

Development of a Dual-Mode Spherical Robot Using a Differential Drive

Abstract—We present a spherical robot, named Mosphero, with dual driving modes to enhance adaptability on flat ground and traversal of deformable terrain. A coaxial dual-pendulum mechanism with bevel gear coupling enables omnidirectional rolling, sharp turns, side-rolling, and modular expandability. In impulsive mode, the pendulums accelerate and brake to generate torque impulses for escaping resistive environments. A control system combines velocity regulation for stable rolling with mode switching for additional torque. Experiments supported by simulations show that closed-loop control reduced the Integral of Absolute Error (IAE) by 87.1% compared with open-loop operation, while impulsive actuation shortened escape time on deformable terrain by a factor of 7.4 and nearly doubled the success rate. The 2D trajectory tests confirmed omnidirectional locomotion, including straight-line motion, sharp turns, and side-rolling.

I. INTRODUCTION

The ability to navigate diverse environments is essential for robots performing complex tasks. Existing mobile platforms, such as wheeled and legged robots, each have structural limitations. Wheeled robots are efficient on flat surfaces but expose internal components to clogging and damage [1], [2], while legged robots can traverse uneven terrain but often suffer from fragile joints and reliability concerns [1], [3]. These constraints have motivated the development of alternative configurations. Spherical robots employ fully enclosed shells that shield internal mechanisms from dust, water, and impacts. Their continuous geometry enables omnidirectional locomotion, self-righting, and stable maneuvering [4]. These properties differentiate spherical robots from conventional wheeled and legged designs.

The actuation mechanisms of spherical robots can be categorized into four major types: (i) direct-drive systems with internal vehicles transmitting torque to the shell [5]–[8], (ii) gravity-based approaches shifting the center of mass with pendulums or movable weights [9]–[11], (iii) angular momentum methods using flywheels [12], [13], and (iv) shape-deformation strategies with compliant shells [14]. Table I summarizes these approaches. The row labeled Motor input refers to the number of motors required to realize locomotion. Omnidirectional locomotion specifies whether the robot can continuously reach arbitrary configurations from any starting orientation. Sharp turn denotes the capability to perform abrupt changes in direction along the trajectory. Expandability indicates whether the mechanism allows additional modules to be mounted on the spherical shell without interfering with locomotion.

Among these categories, pendulum-driven spherical robots (PDSRs) provide a favorable balance. Compared with angular momentum methods, which offer decoupled control but require higher energy consumption, PDSRs achieve omnidirectional locomotion with fewer motors and, importantly,



Fig. 1. Our spherical Robot — Mosphero.

retain the possibility of expandability. These considerations motivate our focus on the single-pendulum configuration as the foundation for the proposed design.

Notable examples of PDSRs include KisBot II, which employs dual pendulums on a cross frame for agile reorientation [15]; XK-I, which introduced a square frame with two external motors—one driving the pendulum and the other enabling tilting—for directional steering [16]; XK-II, which refined this concept by incorporating a rotating plate and ballast box to achieve smaller-radius curved motion [17]; and XK-III, which further extended capability with a geared platform providing a redundant third DoF for omnidirectional maneuverability [18]. Other representative PDSRs include ISR, which separates longitudinal and lateral pendulums to reduce energy consumption [19]; Ren et al.’s design, featuring a sealed single-pendulum structure with orthogonal shafts for enhanced durability [20]; and the commercial GroundBot, a rugged, long-endurance robot optimized for amphibious tasks [21]. A comparative summary of these PDSR designs is provided in Table II.

We report on the development of a novel dual-mode spherical robot, Mosphero, using a differential drive, as shown in Fig. 1. The contribution of this work is three-fold. First, we introduce a dual-mode startup strategy that integrates gravity-based and angular-momentum-based actuation, enabling self-rescue ability. Second, we demonstrate that a single-pendulum configuration can realize continuous omnidirectional locomotion, side rolling, and self-rescue, despite the inherent coupling between pendulum and shell inertia. Third, we design the structure with expandability, a capability rarely achieved in prior PDSRs.

TABLE I
COMPARISON TABLE OF SPHERICAL ROBOTS WITH DIFFERENT DRIVING METHODS

	Single Wheel	Hamster Car	Single Ball	Single Pendulum	Dual Pendulum	Mass Movement	Orthogonal Flywheels	Morphex
Category	Hamster ball / Direct driving			Gravity		Angular momentum	Shape deformation	
Motor input	2	≥ 2	2	2	3	4	2	≥ 2
Omnidirectional locomotion	X	X	V	V*	V*	V	V*	X
Sharp turn	V	V	V	V*	V*	V	V*	X
Expandability	X	X	X	Δ	Δ	V	V	V

* Achievable in most conditions (may fail at singular configurations).

TABLE II
COMPARISON OF RECENT PENDULUM-DRIVEN SPHERICAL ROBOTS

Aspect							
	KisBot II	XK-I	XK-II	XK-III	ISR	Ren et al.	Mosphero
Degrees of freedom	2	2	2	3	2	2	2
Motors	3	3	4	3	2	2	2
Expandability	X	X	X	X	X	Δ	V
Side rolling	V	V	X	X	X	X	V
Omnidirectional locomotion	V	V*	X	X	X	X	V*
Self-rescue	N/A	N/A	N/A	N/A	N/A	N/A	V

* Achievable in most conditions (may fail at singular configurations).

