Testing for the MMX Rover Autonomous Navigation Experiment on Phobos

Lukas Meyer[1] Mallikarjuna Vayugundla[1] Patrick Kenny[2] Michal Smíšek[1] Jens Biele[3]

Alessandro Maturilli[4] Marcus G. Müller[1] Wolfgang Stürzl[1] Martin J. Schuster[1]

Tim Bodenmüller[1] Armin Wedler[1] Rudolph Triebel[1]

all authors are with the
German Aerospace Center (DLR)
firstname.lastname@dlr.de
[1]Institute of Robotics and Mechatronics
Muenchener Str. 20, 82234 Wessling, Germany
[2]Institute for Software Technology
Muenchener Str. 20, 82234 Wessling, Germany
[3]Space Operations and Astronaut Training
Microgravity User Support Center
Linder Höhe, 51147 Köln, Germany
[4]Institute of Planetary Research
Rutherfordstr. 2, 12489 Berlin, Germany

***Abstract*—The MMX rover will explore the surface of Phobos, Mars' bigger moon. It will use its stereo cameras for perceiving the environment, enabling the use of vision based autonomous navigation algorithms. The German Aerospace Center (DLR) is currently developing the corresponding autonomous navigation experiment that will allow the rover to efficiently explore the surface of Phobos, despite limited communication with Earth and long turn-around times for operations. This paper discusses our testing strategy regarding the autonomous navigation solution. We present our general testing strategy for the software considering a development approach with agile aspects. We detail, how we ensure successful integration with the rover system despite having limited access to the flight hardware. We furthermore discuss, what environmental conditions on Phobos pose a potential risk for the navigation algorithms and how we test for these accordingly. Our testing is mostly data set-based and we describe our approaches for recording navigation data that is representative both for the rover system and also for the Phobos environment. Finally, we make the corresponding data set publicly available and provide an overview on its content.**

TABLE OF CONTENTS

978-1-6654-9032-0/23/$31.00 ©2023 IEEE

1. INTRODUCTION AND SCENARIO OVERVIEW

The MMX Mission and the MMX Rover

Martian Moons eXploration (MMX) is a sample-return mission by the Japan Aerospace Exploration Agency (JAXA) to explore the Martian moons Phobos and Deimos. In the course of the mission, the MMX spacecraft will approach and orbit Phobos. It will touch down on Phobos for regolith sample acquisition and return the samples to Earth, with an intermediate observation stop at Deimos [1]. The mission timeline considers the launch in 2024, approach of Phobos in 2025, two landings for sample collection in 2027, and a return of the samples to Earth in 2029 [1]. The scientific goals of the overall mission are outlined in [2].

The MMX spacecraft will deliver a small rover system – the MMX rover – onto the surface of Phobos. The rover deployment is envisioned for 2027 during a rehearsal for the spacecraft landing [1]. It will be ejected from the spacecraft $40 - 100\,\mathrm{m}$ above the surface, free-fall towards Phobos, and bounce several times until coming to rest [1]. An uprighting sequence unfolds the rover's legs and ensures correct orientation of its body [3].

The MMX rover serves as a precursor to the landing sample acquisition process of the MMX spacecraft. The rover is jointly developed by the French Centre National d'Etudes Spatiales (CNES) and the German Aerospace Center (DLR) as main contributors. The MMX rover has three principal goals: a) the scouting of the landing site for the MMX

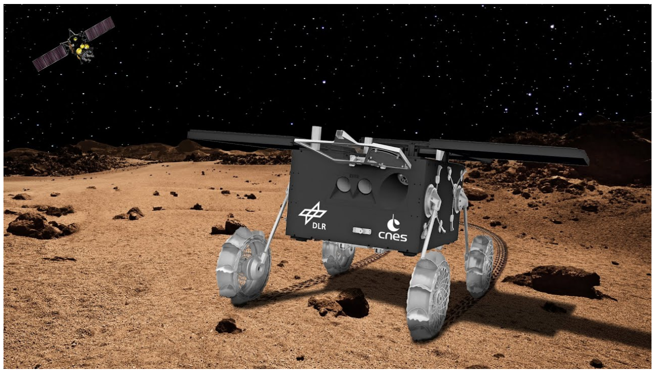

Figure 1: Rendering of the MMX rover on the surface of Phobos. The stereo cameras in the front-center of the body are its primary navigation sensors. Credit: CNES & DLR.

spacecraft for landing risk mitigation, b) demonstrating the technology of low-gravity wheeled locomotion, and c) obtaining measurements with the rover's scientific payloads [4].

The MMX rover has dimensions of about $44 \times 52 \times 35\,\mathrm{cm}$ and a mass of approximately $30\,\mathrm{kg}$ [4]. For locomotion, it has four legs that are equipped with wheels at the end. The rover is skid-steered. Due to the low Phobos gravity, a driving speed of only $0.1 - 4\ {}^{\mathrm{mm}}\!/\!{}_{\mathrm{s}}$ is planned [4]. The rover is equipped with solar panels and batteries to collect and store power for its operation.

As scientific payloads, the MMX rover possesses a Raman spectrometer (RAX), a radiometer (miniRAD), wheel cameras for regolith property observation, and navigation cameras in a stereo configuration [5]. The rover system is described in [3] and the rover-mission's science objectives are outlined in [5]. Its mission duration on the surface of Phobos is planned to be around 100 Earth-days [1].

The NavDLR Experiment

The stereo navigation cameras allow the usage of vision-based autonomous navigation algorithms to facilitate the rover's movements on the moon's surface. This is especially important due to limited communication windows with the rover and command turn-around times of up to a week. The constrained communication results from the MMX spacecraft being the relay between Earth and the rover and that Mars and Phobos regularly block the visibility between the communication nodes. The DLR Institute of Robotics and Mechatronics (DLR-RM) is developing the DLR Autonomous Navigation Experiment (NavDLR), a software component that allows the rover to:

- compute three-dimensional terrain information to detect obstacles and create a local map,
- use sequences of camera images to estimate the Rover's ego-motion on Phobos,
- introduce safety features such as obstacle avoidance or a safety stop on hazard detection.

The navigation pipeline of NavDLR is based on several consecutively used modules, as illustrated in Fig. 2. We outline all details of NavDLR that are relevant for this paper in Section 3; a comprehensive description of the software component is found in [6]. In general, all algorithms and concepts of NavDLR are based on the existing navigation pipeline that was developed at DLR-RM over the last two decades with planetary exploration missions in mind [7]: RM-NAV. It is an extensively tested navigation component that operates on several different systems such as UAVs or planetary exploration rover prototypes and was successfully used during two Moon analog campaigns [8], [9].

In the scope of the MMX rover mission, the autonomous navigation component NavDLR is classified as a technology demonstration experiment rather than a mission-critical component. This is motivated by the fact that the main mission objectives of the rover are initially centered on local tasks at the landing site and only later aim at the exploration of a wider area. Autonomous navigation will therefore only be considered after the principal mission objectives have been fulfilled and it serves itself as a technology demonstration. It is envisioned to expedite the exploration process, compared to the teleoperation-based main mode of driving, and to additionally support the rover by providing safety features to reduce the risk of getting stuck, hitting obstacles, etc.

Testing the NavDLR Experiment

This work describes our testing approach for the NavDLR software. During its development process, we aim at porting parts of the existing RM-NAV software and of its concepts onto the MMX rover system and create a space qualified, verified, and validated software product.

Space mission components, both software and hardware, are subjected to testing at all stages of their development. The development process can be categorized by the Technological Readiness Level (TRL) [10], with TRL 1 denoting initial concepts and the highest level, TRL 9, describing tested components that are fully operational in their space mission. The type of testing highly depends on the TRL level aimed for.

In terms of TRL, the existing RM-NAV stands at a TRL level of 5, as its performance was proven on rover prototype platforms in relevant planetary analog environments. In particular, we used our two Lightweight Rover Units (LRUs) in extensive field tests on the volcano Mt. Etna in Sicily during the ROBEX campaign [8] and during the ARCHES mission [9]. In both cases, together with several other evaluations and experiments, RM-NAV provided robust and accurate navigation performance in planetary analog environments.

Porting the existing software to the MMX rover means to lift the TRL of the software to 8, the highest achievable TRL pre-deployment on Phobos. This includes the following tasks:

- Redesign the software to meet the specific mission needs. In our case this mostly entails focusing on the core navigation functionalities to meet the computational limitations of the MMX rover's on-board computer, while maintaining the robust and accurate performance of the RM-NAV.
- Consider the mission-specific aspects of the MMX rover, such as short and infrequent communication windows and limited data rates. The turn-around time is expected to be approximately one week.
- The MMX rover will not have an inertial sensor. Instead, it features a Sun sensor: a BiSo-64 with a four quadrant photodiode. It infrequently – once or twice per Phobos day (Phod) – measures the attitude of the rover frame with respect to the Sun with a typical accuracy of $\sim 0.1°$. Our navigation approach needs to be adapted accordingly.
- Ensure the successful integration of the navigation with the rover's software architecture.
- Rewrite and restructure the software to meet the space-

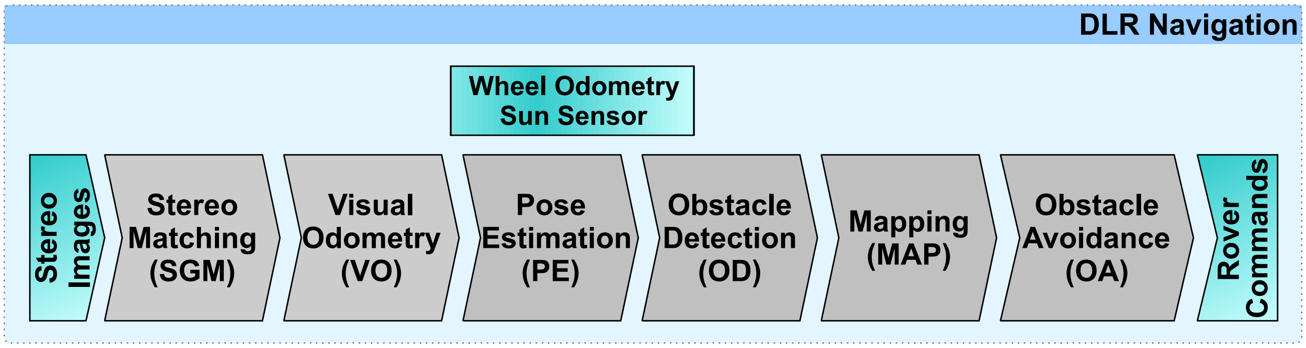

Figure 2: The functionality concept of NavDLR. The navigation pipeline consists of consecutively executed modules, whose data outputs are used by the subsequent modules.

relevant standards and design and quality criteria. These are, in our case, an adaptation of the ECSS Standards[2].
• Consider the specific Phobos environment, which can be seen as strongly similar to the Moon (from a vision-based autonomous navigation perspective) but still has significant differences in terms of e. g., illumination or shadow movement.

On the one hand, the character of NavDLR being a technology demonstration subsequently relaxes the requirements for navigation testing thoroughness compared to a mission-critical software component.

On the other hand, considering the limited communication windows once every few days and the approximate mission timeline of 100 days, post-surface-deployment update opportunities of the navigation software are very limited, thus a special emphasis has to be placed on pre-surface-deployment testing. Therefore, comprehensive testing is nevertheless to be done to successfully enable the autonomous navigation for Phobos.

We present our approach for testing the MMX rover navigation software, which:

• ensures continuous testing during the development process for rapid development iterations,
• verifies algorithmic correctness and the correct re-implementation of the existing algorithms,
• validates the algorithmic performance with respect to the expected environmental conditions on Phobos and additionally considers corner cases for stress testing,
• ensures correct integration in the target hardware and software architecture,
• and, finally, considers the acceptance testing (external verification & validation) on a flight system representation.

Data Set

Our testing is mostly data set-based. This allows us to have comparable results during different stages of the development process, and allows to repeat tests multiple times. We detail our data set generation approaches and the different hardware and software platforms used to generate the data. We make the testing data publicly available and describe the underlying, unified, data structure in this paper. The data is found under `https://rmc.dlr.de/mmx_nav_testing`.

