View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Institute of Transport Research:Publications

IEEE ICRA 2020 Workshop on Opportunities and Challenges in Space Robotics

Into the Unknown – Autonomous Navigation of the MMX Rover on the Unknown Surface of Mars' Moon Phobos

Michal Smíšek, Martin J. Schuster, Lukas Meyer, Mallikarjuna Vayugundla, Florian Schuler, Bernhard-Michael Steinmetz, Marcus G. Müller, Wolfgang Stürzl, Markus Bihler, Jörg Langwald, Daniel Seth, Peter Schmaus, Neal Y. Lii, Patrick Kenny, Andreas Lund, Kilian Höflinger, Klaus Strobl, Tim Bodenmüller, and Armin Wedler

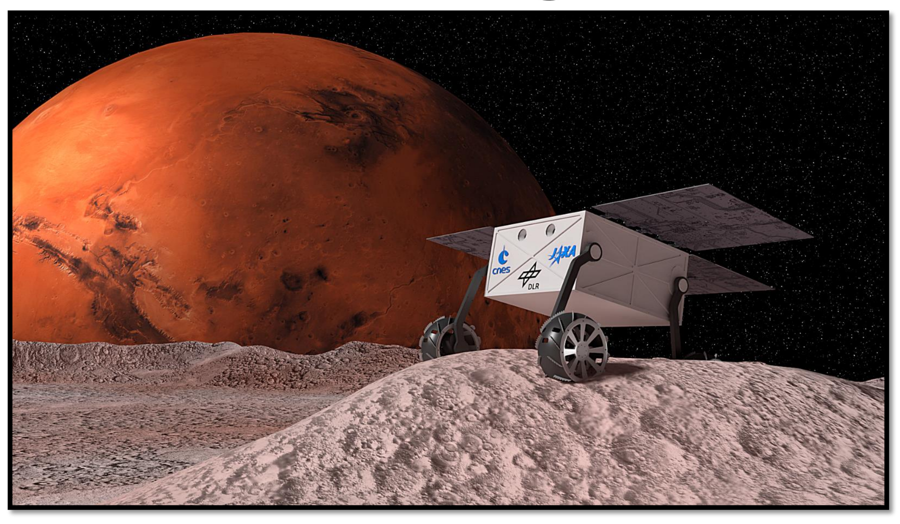
Institute of Robotics and Mechatronics (RM), German Aerospace Center (DLR)

MMX Rover Navigation

The goal of our team is to develop an autonomous navigation solution for the MMX Rover. The rover, built jointly by CNES and DLR, is scheduled to launch onboard the MMX Spacecraft by JAXA, and is destined to operate on the surface of Phobos, the larger of Mars' two moons.

The MMX mission is unprecedented on many levels. In this poster we present a list of risks to be tackled as well as the approaches to be taken in the framework of developing an autonomous navigation software for MMX Rover.

Challenges and Planned Solutions



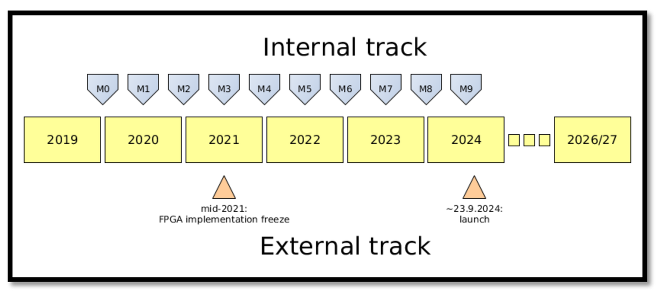
The artist's impression of the MMX rover mission. The main sources of challenges are Phobos, rover configuration, mission constraints and OBC resource limits.

The table below is a categorized list of challenges and corresponding planned solutions.