The remainder of this paper is structured as follows. Section II introduces the overall system, including the mechanical and hardware design, and coordinate definitions. Section III develops the kinematic and dynamic models used for analysis. Section IV presents the control framework, addressing both angular velocity regulation and impulsive momentum-based maneuvers. Section V provides experimental validation through setup description, performance evaluation of the control methods, and demonstrations of planar locomotion.

II. SYSTEM OVERVIEW

A. Mechanism design

The proposed spherical robot is designed to overcome the limited side-rolling and self-rescue capabilities of previous pendulum-driven designs. To this end, torque from two coaxial motors is transmitted through a bevel gear set, enabling two-axis rotation while keeping the motors mechanically aligned, as shown in Fig. 2. This structural arrangement enhances maneuverability beyond prior pendulum-driven spherical robots by allowing sharp turns, omnidirectional rolling (except at singular configurations), and side-rolling maneuvers.

Beyond gravity-driven locomotion, the robot incorporates an impulsive momentum mode to further extend its mobility.

In this mode, both pendulums are accelerated together to accumulate kinetic energy, and then abruptly braked to impart a strong impulsive torque to the outer shell. The result is a powerful burst of motion that can generate the startup force necessary for the robot to escape from deformable terrain. By combining conventional gravity-based control with impulsive actuation, the design ensures that the robot is not only efficient during regular rolling but also capable of handling resistive surface conditions that demand additional torque.

The spherical shell of the robot also functions as a key structural component that supports modularity and expandability. Standardized latching interfaces are integrated into the shell, allowing modular attachment of actuation, sensing, processing, and power units. Since the motors are unified with the shell, this modular framework ensures that the robot can be easily adapted for future upgrades or extended applications. Such a design makes it possible to integrate additional subsystems without requiring a complete redesign.

B. Hardware

At the hardware level, the spherical robot is comprised of four integrated subsystems: actuation, sensing, computation, and power supply. The actuation subsystem consists of a NEMA 23 stepper motor (57BYGH56, CNYOHO, China),

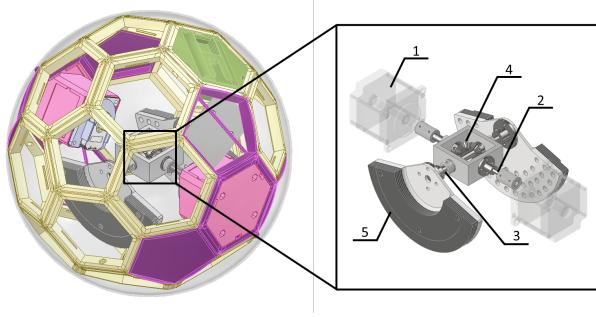


Fig. 2. Mechanical structure of the spherical robot. The left panel shows the assembled robot with highlighted modules: green for power units, purple for control units, and pink for actuation units. The right panel shows a simplified internal mechanism with labeled components: 1: motor, 2: main axis, 3: auxiliary axis, 4: bevel gear set, 5: pendulum.

an incremental encoder (AMT102-V, Same Sky, USA), and a motor driver (TMC2160, Makerbase, China), all coordinated by an Arduino Nano running a proportional–integral (PI) control loop. This dedicated setup offloads real-time motor control from the central processor, ensures the update rate required for closed-loop operation, and provides both the high instantaneous torque and the precision positioning necessary for pendulum actuation. For sensing, an inertial measurement unit (IMU) (ICM-20948, SparkFun Electronics, USA) is paired with a microcontroller (NodeMCU-32S, Espressif, China). The IMU’s digital motion processor (DMP) computes raw acceleration and angular velocity data, which the ESP32 preprocesses before transmitting motion estimates to the central controller via the I²C bus.

The computation subsystem is centered on a single board computer (Raspberry Pi Zero 2W, Raspberry Pi Foundation, UK), which is responsible for high-level tasks such as sensor data acquisition, execution of control algorithms, and coordination of actuation commands. Communication across all subsystems is implemented through a unified I²C bus, ensuring consistent data flow while minimizing wiring complexity. Power is supplied by 11.1 V lithium battery packs: each Arduino is powered by a single pack, each motor is supported by two packs to provide sufficient voltage, and the Raspberry Pi is supplied by a 3.7 V lithium battery regulated through an uninterruptible power supply (UPS-Lite, ACE Design Studio, China) module for stable operation.

C. Coordinate

To model and analyze the robot’s motion, we establish four coordinate frames: the World Frame $\{W\}$, the Robot Frame $\{R\}$, the Shell Frame $\{S\}$, and the Pendulum Frame $\{P\}$, as shown in Fig. 3. The World Frame $\{W\} = (X_W, Y_W, Z_W)$ is an inertial frame where the $X_W - Y_W$ plane coincides with the ground, and the robot’s planar position is defined by the contact point $C_W = (X_{CW}, Y_{CW})$. Two body-fixed frames, $\{R\}$ and $\{S\}$, share a common origin at the geometric center of the sphere. The Robot Frame $\{R\} = (X_R, Y_R, Z_R)$ reflects the robot’s heading: it only rotates about the vertical axis, with Y_R aligned to the forward direction of travel. By contrast, the