[2]https://ecss.nl/hbs/active-handbooks/

2. Related Work

Several planetary exploration systems have been deployed on the surface of celestial bodies in the past two decades, mainly on Mars and the Earth's moon. The former is visited by the Chinese rover Zhurong and multiple NASA systems: the Mars Exploration Rovers (MERs) Spirit and Opportunity, the Mars Science Laboratory (MSL) Curiosity, and the Mars 2020 Mission with the rover Perseverance and the small-scale helicopter Ingenuity. On the latter, the Chinese rovers Yutu and Yutu-2 were landed. Furthermore, other rover systems have been developed for Mars missions, such as the European ExoMars Rover and the Sample Fetch Rover (SFR), which have yet to be launched.

Navigation on Planetary Exploration Rovers

What all of these systems have in common is that they rely on optical information for navigation, be it teleoperation or increasingly autonomous vision-based navigation.

The MER rovers Spirit and Opportunity used localization algorithms based on continuous wheel odometry (WO) and inertial measurements. They increasingly relied on visual odometry (VO), whereas the VO pose updates are computed during step-wise intervals when the rover stands still. Furthermore, stereo images were used to compute terrain models for traversability and hazard analysis. The MER vision components are described in [11], [12], [13]. The testing for ground-based validation was done at the JPL *Marsyard* using the *Rocky 8* testing prototype rover [14] and the *MER Surface System Testbed Lite* rover, the MER engineering model [11]. During the mission itself, the autonomous navigation capabilities were initially used only occasionally – mainly due to the high computational load – but were later on used with higher frequency [11].

The MSL Curiosity uses a similar and updated autonomous navigation approach as its MER predecessors, featuring VO [15], terrain mapping and hazard detection [16], to name a few. Recently, the VO was updated for more efficient computation and to be able to compute pose estimates while the rover actively drives [17].

The initial navigation components for the MSL were tested in a similar manner as for the MER: Using data from the Rocky 8 rover, the DARPA LAGR vehicle, and finally with original Mars data obtained from the MERs [15]. Maimone *et al.* [17] provide a detailed insight into their testing approach for a recent VO update, using the MSL engineering model

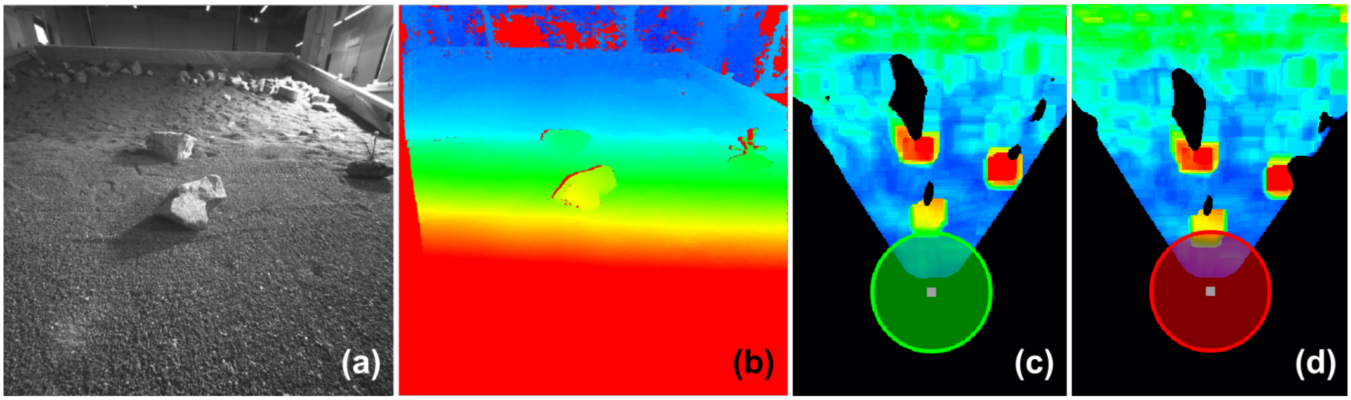

Figure 3: Illustration of the NavDLR navigation solution pipeline in a simplified testing scenario. The monochrome stereo images – the left image is shown as example in (a) – get processed to obtain a depth map (b), which is subsequently evaluated for obstacles whether they are outside (c) or within (d) a hazard boundary. We refer to this simplified test scenario as *Safety Stop Adventure*.

in an indoor setting, unit testing, and a software architecture simulation environment for validation and verification.

NASA's most recent Mars rover, Perseverance, also builds on the navigation heritage of the previous Mars missions. Thanks to fast on-board FPGA processing and modified cameras that allow for shorter exposure times, it can perform online navigation while driving [18]. Rieber *et al.* [19] detail the testing venues that are generally available for testing Perseverance's mobility system, e. g., simulation environments to test the flight software, the *Scarecrow* rover with commercial off-the-shelf (COTS) cameras for navigation testing, and finally, the engineering model, called *Vehicle System Testbed*. The engineering model is described in detail by Matthes *et al.* [20] and the flight software emulation environment for testing is presented in [21].

On the European side, plenty of flavors of autonomous rover navigation algorithms exists, with a comprehensive overview given by [22]. For the ExoMars rover, the autonomous navigation architecture is presented by [23] and the VO algorithm by [24]. Testing the ExoMars navigation entails the usage of a simulated software environment, using an ExoMars prototype in the Airbus *Mars Yard*, running the algorithms on the target CPU, and using images from previous Mars rover missions [23], [25]. For the SFR navigation architecture, similar testing approaches are done using simulation environments and different low- and high-level testbeds [26]. Furthermore, previous stages of the European navigation algorithm development were aided by field tests in analog environments, such as described in [27], [28], [26].

The vision algorithms to aid the Chinese Yutu-2 rover teleoperation on the Moon are described in [29].

Common Testing Aspects

The rover systems discussed above have several aspects regarding navigation testing in common, or at least a subset of these. We summarize them here:

- Simulation environments that emulate the overall rover software are used to test interfaces and interaction of the navigation with the other software components.
- Simulation tools allow photorealistic terrain rendering.
- Testing is done mostly on COTS-based rover prototypes that are not necessarily closely representative of the final system design, but can feature as proxy test platforms for specific testing aspects.
- Field tests with such proxy rovers are performed at planetary analogue sites. Alternatively, recorded data sets from previous field tests are used.
- The software is tested with input data (i. e., images) from previous space missions.
- Test setups are used that integrate a limited subset of the flight-hardware and -software components. They are usually referred to as testbeds or breadboards.
- Engineering models: Fully integrated testbed or breadboard systems that resemble the flight system very closely are used for the final acceptance testing.

Most of these aspects apply to the NavDLR testing approach as well. However, there are a few notable exceptions. All Mars missions were able to iteratively build upon previous missions or at least test with image data directly obtained from Mars. MMX will be the first mission to the surface of Phobos, thus NavDLR cannot build upon previous navigation experiences regarding the Phobos environment. This means that environment-related testing becomes a high priority for us.

Furthermore, all Mars missions to date have mission lifetimes that exceed an Earth-year, which allows for updates on the software and testing the navigation software on the flight system during the on-surface operations. The short mission timeline for the MMX rover and the strong limitations in communication limit the possibility of software updates during on-surface operations (except for software parameter adaptation) and thus NavDLR needs to work out of the box.

The MMX rover will feature a much lower driving speed than comparable Mars or Moon rovers. Combined with the limited mission time, this results in much less distance traveled, requiring the navigation to focus on *local* instead of *global* accuracy.

Available Data Sets for Planetary Navigation

An additional aspect of our work is that we make our recorded testing data publicly available. Several related works exist that publish data sets for planetary rover navigation. One example is the Katwijk Beach Planetary Rover Dataset [30], where a planetary rover prototype performs several long-range traverses on a beach using stereo cameras and lidar.

Another example is the Devon Island data set, where recordings of a pushcart platform equipped with stereo cameras, a Sun sensor, and inclinometers in the Arctic north of Canada are presented [31].

Our in-house works include several data sets as well: One is the MADMAX Mars analogue data set [32], an extensive collection of trajectories with a combined length of $9.2\,\mathrm{km}$. It was recorded using the hand-held Sensor Unit for Planetary Explorations Rovers (SUPER), covering stereo camera data and inertial measurements, among others. Long-range navigation data were recorded with our LRU rovers in a Moon analog setup on Mt. Etna during the ROBEX campaign and are featured in [33].

Finally, the Planetary Data System [34] from NASA makes it possible to access data directly obtained from current and past NASA Mars missions.

3. The DLR autonomous navigation experiment (NavDLR)

The different modules that make up the navigation solution pipeline are shown in Fig. 2. They are described briefly here. For a more detailed view, see our other paper [6].

A visualization of the data products of the NavDLR pipeline is given in Fig. 3 for a simplified navigation scenario. The primary sensors for the navigation are the stereo-cameras that are fixed inside the MMX rover body. The modules of the navigation pipeline run in a sequence and the pipeline starts with the event of acquisition of new stereo images, see Fig. 3 (a).

The first module – as shown in Fig. 2 – is the stereo matching, which first rectifies the input stereo images and reduces the image size by binning. It then uses the semi-global matching (SGM) [35] algorithm to compute a disparity image (Fig. 3 (b)), taking the rectified stereo images as input. From this, a depth image is computed. This depth image and the left camera image are then used by the VO module to compute the relative 6D motion of the camera with respect to stored, previous images (the key frames). The resulting motion is mapped to the rover center and thereby provides the 6D relative pose of the MMX rover with respect to the environment. The pose estimation (PE) module estimates the 6D pose of the MMX rover using the VO estimates as well as other inputs like the intermittently available absolute attitude estimation from the Sun sensor. The modules up until here are relevant for the localization of the MMX rover.

The depth image is then also used by the obstacle detection (OD) module to detect different obstacles in the Phobos environment that need to be avoided by the MMX rover for safe navigation. It generates a 2.5D obstacle-cost map as output that encodes the traversability costs considering the locomotion capabilities of the MMX rover (see Fig. 3 (c) and (d)). In addition, it also generates the 2.5D elevation map as a digital elevation model (DEM) of the environment that is within the NavCAM's view. The mapping (MAP) module takes the 6D pose estimates of the MMX rover provided by the PE module and the traversability 2.5D cost map from OD as inputs and outputs a 2.5D local rolling cost-map that moves with the MMX rover. It is a local map to limit the map-size and therefore the memory consumption. Besides its use for navigation, the map can be sent to Earth as a science product to provide spatial context for measurements by the science instruments.

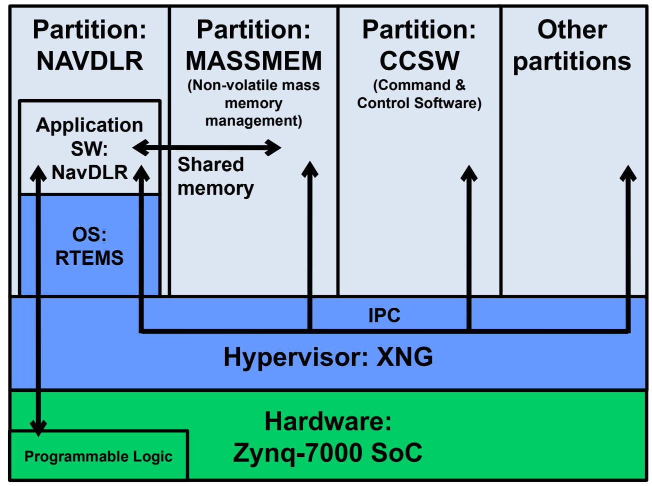

Figure 4: Architecture of the Rover's software partition concept on the OBC. The Xtratum Next Generation (XNG) hypervisor partitions time and OBC resources for different components. The NavDLR software sits in its own partition called NAVDLR and has RTEMS as its operating system.

Finally, the obstacle avoidance (OA) module uses this local rolling cost-map and the MMX rover's pose estimate to generate the necessary motion commands. The motion commands can be provided by OA at different complexity levels. It can serve as a hazard-detection module that triggers safety stops, as shown in Fig. 3 (c) and (d). Alternatively, it enables the rover to follow predefined paths or even to actively avoid obstacles on a path towards a goal position.

The development of NavDLR is closely linked to the architecture and the performance capabilities of the rover's on-board computer (OBC). The OBC consists of a Xilinx Zynq-7000 system-on-chip (SoC) integrated on a board with $512\,\mathrm{MB}$ of DDR3 RAM. The software architecture from NavDLR's perspective is shown in Fig. 4. The OBC runs an Xtratum Next Generation (XNG) hypervisor[3] which separates the different software subsystems into partitions. Communication between the partitions and access to system resources is restricted by the hypervisor to prevent unwanted interference. NavDLR will be in a partition of its own, called the NAVDLR partition. Other partitions contain software to carry out other experiments as well as software relevant to the MMX rover mission like the Command and Control Software (CCSW), memory management, or locomotion control. We use the RTEMS[4] real-time operating system on the NAVDLR partition to run our navigation pipeline.

