i-SAIRAS 2024

Automatic Driving Command Evaluation for the MMX Rover IDEFIX

[1]*Kretschmer, Marie, [1]Buse, Fabian

*lead presenter

[1]marie.kretschmer@dlr.de, German Aerospace Center (DLR), Germany

The exploration rover IDEFIX is part of the JAXA mission "Martian Moons eXploration" (MMX). The rover is designed by the Centre National d'Études Spatiales (CNES) and the German Aerospace Center (DLR) to explore the surface of the Martian moon Phobos. IDEFIX will be the first wheeled robotic system in a milli-g environment. Challenges for the rover mobility include the milli-g environment, the largely unknown surface conditions, and the severely interrupted communication scheme. The early phase of the mission will focus on manual driving via direct commands to the locomotion subsystem to validate the behavior of the rover and previously made assumptions. Efficient planning and evaluation of these commands are required to find the safest path for IDEFIX to the target. A simulation-based algorithm ranks given commands based on risk factors. Monte Carlo-style simulations are used to generate the required data. This paper presents the design and challenges of a simulation-based command evaluation algorithm for the MMX rover IDEFIX.

1. Introduction

The Centre National d'Études Spatiales (CNES) and the German Aerospace Center (DLR) provide the IDEFIX Rover to the JAXA's "Martian Moons eXploration" (MMX) Mission, exploring the Martian moon Phobos. The rover will give context from Phobos's surface, enhancing the scientific quality of the samples returned by the MMX spacecraft. During the main spacecraft's landing rehearsal, the rover will be dropped onto the surface from about 40 to 50m height. After that, the rover will reorient itself from its random landing orientation towards the belly and start its mission on the surface of Phobos. The uncertainties made the development difficult and will render the operations phase challenging. Regolith composition and behavior are complicated to predict [1]. The low gravity of about 1/2000th of Earth's makes laboratory experiments challenging. During the about 100-day-long operations phase, the team in charge of mobility will have to quickly adapt the planning tools to the new information gathered about the environment [2].

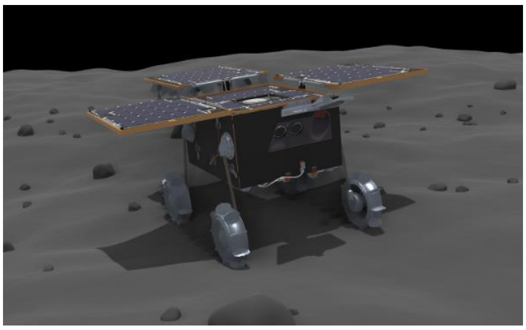

Figure 1 IDEFIX in simulation

The rover's locomotion system comprises four individually driven but unsteered wheels. Each wheel is attached to a 0.275m long leg, whose angle can be controlled [3]. The leg angles are nominally only used during the uprighting and sun-pointing procedures. During driving, the legs are kept stationary, but the driving is not limited to a specific rover pose. [4]

Simulations are used in the rover development to design and verify algorithms, such as the uprighting and sun-pointing functions. During the operation phase, the focus of the simulations shifts toward planning, validating, and understanding driving commands. This paper focuses on the validation part, in which commands previously designed to reach a target are compared against predefined rules to evaluate them automatically. These tools need to be integrated with the overall logic of IDEFIX operations. This integration leads to tight timing requirements as the turnaround time for new commands is often less than a day. The resulting requirements are an automated evaluation of commands that, using sufficient computational power, complete within hours and still provide trustworthy results.

2. Problem Definition

The first mobility phase of IDEFIX after successful landing, uprighting and solar panel deployment on Phobos will focus on manual driving via direct commands to the locomotion subsystem. During this phase no autonomous navigation is used, each path is manually planned based on the telemetry data from the rover, images of IDEFIX navigation cameras and the orbiter images. A simulated camera image is shown in Figure 2. Due to the interrupted communication schedule via the MMX spacecraft and the nominal mission duration of 100 days, the response time on Earth is restricted and should be used efficiently [1]. Therefore, pre-validation and planning of each path is required to move the rover as safely as possible on Phobos. The communication scheme requires to plan each movement two Phobos days in advance. This, in combination with the largely unknown surface behavior of Phobos, leads to inaccuracy in determining the rover’s position and orientation.

