Article

Orientation Control System: Enhancing Aerial Maneuvers for Quadruped Robots

Francesco Roscia [1], **Andrea Cumerlotti** [1,2], **Andrea Del Prete** [2], **Claudio Semini** [1] and **Michele Focchi** [1,3,*]

1   Dynamic Legged Systems (DLS) Lab, Istituto Italiano di Tecnologia (IIT), 16163 Genova, Italy
2   Industrial Engineering Department (DII), University of Trento, 38123 Trento, Italy
3   Department of Information Engineering and Computer Science (DISI), University of Trento, 38123 Trento, Italy
*   Correspondence: michele.focchi@unitn.it

Abstract: For legged robots, aerial motions are the only option to overpass obstacles that cannot be circumvented with standard locomotion gaits. In these cases, the robot must perform a leap to either jump onto the obstacle or fly over it. However, these movements represent a challenge, because, during the flight phase, the Center of Mass (CoM) cannot be controlled, and there is limited controllability over the orientation of the robot. This paper focuses on the latter issue and proposes an Orientation Control System (OCS), consisting of two rotating and actuated masses (flywheels or reaction wheels), to gain control authority on the orientation of the robot. Due to the conservation of angular momentum, the rotational velocity if the robot can be adjusted to steer the robot's orientation, even when the robot has no contact with the ground. The axes of rotation of the flywheels are designed to be incident, leading to a compact orientation control system that is capable of controlling both roll and pitch angles, considering the different moments of inertia in the two directions. The concept was tested by means of simulations on the robot Solo12.

Keywords: legged robot; orientation control; articulated multi-body system

1. Introduction

Legged robots are designed for traversing rough terrain. Different types of gaits, such as trot [1] or crawl [2], have been developed for quadrupedal robots. Thanks to progress over the last two decades, robots have become lighter and stronger, which has enabled them to perform with agile locomotion. However, sometimes it is not possible for the robot to move around or over an obstacle with the gaits mentioned above, and jumps are required.

When the robot is in the air, the CoM moves on a ballistic trajectory, and this is completely defined by the lift-off position and velocity. On the other hand, the base orientation can be changed so as to exploit the conservation of the system's angular momentum. This means that it is possible to control the base angular velocity by changing the inertia of the robot, e.g., changing the configuration of the joints. Nevertheless, the majority of quadrupeds are designed with light legs, resulting in limbs that have little influence on the total angular momentum.

Quadrupedal animals, like cats, can rearrange their tails and trunks to correct their orientations during a fall [3]. Much work in the field of robotics has made use of an additional link, such as a tail, as in [4,5]. This link rotates around an axis that does not pass through the robot's CoM. The distances between the axis of rotation and the CoM of both trunk and tail have a large effect on the total inertia, even with a small tail mass. However, the placement of the additional link makes the resulting robot asymmetric. Moreover, due to its limited range of motion, a tail can be used only for a single jump, not for a repeated sequence [6]. To circumvent these drawbacks, in [7–9] the authors attached a morphable inertial tail with 3 Degrees of Freedom (DoFs) (pitch, yaw, and telescoping) on a monopod, a biped and a quadruped, respectively, to enhance the agility of locomotion and to improve safety in landing.

It is possible to obtain a similar result by creating repetitive circular motions with the feet, like in [10]. Moving the feet outwards increases the robot's total inertia, so if a leg is extended during half of the motion and retracted in the other half, a net angular momentum results on the trunk, causing a rotation. In [11] the authors proposed special heavy boots for Mini Cheetah and used a neural network to calculate online joint trajectories. However, this solution unnecessarily increased the inertia of the legs, complicating the locomotion problem, which could no longer rely on the massless leg assumption.

Another option is to use a Control Moment Gyroscope (CMG). This consists of a wheel, spinning at a constant angular velocity, inside two or three actuated gimbals. Tilting the wheel's axis of rotation generates gyroscopic torque. This system is widespread in spacecraft reorientation [12], but less frequently exploited in robot locomotion, either wheeled [13] or legged [14]. The CMG presents interesting capabilities, but the presence of a pan-tilt unit to drive the gyroscope makes it impractical to mount it on a small, lightweight robot.

Flywheels represent an additional option for controlling a robot's orientation. Changing the angular velocity of a rotating mass attached to the trunk generates a torque that can reorient the robot. This device was first applied for spacecraft orientation [15], and only sporadically investigated in legged locomotion for controlling pitch orientation, both for bipeds [16,17] and quadrupeds [18,19]. Compared with tails, flywheels do not have position limits, and, since their rotational axes pass through the CoM, angular momentum results in holonomic [20]. To get a fast response, it is necessary to have an abrupt change in the flywheel's angular velocity (angular acceleration). Using a brake avoids the employment of a motor able to deliver higher torques [21], keeping the system compact. The motor slowly accelerates the wheel to a certain speed to store angular momentum and, when a reorientation is required, the break stops its spin. Since the effect of the break is unidirectional, a limitation of this approach lies in the fact that it is only possible to generate a rotation of the base in the opposite direction of the flywheel's angular velocity, making the approach unsuitable for applications where *continuous* controllability is required. On the other hand, direct-drive controlled flywheels can create accelerations in both directions, and can implement continuous control laws.

The benefits of this kind of OCS are multiple.

1. Flywheels can correct orientation errors (e.g., due to disturbances or tracking inaccuracies in the angular momentum achieved at the lift-off) during flight in a continuous manner.
2. They can track a time-varying reference, e.g., the robot can land with a desired angular velocity (possibly zero) and orientation.
3. Even in the presence of contact, the landing phase can be enhanced or dynamic gaits, such as a trot, stabilized, by significantly reducing trunk oscillations.
4. The presence of these additional joints, the only functions of which are to control orientation, provides the possibility of relieving the efforts of the legs.

In more complex scenarios, like for a somersault, limbs and OCS can operate in parallel to achieve a rotational angle larger than that achievable only with legs (e.g., due to torque limitation).

