

# SpaceHopper: A Small-Scale Legged Robot for Exploring Low-Gravity Celestial Bodies

Alexander Spiridonov, Fabio Buehler, Moriz Berclaz, Valerio Schelbert, Jorit Geurts, Elena Krasnova, Emma Steinke, Jonas Toma, Joschua Wuethrich, Recep Polat, Wim Zimmermann, Philip Arm, Nikita Rudin, Hendrik Kolvenbach, and Marco Hutter

**Abstract**— We present *SpaceHopper*, a three-legged, small-scale robot designed for future mobile exploration of asteroids and moons. The robot weighs 5.2 kg and has a body size of 245 mm while using space-qualifiable components. Furthermore, *SpaceHopper*'s design and controls make it well-adapted for investigating dynamic locomotion modes with extended flight-phases. Instead of gyroscopes or fly-wheels, the system uses its three legs to reorient the body during flight in preparation for landing. We control the leg motion for reorientation using Deep Reinforcement Learning policies. In a simulation of Ceres' gravity (0.029 g), the robot can reliably jump to commanded positions up to 6 m away. Our real-world experiments show that *SpaceHopper* can successfully reorient to a safe landing orientation within 9.7 deg inside a rotational gimbal and jump in a counterweight setup in Earth's gravity. Overall, we consider *SpaceHopper* an important step towards controlled jumping locomotion in low-gravity environments.

**Index Terms**— Legged Robots, Space Robotics and Automation, Engineering for Robotic Systems

## I. INTRODUCTION

Low-gravity celestial bodies like asteroids and moons are becoming increasingly popular targets for space missions. Their composition makes them valuable both from a scientific and economic perspective [1]. Moreover, the dawn of a new Space Economy also promises to reduce the cost of visiting these objects. The launch of small-scale CubeSats has already become affordable for many enterprises [2]. With this in mind, we aim to develop a robot enabling scalable mobile asteroid exploration. Traditionally used, wheeled locomotion systems, such as the Mars Rover Perseverance [3] or the proposed MMX Rover [4], are not very well suited for asteroids. Low gravity causes loss of wheel traction, and the rough terrain requires a very versatile system [5]. Other robots, such as the asteroid hopper MASCOT [6], JAXAs Minerva [7] or legged climbing robots like LEMUR [8] and ReachBot [9] offer alternative approaches to microgravity mobility. However, their non-articulated design or reliance on specialized grippers or docking adapters make them less

This research was supported in part by the Swiss National Science Foundation through the National Centre of Competence in Digital Fabrication (NCCR dfab). Corresponding author Alexander Spiridonov. Jonas Toma and Wim Zimmerman are with the ZHAW, 8401 Winterthur. All other authors are with the Robotic Systems Lab, ETH Zurich, 8092 Zurich, Switzerland. (e-mail: aspidonov@ethz.ch; fabuehle@ethz.ch; mberclaz@ethz.ch; valerisc@ethz.ch; jgeurts@ethz.ch; ekrasnova@ethz.ch; steinke@ethz.ch; tomajon1@students.zhaw.ch; jwuethrich@ethz.ch; recep.polat@ethz.ch; zim-mewim@students.zhaw.ch; philip.arm@mavt.ethz.ch; rudinn@ethz.ch; hendrik.kolvenbach@mavt.ethz.ch; mahutter@ethz.ch;).

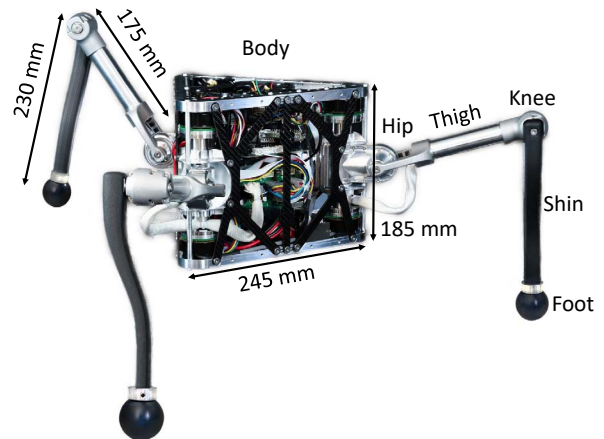


Fig. 1: The small-scale low-gravity robot *SpaceHopper* fully assembled with labeled parts and dimensions.

suited for efficient, wide-ranging exploration of asteroids. Dynamically walking robots such as ANYmal [10] and Spot [11], on the other hand, have shown impressive capabilities to traverse unstructured terrain on Earth. Thus, a variant of these robots has a high potential for mobile exploration of celestial bodies [12]. Some pathfinders, such as Spot Nebula [13] and SpaceBok [14], have already been tested on analog sites but are not yet specifically engineered for space [15]. Moreover, in the low-gravity environment of asteroids, gaits with extended flight phases are increasingly efficient [16]. Dynamic jumping locomotion is also advantageous compared to other walking gaits, as it can swiftly overcome obstacles by simply jumping over them. However, it also poses the challenge of stabilizing the robot's attitude in low-gravity during the jump phases [17]. Traditionally, attitude control in space and on low-gravity celestial bodies uses reaction wheels. However, to save weight and eliminate the complexity of an additional subsystem, one can omit reaction wheels and control the robot's attitude using the legs only. SpaceBok, which only has two degrees of freedom (DOF) per leg, used this idea to perform jumping locomotion in a planar representation of a microgravity environment [18].

