that we have enough information to get a 3D image.

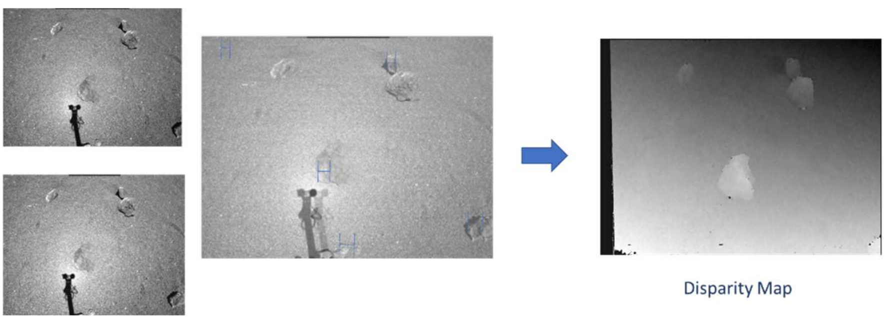

Figure 6 : Disparity map computation process

2.2.4. Digital Elevation Model (DEM) generation and fusion

This process consists in taking each valid pixel of the filtered disparity map and computing the 3D position of the associated point expressed in the DEM reference frame. It is performed in two consecutive steps:

17th International Conference on Space Operations, Dubai, United Arab Emirates, 6 - 10 March 2023.

- Computation of the 3-D point cloud coordinates in the Stereo Bench reference frame: each disparity value is converted into 3-D position using the Stereo Bench characteristics.
- Conversion of the 3-D point cloud into the DEM reference frame: this step implies the knowledge of the frame transformation between the DEM frame and the Stereo Bench frame.

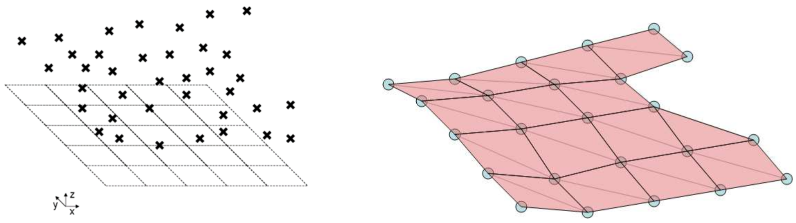

Figure 7 : 3D Point Cloud (left) and Visual Representation of the DEM (right)

2.2.5. Navigation map build

The main purpose of the Navigation Map is to identify and mark the different regions that must be considered as obstacles.

The map classification process involves the assessment of the following characteristics that are computed for every cell.

- o Discontinuity
- o Rover slope (Pitch and Roll)
- o Ground clearance of the rover parts (body and solar arrays)
- o Boogie angles (Left, Right and Rear)

The first step is to analyse discontinuity and worst-case slope characteristics using an Abstract Rover Model (ARM):

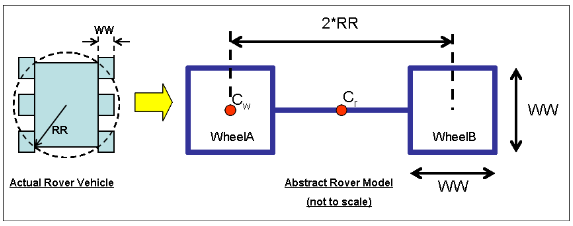

Figure 8 : Abstract Rover model

The second step is to estimate the attitude of the rover main body (defined by the pitch and roll angles), the boogie angles and its height expressed in the DEM frame. Then, it becomes possible to evaluate the risk of collision of the sensitive surfaces with the terrain. This is achieved by discretizing appropriately these surfaces and determining for the corresponding set of points the vertical distance to the DEM.

As the computational burden of estimating the rover configuration on the DEM using a full kinematic model would be too great for real-time applications on space processors, a simplified kinematic model is used to perform this analysis:

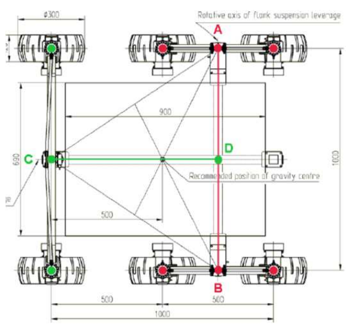

Figure 9 : Rover Configuration Determination Algorithm (RCDA)

To speed up the terrain analysis process, the algorithm relies on tables of precomputed data instead of performing calculations at each call.

For each characteristics, we compute the most conservative values over all possible rover headings. The final Local Navigation Map will merge the classification for each cell as shown in the figure below:

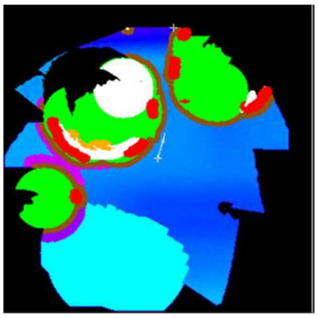

Figure 10 : Navigability Map. (Blue = navigable zones / Other colours = non-navigable areas)

Being the Safety of the Rover critical, a Safety Analysis is performed as presented in section 5. This analysis allows the computation of the margins to be added to the navigability limits to cover for the simplifications made during the navigability computations.