1.1. Proposed Approach and Contribution

This work introduces a compact OCS, based on two flywheels, mounted on the trunk of the lightweight quadruped Solo12 [22]. The open-source hardware and software are reported in [23]. Figure 1 shows the OCS mounted on the trunk of the robot. The contributions of the paper are the following:

*   the design of a novel OCS that enables effective control of the orientation of a legged robot during a jump, while keeping the design simple. In particular, the axes of rotation of the flywheels are set to be incident, enabling *continuous* controllability in both directions of roll and pitch, while keeping the device compact.

- simulations with the quadruped Solo12 were conducted and demonstrated the effectiveness of the proposed approach.
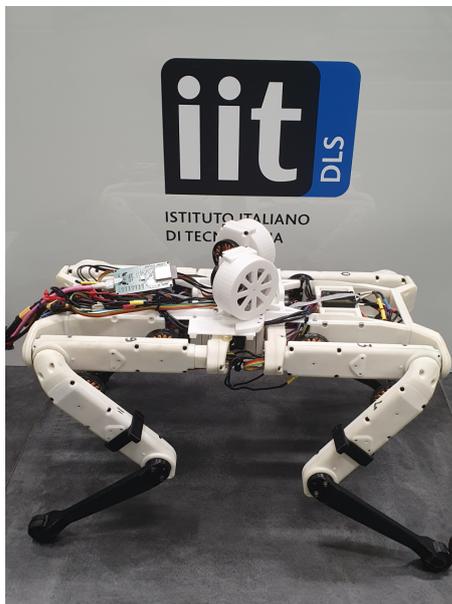

Figure 1. The proposed Orientation Control System for for the 2.5 kg quadruped robot Solo12 consists of two 0.1 kg flywheels with incident rotation axes. Each wheel is located in a 3D-printed shell and mounted on the trunk of the body.

1.2. Outline

The remainder of this paper is organized as follows. In Section 2, the law of conservation of the total angular momentum is described. In Section 3, the design principles are presented, together with the strategy for simultaneously controlling the robot's roll and pitch orientations. Section 4 presents the simulation in different scenarios that demonstrate the capability of our OCS to reject disturbances and to track angular references when there is no contact with the ground, and to dampen base oscillation after touch-down. Conclusions and a possible evolution of the work are reported in Section 5.

2. Background

The starting point for any OCS is the Euler's equation. For any mechanical system, the time derivative of the angular momentum, L computed in respect to a reference point O, fixed in an inertial frame, equals the sum of the moments M_i applied to the system with respect to the same reference point:

$$\dot{L} = \sum_i M_i \tag{1}$$

when the result of the external moments applied to the system is zero, and the Euler's equation simplifies to:

$$\dot{L} = 0 \quad \Rightarrow \quad L(t) = const, \tag{2}$$

which is known as conservation of angular momentum. Referring to legged robots, this condition occurs when the system is not in contact with the ground or with other objects, e.g., during a fall or the flight phase of a jump. In this case, it is possible to change the angular velocity of the base link, thereby changing the joint positions and velocities, as a result of the *non-holonomy* of the total angular momentum [24]. If the angular momentum of a certain body changes, then one of the others must change to maintain the total sum as constant.

Preliminary Analysis

Most quadruped robots are designed in such a way that the largest amount of the mass is located in the main body. As a consequence, the contribution of the moments due to the acceleration of other bodies (i.e., leg links) is moderate.

As an example, consider the case of Solo12. Even if each limb accounts for about 13.4% of the total mass, changes in the total inertia of the robot come with the motion of only the upper leg (thigh) and the lower leg (calf), since the displacement of distribution of the hip mass with the system's CoM is approximately constant. In view of this, each limb is responsible for only 7.5% of the total mass in varying the total inertia of the robot. Flywheels can be used to alleviate the lack of control authority.

3. Orientation Control System

In this section, a procedure to select the inertia of the flywheels is presented, together with a strategy to exploit them so as to simultaneously control the robot's roll and pitch orientations.

3.1. Bounds on the Inertia

Investigation for the selection of the flywheels' inertia can be performed with the Elroy's Beanie model, depicted in Figure 2.
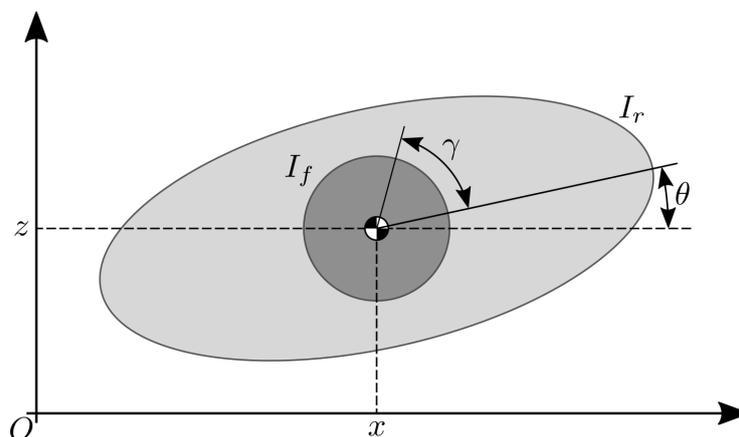

Figure 2. Schematic representation of the Elroy's Beanie model used for the preliminary analysis of the pitch motion.