In this paper, we present *SpaceHopper*, a small-scale, lightweight, and space-qualifiable robot to investigate controlled locomotion in low-gravity scenarios. The proposed system uses jumping as the primary form of locomotion. Moreover, the robot can control its attitude in low-gravity

using solely its legs. Besides demonstrating the mechanical and electrical feasibility of the system, we demonstrate *SpaceHopper*'s jumping and reorientation capability in simulation and partly on the real system. Specifically, we provide the following contributions:

- We present a three-legged, CubeSat-sized, and lightweight robot specifically designed for controlled low-gravity locomotion.
- In a simulation, we validate the *SpaceHopper* design by showing accurate low-gravity attitude control and jumping to commanded positions.
- We demonstrate 2D attitude control and vertical jumping on the hardware using a gimbal and a counterweight setup.

## II. SYSTEM DESIGN

### A. System Overview

*SpaceHopper* is a three-legged robot used as a research platform for low-gravity locomotion in space. We tested *SpaceHopper* in a lab environment (see Sec. IV and V) and accordingly estimate it to be at technology readiness level (TRL) four. The body is triangular and symmetric, with one leg attached to each corner (Fig. 1). *SpaceHopper* can fit into a 27U CubeSat [19] with its legs fully tucked in. At 5.2 kg, the robot is significantly lighter than the 54 kg CubeSat limit.

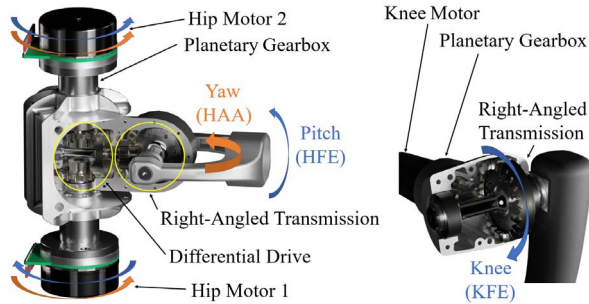


Fig. 2: Cut section of the hip and knee with annotated transmissions, movements, and components.

### B. Leg Design

Jumping and reorientation is the main locomotion goal of *SpaceHopper* and a guiding feature for the leg design. Having three legs is a unique feature of *SpaceHopper* and allows for weight saving compared to a classic four-legged design (one leg weighs 1.17 kg, including all motors). Static standing is possible with three legs, which increases stability compared to a two-legged design. Four legs enable static locomotion, but given the low-gravity environment, this gait is less efficient [16] and not favorable. *SpaceHopper*'s legs each have three active DOFs as this vastly outperforms legs with only two active DOFs in attitude control [18]. Two of those active DOFs are in the hip and are realized through a differential drive (Sec. II-C), which enables hip flexion/extension (HFE) and hip abduction/adduction (HAA). The other active DOF is placed in the knee, enabling flexion/extension (KFE). Here, we used a classical right-angled drive to place the knee motor

efficiently in the thigh (see Fig. 2). The chosen actuators are Maxon EC 45 flat [20] in the hip and ECX Torque 22 M [21] in the knee. We chose these motors for their high torque-to-weight/size ratio and the availability of space-graded versions with the same specifications. The resulting maximum joint torques, accounting for the transmission ratios, are shown in Table I. As depicted in Fig. 2, we chose a pinion-crown gear combination instead of classic bevel gears for all right-angled transmissions because pinion-crown gear combinations allow for looser tolerances in axial positioning [22]. This freedom is advantageous, considering the large temperature fluctuations in space.

|                | Knee  |       | Hip   |             |       |     |
|----------------|-------|-------|-------|-------------|-------|-----|
|                | Motor | Joint | Motor | Pitch & Yaw | Pitch | Yaw |
| Max torque N m | 0.078 | 0.740 | 0.152 | 1           | 2.8   | 1.5 |

TABLE I: Maximum achievable torques with chosen motors, as limited by winding temperature, power, and stall torque.

### C. Hip Design

Many modern quadrupeds like ANYmal [10] by ANYbotics or Boston Dynamics' Spot [23] use a serially linked drivetrain. This drivetrain has a simple mechanical design and high efficiency [24]. Nevertheless, for *SpaceHopper*'s special use case in low-gravity environments, we chose a parallel linked drivetrain in the form of a differential drive for the hip actuation based on the following considerations:

1) *Combined Torque for Jumping*: The differential drive allows both motors to contribute to the pitch torque in all configurations, in contrast to a serially linked drivetrain. *SpaceHopper* can benefit significantly from the differential drive, as jumps rely on the HFE motion for which the motors can work in conjunction and provide double the torque (see Fig. 2). For combined pitch and yaw motions, the torques are split up according to Equation 1, which is not critical, as combined motion only occurs during reorientation when the loads are not significant.

$$\begin{pmatrix} \gamma_{h1} \\ \gamma_{h2} \end{pmatrix} = \begin{pmatrix} 1/(2i_{hd}i_{hr}i_{hd}) & 1/(2i_{hp}) \\ -1/(2i_{hd}i_{hr}i_{hp}) & 1/(2i_{hp}) \end{pmatrix} \cdot \begin{pmatrix} \tau_p \\ \tau_y \end{pmatrix} \quad (1)$$

The torques of the two hip motors and the pitch, yaw joint are  $\gamma_{h1}, \gamma_{h2}, \tau_p, \tau_y$ . The gear ratios of the differential, the right-angled, and the planetary transmission are indicated by  $i_{hd}, i_{hr}, i_{hp}$ , respectively.

2) *Static Motor Placement*: Using a differential drive in the hip allows us to place two motors per leg inside the body. This placement of the motors is very space-efficient and eases thermal management, dust protection, and cable management for space applications.