Type	Challenge	(Planned) Solution
Phobos	• Unmapped terrain	• Use orbiter map + build own map
	• Obstacles endangering rover • Possibly too many obstacles	• Obstacle detection (feature) • Navigation software shuts down
	• Possibly slippery terrain • Possibly featureless terrain • Sharp shadows • Travelling shadows • Dust lifted by the rover	• Visual odometry (feature) • Consider alternative detectors • Consider shadow feature removal • Consider compensation • Dust objects removal
Rover	• Skid-steered locomotion • Possible wheel slip • Body-inserted nav cameras	• Visual odometry (feature) • Visual odometry as primary sensor • Planning in current view
	• Possible camera failure • Possible over-/under-exposure • Possible camera miscalibration	• Camera failure feedback • Exposure feedback • Calibration feedback
	• High-noise IMU • High-slip wheel odometry	• Don't rely on • Don't rely on
OBC	• Many simultaneous processes • FPGA resources limited • Memory limited • CPU limited • Persistent data storage limited	• Use hypervisor to encapsulate • Allocate FPGA resources cautiously • Take max 100 MB RAM • Take max 10 s / stereo image pair • Efficient data structures, compress
Ops	• Very tight energy budget • Infrequent comm windows • Low data throughput • Limited on-Earth testability • Limited confidence in higher modes of autonomy	• One driving day every ~4th day • Implement high(er)-level autonomy • Maximize onboard processing • Utilize commissioning phase • Utilize lower modes first, while commissioning higher ones slowly
MGMT	• Manage workforce fluctuation • Many TBDs on higher level • High sw quality demands	• New-guy-friendly documentation • Stay agile (conflicts with ↓) • Code rules, style, standards, ECSS, standardized build toolchain