4. Testing for NavDLR

All modules of NavDLR that were described in the previous section need to be subjected to extensive testing. Recalling the limited mission time on Phobos, it becomes clear that no further development on the navigation pipeline can be done once the rover has landed, but instead the testing needs to be extensive and be completed beforehand.

[3] https://fentiss.com/products/hypervisor

[4] Real-Time Executive for Multiprocessor Systems https://www.rtems.org

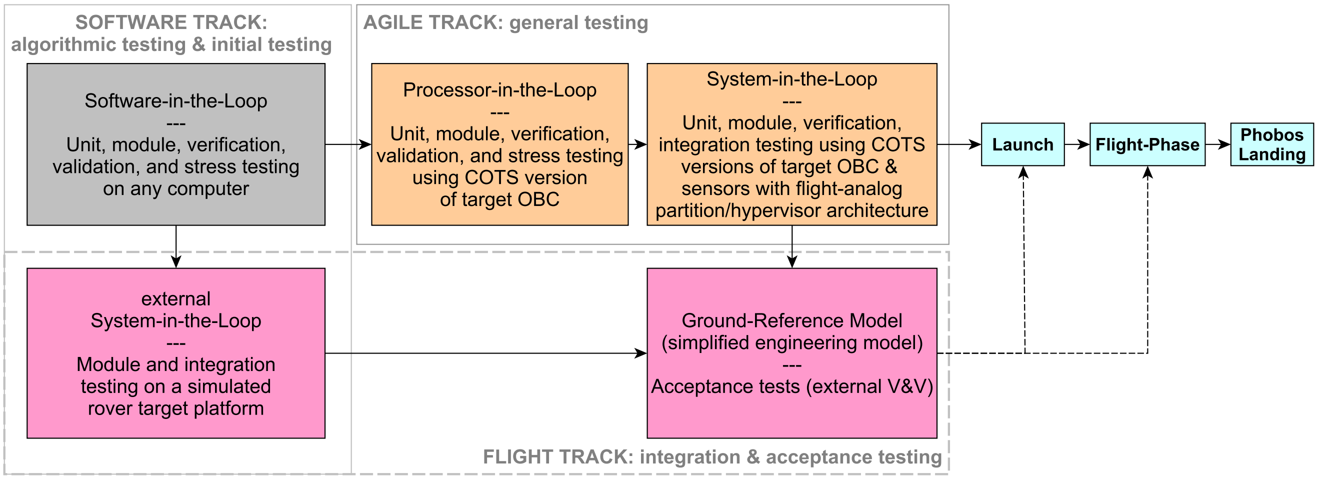

Figure 5: Our different testbeds for NavDLR with respect to MMX rover representativeness. The *software track* focuses on the testing of the algorithms, the *agile track* focuses on a high fidelity mock-up of the on-board computer (OBC) and flight avionics in a project-internal testing setup, the *flight track* has the highest hardware and flight software fidelity but these resources are shared with the full MMX rover project, thus access is highly limited.

Our software development is based on a development approach adopting some agile principles, e. g., iterative and incremental development cycles and continuous integration and testing. However, our project is embedded in a large-scale overall system project that is not agile. Our testing needs to reflect that approach. We therefore outline some aspects of our testing philosophy.

Story-based testing: To follow agile methodologies, we define major development cycles. Each development cycle aims at integrating new features into the code and making existing features more robust. At the end of the development cycle, the features are tested in a new application scenario, i. e., a new task that the rover software needs to solve. We call these tasks *adventures* and make each adventure more challenging for the modules and the pipeline. An example is our first story, the Safety Stop Adventure (SSA), where the test data is a straight trajectory over terrain with obstacles, which is based on real input data. The software needs to signal a hazard detection once the cameras get too close to an object. Even though some of the software modules are only set to a pass-through mode, the SSA did require all internal interfaces to be defined and correct data flow to be enabled, thus representing a major milestone in our development.

Upcoming adventures are, among others, Follow Path Adventure, Evade Stone Adventure, Mapping Adventure, Survive Phobos Adventure (Stress testing).

Continuous, automated, and data set-based testing: The idea is to automate as many tests as possible, and execute these tests continuously for any software update. This obviously includes unit tests for software elements but also tests for modules, up to executing navigation trajectories and running integration tests. To enable the recurring execution of these tests, the sensor data needs to be pre-recorded and readily available. Thus, we rely on a collection of data sets that capture as many aspects of the testing as possible. We call this automated and continuous testing *Software-in-the-Loop*, which constitutes a major part of our testing campaign as *software track* – see Fig. 5. Indeed, any change in our code triggers the automated unit tests and any major release is supposed to trigger all available automated tests on module or even pipeline level.

Limited access to flight hardware – workarounds needed: Finally, a major aspect of testing NavDLR is the fact that we have very limited access to the actual MMX rover flight hardware, its engineering model, and simulated system mock-ups. We therefore consider two testing tracks in parallel: The first is the *flight test track* that is done with the flight hardware mockups and simulations for infrequent, final tests to guarantee compatibility with the actual MMX rover. For more frequent integration testing, we create our own flight-system analog testbed. This *agile test track* is maintained by the NavDLR development team and we use it to emulate all interfaces and the OBC of the MMX rover. This second track allows for fast, independent iterations in testing during the development, which is in accordance with the agile elements of our development philosophy. All testing tracks are visualized in Fig. 5.

Testing Categories

To ensure the desired performance of NavDLR, testing needs to cover multiple different fields. Generally, three major *categories* regarding testing need to be considered.

FUNC – Functionality-oriented and algorithmic testing: Tests that evaluate the algorithmic correctness of NavDLR, using either qualitative evaluation methods or quantitative metrics. It aims at ensuring that a) NavDLR is an algorithmicly correct re-implementation of RM-NAV modules and b) the algorithms provide correct results in terms of absolute measures. These tests can either consider individual modules or the whole navigation pipeline in different scenarios.

SYS – System integration-oriented testing: These are tests to ensure the operability of NavDLR on the flight system. They range from tests that consider NavDLR on actual engineering models of the flight system to tests that run the code on representative CPU architectures. Our testbeds regarding the system integration are conceptualized in Fig. 5 and detailed in Section 5.

ENV – Environment-oriented testing: NavDLR is tested with respect to the expected environmental conditions on Phobos. All aspects of the Phobos surface conditions that we deem relevant for NavDLR are discussed in Section 6. It has to be noted that a high degree of uncertainty exists regarding the regolith composition on Phobos, which strongly influences the scene observed by the navigation cameras and thus, reliable predictions for the navigation performance on Phobos are challenging to produce. The ENV tests try to ensure the functionality of NavDLR for the expected Phobos environment based on a nominal scenario [36]. It furthermore aims at assessing the limitations and robustness of NavDLR for corner cases and environmental conditions that lie at the edge of the envelope of expected environments. The generation of sensor data that encapsulates the Phobos environment is described in Section 7 and our testing with respect to the environment in Section 8

Testing Approaches

We define different testing *approaches*, which cover the extensiveness and the state of development were the tests are used. This differs from the already presented *categories*. The categories highlight the field of application for the test, whereas the approaches highlight the granularity and the expected results.

We define the following test approaches, and which categories they cover:

- **Unit Testing & Module Testing** [FUNC, SYS]: The project uses a test-driven approach to software development. Hence, all parts of the implementation have to be covered by unit and module tests. These are tests that are frequently and automatically executed on the function/module level for all code of NavDLR and usually test small granular aspects of the code.
- **Verification Testing** [FUNC, SYS]: Tests or analysis that confirms that each module and the overall software is made according to specifications, provides correct output for a given input and fulfills testing metrics for ideal sensor inputs. The metrics need to be fulfilled either with respect to external reference measurements (*ground truth*) or with respect to the original navigation components of RM-NAV.
- **Validation Testing** [ENV]: Tests that confirm that a module or the overall software can perform under representative environmental conditions for representative sensor measurements. Clear pass and fail criteria are stated for these testing conditions. Validation tests are done both for the overall pipeline and for each individual module of NavDLR.
- **Integration Testing** [SYS]: Integration testing aims to ensure that NavDLR can run on the MMX rover's OBC using the available system architecture and resources and can correctly interface with other partitions on the OBC.
- **Stress Testing** [ENV]: Stress tests are for the preparation of the NavDLR experiment on Phobos. It is a preliminary assessment of NavDLR's performance under conditions that exceed the scope of the validation, but might possibly be encountered during the deployment on Phobos. These experiment preparation tests can be considered as stress tests for NavDLR. There are no pass/fail criteria, but rather an evaluation of the extent to which NavDLR can cope with potentially extreme environmental conditions.
- **Acceptance Testing (External V&V)** [SYS]: This is an overall NavDLR pipeline test on a system that represents the final flight system as closely as possible to ensure flight readiness of our software. Acceptance testing should test the functionality and correct final integration of the navigation pipeline and the capabilities to retrieve data from the NavDLR partition and to send commands to it. For this test, the NavDLR software is uploaded to the test model target and started, together with all interfacing components subject to testing. Acceptance tests will be limited to a few as-monolithic-as-possible configuration scenarios to account for the limited test time that is available on the acceptance testing platform. The union of performed tests will cover all acceptance testing aspects.

Testing the Requirements

A key element of NavDLR development is the testing of the requirements for a formal verification and validation. In accordance to our testing philosophy, the formal requirement tests are to be automated to the extend possible, and an automated reporting system is planned to be introduced.

For all other requirements, formal qualification campaigns are envisioned that evaluate NavDLR with respect to its requirements as major project milestones.

5. System Integration Testing

The system integration testing – covering the SYS testing category – aims to verify that NavDLR is able to function correctly when running within the flight software and hardware of the MMX rover. We describe how to recreate and use mock-up versions of the flight system to aid our testing, in addition to the highly limited acceptance testing done e. g., on an MMX rover engineering model.

The hypervisor-based software architecture of the rover provides strong separation between each subsystem's software, with clearly defined interfaces. This allows much more testing to be performed at the unit testing level. However, aspects of the software which inherently require system resources or interaction with other subsystems must ultimately have their interfaces with the external components tested. The external interfaces of NavDLR can be divided into several categories, each with different prerequisites for integration testing. NavDLR makes some calls directly to hypervisor functions, in order to query the system time or control its execution. Representative testing of these calls requires NavDLR to be executed in the hypervisor. As the only board-support package available for the hypervisor is for an ARM processor, this in turn requires an ARM target, such as a Zynq-7000.

Therefore, all agile and flight track testing considers an ARM target. In contrast, the Software-in-the-Loop testing of the agile track is executed on a x86 architecture.

