

OmBURo: A Novel Unicycle Robot with Active Omnidirectional Wheel

Junjie Shen¹ and Dennis Hong¹

Abstract—A mobility mechanism for robots to be used in tight spaces shared with people requires it to have a small footprint, to move omnidirectionally, as well as to be highly maneuverable. However, currently there exist few such mobility mechanisms that satisfy all these conditions well. Here we introduce Omnidirectional Balancing Unicycle Robot (OmBURo), a novel unicycle robot with active omnidirectional wheel. The effect is that the unicycle robot can drive in both longitudinal and lateral directions simultaneously. Thus, it can dynamically balance itself based on the principle of dual-axis wheeled inverted pendulum. This paper discloses the early development of this novel unicycle robot involving the overall design, modeling, and control, as well as presents some preliminary results including station keeping and path following. With its very compact structure and agile mobility, it might be the ideal locomotion mechanism for robots to be used in human environments in the future.

I. INTRODUCTION

On flat surfaces, wheeled locomotion is the most widely used mobility mechanism for robots. It has high energy efficiency since an ideal rolling wheel requires no further energy input to sustain its motion. This is in contrast to legged robots which suffer energy loss at heel-strike. For simplicity, most mobile robots have three or more wheels to achieve static stability. However, they can easily become dynamically unstable if the center of mass is too high, or the base of support is too narrow, or the acceleration is too large, or simply the slope is too steep [1]. As a result, the performance is significantly limited for statically-stable wheeled mobile robots.

On the other hand, many interesting mobile robots using only one or two wheels have been created over the past three decades [2], [3], [4]. These robots are dynamically stabilized by closed-loop control. That is, they are able to keep their balance autonomously without the need of static stability from the structure configuration. More interestingly, they utilize inertia for locomotion. In order to move in some direction, there is an initial retrograde wheel movement causing the body to lean towards the goal direction due to inertia, and then the wheels reverse and speed up. As performance limits of mobile robots are pushed, dynamic effects will increasingly come into play to enhance mobility [5], [6]. This becomes extremely true when it comes to robots operating in human environments. The robots are desired to be dynamically stable for better maneuverability, capable of omnidirectional motion, slim enough for moving in a crowd, and yet simple for the sake of safety as well as maintenance.

¹Junjie Shen and Dennis Hong are with the Robotics and Mechanisms Laboratory, the Department of Mechanical and Aerospace Engineering, University of California, Los Angeles, CA 90095, USA {junjieshen, dennishong}@ucla.edu

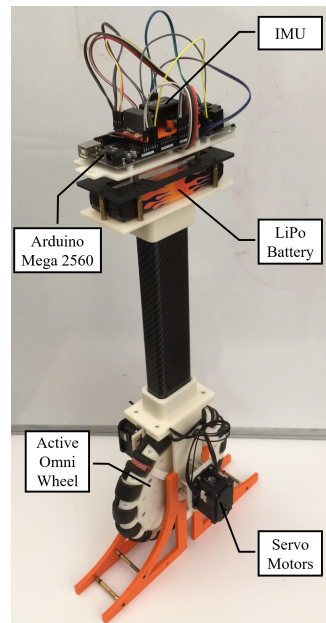


Fig. 1. OmBURo prototype

Some typical examples of dynamically-stable wheeled mobile robots have emerged to this day. Segway type robots [4], [7], [8], [9], [10] with two wheels are able to balance in one direction but cannot immediately move in the lateral direction without first re-orienting the drive system, and thus it cannot maneuver in highly dynamic environments. Besides, it might fall over on an inclined plane if not aligned with its moving direction. The cross section is also relatively large which can create problems in narrow spaces.

A conventional unicycle robot uses a single wheel for longitudinal balancing and another reaction wheel for lateral balancing, which however results in a quite complicated and bulky mechanical structure [2], [3], [11], [12], [13], [14], [15], [16], [17], [18]. It cannot move omnidirectionally either.

A robot that uses a ball instead of a wheel (a.k.a. Ballbot [1], [19], [20]) can move in any arbitrary direction, but its ball drive mechanism makes it unsuitable for ill-conditioned terrains. Essentially, the ball is driven by some rollers and this only works when there is sufficient friction between them. If the ground is wet or dirty, the ball surface can easily be influenced and the coefficient of friction will thus drop dramatically [21], [22]. The transmission may even fail in some extreme scenarios, e.g., slipping or stuck by some small particles from the ground.