This consists of two rigid bodies connected through their CoMs with a revolute joint. One of the bodies represents the robot in its nominal configuration and the other models the OCS, here represented by a single wheel for the sake of simplicity. The aim was to examine the rotational dynamics of the system as a whole. To simplify the analysis, in the following only the effects on the pitch angle of the robot was considered, keeping in mind that the same arguments also apply to the roll. Let us identify the moment of inertia of the robot in the nominal configuration as I_r and the moment of inertia of the two flywheels as I_f, both referred to the system's CoM. The angular momentum L of this system can be written as:

$$L(t) = \left(I_r + I_f\right)\dot{\theta}(t) + I_f\dot{\gamma}(t) \tag{3}$$

where $\dot{\theta}$ and $\dot{\gamma}$ are, respectively, the robot;s pitch rate and the angular speed of the wheel. The robot is driven to reach a desired pitch rate $\dot{\theta}_{des}$ by acting on the wheel speed. Without loss of generality, one can assume the flywheel is stationary with the robot at the instant at which the reorientation maneuver starts ($\dot{\gamma}_0 = 0$), having a system angular momentum of $L_0 = \left(I_r + I_f\right)\dot{\theta}_0$. Under the condition of conservation of the angular momentum, this

quantity is constant over time and it is possible to estimate the lower bound for I_f, given a desired pitch rate and the maximum velocity of the flywheels $\dot{\gamma}_{max}$:

$$I_f \geq I_r \frac{\Delta\dot{\theta}}{\Delta\dot{\theta} + \dot{\gamma}_{max}}. \tag{4}$$

where $\Delta\dot{\theta} = \left|\dot{\theta}_0 - \dot{\theta}_{des}\right|$ is the base velocity variation. Figure 3 reports the lower bound of I_f given the desired base velocity variation and the maximum velocity of the actuator.
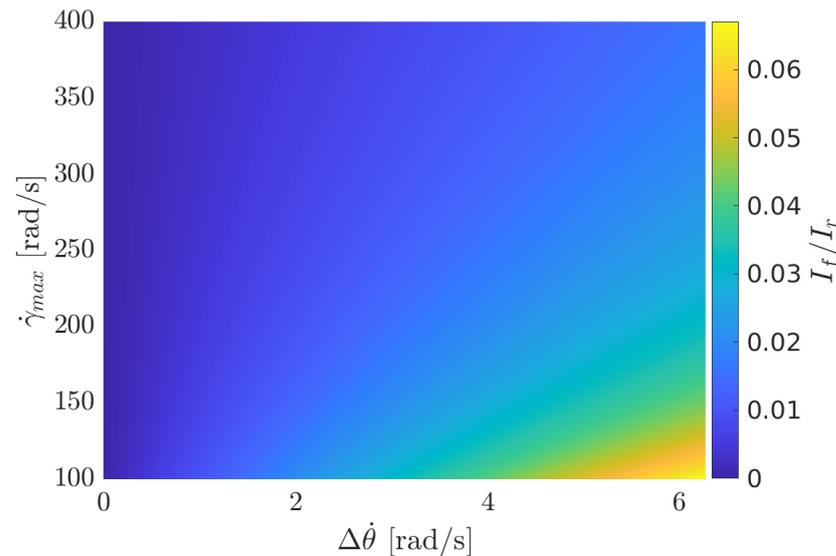

Figure 3. Minimum inertia of the OCS (normalized by the robot's inertia) necessary to achieve a base velocity variation $\Delta\dot{\theta}$, given the actuation bound $\dot{\gamma}^{max}$.

3.2. Flywheels' Axes of Rotation

The orientation of the axis of rotation of each flywheel influences their contribution to the total angular momentum. To meet the specification of controlling both the robot roll and pitch, the axes of rotation of the left and right wheels, identified in the base reference frame with the unit vectors $\hat{\boldsymbol{a}}_l$ and $\hat{\boldsymbol{a}}_r$, are set to be incident, laying on a plane parallel to the $xy-$plane of the base reference frame (see Figure 4). To remove unnecessary complications, the authors designed two identical modules, to be mounted on the trunk of Solo12, symmetrically, with respect to the robot's sagittal plane. Let denote $\alpha \leq \pi/2$ with the non-negative incident angle between the wheels' axes of rotation and the robot's lateral direction. The matrix:

$$\boldsymbol{C} = \left[\begin{array}{cc} \hat{\boldsymbol{a}}_l & \hat{\boldsymbol{a}}_r \end{array}\right] = \left[\begin{array}{cc} \sin(\alpha) & -\sin(\alpha) \\ \cos(\alpha) & \cos(\alpha) \\ 0 & 0 \end{array}\right] \tag{5}$$

maps the flywheel moments into scalar torques $\boldsymbol{u} = \left[\begin{array}{cc} \tau_{fl} & \tau_{fr} \end{array}\right]^T$ about the flywheel axes $\hat{\boldsymbol{a}}_l$ and $\hat{\boldsymbol{a}}_r$ (expressed in the base frame). As long as $\boldsymbol{C}$ is in full column rank, it is possible to control both roll and pitch angles. If $\alpha = 0$, the roll becomes uncontrollable through the OCS; otherwise, if $\alpha = \pi/2$, the pitch becomes uncontrollable. The angle is selected considering the ratio of the eigenvalues along the x and y directions of the ellipsoid of the robot's inertia(Figure 5):

