machines

MDPI

Article

The Concept of the Virtual Pose Instruction Plane (VPIP) for Controlling Rod-Driven Spherical Robots

Jasper Zevering [1,*], Joshua Braun [1], Martin Hesse [1], Kedus Mathewos [1], Dorit Borrmann [2], Anton Bredenbeck [3] and Andreas Nüchter [1]

[1] Computer Science XVII: Robotics, University of Wuerzburg, 97074 Wuerzburg, Germany; joshua.braun@stud-mail.uni-wuerzburg.de (J.B.); martin.hesse@uni-wuerzburg.de (M.H.); k.mathewos@jacobs-university.de (K.M.); andreas.nuechter@uni-wuerzburg.de (A.N.)
[2] Faculty of Electrical Engineering, Technical University of Applied Sciences Würzburg-Schweinfurt, 97421 Schweinfurt, Germany; dorit.borrmann@thws.de
[3] Biomorphic Intelligence Lab, Faculty of Aerospace Engineering, 2629 HS Delft, The Netherlands; a.bredenbeck@tudelft.nl
* Correspondence: jasper.zevering@uni-wuerzburg.de

Abstract

The exploration of lunar caves is a critical aspect of the space exploration program of the European Space Agency (ESA). To facilitate this mission, the DAEDALUS study investigated a novel spherical robot design in 2021. The proposed robot uses a unique telescopic linear rod mechanism to generate rotation and hence locomotion. This drive mechanism requires a dedicated control scheme to ensure both locomotion and simultaneously stabilization of the robot. The overall task of following a curved trajectory is also a problem that cannot be solved by simple algorithms. In this work, we introduce, calculate, and simulate a solution for these tasks, the Virtual Pose Instruction Plane (VPIP). The VPIP breaks the problem of multiple independent controllable rods down to two controllable parameters (roll and pitch of the plane), which control the linear motion velocity, balance and ultimately curvature motion of the robot. Initial simulations show that both speed and cornering can be controlled by the VPIP.

Keywords: control; spherical robot; telescopic linear-driven rotation

Check for updates

Academic Editors: Luca Bruzzone and Luigi Tagliavini

Received: 14 March 2026
Revised: 19 April 2026
Accepted: 22 April 2026
Published: 26 April 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Despite the large variety of shapes and locomotion systems of robots, only four types are used as a baseline for extraterrestrial ground exploration: robots having mobility based on wheels, tracks, legs, and combinations of them [1].

Through ESA's SysNova assessment scheme [2], a spherical robot for lunar cave exploration was developed: the "Descent And Exploration in Deep Autonomy of Lava Underground Structures (DAEDALUS)" [3]. Designed for such a lunar cave exploration, the shape of the robot [4], locomotion system, and all other aspects are built to suit the requirements for exploring lunar caves and the restrictions associated with the use of optical cameras and laser scanners. The robot (cf. Figure 1) has its spherical shape due to environmental reasons and mechanisms to account for the payload requirements.

It was from this project that the Telescopic Linear-Driven Rotation (TLDR) locomotion concept emerged, which we introduced in [5]. We stated a more dedicated controller for locomotion and balancing as future work, as this is a challenging control problem. In this work, we provide a solution to this control problem. The VPIP concept also applies to other

linear driver rotation formats, such as distributed telescopic linear actuated motion [6], polyhedral linear induced rotation [7], or linearly driven extendable shell wedges [8], thereby simplifying a multi-linear actuator system to a two-component control system. The mathematics we present is for symmetrically arranged telescopic linear actuators such as DAEDALUS, but the principle can be easily transferred to all linearly activated spherical robots.

Figure 1. Rendering of the DAEDALUS robot concept. The spherical robot holds 2 rings with eight rods each on both sides. Apart from locomotion those rods are also used for tasks such as heat radiation, stabilization, and climbing. The main structure in the center holds laser scanners, optical sensors, a battery, a board computer and other payloads.

2. Background

Telescopic Linear-Driven Locomotion uses protruding rods that extend from a spherical robot to set the system in motion. In general, there are two ways in which these rods can contribute to locomotion: by pushing on the ground behind the robot and by leveraging their weight in front. The latter requires the rods to have substantial weight to be effective, reducing the overall efficiency. Thus, it will be disregarded for the rest of this work. For a spherical robot of radius r, pushing is only possible with rods within a suitable range of angles to the ground. Figure 2 indicates the variables used in the following. To illustrate the necessity of the VPIP, we will briefly introduce the direct drive approach and the balancing approach.

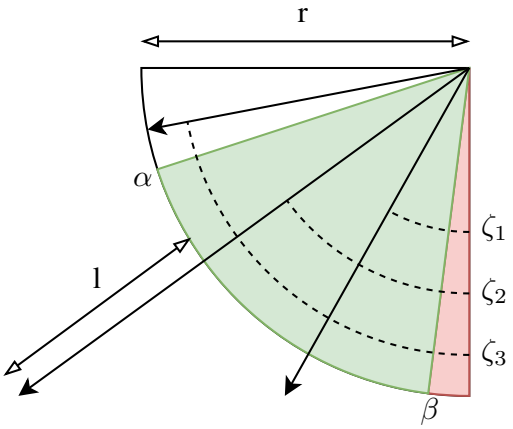

Figure 2. Three rods and their respective angle with respect to the ground ($\zeta_1, \zeta_2, \zeta_3$) as well as the other angles (α, β) describing the minimum and maximum angles during which pushing is possible. r represents the radius of the sphere and l the current length of the extended rod. The red area marks the non-extension area due to not reaching past β, and the green area marks the extension area between α and β.

2.1. Pushing

Let us consider the angle ζ_x of a rod x to be zero when the rod is perpendicular to the ground in the forward direction of the robot. The angle then increases counterclockwise until the wraparound at 2π. The rod can propel the robot forward by pushing as soon as the robot rolls over the rod and as long as the rod can reach the ground. Given the maximal extension length of the rods l_{max}, the suitable range is defined by the angles

$$\alpha < \min\left[\arccos\left(\frac{r}{r + l_{max}}\right), \frac{\pi}{2}\right], \quad \beta > 0\,. \tag{1}$$

At any angle between α and β, the length of the rod must be extended to

$$l_x = \frac{r}{\cos\zeta_x} - r\,, \tag{2}$$

to establish contact with the ground.

By taking the derivative of (2) and solving for the derivative of the angle of the rod with respect to the ground (which is equivalent to the rotational velocity of the sphere), it can be shown that rods with a constant extension velocity ($\dot{l}_x = const$) cannot sustain a constant rotational velocity of the sphere. For details, please refer to [5].