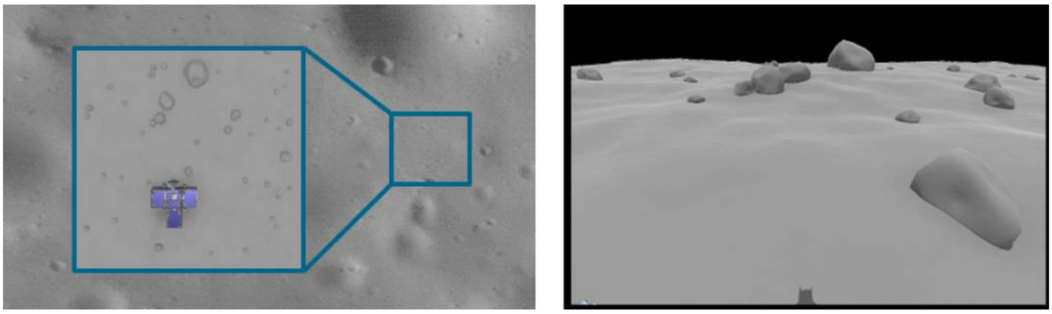

Figure 2 Available images for planning. Left: example for image and the top view of the simulation visualisation. Right: simulated image of IDEFIX front NavCAM

A driving command is defined by three input parameters: maximum wheel rate, delta position, and delta heading [4]. The delta position is the effective commanded distance the rover should travel from its start position to its target on a circular track. The delta heading is the angle between the current rover yaw orientation and the target yaw orientation, and the maximum wheel rate defines the speed of the fastest wheel. Figure 4 shows one ideal driving command on the xy-plane. This enables the rover to traverse the surface in straight lines, curves, or perform point turns [4]. Combining multiple commands to a driving command chain enables the rover to drive to a given target with a defined heading.

The challenge is to find the optimal set of driving commands for safe and efficient rover operation while considering the uncertainty of the environment and the inaccuracies in rover position, orientation, and environment. This process requires a simulation-based tool enabling the team on Earth to evaluate the risk of each path by simulating the path in a reconstructed environment while capturing the uncertainties in a Monte Carlo-style approach. Ranking of the different command sets is based on an evaluation index that compares the resulting movements with predefined movement rules. The movement rules consider system and subsystem limitations as well as operational, system, and safety constraints. These rules define thresholds for allowed position and heading drift, rover tilt in roll, and pitch direction, movement properties like wheel sinkage and slippage, as well as the distance to obstacles that may be critical for rover operations. The movement rules also limit the allowed velocity, acceleration, the largest allowed turn angle and the minimum delta position and delta heading combination to reduce the error related to the driving command.

3. Evaluation Algorithm

The path evaluation algorithm identifies and validates each driving command chain's risk and defines a ranking between the path options. The algorithm is visualized in Figure 3. The evaluation process is performed for each path and is repeated until the individual risk of all path options are calculated. At the end the paths are ranked based on the results.

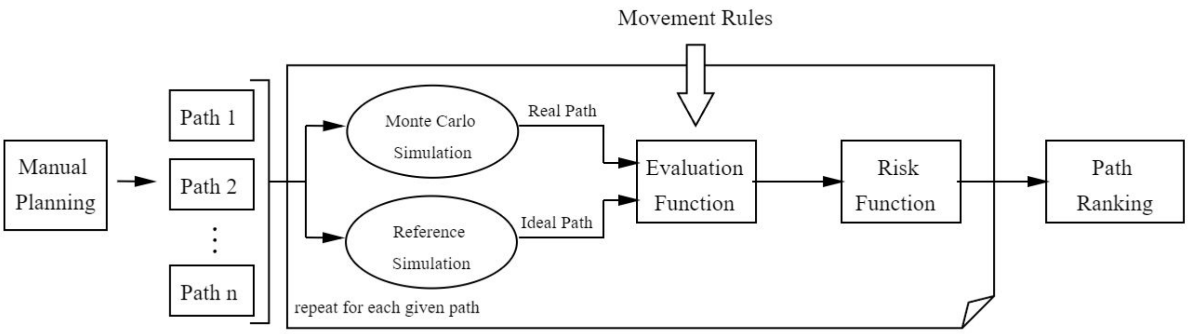

Figure 3 Architecture of the Path Evaluation Algorithm

The simulation is divided into two parts. First, the reference path is simulated to define the reference for the risk assessment. This path is based on the assumed rover position, orientation, regolith, and environment. A first superficial risk assessment can be performed for the reference path to exclude path options with risks above a certain threshold, for example, excluding paths with significant environmental collisions. The second part is a Monte Carlo-style simulation to capture uncertainties, varying them between each simulation run with agreed distributions. With all results available, the evaluation function computes the metrics outlining path performance and risk This function also uses the mentioned movement rules and thus incorporates the latest available information on system health and environment understanding. Each independently computed value corresponds to a possible risk that can occur during movements. These values are the position, orientation, wheel surface interaction, and the risk of hitting an obstacle with the rover. The results are then summarized into a single metric in the risk function to describe the total risk of each path compared to the reference path.