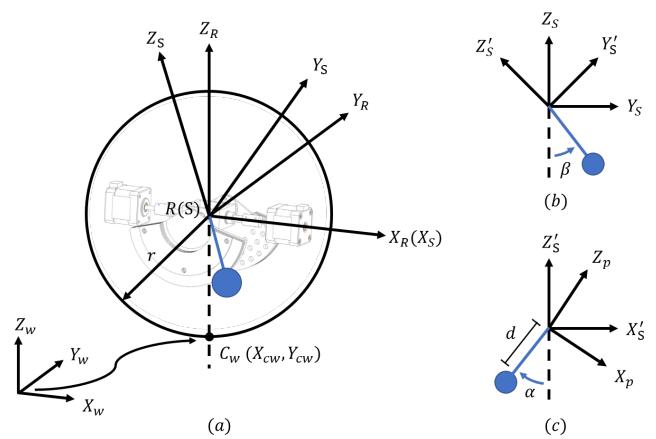


Fig. 3. Definition of system variables and coordinate frames for the spherical robot. (a) Relationship between the inertial World Frame $\{W\}$ and the body-fixed Robot Frame $\{R\}$ and Shell Frame $\{S\}$, showing the contact point C_W . (b) The pendulum roll angle, β , defined as a rotation about the shell’s X_S axis. (c) The pendulum tilt angle, α , defined as a rotation relative to the vertical Z_S axis.

Shell Frame $\{S\} = (X_S, Y_S, Z_S)$ is rigidly attached to the outer shell and undergoes full three-dimensional rotation, with its orientation relative to $\{W\}$ parameterized by ZYX Euler angles (ϕ, θ, ψ) . Together, these coordinate frames establish a complete reference for describing the robot’s motion in both global and body-centered perspectives. The coordinate definitions are illustrated in Fig. 3.

The internal driving mechanism is modeled as a simple pendulum attached to the sphere’s center and offset by a distance d along the $-Z_P$ axis of the pendulum frame. Its orientation relative to $\{S\}$ is determined by two angles. The roll angle β is defined as a rotation about the X_S axis, producing an intermediate frame $\{S'\}$. The tilt angle α is then defined by rotation about the $X_{S'}$ axis within the $X_{S'} - Z_{S'}$ plane, measured from the vertical $Z_{S'}$ axis. This sequence of rotations establishes the pendulum frame $\{P\}$, whose orientation is uniquely specified by α and β . These two parameters directly determine the pendulum’s position, and thus the location of the internal mass. By controlling α and β , gravitational torque is generated on the shell, driving the rolling motion of the robot. As a result, these angles serve as the fundamental control inputs of the system.

III. MODEL

A. Kinematics

1) *Forward Kinematics:* The forward kinematics describe how the pendulum orientation determines the contact point on the spherical shell. Assuming the pendulum points downward (parallel to the Z_W axis), the shell contact point on the sphere of radius r is parameterized by (α, β) as

$$\begin{aligned} x_S &= -r \sin \alpha \\ y_S &= r \cos \alpha \sin \beta \\ z_S &= -r \cos \alpha \cos \beta \end{aligned} \quad (1)$$

TABLE III
LIST OF SYMBOLS AND DEFINITIONS

Symbol	Definition
$\{W\}$	World frame
$\{R\}$	Robot frame
$\{S\}$	Shell frame
$\{P\}$	Pendulum frame
x_S, y_S, z_S	Contact point between shell and floor expressed in $\{S\}$
\vec{l}	Contact line increment in the inertial (world) frame
\vec{l}_S	Contact line increment in the shell frame
ϕ, θ, ψ	ZYX Euler angles of $\{S\}$ w.r.t. $\{W\}$
α	Pendulum tilt angle, rotation about X_S , yielding $\{P\}$
β	Pendulum roll angle, rotation about X_S , yielding $\{S'\}$ (motor command angle)
γ	Angle of inclined plane
u	Pendulum roll angle, rotation about X_R (control input)
ϕ_r, ϕ_l	Turning angles of right and left motors
M	Total mass of the robot
m_p	Mass of pendulum
I_c	Spherical shell mass moment of inertia about point C_W
I_p	Pendulum mass moment of inertia about center of mass
r	Radius of spherical shell
d	Pendulum's center of mass offset along $-Z_P$ axis
d_G	Radius of robot mass center

Taking differentials yields the Jacobian

$$\underbrace{\begin{bmatrix} dx_S \\ dy_S \\ dz_S \end{bmatrix}}_{d\vec{l}_S} = \underbrace{\begin{bmatrix} -r\cos\alpha & 0 \\ -r\sin\alpha\sin\beta & r\cos\alpha\cos\beta \\ r\sin\alpha\cos\beta & r\cos\alpha\sin\beta \end{bmatrix}}_{J(\alpha, \beta)} \begin{bmatrix} d\alpha \\ d\beta \end{bmatrix} \quad (2)$$

Let \mathbf{R}_i^0 be the shell-to-inertial rotation at step i . The no-slip contact-line increment in the inertial frame is

$$d\vec{l} = \mathbf{R}_i^0 d\vec{l}_S, \quad s = \|d\vec{l}\|$$

On a flat floor, the sphere undergoes an incremental rotation about the axis $\hat{k} = \hat{Z}_W \times \left(\frac{d\vec{l}}{\|d\vec{l}\|} \right)$ by the angle $\theta = s/r$. Using an axis-angle update,

$$\mathbf{R}_i^{i+1} = \text{ROT}(\hat{k}, \theta) \quad (3)$$

Finally, the current position in the inertial frame can be obtained by integrating the arc elements at each time step,

$$[x_w, y_w, z_w]^T = \sum_0^i \mathbf{d}\vec{l} \quad (4)$$

2) *Inverse Kinematics:* The inverse kinematics describe how to compute the pendulum angles from the shell contact position. From Eq. (1), we can obtain the relation between α and β based on the projection of the pendulum onto the sphere. The expressions are given as

$$\alpha = -\sin^{-1}\left(\frac{x_S}{r}\right), \quad \beta = \tan^{-1}\left(\frac{y_S}{-z_S}\right) \quad (5)$$

where (x_S, y_S, z_S) denotes the shell contact position in the shell frame and r is the sphere radius.