2.2. Balancing

Since the spherical robot can rotate freely in all directions, the lateral motion must be carefully controlled to keep the rods aligned with the direction of travel. Since the rods can only provide a constant extension and retraction velocity and do not have feedback on their extension length, the control takes the form of a two-point controller with hysteresis. We choose a hysteresis band of $\pm 3^\circ$ in which the controller maintains its current state. Outside this band, the rods extend on the side the robot tilts towards and retract on the opposing side. In our past work, this has proven sufficient to maintain an upright position [5].

2.3. Related Work

Control strategies for spherical robots are well established for the dominant drive classes. Pendulum- and inertial drive unit (IDU)-driven robots have benefited from decades of research on nonlinear dynamics, feedback linearisation, linear quadratic regulator (LQR), and model-predictive control, as comprehensively surveyed by Diouf et al. [9]. Recent studies continue to refine these methods, including omniwheel-platform control [10], closed-loop stabilization for eccentric-pendulum designs [11], and high-speed trajectory tracking [12]. In contrast, the control landscape for linearly driven spherical robots remains less developed. PodRacle [8] derives kinematic equations for shell-wedge actuation but employs an open-loop, inertial measurement unit (IMU)-referenced sequence of preprogrammed leg movements without closed-loop stabilization of the rolling direction. The Sea Urchin robot [6] similarly supports only semi-autonomous preprogrammed motion patterns and lacks a feedback controller. Mochibot [13] introduces an imaginary-contact-surface concept to compute leg lengths for trajectory following; however, the robot functions as a continuous crawler rather than a dynamically stabilized rolling sphere, resulting in a fundamentally different control problem. Most recently, Xu et al. [14] explicitly state that their polyhedral linear-driven robot supports only manual remote control and identify closed-loop control as an outstanding area for future research. In summary, although the mechanics of linear rod actuation have been investigated in several prototypes, no dedicated control strategy for simultaneous locomotion and lateral stabilization of a rolling TLDR robot has been published. This work addresses this specific gap.

3. Virtual Pose Instruction Plane (VPIP)

We now introduce a more detailed algorithm. Rather than combining the separately evaluated instructions of locomotion and balancing, we want to evaluate the state of the poles based on the need for both of them. We therefore control the poles by absolute length. This brings us from the evaluation in 2D, as was done separately for balance and locomotion, to an evaluation in three dimensions. We therefore introduce the Virtual Pose Instruction Plane (VPIP). The VPIP is a theoretical plane that determines the lengths of poles. It is fixed to the robot at the same point where the robot touches the ground. The poles extend and retract to the length needed to reach the VPIP. Two angles describe the relevant orientation of the plane: ϕ_{VPIP} and θ_{VPIP}. They have the same direction as the internal ϕ and θ of the robot. Figure 3 illustrates the VPIP.

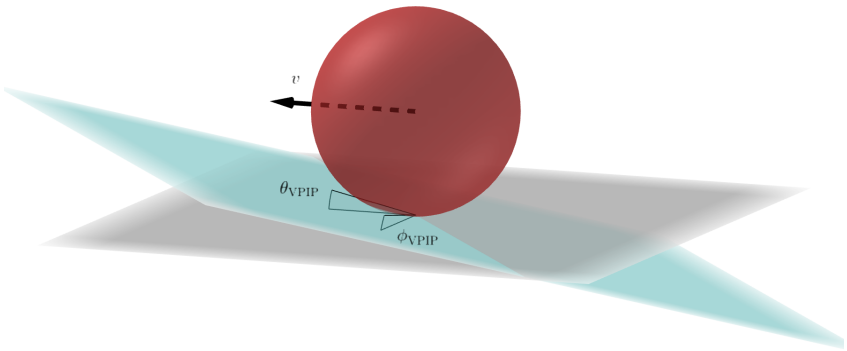

Figure 3. Visualization of VPIP (blue) and the describing angles ϕ_{VPIP} and θ_{VPIP}. The robot (red sphere) rolls into the v direction. The grey plane shows the actual ground.

We define the y-axis in rolling direction, the z-axis perpendicular to the flat ground passing the midpoint of the robot, and the x-axis to complete the right-hand system. Note that this formulation assumes a flat ground surface as a reference. Through the closed-loop nature of the VPIP, moderate terrain variations are handled implicitly via reactive correction, though at the cost of the disturbance having to manifest before it can be compensated. We discuss a proactive extension using terrain estimation in the conclusion, called VPIM. Let the contact point of the robot $\mathbf{r}_0$ to the ground be always at $(0,0,0)$, and $\mathbf{n}_{\text{VPIP}}$ be the normal vector on the plane, and $\mathbf{r}$ any point on the plane; then, the VPIP Π_{VPIP} is defined by

$$\Pi_{\text{VPIP}} : \mathbf{n}_{\text{VPIP}} \cdot (\mathbf{r} - \mathbf{r}_0) = \frac{\mathbf{n}_{\text{VPIP}}}{||\mathbf{n}_{\text{VPIP}}||} \cdot \mathbf{r} = 0 \tag{3}$$

$$\begin{bmatrix} \tan(\phi_{\text{VPIP}}) \\ \tan(\theta_{\text{VPIP}}) \\ 1 \end{bmatrix} \cdot \frac{1}{\sqrt{\tan(\phi_{\text{VPIP}})^2 + \tan(\theta_{\text{VPIP}})^2 + 1}} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = 0$$

$$\Leftrightarrow \tan(\phi_{\text{VPIP}}) \cdot x + \tan(\theta_{\text{VPIP}}) \cdot y + z = 0\,. \tag{4}$$

The idea is to adjust the pole length such that it does not touch the ground but the VPIP. Therefore, if we tilt the VPIP, as shown in Figure 3, the poles on the left front side (seen in the direction of velocity) retract and the poles on the rear right side extend. For the other directions (front right, rear left), we cannot state general actions for every pole, as it depends on their ζ. For every linear-driven rotation spherical robot, the steps for setting up are as follows:

- Find the equation for each linear pushing rod/pole with respect to the bottom point of the sphere.
- Set this equation equal to Equation (4).
- Solve for the length l.

The control itself then takes place as follows:

- The two angles of the VPIP are controlled using a controller of choice (we use PID) based on the speed and desired lateral tilt angle (perspective curve angle) as input.
- The two output angles are used in the intersection equation which was solved for l.
- The rods are then attempted to be extended to the length l (this does not necessarily have to be achieved).