Last but not least, Honda U3-X [23] is a mono-wheel personal mobility device unveiled in 2009. It's equipped with

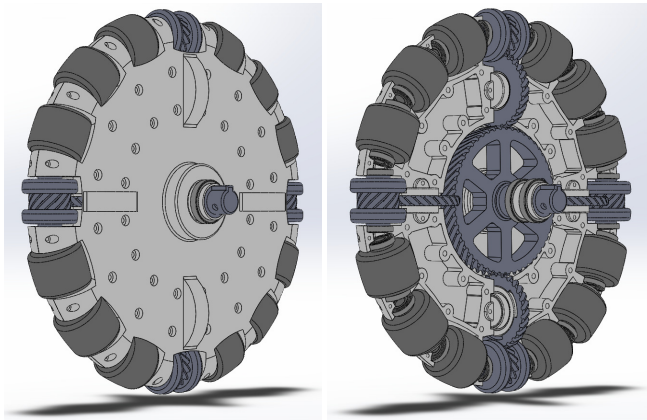


Fig. 2. Active omnidirectional wheel with/without housing enclosure

Honda Omni Traction Drive System, which is known as the world's first wheel structure that enables movement in all directions. U3-X is highly compact and easy to maintain due to its unique roller-on-tire friction drive transmission. However, its capabilities are consequently limited. Similar to Ballbot, it is desired to operate only on a clean and dry surface. Besides, it suffers rapid tire wear [24]. A normal tire makes contact with the ground once every revolution. On U3-X, the tire now has many more points of contact with the driving rollers which operate at a pretty high speed. Furthermore, the system needs to overcome velocity difference at these contact points, which in turn accelerates tire wear and causes energy loss as heat [25].

Inspired by the previous work, we have been developing OmBURo: a novel unicycle robot with active omnidirectional wheel. The mobility mechanism is desired to be as elegant as friction drive while avoiding its intrinsic problems, e.g., terrain limitation, tire wear, energy loss. The rest of this paper is organized in the following manner: Section II describes the physical system; mathematical modeling is derived in Section III; locomotion control strategy is elaborated in Section IV; experimental results are shown in Section V; and Section VI summarizes our contributions and future work.

II. SYSTEM DESCRIPTION

This section describes the physical system of the prototype as shown in Fig. 1. The major components are the active omnidirectional wheel, actuators, and sensors.

A. Active Omnidirectional Wheel

Omnidirectional wheels are wheels with small free rollers around the circumference which are perpendicular to the spinning direction. The effect is that omni wheels can be driven in both longitudinal and lateral directions without changing the orientation of the main wheel. However, since the rollers are passive, several omni wheels have to work in conjunction with each other to perform the desired driving maneuvers such as lateral side to side movement.

Recent studies have been focused on how to actuate the peripheral rollers in a conventional omni wheel (thus called *active omnidirectional wheel*) so that a single wheel unit can

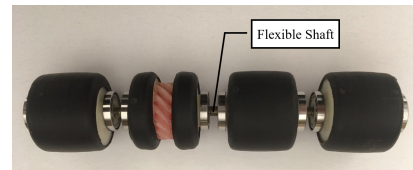


Fig. 3. Every four rollers are attached to one flexible shaft. One of them is driven via the helical gears.

actively move in any arbitrary direction. [26] first proposed a differential gear mechanism (in total 3 gear pairs with 1:24 gear ratio from input shaft to roller) to drive each roller. In order to accommodate as many rollers as possible for a smoother circumference, the output gear of the differential has to have a small radius, which in turn sacrifices output torque dramatically at the roller. In the end, it would be difficult to satisfy design requirements for highly dynamic locomotion. To increase output torque, [27] inserted a long gear train between the differential and the roller (5 gear pairs with 1:8 gear ratio). However, this deteriorates energy loss because more gear pairs result in more gear friction to overcome [28]. Besides, not all the rollers are directly driven via the gear mechanism but connected to the driven ones via universal joint, which is not a constant-velocity joint, i.e., the rollers have different rotating speeds in operation. [29] used double universal joint to solve the problem but ended up with a quite large gap between the rollers, causing lethal vibration issues when rolling.

Fig. 2 shows the mechanical structure of the proposed active omnidirectional wheel, aiming to improve the existing works for better implementation on a unicycle robot. Among the sixteen rollers, only four of them are directly driven via the helical gears. The gear mechanism is not used for all the rollers mainly to reduce energy loss from gear friction as well as weight and inertia for better maneuverability. To drive the rest rollers, flexible shaft is used as shown in Fig. 3. It is a rotating coil which is flexible in bending but still has high torsional stiffness, so that constant velocity is guaranteed among the rollers. Compare to [26], [27], [29], the mechanism is simpler but still functional; it has larger output torque due to a higher gear ratio (1:4); all the rollers are synchronized while the gap between them is smaller, leading to a desired smoother rolling motion.

B. Actuators

The prototype uses two Dynamixel MX-64 servo motors as actuators. One is directly driving the main wheel in the longitudinal direction while the other is actuating the rollers laterally through the helical gears and flexible shaft.

C. Sensors

A LORD Microstrain 3DM-GX4-25 Inertial Measurement Unit (IMU) provides Kalman-filtered Euler angles and rates of the unicycle body with respect to gravity. Additionally, each servo motor is equipped with a 4096 PPR encoder (i.e., 0.088° resolution) detecting motor shaft angular position and velocity. Full-state feedback control can thus be achieved.