3) *Motor Kinematics:* The motor kinematics define the coupling between the motor rotations and pendulum angles through the gear mechanism. The gear mechanism introduces a kinematic relationship between the motor and pendulum angles. This relationship can be expressed using the following transformation matrices:

$$\begin{bmatrix} d\phi_r \\ d\phi_l \end{bmatrix} = \begin{bmatrix} -1 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} d\alpha \\ d\beta \end{bmatrix} \quad (6)$$

B. Dynamics

The dynamic model is formulated using the Lagrangian method to capture the coupled motion of the shell and pendulum. The equations of motion for the spherical shell-pendulum system are derived using the Lagrangian formalism, a standard approach for modeling this class of robots [22]. As illustrated in Fig. 4, the system consists of a spherical shell of radius r and an internal pendulum of length d , mass m_p , and moment of inertia of I_p , pivoted at the shell's center. The robot is assumed to roll without slipping on a plane inclined at an angle γ . The configuration of this system is described by two generalized coordinates: the pendulum's angle u and the shell's absolute rotation angle β . Consequently, the net rotation of the shell is given by $\phi = \beta - u$.

The kinetic energy of the system consists of contributions from the shell rotation and the pendulum motion. The total kinetic energy T combines the rotational energy of the shell and the kinetic energy of the pendulum mass. It is expressed as:

$$T = \frac{1}{2} I_c (\dot{\beta} - \dot{u})^2 + \frac{1}{2} I_p \dot{u}^2 + \frac{1}{2} m_p ((r(\dot{\beta} - \dot{u}) + d\dot{u} \cos(u))^2 + (d\dot{u} \sin(u))^2) \quad (7)$$

The total gravitational potential energy V of the system, considering the positions of both the shell's center and the pendulum mass on the inclined plane, is given by:

$$V = Mg(r(\beta - u) \sin(\gamma) + r \cos(\gamma) - d_G \cos(\gamma + u)) \quad (8)$$

The non-holonomic constraint of rolling without slip is maintained by a static friction force. Since this force does no work, the system can be treated as conservative. The equation of motion is therefore given by the simplified Euler-Lagrange equation:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{u}} \right) - \frac{\partial L}{\partial u} = 0 \quad (9)$$

Substituting the expressions for T and V into the Lagrange equation, we obtain the equation of motion:

$$(I_c + I_p + m_p(r^2 + d^2 - 2rd \cos(u)))\ddot{u} + (m_p(rd \cos(u) - r^2) - I_c)\ddot{\beta} + m_p rd \sin(u)\ddot{u}^2 + Mg(d_G \sin(\gamma + u) - r \sin(\gamma)) = 0 \quad (10)$$

This relation forms the equation of motion of a pendulum-driven spherical robot moving on an inclined plane.

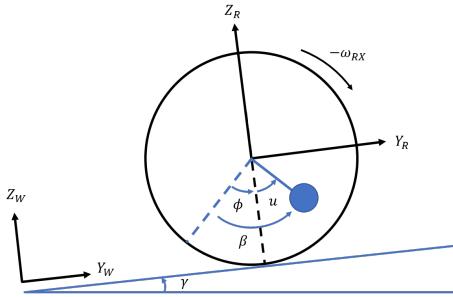


Fig. 4. Dynamic model of the pendulum-driven spherical robot on a plane inclined at angle γ , illustrating the generalized coordinates u (pendulum angle) and β (shell absolute rotation).

Having derived the equation of motion, we now incorporate the rolling constraint to obtain the final dynamic model. According to the pure rolling constraint,

$$y_w = r\phi \quad (11)$$

Since $\phi = \beta - u$, we obtain

$$u = \beta - \frac{y_w}{r} \quad (12)$$

Substituting this constraint into the equation of motion yields the final dynamic model relating the robot's linear displacement, y_w , to the shell's absolute rotation, β . This form of the equation is then used for subsequent analysis and control design.

IV. MOTION CONTROL

Our robot's motion is governed by a hierarchical control architecture, which decouples low-level motor actuation from high-level behavioral planning. This architecture consists of a low-level PI motor controller running on an Arduino and a high-level controller on a Raspberry Pi that implements two distinct modes: angular velocity control for stable locomotion and impulsive momentum control for dynamic maneuvers. The following sections detail each component of this strategy.