2.2.6. Regional Navigation map (only for AutoTrav mode)

The Local Navigation map contains useful information for the path planning, however, the fact of storing the navigation maps has a high impact in terms of memory usage if wide zones are stored.

In order to keep all the obstacles observed during the traverse with a minimized usage of memory, a Regional Navigation Map is generated using an obstacle vector representation. This improves the visibility of the past navigated zones that could be helpful in order to perform a return mission or to get out of difficult situations.

The regional navigation map computation follows the obstacle's contour from the Local navigation map and saves the contour information in the segmented map. A global regional map is created by merging the segmented maps of consecutives navigation stops.

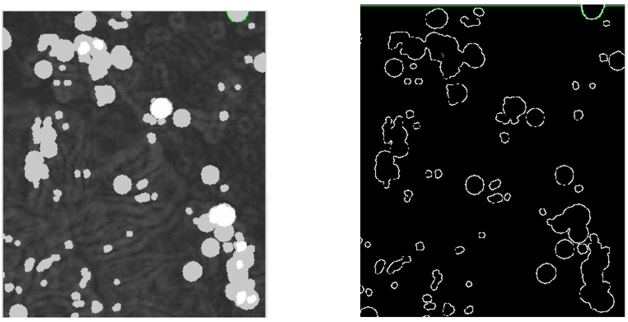

Figure 11 : Local Navigation bitmap (left) and Regional Navigation obstacle map (right)

2.2.7. Regional Path Planning (only for AutoTrav mode)

The Regional Path Planning manages the Regional Navigation Map and computes the optimised long-range path from the Rover position to a given Goal.

The Regional path planning is based on the tangent graph theory in which the nodes are constituted of Start Point, Goal Points and Tangent points between the start point, the goal point and the obstacles.

The interface between the Regional Path Planning and the Local Path Planning is a Local SubGoal, which is defined as the last navigable point of the Regional Path included in the navigable zone around the Rover.

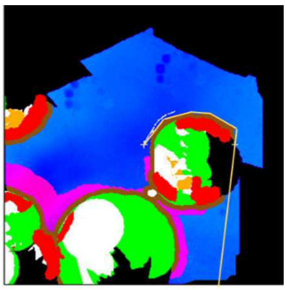

Figure 12 : Regional Subgoal to Local Navigation Computation

2.2.8. Local Path Planning (only for AutoTrav mode)

The algorithm consists in searching for the best valid path allowing to reach the Local Subgoal while satisfying the rover curvature and safety constraints. To reduce the dimension of the search space, the algorithm examines a finite number of path candidates that are represented as a series of curved arcs with a given length. The search is performed as following:

- Check the validity of a set of arc candidates starting from the current best node.
- Compare these valid arc candidates considering the estimated cost to reach the goal
- Store all valid arc candidates in the order of increasing cost and select the best one for the next step
- Backtrack in the search tree if an exploration step fails to yield valid candidates. The exploration starts again from the next best candidate.

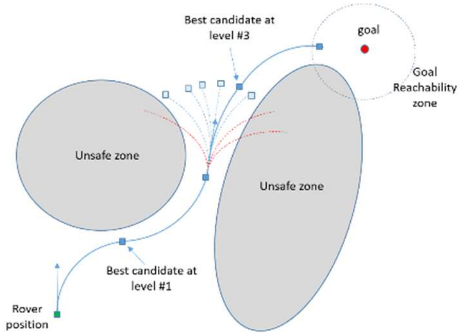

Figure 13 : Local Path Planning Algorithm

2.2.9. Check Path (only for Check Path mode)

This functionality analyzes the trajectory commanded by the ground (Path or Point Turn) and determines whether its execution is safe or not.

The processing to be performed depends on the type of trajectory commanded by the ground:

- **Path**: analysis of the safety of the rover to follow the oncoming part of a traverse defined by ground. Two approaches are implemented, conservative (full DEM and yaw angle analysis) and optimized (reduced number of DEM cells and yaw angles).
- **Point Turn**: analysis of the safety of the rover to perform a point turn in the current rover position.

The previous sections described the main algorithms allowing the Autonomous Navigation of the Exomars Rover. Additional algorithms are needed to perform initialisation, storing and monitoring activities among others. For the shake of clarity, these algorithms are not detailed in the present document.

2.3. Software Architecture

The software architecture is common to the two AutoNav solutions on-board, this enables the accommodation of the two A.N. solutions with minimal impact on the overall rover system.

Two processors are present in the software architecture:

- The Main Processor (PM): is in charge of the Rover System software including the Mobility Manager. The Mobility manager is responsible of sequencing the AN Activities.
- The co-Processor (coPM): is in charge of the Visual Localisation and Autonomous Navigation Functionalities processing. An Interface Software assures the correct communication between the PM and the coPM.