Interactions with other partitions via the hypervisor's inter-partition communication (IPC) channels, such as for sending life signals or receiving telecommands, require not only the hypervisor itself, but also other partitions, real or mocked, as communication partners. Communication of larger data blocks, like images or maps, is performed using shared memory rather than the hypervisor's IPC channels. This requires communication partners, real or mocked, but not necessarily the hypervisor. NavDLR calculates depth images from the stereo navigation camera images using SGM. In order to accelerate this calculation, it is performed on the programmable logic (FPGA) of the Zynq-7000. Testing this component, and the software's interaction with it, therefore requires a Zynq-7000 SoC. Finally, ensuring that NavDLR is able to operate within its CPU-time and memory constraints is also best achieved when compiled for and executed on the

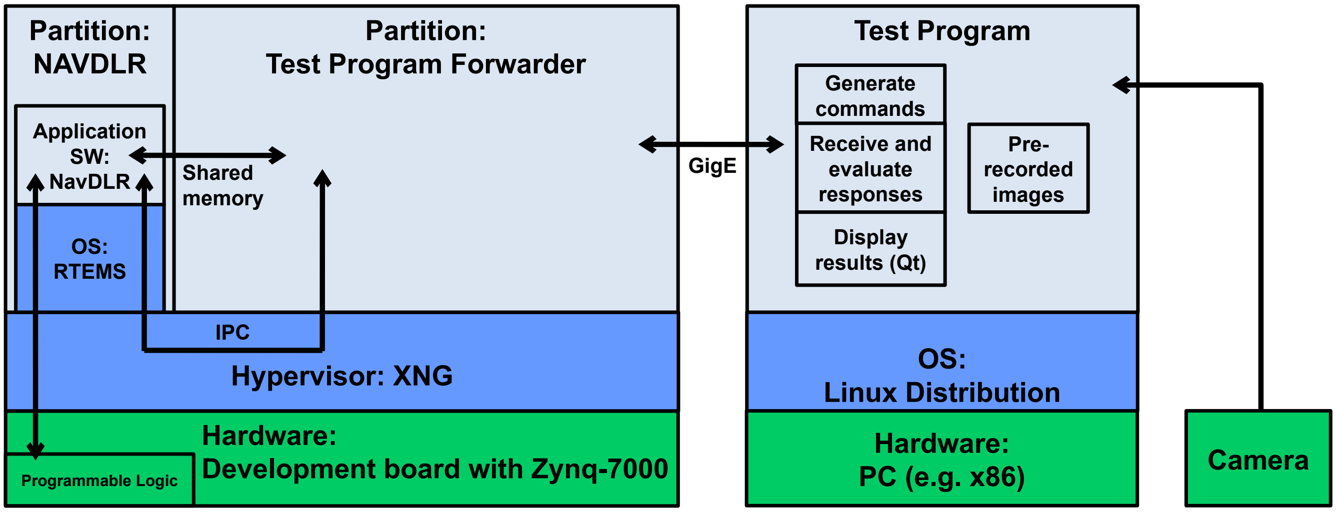

Figure 6: NavDLR's agile-track System-in-the-Loop test system: A development board with a representative Zynq SoC runs the Xtratum hypervisor, the NAVDLR partition and the test program forwarder in a separate partition. The forwarder connects via Gigabit Ethernet to a test program running on a general purpose computer to command and test NAVDLR. The test program can provide NAVDLR either with pre-recorded sensor data or live data, e. g., images from a camera.

target Zynq platform.

In order to test these aspects, two integration test systems (Figs. 6 and 7) are under development. One is developed at the NavDLR internal project level on the *agile track*, one at the system level for the overall MMX rover on the *flight track* – recall Fig. 5.

Agile Track System-in-the-Loop

The first system is the NavDLR internal platform and is shown in Fig. 6. The center of this system is a SoC from the Zynq-7000 family, as will be used in the flight OBC. This chip is on a commercial development board, providing a power supply and access to peripherals. The SoC is connected via Gigabit Ethernet to a PC running a general-purpose Linux distribution. This Linux-PC can be connected to cameras to provide live image data to NavDLR, or it can provide pre-recorded images from existing data sets, as discussed later in Section 7. The PC runs a test program, which communicates with the SoC to facilitate testing of NavDLR. It can receive data from NavDLR's outputs, such as telemetry, life status signals, and non-volatile memory requests. It also sends data to NavDLR, both as a direct response to requests (such as acknowledgments) and to initiate certain behavior for specific tests (such as sending telecommands). The PC's test program also evaluates the responses from NavDLR to see whether they match expectations. In this way, the Linux-PC's test program functions as a high-level mock. The PC's test program is also able to access NavDLR's internal data, which is not externally accessible under flight conditions, by accessing NavDLR's memory. This allows more convenient and detailed insight into NavDLR's behavior for development and debugging. By having all interfaces as mocks under the NavDLR development team's control, we are able to easily generate edge cases, such as invalid messages sent to NavDLR, and verify that NavDLR is able to handle these appropriately. The Linux-PC's test program makes use of Qt to provide a graphical user interface to help control manual tests. It additionally allows to visualize the results, e. g., data products like the ones shown in Fig. 3.

In order to allow the Linux-PC's test program access to NavDLR via Ethernet without modifying NavDLR itself, a second component to the test program is also developed. This component is executed on the SoC as a separate partition running on the Zynq-7000's other CPU core. As this program is necessarily running on embedded hardware with all the limitations and challenges this brings, as much logic as possible is off-loaded to the PC's test program. The component on the Zynq acts only as a forwarder of messages, reading data from the PC via Ethernet, converting to the necessary format and forwarding to NavDLR via the same inter-partition communication mechanisms which NavDLR uses in flight, and similarly forwarding data from NavDLR back to the PC.

This system will be used in two situations. In the first, it will be connected to our continuous integration infrastructure to allow the software to be built for the target architecture and executed in a representative hardware and software environment. This allows rapid and convenient testing, such that they can be executed with every software update, similar to unit tests, but at a higher integration level. In the second situation, this system will be connected to live cameras which are mounted on the LRU as a movable platform (like in Fig. 9) in the Planetary Exploration Lab (see Section 7) for dedicated testing campaigns. This allows a realistic simulation of the full NavDLR pipeline in a closed navigation loop, consisting of stereo image capture, data transfer to NavDLR, processing, generation of movement commands and appropriate movement of the cameras, as if they were mounted on the MMX rover.

As side note, the *Agile Track Processor-in-the-Loop* from Fig. 5 is the same SoC, but running a Linux distribution as operating system. This allows testing for the architecture and resource consumption in a more easily accessible environment, running the same automated tests as for the Software-in-the-Loop.

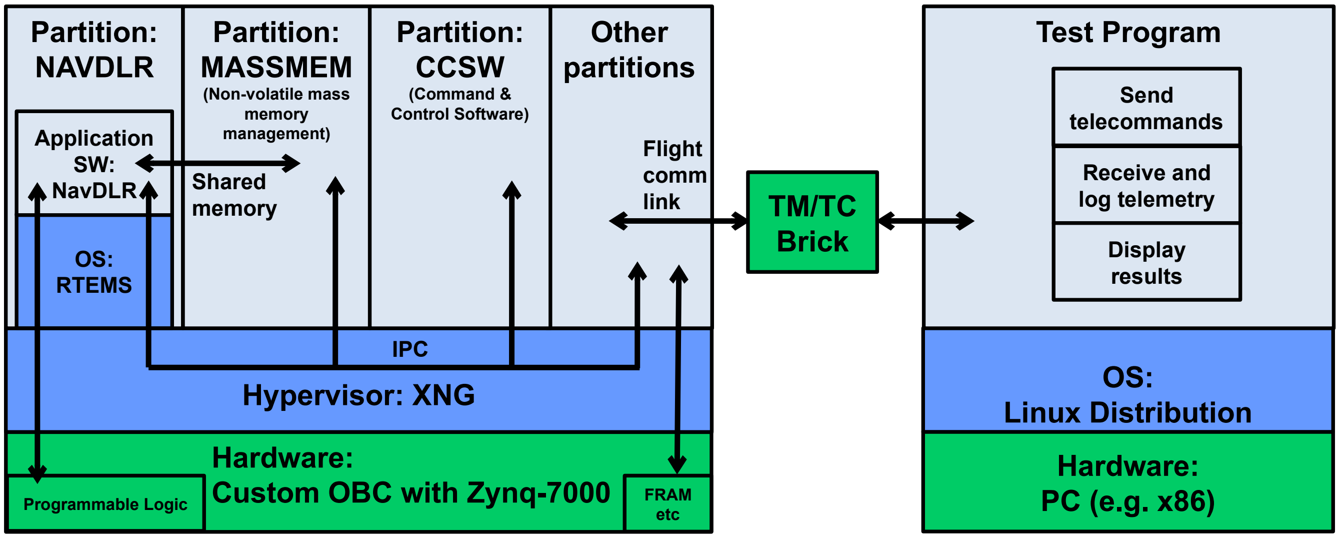

Figure 7: The MMX rover's flight-track system-in-the-loop test system: A custom on-board computer with a representative Zynq SoC runs the Xtratum hypervisor, the NAVDLR partition and other real partitions. A TM/TC brick connects the test PC to the on-board computer using a flight-representative communication link. The test program on the PC can send and receive commands to NAVDLR via the brick and the command and control partition.

Flight Track System-in-the-Loop

The second system is developed in the context of the overall rover project and is shown in Fig. 7. It is the system-in-the-loop testing platform as part of the flight track. This system also centers on a Zynq-7000, but as part of a custom-designed OBC much closer to that which will be used in the flight model. This includes additional hardware components, such as Ferroelectric Random Access Memory (FRAM) for storage of state data of the rover, and the flight equivalent communication link. The test system includes a *TM/TC brick* which emulates the MMX spacecraft end of the communication link, as well as a suite of software running on a test PC, which allows the generation of telecommands and the logging of telemetry.

In this system, the SoC contains a full suite of real flight software, in so far as it has been developed. This means no mocking framework is required; NavDLR interfaces with other real components. This provides a more representative hardware and software environment than our NavDLR internal integration test setup. Testing on this system allows detection of any issues arising from timing differences between the mock and real versions, or from any potential differences between the mock's behavior, based on the NavDLR development team's understanding of the interface specifications, and the real behavior. However, some software components will be ready only late in the project, as they are developed concurrently with NavDLR. Additional complications are that we only have limited access to this system and generating edge cases with it is more difficult than using mocks.

Acceptance Testing on the Flight Track (External V & V)

As is oftentimes the problem in space missions, the access of high-level software, such as NavDLR, to flight hardware, before launch, will be limited due to timeline pressure. To reflect the need for acceptance testing of NavDLR, its interfaces with other subcomponents - central rover software components, Sun sensor, locomotion, navigation cameras - will be tested using a setup involving engineering-model hardware components and a software simulator mimicking the functioning of the rover system. In addition to that, the path commanding capabilities of the OA module will be verified in a software simulator only. Potential testing on an engineering model during the flight phase after the mission launch is an ongoing discussion.

6. PHOBOS ENVIRONMENT RISKS

Testing of NavDLR needs to take the surface conditions of Phobos into account. A summary of relevant Phobos environment aspects was compiled in the Phobos environment requirement document (ERD) [36]. As a general rule of thumb, Phobos is assumed to be similar to the Earth's moon in aspects regarding vision-based autonomous navigation, but darker – with several exceptions. We briefly state all properties of Phobos' surface that we deem to be relevant for our autonomous navigation algorithm. We furthermore assess the risks of the environmental influences for NavDLR's performance.

Illumination and Optical Properties

Like the Moon, Phobos lacks an atmosphere. Thus, the incoming light is the unfiltered solar spectrum with a predictable – strong – intensity. Diffuse light (usually mostly caused by an atmosphere) is almost absent due to the low albedo of Phobos' surface, resulting in sharp shadows with strong contrasts between the illuminated and shadowed areas [36].

The landing site of the rover is selected to be the side of Phobos that faces away from Mars (Phobos is tidally locked) [1], thus no additional illumination reflected from Mars needs to be considered.

The distance from Phobos to the Sun oscillates between

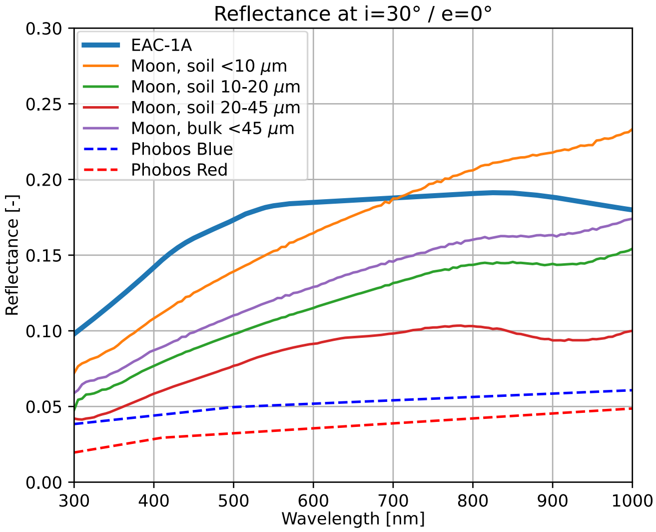

Figure 8: Reflectance properties of Phobos (red and blue areas) and the Moon compared to the EAC-1A analog soil. The Moon data are measurements of sample number 14163 from Apollo 14. The Phobos values are predictions from the ERD.

approximately 1.4 and 1.65 AU [36]. The intensity of the sunlight is therefore reduced by a factor of 0.5 to 0.35 compared to the sunlight received on top of Earth's atmosphere.