This process is then repeated at the desired control frequency.

4. Implementation for TLDR Robots

We will now derive the equations for the Setup of the type of TLDR robot DAEDALUS is based on, so there will be two parallel Discs with eight rods each (refer to Figure 1 for visualization). To find the needed length of a pole to touch the VPIP, we first need to find the equation of a line of a pole in this representation. Therefore, we define the point $\mathbf{p}_m$ in the middle of the side disc and the point $\mathbf{p}_t$ as the tip of the fully retracted pole. Figure 4 shows the two points in the three-dimensional view as well as the two two-dimensional views for easier calculation.

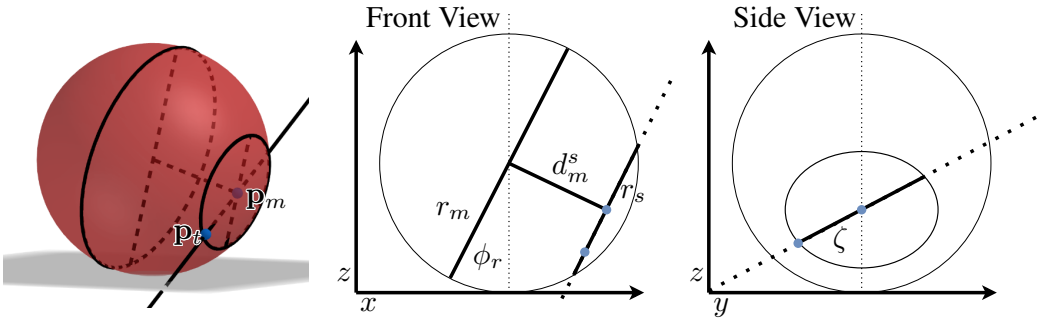

Figure 4. Illustration of pole line for VPIP evaluation. The blue dots ($\mathbf{p}_t$ and $\mathbf{p}_m$) are used to calculate the line.

We derive from this figure that

$$\mathbf{p}_m = \begin{bmatrix} \cos(\phi_r) \cdot d_m^s \\ 0 \\ r_m - \sin(\phi_r) \cdot d_m^s \end{bmatrix}, \tag{5}$$

and with that, we can derive $\mathbf{p}_t$ as a combination of the point $\mathbf{p}_m$ and the vector $\mathbf{d}_{mt}$ from $\mathbf{p}_m$ to $\mathbf{p}_t$:

$$\mathbf{p}_t = \mathbf{p}_m + \mathbf{d}_{mt} = \mathbf{p}_m + \begin{bmatrix} -\cos(\zeta) \cdot r_s \cdot \sin(\phi_r) \\ -\sin(\zeta) \cdot r_s \\ -\cos(\zeta) \cdot r_s \cdot \cos(\phi_r) \end{bmatrix}. \tag{6}$$

d_m^s is not variable in the case of a sphere but defined by

$$d_m^s = \sin\left(\arccos\left(\frac{r_s}{r_m}\right)\right) \cdot r_m \,. \tag{7}$$

For both sides of the pole discs, there needs to be an adaption. We do this by mirroring $\mathbf{p}_m$ at the main disc plane. Therefore, we define $\mathbf{p}_{ml}$ as the $\mathbf{p}_m$ for the middle of the left disc and $\mathbf{p}_{mr}$ of the right (seen in the rolling direction). This leads to

$$\mathbf{p}_{ml} = \begin{bmatrix} -\cos(\phi_r) \cdot d_m^s \\ 0 \\ r_m - \cdot \sin(\phi_r) \cdot d_m^s \end{bmatrix}. \tag{8}$$

To work out an overall solution, we assign the side factor $s_f = -1$ to the left side and $s_f = 1$ to the right side and write the general $\mathbf{p}_m$ as

$$\mathbf{p}_m = \begin{bmatrix} s_f \cdot \cos(\phi_r) \cdot d_m^s \\ 0 \\ r_m - s_f \cdot \sin(\phi_r) \cdot d_m^s \end{bmatrix}. \tag{9}$$

Now, we calculate the line of a specific pole, which goes through the point $\mathbf{p}_m$ with the direction $\mathbf{d}_{ms}$. Let ζ be the angle of the pole, ϕ_r the actual roll angle of the robot, r_s the radius of the disc inside the spherical robot wherein the poles lie, r_m the radius of the sphere, d_m^s the distance between the midpoint of the sphere and the midpoint side disc of the poles and s_f the side factor; then, the line of this pole $L_p(\zeta, \phi_r, s_f)$ is defined by

$$\begin{aligned} L_p(\zeta, \phi_r, s_f) &= \left\{ \begin{bmatrix} x \\ y \\ z \end{bmatrix} \middle| \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} p_m^x \\ p_m^y \\ p_m^z \end{bmatrix} + \lambda \begin{bmatrix} d_{mt}^x \\ d_{mt}^y \\ d_{mt}^z \end{bmatrix} | \lambda \in \mathbb{R} \right\} \\ &= \left\{ \begin{bmatrix} x \\ y \\ z \end{bmatrix} \middle| \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} s_f \cdot \cos(\phi_r) \cdot d_m^s \\ 0 \\ r_m - s_f \cdot \sin(\phi_r) \cdot d_m^s \end{bmatrix} \right. \\ &\quad \left. + \lambda \begin{bmatrix} -\cos(\zeta) \cdot r_s \cdot \sin(\phi_r) \\ -\sin(\zeta) \cdot r_s \\ -\cos(\zeta) \cdot r_s \cdot \cos(\phi_r) \end{bmatrix} | \lambda \in \mathbb{R} \right\}. \end{aligned} \tag{10}$$