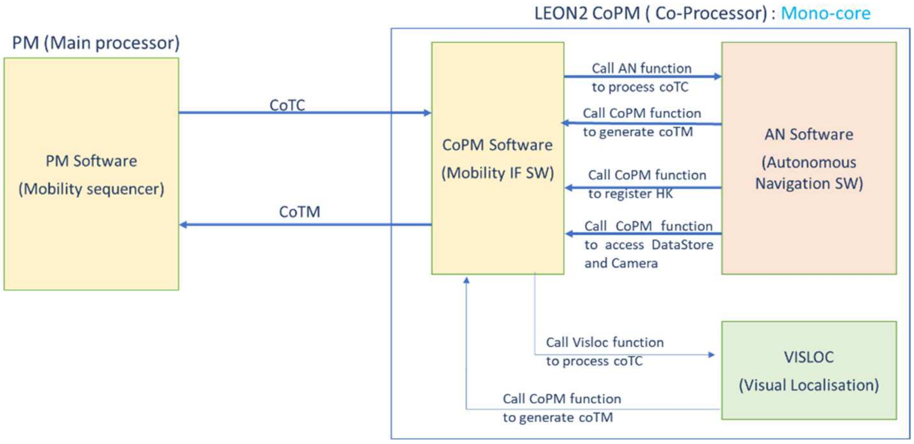

Figure 14 :Flight Software Architecture

The communication mechanisms between the PM and the coPM are based in SpaceWire communication via coTC/coTM definition. The AN coTC/coTM ICD is common to the two AN solutions, assuring the compatibility with the PM software.

A coTC is sent to the coPM to trigger the activation of a specific computation; the interfaces of the coTC can include information to be used by the coPM for the computation. Once the processing of the coTC has been completed by the coPM Software one or several coTMs can be sent back to the PM informing about the result.

For example, the coTC/coTM management for the image processing works as follows:

- The corresponding coTC will be sent by the PM including in its interface information about the memory slot in which the stereo images can be found and some characteristics about the image acquisition (temperature, PTU pose, exposure time, etc.)
- Using this information the AN Software starts the processing of the stereo-pair informing of the status of the computations via several coTMs that are sent back to the PM Manager (Success or Failed) during the processing.

Table 4. CoTC/CoTM example

coTc	coTM
coTC_78 : NavCtrl_doPerception	coTM_7_8 : Preliminary computation and configuration coTM_7_9 : Image retrieving and exposure analysis coTM_7_10 : Disparity map computation coTM_7_11 : DEM Terrain model computation coTM_7_12 : Navigability map computation

3. AutoNav Software Development Approach

The software development has been performed in consecutive steps to adapt the existing CNES rover autonomous navigation framework (EDRES) to the Exomars project.

This adaptation phase is performed along three main axes: addition of functionalities to meet the mission requirements and architecture, optimization of algorithms for the ExoMars specific geometry / kinematics and finally software tailoring to reach the desired quality level (B-Criticality as defined by ECSS).

Incremental versions of the software have been produced during the course of the project and required specific validation objectives and activities.

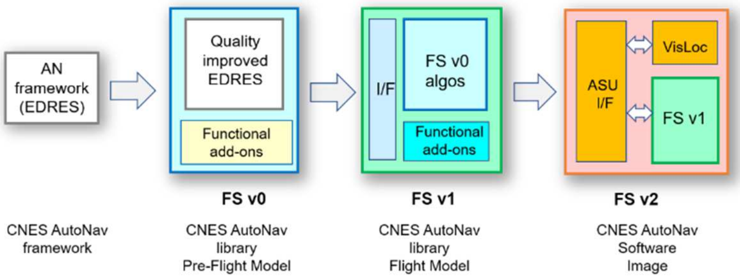

Figure 15 : AN Software Development Approach

1. **Software v0**: ExoMars CNES AutoNav library – Pre-Flight Model

This software is inherited for the most part from the CNES Autonomous Navigation framework (EDRES). The main modifications performed were the following:

- Architecture adaptation and memory management
- Software tailoring to maximize performance (LEON2 specifics)
- Software tailoring to satisfy ExoMars coding standards (first step)

2. **Software v1**: ExoMars CNES AutoNav library – Flight Model

This library designed to run in perfect compliance with the flight computer constraints, it is derived from the Software v0 through an adaptation along the following axes:

- Interface customization to satisfy ASU I/F specifications (coTC/coTM ICD).
- Complementary software tailoring to satisfy ExoMars coding standards (final step)
- Addition of mechanisms to maximize control flexibility, observability and maintenance

3. **Software v2**: ExoMars CNES AutoNav - Software Image

This software image is to be run on the ExoMars Co-processor and its production process can be summarized as follows:

- No source code evolution of the FSv1 library
- FS v1 library is linked with object codes implementing the Visual Localization module and the ASU component ensuring communication with the main processor

The distinction between v1 and v2 is necessary since the validation objectives and environment are different as presented in the next section.