In order to achieve the same range of motion (ROM) in hip flexion/extension as offered by a serially linked drivetrain, the differential drive is paired with a right-angled, second-stage transmission, as shown in Fig. 2. The resulting ROM is  $\pm 85^\circ$ , which allows *SpaceHopper* to develop higher take-off velocities while jumping off. The hip abduction/adduction

has a ROM of  $\pm 70^\circ$ , while the knee has  $\pm 170^\circ$ . The second stage transmission also enables fine-tuning the torque of the respective hip motions independently. As jumping requires higher torque than reorientation, *SpaceHopper's* hip pitch motion has twice the torque of the hip yaw motion ( $i_{hr} = 2$ ).

#### D. Body Design

The triangular body shape, maximizes the ROM in the yaw DOF while minimizing the size of the body to save weight. The internal structure is 3D printed, which allows for high design flexibility. Thanks to this flexibility, we can tightly integrate the body internals and use vibration dampers to decouple the electronics stack from the load-bearing structure.

#### E. Materials

Most of the structural parts in the legs and the main triangles of the body are manufactured out of aerospace aluminum 7075, which has a good strength-to-weight ratio and is commonly used in space applications [25]. The shin and the body's internal electronics structure consists of 3D printable carbon-reinforced composites, which strikes a good tradeoff between the cost of manufacturing and structural robustness. The gears and shafts are manufactured from hardened steel (16MnCr5 and 30CrNiMo8, respectively) to endure the high surface pressures and bending forces. Finally, the side structure is water-jet cut out of 2 mm thick, layered carbon fiber for high tensile strength and low weight.

#### F. Electrical System

In the center of the electronics stack is a 7S1P lithium-ion battery pack managed by a Tiny BMS s516 v2.1 battery management system [26]. We use Lithium-ion cells because of their high power-to-weight ratio. A custom-built power distribution board (PDB) distributes the power from the battery to the motors and the onboard processing unit (Nvidia Jetson Nano). An STM32 microcontroller [27] connects to an internal temperature sensor, the BMS, and the Nvidia Jetson Nano. The STM32 chip shuts down the system in case of overheating or other failures, which adds to the system's robustness. We use the Nvidia Jetson Nano because it allows for efficient inference of the control policy via its GPU cores. We use three Maxon EPOS4 Compact 24/5 EtherCAT 3-axes motor controllers [28]. They are compact and support regenerative braking, which increases efficiency and prolongs mission duration. The robot also has three time-of-flight laser range sensors (LRFs) [29], each with a maximum range of 4 m, delivering height information.

### III. CONTROL SYSTEM OVERVIEW

#### A. Control Architecture

The control framework of *SpaceHopper* uses shared memory buffers that communicate with the peripherals via the Robot Operating System (ROS) [30]. The control framework comprises three parts: a low-level controller, a high-level controller, and a state estimator. The state estimator receives joint states, motion capture data, and laser-ranging sensor information and uses it to estimate the height and attitude of

the main body (see Fig. 3). The high-level controller processes the data received by the state estimator and computes desired joint positions. These desired joint positions are sent to the low-level controller, which maps them to desired motor positions and sends them to the motor controllers through an EtherCAT network.

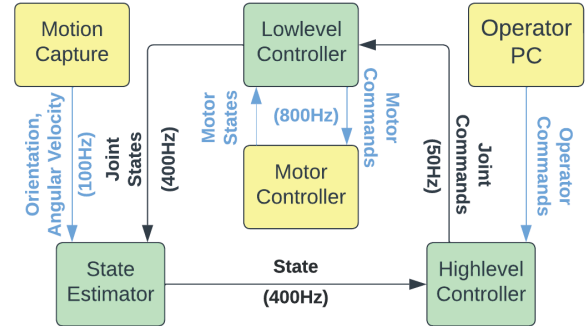


Fig. 3: Dataflow visualization inside *SpaceHopper* during reorientation test. Green boxes show software modules running on the onboard computer, and yellow boxes show peripheral modules.

#### B. Locomotion Controller

Developing a robust controller for low-gravity jumping locomotion is challenging. Due to the extended flight phases, small torques on the body during jump-off compound to large changes in orientation. For a safe landing, controlling the robot's attitude during the flight phase is paramount. However, reorientation using the legs is difficult. The kinematics of a free-floating space manipulator are nonholonomic, and dynamic singularities can significantly restrict the controllable workspace [31]. Model-based control methods exist that compute approximately optimal trajectories using non-linear optimization. However, they are computationally slow and do not scale well to multiple legs [32]. Other methods avoid the dynamic singularities by applying small circular motions with the leg [33]. However, these solutions result in prolonged reorientation times. On the other hand, model-free Deep Reinforcement Learning (DRL) has already been used successfully for free-floating attitude control of the robot *SpaceBok* [18]. Thus, we build upon this previous result and use a modified version of the proximal policy optimization (PPO) algorithm [34], [35] to train locomotion policies for *SpaceHopper*. The resulting locomotion policies are neural networks that take observations as inputs and output desired joint positions. The policy's architecture is a Multi-Layer Perceptron (MLP) with three hidden layers of size [512, 256, 128].