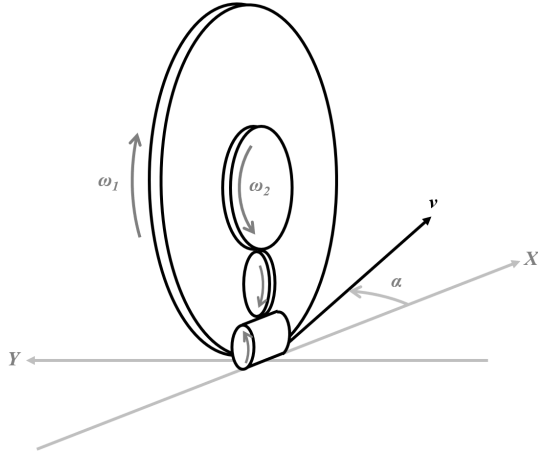


Fig. 4. Velocity kinematics diagram

D. Others

The prototype is designed to be completely self-contained, powered by a 14.4 V 3300 mAh LiPo battery and controlled by an Arduino Mega 2560 board at 125 Hz. The Arduino communicates with the servo motors via an RS-485 interface while the IMU via an RS-232 interface.

III. MODELING

This section derives the mathematical modeling of kinematics and dynamics for OmBURo.

A. Kinematics

Fig. 4 shows the velocity kinematics diagram of OmBURo. Note that the kinematic model is derived assuming it is always at the nominal upright configuration.

Given the motor speeds ω_1 and ω_2 , the velocity v of OmBURo is determined to be

$$\underbrace{\begin{bmatrix} v_x \\ v_y \end{bmatrix}}_v = \begin{bmatrix} \omega_1 R \\ n(\omega_1 + \omega_2)r \end{bmatrix} = \underbrace{\begin{bmatrix} R & 0 \\ nr & nr \end{bmatrix}}_J \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix}, \quad (1)$$

where n is the inverse of the gear ratio. The radius of the main wheel is denoted as R . The radius of the roller is denoted as r . Since the Jacobian matrix J is nonsingular, we can inversely calculate the motor speeds based on the desired OmBURo moving velocity. Table I summarizes the relationship between the motor speeds and OmBURo moving direction α .

TABLE I
OMBURo MOVING DIRECTION

Motor Speeds	Moving Direction
$\omega_1 > 0, \omega_1 + \omega_2 = 0$	$\alpha = 0$
$\omega_1 > 0, \omega_1 + \omega_2 > 0$	$0 < \alpha < \pi/2$
$\omega_1 = 0, \omega_1 + \omega_2 > 0$	$\alpha = \pi/2$
$\omega_1 < 0, \omega_1 + \omega_2 > 0$	$\pi/2 < \alpha < \pi$
$\omega_1 < 0, \omega_1 + \omega_2 = 0$	$\alpha = \pi$
$\omega_1 < 0, \omega_1 + \omega_2 < 0$	$\pi < \alpha < 3\pi/2$
$\omega_1 = 0, \omega_1 + \omega_2 < 0$	$\alpha = 3\pi/2$
$\omega_1 > 0, \omega_1 + \omega_2 < 0$	$3\pi/2 < \alpha < 2\pi$

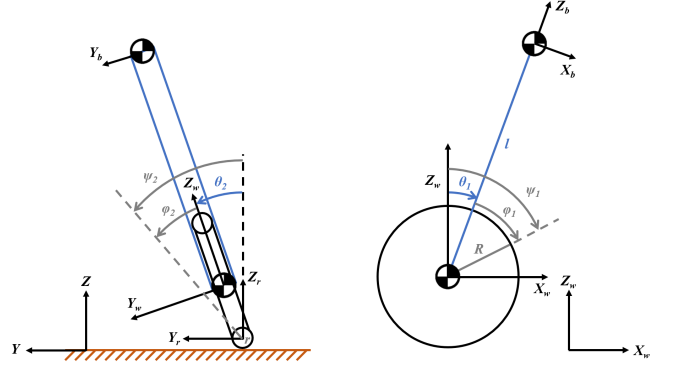


Fig. 5. OmBURo model

To derive the position kinematics, we can simply integrate both sides of (1) over time and end up with

$$\begin{bmatrix} p_x \\ p_y \end{bmatrix} = \mathbf{J} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}, \quad (2)$$

where p_x and p_y indicate the position of OmBURo while β_1 and β_2 indicate the motor rotation angles.

B. Dynamics

Some major assumptions are established in the first place: all the components are rigid bodies; the ground is even and hard; the wheel is always in contact with the ground without slipping; and yaw dynamics is negligible.

Fig. 5 shows the model of OmBURo in detail. The inertial frame is denoted as XYZ . The roller frame is denoted as $X_r Y_r Z_r$ corresponding to the inertial frame. The wheel frame is denoted as $X_w Y_w Z_w$ with the X_w -axis always parallel to the X_r -axis. The body frame is denoted as $X_b Y_b Z_b$ with the Y_b -axis always parallel to the Y_w -axis. θ_1 is the body angle from Z_w to Z_b along Y_w . θ_2 is the body angle from Z_r to Z_w along X_r . ψ_1 is the rotation angle of the wheel along Y_w . ψ_2 is the rotation angle of the roller along X_r . φ_1 is the relative rotation angle between the body and the wheel along Y_w . φ_2 is the relative rotation angle between the body and the roller along X_r . The distance between the wheel frame and the body frame is denoted as l .