4. AutoNav Validation and Verification Approach

The first validation regards the AutoNav v0 version and relies mostly on a numeric simulator developed by CNES, the second phase concerns the AutoNav v1 version and requires an adaptation of the test environment to include the emulation of the specific ADS-UK software interface and the mobility manager.

The third validation phase concerns the software image to be uploaded on board (Software v2). This validation implies to switch to a specific numeric simulator (NVSF) provided by ADS-UK. This simulator emulates the on-board computing architecture and the operational chain down to the operator level.

The final step before flight acceptance is performed by TAS-I at the Rover Operation Control Center facilities using the ExoMars Ground Test Model.

The validation workflow is the following:

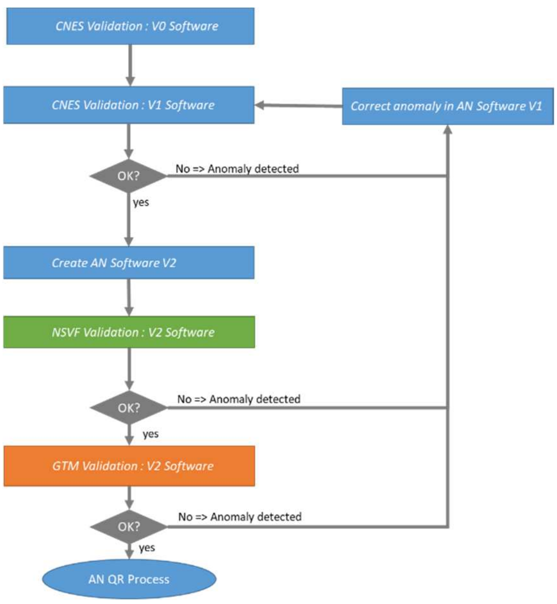

Figure 16 : AN Software Validation Workflow

4.1. CNES Validation : V0 Software

This validation allows to increase the confidence on the adaptations performed to the inherited AN components and to validate the add-on components introduced to cover Exomars needs. As the Software coTC/coTM interfaces were not formally defined yet, this validation step was important to de-risk the development and the planning.

This validation relies on a numeric simulator developed by CNES that allows performing Software-in-the-loop (SIL) and Processor-in-the-loop (PIL) campaigns.

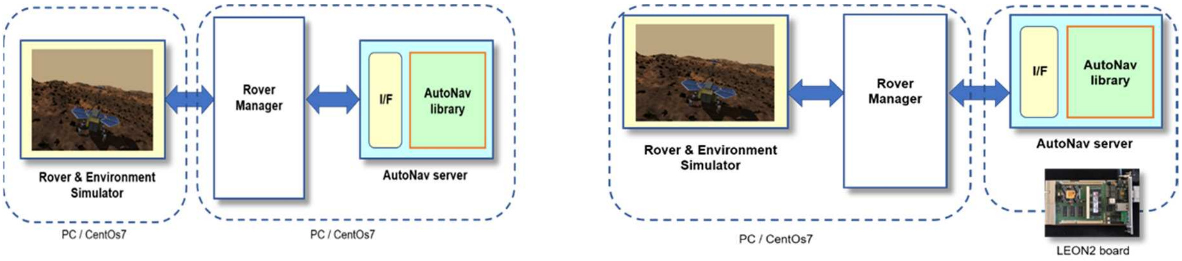

Figure 17 : Software-in-the-loop test bench (left) and Processor-in-the-loop test bench (right)

This preliminary validation consists mainly on:

- Unitary testing for the new or adapted functionalities
- Integration testing (SIL): Two cycles are analyzed to assure the correct integration of the AN Software and to validate the products and the sequencing.
- Functional testing to analyze the AN behavior and detect possible anomalies or improvements (SIL). Specific scenarios with different configurations are run to analyses the behavior of the AN software and to validate the generated products.
- Non-regression testing (PIL). The non-regression tests consist on re-running the functional tests in the Leon2 environment. Then, the AN products are analysed so as to verify the non-regression.
- Performance testing (SIL). Several Monte-Carlo scenarios are run in terrains with different difficulties and some metrics are computed, for example, the total length of the traverse, the number of navigation stops, the number of point turns commanded, etc.

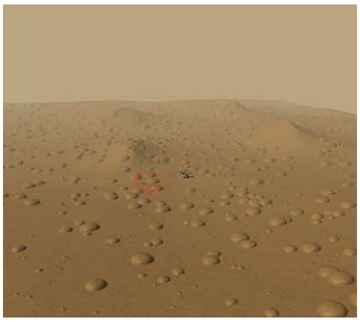

Figure 18 : Rover in the CNES simulation environment

- Computation time performances (PIL). Re-running some nominal scenarios, the computation time of each functionality is analyzed on the LEON2 board and statistics are computed.