The soil has a rather featureless reflectance spectrum except a variable red slope (more reflective in the red than in the blue spectral regions), similar but weaker than seen for lunar soil: Phobos' geometric albedo at visible wavelengths is very low, around 7 % compared with 12 % for the Moon [36]. In Fig. 8, we plot the expected standard reflectance ($i = 30°$ and $e = 0°$, where i denotes the incidence angle and e the emission angle) of Phobos soil using estimations from orbital observation data [36]. Note that for Phobos, two different principal categories of soil are observed in orbital data. These are general regions on Phobos, where the regolith is overall very dark, but slightly differs in the spectral reflectance, that is a *red* and a *blue* region.

The figure additionally shows the results of an analysis of an Apollo 14 sample of different grain sizes for reference [37]. We see that Phobos reflectance is approximately three times lower than that of the Moon, but in general one can say that both celestial bodies have similar and dark soils with continuous reflectance spectra.

Risk – Our risk assessment regarding the illumination concludes that the strong contrasts need to be considered in the testing approach, especially for the capability of the SGM to compute disparity values in areas that are over- or underexposed. Differences in illumination of the scene compared to e. g., the Moon can be mitigated by different camera exposure settings, thus similar test procedures can be applied. Motion blur because of longer exposure times is considered a low risk due to low driving speed of the rover.

Moving Shadows

Phobos orbits Mars in a tidally locked setup and therefore has a rotation with respect to the Sun that is equal to its orbital period $T_{\mathrm{orbit}} = 7.66\,\mathrm{h}$ [36]. The Sun therefore moves relative to objects on Phobos (e. g., stones) with the same angular velocity during a Phod, casting moving shadows.

The relative angular velocity casting these shadows can be calculated as

$$\omega_{\mathrm{shadows}} \approx \frac{360°}{T_{\mathrm{orbit}}} = 47.01 \frac{°}{\mathrm{h}}.$$

Recall that the rover itself is envisioned to drive with a very low velocity of a few millimeters per second, which can cause scenarios where the shadow movement is the principal visual change in a scene observed by the rover's cameras. As outlined in Section 3, the VO tracks corner features (that are features with strong contrast changes at the feature location) in a scene over several frames, which can potentially be the edges of the shadows.

Risk – We identified a high risk that the borders of the shadows will be tracked by the VO and their movement will cause an invalid ego-motion estimate by the VO. This risk is of principal importance and will have to be investigated thoroughly, creating task-specific test data sets, in simulation and in the laboratory.

Soil Composition and Rock Distribution

The surface soil composition and the rock distribution on Phobos is an unknown factor with a high uncertainty [5]. Due to the limited resolution of existing orbital images, the surface properties need to be estimated using data from other celestial bodies, e. g., the Moon [38]. The surface composition is of importance to NavDLR, as the VO and partially also the SGM need structured visual information to observe the rover movement or compute depth images.

In the case of very smooth surfaces, e. g., a pure layer of dust, and few to no visible rocks, poor performance can be expected from NavDLR. On the other hand, having a very rocky surface could potentially render the rover's locomotion and thus our obstacle detection obsolete.

There are no high-resolution images of Phobos available [5], however, the ERD [36] generally considers a rock and stone distribution similar to the Earth's moon. The ERD discusses several surface scenarios. In the end, it defines several *design cases* as potential scenarios, considering e. g., a dusty surface, rough sand, or powder with potentially dozens of rocks visible in the navigation camera images.

Risk – Our risk assessment requires an evaluation of the impact of surface smoothness to define the operational boundaries of NavDLR, especially for the feature tracking by the VO as well as for the disparity computation by the SGM. The susceptibility of the vision algorithm to surface smoothness is intrinsic to the algorithm's conceptual design. We expect the nominal scenarios defined in [36] to constitute navigation scenarios that are well suited for NavDLR. Extreme corner cases like the absence of rocks and a purely powder surface will only be considered in a limited number of stress tests.

Radiation

According to [36], the radiation environment of Phobos is identical to interplanetary space – with the only exception that the body of Phobos itself provides some shielding from the radiation. This therefore results in a high radiation environment with negative effects on the hardware.

Virmontois *et. al.* [39] describe in detail radiation effects on the CASPEX camera model with a CMV4000 chip, that is the camera model of the MMX rover. The space radiation is expected to cause the following two principal effects on the camera images:

- The total ionizing dose (TID) assumption for the mission is discussed in [4] and is approximated to $2.5\,\mathrm{kRad}$ in Silicon for a $3\,\mathrm{mm}$ aluminium shield thickness. As seen in [39, Fig. 4], the TID increases the dark current for all pixels, thus we expect an elevated mean dark current level for the camera chip, thus increased sensor noise. Nevertheless, the mission's TID lies within the acceptable boundaries of the camera, as the tests in [39] show.
- The displacement damage dose (DDD) for the expected radiation will increase the hot pixel count on the camera chip [39, Fig. 6].

Risk – Our risk assessment concludes that radiation-induced noise and hot pixels on the camera images can cause the SGM and the VO to fail to find corresponding matches between stereo pairs (for depth) or matches between corner features over consecutive images (for the VO pose estimation). The binning of the camera image for NavDLR is considered to reduce the influence of radiation noise, as any disturbances are each averaged over four pixels. We nevertheless consider space radiation a risk of medium importance and plan to test the robustness of NavDLR to these effects. Furthermore, mitigation strategies such image filtering to remove hot pixels ought to be considered.

Electrostatic Dust Lofting

Phobos is expected to contain fine-grained regolith that is produced mainly by thermal fragmentation and micrometeorite impacts. On airless planetary bodies, incoming ionizing (primarily solar UV) radiation and plasma potentially creates electrostatic lofting of dust particles several meters above the surface and a complex plasma sheath near the surface [40], which is expected to apply to Phobos as well. Once the dust is lofted, its charge will at some point be neutralized depending on the photoemission flux and plasma electron/ion fluxes in the plasma sheath, causing the particles ultimately to return to the ground (lofting velocities are expected to be much smaller than escape velocity for Phobos). Dust lofting is expected to occur between sunrise and noon of a Phod. There is a high uncertainty regarding the parameter values, in particular the flux in particles or mass per unit area per unit time to predict this phenomenon quantitatively for Phobos.

Risk – These particles can potentially accumulate on the rover, but this is mainly a concern for the solar panels and less critically for the camera lenses. There is a slight potential risk for our navigation if the camera lenses would get covered with a significant dust layer such that contrast suffers, but due to the high uncertainty on this phenomenon, no predictions or assessments can be made. Lofted dust particles observable in the camera scene – similar to the lunar horizon glow [40] – are expected to be no risk to the navigation algorithm as the particle density is very low.

Other Environmental Risks

There are other environmental risks that we do not detail further, as they are either common on Earth as well (consider e. g., lens flares) or are yet unknown due to the little available information on Phobos conditions.

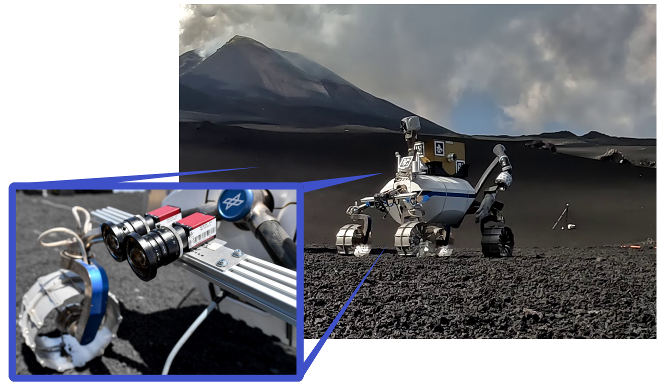

Figure 9: SUPER stereo-camera setup, here mounted between the front wheels on the LRU rover at a height similar to the MMX rover. Credits: Felix Oprean, DLR.

7. Generation of Representative Testing Data

To obtain the testing data, one needs to consider two principal aspects:

- A sensor suite that is able to perceive the environment similar to the MMX rover sensors.
- Environments that resemble the Phobos environment as closely as possible.

Sensor Unit for Planetary Explorations Rovers (SUPER)

First, we present our principal sensor suite. In [32], we established that handheld navigation testing constitutes a valuable and logistically less challenging alternative to tests with fully integrated robots. We used the SUPER in the desert of Morocco to create a multitude of data sets. SUPER is a sensor stack that consists of stereo cameras, an IMU, an on-board computer and an independent power supply. It was carried by hand to record data for the MADMAX data set [32].

To record the testing data for NavDLR, we follow a very similar approach. We adapted the SUPER cameras to be similar to the MMX rover, by mounting two Allied Vision Mako G-419C cameras (CMV4000 sensor with Bayer pattern) in a stereo configuration with a $60\,\mathrm{mm}$ baseline. Furthermore, we equip these cameras with Kowa LM6HC lenses to recreate the wide field of view of the MMX rover. The camera' specification is shown in Table 1 and the setup is shown in Fig. 9. For testing, we decouple the SUPER sensors from its computing body and use the sensor-stack in different configurations as outlined below.

Mt. Etna

The volcano Mt. Etna is a well-known planetary analog testing site that provides an environment similar to the Moon in terms of appearance and several geological aspects [41, p. 109ff]. Etna is frequently used for planetary robotic tests, for example our Moon-analog campaigns ROBEX [8] and ARCHES [9], or other tests that validate space exploration instruments, e.g., testing a ground penetrating radar for the ExoMars mission [42].

We use the opportunity of the ARCHES field test [9] to record Phobos analog navigation data there as well. Our experiment site is located at approximately $2700\,\mathrm{m}$ altitude. It is a slope

Table 1: MMX rover and SUPER camera specifications, showing different and *shared* properties.

	MMX rover	SUPER
Chip	*CMV4000*	
Sensor	*2048 × 2048 RGB (Bayer pattern), pitch 5.5 × 5.5 µm*	
Image for NavDLR	*1024 × 1024 grayscale, binning mode*	
Optics	Custom made (CNES)	Kowa LM6HC
Focal Length	7.85 mm	6 mm
Diagonal Field of View	118°	108°
Horizontal Field of View	81°	88°
Baseline	*horizontal, 60 mm*	
Camera Height above Ground	30-35 cm	adjustable

with varying steepness of approximately $500\,\mathrm{m}$ length with planar areas in between, positioned between the Cisternazza crater and the Monte Escrivà, one of the smaller volcanic cones located on the flank of Etna. The experiment site mostly consists of gravel with a diameter of few millimeters to centimeters, but additionally features scattered basaltic stones and ridges of significantly larger sizes. An impression of the area can be seen in Fig. 9 (right).

For data recording, we mount the SUPER camera stack between the wheels of our LRU Rover (see Fig. 9, left) and integrate them with the rover's power supply and OBC. This setup allows us to record Moon (Phobos) analog data with a testing platform that features the movement characteristics of a rover system, the correct sensor height, and even an illumination that is getting close to the target environment in terms of intensity. For ground truth (GT), we use a RTK GNSS.

We recorded a very diverse set of trajectories on Etna, with a total length of several hundreds of meters. They include long range straight drives as well as a curve-rich mapping of closer spaces. We recorded both in areas where obstacles are mostly absent, and in obstacle-rich environments.

However, not all environmental aspects can be correctly captured with our data recording on Etna. The coarse gravel of Etna, the relatively high movement speed of the LRU compared to the MMX rover, and the diffuse illumination due to the atmospheric scattering of the sunlight alter the conditions compared to the expected Phobos environment.

Planetary Exploration Laboratory (PEL)

Our second location for test-data recording is the Planetary Exploration Laboratory (PEL) of the Institute of Robotics and Mechatronics. It is a $10 \times 5\,\mathrm{m}$ sandbed with variable slope, filled with exchangeable soil. It can be seen in Fig. 10. The PEL features an ARTrack system to determine the sensor stack's pose using infrared markers. For the data from PEL, we mostly mount the SUPER cameras on a beam that is guided by a linear slide and whose orientation can be modified, to ensure smooth camera movements on predetermined paths for repeatability. This setup can be seen in the upper part of Fig. 10. Finally, in PEL we can change the number and the positions of the illumination sources, allowing us to test e. g., for moving shadows. PEL is currently in the process of undergoing the following modifications in order to resemble Phobos-conditions.

EAC-1A Soil as Phobos Analog—The EAC-1A lunar regolith simulant [43] is Königswinter basanite (similar to basalt). It looks fairly black. It is available in large quantities and can be used for testing, for example in the LUNA simulation hall, which is going to be built at the DLR/ESA premises in Cologne, Germany [43]. Engelschiøn *et al.* [43] analyze the physical and chemical properties of EAC-1A and conclude that it can serve as a low-cost lunar analog. We complement this analysis with the spectral properties of EAC-1A that we show in Fig. 8 for the standard reflectance and compare the measurements with reference data from Moon and with estimates for Phobos.