Lagrange formulation is used to derive the equations of motion which take the form:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{F}(\dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \mathbf{W}(\mathbf{u}), \quad (3)$$

where $\mathbf{q} = [\theta_1, \theta_2, \varphi_1, \varphi_2]^T$ is the vector of generalized coordinates and $\mathbf{u} = [u_1, u_2]^T$ is the vector of generalized forces applied at $\varphi = [\varphi_1, \varphi_2]^T$, $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ represents the vector of Coriolis and centrifugal forces, $\mathbf{F}(\dot{\mathbf{q}})$ represents the vector of viscous friction forces, $\mathbf{G}(\mathbf{q})$ stands for the vector of gravitational forces, and $\mathbf{W}(\mathbf{u})$ stands for the vector of control inputs. The details of derivation are shown here (<http://arxiv.org/abs/2001.07856>).

Note that the two relative rotation angles φ_1 and φ_2 are not involved in the equations of motion since they only determine the position of OmBURo due to the non-slipping assumption. And the position would not influence its dynamics based on the uniform-terrain assumption.

IV. CONTROL

This section discusses the development of locomotion control strategy for OmBURo.

A. State-space Realization

Since the equations of motion are highly nonlinear and complicated while we always expect OmBURo to operate in the neighborhood of the nominal upright configuration ($\theta_1 = \theta_2 \approx 0, \dot{\theta}_1 = \dot{\theta}_2 \approx 0$), it's reasonable to linearize (3) at this configuration, which yields

$$a_1\ddot{\theta}_1 + a_2\ddot{\varphi}_1 + \mu_g\dot{\theta}_1 + \mu_g\dot{\varphi}_1 + a_3\theta_1 = 0, \quad (4)$$

$$a_4\ddot{\theta}_2 + a_5\ddot{\varphi}_2 + \mu_g\dot{\theta}_2 + \mu_g\dot{\varphi}_2 + a_6\theta_2 = 0, \quad (5)$$

$$a_2\ddot{\theta}_1 + M_{33}\ddot{\varphi}_1 + \mu_g\dot{\theta}_1 + (\mu_g + \mu_1)\dot{\varphi}_1 = u_1, \quad (6)$$

$$a_5\ddot{\theta}_2 + M_{44}\ddot{\varphi}_2 + \mu_g\dot{\theta}_2 + (\mu_g + \mu_2)\dot{\varphi}_2 = u_2, \quad (7)$$

where the parameters $a_1, a_2, a_3, a_4, a_5, a_6$ are shown here (<http://arxiv.org/abs/2001.07856>). (4)–(7) indicates that the linearized OmBURo dynamics can be completely decoupled in the X and Y directions. (4) and (6) are only related to θ_1, φ_1 , and u_1 , while (5) and (7) are only related to θ_2, φ_2 , and u_2 , which is essentially dual-axis wheeled inverted pendulum model. Solving (4)–(7) leads to the state-space realization

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad (8)$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ A_{31} & 0 & A_{33} & 0 & A_{35} & 0 \\ 0 & A_{42} & 0 & A_{44} & 0 & A_{46} \\ A_{51} & 0 & A_{53} & 0 & A_{55} & 0 \\ 0 & A_{62} & 0 & A_{64} & 0 & A_{66} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ B_{31} & 0 \\ 0 & B_{42} \\ B_{51} & 0 \\ 0 & B_{62} \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \varphi_1 \\ \varphi_2 \end{bmatrix}.$$

The elements of \mathbf{A} and \mathbf{B} are listed here (<http://arxiv.org/abs/2001.07856>). The pair of (\mathbf{A}, \mathbf{B}) is verified to be controllable and all the states can be measured directly.

B. Locomotion Control

The objective of locomotion control for OmBURo is to track the reference velocity while keeping balance of its body. It is realized by the control system shown in Fig. 6, which has one feedforward part and two closed loops: an inner loop control based on linear quadratic regulator (LQR) and an outer loop compensator using proportional-integral (PI) control.

The feedforward part feeds the desired OmBURo velocity into the inner loop control as the reference signal. Note that $\dot{\psi}_{\text{ref}}$ and $\dot{\varphi}_{\text{ref}}$ are equivalent since $\dot{\psi} = \dot{\varphi} + \dot{\theta}$ while $\dot{\theta} = 0$ at steady state.

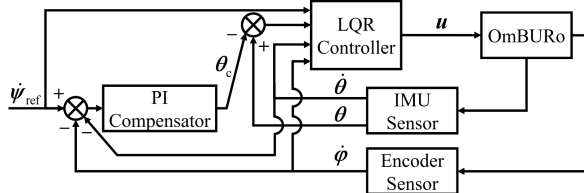


Fig. 6. Locomotion control system

With full state feedback, LQR is used to track the constant reference velocity $\dot{\psi}_{\text{ref}}$. Using the feedback control law $\mathbf{u} = -\mathbf{K}\mathbf{x} + \mathbf{H}\dot{\psi}_{\text{ref}}$, (8) becomes

$$\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{BK})\mathbf{x} + \mathbf{BH}\dot{\psi}_{\text{ref}} \\ \mathbf{y} = \begin{bmatrix} \dot{\psi}_1 \\ \dot{\psi}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}}_{\mathbf{C}}\mathbf{x}, \quad (9)$$

where \mathbf{K} is the gain matrix computed by MATLAB's *lqr* command with proper choice of the weighting matrices. Considering the constant reference velocity $\dot{\psi}_{\text{ref}}$ as a step input, the steady state output vector is thus given by

$$\lim_{t \rightarrow \infty} \mathbf{y}(t) = \mathbf{C}(\mathbf{BK} - \mathbf{A})^{-1} \mathbf{BH}\dot{\psi}_{\text{ref}}. \quad (10)$$