Upon the validation campaign, verification activities are also performed as for example:

- Memory consumption and stack utilization.
- Code quality analysis (using quality tools as cppcheck, understand, etc..)
- Code coverage (using gcov/lcov tool).

4.2. CNES Validation : V1 Software

The main objective of this campaign is the functional and performance validation of the algorithm in its reference implementation. The compliance to the relevant requirements is demonstrated during this validation phase.

The validation relies on an adaptation of the CNES test benches (SIL and PIL) to take into account the final software interfaces defined by the Software Architecture presented in section 2.3. An emulator of the coTC/coTM interfacing and the final sequencing is put in place to allow the validation of the AN Software in its final version.

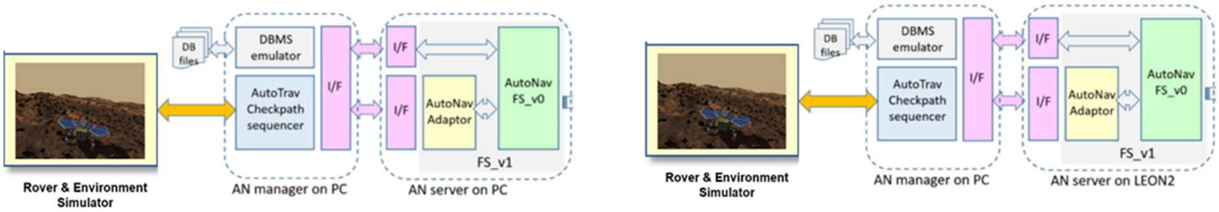

Figure 19 : Software-in-the-loop test bench (left) and Processor-in-the-loop test bench (right)

The same validation and verification activities defined for the Software v0 are carried out for the AN Software v1; several adaptations are needed to take into account the new architecture and to assure the requirements coverage:

- New Unitary Tests are added for the new functionalities and to complete code coverage analysis.
- Integration testing is completed to address all the possible coTC/coTM specified in the ASU Interfaces ICD and to validate its behavior.
- Functional testing is completed; specific scenarios are added to address the different functionalities and to assure the system requirements coverage.

This campaign is run at each formal evolution of the V1 Software in order to check the non-regression or the impacts of the evolution in the campaign.

4.3. NSVF Validation : V2 Software

The Numerical Software Validation Facility (**NSVF**) is the official facility where the Rover On-board Software design is validated, the NSVF is developed by ADS. A replica of this tool is delivered by ADS-UK to CNES, with the purpose of allowing the validation of the CNES AutoNav solution.

NSVF is a numerical simulation facility composed by the high-fidelity model of the on-board computer and Rover Vehicle Simulator with models of the Rover equipment, environment as well as the vehicle's full dynamics and kinematics. Therefore, the NSVF allows closed-loop simulation with a full image of the on-board software hosted by the OBC simulator. However, this simulator includes severe testing constraints:

- the time needed to upload a new binary image is to be measured in hours
- the time needed to initialize the bench is about ½ hour
- the time needed to retrieve all relevant data after a single test is to be measured in tens of minutes
- teleporting the rover to a specific position/heading configuration is not possible (the bench needs to be reset).

The final version of the NSVF was received by CNES late in the project. Therefore, the NSVF validation campaign put in place was designed to optimize the resources and the planning trying to overcome the constraints described above. The main objective of the NSVF campaign is to validate the correct behavior of the CNES AN Flight Software (v2) and verify a set of system requirements.

Several validation steps are carried out:

- <u>Interface testing:</u> Using a simplified binary image, the whole set of interfaces are exercised and is possible to check the different command sequences as well as the input/output parameters conveyed by each individual coTC/coTM.
- <u>Visual Localization non-regression tests</u>: Verifying that the fact of linking the AN Software in the coPM Image does not affect the performances of the VisLoc functionality.
- <u>Functional Testing</u>: These tests have as objective to verify the correct behavior of the AN in a set of specific scenarios that allow the verification of several system requirements. Both AutoTrav and CheckPath modes are validated and several situations of Abort/Resume of the AN Software are exercised. In order to optimize the test duration, only a few navigation cycles are performed for each one of the tests.
- <u>Perception Unit Tests:</u> The NSVF is more representative in terms of shading than the CNES simulator. Therefore, several tests are performed to validate the AN behavior in different shading conditions.
- <u>Robustness Tests:</u> This group of tests consists in exercising the different AutoTrav capabilities in presence of terrain difficulties and assessing the software reliability over longer distances up to 20 navigation cycles.

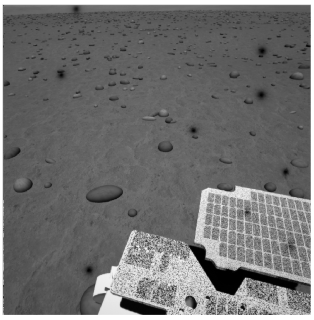

Figure 20 : Stereo acquisition in the NSVF Simulation environment (image credit ADS-UK)

4.4. GTM Validation : V2 Software

The Rover Module Ground Test Model (RM GTM) provides a complete functioning hardware build of the Rover Vehicle. Physically, the GTM is the model most similar to the Proto Flight Model (PFM), it is approximately representative of mass properties except that some redundancies are not be present.