EAC-1A is optically rather well representative of the Moon. It is also featureless and red-sloped. But its reflectance is 3 - 6 times higher than Phobos soil, while its red slope is approximately representative. The mean grain size of EAC-1A is consistent with estimated models from [38] for Phobos soil, and matches model 1 in particular - but all Phobos models of [38] are putative. The conclusion is that a simple reduction of exposure times by a factor of 4 - 5 if using EAC-1A as a backdrop for rover navigation simulations is closely equivalent to a real scene on Phobos. We are therefore upgrading PEL with large quantities of EAC-1A to improve its Phobos representativity.

Illumination of PEL— We adapt PEL with respect to the illumination conditions with an approach that is similar to [44]. First of all, we use a strong illumination source with a continuous spectrum, the *Radium HRI-T 2000 W* metal halide lamp with quartz burner in a flood-light setup, similar to what is found in sports stadiums. The light has a color temperature of $7200\,\mathrm{K}$. As a smaller and more mobile addition, we are currently evaluating the procurement of tungsten-fresnel cinematic illumination sources, as was done by [44], however these feature color temperatures around $3000\,\mathrm{K}$, thus representing a trade-off between mobility and fidelity.

Second, the laboratory has to be modified to eliminate as much diffuse illumination as possible. In our case, this results in the use of black theater molton (cotton fleece) curtains hung around the testbed. Additionally, we will consider painting the walls and ceiling in black.

OAISYS Simulator

OAISYS [45] is a simulator for unstructured outdoor environments with a focus on planetary surroundings. The simulator can automatically generate a variety of different worlds based on configuration files. It is based on the free and open-source 3D computer graphics software Blender. As a

Figure 10: View of DLR-RM's Planetary Exploration Laboratory (PEL). The SUPER camera stack is mounted on a beam that is moved via a linear slide. The ground truth is measured via the ARTrack tracking system using spherical reflective markers. PEL is currently upgraded to a Phobos analog setup.

result, the simulator can achieve photorealistic image results, which are important for testing perception-based navigation components. For future space-exploration tasks which might also use modern machine learning techniques, OAISYS also renders material-based semantic information. The simulator is open source and is extensible, which gives it advantages compared to other planetary simulators like PANGU [46].

We use OAISYS to generate Phobos analog image data for predefined environments and movement trajectories. This provides us with the benefit of having exact ground-truth information and the ability to create representative illumination scenarios.

Rover System Simulator

Recall that one of the features of NavDLR is an active obstacle avoidance – OA – which cannot be tested purely on data or with the OAISYS simulation. Instead, it needs a testing setup that allows for an active feedback loop regarding

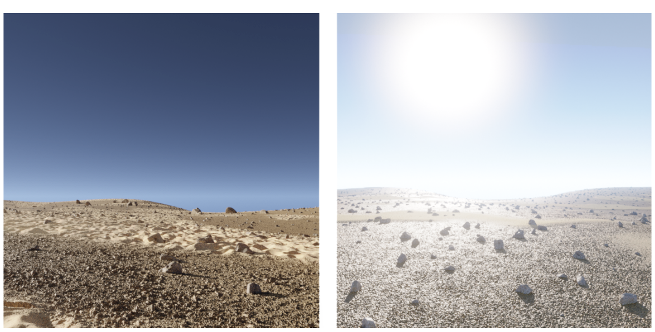

Figure 11: Photorealistic example images from the simulator OAISYS. While the impressions here show a desert with a blue sky, we are in the process of extending the simulator to generate Phobos-like surfaces without an atmosphere. We use this tool to create any Phobos scenarios that either require exact ground truth or cannot be replicated in a real-life setup.

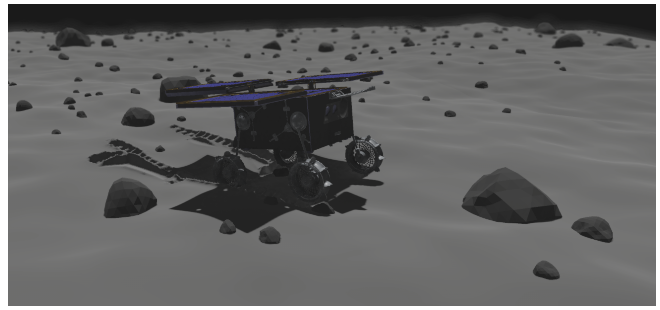

Figure 12: Simulation of the MMX rover system, its locomotion, and its interaction with the Phobos surface.

motion commands. For this, we interface our software with a system simulation of the MMX rover [47]. This simulation provides a detailed representation of the rover system and its interaction with the ground while driving. Motion commands that NavDLR sends to the simulation are translated into wheel motions. The camera images are provided as well, however not in photorealistic conditions. We nevertheless argue that the optical properties of navigation are tested otherwise and here only the active feedback loop of the NavDLR with the rover matters. A rendered scene from the simulator with the rover on Phobos is shown in Fig. 12.

Pyxel Radiation Simulator

As stated in Section 6, we expect the radiation effects on the camera images to pose a moderate risk to our navigation approach. To test the robustness of our navigation with respect to TID and DDD radiation effects, we use the PYXEL software tool [48]. It is a tool developed by ESA and allows the alteration of existing images by overlaying radiation noise in accordance with the camera sensor type and the expected character of space radiation. As it is a post-processing tool, we can potentially alter all our data sets and compare the change in navigation accuracy with added radiation influence.

8. ENVIRONMENTAL TESTING

We established the environmental risks in Section 6 and stated how we generate sensor data for these environmental risks in Section 7. Now, we focus on the consideration of these aspects in our testing campaign, which lies in the ENV category as discussed in Section 4.

As previously stated, the high uncertainty regarding the Phobos surface environment complicates the generation of comprehensive testing campaigns. Instead, our approach is to define reference cases that we use for our **validation tests**. Subsets of all other environment-parameter variations are investigated in the **stress tests**.

The validation tests will require NavDLR to accurately navigate, map, and detect all obstacles, in three different environments: We consider the Etna data as the first Phobos validation scenario. Even though these data do not capture the full spectrum of the Phobos environment, it is the best test case in terms of illumination strength, length of trajectories, rover motion, and scenario completeness. The second validation scenario is simulated by OAISYS featuring the most likely Phobos stone distribution, radiation effects on the camera, moving shadows, and a rover speed of $1\,\mathrm{mm/s}$. The third and final validation scenario replicates the second scenario as

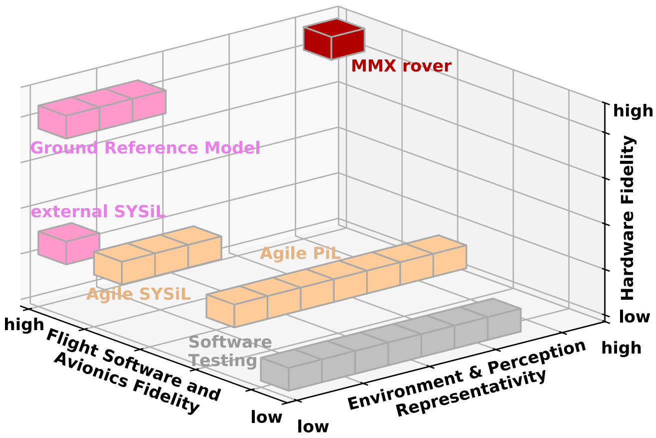

Figure 13: The testing dimensions for NavDLR. The actual MMX rover on Phobos is the highest level in all three dimensions. Our testing effort aims at covering the dimensions individually.

closely as possible, but with real-world data from PEL instead of simulation. The radiation effects will be added in a post-processing step of the image data. The PEL scenario allows for more realistic sensor effects than the simulation, but it is likely the parameters cannot be applied as precisely as in the simulation scenario.

All other environmental aspects are not part of the validation. Instead, variations of these parameters are used in the stress testing to create preliminary reports on the limits and the expected behavior of NavDLR.

9. Data Set

Complementary to this paper, we also publish the data that we use for testing NavDLR. Note that the development and testing of NavDLR is an ongoing process, thus the available data will be updated continuously. Our data is published online under `https://rmc.dlr.de/mmx_nav_testing`.

The data that we publish is related to the ENV category and presents our efforts with respect to to our preparation regarding the environmental conditions on Phobos.

Currently, data from experiments is available that are in the categories of *Unit Testing*, *Verification Testing*, the *Safety Stop Adventure*, and partially already the *Validation Testing*. Data for extreme environment cases i. e., the *Stress Testing*, are the last to be created, according to our project plan.

The data of all experiments are sorted in a standardized way, which is illustrated in the appendix. The input images are provided in the standardized `.png` format. The intermediate image products, such as depth or disparity use the `.pfm` and `.pgm` formats to store floating point and integer values. All other data is provided in text format. Note that we use timestamps to link all corresponding data: the images and image products have timestamps in their filename; the text files have one timestamp per data point.

Note two aspects: First, the GT data can vary from experiment to experiment. Outdoor data has only positional ground truth but indoor or simulative experiments provide full 6DoF poses as GT. Second, there are additional metadata files and visualization of the experiment scene that do not fit to the previously described data categories, but are only meant for illustrative purposes.

10. Discussion & Conclusion

This paper outlines our comprehensive testing for the NavDLR autonomous navigation of the MMX rover on Phobos. We discuss the three principal categories of testing in detail: testing the software algorithm for correctness and accuracy, testing the integration into the MMX rover OBC, and considering the environment of Phobos. The last point is of principal importance for NavDLR, as there is little information available regarding the Phobos environment and a wide range of environmental parameters need to be considered. Our testing of NavDLR for system integration is steered by the limited availability of the flight system-representative engineering models for testing, and our resulting adaptation is the creation of NavDLR-internal integration platforms.

As discussed, there are plenty of aspects for testing NavDLR to be considered, regarding our different test platforms and can be summarized into three different dimensions: This encapsulates the state of the flight software, in which environment NavDLR is tested, and on what hardware our component is executed. To categorize it, we create an overview analog to [20], which is shown in Fig. 13. The MMX rover on Phobos will combine the complete hardware and flight software in the actual mission environment. No testing setup here on Earth can easily capture all these three dimensions at once, however, each aspect can be tested individually.

Fig. 13 shows that our approach for testing NavDLR covers all three dimensions individually. Thus, it can be considered a comprehensive approach to ensure successful execution of the DLR autonomous navigation experiment on Phobos.

Appendix

The data set structure is identical for all different experiments, each having a unique data id. For each recorded experiment, the file structure is as follows:

```
DATA_ID
│
├── README.md # Metadata regarding the experiment
│
├── calibration
│   # All Data Regarding Calibration
│   │   # raw calibration information
│   ├── camera_calibration_callab.cal
│   │   # calib info of rectified left camera image
│   ├── camera_info_left.txt
│   │   # calib info of rectified right camera image
│   ├── camera_info_right.txt
│   │   # principal stereo information
│   ├── camera_stereo_parameters.txt
│   │   # calibrated transformation of camera
│   │   # to robot center frame
│   └── tf__robot_to_camera_left.csv
│
├── camera
│   # Rectified left and right camera images,
│   # unique corresponding timestamp (ts)
│   ├── img_rect_left_{ts}.png
│   ├── img_rect_right_{ts}.png
│   └── ...
⋮
```

```
⋮
├── cfg
│   # All relevant configuration files
│     ├── stereo_sgm_param.txt
│     └── ...
│
├── experiment_description
│   # Information regarding the experiment setup and
│   # camera scene view with tracking data visualized
│     ├── frame_setup.pdf
│     ├── scene_setup.pdf
│     ├── scene_camera_left_{ts}.jpg
│     └── ...
│
├── imu
│   # IMU data for attitude (in case it is available)
│     └── imu.csv
│
├── ground_truth
│   # ground truth (gt) tracking of different types
│         # Rover frames global tracking
│     ├── world_to_camera_left_gt.csv
│     ├── world_to_robot_gt.csv
│         # Scene Objects w.r.t to robot/worldframes
│     ├── camera_left_gt_to_object_X.csv
│     ├── robot_gt_to_object_X.csv
│     ├── world_to_object_X.csv
│         #Robot frames w.r.t. their start
│     ├── camera_left_start_to_camera_left_gt.csv
│     └── robot_start_to_robot_gt.csv
│
├── sgm
│   # disparity and depth data with unique timestamp
│     ├── img_depth_{ts}.pfm
│     ├── img_disp_{ts}.pfm
│     ├── img_disp_{ts}.pgm
│     └── ...
│
└── vo
     # visual odometry poses; image feature lists
     ├── img_corners_{harris/agast}_left_{ts}.txt
     ├── vo_out_absolute.csv
     ├── vo_out_relative.csv
     └── ...
```

Acknowledgments

We thank our colleagues from CNES and DLR who continuously supported our development of NavDLR, repeatedly answered our questions, and helped us with valuables reviews and feedback.