A. Angular Velocity Control

To regulate the robot's forward velocity in one dimension on flat ground, we apply a PI controller based on the IMU's x -axis angular-rate measurement. When $\gamma = 0$, the system equilibrium corresponds to the pendulum pointing downward and perpendicular to the Z_R axis. A forward lean of the pendulum (positive tilt relative to the Z_R axis) causes the sphere to accelerate, while a backward lean results in deceleration (see Fig. 4). Let $\omega_{ref}(t)$ denote the desired velocity (expressed in the same angular units as the sensor), and let $\omega_{RX}(t)$ denote the gyroscope reading. The control error and input are defined as

$$e(t) = \omega_{ref}(t) - \omega_{RX}(t), \quad (13)$$

$$u(t) = \beta(t) + \phi(t) = -\left(K_p e(t) + K_i \int_0^t e(\tau) d\tau\right), \quad (14)$$

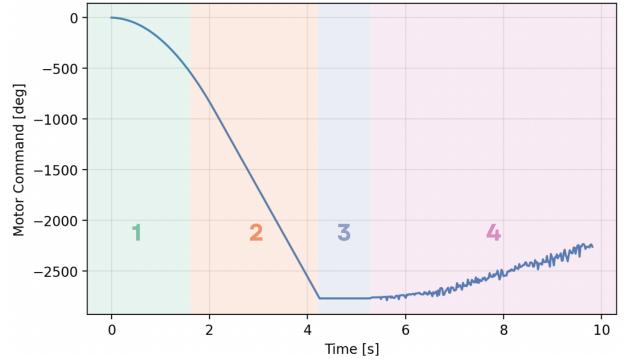


Fig. 5. Motor command angle $\beta(t)$ under the impulsive control strategy with four phases: 1: acceleration, 2: cruise at ω_{max} to N revolutions, 3: stop/release to $\omega_p(t) = 0$, and 4: closed-loop angular velocity control (Sec. IV-A). Background shading and numerals (1–4) indicate the phases.

TABLE IV
KEY PHYSICAL PARAMETERS OF THE SPHERICAL ROBOT

System Parameter	Value
r	0.15 m
I_c	0.315 kg·m ²
I_p	0.0076 kg·m ²
M	4.315 kg
m_p	0.925 kg
d	0.044 m

where $u(t)$ is the control input angle of the pendulum with respect to the equilibrium point, clipped to the range $[-90^\circ, 90^\circ]$ to avoid excessive input and to limit the torque demand. Here $K_p, K_i > 0$ are the proportional and integral gains. The leading negative sign reflects our sign convention: the shell's rotation direction is opposite the robot's forward y -motion, so the controller output must be inverted to produce the desired acceleration.

B. Impulsive Momentum Control

To execute high-torque maneuvers, the robot employs an impulsive actuation strategy based on a “wind-up and release” of the internal pendulum. In this mode, the pendulum is accelerated to a maximum angular velocity of ω_{max} and then braked abruptly after a displacement of $N = n + 0.75, n \in \mathbb{Z}$ revolutions, thereby generating a large impulsive torque on the shell. This revolution profile ensures that the pendulum not only reaches ω_{max} but is also positioned horizontally ($u = 90^\circ + 360^\circ n, n \in \mathbb{Z}$) at the braking instant. Such orientation maximizes the effective moment arm, enabling efficient conversion of the pendulum's angular momentum into a transient torque impulse that provides a rapid “kick-start” for the robot. The complete control sequence for the impulsive mode is depicted in Fig. 5. This sequence facilitates a sudden release of stored energy followed by a transition to feedback control.

V. EXPERIMENTAL RESULTS

To validate the effectiveness of the motion control strategies detailed in the previous section, we conducted exper-

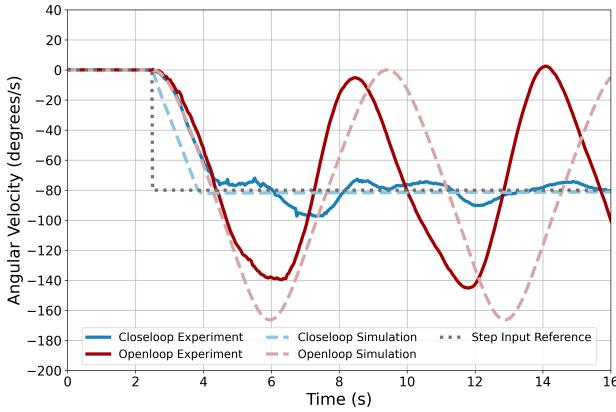


Fig. 6. Comparison of angular velocity control performance. The velocity controller ($K_p = 13, K_i = 0.5$) successfully tracks the -80 deg/s command reference, while the open-loop system exhibits highly unstable oscillations.

iments focused on the robot's three core features: stable locomotion via closed-loop control, high-torque self-rescue using an impulsive mode, and true omnidirectional maneuverability.

A. Experimental setup

All experiments were performed in the laboratory under the ground truth measurement system. For motion tracking, three reflective markers were attached to the robot's shell, and a 13-camera Vicon motion capture system recorded ground-truth data at 500 Hz for subsequent analysis. To provide a theoretical baseline for comparison, the robot's dynamic response was simulated using a model implemented in MATLAB/Simulink as well. The model's fidelity was enhanced by incorporating the natural frequency of the stepper motor system, which was determined to be approximately 2.3 rad/s through system identification. The key physical parameters used for both the hardware and the simulation are listed in Table IV.

B. Angular Velocity Control Performance Evaluation

The dynamic coupling between the pendulum and shell inertia can induce wobbling. This section validates the performance of our PI-based ($K_p = 13, K_i = 0.5$) velocity controller, first by demonstrating its necessity over an open-loop approach, and second by confirming its robustness across a range of operating speeds. To establish the necessity of feedback control, an experiment was conducted on a medium-density fiberboard surface where the robot was commanded to track a constant forward angular velocity of -80 deg/s. We compared the velocity controller with a baseline open-loop strategy, plotting both simulated and experimental results in Fig. 6.