DLR

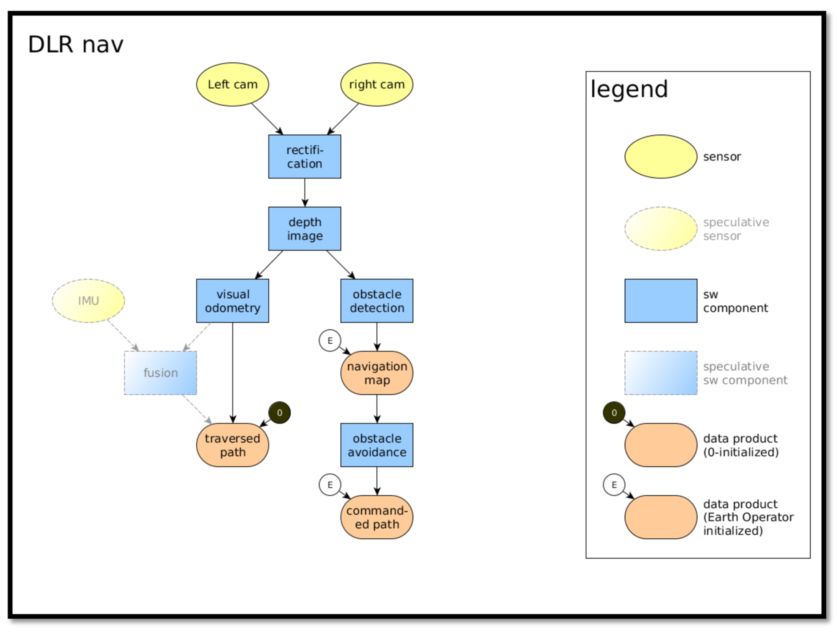
Time Plan and Organization



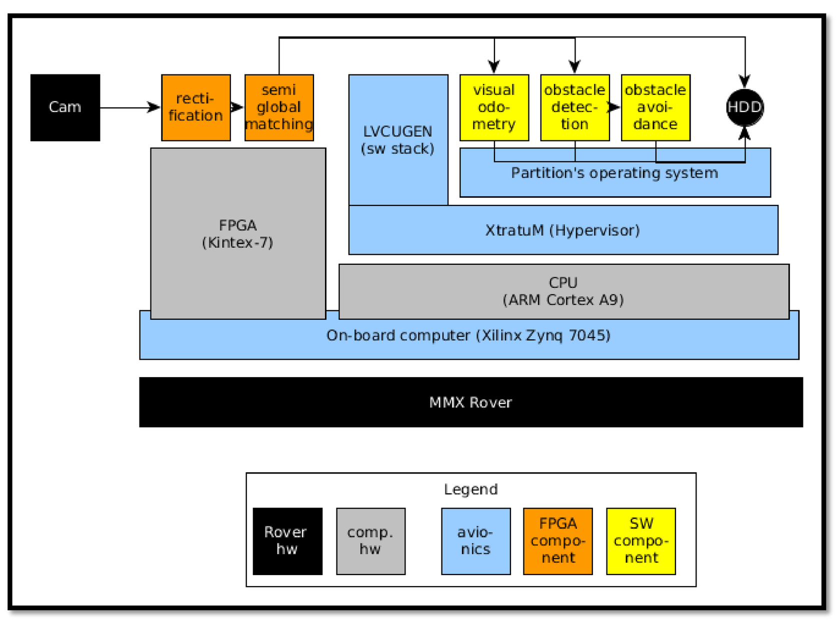
We adopt an iterative software development model. Each iteration lasts 6 months. At its end, a new version of navigation solution exists.

Each iteration begins and ends with an Internal Milestone Review Meeting. Current status is reviewed and next period is planned. FPGA implementation should be frozen by mid 2021, software implementation at the latest before the launch (September 2024).

Software Solution



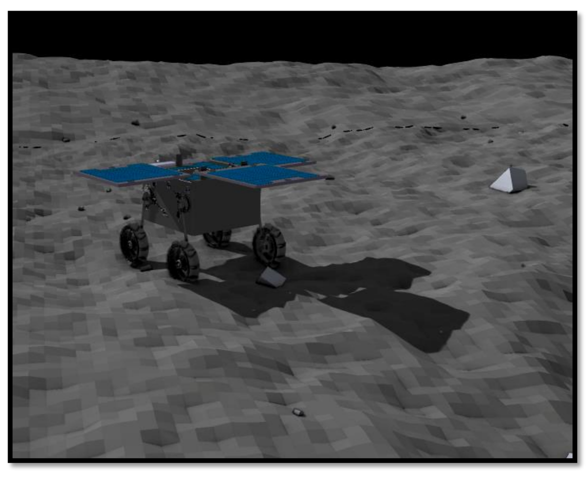
Navigation sw architecture



Software-to-hardware plan

From stereo image pairs, depth images are computed on an FPGA. This is the basis for a stereo visual odometry to estimate the robot's trajectory. Depth images and camera images will be used by obstacle classification and possibly further mapping modules to create maps. Such obstacle and map information can then be used to realize autonomous emergency stop behavior up to future reactive obstacle avoidance or path planning modules to support (semi-)autonomous operation.

Test Plan



Simulation (credit: DLR-SR)

Dataset acquisition (bg: DFKI SherpaTT)

Laboratory test

A resource-efficient test plan of consecutively increasing complexity to bootstrap the development is being formulated. We consider:

1. Tracks: Software, Agile model (COTS version), Flight model
2. Integration: Software-/Processor-/Hardware-/Rover-in-the-Loop
3. Test types: Simulation, Dataset-driven, Live experiment testing
4. Test levels: Unit test, component verification, ideal operation, Phobos nominal operation, extreme conditions

[1] Bertrand et al., "Roving on Phobos: Challenges of the MMX Rover for Space Robotics," *ASTRA 2019.*
[2] Ulamec et al., "A Rover for the JAXA MMX Mission to Phobos," *IAC* 2019.
[3] Schuster et al., "Towards Autonomous Planetary Exploration: The Lightweight Rover Unit (LRU), its Success in the SpaceBotCamp Challenge and Beyond," *JINT 2017.*