We use λ as an indication of the extended length as we have the midpoint of the pole disc as a fixed point for the line and then λ multiplied by the gradient of the pole. So, if we normalize the vector of the gradient, λ represents the actual length of the pole but including r_s. Therefore, let us write λ as $r_s + l$. Mathematically, this leads to the possibility of l becoming negative or l larger than $l_{\max}$. In these cases, we need to incorporate the practical limitations of the pole. Negative extensions stop at full retraction, whereas larger extensions than possible are just full extensions. For this representation of λ, the vector that it scales needs to be normalized. For the normalization we divide $\mathbf{d}_{mt}$ by its length $||\mathbf{d}_{mt}||$. Per definition $||\mathbf{d}_{mt}|| = r_s$ as $\mathbf{d}_{mt}$ is the vector from $\mathbf{p}_m$ to $\mathbf{p}_t$. This leads to the definition of the line

$$L_p(\zeta, \phi_r, s_f) = \left\{ \begin{bmatrix} x \\ y \\ z \end{bmatrix} \middle| \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} p_m^x \\ p_m^y \\ p_m^z \end{bmatrix} + \lambda \frac{\mathbf{d}_{mt}}{||\mathbf{d}_{mt}||} | \lambda \in \mathbb{R} \right\}$$

$$= \left\{ \begin{bmatrix} x \\ y \\ z \end{bmatrix} \middle| \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} s_f \cdot \cos(\phi_r) \cdot d_m^s \\ 0 \\ r_m - s_f \cdot \sin(\phi_r) \cdot d_m^s \end{bmatrix} \right.$$

$$\left. + \frac{r_s + l}{r_s} \cdot \begin{bmatrix} -\cos(\zeta) \cdot r_s \cdot \sin(\phi_r) \\ -\sin(\zeta) \cdot r_s \\ -\cos(\zeta) \cdot r_s \cdot \cos(\phi_r) \end{bmatrix} | l \in \mathbb{R} \right\} \tag{11}$$

$$= \left\{ \begin{bmatrix} x \\ y \\ z \end{bmatrix} \middle| \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} s_f \cdot \cos(\phi_r) \cdot d_m^s \\ 0 \\ r_m - s_f \cdot \sin(\phi_r) \cdot d_m^s \end{bmatrix} \right.$$

$$\left. + (r_s + l) \cdot \begin{bmatrix} -\cos(\zeta) \cdot \sin(\phi_r) \\ -\sin(\zeta) \\ -\cos(\zeta) \cdot \cos(\phi_r) \end{bmatrix} | l \in \mathbb{R} \right\}. \tag{12}$$

Note that we set $l \in \mathbb{R}$ despite the physical limitation of $0 \leq l \leq l_{\max}$, as there may exist an intersection of the line of the rod with the VPIP, but for an $l > l_{\max}$. In this case, full extension is required. For negative values, a full retraction is required. For the extreme scenario that there is no solution, which is only the case with exactly identical angles of the VPIP and the rod, the last state of expansion or retraction is held.

Now, we evaluate the intersection of $L_p(\zeta, \phi_r, s_f)$ and Π_{VPIP}. This is the point where the poles need to extend to. Therefore, we set the components of Equation (12) into Equation (4), abbreviating $\sin x$ with $S(x)$, $\cos x$ with $C(x)$, and $\tan x$ with $T(x)$, we get

$$T(\phi_{\text{VPIP}}) \cdot L_p(\zeta, \phi_r, s_f)_x \tag{13}$$

$$+T(\theta_{\text{VPIP}}) \cdot L_p(\zeta, \phi_r, s_f)_y \tag{14}$$

$$+L_p(\zeta, \phi_r, s_f)_z = 0\,. \tag{15}$$

Substituting the components of $L_p(\zeta, \phi_r, s_f)$ into Equation (15) yields:

$$\begin{aligned} & T(\phi_{\text{VPIP}}) \cdot \left[s_f \cdot C(\phi_r) \cdot d_m^s - (r_s + l) \cdot C(\zeta) \cdot S(\phi_r) \right] \\ + \; & T(\theta_{\text{VPIP}}) \cdot \left[-(r_s + l) \cdot S(\zeta) \right] \\ + \; & \left[r_m - s_f \cdot S(\phi_r) \cdot d_m^s - (r_s + l) \cdot C(\zeta) \cdot C(\phi_r) \right] = 0\,. \end{aligned} \tag{16}$$

Factoring out $(r_s + l)$:

$$\begin{aligned} & \overbrace{T(\phi_{\text{VPIP}}) \cdot s_f \cdot C(\phi_r) \cdot d_m^s + r_m - s_f \cdot S(\phi_r) \cdot d_m^s}^{=:N} \\ - \; & (r_s + l) \underbrace{\left[T(\phi_{\text{VPIP}}) \cdot C(\zeta) \cdot S(\phi_r) + T(\theta_{\text{VPIP}}) \cdot S(\zeta) + C(\zeta) \cdot C(\phi_r) \right]}_{=:D} = 0\,. \end{aligned} \tag{17}$$

Solving for $(r_s + l)$ and subtracting r_s yields $l = -r_s + \frac{N}{D}$, which, substituting back N and D, leads to

$$l = -r_s + \frac{T(\phi_{\text{VPIP}}) \cdot s_f \cdot C(\phi_r) \cdot d_m^s + r_m - s_f \cdot S(\phi_r) \cdot d_m^s}{T(\phi_{\text{VPIP}}) \cdot C(\zeta) \cdot S(\phi_r) + T(\theta_{\text{VPIP}}) \cdot S(\zeta) + C(\zeta) \cdot C(\phi_r)}\,. \tag{18}$$

With l, one could find the corresponding (x, y, z) of the intersection by putting l into Equation (12). However, for our evaluation, l is the wanted variable for controlling the poles to touch the VPIP; therefore, we refer to this as l_{VPIP}. This computation needs to be done for all poles in each control cycle. During regular motion we reduce the computational effort by ignoring all poles between 0.5π rad and 1.5π rad, as they need particular, extreme cases for them to interact. The overall control mechanism always controls θ_{VPIP} and ϕ_{VPIP} and sets the computed l_{VPIP} as reference value for the l of each pole. For the angles of the VPIP, we will not derive a full control system as this exceeds the scope of this paper and will be addressed in further research, but we give a short overview of the expected behavior. ϕ_{VPIP} needs to compensate for the error ϵ_ϕ between the roll angle of the robot ϕ_r and the desired ϕ_r, and therefore is in the same direction as the roll of the robot ϕ. So, if the robot rotates to the right, the VPIP will lower the right and raise the left side. This leads to an extension of the poles on the right side and a lowering of the poles on the left, which is the correct action for countering tipping to the right. The idea is to control ϕ_{VPIP} in a closed-loop manner on the basis of ϵ_ϕ. The desired ϕ_r is 0 for pure balancing. Other values lead to a curvature in combination with a forward locomotion. In the same manner as roll, the pitch θ_{VPIP} is controlled by the commanded rotation speed c_ω and the resulting error ϵ_ω. Lowering the VPIP on the back and raising it on the front side of the robot will

lead to retraction of poles to the front and extension of poles to the back, leading to forward motion. As the ζ change during this process, l_{VPIP} changes for specific poles, with the actual VPIP staying the same. Therefore, a $\theta_{\text{VPIP}} \neq 0\,\text{rad}$ leads to a continuous motion. With increasing c_ω, θ_{VPIP} needs to be chosen steeper. This is also done using a closed-loop controller. Figure 5 depicts the resulting controller architecture.

