

# Electro-Hydraulic Rolling Soft Wheel: Design, Hybrid Dynamic Modeling, and Model Predictive Control

Khoi Ly<sup>✉</sup>, Jatin V. Mayekar<sup>✉</sup>, Sarah Aguasvivas, Christoph Keplinger<sup>✉</sup>,  
Mark E. Rentschler<sup>✉</sup>, Senior Member, IEEE, and Nikolaus Correll<sup>✉</sup>, Senior Member, IEEE

**Abstract**—Locomotion through rolling is attractive compared to other forms of locomotion thanks to uniform designs, high degree of mobility, dynamic stability, and self-recovery from collision. Despite previous efforts to design rolling soft systems, pneumatic and other soft actuators are often limited in terms of high-speed dynamics, system integration, and/or functionalities. Furthermore, mathematical description of the rolling dynamics for this type of robot and how the models can be used for speed control are often not mentioned. This article introduces a cylindrical-shaped shell-bulging rolling soft wheel that employs an array of 16 folded-HASEL actuators as a mean for improved rolling performance. The actuators represent the soft components with discrete forces that propel the wheel, whereas the wheel’s frame is rigid but allows for smooth, continuous change in position and speed. We discuss the interplay between the electrical and mechanical design choices, the modeling of the wheel’s hybrid (continuous and discrete) dynamic behavior, and the implementation of a model predictive controller (MPC) for the robot’s speed. With the balance of several design factors, we show the wheel’s ability to carry integrated hardware with a maximum rolling speed at 0.7 m/s (or 2.2 body lengths per second), despite its total weight of 979 g, allowing the wheel to outperform the existing rolling soft wheels with comparable weights and sizes. We also show that the MPC enables the wheel to accelerate and leverage its inherent braking capability to reach

desired speeds—a critical function that did not exist in previous rolling soft systems.

**Index Terms**—Development and prototyping, electrohydraulic actuators, model predictive control, optimization, self-rolling wheel.

## I. INTRODUCTION

ROLLING exhibits high degree of mobility, inherent stability, and self-recovery from collision when compared to other robotic locomotion methods such as legged, crawling, and hopping [1]; also, rolling is much more energy-efficient, since a rolling system does not need to expend energy lifting its body for movement [2], [3]. In addition to these inherent advantages, numerous research efforts have presented systems’ abilities to carry payloads [4]–[7], provide protections to internal electrical and mechanical components [4], [5], [7]–[9], and navigate difficult environments [4], [6], [8], [10], [11]. Such favorable traits make robotic rolling locomotion a widely investigated subject for space exploration [4], [12], search and rescue [13], and military applications [14].

In the literature, systems with continuous rolling locomotion can be further categorized based on three operating principles. The most popular form of rolling is shift of the center of mass (COM) of a cylinder or sphere away from its neutral position, causing the cylinder or sphere to fall in that direction and thus roll along [1], [7], [8], [10], [15]–[26]. These systems are commonly characterized by rigid cylindrical or spherical outer shell and internal components, such as gears, servos, and motors, which have a well-established history of research and optimization, especially in power output, efficiency, part selections, and reliability. To the best of authors’ knowledge, there were only two rolling systems driven by soft actuators [27], [28] that are based on shift of COM.

Another class of rolling systems are distinguished for the ability to deform their geometry, inducing structural instability which generates the rolling motion [29]–[34]—a notable subclass of these designs is rolling of tensegrity structures that has been targeted for space exploration [6], [35]. These designs are often characterized by their modular links that can interact with one another to create global deformed shapes. Previous works on geometry-deformable rolling systems have demonstrated single-mode (only rolling) and multimode (rolling and

Manuscript received February 13, 2022; accepted April 2, 2022. This work was supported by the National Science Foundation CPS program under Grant 1739452, in part by the Air Force Office of Scientific Research under Grant 83875-11094, and in part by the Army Research Office under Grant W911NF-18-1-0203, which was used to purchase laboratory equipment to characterize and fabricate actuators. This paper was recommended for publication by Associate Editor Rafael Murrieta-Cid and Editor Mark Yim upon evaluation of the reviewers’ comments. (*Corresponding author: Khoi Ly*)

Khoi Ly, Jatin V. Mayekar, and Mark E. Rentschler are with the Department of Mechanical Engineering, University of Colorado Boulder, Boulder, CO 80309 USA (e-mail: khoi.ly@colorado.edu; jama3012@colorado.edu; mark.rentscher@colorado.edu).