The figure reveals two key findings. First, the open-loop system is inherently unstable; both the experimental response and the simulation exhibit oscillations, confirming that our model accurately predicts this instability. Second, the velocity controller effectively stabilizes the system. The

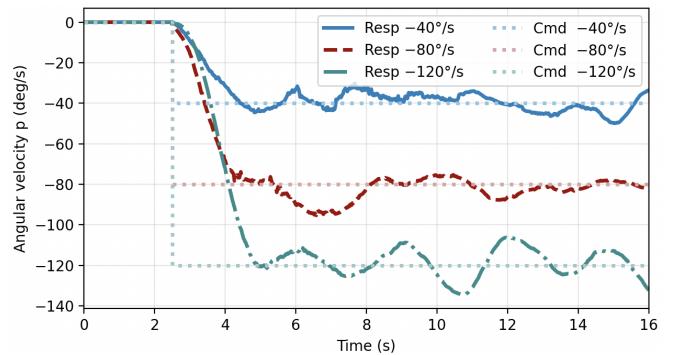


Fig. 7. Closed-loop velocity tracking performance for multiple reference speeds. The controller successfully tracks step commands of -40 , -80 , and -120 deg/s.

TABLE V
ANGULAR VELOCITY CONTROL PERFORMANCE OF MOSPERHO

Control Method	Ref (deg/s)	Rise Time (s)		IAE (deg/s)	
		Mean	Std	Mean	Std
Open-loop	-80	1.23	0.06	399.1	49.6
Closed-loop	-40	2.22	0.94	57.38	18.73
Closed-loop	-80	1.41	0.24	51.5	3.43
Closed-loop	-120	1.36	0.07	66.7	14.47

experimental response rapidly converges to the target velocity, tracking the ideal simulated velocity profile with minimal error. This strong alignment between the physical experiment and the simulation validates our dynamic model's predictive accuracy. Performance was evaluated using Integral of Absolute Error (IAE) index in the interval [6, 16] s from the experimental trials, with results averaged over five trials and summarized in Table V. On average, the velocity controller achieved an IAE that was 7.75 times lower than that of the open-loop controller.

To further validate the velocity controller's performance, we commanded a series of step inputs with target angular velocities of -40 , -80 , and -120 deg/s. Fig. 7 shows the robot's tracking performance for all three reference velocities. In each case, the velocity controller converges to and maintains the target speed. The corresponding IAE values in the interval [6, 16] s are summarized in Table V, yielding 57.38, 51.7, and 66.7 deg·s, respectively. The rise times of the closed-loop controller for these reference speeds are 2.22, 1.41, and 1.36 s, respectively, while the open-loop case shows a rise time of 1.23 s. The rise time responses remain nearly identical across different reference speeds and control strategies, as the robot operates near full power.

C. Impulsive Control Performance Evaluation

To validate the efficacy of the impulsive momentum controller on deformable terrain, we conducted two complementary experiments. The controller was implemented with a displacement of $N = 5.75$ revolutions and a maximum angular velocity of $\omega_{\max} = -720^\circ/\text{s}$. The first experiment

TABLE VI

MEASURED ASCENT ERROR AND STANDARD DEVIATION BETWEEN SIMULATED AND EXPERIMENTAL RESULTS AT DIFFERENT INCLINE ANGLES

Incline angle (deg)	3	3.5	4	4.5	5	5.5	6
Error (mm)	-2.73	-0.97	-0.04	0.38	-0.29	-0.03	-0.19
Error (%)	-48.15	-32.77	-2.15	26.76	-24.37	-2.86	-19.79
Std (mm)	0.47	0.20	0.52	0.29	0.19	1.02	0.77

TABLE VII

DEFORMABLE TERRAIN TRAVERSAL PERFORMANCE OF MOSPHERO

	Velocity controller	Impulsive controller
Success Rate (%)	35.7	71.4
Mean Escape Time (s)	10.64	1.43
Std. Dev. of Time (s)	4.11	0.35

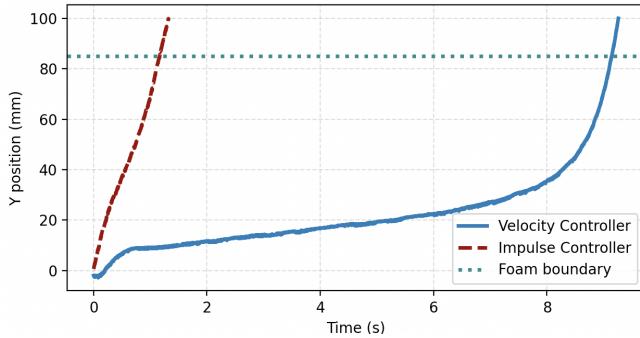


Fig. 8. Escape performance from a soft surface. The trajectories show the robot's position over time, with the foam boundary marked.

quantitatively evaluates torque generation on a variable-incline ramp, while the second demonstrates practical performance in traversing deformable terrain.

1) *Validation of Slope Ascent Capability:* To quantitatively evaluate the robot's slope ascent capability, a comparative analysis was performed between the velocity controller and the impulsive controller. The dynamic model detailed in Sec. III-B was implemented in MATLAB/Simulink to provide a theoretical baseline, which was validated against empirical data. For the velocity controller, both the simulation and experimental results consistently showed that the robot could climb slopes lower than 2.5° , while failing to ascend slopes steeper than 3° . This correspondence indicates that the dynamic model effectively captures the inherent limitations of the closed-loop control strategy.