3.1 Simulation

The simulations used in this context are based on the simulations used in many other applications in the IDEFIX rover project [5]. A multi-physics model defining the rover's electro-mechanical system based on Modelica lays the foundations. Different levels of detail are available for the various components. In this application, the focus is not on the internal mechanism of the rover. Thus, it is assumed that, for example, the locomotion system functions optimally, and thus, ideal outputs for all motors can be assumed.

Similarly, other components like shutters and solar panels are assumed to be rigid. This configuration allows fast simulation speeds. The same logic applies to the model of the environment, which is based on the DLR ContactDynamics Library for Modelica [6]. The focus here is on the regolith contact using the SCM contact model (see [6]) to model accurate regolith interactions with deformations. These modeled soil deformations are essential as the tracks left behind by the front wheels can significantly impact the rear wheels. Parts of the actual flight software are used to control the rover, especially the locomotion system. A mockup of the main operating software is used to integrate the locomotion flight software. Commands to the simulated rover are defined in a Lua script, with commands matching the telecommands given to the actual rover. This setup allows easy and accurate execution of locomotion telecommands. This simulator configuration is exported as a Functional Mockup Unit (FMU), is packaged with other required resources, and can be executed by a Python library. Results are stored in an HDF5 file, which is further processed.

3.2 Evaluation

During the evaluation process, each evaluation index I is calculated independently based on the resulting movement, the environment and the movement rules for the rover. Those movement rules might change during the mission. Eight indices are calculated: Position drift I_{pos} heading drift I_{θ}, roll orientation deviation I_{ψ}, pitch orientation deviation I_{ϕ}, wheel slip rate I_s, wheel slip angle I_{β}, wheel sinkage I_{ds} and the risk of encountering an obstacle I_{obs}. Each risk may result from several interdependent causes. For example, wheel slip can be caused by the regolith, the inclination, an obstacle collision or the rover being steered with a too large heading angle. Single causes may impact multiple risks. For example, a collision, with an obstacle will be detected as such but also affect position, heading and slip.

The evaluation index I_{max} is the total risk of all individual risks multiplied with weighting factors w. The index is specified between 0 and 1. The weightings can be changed depending on the desired goal of the movement. To get an idea of the average behavior of the trajectory, the mean evaluation index $\bar{I}_{max}$ is calculated over the entire number n of simulations.

$$I_{max} = max\left(I_{obs} \cdot w_{obs}; I_{pos} \cdot w_{pos}; I_{\beta} \cdot w_{\beta}; I_s \cdot w_s; I_{\theta} \cdot w_{\theta}; I_{\psi} \cdot w_{\psi}; I_{\phi} \cdot w_{\phi}; I_{ds} \cdot w_{ds}\right) \quad (1)$$

$$\bar{I}_{max} = \frac{1}{n} \sum_{j=1}^{n} I_{max,j} \quad (2)$$

The final path ranking is done by comparing the mean evaluation index of all given path options with each other. Also, each path's total time and energy consumption is included in the ranking. Values of around 0.5 represent the expected risk in the environment. This is due to the predicted sinkage, wheel slip, and orientation deviations due to the terrain. This study assumes a nominal wheel slip of 20 % while driving on soft soil. A slip above 70% is marked as critical. The following gives more details on the position and obstacle indices. For further explanations of the other evaluation values, see [7].

Position

Two types of paths are compared to describe the positional error. The theoretical path is a projection of the commanded path onto the terrain with an assumed fixed slip, and the actual path is the path the rover took during the dynamic simulation. First, the local error describes the Euclidean distance between the theoretical and actual paths for each single run during the Monte Carlo simulation at each time step. Second, the global error describes the Euclidean distance between each actual simulation run and the actual path of the reference simulation at each time step. Thus, the local error describes the accuracy, while the global error gives a statistical representation of the deviations to be expected. The global error is compared against a tolerance that is defined around the reference path, defining the area in which it is expected that IDEFIX should move during command execution.