In order to track the reference well, \mathbf{H} is determined to be

$$\mathbf{H} = \left(\mathbf{C}(\mathbf{BK} - \mathbf{A})^{-1} \mathbf{B} \right)^{-1} \quad (11)$$

so that $\lim_{t \rightarrow \infty} \mathbf{y}(t) = \dot{\psi}_{\text{ref}}$. The inverse exists because there are as many control inputs as independent controlled outputs.

The outer loop control essentially compensates for the sensor bias as well as the modeling errors. Generally, it is inevitable to have these problems when implementing the control algorithms on the real physical system. For example, to keep OmBURo at a fixed position while balancing it, we have $\dot{\psi}_{\text{ref}} = \mathbf{0}$ and (8) implies

$$\lim_{t \rightarrow \infty} \boldsymbol{\theta}(t) = -\mu_g \begin{bmatrix} 1/a_3 & 0 \\ 0 & 1/a_6 \end{bmatrix} \dot{\psi}_{\text{ref}} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (12)$$

by setting $\ddot{\theta} = \ddot{\varphi} = \mathbf{0}$ and $\dot{\theta} = \mathbf{0}$. However, the real balanced body angle is actually unknown and probably nonzero mainly due to IMU bias and shifted center of mass (CoM) from the central axis. Using just LQR, OmBURo will move towards its CoM and leave away from its initial position until it falls. With that in mind, we can actually use this information from the encoder to correct the body angle θ via a PI compensator. The proportional and integral gains are experimentally tuned.

V. EXPERIMENTAL RESULTS

Several tests were conducted to evaluate the performance of the physical system along with the proposed control strategy, which can be found online (<https://youtu.be/avtlx1-X3lo>). Corresponding experimental results are demonstrated in this section. Note that in all the wheel trajectory plots, the coordinate system corresponds to Fig. 4 and OmBURo always starts at the origin.

A. Station Keeping & Disturbance Rejection

In this test, we first want to keep OmBURo at a fixed position while balancing it. Accordingly, we have $\dot{\psi}_{\text{ref}} = \mathbf{0}$. OmBURo was released out of balance but was able to quickly correct its body orientation via its wheel motion. Fig. 7(a) shows its wheel trajectory. We can see that OmBURo was keeping its position within the neighborhood of some point at steady state, a nearly circular region with a diameter of about 5 cm. Fig. 7(b) shows the corresponding time series

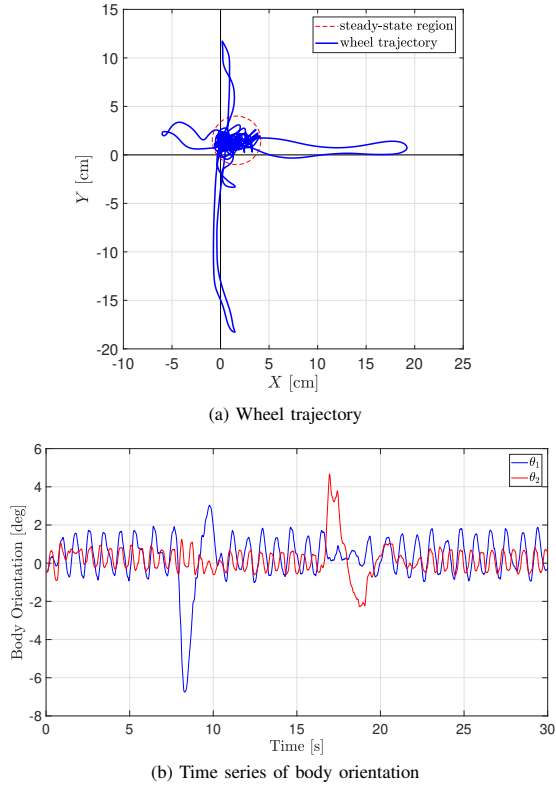


Fig. 7. Experimental results of station keeping and disturbance rejection. While station keeping, OmBURo was intentionally pushed in the positive X -direction around $t = 8$ s and later in the Y -direction around $t = 17$ s. It was able to recover within 3 seconds in both scenarios.

of its body orientation. Note that the real balanced body angle (from IMU) is nonzero at steady-state, as mentioned before.

To further test the robustness of the whole system, a disturbance rejection experiment was carried out based on station keeping. When OmBURo was balancing itself at a fixed position, its body was intentionally pushed in the positive X -direction at time around 8 seconds. It regained its balance and went back to the original position within 3 seconds, as shown in Fig. 7. Another push was made in the negative Y -direction at time around 17 seconds and OmBURo was able to recover as well. The large deviation from the initial condition implies a quite decent stability region for the prototype, and it can be further improved by using more powerful actuators.

