2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Heading Control for Obstacle Avoidance using Dynamic Posture Manipulation during Tumbling Locomotion

Adarsh Salagame[1], Kruthika Gangaraju[1], Eric Sihite[2], Gunar Schirner[1], Alireza Ramezani[1*]

***Abstract*— Passive tumbling structures are energy efficient, but often sacrifice control authority due to their under actuated nature. Unlike many passive tumbling robots, Northeastern University's COBRA is a snake robot with eleven articulated joints that transforms into a wheel-like structure with a high degree of posture control during tumbling, and using this posture manipulation, COBRA can control its forward velocity and heading angle while tumbling. This paper presents a mathematical framework that describes the dynamics of posture manipulation during tumbling and identifies two types of control actions that allow it to control its movement. This is validated in hardware testing to demonstrate obstacle avoidance during passive tumbling using only posture manipulation.**

I. Introduction

Navigating challenging terrain poses a hurdle for mobile robots. Legged robots that dynamically interact with their surroundings to adjust their center of mass show great promise [1]–[3] in this regard. Unlike traditional wheeled systems with fixed points of contact, legged robots can select their contact locations to navigate diverse environments. However, steep inclines present unique obstacles that demand further investigation [4].

Terrestrial and extraterrestrial rugged landscapes have scientific significance [5], [6], however traversing them is a challenge. The lunar terrain, for example, is marked by craters stretching over tens of kilometers in diameter, with slopes as steep as tens of degrees covered in porous regolith (lunar topsoil) that makes locomotion challenging [7].

In 2022, NASA's Innovative Advanced Concepts (NIAC) Program invited academic institutions in the United States to participate in the Breakthrough, Innovative, and Game-changing (BIG) Idea competition, seeking novel mobility systems capable of traversing lunar craters [8], [9]. Northeastern University secured NASA's prestigious Artemis Award with their proposal of COBRA (Crater Observing Bio-inspired Rolling Articulator) [10], an articulated robot tumbler showcased in Fig. 1. Open loop tumbling locomotion was successfully demonstrated on the steep slopes of hills near Pasadena, where the competition was held. This accomplishment led to the research question at hand, centered on active posture manipulation and heading angle steering for closed-loop control.

[1]This author is with the Department of Electrical and Computer Engineering, Northeastern University, Boston MA salagame.a, gangaraju.k, G.Schirner, *a.ramezani@northeastern.edu

[2]This author is with California Institute of Technology, Pasadena CA esihite@caltech.edu

*Indicates corresponding author.

Fig. 1. Illustrates a frontal view of COBRA [11]–[13] engaged in tumbling while navigating around an obstacle using posture manipulation.

This study develops a mathematical framework for tumbling locomotion and explores control methodologies for tracking. We first present rigorous mathematical models of tumbling, followed by experimental validation. Finally, we employ an optimization framework to derive solutions for heading angle control and demonstrate path tracking for obstacle avoidance in experiments.

Tumbling robots have gained attention in the robotics community for their energy-efficient design and adaptability in challenging environments. Passive systems like NASA/JPL's Mars Tumbleweed Rover [14] are valued for their minimal energy consumption, making them ideal for remote exploration, while active rolling robots such as MIT's Kickbot [15] offer robust, omnidirectional mobility, benefiting from a low center of gravity and the ability to traverse uneven terrain. The spherical nature of these robots also allows for excellent maneuverability in confined spaces.

Despite these advantages, tumbling locomotion presents challenges, particularly in control design. Passive rolling systems, while efficient, often sacrifice controllability, relying on morphology or posture to maneuver. Additionally, the use of the entire body for movement complicates sensor placement and localization.

979-8-3503-7769-9/24/$31.00 ©2024 IEEE

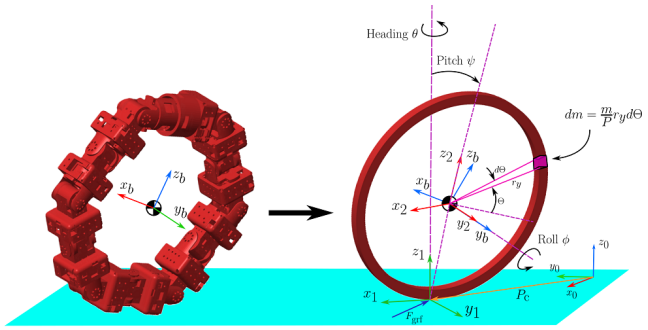

Fig. 2. Illustrates model parameters, coordinate frames, etc., used to derive Eqs. 1, 7, 22, and 23.