This tolerance area is defined by a radius. At the start, this is set to the standard deviation of the uncertainty of the initial position. During command execution, the tolerance radius grows based on the current command, staying constant when not turning and growing while turning, see Figure 4. The final risk is the ratio of the final global error and the final tolerance radius, exceeding when the average rover position is outside the tolerance.

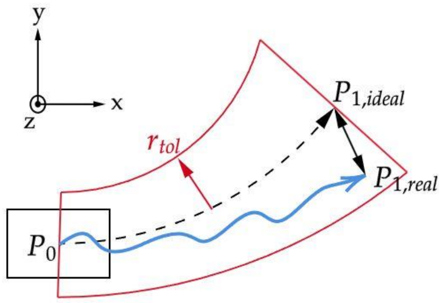

Figure 4 Position risk definition

$$I_{pos} = \frac{\sqrt{(x_{ideal} - x_{real})^2 + (y_{ideal} - y_{real})^2 + (z_{ideal} - z_{real})^2}}{r_{tol}} \quad (3)$$

Obstacle Index

Rocks of different sizes are the obvious obstacles on the surface, but other areas will also be deemed risky. Highly inclined terrain, like corners of craters or areas with unknown surface conditions due to the limitation of the navigation images, are possible obstacles that should be avoided. Hitting obstacles with the wheels, chassis, or solar panels can lead to mission failure. IDEFIX can get stuck, fall over, or move to an uncontrolled position, or the collision can damage wheels, legs, solar panels, or the chassis. Figure 5 shows IDEFIX while encountering different obstacles.

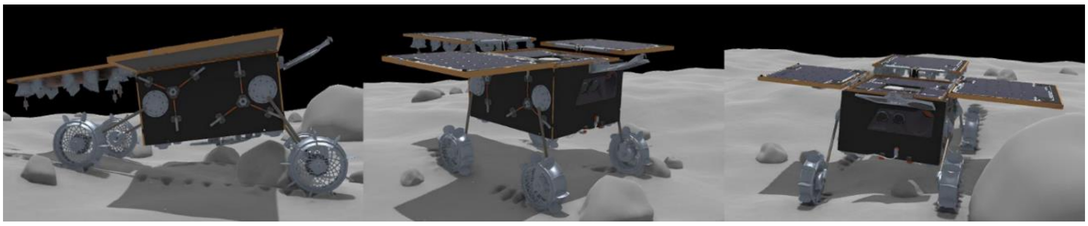

Figure 5 Obstacles in the range of IDEFIX. Left: obstacle under the chassis; Center: obstacle on the wheel path; Right: obstacle and the left solar panel

The locomotion subsystem limits the size and type of obstacles that can be passed safely. All obstacles near the rover are identified and ranked to evaluate their risk. In the case of rocks, this depends on their size, but in general, this maps to the impact when hit on the rover: Level 1 (low impact), Level 2 (medium impact), and Level 3 (critical impact). The obstacle collision check is performed on the wheels, the chassis, and the solar panels individually. The evaluation index for obstacle collision $I_{obs,k,l}$ is calculated by multiplying the detected number of obstacles $n_{obs,k,l}$ with a risk factor $f_{obs,l}$ depending on its level.

$$I_{obs,k,l} = n_{obs,k,l} \cdot f_{obs,l} \tag{4}$$

where $k \in \{wheel, solar\ panel, chassis\}$ and $l \in \{L1, L2, L3\}$. The chance of hitting an obstacle of a given level is used for the total evaluation. A total numeric risk can be deferred by combining the chance with a weighting factor based on the obstacle level. Both the chance and risk allow a good understanding of the collision risk during a movement of IDEFIX.

4 Example

To validate the path evaluation algorithm, an example scenario with different command chains to the same target is designed. In this scenario, a target one meter in front of the rover is selected. Figure 6 shows a simulated image of IDEFIX NavCams. The NavCam images only provide information about the near surroundings. There is a rock located near the right side of the rover. This obstacle might impact the rover while passing by it. Figure 7 shows the initial assumed position of IDEFIX for this scenario, as well as the nearby obstacles with an included buffer zone as an extra safety margin. Three possible driving command chains are compared, indicated by the blue, red, and green lines in Figures 6 and 7.