B. Velocity Tracking

In this test, we want to move OmBURo with a constant velocity. To avoid latent instability caused by the sudden state change, a trapezoid reference is predefined instead of a step reference. Fig. 8 shows data compared with the reference. One interesting observation is that the wheel motion almost always had an opposite behavior to body orientation. The fact is that to move in some given direction, OmBURo however starts with a quick opposite movement, causing its body to lean towards the goal direction due to inertia, and then reverses its wheel and speeds up using its dynamics.

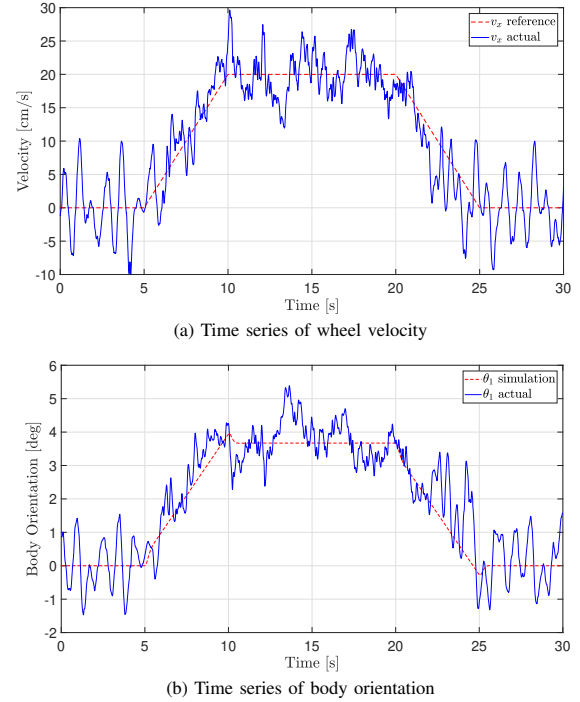


Fig. 8. Experimental results of velocity tracking. In order for OmBURo to move with a constant velocity, its body needs to lean for a certain angle against ground friction, as suggested by (12). Due to limited space, results are shown only for the X -direction.

C. Path Following

In this test, we want to make OmBURo follow some predefined paths. Since it has no extrinsic sensors for now, OmBURo was remotely controlled to adjust its velocity reference during the test. Fig. 9 shows two scenarios, namely (a) a triangular path and (b) a circular path. These motions were intentionally made slow and steady so that OmBURo could be tightly controlled to follow the path well, but it might not be the best approach judging from the jagged profile. Nevertheless, the capability of following these paths indicates that the overall platform is good enough to implement trajectory tracking in terms of both desired position and velocity.

D. Balancing on an Inclined Plane

Finally, we tried to make OmBURo balance itself on an inclined plane. Similar to balancing on a horizontal plane, we set $\psi_{\text{ref}} = \mathbf{0}$ and just released it. It first rolled down the slope for some distance but was still able to quickly balance itself and eventually keeping its position within some area, as shown in Fig. 10(a, b). Ideally, in order to balance on a slope, its CoM should be right above the contact point. For simplicity, here we are only considering the situation when OmBURo is aligned with the inclination angle, as shown in Fig. 10(c). The relationship between the pitch angle θ_1 and the inclination angle γ is thus determined to be

$$m_b l \sin \theta_1 = (m_b + m_w) R \sin \gamma, \quad (13)$$

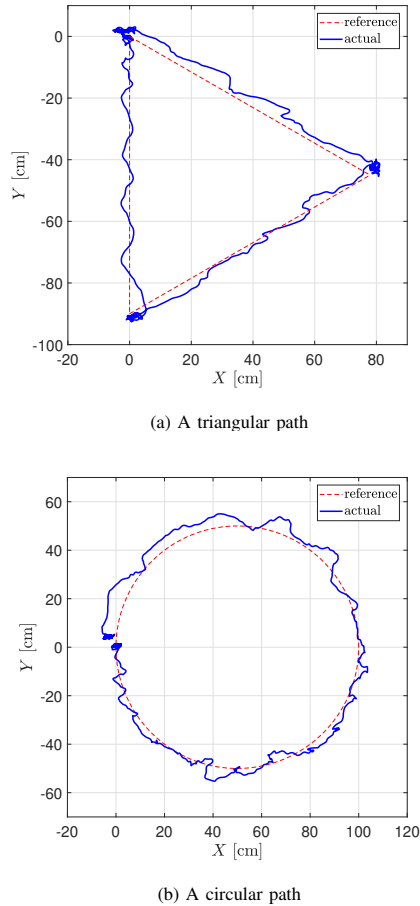


Fig. 9. Experimental results of path following. The red dashed line is the reference while the blue line is the actual wheel trajectory. OmbURo was remotely controlled to adjust its velocity reference during the test.

where m_w and m_b are the masses of the wheel and body, respectively. Fig. 10(d) shows calculated values compared with measured data range at steady state. The largest feasible inclination angle is around 30 degrees, presumably limited by the ground friction coefficient. Unlike Segway type robots, which might fall over on an inclined plane if not aligned with its moving direction, we further changed the yaw angle manually during the test and OmbURo was still capable of station keeping.