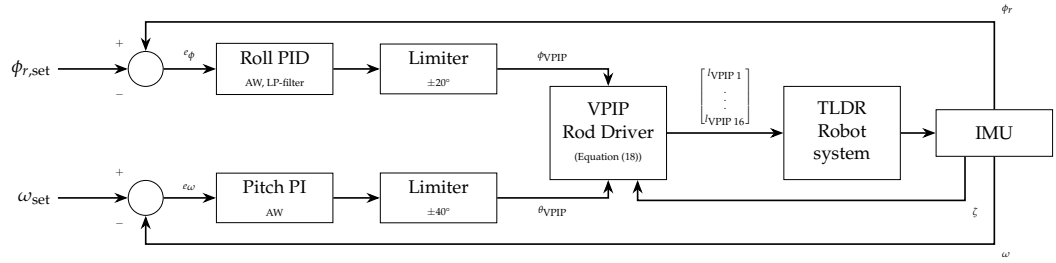

Figure 5. Control architecture of the VPIP system. The roll PID (with Anti-Windup (AW) and low-pass filter) and pitch PI (with Anti-Winddup) controller command the VPIP angles ϕ_{VPIP} and θ_{VPIP}, from which the VPIP rod driver computes the required extension lengths l_{VPIP} for all 16 rods via Equation (18). Feedback is provided by the onboard IMU.

5. Simulation and Results

To validate the principle of the VPIP, we simulate the spherical robot with 16 rods in Gazebo (see Figure 6).

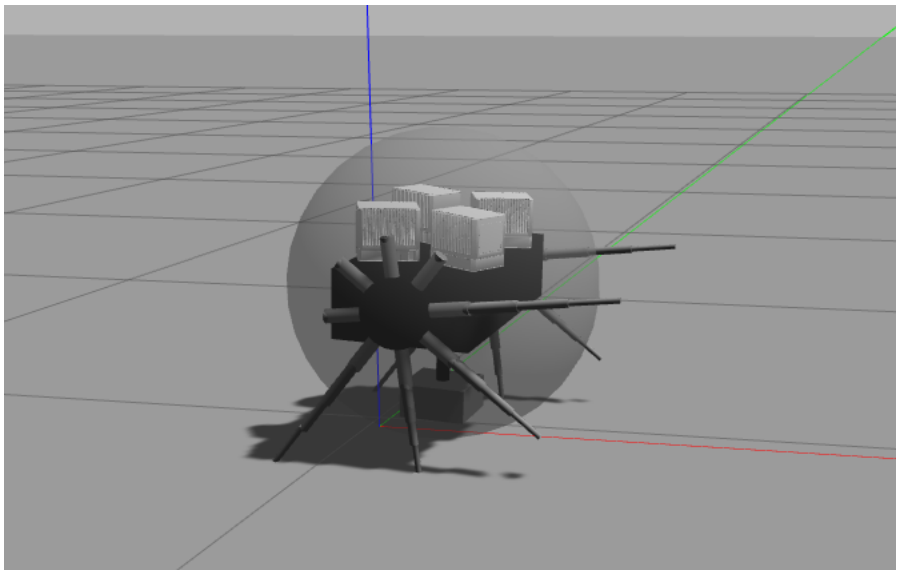

Figure 6. Gazebo simulation of a TLDR robot. The 16 rods extend and lead to rotation, hence translation of the spherical robot.

All data of the simulated robot matches the dimensions of the concept of the original DAEDALUS robot. Table 1 shows the used paramteres. To better understand the underlying dynamics for interpreting results and behaviors, please refer to our work on the dynamics of TLDR robots [15].

In these initial simulations, we want to show that it is possible to control both speed and lateral tilt angle via the VPIP. We also want to conduct a brief initial test to demonstrate that this lateral tilt allows for variable cornering. More specific movements for precise path tracking and exact cornering control must be part of future research.

Table 1. Simulation parameters of the DAEDALUS robot model in Gazebo.

Parameter	**Value**
Sphere	
Radius r_m	0.232 m
Mass	3 kg
Inertia (all axes)	$\frac{2}{3}mr^2 = 0.108\,\text{kg}\,\text{m}^2$
Contact friction * μ_1/μ_2	200.0/100.0
Inner structure	
Baseplate mass **	15 kg
Battery mass	7 kg
Rods (per rod, 3 telescopic segments)	
Number of rods	16 (8 per side)
Segment length	0.07 m
Max. extension $l_{\max}$	0.21 m
Extension velocity	0.1 m/s
Mass per rod	1.4 kg
VPIP Controllers	
Pitch PI (k_p/k_i)	10.1/2.1
Roll PID ($k_p/k_i/k_d$, f_c)	3.0/2.0/3.5, 50 Hz
Max. $\theta_{\text{VPIP}}/\phi_{\text{VPIP}}$	40°/20°
Anti-windup *** pitch/roll	10°/5°
Environment & Control	
Simulation environment	Gazebo, flat ground ****
Overall Control frequency	100 Hz

* High friction values are chosen to prevent numerical slipping artifacts in Gazebo. ** The baseplate is simulated with the weight of all non-included Subsystems of the final robot. *** Anti-Windup limits the Integral Part of the PID. **** Flat surfaces are used for the tests shown in this work. However, the functionality has also been successfully tested on uneven ground.

Figure 7 shows the resulting rotational velocity ω of the robot for different static pitch angles of the VPIP, as well as the simple Algorithm from [5] which does not use VPIP. There are multiple key findings. First, the VPIP approach does yield higher accelerations as well as a higher maximum ω compared the simple algorithm. Second, we see that there is a minimum VPIP angle for movement, as a θ_{VPIP} of 10° does not lead to acceleration but oscillation. Third, a θ_{VPIP} of 30° leads to the maximum possible ω. Steeper θ_{VPIP} also cause oscillations. And fourth, a θ_{VPIP} of 40°, which does not lead to a maximum ω, does lead to the highest acceleration. Especially the third and fourth findings show the potential of a closed-loop controller. The results correspond to the expected behavior, as the maximum ω is highly dependent on a well-timed sequence of all used rods, as they easily can contradict each other if the timing is sub-optimal [5]. Also, too high pitches lead to oscillation in the roll direction, which has a negative effect on locomotion.