In contrast, the impulsive controller exhibited improved performance, allowing the robot to traverse inclines that the velocity controller could not overcome. Since the velocity controller could already climb slopes up to about 2.5° , the evaluation for the impulsive control began from 3° , where the velocity controller failed. Table VI presents the ascent height error between the simulated and experimental results across various incline angles. The error is expressed both in millimeters and as a percentage relative to the experimental measurements, with the standard deviation indicating trial-to-trial variability. Noticeable error at 3.0° and 3.5° are likely

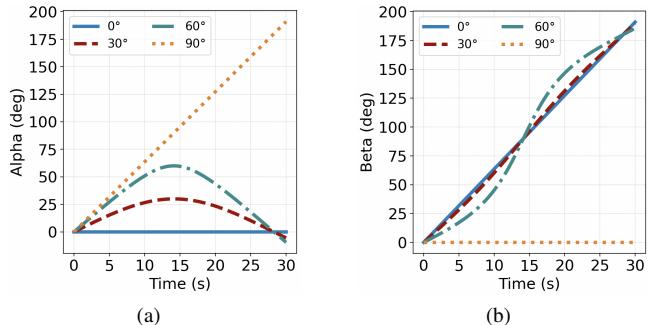


Fig. 9. Commanded motor angles for the 0° , 30° , 60° , and 90° 2D movement experiments: (a) α and (b) β motor angles.

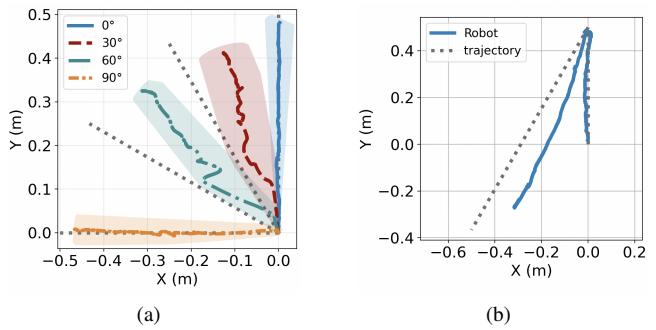


Fig. 10. Experimental results of the 2D movement experiments. (a) Robot trajectories for commanded directions at 0° , 30° , 60° , and 90° . (b) Robot trajectory of the 30° turn maneuver.

caused by rolling friction effects associated with the longer rolling distance, which were not considered in the dynamic model. The results validate the dynamic model, indicating that the simulated ascent heights capture the experimental trend across different inclines.

2) *Application in Deformable Terrain Traversal:* The second experiment was conducted to demonstrate the robot's practical utility in traversing deformable terrain. In this test, the robot was placed on an Ethylene-Vinyl Acetate foam surface with a density of 0.058 g/cm^3 and tasked with crossing a designated boundary located 84 mm away. The escape time was recorded for each control strategy. Trials in which the robot did not cross the boundary within 30 s were counted as failures. The performance difference is shown in Fig. 8 and quantified in Table VII. The impulsive controller achieved a mean escape time of 1.43 s with a success rate of 71.4%, enabled by its high torque "kick-start." The velocity controller, by contrast, required 10.64 s on average and succeeded in only 35.7% of trials due to rolling resistance. Thus, the impulsive method was over 7.4 times faster in successful trials, confirming that the higher torque observed in the slope test translates directly to improved traversal performance.

D. Two-dimensional Movement

In this section, we evaluate the robot's two-dimensional (2D) movement capability through two experiments: (i) commanding straight-line trajectories at 0° , 30° , 60° , and

90°, and (ii) executing a sharp 30° turn maneuver. By integrating inverse and motor kinematics (Sec. III-A), the commanded trajectories were translated into motor angles α and β (Fig. 9a, Fig. 9b), with the resulting motions shown in Fig. 10a. Each trajectory result represents the average of five trials, with the shaded region indicating the one standard deviation envelope. The robot exhibits strong performance in the 0° and 90° trajectories. However, due to shell deformation, the 30° and 60° trajectories deviate slightly from the intended path. Nevertheless, the results confirm the robot's omnidirectional locomotion capability, with the 90° trajectory further demonstrating its ability to perform side-rolling. Finally, in the second experiment, we tested a 30° turn maneuver (Fig. 10b). Although the trajectory deviates slightly for the same reasons noted above, the experiment shows that the robot is capable of executing sharp turns.

VI. CONCLUSIONS

In this paper, we presented the design and implementation of Mosphero, a dual-mode pendulum-driven spherical robot. The coaxial dual-pendulum with bevel gear coupling enables omnidirectional rolling, side-rolling, and sharp turns, while the combination of continuous rolling and impulsive momentum actuation allows the robot to traverse deformable terrain.

The robot's dynamics were formulated using the Lagrangian method and verified in MATLAB/Simulink, accurately predicting open-loop instability, closed-loop stabilization under PI velocity control, and slope-climbing performance trends. Experiments showed that closed-loop control reduced IAE by 87.1%, while impulsive actuation shortened escape time on deformable terrain by a factor of 7.4 and nearly doubled the success rate. 2D trajectory tests confirmed omni-directional locomotion, including side-rolling maneuvers.

Future work will extend the controller to closed-loop trajectory tracking, incorporate an enhanced motor for greater escape torque, and exploit the modular design to integrate advanced sensing and task-specific payloads.