Typically, DRL-based locomotion controllers are trained in simulation and then transferred to the real world using sim-to-real techniques [36]. We train the policies in the IsaacGym simulation environment [37] for fast training, thanks to its GPU acceleration. To transfer the policies to the real robot, we employ domain randomization [38] of the robot's link masses, the center of mass, DOF frictions, and P&D-Gains.

| Policy                      | Observations  |
|-----------------------------|---|
| Attitude Control Simulation | $\mathbf{q}, \dot{\mathbf{q}}, \mathbf{q}_b, \boldsymbol{\omega}_b, \mathbf{a}_{t-1}$   |
| Attitude Control Gimbal     | $\mathbf{q}, \mathbf{q}_b, \boldsymbol{\omega}_b, \mathbf{a}_{t-1}$   |
| Jumping Simulation          | $\mathbf{q}, \dot{\mathbf{q}}, \mathbf{r}_b, \mathbf{r}_b^*, \mathbf{v}_b, \mathbf{q}_b, \boldsymbol{\omega}_b, \mathbf{a}_{t-1}$ |

|                 |   |                 |                              |
|-----------------|---|-----------------|------------------------------|
| orientation_3d: | $\ \text{rot\_vec}(\mathbf{q}_b)\ _2$   | orientation_2d: | $7 - ( \theta  +  \varphi )$ |
| action_rate:    | $\ \mathbf{a}_t - \mathbf{a}_{t-1}\ _2$   | torques:        | $\ \boldsymbol{\tau}\ _2^2$  |
| dof_vel:        | $\ \dot{\mathbf{q}}\ _2$  | dof_acc:        | $\ \ddot{\mathbf{q}}\ _2$    |
| collision:      | $\sum_j \mathbb{1}(\ \mathbf{f}_j\ _2 > 0.2)$   | height:         | $ r_{b,3} $                  |
| pos_cmd:        | $1 - \ \mathbf{r}_b - \mathbf{r}_b^*\ _2 / \ \mathbf{r}_b^*\ _2$                              |                 |                              |
| dof_limits:     | $\ \text{ReLU}(\mathbf{q}_l - \mathbf{q})\ _1 + \ \text{ReLU}(\mathbf{q} - \mathbf{q}_u)\ _1$ |                 |                              |

|                       |                  |   |                                    |
|-----------------------|------------------|---|------------------------------------|
| $\mathbf{q}$ :        | Joint Positions  | $\dot{\mathbf{q}}, \ddot{\mathbf{q}}$ : | Joint Velocities and Accelerations |
| $\mathbf{r}_b$ :      | Body Position    | $\mathbf{v}_b$ :                        | Body Velocity                      |
| $\mathbf{q}_b$ :      | Body Quaternions | $\boldsymbol{\omega}_b$ :               | Body Angular Velocity              |
| $\mathbf{a}_t$ :      | Actions          | $\mathbf{r}_b^*$ :                      | Commanded Body Position            |
| $\mathbf{q}_l$ :      | Lower DOF Limit  | $\mathbf{q}_u$ :                        | Upper DOF Limit                    |
| $\boldsymbol{\tau}$ : | Joint Torques    | $\mathbf{f}_j$ :                        | Contact Forces of Body $j$         |

TABLE II: The policy observations and reward components.

#### IV. ATTITUDE CONTROL

Since low gravity attitude control is a critical capability, we investigate it separately. In a first step, we train a policy using DRL to reorient while freely floating in zero gravity. We then validate the reorientation capability on the real system. Creating an accurate low gravity reorientation test in Earth's gravity for a system with 3DOF per leg is challenging. To this end, we developed a custom gimbal test rig that offloads the gravity from the body while allowing it to rotate freely. We also train a DRL policy to reorient in this test setup. While this gravity offload system substantially alters the robot's dynamics, it allows us to test the mechanical, electrical and control systems during attitude control maneuvers. Moreover, it helps us assess the sim to real performance.

##### A. Simulation Study

1) *Experiment Setup*: Since the robot effectively experiences weightlessness between jump-off and landing, we simulate the robot floating in zero gravity. The robot starts in a random initial orientation and then reorients to an upright landing configuration.

2) *Controller*: We use a DRL policy. All policy observations and reward components are listed in Table II. The total reward used during training is:

$$r = c_1 \cdot r_{\text{orientation\_3d}} + c_2 \cdot r_{\text{action\_rate}} + c_3 \cdot r_{\text{torques}} + c_4 \cdot r_{\text{dof\_limits}} \quad (2)$$

Where  $(c_1, c_2, c_3, c_4) = (-1, -0.04, -0.15, -3)$ . Here,  $r_{\text{orientation\_3d}}$  penalizes non upright attitudes.  $r_{\text{action\_rate}}$  prevents high-frequency oscillations in the joints,  $r_{\text{torques}}$  encourages the policy to find a low-torque solution and  $r_{\text{dof\_limits}}$  penalizes joint positions close to the ROM limit.

3) *Results*: After training, the robot can reorient itself to an upright orientation precisely and rapidly. Fig. 4 shows the robot changing its attitude from a random initial orientation to upright. The policy reaches an upright orientation within 1 s (see left panel of Fig. 5). Moreover, the attitude control is robust, with a mean smallest angle orientation error of 0.501 deg for twenty random initial orientations, as seen in the right panel of Fig. 5. Overall, considering that a jump with a distance of 6 m in Ceres gravity (0.029 g) takes around 8 s (see Fig. 8), the reorientation is fast enough to reach an upright landing configuration.



Fig. 4: *SpaceHopper* reorienting itself to the upright orientation, which is displayed as a black line.

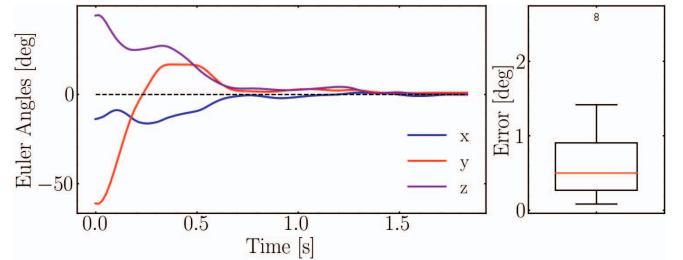


Fig. 5: *SpaceHopper*'s body orientation (Euler angles  $x, y, z$ ) in simulation during reorientation from random initial attitude to upright (left panel), and a box plot of final orientation errors for twenty random initial orientations (right panel).