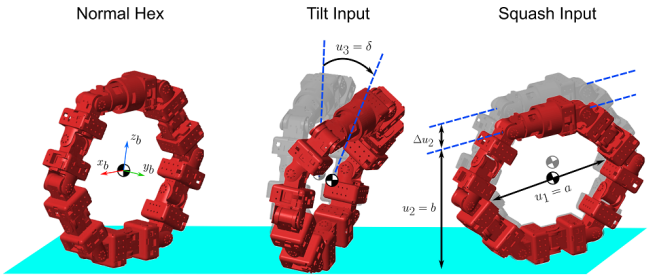

Fig. 3. Illustrates posture manipulation by considering two imaginary actuators, denoted as u_1 and u_2, which act along the principal axes of the ring to induce planar deformations and one imaginary actuator denoted u_3 to induce off-plane deformation.

Early designs like Rollo [16] and SMR [17] used a spherical shell with a spring-mass system for motion, but faced issues with maintaining contact between the driving wheel and sphere. Later systems, such as Sphericle [18] and Festo's SRR [19], improved upon these designs with car-driven mechanisms, though they remained unstable on rough terrain.

Weight-shifting mechanisms, like those in Spherobot [20] and August Robot [21], offer better control but require additional weights, reducing energy efficiency. More recent approaches using deformable structures, such as Ritsumeikan's Deformable Robot [22] and Ourobot [23], provide a more efficient way to control the center of mass.

Northeastern University's COBRA builds on this concept, featuring a morphing body for multi-modal locomotion, including tumbling, slithering, and sidewinding. COBRA's unique head-tail locking mechanism enhances its ability to tumble dynamically across rough surfaces, while its active posture control allows for two-dimensional steering. This paper focuses on COBRA's tumbling locomotion and presents an overview of the head-tail locking mechanism.

Brief Overview of COBRA Hardware

As shown in Fig. 1, COBRA has eleven actuated joints. The head module houses the computing system, radio antenna for lunar communication, and an IMU for navigation, while the tail contains an interchangeable payload. The ten identical joint modules between them each include a joint actuator and battery.

A latch mechanism in the head connects with a passive element in the tail, forming a closed-loop tumbling structure. COBRA can adjust its posture within this loop using its actuated joints to control tumbling motion, which is tested here for heading regulation during tumbling.

II. Reduced-Order Model (ROM) Derivations

In the tumbling configuration, COBRA forms an elliptical ring, as shown in Fig. 2. Our analysis is based on the following assumptions: (1) The ring has a negligible cross-sectional area (Fig. 2); (2) The mass is uniformly distributed; (3) The shape is defined by the principal axes, u_1 and u_2, controlled by the robot's joints (Fig. 2); and (4) The postures are symmetric, ensuring the ellipse's center aligns with the center of mass (CoM).

Using these assumptions, we derive the ring's inertia tensor, $\mathcal{I}(u_i)$, where u_i adjusts the ring's shape. This tensor is then used to derive the nonholonomic equations of motion for the rolling ring, capturing how control actions affect its posture.

A. Cascade Model

In our approach to mathematically formalizing tumbling servoing through shape manipulations in the ring, we introduce a cascade nonlinear model structured as follows:

$$\Sigma_{tbl} : \dot{x} = f(x, y)$$
$$\Sigma_{pos} : \begin{cases} \dot{\xi} = f_\xi(\xi, u) \\ y = h_\xi(\xi) \end{cases} \tag{1}$$

Here, x and y represent the state vector and output function, which include the ring's orientation (via Euler angles), CoM position, and the mass moment of inertia about its body axes (x-y-z). The function $f(.)$ governs the state dynamics.

The terms f_ξ and h_ξ represent the dynamics of the internal states ξ (see Section II). The control input u applies actuation along the ring's principal axes, as shown in Figs. 3 and 2. The cascade model separates the dynamics into tumbling (Σ_{tbl}) and posture manipulation (Σ_{pos}), where tumbling is controlled by internal posture adjustments to maintain the robot's shape in the inertial frame. The following subsections derive the equations for these models.

B. Governing Dynamics for Posture Manipulation Σ_{pos}

As shown in Fig. 2, the control actions $u = [u_1, u_2]^\top$ adjust the principal axes for the ring. This section derives the governing equations for posture dynamics.