VI. CONCLUSION

In this paper, first, a novel active omnidirectional wheel was proposed. For conventional omni wheels, the peripheral rollers can only be passively driven. As a result, several omni wheels have to work together for omnidirectional drive system. Since the rollers can now be directly actuated, a single wheel unit is sufficient to actively move in any arbitrary direction.

Second, an investigation of the existing omnidirectional mobility mechanisms was conducted. Friction drive transmission on Ballbot and U3-X is simple and elegant. However, it has limitations on ill-conditioned terrains. It also suffers rapid tire wear and energy loss as heat. Gear transmission avoids

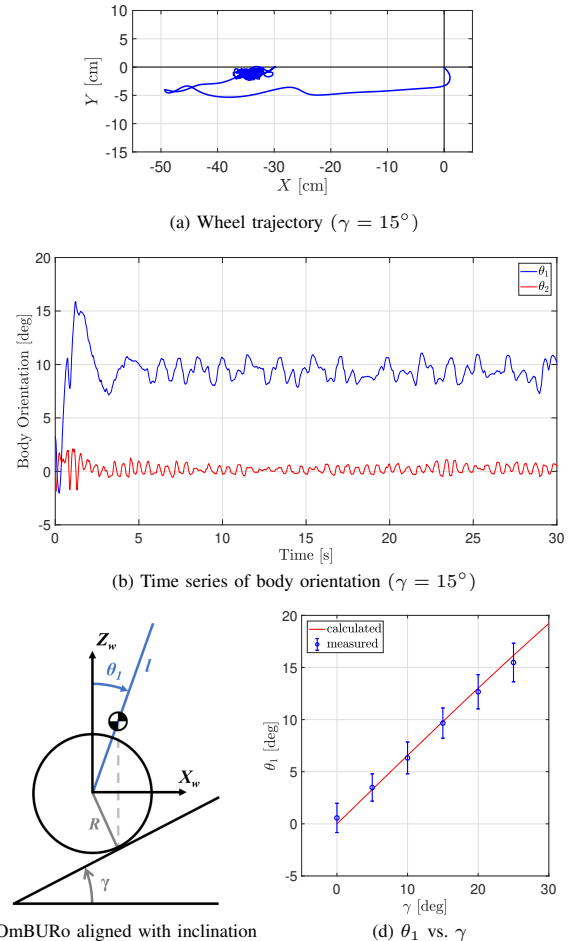


Fig. 10. Experimental results of balancing on an inclined plane. After release, OmbURo first rolled down the slope for some distance but was still capable of station keeping eventually. Furthermore, it was able to adapt to different inclination angles by adjusting its body orientation.

these problems but results in a large gap between the rollers. Future work will be concentrated on how to minimize the gap for a smoother rolling motion. Besides, a better structure can be explored to drive the bottom roller only while freeing the rest, thus enhancing energy efficiency.

Third, a new unicycle robot, OmbURo, was built based on the proposed active omnidirectional wheel. Accordingly, an interesting unicycle model was derived and a corresponding locomotion control strategy was designed. The performance is good enough to implement trajectory tracking in terms of both desired position and velocity. It might be eventually used in human environments due to its very simple and compact structure, as well as agile and omnidirectional mobility. Future work will be further focused on its locomotion control design, e.g., steering control and trajectory tracking.

ACKNOWLEDGMENT

This work is partially supported by ONR through grant N00014-15-1-2064. The authors thank Xiaoguang Zhang and Yeting Liu for helping realize the physical system as well as Zhaoxing Deng for helping with the experiments.