##### B. Testing on Hardware

1) *Experiment Setup*: We test the capabilities of *SpaceHopper* using a custom gimbal, allowing the robot to rotate freely in two degrees of freedom. The inner ring can rotate with the angle  $\varphi$  and the outer ring with angle  $\theta$  (see Fig. 6). The robot starts in a random initial configuration of  $\varphi, \theta$  and has to reach an upright attitude. We track the body orientation and angular velocity using a Vicon motion capture system.

2) *Controller*: We use a DRL policy. The total reward is:

$$r = c_1 \cdot r_{\text{orientation\_2d}} + c_2 \cdot r_{\text{action\_rate}} + c_3 \cdot r_{\text{torques}} + c_4 \cdot r_{\text{dof\_limits}} + c_5 \cdot r_{\text{dof\_acc}} + c_6 \cdot r_{\text{dof\_vel}} \quad (3)$$

Where  $(c_1, c_2, c_3, c_4, c_5, c_6) = (0.15, -0.06, -0.01, -1, -4 \cdot 10^{-6}, -0.01)$ .  $r_{\text{orientation\_2d}}$  encourages the robot to be upright. We use  $r_{\text{dof\_vel}}$  and  $r_{\text{dof\_acc}}$  to avoid sim-to-real problems. The environments terminate when the robot collides with itself or with a part of the gimbal.

3) *Results*: We analyze the performance of *SpaceHopper*'s reorientation capability by releasing the robot at twenty random initial configurations of the gimbal angles  $\varphi$  and  $\theta$ . *SpaceHopper* reaches an upright attitude from all configurations. The left panel of Fig. 7 shows the change of attitude for one random initial configuration. It takes the robot 5 s

to reach the upright orientation. The mean final orientation error is 9.7deg over twenty random initial configurations (see right panel of Fig. 7). Despite the significant dynamics mismatches, which we further elaborate on in the Discussion VI-C, the promising results of this experiment serve as a positive indicator for *SpaceHopper's* capabilities in future more realistic micro-gravity experiments.

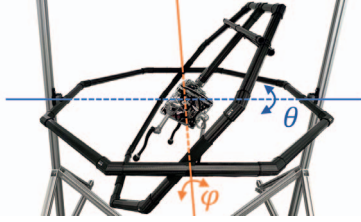


Fig. 6: *SpaceHopper* mounted in the gimbal test stand. The inner ring angle is  $\varphi$  in orange, and the outer ring angle  $\theta$  in blue.

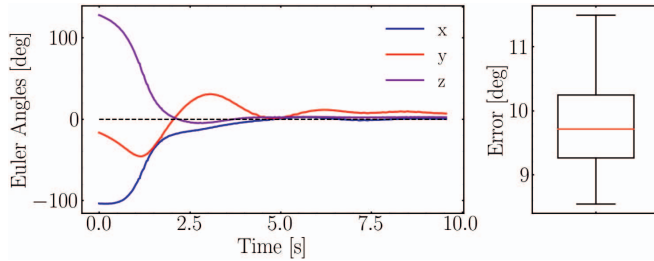


Fig. 7: *SpaceHopper's* body orientation (Euler angles  $x, y, z$ ) in the gimbal during reorientation from random initial attitude to upright (left panel), and a box plot of final orientation errors for twenty random initial orientations (right panel).

## V. JUMPING LOCOMOTION

Demonstrating controlled low-gravity jumping locomotion is the overarching goal of *SpaceHopper*. Integration of jumping, attitude control, and landing is necessary for successful jumping locomotion. We first demonstrate controlled jumping locomotion in a simulation of Ceres (0.029 g). Since validation of this controller on the real hardware in Earth's gravity requires extensive testing facilities [39], we restrict ourselves to showing the real robot's jumping capability only for vertical jumping in a counterweight setup.

### A. Simulation Study

1) *Experiment Setup*: The simulation environment consists of a flat ground plane with the gravity of Ceres. The robot starts on the ground and is commanded to jump to a goal body position  $\mathbf{r}_b^*$  with radius 6 m around the initial position.

2) *Controller*: We propose an end-to-end approach for jumping, reorientation, and landing. A DRL policy trained simultaneously for all three tasks is not limited by heuristic transitions between controllers. The total reward is:

$$r = c_1 \cdot r_{\text{pos\_cmd}} + c_2 \cdot r_{\text{orientation\_3d}} + c_3 \cdot r_{\text{action\_rate}} + c_4 \cdot r_{\text{torques}} + c_5 \cdot r_{\text{collision}} + c_6 \cdot r_{\text{height}} \quad (4)$$

The reward component weights have the following values:  $(c_1, c_2, c_3, c_4, c_5, c_6) = (1.5, -0.4, -0.05, -0.4, -15, 0.19)$ .

Here,  $r_{\text{pos\_cmd}}$  encourages reducing the distance to the goal position,  $r_{\text{collision}}$  penalizes collisions with the ground,  $r_{\text{orientation\_3d}}$  enforces an upright body attitude and  $r_{\text{height}}$  incentivizes the exploration of solutions involving jumping.