While no closed-form equation exists for the perimeter of an ellipse, several approximations exist that are sufficiently accurate. One such approximation is employed in this work to maintain the perimeter at a fixed value as the length of the axes changes. To calculate the inertia tensor of the line mass, we use a simpler integral-based method, computable through numerical integration.

Consider the general equation of the center line in the ring depicted in Fig. 3 in the x-z plane of the body frame with

principal axes of length a and b,

$$\frac{p_{i,x}^2}{a^2} + \frac{p_{i,z}^2}{b^2} = 1 \tag{2}$$

where $p_i = [p_{i,x}, 0, p_{i,z}]^\top$ denotes the body-frame coordinates of a point on the ring. Consider the following change of variables:

$$\begin{aligned} \xi_1 &= r_y C_\theta \\ \xi_2 &= r_y S_\theta \end{aligned} \tag{3}$$

where r_y and θ are polar coordinates and are shown in Fig. 3. S_θ and C_θ denote $\sin\theta$ and $\cos\theta$. We take the time-derivative of the equation above and Eq. 2, which yields

$$\begin{aligned} \dot{\xi}_1 &= \dot{r}_y C_\theta - \xi_2 \dot{\theta} \\ \dot{\xi}_2 &= \dot{r}_y S_\theta + \xi_1 \dot{\theta} \\ \frac{\xi_1 \dot{\xi}_1}{a^2} &- \frac{2\xi_1^2 u_1}{a^3} + \frac{\xi_2 \dot{\xi}_2}{b^2} - \frac{2\xi_2^2 u_2}{b^3} = 0 \end{aligned} \tag{4}$$

where $\dot{a} = u_1$ and $\dot{b} = u_2$. The perimeter of the ring is fixed and given by the following equation

$$P = \int_0^{2\pi} \sqrt{a^2 C_\theta^2 + b^2 S_\theta^2}\, d\theta \tag{5}$$

therefore, we can write the following relationship between $\dot{P}$ and $\dot{\theta}$

$$\dot{P} = \left(\sqrt{a^2 C_\theta^2 + b^2 S_\theta^2} \right) \dot{\theta} = 0 \tag{6}$$

This equation constitutes the remaining ordinary differential equations necessary to establish the state-space model for the posture dynamics. By defining $\xi_3 = r_y$, $\xi_4 = \theta$, $\xi_5 = a$, and $\xi_6 = b$, and considering Eqs. 2, 3, 4, 6, the state-space model governing the state vector $\xi = [\xi_1, \ldots, \xi_6]^\top$ is given by

$$\begin{bmatrix} 1 & 0 & -C_{\xi_4} & \xi_2 & 0 & 0 \\ 1 & 0 & -S_{\xi_4} & \xi_1 & 0 & 0 \\ \frac{\xi_1}{\xi_5^2} & \frac{\xi_2}{\xi_6^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \gamma(\xi) & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \\ \dot{\xi}_4 \\ \dot{\xi}_5 \\ \dot{\xi}_6 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{2\xi_1^2}{\xi_5^3} & \frac{2\xi_2^2}{\xi_6^3} \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{7}$$

where $\gamma(\xi) = \sqrt{\xi_5^2 C_{\xi_4}^2 + \xi_6^2 S_{\xi_4}^2}$. The matrix in the left-hand side of Eq. 7 is invertible, and therefore, the normal form $\dot{\xi} = f_\xi(\xi, u)$ can be obtained, which is skipped here. Now, it is possible to show that the mass moments of inertia about the body-frame x, y, and z axes, denoted by $\mathcal{I}_{xx}$, $\mathcal{I}_{yy}$, and $\mathcal{I}_{zz}$, are functions of the hidden state vector ξ.

The mass of the differential element on the ring can be calculated assuming uniform distribution as follows:

$$dm = \frac{m}{P} dP = \frac{m}{P} \gamma(\xi) d\xi_4 \tag{8}$$

where m is the total mass of the elliptical ring. Thus, the mass moment of inertia around each body frame axis can be obtained by:

$$\begin{aligned} \mathcal{I}_{kk} &= \frac{m}{P} \int_{\xi_4} r_k^2 \gamma(\xi) d\xi_4, \quad k \in \{x, y, z\} \\ r_x &= \xi_3 C_{\xi_4} \\ r_y &= \xi_3 \\ r_z &= \xi_3 S_{\xi_4} \end{aligned} \tag{9}$$

In the equation above, the output function $y = h_\xi(\xi) = [\mathcal{I}_{xx}, \mathcal{I}_{yy}, \mathcal{I}_{zz}]^\top$ encapsulates the mass moments of inertia. Next, we will derive the equations of motion for the tumbling ring using these posture dynamics as follows.