$$\alpha^* = \tan^{-1}\left(\frac{I_{r,xx}}{I_{r,yy}}\right). \tag{6}$$
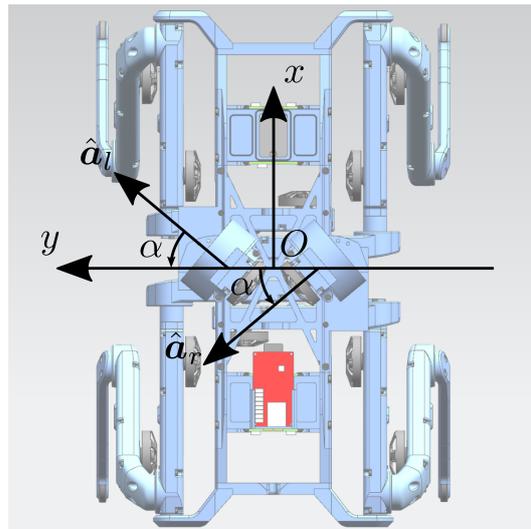
Figure 4. Top view of Solo12 with the proposed OCS. The $x-$ and $y-$axes embody the forward and left directions of the robot base, respectively. The figure also illustrates the unit vectors $\hat{a}_l$ and $\hat{a}_r$ that represent the axes of rotation of the left flywheel and of the right one and α is the angle of incidence, defined in Section 3.2.
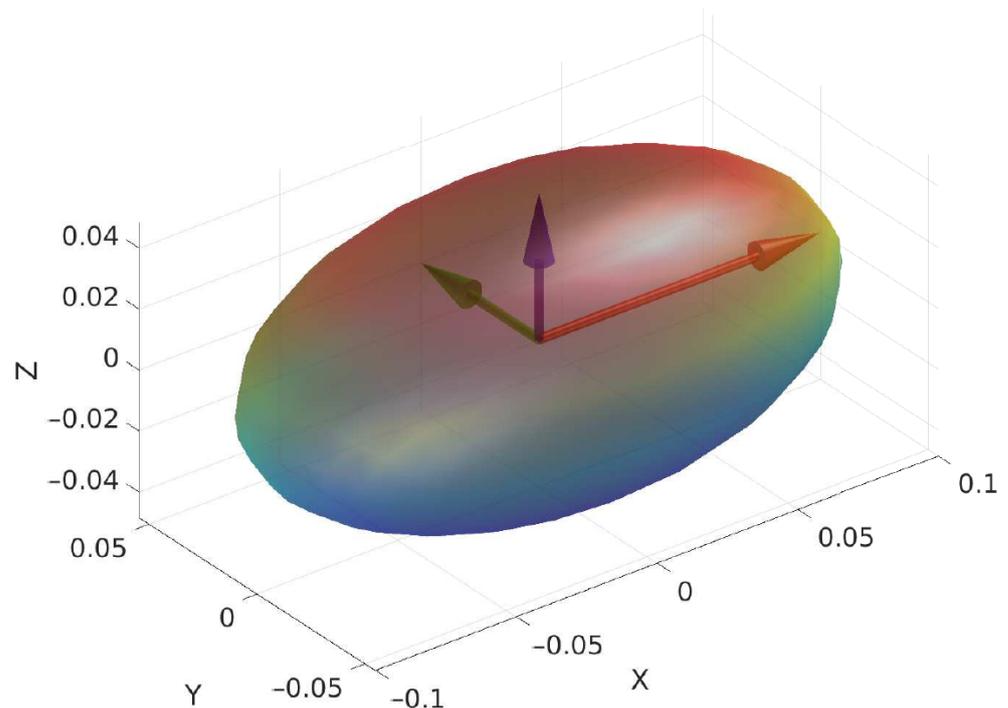

Figure 5. The inertia tensor can be seen as an ellipsoid. Its principal axes are in the direction of the eigenvectors of the tensor and their lengths depend on the eigenvalues.

In the case of Solo12, this angle is about $40°$. With these considerations in mind, the angular momentum produced by the wheels, expressed in the robot base frame, was:

$$
\begin{aligned}
{}_b\boldsymbol{L}_f &= {}_b\boldsymbol{L}_{fl} + {}_b\boldsymbol{L}_{fr} \\
&= I_{f,zz}\,\boldsymbol{\omega}_{fl} + I_{f,zz}\,\boldsymbol{\omega}_{fr} \\
&= I_{f,zz}
\begin{bmatrix}
\left(\omega_{fl} - \omega_{fr}\right)\sin(\alpha) \\
\left(\omega_{fl} + \omega_{fr}\right)\cos(\alpha) \\
0
\end{bmatrix}
\end{aligned}
\tag{7}
$$

in which $\boldsymbol{\omega}_{fl} = \omega_{fl}\,\hat{\boldsymbol{a}}_l$ and $\boldsymbol{\omega}_{fr} = \omega_{fr}\,\hat{\boldsymbol{a}}_r$ are the angular velocity vectors of the two wheels, and ω_{fl} and ω_{fr} are the angular speeds provided to each flywheel by its actuation system. The latter equation shows that the difference of the two angular speeds impacts on roll rotations, while their difference can be used to adjust the pitch, see Figure 6. Using the definition of $\boldsymbol{C}$, (7) is rewritten as:

$$\begin{aligned}
{}_b\boldsymbol{L}_f &= I_{f,zz} \begin{bmatrix} \sin(\alpha) & -\sin(\alpha) \\ \cos(\alpha) & \cos(\alpha) \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \omega_{fl} \\ \omega_{fr} \end{bmatrix} \\
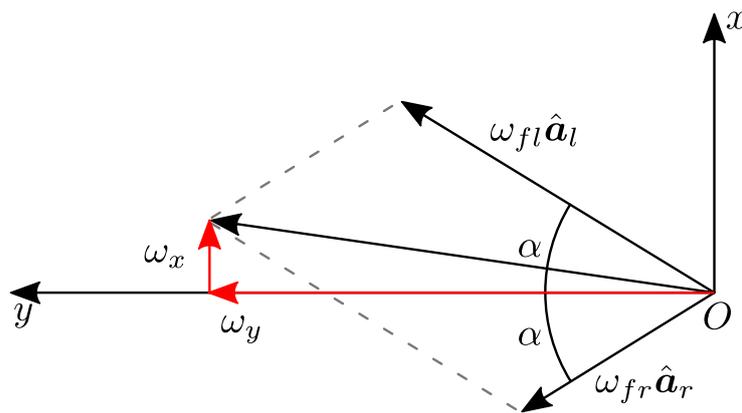&= I_{f,zz}\boldsymbol{C}\boldsymbol{\omega}_f.
\end{aligned} \tag{8}$$

Figure 6. Having incident rotation axes, the OCS allows control of both the robot's roll and pitch. Notice that the roll angle is influenced by the difference of the angular speeds of the flywheels, $\omega_x = \left(\omega_{fl} - \omega_{fr}\right)\sin(\alpha)$. On the contrary, the pitch angle depends on the sum of the angular speeds $\omega_y = \left(\omega_{fl} + \omega_{fr}\right)\cos(\alpha)$.