3) *Results*: As shown in Fig. 9, this policy can handle jumping, reorientation, and landing end-to-end. The robot reaches a maximum height of 2.55 m, stabilizes its attitude, and manages to land on its feet within 9 s. For 100 random position commands at a distance of 6 m, the policy achieves an average position error of 0.316 m. The worst case position error is 0.843 m. Most importantly, the robot's body, thighs, and shins never collide with the ground. Figure 8 confirms our assumption that jumping requires higher pitch than yaw torques.

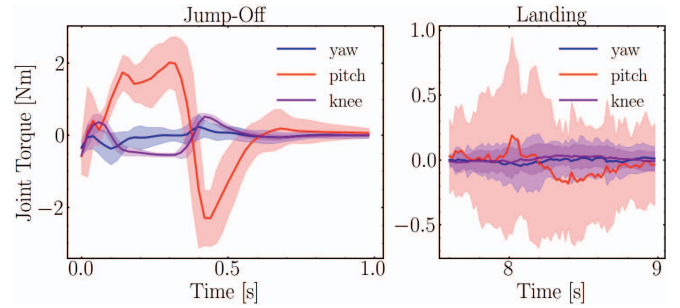


Fig. 8: The yaw, pitch, and knee joint torques during jumps to 100 random positions in simulation. The thick lines are the mean and the shaded areas are the standard deviations. We average the joint torques from different legs. The robots leave the ground around the 0.4 s mark.

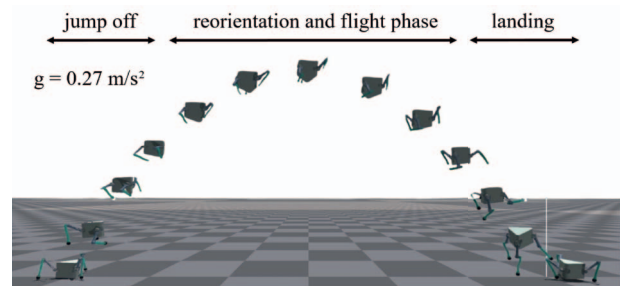


Fig. 9: *SpaceHopper* in Ceres gravity, jumping to a 6 m far commanded position (white line).

### B. Testing on Hardware

1) *Experiment Setup*: *SpaceHopper's* drivetrain is not powerful enough to allow jumping on Earth. We use a simple pulley-counterweight setup that attaches to the top of the robot's main body to offload the gravity from the body. Using a counterweight of 3.9 kg, we simulate a gravity of  $2.5 \text{ m/s}^2$  on the body's vertical axis. We chose this value so that the robot does not exceed the maximum height of the test stand while jumping.

2) *Controller*: Due to the limitations of the test setup (see Subsec. VI-C), we only validate the jumping capability of the mechanical and electrical systems. Accordingly, we omit a full DRL-based control approach and use a simple controller that tracks a hand-crafted joint-space trajectory

with the height estimation from the laser-ranging sensors. The trajectory consists of a leg extension to initiate the jump and a contraction during the flight phase to prepare for the next jump. This trajectory is the same for all three legs and gets triggered when the robot is below a certain height.

3) *Results*: Fig. 10 shows *SpaceHopper* jumping to a height of 1.2m, landing and immediately jumping off again. In this continuous jumping mode, *SpaceHopper* achieves at least four and up to 15 consecutive jumps before a human has to intervene. The failure mode is a result of the open-loop controller design. Over time, minor disturbances accumulate, which cause the robot and rope of the counterweight to swing. After a human stabilizes the swinging motion, the robot can resume jumping for up to 80 min.

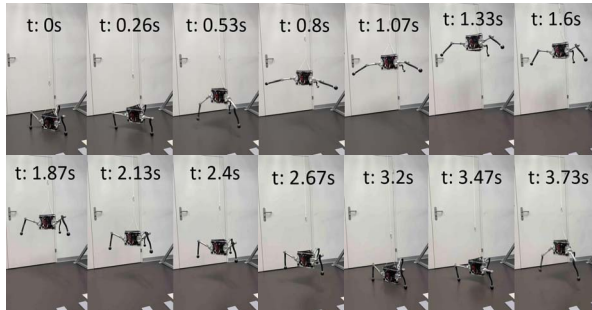


Fig. 10: *SpaceHopper* jumping continuously with a counterweight of 3.9kg being attached to the top of the body over a pulley system.

## VI. DISCUSSION

### A. Space-qualification of components

As mentioned in various sections, we designed *SpaceHopper* towards operation in space by using appropriate materials (aluminum 7075 and carbon fiber), space-qualifiable motors, a differential drive design with crown gears, and lithium-ion batteries. However, the system is not fully space-graded yet. For example, the electronics are not radiation-shielded, and adequate thermal insulation is vital to survive in space. A first feasibility study regarding a thermal assessment for a lunar mission has already been done [40]. Furthermore, the system relies on an external motion capture system during testing. Therefore, an internal state estimation is needed that can estimate the robot's position and velocity relative to an inertial frame. The current sensor layout with three LRFs is inadequate to perform this task. Fusing an IMU (accelerometer and gyroscope), a camera, a LRF, and performing Range-Visual-Inertial Odometry (range-VIO) could be a possible solution [41], [42].

### B. Safety and Robustness of DRL for Space Exploration

The cost of space missions is significant, making safe and robust controllers a critical component of the system. Standard model-free RL has no theoretical safety guarantees. However, as seen in our experiments and prior work [43], DRL controllers, in practice, display excellent robustness even in highly complex environments. Thanks to modern training techniques such as domain randomization, resulting policies

can generalize to unseen target environments. Furthermore, safe RL is an active field of research [44]. In the future, such findings may be integrated into control approaches for additional theoretical guarantees.