References

[1] T. Nakamura, H. Ikeda, T. Kouyama, H. Nakagawa, H. Kusano, H. Senshu, S. Kameda, K. Matsumoto, F. Gonzalez-Franquesa, N. Ozaki, Y. Takeo, N. Baresi, Y. Oki, D. J. Lawrence, N. L. Chabot, P. N. Peplowski, M. A. Barucci, E. Sawyer, S. Yokota, N. Terada, S. Ulamec, P. Michel, M. Kobayashi, S. Sasaki, N. Hirata, K. Wada, H. Miyamoto, T. Imamura, N. Ogawa, K. Ogawa, T. Iwata, T. Imada, H. Otake, E. Canalias, L. Lorda, S. Tardivel, S. Mary, M. Kunugi, S. Mitsuhashi, A. Doressoundiram, F. Merlin, S. Fornasier, J.-M. Reess, P. Bernardi, S. Imai, Y. Ito, H. Ishida, K. Kuramoto, and Y. Kawakatsu, "Science operation plan of Phobos and Deimos from the MMX spacecraft," *Earth, Planets and Space*, vol. 73, no. 1, 2021, cited by: 8; All Open Access, Gold Open Access, Green Open Access.

[2] K. Kuramoto, Y. Kawakatsu, M. Fujimoto, A. Araya, M. A. Barucci, H. Genda, N. Hirata, H. Ikeda, T. Imamura, J. Helbert, S. Kameda, M. Kobayashi, H. Kusano, D. J. Lawrence, K. Matsumoto, P. Michel, H. Miyamoto, T. Morota, H. Nakagawa, T. Nakamura, K. Ogawa, H. Otake, M. Ozaki, S. Russell, S. Sasaki, H. Sawada, H. Senshu, S. Tachibana, N. Terada, S. Ulamec, T. Usui, K. Wada, S.-i. Watanabe, and S. Yokota, "Martian moons exploration MMX: sample return mission to Phobos elucidating formation processes of habitable planets," *Earth, Planets and Space*, vol. 74, no. 1, 2022.

[3] S. Ulamec, P. Michel, M. Grott, U. Böttger, H.-W. Hübers, N. Murdoch, P. Vernazza, K. Özgür, J. Knollenberg, K. Willner, M. Grebenstein, S. Mary, P. Chazalnoel, J. Biele, C. Krause, T.-M. Ho, C. Lange, J. T. Grundmann, K. Sasaki, M. Maibaum, O. Küchemann, J. Reill, M. Chalon, S. Barthelmes, R. Lichtenheldt, R. Krenn, M. Smíšek, J. Bertrand, A. Moussi, C. Delmas, S. Tardivel, D. Arrat, F. IJpelaan, L. Melac, L. Lorda, E. Remetean, M. Lange, O. Mierheim, S. Reershemius, T. Usui, M. Matsuoko, T. Nakamura, K. Wada, H. Miyamoto, K. Kuramoto, J. LeMaitre, G. Mas, M. Delpech, L. Celine, A. Rafflegeau, H. Boirad, R. Schmisser, C. Virmontois, C. Cenac-Morthe, A. Besson, and F. Rull, "A rover for the JAXA MMX Mission to Phobos," in *70th International Astronautical Congress*, 2019.

[4] J. Bertrand, S. Tardivel, F. Ijpelaan, E. Remetean, A. Torres, S. Mary, M. Chalon, F. Buse, T. Obermeier, M. Smíšek, A. Wedler, J. Reill, and M. Grebenstein, "Roving on Phobos: Challenges of the MMX rover for space robotics," in *15th Symposium on Advanced Space Technologies in Robotics and Automation*, 2019.

[5] P. Michel, S. Ulamec, U. Böttger, M. Grott, N. Murdoch, P. Vernazza, C. Sunday, Y. Zhang, R. Valette, R. Castellani, J. Biele, S. Tardivel, O. Groussin, L. Jorda, J. Knollenberg, J. T. Grundmann, D. Arrat, G. Pont, S. Mary, M. Grebenstein, H. Miyamoto, T. Nakamura, K. Wada, K. Yoshikawa, and K. Kuramoto, "The MMX rover: performing in situ surface investigations on Phobos," *Earth, Planets and Space*, vol. 74, no. 1, 2022.

[6] M. Vayugundla, T. Bodenmüller, M. J. Schuster, M. G. Müller, L. Meyer, P. Kenny, F. Schuler, M. Bihler, W. Stürzl, B.-M. Steinmetz, J. Langwald, A. Lund, R. Giubilato, A. Wedler, R. Triebel, M. Smíšek, and M. Grebenstein, "The MMX Rover on Phobos: The preliminary design of the DLR Autonomous Navigation Experiment," in *2021 IEEE Aerospace Conference (50100)*, 2021, pp. 1–18.

[7] A. Wedler, M. J. Schuster, M. G. Müller, B. Vodermayer, L. Meyer, R. Giubilato, M. Vayugundla, M. Smisek, A. Dömel, F. Steidle, P. Lehner, S. Schröder, E. Staudinger, B. Foing, and J. Reill, "German Aerospace Center's advanced robotic technology for future lunar scientific missions," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 379, no. 2188, 2021.

[8] A. Wedler, M. Vayugundla, H. Lehner, P. Lehner, M. J. Schuster, S. G. Brunner, W. Stürzl, A. Dömel, H. Gmeiner, B. Vodermayer, B. Rebele, I. L. Grixa, K. Bussmann, J. Reill, B. Willberg, A. Maier, P. Meusel, F. Steidle, M. Smíšek, M. Hellerer, M. Knapmeyer,

F. Sohl, A. Heffels, L. Witte, C. Lange, R. Rosta, N. Toth, S. Völk, A. Kimpe, P. Kyr, and M. Wilde, "First results of the ROBEX analogue mission campaign: Robotic deployment of seismic networks for future lunar missions," in *Proceedings of the International Astronautical Congress*. International Astronautical Federation, 2017.

[9] M. J. Schuster, M. G. Müller, S. G. Brunner, H. Lehner, P. Lehner, R. Sakagami, A. Dömel, L. Meyer, B. Vodermayer, R. Giubilato, M. Vayugundla, J. Reill, F. Steidle, I. von Bargen, K. Bussmann, R. Belder, P. Lutz, W. Stürzl, M. Smíšek, M. Maier, S. Stoneman, A. F. Prince, B. Rebele, M. Durner, E. Staudinger, S. Zhang, R. Pöhlmann, E. Bischoff, C. Braun, S. Schröder, E. Dietz, S. Frohmann, A. Börner, H. Hübers, B. Foing, R. Triebel, A. O. Albu-Schäffer, and A. Wedler, "The ARCHES space-analogue demonstration mission: Towards heterogeneous teams of autonomous robots for collaborative scientific sampling in planetary exploration," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5315–5322, 2020.

[10] Office of the Chief Engineer, "NASA Procedural Requirements, NPR 7123.1C, Appendix E. Technology Readiness Levels," National Aeronautics and Space Administration, Tech. Rep., 2020.

[11] M. Maimone, Y. Cheng, and L. Matthies, "Two years of visual odometry on the Mars Exploration Rovers," *Journal of Field Robotics*, vol. 24, no. 3, pp. 169–186, 2007.

[12] L. Matthies, M. Maimone, A. Johnson, Y. Cheng, R. Willson, C. Villalpando, S. Goldberg, A. Huertas, A. Stein, and A. Angelova, "Computer vision on Mars," *International Journal of Computer Vision*, vol. 75, no. 1, pp. 67–92, 3 2007.

[13] M. W. Maimone, P. C. Leger, and J. J. Biesiadecki, "Overview of the Mars Exploration Rovers' autonomous mobility and vision capabilities," in *IEEE International Conference on Robotics and Automation, Space Robotics Workshop*, 2007.

[14] D. Helmick, Y. Cheng, D. Clouse, L. Matthies, and S. Roumeliotis, "Path following using visual odometry for a Mars rover in high-slip environments," in *2004 IEEE Aerospace Conference Proceedings (IEEE Cat. No.04TH8720)*, vol. 2, 2004, pp. 772–789 Vol.2.

[15] A. E. Johnson, S. B. Goldberg, Y. Cheng, and L. H. Matthies, "Robust and efficient stereo feature tracking for visual odometry," in *2008 IEEE International Conference on Robotics and Automation*, 2008, pp. 39–46.

[16] A. Rankin, M. Maimone, J. Biesiadecki, N. Patel, D. Levine, and O. Toupet, "Mars Curiosity rover mobility trends during the first 7 years," *Journal of Field Robotics*, vol. 38, no. 5, pp. 759–800, 2021.

[17] M. Maimone, N. Patel, A. Sabel, A. Holloway, and A. Rankin, "Visual odometry thinking while driving for the Curiosity Mars Rover's three-year test campaign: Impact of evolving constraints on verification and validation," in *2022 IEEE Aerospace Conference*, 2022, pp. 1–10.

[18] V. Verma, F. Hartman, A. Rankin, M. Maimone, T. Del Sesto, O. Toupet, E. Graser, S. Myint, K. Davis, D. Klein, J. Koch, S. Brooks, P. Bailey, H. Justice, M. Dolci, and H. Ono, "First 210 solar days of Mars 2020 Perseverance robotic operations - mobility, robotic arm, sampling, and helicopter," in *2022 IEEE Aerospace Conference*, 2022, pp. 1–20.

[19] R. Rieber, M. McHenry, P. Twu, and M. M. Stragier, "Planning for a Martian road trip - the Mars2020 mobility systems design," in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 01–18.

[20] C. Matthes, M. Stumbo, J. Foley, T. Dang, and J. T. Rojas, "How to build a rover: An overview of the Mars 2020 mission's Vehicle System Testbed," in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–12.

[21] S. Brooks, T. Litwin, J. Biesiadecki, N. Abcouwer, T. Del Sesto, M. McHenry, S. Myint, P. Twu, and D. Wai, "Testing Mars 2020 flight software and hardware in the Surface System Development Environment," in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–13.

[22] L. Gerdes, M. Azkarate, J. R. Sánchez-Ibáñez, L. Joudrier, and C. J. Perez-del Pulgar, "Efficient autonomous navigation for planetary rovers with limited resources," *Journal of Field Robotics*, vol. 37, no. 7, pp. 1153–1170, 2020.

[23] M. Winter, S. Rubio, R. Lancaster, C. Barclay, N. Silva, B. Nye, and L. Bora, "Detailed description of the high-level autonomy functionalities developed for the ExoMars Rover," in *Proceedings of the Symposium on Advanced Space Technologies in Robotics and Automation*, 06 2017.

[24] R. Ward, W. Hamilton, N. Silva, and V. Pereira, "ExoMars VisLoc – the visual localisation system for the ExoMars rover," in *DASIA 2016 - Data Systems In Aerospace*, ser. ESA Special Publication, L. Ouwehand, Ed., vol. 736, Aug. 2016, p. 17.

[25] K. McManamon, R. Lancaster, and N. Silva, "ExoMars rover vehicle perception system architecture and test results," in *12th Symposium on Advanced Space Technologies in Robotics and Automation*. ASTRA, 2013.

[26] P. Weclewski, R. Marc, B. Brayzier, W. Hamilton, C. Barclay, M. Dinsdale, A. Daoud-Moraru, A. Dysli, M. Braun, D. Hamill, T. Cooper, V. Croes, D. Aspinal, and G. Roy, "Sample Fetch Rover guidance, navigation and control subsystem - an overview," in *Proceedings of the Symposium on Advanced Space Technologies in Robotics and Automation*, 06 2022.

[27] S. Gunes-Lasnet, M. Winnendael, G. Diaz, S. Schwenzer, D. Pullan, A. Kisdi, J.-L. Josset, V. Ciarletti, D. Barnes, A. Griffiths, G. Paar, E. Allouis, L. Waugh, M. Woods, and A. Shaw, "SAFER: The promising results of the Mars mission simulation in Atacama, Chile," in *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space*. ESA, 06 2014.