C. Governing Dynamics for Tumbling Σ_{tbl}

Consider the ring in Fig. 2 equipped with virtual actuators u_i along its principal axes for posture control. We define the following frames of reference: (1) the world frame x_0-y_0-z_0; (2) the contact frame x_1-y_1-z_1 at the contact point p_c, with the z-axis perpendicular to the ground; (3) the gimbal frame x_2-y_2-z_2 at the CoM p_{cm}, inert to the body's motion; and (4) the body frame x_b-y_b-z_b at the CoM, rotating with the ring.

An inclination is introduced between the contact and world frames, forming an inclined plane at an angle α. The ring's orientation R_b^0, described in roll, pitch, and yaw angles θ, ψ, and ϕ, is parameterized as follows:

$$R_b^0 = R_z(\theta) R_y(\phi) R_x(\psi) \tag{10}$$

The angular velocity vector in the body frame $\boldsymbol{\omega_b} = [\omega_{b,x}, \omega_{b,y}, \omega_{b,z}]^\top$, in terms of $\dot{\theta}$, $\dot{\psi}$, and $\dot{\phi}$, is expressed as:

$$\begin{aligned} \omega_{b,x} &= \dot{\psi} \sin(\theta) \sin(\phi) + \dot{\theta} \cos(\phi) \\ \omega_{b,y} &= \dot{\psi} \sin(\theta) \cos(\phi) - \dot{\theta} \sin(\phi) \\ \omega_{b,z} &= \dot{\psi} \cos(\theta) + \dot{\phi} \end{aligned} \tag{11}$$

From the Σ_{pos} model, the ring's principal moments of inertia are denoted as y_1, y_2, and y_3. The angular momentum of the ring about p_{cm} is represented by

$$\boldsymbol{H_b} = \begin{bmatrix} y_1 & 0 & 0 \\ 0 & y_2 & 0 \\ 0 & 0 & y_3 \end{bmatrix} \boldsymbol{\omega_b} \tag{12}$$

We define the radius of rotation as the vector extending from p_{cm} to p_c.

Since the ring is in pure rolling at the contact point p_c, three constraints must be considered, including one holonomic constraint ($v_{c,z} = 0$) and two nonholonomic constraints ($v_{c,x} = 0$) and ($v_{c,y} = 0$), where $\boldsymbol{v_c} = [v_{c,x}, v_{c,y}, v_{c,z}]^\top$ denotes the contact velocity.

We formulate the equations of motion by resolving the linear and angular momentum balances concerning the ring's

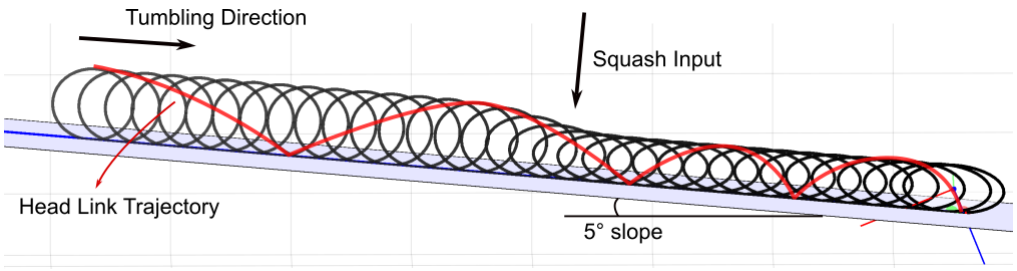

Fig. 4. Shows the overlaid snapshots of the tumbling ring squashed along its body axes.

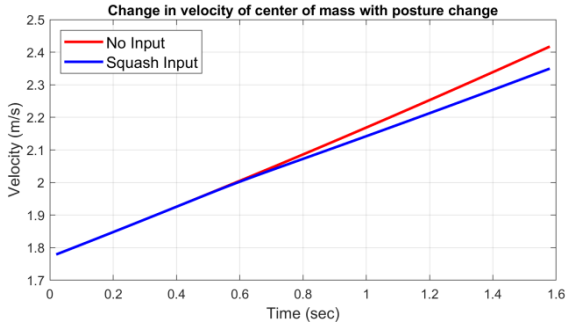

Fig. 5. Depicts the change in the tumbling velocity due to the posture manipulation in the cascade model given by Eq. 1.