### C. Simulating low-gravity scenarios

While the gimbal and the counterweight setup allow us to test the reorientation and jumping of *SpaceHopper* at Earth's gravity, they lack in accurately representing the robot's dynamics in low-gravity environments. The Earth's gravity still acts on the legs for both gravity offload systems. The inertia added through the gimbal test setup changes the system dynamics further. Additionally, imperfections in balancing the gimbal lead to constant torques acting on the robot, which increases the attitude error and leads to longer stabilization times. Testing only vertical jumping also falls short of validating pronking locomotion on real hardware. Moreover, the fast jumping dynamics cause slack in the counterweight rope, which introduces additional inaccuracies and makes sim to real transfer of DRL policies difficult. In future work, we plan a parabolic flight testing campaign that will overcome these limitations and further allow us to test the main locomotion goal and increase *SpaceHopper's* TRL.

### D. Adaptation to Irregular Terrain

The irregular terrain of asteroids poses significant challenges for non-articulated hopping robots like MINERVA [7]. Unpredictable bouncing after landing severely limits precise locomotion. We hypothesize that a limbed system can adapt its legs for a soft landing, which allows for more controlled interactions with the surface. Future work will need to validate our control approach in this setting, especially on soft and granular media that may require different foot designs.

## VII. CONCLUSION AND FUTURE WORK

In conclusion, *SpaceHopper* is a research platform to investigate highly dynamic legged locomotion for exploring low-gravity celestial bodies, such as asteroids and moons. Unique features of the design, such as three legs, lightweight construction, small size, and a differential drive train, make the system well-adapted to jumping locomotion in low-gravity. In a zero gravity simulation, *SpaceHopper* can change its attitude in 1s. In a simulation of Ceres' gravity, it can also jump to commanded positions at a distance of 6m with an average position error of 0.316m. The real robot reaches an upright attitude within 5s with a mean orientation error of 9.7deg inside a gimbal test setup. The mechanical and electrical systems allow repeated vertical jumps in a counterweight setup. Due to the low-gravity optimized design of *SpaceHopper*, creating a fitting test on Earth is a significant challenge. A natural follow-up of this work is to test *SpaceHoppers'* capabilities in the microgravity environment of a parabolic flight. Further work is also necessary to show jumping on granular media, as often found on asteroids and moons. Finally, the state estimation must be extended and validated to work in the target environment.