The RM GTM test bench has been used to support integration and de-risk all related AIT functional verification activities. It is used also for Functional & Performance Tests in End to End Configuration including Mobility and Navigation & Path Planning functionalities.

These GTM Validation activities are conducted in the Mars Yard of the Rover Operations Control Center (ROCC) by ALTEC, as in real operations, with the support of CNES Team for expertise to analyze the test results and in case of anomalies.

Figure 21: GTM Rover model in ALTEC facilities (image credit ESA/TAS-I/ALTEC)

Two different validation campaigns are run on the GTM:

4.4.1. Confident Test Campaign

This campaign is defined by CNES. Several sequences are run to verify the behavior of the AN Software both in AutoTrav and CheckPath mode and using different sets of AN configurations.

The telemetry of the different sequences is recovered and analyzed by the CNES Team in order to verify the correct behavior of the AN.

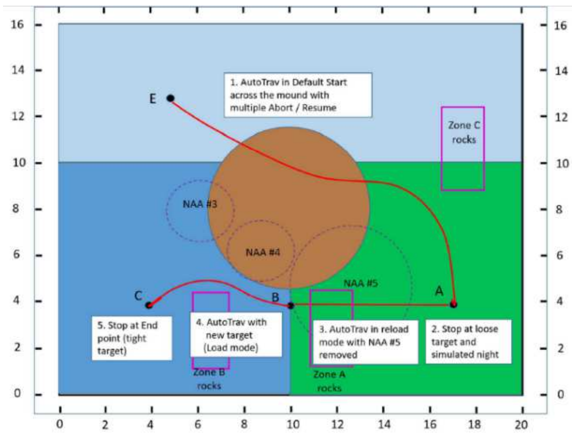

Figure 22 : Test sequence part of CNES Confident Test Campaign

4.4.2. IST/SVT Campaign

This campaign correspond to the formal and final validation step of the AN Software before the Qualification Review of the Software. It is defined by ESA/TAS-I/ALTEC and its objective is to formally verify the correct behavior the AN Software.

This campaign is common to the two AN Solution implemented on-board. Several sequences are run, both in AutoTrav and CheckPath and the telemetry is recovered to be analysed by the ROCC team and by the AN experts.

5. AutoNav Safety Requirements Verification

Demonstrating safety for the mission is critical; therefore, special care is taken in the Safety Requirements Verification. The approach applied is to assure that the uncertainties of the different methods are taken into account in the definition of the different navigability criteria as margins to be applied. This is addressed by different MonteCarlo simulation campaigns.

The adopted technique to deal with uncertainty is built upon the following pillars:

5.1. Perception functionality characterization

This process compares the resulting DEM generated from the stereo-images with an accurate ground truth DEM.

CNES disposed of an end-to-end perception performances test-bench that uses a real hardware and a laser scanner. However, this test-bench uses a camera model that is not fully representative to Exomars HW, so the results may not be directly extrapolated to Exomars. Therefore, the source of the images and the ground truth used for this analysis are generated with a high-fidelity simulator configured with the Exomars Stereobench characteristics. In order to improve the representability of the analysis, a rectification error model is added to the dispartity map to include remaining calibration errors that may affect the 3D reconstruction performances.

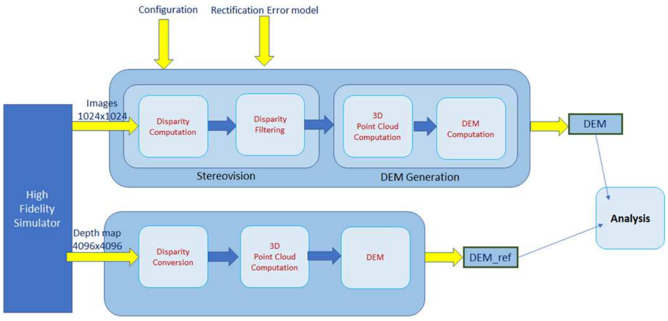

Figure 23 : Reconstruction Error Test Bench

Scenarios of different terrain difficulty are analyzed as well as specific rock reconstruction characterization. This allows to create an abacus of scenarios that will be used in the next step to determine a budget error attached to the DEM generation.

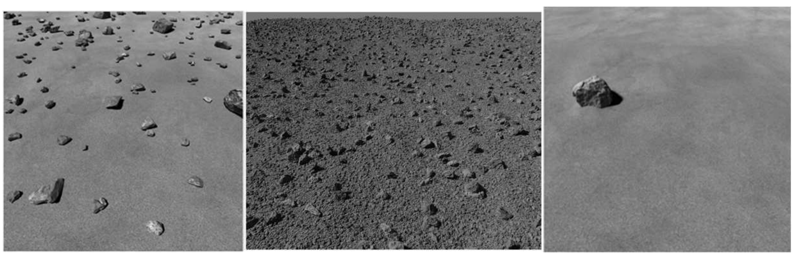

Figure 24 : High-Fidelity Image Simulator

5.2. Budget error attached to the 3D reconstruction accuracy

The analysis of different terrains and specific rocks allows the computation of statistics in terms of DEM reconstruction performances.