Therefore, we now use a simple PI-controller to control the pitch of VPIP (θ_{VPIP}) and use the resulting ω as return to create a closed-loop control. Figure 8 shows the first result of the step response of a PI-controlled VPIP pitch angle. This leaves room for improvement and will be refined in future research, but it shows the basic capability of the closed-loop control approach of the VPIP. The standstill in particular still holds oscillations, which we intend to reduce by tuning the controller.

The observed behavior in Figure 8 is due to the interplay between rod extension velocity and the rotational dynamics of the sphere. At $\theta_{\text{VPIP}} = 30$, the changes in the commanded extension lengths roughly match the maximum achievable extension velocity of the rods, resulting in the optimal number of rods contributing constructively to forward motion. Crucially, at this angle not only do more rods contact the ground simultaneously, but each

maintains contact for a longer duration, allowing each to transfer more force/energy to the sphere. This combination of well-timed contact and sustained force transfer yields the highest ω. For steeper angles like ($\theta_{\text{VPIP}} = 40$), rods contact the ground with higher instantaneous force, leading to higher acceleration, but the contact duration itself decreases as immediately the rod needs to extend faster than its maximum speed to stay in contact with the ground and fewer rods therefore reach the required extension length in time (in our example only one or none rods touched the ground at any time at commanded VPIPs with $\theta_{\text{VPIP}} > 30$, limiting the achievable maximum ω.

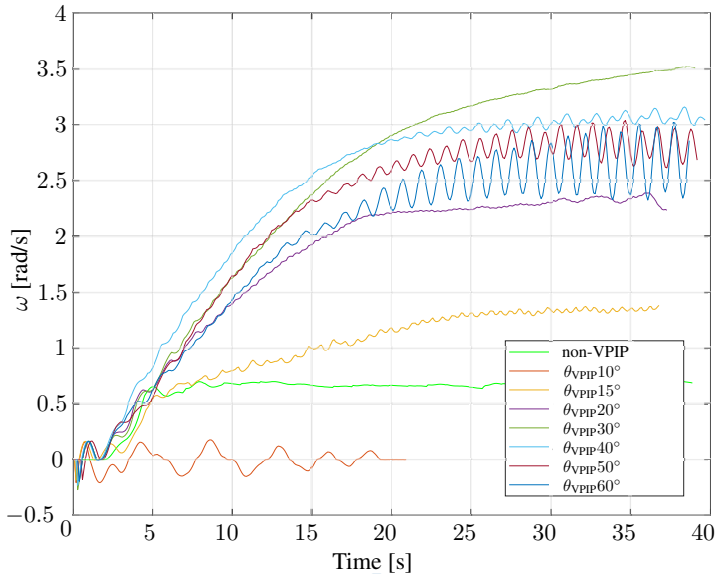

Figure 7. Rotational velocity of the robot using a simple push algorithm introduced in [5] and different static pitch angles for the VPIP (θ_{VPIP}) angles, using the Gazebo simulation. These are all open-loop control strategies without a set end-θ.

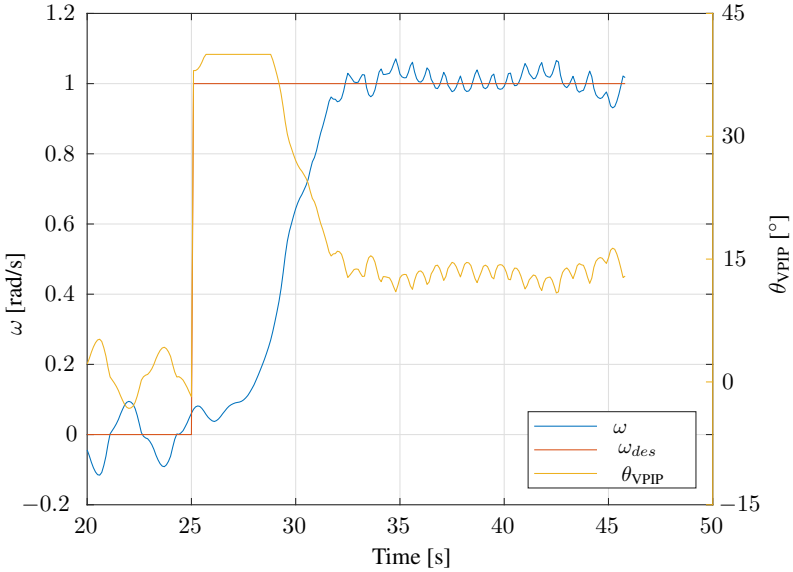

Figure 8. The resulting rotational velocity of the robot and θ_{VPIP} using a PI controller for the VPIP.

The oscillation at $\theta_{\text{VPIP}} = 10$ stems from the same fundamental limitation, but on the retraction side. At shallow commanded VPIP angles, forward-facing rods are geometrically closer to the ground and must retract further and therefore faster to clear it. Just as a rod at a steep angle cannot extend fast enough to maintain ground contact, a rod at a shallow angle cannot retract fast enough to avoid the sphere rolling onto it, causing a

bounce, leading to an oscillation back and forth (as the pushing rods will again push). This small oscillation again excites the internal pendulum mass (DAEDALUS holds a second locomotion mechanism, as its battery functions as Pendulum) and leads to higher persistent oscillation. This minimum angle threshold is therefore a direct consequence of the finite rod actuation speed, which represents a fundamental mechanical constraint of the TLDR concept first discussed in [5]. While higher control frequencies and refined retraction strategies will reduce this threshold considerably, a nonzero minimum angle is expected to persist under any realistic actuation speed. Observing and therefore confirming this boundary behavior (minimal and maximum) is one of the intended contributions of this work. Creating and improving strategies to improve both maximum and minimum speed, is up to future work.

Next, we add the control of the roll angle of the VPIP (ϕ_{VPIP}). The pure balancing of the robot, i.e., holding the roll of the robot ϕ_r at 0°, was already achieved in the original simple algorithm. But creating and holding $\phi_r \neq 0°$ was not achieved. Controlling ϕ_{VPIP} in a closed-loop manner with ϕ_r as feedback leads to robust control of the roll angle of the robot. Initial experiments showed that both a PID and an LQR are viable approaches for roll angle control. As a PID with a filtered derivative term represents a simpler and more accessible baseline, we adopt this as the controller of choice for the present work. Since the PID output was limited to prevent unwanted VPIP behavior (steeper angles lead to fewer poles touching the ground) , a Conditional Integration Anti-Windup (AW) was added to prevent the integral term from accumulating excessively during saturation, which otherwise causes large overshoots and instability. Additionally, a low-pass filter was integrated into the derivative term to suppress high-frequency noise observed in the derivative output, improving the controller's stability and making it more robust for potential real-world applications where sensor noise is unavoidable. We explicitly make no statement about an optimal controller for the two angles of the VPIP, and a detailed comparison of several controller structures and their respective fine-tuning is left for future research. Figure 9 shows the establishment of three different angles of the robot by controlling the roll angle of the VPIP.