3.3. Inertia Selection

Once the desired inertia I_f is selected, according to the 2D Elroy's Beanie model, it is used to realize the 3D OCS (Figure 7). Two identical flywheels were designed to have the shape of hollow cylinders. This shape increases the inertia by locating the mass far away from the rotation axis. The inertia tensor expressed in its principal axes, $\boldsymbol{I}_f = \text{diag}\left\{I_{f,xx},\ I_{f,yy},\ I_{f,zz}\right\}$, depends on the cylinders' inner and outer radii, r and R, its height h and material density ρ:

$$\begin{aligned}
I_{f,xx} &= I_{f,yy} = \frac{1}{12}\pi\rho h\left(3\left(R^4 - r^4\right) + h^2\left(R^2 - r^2\right)\right) \\
I_{f,zz} &= \frac{1}{2}\pi\rho h\left(R^4 - r^4\right)
\end{aligned} \tag{9}$$

Notice that in the Elroy's Beanie model, there is a single body that models the complete OCS. The inertia introduced in the Elroy's Beanie model has to be split between the two flywheels: $I_{f,zz} = I_f/(2\cos(\alpha))$. The parameters R and h can be set to have a compact OCS and ρ depends on the chosen material, which was stainless steel in our case. The inner radius r could be adjusted to obtain the desired inertia. Spokes with negligible mass connected the wheel to the motor shaft. All the parameters are reported in Table 1, together with the selected inertia and mass of a single flywheel.
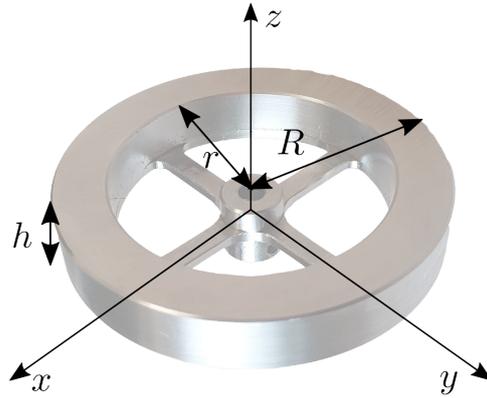
Figure 7. The final design of the flywheel, shown together with its principal axes of inertia. Here, h is the wheel thickness, r is the inner radius of the wheel and R is the radius of the outer one.

Table 1. Sizes and dynamic parameters of a single flywheel.

Parameter	Value	Unit
r	2.20×10^{-2}	m
R	3.00×10^{-2}	m
h	1.02×10^{-2}	m
ρ	$7.86 \times 10^{+3}$	kg/m^3
m	1.02×10^{-1}	kg
I_{xx}, I_{yy}	3.64×10^{-5}	$\text{kg} \cdot \text{m}^2$
I_{zz}	7.11×10^{-5}	$\text{kg} \cdot \text{m}^2$

3.4. Flywheels' Control Law

To derive a control law based on the robot's base orientation, we made use of (2), expressing all the contributions to the time derivative of total angular momentum with respect to the base reference frame:

$$_b\boldsymbol{I}_f \,_b\dot{\boldsymbol{\omega}}_f + _b\boldsymbol{I}_r \,_b\dot{\boldsymbol{\omega}}_r + _b\boldsymbol{\omega}_r \times (_b\boldsymbol{I}_r \,_b\boldsymbol{\omega}_r) = 0 \tag{10}$$

From this expression, the moment on the base caused by the acceleration of the flywheels is defined, and can be used as a feedback torque $\boldsymbol{\tau}_{fb}$:

$$\begin{aligned} \boldsymbol{\tau}_{fb} &= \boldsymbol{I}_f \dot{\boldsymbol{\omega}}_f \\ &= -_b\boldsymbol{I}_r \dot{\boldsymbol{\omega}}_b - _b\boldsymbol{\omega}_r \times (_b\boldsymbol{I}_r \,_b\boldsymbol{\omega}_r) \end{aligned} \tag{11}$$

Proportional and derivative action $\boldsymbol{K}_p \boldsymbol{e} + \boldsymbol{K}_d \dot{\boldsymbol{e}}_r$ is used, where $\boldsymbol{K}_p$ and $\boldsymbol{K}_d$ are diagonal and positive definite gain matrices for the error in attitude and angular velocity. The orientation error $\boldsymbol{e} \in SO(3)$ requires that the algebra of the special rotational group is computed. To avoid singular configurations, orientations are represented with quaternions. The derivative error can be computed using $\dot{\boldsymbol{e}} = _b\boldsymbol{\omega}_r{}^{des} - _b\boldsymbol{\omega}_r$, in which $_b\boldsymbol{\omega}_r{}^{des}$ and $_b\boldsymbol{\omega}_r$ are, respectively, the desired and actual angular velocities of the base:

$$\boldsymbol{\tau}_{fb} = -_b\boldsymbol{I}_r \left(\boldsymbol{K}_p \boldsymbol{e} + \boldsymbol{K}_d \dot{\boldsymbol{e}} \right) - _b\boldsymbol{\omega}_r \times (_b\boldsymbol{I}_r \,_b\boldsymbol{\omega}_r). \tag{12}$$

Projecting the moment $\boldsymbol{\tau}_{fb}$ onto the flywheel axes with $\boldsymbol{C}^T$, the control action $\boldsymbol{u}$ yields

$$\boldsymbol{u} = \boldsymbol{C}^T \boldsymbol{\tau}_{fb}. \tag{13}$$

4. Results of Simulations

To validate our OCS, three simulations on different scenarios are reported. The authors wanted to test the capability of the proposed approach in terms of the following: to reject a disturbance when the robot was in the flight phase of a jump, and to dampen trunk oscillations after touch-down, and to work in parallel with the joints of the legs to achieve a highly dynamic motion. All the simulations were performed using ROS [25], together with Gazebo [26]. The physics-related parameters are reported in Table 2. References for the joints of the legs were computed off-line using Crocoddyl, an optimal control library for robots based on Differential Dynamic Programming (DDP) algorithms. It uses Pinocchio for fast computation of robot dynamics and the analytical derivatives. References q_{ref}, $\dot{q}_{ref}$ and τ_{ref} for joint positions, velocities and torques were then executed on-line with a proportional-derivative joint controller:

$$\tau_j = K_{p,j}(q_{ref} - q) + K_{d,j}(\dot{q}_{ref} - \dot{q}) + \tau_{ref} \tag{14}$$

The OCS was commanded to track base orientation references using the control law introduced in Section 3.4 coded in Python. (The video with all the simulations is available at web page https://www.mdpi.com/article/10.3390/s23031234/s1, see Supplementary Material).