For example, the specific rock analysis focuses on the possible discontinuity errors (related to the error in the height and position of the rock):

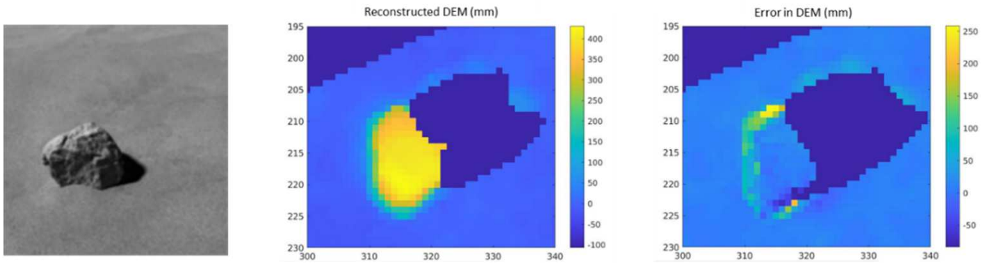

Figure 25 : Rock analysis: rock image (left), Reference DEM (centre), reconstruction errors (right)

The most important errors, due to the stereo method itself, appear in the edge of the obstacles, as observed in the example. These errors remain localised and do not have a major impact in the safety analysis as they tend to enlarge the obstacles.

One of the main conclusions of the analysis focused on distances lower than 3m is that for targeted rock analysis, the rock height error (3-sigma) do not exceed 2.25cm. The computed errors shall be added as margins to the initial navigability criteria.

5.3. Rover Configuration Determination Algorithm Error model

Construction of an error model for the Rover Configuration Determination Algorithm (RCDA) that estimates the rover pose and computes the safety margins.

The navigation map labelling relies on a simplified rover kinematic model (presented in section xx) to compute the rover pitch/roll angles, the boogie angles and the ground clearance. Due to the model approximations, the estimated rover configuration on each cell of the map is obviously affected by some inaccuracies.

- True positive : The method has correctly labelled the cell as traversable
- True negative : The method has correctly labelled the cell as non-traversable
- False positive : The method has incorrectly labelled the cell as traversable (it is not)
- False negative : The method has incorrectly labelled the cell as non-traversable (it is actually)

False negative cells can be tolerated to some extent but false positive shall be absolutely avoided. The setting of the traversability constraints shall be carefully tuned to achieve this objective and this relies on the introduction of appropriate margins.

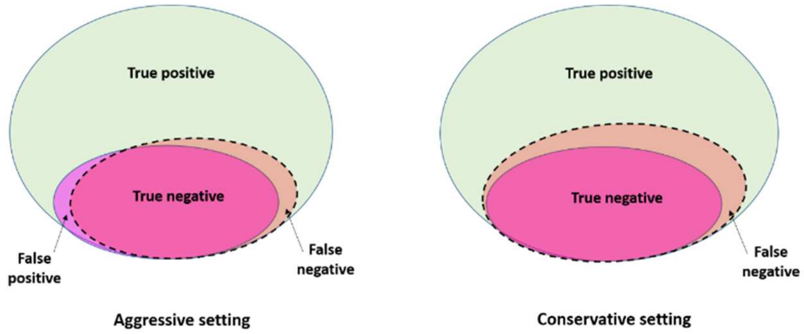

Figure 26 : Influence of the Traversability Constraints setting.

The determination of the margins consists in building an error model for the different rover characteristics that are estimated by the RCDA method:

- Rover tilt (pitch / roll)
- Ground clearance
- Boogie angles

In absence of any appropriate analytical method to tackle the problem, the approach finally adopted is based on a typical Monte Carlo campaign. This campaign consists in the generation of different relevant terrains and in the determination for each terrain of the labelling differences between the two methods.

The margin assessment consists in determining the worst case rover configuration with two different methods: the full rover kinematic model and then the approximated RCDA used in the AN Software.

The analysis consists therefore in determining the statistical characteristics of the false positive cases and this implies the existence of some initial set of thresholds (limits of each criteria)

For example, the Monte-Carlo analysis performed for a set of 33000 terrains gave the following margins to be applied to an initial setting for the different criteria:

RCDA parameter	Initial setting	Setting with margins (3σ)
Pitch	20 dg	18.23 dg
Roll	20 dg	17.5 dg
Ground clearance	0 cm	31.76 mm
Boogie angles	18 dg	14.76 dg

6. Results

This section presents an overview of the results obtained during the GTM test campaign that was the preliminary step to the Qualification Review Process for the CNES AN Software.