CoM. The resulting equations, derived from applying the balance laws alongside the non-integrable constraints, constitute a set of differential equations describing the ring's orientation and the lateral translation of its CoM over time.

This system of equations is expressed in first-order form

$$\dot{x} = f(x, y) = M^{-1}(x, y)N(x, y) \tag{13}$$

for numerical integration in MATLAB where the nonlinear terms $M(.)$ and $N(.)$ are given in the Appendix Section. The state vector is represented as $x = [\theta, \psi, \phi, \dot{\theta}, \dot{\psi}, \dot{\phi}, p_{c,x}, p_{c,y}]^\top$. Subsequently, we propose a controller based on the collocation method to utilize u for steering the heading angle of the tumbling ring.

III. Controls

To tackle this control problem, involving the regulation of the ring's heading angle within the tumbling dynamics Σ_{tbl} using the ring's posture dynamics Σ_{pos}, we introduce the following cost function:

$$J = \|\theta - \theta_d\|_2 \tag{14}$$

where θ_d denotes the desired heading angle. This cost function J is evaluated within the framework provided by the cascade model specified in Eq. 1 to find open-loop trajectories.

We discretize Eq. 1 temporally, dividing it into n discrete time intervals $(t_i,\ i = 1, \ldots, n, \quad 0 \leq t_i \leq t_f)$ to yield the following system of equations:

$$\dot{x}_i = f_i(x_i, y_i), \quad i = 1, \ldots, n, \quad 0 \leq t_i \leq t_f \tag{15}$$

Here, x_i encapsulates the values of the state vector x at the i-th discrete time, while y_i represents the principal mass moment of inertia of the ring at the i-th discrete time. f_i denotes the governing dynamics of tumbling at the i-th discrete time.

We organize all discrete values from x_i and y_i into vectors $X = \left[x_1^\top(t_1), \ldots, x_n^\top(t_n)\right]^\top$ and $Y = \left[y_1^\top(t_1), \ldots, y_n^\top(t_n)\right]^\top$, respectively.

To ensure continuity, we impose $2n$ boundary conditions at the boundaries of n discrete periods, given by:

$$r_i\left(x(0), x\left(t_f\right), t_f\right) = 0, \quad i = 1, \ldots, 2n \tag{16}$$

Considering the three entries in y, we establish three inequality constraints to maintain the principal mass moment of inertia within bounds. These constraints, denoted as g_i, are defined as follows:

$$g_i(x(t_i), y(t_i), t_i) \geq 0, \quad i = 1, 2, 3, \quad 0 \leq t_i \leq t_f \tag{17}$$

To approximate the nonlinear dynamics of the tumbling ring, we employ a method based on polynomial interpolations, greatly simplifying computational efforts. Let's consider the n time intervals, defined as follows:

$$0 = t_1 < t_2 < \ldots < t_n = t_f \tag{18}$$

We stack the states x_i and the principal mass moment of inertia terms y_i from the ring at these discrete times into a single vector denoted as $\mathcal{Y}$. Additionally, we append the final discrete time t_f as the last entry of $\mathcal{Y}$, allowing the optimizer to determine the tumbling speed.

$$\mathcal{Y} = \left[x_1^\top, \ldots, x_n^\top, y_1^\top, \ldots, y_n^\top, t_f\right]^\top \tag{19}$$

We approximate the output function $y_i(t_i)$ at time $t_i \leq t < t_{i+1}$ using the linear interpolation function $\tilde{y}$ between $y_i(t_i)$ and $y_{i+1}(t_{i+1})$, given by

$$\tilde{y} = y_i\left(t_i\right) + \frac{t - t_i}{t_{i+1} - t_i}\left(y_{i+1}\left(t_{i+1}\right) - y_i\left(t_i\right)\right) \tag{20}$$

Similarly, we interpolate the states $x_i(t_i)$ and $x_{i+1}(t_{i+1})$ using a nonlinear cubic interpolation, ensuring continuity. This interpolation function, denoted as $\tilde{x}$, is determined by