Table 2. Physics-related parameters used to simulate the robot dynamics in Gazebo.

Parameter	Value
Step size	0.001 s
Real time update rate	250
Physics engine	Open Dynamics Engine (ODE)
Solver	Quick (Projected Gauss-Seidel method)
Iterations	50
Successive Over Relaxation parameter	1.3
Rescaling Moment of Inertia	no
Friction model	Pyramid

The necessity of having an OCS was revealed in the first test, as disturbance occurred when the robot had no contact with the ground. During the flight phase of a forward jump, 0.1 s after the lift-off, an external disturbance moment was applied to the robot's base, deflecting the robot's orientation. This disturbance was set to $\tau_{dist} = \begin{bmatrix} 0.2 & 0.8 & 0.0 \end{bmatrix}^T \text{N} \cdot \text{m}$ and applied for 0.05 s. If the flywheels were not used, Solo12 fell after touch-down. Instead, when using the flywheels it was possible to drive the robot to a safe configuration after landing (Figure 8), without the need to implement a specific landing strategy, such as the one in [27].

The second test demonstrated the ability of the OCS to reorient the base. The robot fell from a height of 1.0 m with an initial pitch orientation of 30° and zero base angular velocity. If the flywheels were not actuated, the robot touched the ground with the same initial orientation and the trunk oscillated. If the OCS was enabled, it drove the robot's base to be horizontal when it was still in the air and the oscillations after touch-down were drastically reduced, both in the pitch angle and forward direction (Figure 9).

Finally, the ability of the flywheels to relieve the efforts of the leg joints to achieve a highly dynamic movement was shown with a back-flip (Figure 10). For this, we targeted a space application, carrying out a simulation with lunar gravity acceleration (1.62 m/s²). In this way, it was possible to obtain high jumps with a long flight phase, without having to select more powerful actuators. The leg joint trajectory computed off-line described a purely vertical jump of 1 m, having a flight phase that lasted 2 s. Right after the lift-off, the flywheels started the reorientation task of performing a back-flip, by means of a spin of 360° on the pitch. For this maneuver, the value of the incident angle α was set to 0°, since no roll rotation was required. Our hardware design allowed manual change of this value before performing the task. This simulation demonstrated that the OCS alleviated the effort

applied on the legs. Indeed, it would not have been possible to do a back-flip without the flywheels, because the legs were only used to achieve the linear motion.
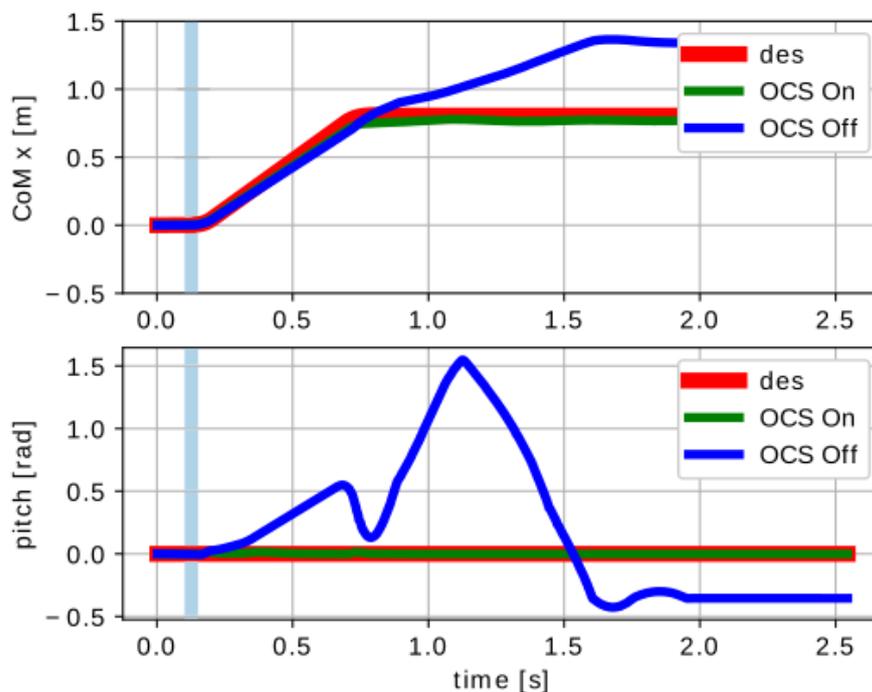
Figure 8. First Test: simulation results showing the CoM_x and pitch trajectory vs. time plots. A disturbance moment τ_{dist} on the trunk could be compensated for only if the OCS was enabled. If it was disabled, the robot was unable to restore a safe configuration after touch-down and, eventually, fell. The light blue area represents the interval of time in which the disturbance was applied.
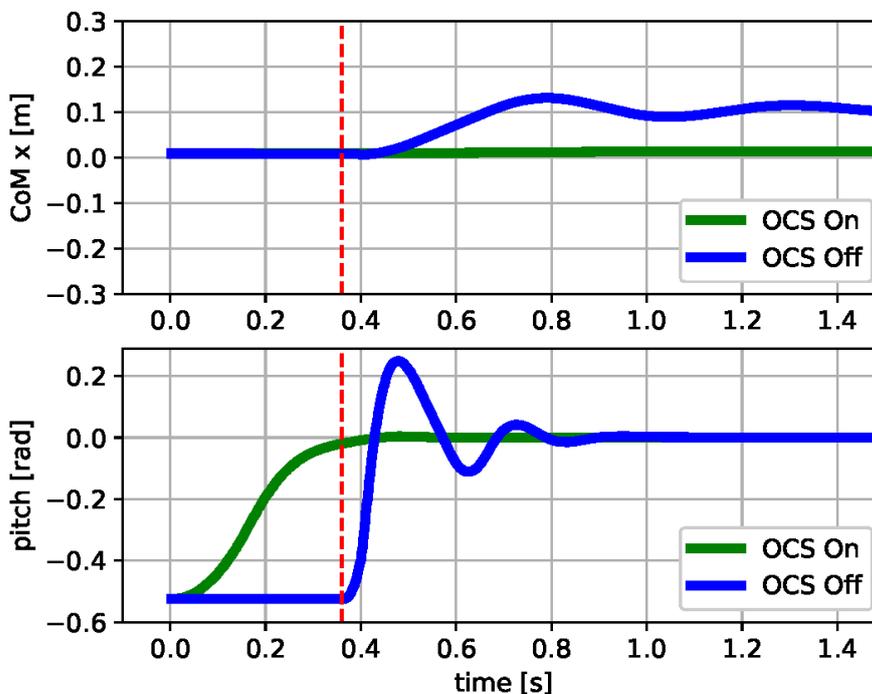
Figure 9. Second Test: simulation results showing the CoM_x and pitch trajectory vs. time plots. The OCS drove the robot's orientation during a fall so as to be horizontal. This allowed dampening of the base oscillations after touch-down (vertical dashed line), even without implementing a landing strategy.
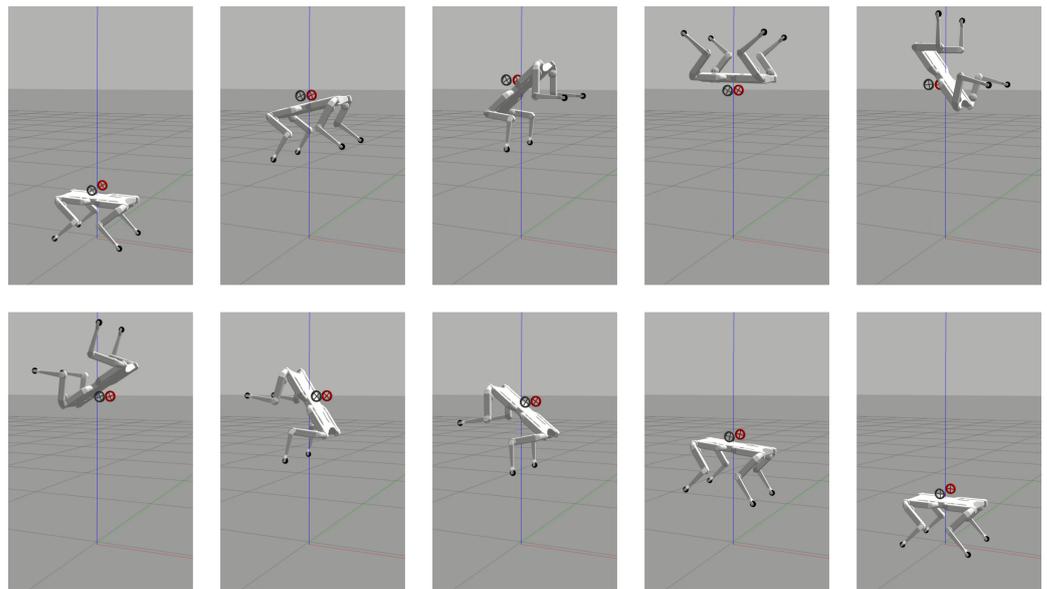
Figure 10. Third Test: sequence of snapshots of Solo12 performing a back-flip in the Gazebo environment. The OCS alleviated the effort requested of the leg joints, that, in this case, could only be used to accomplish the vertical motion. The red, green and blue lines represent the axes of the inertial (world) reference frame.