REFERENCES

- [1] T. B. Lauwers, G. A. Kantor, and R. L. Hollis, "A dynamically stable single-wheeled mobile robot with inverse mouse-ball drive," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pp. 2884–2889, May 2006.
- [2] A. Schoonwinkel, *Design and Test of a Computer-stabilized Unicycle*. PhD thesis, Stanford University, Stanford, CA, USA, 1988. AAI8808416.
- [3] D. W. Vos and A. H. Von Flotow, "Dynamics and nonlinear adaptive control of an autonomous unicycle: theory and experiment," in *29th IEEE Conference on Decision and Control*, pp. 182–187 vol.1, Dec 1990.
- [4] Y. Ha and S. Yuta, "Trajectory tracking control for navigation of self-contained mobile inverse pendulum," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94)*, vol. 3, pp. 1875–1882 vol.3, Sep. 1994.
- [5] H. B. Brown and Yangsheng Xu, "A single-wheel, gyroscopically stabilized robot," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 4, pp. 3658–3663 vol.4, April 1996.
- [6] Y. Xu and Y. Ou, *Control of Single Wheel Robots*. Springer Tracts in Advanced Robotics, Springer Berlin Heidelberg, 2005.
- [7] Y. Kim, S. H. Kim, and Y. K. Kwak, "Dynamic analysis of a nonholonomic two-wheeled inverted pendulum robot," *Journal of Intelligent and Robotic Systems*, vol. 44, pp. 25–46, Sep 2005.
- [8] D. Pratama, F. Ardilla, E. H. Binugroho, and D. Pramadihanto, "Tilt set-point correction system for balancing robot using pid controller," in *2015 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, pp. 129–135, Aug 2015.
- [9] B. Gao, X. Zhang, W. Yao, Y. Guo, Y. Bao, and J. Xie, "Dynamics and motion control of an underactuated two-wheeled mobile robot," in *2012 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 179–184, May 2012.
- [10] N. Uddin, "Adaptive control system design for two-wheeled robot stabilization," in *2018 12th South East Asian Technical University Consortium (SEATUC)*, vol. 1, pp. 1–5, March 2018.
- [11] R. Nakajima, T. Tsubouchi, S. Yuta, and E. Koyanagi, "A development of a new mechanism of an autonomous unicycle," in *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robot and Systems. Innovative Robotics for Real-World Applications. IROS '97*, vol. 2, pp. 906–912 vol.2, Sep. 1997.
- [12] M. A. Rosyidi, E. H. Binugroho, S. E. Radin Charel, R. S. Dewanto, and D. Pramadihanto, "Speed and balancing control for unicycle robot," in *2016 International Electronics Symposium (IES)*, pp. 19–24, Sep. 2016.
- [13] J. Xu, A. Al Mamun, and Y. Daud, "Pendulum-balanced autonomous unicycle: Conceptual design and dynamics model," in *2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM)*, pp. 51–56, Sep. 2011.
- [14] Y. Rizal, C. Ke, and M. Ho, "Point-to-point motion control of a unicycle robot: Design, implementation, and validation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4379–4384, May 2015.
- [15] J. Hong, C. Huang, K. Lu, and T. Yeh, "Analysis and design of a passive steering mechanism for a pedaled, self-balanced, personal mobility vehicle," in *Proceedings of the 2013 IEEE/SICE International Symposium on System Integration*, pp. 179–184, Dec 2013.
- [16] H. Jin, J. Hwang, and J. Lee, "A balancing control strategy for a one-wheel pendulum robot based on dynamic model decomposition: Simulations and experiments," *IEEE/ASME Transactions on Mechatronics*, vol. 16, pp. 763–768, Aug 2011.
- [17] H. Jin, T. Wang, F. Yu, Y. Zhu, J. Zhao, and J. Lee, "Unicycle robot stabilized by the effect of gyroscopic precession and its control realization based on centrifugal force compensation," *IEEE/ASME Transactions on Mechatronics*, vol. 21, pp. 2737–2745, Dec 2016.
- [18] L. Guo, K. He, and Y. Song, "Dynamic modeling and control of a kind of unicycle robot," in *2017 36th Chinese Control Conference (CCC)*, pp. 6901–6905, July 2017.
- [19] Masaaki Kumagai and Takaya Ochiai, "Development of a robot balancing on a ball," in *2008 International Conference on Control, Automation and Systems*, pp. 433–438, Oct 2008.
- [20] L. Hertig, D. Schindler, M. Bloesch, C. D. Remy, and R. Siegwart, "Unified state estimation for a ballbot," in *2013 IEEE International Conference on Robotics and Automation*, pp. 2471–2476, May 2013.
- [21] M. West and H. Asada, "Design of Ball Wheel Mechanisms for Omnidirectional Vehicles With Full Mobility and Invariant Kinematics," *Journal of Mechanical Design*, vol. 119, pp. 153–161, June 1997.
- [22] Engineering Toolbox, *Friction and Friction Coefficients for various Materials*, 2004. [online] Available at: https://www.engineeringtoolbox.com/friction-coefficients-d_778.html [Accessed 2019-12-2].
- [23] H. Gomi and T. Takenaka, "Frictional drive device and inverted pendulum type vehicle using the same," Mar 2011.
- [24] D. J. Rodríguez Obando, J. J. Martínez Molina, and C. Bérenguer, "Condition Monitoring of a Friction Drive System," in *3rd IEEE Colombian Conference on Automatic Control (CCAC 2017)*, Proceedings of 2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC), (Cartagena de Indias, Colombia), IEEE, 2017.
- [25] B. Bhushan, *Modern Tribology Handbook, Two Volume Set*, ch. 6, pp. 235–272. Mechanics & Materials Science, CRC Press, 2000.
- [26] Y. Uehara, "Omni traction wheel system and methods of operating the same," Apr 2013.
- [27] M. Komori, K. Matsuda, T. Terakawa, F. Takeoka, H. Nishihara, and H. Ohashi, "Active omni wheel capable of active motion in arbitrary direction and omnidirectional vehicle," *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, vol. 10, no. 6, pp. JAMDSM0086–JAMDSM0086, 2016.
- [28] E. Buckingham, *Analytical Mechanics of Gears*, ch. 19, pp. 395–425. Dover Publications, Inc., 1949.
- [29] S. M. Samarasinghe and M. Parnichkun, "Pitch control of an active omni-wheeled unicycle using lqr," in *2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP)*, pp. 98–101, Jan 2019.