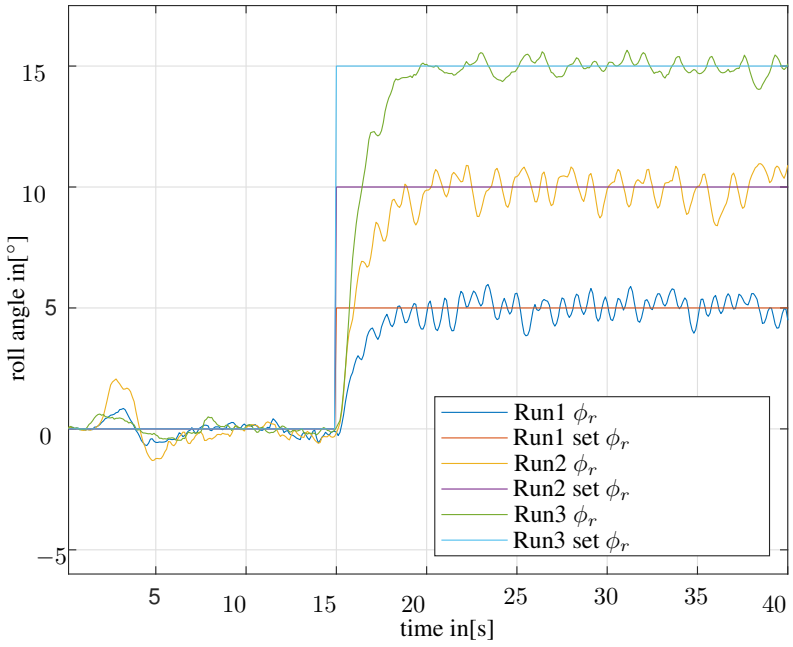

Figure 9. Resulting ϕ_r when controlling the roll of the VPIP with an PID controller. There were three test runs with different desired ϕ_r.

Testing different values for the PI controller shows that the controller of the ϕ_{VPIP} is much more robust and quicker to tune than the PID for θ_{VPIP}. The reasons for this needs to be investigated in further research.

The last step is to combine balancing and forward motion by simultaneously controlling roll and pitch of the VPIP. First tests show the feasibility of complete control, as it is possible to roll smooth curves with the spherical robot. We want to show, that the curve radius is dependent on ϕ_r. Figure 10 shows the x-y-plotting of three test runs with different VPIP-controlled ϕ_r and a VPIP-controlled ω. In the presented runs (each 30 s long), desired roll angles of 5, 10, and 15 resulted in circle diameters of approximately 2.6 m, 2.0 m, and 1.7 m respectively, indicating the expected inverse relationship between roll angle and curve radius. A systematic quantitative characterization of this relationship across a wider range of parameters is left for future work.

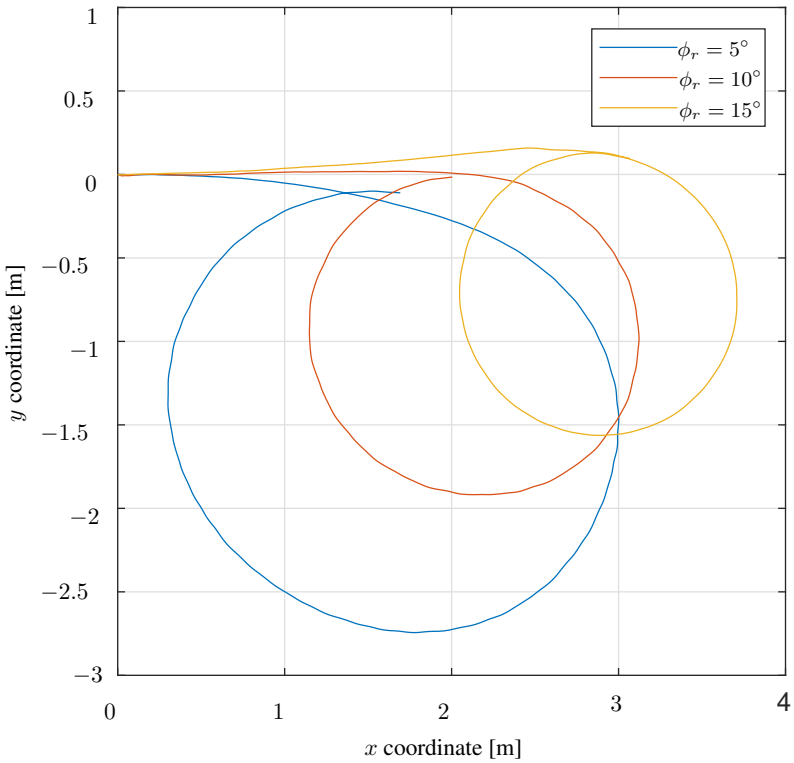

Figure 10. Top view of the rolled path of the TLDR robot, controlling both its ω, as well as two different ϕ_r via VPIP (PID for ϕ_r, PI for $\omega = 1\text{rad/s}$). Time duration each 30 s.

This paves the way for complete path following and simultaneous stabilization with VPIP. A further step is to have a cascaded control loop and control the desired ϕ_r of the robot over a desired curve radius. This will enable full path following for a TLDR robot. Even with the potential for improvement of the ω control with the help of a more fine-tuned PI controller, or even with other control concepts, we state the VPIP to be a working control strategy for a TLDR robot.

6. Conclusions

For combining the balancing and locomotion of spherical robots using rod-driven locomotion, we introduced the idea of a virtual plane controlling the robot, the virtual pose instruction plane (VPIP), and established a mathematical solution. The concept introduces two controllable parameters (roll and pitch of the plane) to control the balance and movement velocity of a spherical robot. We tested the first implementation successfully in simulation and showed the feasibility of a full-VPIP control strategy for the TLDR robot.

All in all, the VPIP seems to be a promising approach for the TLDR to become a robust locomotion approach.