5. Conclusions

This work presented the design of a novel OCS, composed of two flywheels, that enable control of the trunk of a legged robot's platform, increasing the accuracy of aerial maneuvers during the under-actuated phases (i.e., flight phase), as well as enhancing stability when in contact by damping the oscillations. The novelty of the design, which involves flywheels attached with incident rotational axes on the trunk, is that it allows control of the orientations in both the roll and pitch directions, while keeping the device compact. The effect is optimized considering the inertial property of the mechanical structure. Several simulations were performed with the quadruped Solo12 that demonstrated the effectiveness of the proposed approach in the following: to reject disturbances during the flight phase, to stabilize the platform after touch-down, even in absence of a specific landing strategy, and to achieve a fast reorientation maneuver (a back-flip) in a reduced gravity environment.

Possible applications of the presented OCS include, but are not limited to, efficient adjustment of the posture of quadruped robots walking or jumping on uneven terrains. As proved in the third simulation (back-flip), our approach improves the capabilities of quadrupeds in space application, enabling fast locomotion by means of leaps, ensuring a reactive control action on the robot's angular momentum. Furthermore, the method presented in Section 3 in designing the OCS does not depend on a specific platform, and, thus, it can be replicated for reorienting mechanical structures with different morphologies, e.g, monopods or bipeds. A construction worker's backpack could contain two flywheels with incident rotation axes, which, in the event of a fall from scaffolding, could be used to reorient a human body for impact on the ground with an upright posture, using the same controller proposed in this work.

In future works, the authors plan to demonstrate our concept performing experiments with a real platform. The control strategy for defining the base's desired angular velocity could be improved using Nonlinear Model Predictive Control (NMPC), which takes into consideration future samples of the orientation reference. This feature is expected to allow the stabilization of the yaw to a desired value, so that, if it is not locally controllable, enhances the non-holonomy property of the angular momentum, i.e., doing a preliminary roll and pitch maneuver. A control strategy that may lead to the same result is dynamic feedback linearization, widely used for flight with quadricopters.