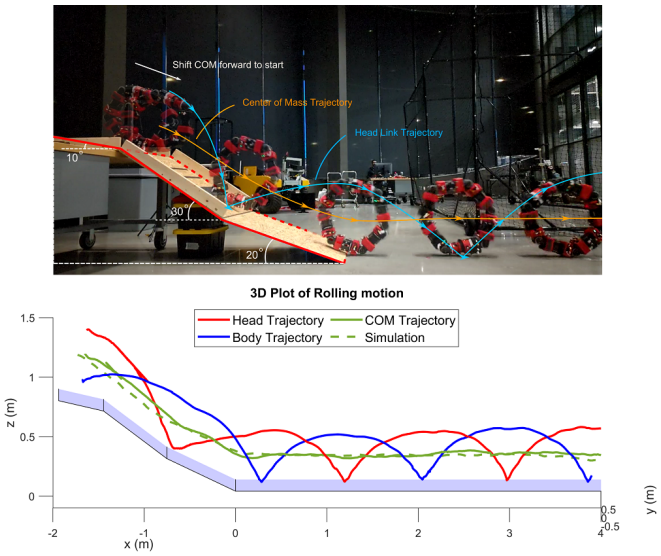

Fig. 6. Top figure Shows the experimental setup. Bottom figure depicts high fidelity model predictions of the tumbling system behavior.

the following system of equations:

$$\begin{aligned}
\tilde{x}(t) &= \sum_{k=0}^{3} c_k^j \left(\frac{t-t_j}{h_j}\right)^k, \quad t_j \le t < t_{j+1}, \\
c_0^j &= x\left(t_j\right), \\
c_1^j &= h_j f_j, \\
c_2^j &= -3x\left(t_j\right) - 2h_j f_j + 3x\left(t_{j+1}\right) - h_j f_{j+1}, \\
c_3^j &= 2x\left(t_j\right) + h_j f_j - 2x\left(t_{j+1}\right) + h_j f_{j+1}, \\
\text{where } f_j &:= f\left(x\left(t_j\right), y\left(t_j\right)\right), \quad h_j := t_{j+1} - t_j.
\end{aligned} \tag{21}$$

This cubic interpolation $\tilde{x}$ for x ensures continuity at discrete points and at the midpoint of sample times. Thus, the only remaining constraints in the nonlinear programming problem are the collocation constraints at the midpoint of $t_i - t_{i+1}$ time intervals, the inequality constraints at t_i, and the constraints at t_1 and t_f, all of which are handled within the optimization process. We address this optimization problem using MATLAB's fmincon function.

IV. Results

The dynamics framework was implemented in MATLAB for planar posture manipulation. Figure 4 shows simulation snapshots where an initial angular velocity of 5 rad/s is applied, followed by a reduction of the ellipse's axis from 0.6 m to 0.5 m at 0.5 seconds, adjusting u_1 to maintain a 2 m perimeter. The corresponding velocity change due to inertia variation is plotted in Fig. 5. Hardware implementation of this actuation mode, requiring a rotation-invariant shape, necessitates closed-loop control and precise contact location estimation.

Steering control via the tilt input u_3 was tested in both high-fidelity Simulink simulations and hardware experiments. Figure 6 shows the experimental setup, where the robot is released down a ramp and tracked with motion capture cameras. The resulting trajectory is compared with the simulation (Fig. 6).

Fig. 7. Snapshots demonstrating obstacle (box) avoidance using posture manipulation in COBRA.

To test steering control, we place an obstacle in the path of the robot and use posture manipulation with a fixed tilt input of 20° to change the heading and avoid the obstacle. Figure 7 shows snapshots of the robot avoiding the obstacle. Figure 8 shows the trajectory, with the robot successfully avoiding the obstacle and returning to its path after successive tilt actuation to -20° and back to $+20^\circ$.

V. Concluding Remarks

This study presents a mathematical framework for controlling COBRA's tumbling velocity and heading. Using a cascade model in a collocation framework, we determine how to compress or tilt the robot to achieve desired heading angles. Our experiments show that compression modulates speed, allowing acceleration or deceleration, while off-plane tilts enable active steering. These results are validated with successful open-loop demonstrations. Future work will refine this framework with closed-loop control for posture manipulation, incorporating position and orientation estimation.

Appendix

The nonlinear term M is given by

$$M(x,y) = \left[\begin{array}{ccc|c} y_1 S_\theta & 0 & 0 & \\ 0 & y_2 + mr^2 & 0 & \mathbf{0} \\ \left(y_3 + mr^2\right) C_\theta & 0 & y_3 + mr^2 & \\ \hline & \mathbf{0} & & \mathbf{I} \end{array}\right] \tag{22}$$