The Confidence Test Campaign was run from 9th to 13th May 2022 with CNES on site participation. CNES proposed three nominal scenarios that were agreed with ESA, TAS-I and ALTEC. Each scenario was composed of several segments allowing the verification of the main AN Software functionalities exercising CheckPath and AutoTrav mode across the terrain with different degrees of difficulty.

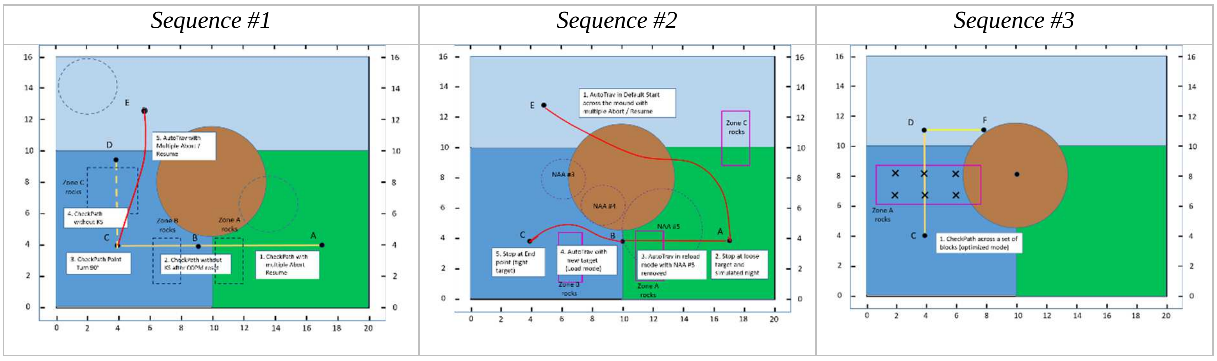

Figure 27 : Three nominal sequences proposed by CNES composed of 10 segments in total

In conclusion, more than 50m of navigation system driving were achieved. All segments were considered as passed except one who was considered as partially passed. One minor anomaly prevented the rover to reach precisely the goal for the last segment of the first sequence. The rover stopped at less than 1m from the goal and considered it as non-reachable due to a path-planning constraint that was not properly implemented.

As example, the results obtained for the second sequence are presented:

Figure 28: Trajectories commanded superimposed into the navigation maps for sequence #2

For this sequence the rover reached the consecutive goals avoiding all the obstacles, the behavior of the AN Software was fully nominal.

As the Confidence Test run smoothly, some complementary situations were exercised to test the limits of the AN capabilities. The behavior of the AN Software during these complementary tests was considered as nominal.

After the correct completion of the Confidence Test Campaign and the correction of the minor anomalies, the formal IST/SVT4 Campaign took place on May 19th 2022. The test sequence, that was common to the two AN Solutions on Exomars, was specified by ESA, TAS-I and ALTEC. The test sequence was a succession of segments in CheckPath and AutoTrav mode activating several functionalities of the AN Software.

Six segments were executed and all passed except the last one that triggered a Path Planning failure. This behavior was partially caused by a wrong configuration of the traverse that was not initialized in the expected configuration. A minor anomaly was raised to correct a very conservative check implemented on the initial conditions that triggered the error.

As a conclusion, the GTM tests allowed to validate the CNES AN in the most representative rover scenario. The behavior of the CNES AN was considered as nominal and the performances of the solution were compliant with the project requirements.

7. Conclusions

The development, validation and verification approach described before has been carried out for the CNES AN Software during the past years.

The realisation of the GTM Test Campaign was an important milestone for the CNES AN Software and the accomplishment of the effort dedicated by the CNES Team to achieve the validation of the Software and overcome the very specific technical and schedule challenges imposed to the whole software project.

The results of the GTM Test Campaigns, as presented in section 6, demonstrated a very good maturity of CNES AN Software that responded to the needs of the project, assuring the safety of the rover while keeping good computation time performances.

Currently, the Qualification Review process of the CNES AN Software is on-going. Taking into account the current status of the Exomars project with a launch date not before 2028, the on-coming activities are mainly related to the preparation and execution of the Maintenance Plan. During this maintenance period, it is possible that some functional updates, coming from the project needs, would affect the AN Software; in this case, CNES will envision an update of the CNES AN Software. Additionally, some code quality improvements are expected to be integrated through the maintenance process. Delta testing and non-regression testing activities covering CNES, NSVF and GTM test environments may be put in place to validate the Software evolutions before a formal Software Acceptance Review.

References

[1] Autonomous navigation: a development roadmap for Exomars. L. Rastel, M. Maurette, ASTRA 2006.

[2] Robotic planetary exploration: autonomous navigation in cluttered unknown environments A.Torres AIAA 2012

[3] Onboard Autonomous Path Planning for a Planetary Exploration Rover A.Rusu, S.Moreno AIAA 2013

[4] Perception and autonomous navigation for planetary missions M.Delpech ASTRA 2015