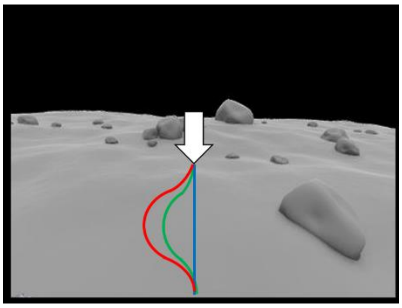

Figure 6 Simulated NavCam image of IDEFIX with possible path options

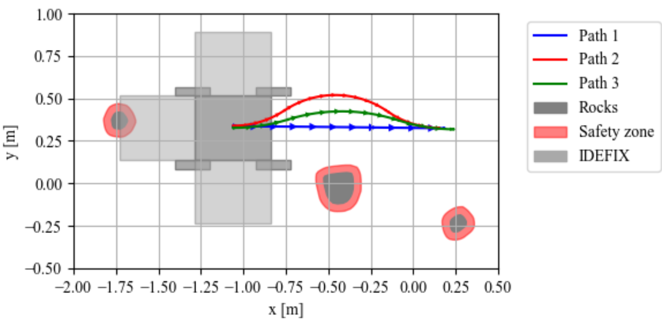

Figure 7 Top view of IDEFIX initial position and the reference path trajectories

Path 1 in blue commands the rover straight, expecting an obstacle collision. Paths 2 and 3 circumnavigate the obstacle but are based on different desired clearances to the obstacle. The results of the evaluation process are visualized in Figure 8 via a color map in the xy-plane. The colors represent the total risk between 0 and 1 over the whole trajectory, where 1 indicates a critical area. The xy-coordinate corresponds to the central rover reference frame. If a point is only crossed by a single path, the maximum of all individual risk factors is shown. If multiple simulations pass a single point, the average risk of the run-specific maxima is used.

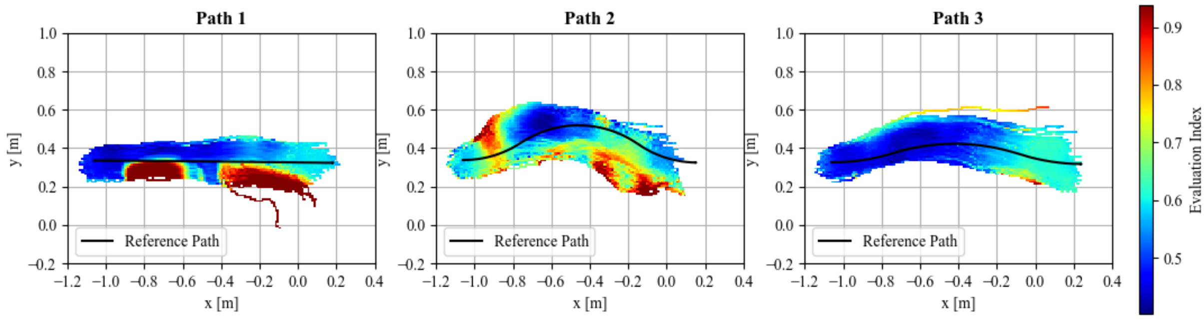

Figure 8 Result map

The analysis of Path 1 clearly shows two critical areas, see Figure 8. They are located just before and after the obstacle, which is on the path of the right wheels as shown in Figure 7. The obstacle itself is only classified as a Level 2 obstacle. Thus, the collision with the obstacle itself is not critical, but the resulting slip, when the wheels need to traverse the obstacle, leads to high risk. If the rover's initial position is far enough in the y direction, the path of IDEFIX severely deviates. In these cases, the entire wheel hits the rock, causing the wheel to get stuck and the rover to drift to the right. In total, 92% of all simulations for Path 1 result in a collision with the obstacle. Path 2 aims to avoid the obstacle at a considerable distance at the cost of sharper turns. However, the increased heading angle in Path 2 significantly increases the slip ratio and the slip angle of the wheel, visible in the red areas at the beginning of the movement. This high slip leads to an increase in position error, resulting in collisions with the obstacle in 38% of all simulations. Path 3 follows a smoother trajectory and has the best overall performance. The total risk is 51%, and a probability of 28.4% of an obstacle collision. The lower slip leads to better position accuracy and, thus, a reduced risk of collision.