Abbreviations

The following abbreviations are used in this manuscript:

DoF	Degree of Freedom
CoM	Center of Mass
OCS	Orientation Control System
CMG	Control Moment Gyroscope
DDP	Differential Dynamic Programming
NMPC	Nonlinear Model Predictive Control

References

1. Barasuol, V.; Buchli, J.; Semini, C.; Frigerio, M.; De Pieri, E.R.; Caldwell, D.G. A reactive controller framework for quadrupedal locomotion on challenging terrain. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 2554–2561.
2. Focchi, M.; Orsolino, R.; Camurri, M.; Barasuol, V.; Mastalli, C.; Caldwell, D.G.; Semini, C. Heuristic planning for rough terrain locomotion in presence of external disturbances and variable perception quality. In *Advances in Robotics Research: From Lab to Market*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 165–209.
3. Kane, T.; Scher, M. A dynamical explanation of the falling cat phenomenon. *Int. J. Solids Struct.* **1969**, *5*, 663–670. [CrossRef]
4. Chu, X.; Lo, C.H.D.; Ma, C.; Au, K.W.S. Null-Space-Avoidance-Based Orientation Control Framework for Underactuated, Tail-Inspired Robotic Systems in Flight Phase. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3916–3923. [CrossRef]
5. Wenger, G.; De, A.; Koditschek, D.E. Frontal plane stabilization and hopping with a 2DOF tail. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Republic of Korea, 9–14 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 567–573.
6. Johnson, A.M.; Libby, T.; Chang-Siu, E.; Tomizuka, M.; Full, R.J.; Koditschek, D.E. Tail assisted dynamic self righting. In *Adaptive Mobile Robotics*; World Scientific: Singapore, 2012; pp. 611–620.
7. An, J.; Ma, X.; Lo, C.H.D.; Ng, W.; Chu, X.; Au, K.W.S. Design and Experimental Validation of a Monopod Robot with 3-DoF Morphable Inertial Tail for Somersault. *IEEE/ASME Trans. Mechatron.* **2022**, *7*, 5072–5083. [CrossRef]
8. An, J.; Chung, T.; Lo, C.H.D.; Ma, C.; Chu, X.; Au, K.S. Development of a bipedal hopping robot with morphable inertial tail for agile locomotion. In Proceedings of the 2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob), New York City, NY, USA, 29 November–1 December 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 132–139.
9. Tang, Y.; An, J.; Chu, X.; Wang, S.; Wong, C.Y.; Au, K. Towards Safe Landing of Falling Quadruped Robots Using a 3-DoF Morphable Inertial Tail. *arXiv* **2022**, arXiv:2209.15337.
10. Hoffman, E.M.; Paolillo, A. Exploiting visual servoing and centroidal momentum for whole-body motion control of humanoid robots in absence of contacts and gravity. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 2979–2985.

11. Kurtz, V.; Li, H.; Wensing, P.M.; Lin, H. Mini cheetah, the falling cat: A case study in machine learning and trajectory optimization for robot acrobatics. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 4635–4641.
12. Yoon, H.; Tsiotras, P. Spacecraft adaptive attitude and power tracking with variable speed control moment gyroscopes. *J. Guid. Control. Dyn.* **2002**, *25*, 1081–1090. [CrossRef]
13. Brown, H.B.; Xu, Y. A single-wheel, gyroscopically stabilized robot. In Proceedings of the IEEE International Conference on Robotics and Automation, Minneapolis, MN, USA, 22–28 April 1996; IEEE: Piscataway, NJ, USA, 1996; Volume 4; pp. 3658–3663.
14. Mikhalkov, N.; Prutskiy, A.; Sechenev, S.; Kazakov, D.; Simulin, A.; Sokolov, D.; Ryadchikov, I. Gyrubot: Nonanthropomorphic stabilization for a biped. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), , Xi'an, China, 30 May–5 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 4976–4982.
15. Oland, E.; Schlanbusch, R. Reaction wheel design for cubesats. In Proceedings of the 2009 4th International Conference on Recent Advances in Space Technologies, Istanbul, Turkey, 11–13 June 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 778–783.
16. Brown, T.L.; Schmiedeler, J.P. Reaction Wheel Actuation for Improving Planar Biped Walking Efficiency. *IEEE Trans. Robot.* **2016**, *32*, 1290–1297. [CrossRef]
17. Xiong, X.; Ames, A.D. Sequential motion planning for bipedal somersault via flywheel slip and momentum transmission with task space control. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 3510–3517.
18. Kolvenbach, H.; Hampp, E.; Barton, P.; Zenkl, R.; Hutter, M. Towards jumping locomotion for quadruped robots on the moon. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 5459–5466.
19. Vasilopoulos, V.; Machairas, K.; Papadopoulos, E. Quadruped pronking on compliant terrains using a reaction wheel. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 3590–3595.
20. Machairas, K.; Papadopoulos, E. On quadruped attitude dynamics and control using reaction wheels and tails. In Proceedings of the 2015 European Control Conference (ECC), Linz, Austria, 15–17 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 753–758.
21. Gajamohan, M.; Merz, M.; Thommen, I.; D'Andrea, R. The cubli: A cube that can jump up and balance. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 3722–3727.
22. Grimminger, F.; Meduri, A.; Khadiv, M.; Viereck, J.; Wüthrich, M.; Naveau, M.; Berenz, V.; Heim, S.; Widmaier, F.; Flayols, T.; et al. An open torque-controlled modular robot architecture for legged locomotion research. *IEEE Robot. Autom. Lett.* **2020**, *5*, 3650–3657. [CrossRef]
23. Open Dynamic Robot Initiative (ODRI): Max-Planck Institute for Intelligent System, New York University's Tandon School of Engineering, LAAS/CNRS. Available online: https://open-dynamic-robot-initiative.github.io/ (accessed on 5 December 2022).
24. Wieber, P.B.; Tedrake, R.; Kuindersma, S. Modeling and control of legged robots. In *Springer Handbook of Robotics*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 1203–1234.
25. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A.Y. ROS: An Open-Source Robot Operating System. In Proceedings of the ICRA Workshop Open Source Softw, Kobe, Japan, 12–17 May 2009; Volume 5, p. 5.
26. Koenig, N.; Howard, A. Design and use paradigms for gazebo, an open-source multi-robot simulator. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No. 04CH37566), Sendai, Japan, 28 September–2 October 2004; IEEE: Piscataway, NJ, USA, 2004; Volume 3, pp. 2149–2154.
27. Jeon, S.H.; Kim, S.; Kim, D. Real-time Optimal Landing Control of the MIT Mini Cheetah. *arXiv* **2021**, arXiv:2110.02799.