REFERENCES

- [1] C. Li, A. O. Pullin, D. W. Haldane, H. K. Lam, R. S. Fearing, and R. J. Full, "Terradynamically streamlined shapes in animals and robots enhance traversability through densely cluttered terrain," *Bioinspiration & Biomimetics*, vol. 10, no. 4, p. 046003, Jun. 2015.
- [2] M. B. Bahar, S. S. Abdullah, M. S. Mohd Aras, M. H. Harun, M. K. Aripin, and F. Ali Ibrahim, "A study on the terrestrial mobility of a spherical amphibian robot," *IIUM Engineering Journal*, vol. 25, no. 2, pp. 309–324, Jul. 2024.
- [3] H. Sun, M. Li, J. Song, and Y. Wang, "Research on self-reconfiguration strategy of modular spherical robot," *International Journal of Advanced Robotic Systems*, vol. 19, no. 2, p. 17298806221081665, 2022.
- [4] W. Ren, Y. Wang, H. Liu, S. Jin, Y. Wang, Y. Liu, Z. Zhang, T. Hu, and G. Li, "Spherical robot: A novel robot for exploration in harsh unknown environments," *IET Cyber-Systems and Robotics*, vol. 5, no. 4, p. e12099, 2023.
- [5] A. Halme, T. Schönberg, and Y. Wang, "Motion control of a spherical mobile robot," in *Proceedings of the 4th International Workshop on Advanced Motion Control (AMC '96-MIE)*, vol. 1, 1996, pp. 259–264.
- [6] J. Alves and J. Dias, "Design and control of a spherical mobile robot," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 217, no. 6, pp. 457–467, 2003.
- [7] W. H. Chen, C. P. Chen, W. S. Yu, C. H. Lin, and P. C. Lin, "Design and implementation of an omnidirectional spherical robot omnicon," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2012, pp. 719–724.
- [8] W. H. Chen, C. P. Chen, J. S. Tsai, J. Yang, and P. C. Lin, "Design and implementation of a ball-driven omnidirectional spherical robot," *Mechanism and Machine Theory*, vol. 68, pp. 35–48, 2013.
- [9] V. Kaznov and M. Seeman, "Outdoor navigation with a spherical amphibious robot," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 5113–5118.
- [10] S. S. Ahn and Y. J. Lee, "Novel spherical robot with hybrid pendulum driving mechanism," *Advances in Mechanical Engineering*, vol. 2014, no. Article ID 687836, pp. 1–14, 2014.
- [11] R. Mukherjee, M. A. Minor, and J. T. Pukrushpan, "Simple motion planning strategies for spherobot: a spherical mobile robot," in *Proceedings of the 38th IEEE Conference on Decision and Control (CDC)*, vol. 3, 1999, pp. 2132–2137.
- [12] S. Bhattacharya and S. K. Agrawal, "Spherical rolling robot: A design and motion planning studies," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 835–839, 2000.
- [13] Q. Jia, Y. Zheng, H. Sun, H. Cao, and H. Li, "Motion control of a novel spherical robot equipped with a flywheel," in *Proceedings of the International Conference on Information and Automation (ICIA)*, 2009, pp. 893–898.
- [14] T. Aoki, S. Ito, and Y. Sei, "Development of quadruped walking robot with spherical shell: Mechanical design for rotational locomotion," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5706–5711.
- [15] S.-S. Ahn and Y.-J. Lee, "Novel spherical robot with hybrid pendulum driving mechanism," *Advances in Mechanical Engineering*, vol. 6, p. 456727, 2014.
- [16] T. Tao, M. Yang, J. Huo, H. Zhang, K. A. Neusyipin, and M. Guo, "Design of autonomous disturbance rejection speed controller for spherical robot," in *Advances in Guidance, Navigation and Control*, L. Yan, H. Duan, and X. Yu, Eds. Singapore: Springer Singapore, 2022, pp. 5097–5108.
- [17] G. Cui, H. Yang, K. Xiong, H. Zhang, and M. Guo, "Research on the trajectory tracking control of a swinging spherical robot," in *2021 6th International Conference on Control and Robotics Engineering (ICCRE)*, 2021, pp. 76–80.
- [18] R. Lin, J. Huo, X. Yang *et al.*, "Xk-iii: A spherical robot with redundant degrees of freedom," *Journal of Intelligent & Robotic Systems*, vol. 110, no. 1, p. 104, 2024.
- [19] S. Asiri, F. Khademianzadeh, A. Monadjemi, and P. Moallem, "The design and development of a dynamic model of a low-power consumption, two-pendulum spherical robot," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 5, pp. 2406–2415, 2019.
- [20] W. Ren, Y. Wang, H. Liu, S. Jin, Y. Wang, Y. Liu, Z. Zhang, T. Hu, and G. Li, "Spherical robot: A novel robot for exploration in harsh unknown environments," *IET Cyber-Systems and Robotics*, vol. 5, no. 4, p. e12099, 2023.
- [21] M. Seeman, M. Broxvall, A. Saffiotti, and P. Wide, "An autonomous spherical robot for security tasks," in *2006 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety*, 2006, pp. 51–55.
- [22] M. Roozegar and M. J. Mahjoob, "Modelling and control of a non-holonomic pendulum-driven spherical robot moving on an inclined plane: simulation and experimental results," *IET Control Theory & Applications*, vol. 11, no. 4, pp. 541–549, 2017.