The map also illustrates that moving too far to the right increases the position and heading error so that the rover moves more to the left as commanded. Analyzing the data shows that this drift results from the terrain. The rover is moving slightly downhill, which increases the slip ratio of the wheels as well as the roll and pitch orientation of the rover, and therefore, the rover changes its path. A good measure of the ability to reach the target position is the statistics of the final local error. Figure 9 shows the local error as a box plot. The mean deviation occurs on Path 1 with a value of 0.052 m, which is due to the obstacle collision. The largest variance appears on Path 2. The position error ranges from a minimum value of 0.002 m to a maximum value of 0.113 m. The reason for this deviation is high slip during turning. This high slip is shown by the red areas in Path 2 in Figure 8. The left rear wheel is the inner wheel of the narrow left turn, resulting in a high slip of 85%. When the rover is turned to the right, the left rear wheel is navigated through the wheel track of the other wheels. This effect again results in a higher slip. These two high slip incidents lead to significant errors in the position and heading of the entire rover. The standard deviation of Path 3 is 0.02 m to 0.04 m. Consequently, this is the path with the smallest position variance. Table 1 summarizes the evaluation results. It contains the average risk of each operation, the position and heading error of the target, the percentage probability of hitting an obstacle, the energy required, and the total time.

Table 1 Path evaluation results.

	Risk [%]	Obstacles [%]			Energy [mAs]	Time [s]
		L1	L2	L3		
Path 1	64.28	0.0	89.0	0.0	648.32	310.4
Path 2	72.74	0.0	38.1	0.0	1141.55	557.4
Path 3	50.82	0.0	28.4	0.0	928.49	467.9

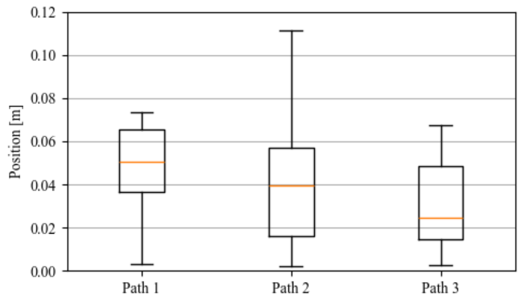

Figure 9 Box plot of the position error

5 Conclusion

Simulation of different command chains and path alternatives are key for efficient operations of the IDEFIX rover on Phobos. Risk identification and ranking of the alternatives are needed in order to find the safest path. An algorithm was developed to evaluate and rank different path options to a target position as explained in this text. The functionality of the developed algorithm is demonstrated by the example scenario where different driving command chains are evaluated and ranked. The algorithm identifies the origin of individual risks during the movement, like obstacles, slip, or the slope of the terrain. With the visualization, it is possible to show high-risk areas on the surface. Different driving command chains can be ranked depending on their evaluation index and required operation time. Due to the possibility to individually change the weightings of the evaluation parameters, a specified ranking of the given driving command chains is possible.

References

[1] F. Buse *et al.*, "Mobility on the Surface of Phobos for the MMX Rover - Simulation-aided Movement planning" in *17th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA)*, Leiden, The Netherlands, 2023.

[2] S. Barthelmes *et al.*, "Driving in Milli-G: The Flight Model of the MMX Rover Locomotion Subsystem and its Integration & Testing in the Rover" in *17th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA)*, Leiden, The Netherlands, 2023.

[3] S. Ulamec *et al.*, "A Rover for the JAXA MMX Mission to Phobos" in *70th International Astronautical Congress (IAC)*, Washington, DC, USA, 2019.

[4] J. Skibbe, S. Barthelmes und F. Buse, "Locomotion Control Functions for the Active Chassis of the MMX Rover" in *2021 IEEE Aerospace Conference*, 2021, doi: 10.1109/AERO50100.2021.9438423.

[5] F. Buse, A. Pignède, J. Bertrand, S. Goulet und S. Lagabarre, "MMX Rover Simulation - Robotic Simulations for Phobos Operations" in *2022 IEEE Aerospace Conference*, Big Sky, MT, USA, 2022, doi: 10.1109/AERO53065.2022.9843391.

[6] F. Buse, A. Pignède und S. Barthelmes, "A Modelica Library to Add Contact Dynamics and Terramechanics to Multi-Body Mechanics" in *15th International Modelica Conference*, Aachen, Germany, 2023, 433-442, doi: 10.3384/ecp204433.

[7] M. Kretschmer "Development of an Automatic Mobility TC Evaluation for the OPS Phase of the MMX Rover IDEFIX" Master thesis, FH Aachen – University of Applied Sciences, Aachen, Germany, 2024