## REFERENCES

- [1] M. Elvis, "Let's mine asteroids — for science and profit," *Nature*, vol. 485, no. 7400, pp. 549–549, May 2012.
- [2] A. M. Yazici and S. Darici, "The new opportunities in space economy," *İnsan ve Toplum Bilimleri Araştırmaları Dergisi*, vol. 8, no. 4, pp. 3252–3271, 2019.
- [3] J. Green, "Perseverance rover and its search for life on mars," *Communications of the Byurakan Astrophysical Observatory*, pp. 464–469, 01 2021.
- [4] H.-J. Sedlmayr, *et al.*, "Mmx - development of a rover locomotion system for phobos," 03 2020, pp. 1–10.
- [5] H. Kolvenbach, M. Breitenstein, C. Gehring, and M. Hutter, "Scalability analysis of legged robots for space exploration," in *Unlocking imagination, fostering innovation and strengthening security: 68th International Astronautical Congress (IAC 2017)*, vol. 16. Curran, 2018, pp. 10 399–10 413.
- [6] T.-M. Ho, *et al.*, "Mascot—the mobile asteroid surface scout onboard the Hayabusa2 mission," *Space Science Reviews*, vol. 208, no. 1, pp. 339–374, 2017.
- [7] H. Yabuta, "Arrival, touchdown and sequel to the voyage of Hayabusa2," *Nature Astronomy*, vol. 3, no. 4, pp. 287–289, Apr. 2019.
- [8] A. Parness, N. Abcouwer, C. Fuller, N. Wiltsie, J. Nash, and B. Kennedy, "Lemur 3: A limbed climbing robot for extreme terrain mobility in space," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5467–5473.
- [9] T. G. Chen, *et al.*, "Reachbot: A small robot with exceptional reach for rough terrain," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 4517–4523.
- [10] M. Hutter, *et al.*, "Anymal - toward legged robots for harsh environments," *Advanced Robotics*, vol. 31, no. 17, pp. 918–931, 2017. [Online]. Available: <https://doi.org/10.1080/01691864.2017.1378591>
- [11] A. Bouman, *et al.*, "Autonomous spot: Long-range autonomous exploration of extreme environments with legged locomotion," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 2518–2525.
- [12] H. Kolvenbach, "Quadrupedal robots for planetary exploration," Ph.D. dissertation, ETH Zurich, 2021.
- [13] A. Agha, *et al.*, "Nebula: Quest for robotic autonomy in challenging environments; TEAM costar at the DARPA subterranean challenge," *CoRR*, vol. abs/2103.11470, 2021. [Online]. Available: <https://arxiv.org/abs/2103.11470>
- [14] P. Arm, *et al.*, "Spacebok: A dynamic legged robot for space exploration," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6288–6294.
- [15] H. Kolvenbach, *et al.*, "Traversing steep and granular martian analog slopes with a dynamic quadrupedal robot," in *Field Robotics*, 2022.
- [16] H. Kolvenbach, C. D. Bellicoso, F. Jenelten, L. Wellhausen, and M. Hutter, "Efficient gait selection for quadrupedal robots on the moon and mars," in *14th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS 2018)*. ESA Conference Bureau, 2018.
- [17] H. Kolvenbach, E. Hampp, P. Barton, R. Zenkl, and M. Hutter, "Towards jumping locomotion for quadruped robots on the moon," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 5459–5466.
- [18] N. Rudin, H. Kolvenbach, V. Tsounis, and M. Hutter, "Cat-like jumping and landing of legged robots in low-gravity using deep reinforcement learning," *CoRR*, vol. abs/2106.09357, 2021. [Online]. Available: <https://arxiv.org/abs/2106.09357>
- [19] "Cubesat - deployer standards," <https://www.eoportal.org/other-space-activities/cubesat-deployer-standards>, Last accessed: 05.09.2023.
- [20] "Ec 45 flat 42.8 mm, brushless, 50 watt, with hall sensors," <https://www.maxongroup.com/maxon/view/product/motor/ecmotor/ecflat/ecflat45/651610>, Last accessed: 07.09.2023.
- [21] "EcX torque 22 m 22 mm, brushless, with hall sensors," <https://www.maxongroup.com/maxon/view/product/motor/ecmotor/ECX/ECX22/ECX122M4ZF46C4IL1Y479A>, Last accessed: 07.09.2023.
- [22] U. Kissling and S. Beermann, "Face gears: Geometry and strength," *Gear Technol*, vol. 1, no. 2, pp. 54–61, 2007.
- [23] "Spot. boston dynamics," <https://www.bostondynamics.com/products/spot>, Last accessed: 08.03.2023.
- [24] A. Abate, J. W. Hurst, and R. L. Hatton, "Mechanical antagonism in legged robots." in *Robotics: Science and Systems*, vol. 6. Ann Arbor, MI, 2016.
- [25] "NASA - Materials Data Handbook - Aluminum Alloy 7075," <https://ntrs.nasa.gov/citations/19720022809>, Last accessed: 15.09.2023.
- [26] "Battery management system (bms) 30a," <https://enepaq.com/product/battery-management-system-bms-30a/>, Last accessed: 05.09.2023.
- [27] "Stm32g431kb," <https://www.st.com/en/microcontrollers-microprocessors/stm32g431kb.html>, Last accessed: 05.09.2023.
- [28] "Epos4 compact 24/5 ethercat 3-axes, digital positioning controller, 5 a per axis, 10 - 24 vdc," <https://www.maxongroup.com/maxon/view/product/control/Positionierung/684519>, Last accessed: 07.09.2023.
- [29] "Time-of-flight (tof) ranging sensor with advanced multi-zone and multi-object detection," <https://www.st.com/en/imaging-and-photonics-solutions/vl531l.html>, Last accessed: 07.09.2023.
- [30] M. Quigley, *et al.*, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.
- [31] E. G. Papadopoulos, *Nonholonomic Behavior in Free-floating Space Manipulators and its Utilization*. Boston, MA: Springer US, 1993, pp. 423–445. [Online]. Available: [https://doi.org/10.1007/978-1-4615-3176-0\\_11](https://doi.org/10.1007/978-1-4615-3176-0_11)
- [32] C. Fernandes, L. Gurvits, and Z. Li, "Attitude control of space platform/manipulator system using internal motion," in *Proceedings 1992 IEEE International Conference on Robotics and Automation*, 1992, pp. 893–898 vol.1.
- [33] Z. Vafa and S. Dubowsky, *On the Dynamics of Space Manipulators Using the Virtual Manipulator, with Applications to Path Planning*. Boston, MA: Springer US, 1993, pp. 45–76. [Online]. Available: [https://doi.org/10.1007/978-1-4615-3588-1\\_3](https://doi.org/10.1007/978-1-4615-3588-1_3)
- [34] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *CoRR*, vol. abs/1707.06347, 2017. [Online]. Available: <http://arxiv.org/abs/1707.06347>
- [35] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," *CoRR*, vol. abs/2109.11978, 2021. [Online]. Available: <https://arxiv.org/abs/2109.11978>
- [36] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: a survey," *CoRR*, vol. abs/2009.13303, 2020. [Online]. Available: <https://arxiv.org/abs/2009.13303>
- [37] V. Makoviychuk, *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *CoRR*, vol. abs/2108.10470, 2021. [Online]. Available: <https://arxiv.org/abs/2108.10470>
- [38] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," *CoRR*, vol. abs/1703.06907, 2017. [Online]. Available: <http://arxiv.org/abs/1703.06907>
- [39] O. Bekdash, *et al.*, "Development and evaluation of the active response gravity offload system as a lunar and martian eva simulation environment." 2020 International Conference on Environmental Systems, 2020.
- [40] M. Trentini, P. Arm, G. Valsecchi, H. Kolvenbach, and M. Hutter, "Concept study of a small-scale dynamic legged robot for lunar exploration," in *IAC-23: IAF SPACE EXPLORATION SYMPOSIUM. BCC B3: IAF, October 3 2023, paper code: IAC-23,A3,2B,7,x78250; Session: 2B. Moon Exploration – Part 2*. [Online]. Available: <https://iafastro.directory/iac/paper/id/78250/summary/>
- [41] J. Delaune, D. S. Bayard, and R. Brockers, "Range-visual-inertial odometry: Scale observability without excitation," *CoRR*, vol. abs/2103.15215, 2021. [Online]. Available: <https://arxiv.org/abs/2103.15215>
- [42] B. Balaram, *et al.*, "Mars helicopter technology demonstrator," in *2018 AIAA Atmospheric Flight Mechanics Conference*, 2018, p. 0023.
- [43] N. Rudin, D. Hoeller, M. Bjelonic, and M. Hutter, "Advanced skills by learning locomotion and local navigation end-to-end," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 2497–2503.
- [44] S. Gu, *et al.*, "A review of safe reinforcement learning: Methods, theory and application," 2023.