[28] M. Woods, A. Shaw, E. Tidey, B. V. Pham, L. Simon, R. Mukherji, B. Maddison, G. Cross, A. Kisdi, W. Tubby, G. Visentin, and G. Chong, "Seeker — autonomous long-range rover navigation for remote exploration," *Journal of Field Robotics*, vol. 31, no. 6, pp. 940–968, 2014.

[29] J. Wang, J. Li, S. Wang, T. Yu, Z. Rong, X. He, Y. You, Q. Zou, W. Wan, Y. Wang, S. Gou, B. Liu, M. Peng, K. Di, Z. Liu, M. Jia, X. Xin, Y. Chen, X. Cheng, and X. Liu, "Computer vision in the teleoperation of the Yutu-2 rover," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. V-3-2020, pp. 595–602, 08 2020.

[30] R. A. Hewitt, E. Boukas, M. Azkarate, M. Pagnamenta, J. A. Marshall, A. Gasteratos, and G. Visentin, "The Katwijk beach planetary rover dataset," *The International Journal of Robotics Research*, vol. 37, no. 1, pp. 3–12, 2018.

[31] P. Furgale, P. Carle, J. Enright, and T. Barfoot, "The Devon Island rover navigation dataset," *The International Journal of Robotics Research*, vol. 31, no. 6, pp. 707–713, 2012.

[32] L. Meyer, M. Smíšek, A. Fontan Villacampa, L. Oliva Maza, D. Medina, M. J. Schuster, F. Steidle, M. Vayugundla, M. G. Müller, B. Rebele, A. Wedler, and R. Triebel, "The MADMAX data set for visual-inertial rover navigation on Mars," *Journal of Field Robotics*, vol. 38, no. 6, pp. 833–853, 2021.

[33] M. Vayugundla, F. Steidle, M. Smíšek, M. Schuster, K. Bussmann, and A. Wedler, "Datasets of long range navigation experiments in a Moon analogue environment on Mount Etna," in *Proceedings of the International Symposiom on Robotics*, 01 2018.

[34] NASA, "NASA Planetary Data System - Cartography and Imaging Sciences Discipline Node," https://pds-imaging.jpl.nasa.gov/, 2022, retrieved September 1, 2022.

[35] H. Hirschmüller, "Semi-Global Matching - motivation, developments and applications," in *Photogrammetric Week*, D. Fritsch, Ed. Wichmann, September 2011, pp. 173–184.

[36] S. Tardivel, J. Biele, F. Buse, A. Hoerdt, R. Lichtenheldt, P. Michel, K. Ogawa, S. Schroeder, K. Willner, F. Wolff, and R. Ziese, "Phobos Environment Requirement Document for the MMX rover mission," CNES, DLR, et al., Tech. Rep., 2019, internal.

[37] C. M. Pieters, L. A. Taylor, D. McKay, S. Wentworth, R. Morris, and L. Keller, "Spectral characterization of lunar mare soils," in *Lunar and Planetary Science Conference*, Mar. 2000, p. 1865, data archived under: https://web.archive.org/web/20210426140726/http://www.planetary.brown.edu/relabdocs/LSCCsoil.html.

[38] H. Miyamoto, T. Niihara, K. Wada, K. Ogawa, H. Senshu, P. Michel, H. Kikuchi, R. Hemmi, T. Nakamura, A. Nakamura, N. Hirata, S. Sasaki, E. Asphaug, D. Britt, P. Abell, R. Ballouz, O. Banouin, N. Baresi, M. Barucci, and K. Willner, "Surface environment of Phobos and Phobos simulant UTPS," *Earth, Planets and Space*, vol. 73, 12 2021.

[39] C. Virmontois, J.-M. Belloir, M. Beaumel, A. Vriet, N. Perrot, C. Sellier, J. Bezine, D. Gambart, D. Blain, E. Garcia-Sanchez, W. Mouallem, and A. Bardoux, "Dose and single-event effects on a color CMOS camera for space exploration," *IEEE Transactions on Nuclear Science*, vol. 66, no. 1, pp. 104–110, 2019.

[40] X. Wang, J. Schwan, H.-W. Hsu, E. Grün, and M. Horányi, "Dust charging and transport on airless planetary bodies," *Geophysical Research Letters*, vol. 43, no. 12, pp. 6103–6110, 2016.

[41] L. Preston, M. Grady, and S. Barber, "CAFE – Concepts for Activities in the Field for Exploration – TN2: the catalogue of planetary analogues," The Planetary and Space Sciences Research Institute, The Open University, UK, Dec. 2012, under ESA contract: 4000104716/11/NL/AF.

[42] V. Ciarletti, S. Clifford, A. J. Vieau, B. Lustrement, R. Hassen Khodja, and P. Cais, "Results from the first field tests of the WISDOM GPR (2018 ExoMars mission)," in *42nd Lunar and Planetary Science Conference*. Houston: Lunar and Planetary Institute, 2011.

[43] V. Engelschiøn, S. Eriksson, A. Cowley, M. Fateri, A. Meurisse, U. Kueppers, and M. Sperl, "EAC-1A: A novel large-volume lunar regolith simulant," *Scientific Reports*, vol. 10, p. 5473, Mar. 2020.

[44] P. Ludivig, A. Calzada-Diaz, M. O. Mendez, H. Voos, and J. Lamamy, "Building a piece of the Moon: Construction of two indoor lunar analogue environments," in *Proceedings of the International Astronautical Congress*. International Astronautical Federation, 2020.

[45] M. G. Müller, M. Durner, A. Gawel, W. Stürzl, R. Triebel, and R. Siegwart, "A photorealistic terrain simulation pipeline for unstructured outdoor environments," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 9765–9772.

[46] S. Parkes, I. Martin, M. Dunstan, and D. Matthews, "Planet surface simulation with PANGU," in *Int. Conf. on Space Operations*, 2004.

[47] F. Buse, A. Pignède, J. Bertrand, S. Goulet, and S. Lagabarre, "MMX rover simulation - robotic simulations for Phobos operations," in *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–14.

[48] D. Lucsanyi, T. Prod'homme, H. Smit, F. Lemmel, P.-E. Crouzet, P. Verhoeve, and B. Shortt, "Pyxel: a novel and multi-purpose Python-based framework for imaging detector simulation," in *High Energy, Optical, and Infrared Detectors for Astronomy VIII*, A. D. Holland and J. Beletic, Eds., vol. 10709, International Society for Optics and Photonics. SPIE, 2018, p. 107091A.

BIOGRAPHY

Lukas Meyer *is a researcher in the department of "Perception and Cognition" at the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), since 2018. He works in the field of planetary robotics with a focus on testing navigation algorithms and perception for robust manipulation. His work includes the testing of navigation solutions during planetary analogue campaigns on a volcano or in the desert. Lukas received his Master's and Bachelor's degree in Mechanical Engineering and Engineering Science from the Technical University of Munich, Germany.*

Mallikarjuna Vayugundla *is a researcher in the department of "Perception and Cognition" at DLR. He is working on autonomous navigation algorithms for rovers. He joined DLR as a full-time researcher in 2016. He received his Masters degree in Autonomous Systems from the Bonn-Rhine-Sieg University of Applied Sciences in 2015.*

Patrick Kenny *is a researcher in the Onboard Software Systems group, Institute for Software Technology at DLR. Patrick has worked in the medical device and automotive industries before joining the DLR in 2020, where his work includes robust, distributed software platforms for spacecraft. He received his Bachelor's degree in Engineering (Mechatronics) and his Master's degree (Engineering Science) from the University of Auckland, New Zealand.*

Michal Smíšek *is a researcher in the department of "Perception and Cognition" at DLR. His specialization is software development and the calibration of cameras. Having previously been involved in several European space robotics projects ending with planetary analogue demonstrations, he is responsible for the management of the development of the MMX navigation solution NavDLR. Michal received his M.Sc. and PhD. degrees in Applied Mathematics from the Slovak University of Technology.*

Jens Biele *studied experimental physics at the University of Kaiserslautern and at Imperial College, London. He did a PhD in geosciences at the Free University Berlin with the Alfred-Wegener-Institute for Polar and Marine Research in 1998. He then spent one year as a Postdoc with the Max-Planck-Institute for Chemistry in Mainz. Since 1999 he works at DLR in Cologne as research associate and as in payload manager in various projects (Rosetta Lander, MASCOT, MMX rover) and has been involved in a number of major solar system exploration studies. His field of special expertise is payloads and small systems, in particular landers, for missions to small bodies in the solar system, as well as thermophysics, IR radiometry and regolith properties.*

Alessandro Maturilli *is a staff researcher and deputy leader in the department of "Planetary Laboratories" at the Institute of Planetary Research, DLR, since 2016. He works in the field of planetary spectroscopy with a focus on mineral/rock characterization of planetary surfaces. He is the manager of the Planetary Spectroscopy Laboratory (PSL) and Co-I or participating scientist on many planetary missions from ESA, NASA, and JAXA. Alessandro received his Master's degree in Mathematics from the University "La Sapienza" in Rome, Italy and his PhD in Aerospace Engineering from the University "Federico II" of Naples, Italy*

Marcus G. Müller *is a researcher in the department of "Perception and Cognition" at DLR since 2016 and Ph.D. student at ETH Zurich. He is the leader of the MAV Exploration Team at the Institute of Robotics and Mechatronics (DLR-RM), where he is working on autonomous navigation algorithms for MAVs. Before joining DLR he conducted research at the Jet Propulsion Laboratory (JPL) of NASA in Pasadena, USA, where he worked on visual inertial navigation for MAVs and on radar signal processing. Marcus received his Master's and Bachelor's degree in Electrical Engineering from the University of Siegen, Germany.*

Wolfgang Stürzl *is a senior research scientist in the department of "Perception and Cognition" at the Institute of Robotics and Mechatronics of DLR. His research interests include computer vision for mobile robots, in particular using multi-camera and wide-angle imaging systems, and bio-inspired visual navigation of flying systems.*

Martin J. Schuster *is senior researcher and leader of the Planetary Exploration Operations team at the Institute of Robotics and Mechatronics of DLR. He received his Ph.D. (Dr.-Ing.) degree in Computer Science (CS) from the University of Bremen in 2019, his first Master's degree in CS from the Georgia Institute of Technology in Atlanta, USA in 2010, his Bachelor's and second Master's degree in CS from the Technical University of Munich in 2008 and 2011, and his Bachelor's degree in Philosophy from the Ludwig-Maximilians-Universität München (LMU) in 2014. His research focus is on multi-robot SLAM in GNSS-denied areas.*

Tim Bodenmüller *is senior researcher at the Institute for Robotics and Mechatronics at DLR. He joined DLR as a full-time researcher in 2001. and is working on 3D-sensing, robotic middleware and software design. He further is member of the institutes software engineering group. He received his Dipl.-Ing. degree in electrical and information engineering in 2001 from the Technical University of Darmstadt and his PhD in electrical engineering in 2009 from the Technical University of Munich.*

Armin Wedler *is a senior researcher and head of the planetary exploration group at the institute of Robotics and Mechatronics of DLR. He received his Ph.D. (Dr.-Ing.) degree in mechanical engineering from the University of Hanover in 2010, his diploma and bachelor degrees in robotics and mechatronics in 2004 also from the University of Hanover. Mr. Wedler worked in several space and research projects such as Exomars, Lunar Lander, Dexhand and the light weight rover unit (LRU). He is project leader or spokesperson for several projects and is also active in the ISCEG. His research focuses on the developments of intelligent mobile platforms for exploration and terrestrial usages.*

Rudolph Triebel *received his PhD in 2007 from the University of Freiburg in Germany. The title of his PhD thesis is "Three-dimensional Perception for Mobile Robots". From 2007 to 2011, he was a postdoctoral researcher at ETH Zurich, where he worked on machine learning algorithms for robot perception within several EU-funded projects. Then, from 2011 to 2013 he worked in the Mobile Robotics Group at the University of Oxford, where he developed unsupervised and online learning techniques for detection and classification applications in mobile robotics and autonomous driving. Since 2013, Rudolph works as a lecturer at the Technical University of Munich, where he teaches master level courses in the area of Machine Learning for Computer Vision. In 2015, he was appointed as leader of the Department of Perception and Cognition at the Robotics Institute of DLR.*