Needless to say, a lot of work remains to be done. The tuning of the controllers for the angles of VPIP is suboptimal and leaves room for improvement, either with fine-tuning or with the use of other control concepts. Also, a further step is to evaluate the VPIP approach on a real prototype. Once the VPIP itself is sufficiently robust, we plan to extend the concept beyond a single plane. Rather than assuming a flat reference surface, we aim to generalise the VPIP to incorporate a map of the immediate surroundings od the robot—leading to the Virtual Pose Instruction Map (VPIM). Whilst the VPIP already reacts to terrain disturbances reactively due to its control-loop structure, the VPIM will enable proactive adaptation by directly incorporating terrain fluctuations via the point cloud obtained by the LiDAR before they manifest as control errors. This extension will then be seamlessly integrated with the LiDAR sensors planned for the DAEDALUS mission: Beyond their role as scientific instruments for cave mapping, these sensors will simultaneously serve as terrain estimators for the control mechanism—a fruitful symbiosis that would make spherical TLDR robots significantly more capable in the harsh, unknown terrains of lunar caves. This work provides the conceptual as well as mathematical foundation for this idea. We are currently working on this VPIM approach and look forward to presenting the results in the future.

Overall, the VPIP concept holds potential for advancing the field of spherical robot locomotion, and we look forward to further exploring its capabilities and applications in future research.

Author Contributions: Conceptualization, J.Z.; methodology, J.Z.; software, J.Z., J.B., M.H. and K.M.; validation, J.Z., J.B. and M.H.; formal analysis, J.Z.; investigation, J.Z.; data curation, J.Z., J.B. and M.H.; writing—original draft preparation, J.Z.; writing—review and editing, J.Z., A.B. and D.B.; visualization, J.Z., J.B. and M.H.; supervision, D.B. and A.N.; project administration, A.N.; funding acquisition, A.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the European Space Agency (ESA) Contract No. 4000130925/20/NL/GLC for the "DAEDALUS—Descent And Exploration in Deep Autonomy of Lava Underground Structures" Open Space Innovation Platform (OSIP) lunar cave-system study and the Elite Network Bavaria (ENB) for providing funds for the academic program "Satellite Technology".

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Seeni, A.; Schafer, B.; Rebele, B.; Tolyarenko, N. Robot mobility concepts for extraterrestrial surface exploration. In *Proceedings of the 2008 IEEE Aerospace Conference*; IEEE: Piscataway, NJ, USA, 2008; pp. 1–14.
2. Agency, E.S. SysNova. Available online: http://www.esa.int/Enabling_Support/Preparing_for_the_Future/Discovery_and_Preparation/SysNova2 (accessed on 9 March 2026).
3. Rossi, A.P.; Maurelli, F.; Unnithan, V.; Dreger, H.; Mathewos, K.; Pradhan, N.; Corbeanu, D.A.; Pozzobon, R.; Massironi, M.; Ferrari, S.; et al. DAEDALUS-Descent And Exploration in Deep Autonomy of Lava Underground Structures: Open Space Innovation Platform (OSIP) Lunar Caves-System Study. 2021. Available online: https://opus.bibliothek.uni-wuerzburg.de/frontdoor/index/index/docId/22791 (accessed on 21 April 2026).
4. Kesner, S.B.; Plante, J.S.; Boston, P.J.; Fabian, T.; Dubowsky, S. Mobility and power feasibility of a microbot team system for extraterrestrial cave exploration. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*; IEEE: Piscataway, NJ, USA, 2007; pp. 4893–4898.

5. Zevering, J.; Borrmann, D.; Bredenbeck, A.; Nüchter, A. The concept of rod-driven locomotion for spherical lunar exploration robots. In *Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*; IEEE: Piscataway, NJ, USA, 2022; pp. 5656–5663.
6. Mateos, L.A. Bionic sea urchin robot with foldable telescopic actuator. In *Proceedings of the 2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*; IEEE: Piscataway, NJ, USA, 2020; pp. 1063–1068.
7. Yang, D.; Liu, Y.; Yu, Y. A General Locomotion Approach for a Novel Multi-legged Spherical Robot. In *Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA)*; IEEE: Piscataway, NJ, USA, 2023; pp. 10146–10152.
8. Ningrum, E.S.; Brata, S.W.; Fauziah, K.P.; Binugroho, E.H. Introducing the PodRacle, spherical mode rolling locomotion on flat surface. In *Proceedings of the 2016 International Conference on Knowledge Creation and Intelligent Computing (KCIC)*; IEEE: Piscataway, NJ, USA, 2016; pp. 8–13.
9. Diouf, A.; Belzile, B.; Saad, M.; St-Onge, D. Spherical Rolling Robots—Design, Modeling, and Control: A Systematic Literature Review. *Robot. Auton. Syst.* **2024**, *175*, 104657. https://doi.org/10.1016/j.robot.2024.104657.
10. Bu, S.; Yan, L.; Gao, X.; Wang, G.; Zhao, P.; Chen, I.M. Design and Motion Control of Spherical Robot with Built-in Four-Wheel Omnidirectional Mobile Platform. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 1–10. https://doi.org/10.1109/TIM.2023.3267524.
11. Pravecek, D.J.; Oevermann, M.J.; Thomas, G.C.; Ambrose, R.O. Empirically Compensated Setpoint Tracking for Spherical Robots with Pressurized Soft-Shells. *IEEE Robot. Autom. Lett.* **2025**, *10*, 2136–2143. https://doi.org/10.1109/LRA.2025.3527308.
12. Zhang, B.; Zhang, F.; Chen, H.; Wang, Y.; Hao, J.; Luo, Z.; Li, G. A High-Speed Capable Spherical Robot. *arXiv* **2025**, arXiv:2511.01288. https://doi.org/10.48550/arXiv.2511.01288.
13. Nozaki, H.; Kujirai, Y.; Niiyama, R.; Kawahara, Y.; Yonezawa, T.; Nakazawa, J. Continuous Shape Changing Locomotion of 32-Legged Spherical Robot. In *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*; IEEE: Piscataway, NJ, USA, 2018; pp. 2721–2726. https://doi.org/10.1109/IROS.2018.8593791.
14. Xu, L.; Ren, R.; Wei, X.; Lee, H.; Zhang, H.; Yu, K.; Li, Y.; Liu, C.; Gao, S.; Lu, L. Design and Development of a Deformable Spherical Robot for Amphibious Applications. In *Proceedings of the 2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*; IEEE: Piscataway, NJ, USA, 2025; pp. 6933–6938. https://doi.org/10.1109/IROS60139.2025.11246697.
15. Zevering, J.; Borrmann, D.; Bredenbeck, A.; Nüchter, A. Dynamics of Spherical Telescopic Linear Driven Rotation Robots. In Proceedings of the 16th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2023), European Space Agency (ESA), Leiden, The Netherlands, 18–20 October 2023.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.