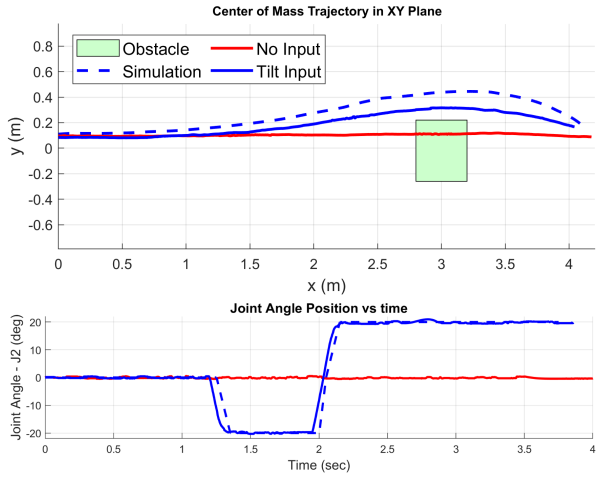

Fig. 8. Top figure shows the top view of the center of mass trajectory of the tumbling COBRA and obstacle avoidance. Bottom figure depicts the desired joint trajectory applied to the robot from the obstacle avoidance experiment.

where $\mathbf{0}$ and $\mathbf{I}$ denote zero and identity block matrices, respectively. r signifies the distance to the contact point. The nonlinear vector N is defined as follows:

$$N(x,y) = \begin{bmatrix} (y_3 - 2y_1)\,\dot{\psi}\dot{\theta}C_\theta + y_1\dot{\theta}\dot{\phi} \\ \beta_1 \\ \left(y_3 + 2mr^2\right)\dot{\psi}\dot{\theta}S_\theta \\ \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \\ \beta_2 \\ \beta_3 \end{bmatrix} \tag{23}$$

where β_i are given by:

$$\beta_1 = \left(y_1 - y_3 - mr^2\right)\dot{\psi}^2 S_\theta C_\theta - \left(y_3 + mr^2\right)\dot{\psi}\dot{\phi}S_\theta - mgrC_\theta$$
$$\beta_2 = -r\dot{\psi}C_\psi C_\theta + r\dot{\theta}S_\psi S_\theta - r\dot{\phi}C_\psi$$
$$\beta_3 = -r\dot{\psi}S_\psi C_\theta - r\dot{\theta}C_\psi S_\theta - r\dot{\phi}S_\psi$$

REFERENCES

[1] H. Wang, Y. F. Zheng, Y. Jun, and P. Oh, "DRC-hubo walking on rough terrains," in *2014 IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, Apr. 2014, pp. 1–6.

[2] M. Focchi, R. Orsolino, M. Camurri, *et al.*, "Heuristic Planning for Rough Terrain Locomotion in Presence of External Disturbances and Variable Perception Quality," en, in *Advances in Robotics Research: From Lab to Market: ECHORD++: Robotic Science Supporting Innovation*, ser. Springer Tracts in Advanced Robotics, Cham: Springer International Publishing, 2020, pp. 165–209.

[3] C. Mastalli, M. Focchi, I. Havoutis, *et al.*, "Trajectory and foothold optimization using low-dimensional models for rough terrain locomotion," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 1096–1103.

[4] H. Kolvenbach, P. Arm, E. Hampp, *et al.*, *Traversing Steep and Granular Martian Analog Slopes With a Dynamic Quadrupedal Robot*, Jun. 2021.

[5] X. Zeng, C. He, H. Oravec, A. Wilkinson, J. Agui, and V. Asnani, "Geotechnical Properties of JSC-1A Lunar Soil Simulant," EN, *Journal of Aerospace Engineering*, vol. 23, no. 2, pp. 111–116, Apr. 2010.

[6] G. B. Sanders and W. E. Larson, "Progress Made in Lunar In Situ Resource Utilization under NASA's Exploration Technology and Development Program," EN, pp. 457–478, Jul. 2012.

[7] M. M. Connors, D. B. Eppler, and D. G. Morrow, "Interviews with the Apollo lunar surface astronauts in support of planning for EVA systems design," Ames Research Center, Technical Report, Sep. 1994. [Online]. Available: https://repository.hou.usra.edu/handle/20.500.11753/674 (visited on 09/16/2021).

[8] *Big Idea — NASA's Breakthrough, Innovative, and Game-changing (BIG) Idea Challenge*, en-US.

[9] *2022 Extreme Mobility Challenge Finalists — Big Idea*, en-US.

[10] K. Damadeo, *Northeastern Slithers to the Top with BIG Idea Alternative Rover*, und, Text, Nov. 2022.

[11] A. Salagame, N. Bhattachan, A. Caetano, *et al.*, *How Strong a Kick Should be to Topple Northeastern's Tumbling Robot?* Nov. 2023.

[12] S. Jiang, A. Salagame, A. Ramezani, and L. Wong, *Hierarchical RL-Guided Large-scale Navigation of a Snake Robot*, Dec. 2023.

[13] S. Jiang, A. Salagame, A. Ramezani, and L. Wong, *Snake Robot with Tactile Perception Navigates on Large-scale Challenging Terrain*, Dec. 2023.

[14] A. Behar, J. Matthews, F. Carsey, and J. Jones, "NASA/JPL Tumbleweed polar rover," in *2004 IEEE Aerospace Conference Proceedings (IEEE Cat. No.04TH8720)*, vol. 1, Mar. 2004, 395 Vol.1.

[15] C. Batten and D. Wentzla, "Kickbot: A Spherical Autonomous Robot," en,

[16] A. Halme, T. Schonberg, and Y. Wang, "Motion control of a spherical mobile robot," in *Proceedings of 4th IEEE International Workshop on Advanced Motion Control - AMC '96 - MIE*, vol. 1, Mar. 1996, 259–264 vol.1.

[17] G. Reina, M. Foglia, A. Milella, and A. Gentile, "Rough-terrain traversability for a cylindrical shaped mobile robot," in *Proceedings of the IEEE International Conference on Mechatronics, 2004. ICM '04.*, Jun. 2004, pp. 148–153.

[18] A. Bicchi, A. Balluchi, D. Prattichizzo, and A. Gorelli, "Introducing the "SPHERICLE": An experimental testbed for research and teaching in nonholonomy," in *Proceedings of International Conference on Robotics and Automation*, vol. 3, Apr. 1997, 2620–2625 vol.3.

[19] A. Western, M. Haghshenas-Jaryani, and M. Hassanalian, "Golden wheel spider-inspired rolling robots for planetary exploration," en, *Acta Astronautica*, vol. 204, pp. 34–48, Mar. 2023.

[20] R. Mukherjee, M. Minor, and J. Pukrushpan, "Simple motion planning strategies for spherobot: A spherical mobile robot," in *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No.99CH36304)*, vol. 3, Dec. 1999, 2132–2137 vol.3.

[21] A. H. Javadi A. and P. Mojabi, "Introducing Glory: A Novel Strategy for an Omnidirectional Spherical Rolling Robot," *Journal of Dynamic Systems, Measurement, and Control*, vol. 126, no. 3, pp. 678–683, Dec. 2004.

[22] Y. Sugiyama and S. Hirai, "Crawling and Jumping by a Deformable Robot," *The International Journal of Robotics Research*, vol. 25, no. 5-6, pp. 603–620, May 2006.

[23] J. Paskarbeit, S. Beyer, M. Engel, A. Gucze, J. Schröder, and A. Schneider, "Ourobot—A sensorized closed-kinematic-chain robot for shape-adaptive rolling in rough terrain," en, *Robotics and Autonomous Systems*, vol. 140, p. 103 715, Jun. 2021.

[24] A. Salagame, S. Manjikian, C. Wang, *et al.*, *A Letter on Progress Made on Husky Carbon: A Legged-Aerial, Multi-modal Platform*, Jul. 2022.

[25] E. Sihite, A. Kalantari, R. Nemovi, A. Ramezani, and M. Gharib, "Multi-Modal Mobility Morphobot (M4) with appendage repurposing for locomotion plasticity enhancement," en, *Nature Communications*, vol. 14, no. 1, p. 3323, Jun. 2023.

[26] A. Salagame, M. Gianello, C. Wang, *et al.*, *Quadrupedal Locomotion Control On Inclined Surfaces Using Collocation Method*, Dec. 2023.

[27] E. Sihite, P. Dangol, and A. Ramezani, "Optimization-free Ground Contact Force Constraint Satisfaction in Quadrupedal Locomotion," in *2021 60th IEEE Conference on Decision and Control (CDC)*, Dec. 2021, pp. 713–719.

[28] K. Liang, E. Sihite, P. Dangol, A. Lessieur, and A. Ramezani, "Rough-Terrain Locomotion and Unilateral Contact Force Regulations With a Multi-Modal Legged Robot," in *2021 American Control Conference (ACC)*, May 2021, pp. 1762–1769.

[29] E. Sihite, A. Ramezani, and M. Gharib, *Dynamic modeling of wing-assisted inclined running with a morphing multi-modal robot*, Nov. 2023.