Sarah Aguasvivas is with the Department of Computer Science, University of Colorado Boulder, Boulder, CO 80309 USA (e-mail: sarah.aguasvivas.manzano@colorado.edu).

Christoph Keplinger is with the Department of Mechanical Engineering, University of Colorado Boulder, Boulder, CO 80309 USA, and also with the Robotic Materials Department, Max Planck Institute for Intelligent Systems, 70569 Stuttgart, Germany (e-mail: christoph.keplinger@colorado.edu).

Nikolaus Correll is with the Materials Science and Engineering Program, University of Colorado Boulder, Boulder, CO 80309 USA, and also with the Department of Computer Science, University of Colorado Boulder, Boulder, CO 80309 USA (e-mail: nikolaus.correll@colorado.edu).

This article has supplementary material provided by the authors and color versions of one or more figures available at <https://doi.org/10.1109/TRO.2022.3167438>.

Digital Object Identifier 10.1109/TRO.2022.3167438

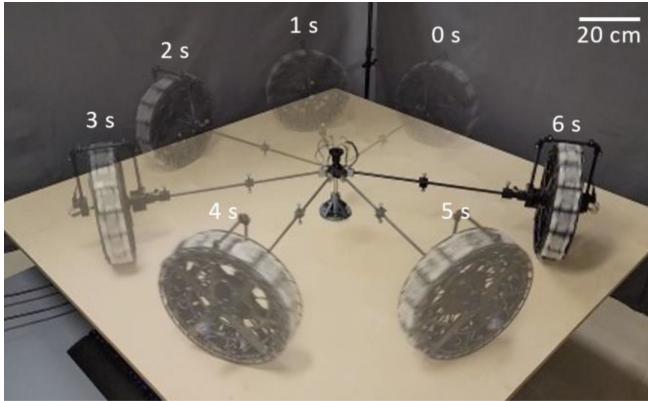


Fig. 1. Motion sequence of the electrohydraulic rolling soft wheel around a pivot on a square platform. The direction of travel is counterclockwise from the top view. The rolling soft wheel has a non-zero initial linear speed at time 0 s.

hopping) locomotion, which is useful for navigating through rough terrain and tall obstacles. However, due to the structural intricacy, most systems with deformable geometry require more complex designs, fabrication procedures, and kinematic and dynamic models, thus making the systems difficult to control.

While the shells of geometric deforming systems and systems that roll based on shift of COM can be both rigid and soft ([11], [27], [28], [31], [32], [34]), those of shell-bulging systems are mostly soft [34], [36]–[40]. These designs often have soft shell segments that can independently protrude and generate contact forces between the system and ground that propel the system forward. Unlike rolling systems with rigid shells, compliance of the soft shells allows for absorbing impact energy [5], [6] and improved traction and conformity to the surface at contact, which enhances stability on rough terrain [41]. However, one of the main limitations of these systems is the tradeoff between performances and the required integration of hardware and software that have not been addressed in the literature. Specifically, most shell-bulging soft systems are pneumatic-based, hydraulic-based, shape-memory-alloy-based, and electrostatic-based, which have significantly low power-to-weight ratio due to bulky external driving components, which results in slow dynamics [36]–[40]. While [33] has demonstrated high locomotion speed greater than 1.43 body length per second (blps), the designs in [33] and [36]–[40] as systems lack of integrated hardware (sensors, actuators, and actuator drivers) and software (closed loop controllers) that rolling robots often require. Furthermore, compared to continuous dynamics of systems with shifting COM and discrete dynamics of geometry-deformable rolling systems, shell-bulging soft systems are systems with hybrid dynamics whose input forces are discontinuous and output velocities are continuous, making the design and integration of controller (currently nonexistent) more challenging.

With hydraulically amplified self-healing electrostatic (HASEL or electrohydraulic) actuators [42], [43] and their recent improvements in electrical interface [43] and performance [44], we introduce a novel shell-bulging locomotion of a self-rolling wheel (Fig. 1) that is powered by folded-HASEL actuators. Our design outperforms existing shell-bulging rolling

wheels in several metrics, addresses several challenges of system integration, and demonstrates functionalities that did not exist in previous works. Specifically, the HASEL-driven self-rolling wheel, thanks to its hybrid structure of soft actuators placed around a circular rigid skeletal frame, demonstrates much faster rolling characteristics, attaining a rolling maximum speed of 0.7 m/s (2.2 blps), despite being relatively heavy at 979 g. Through this article, we present four key contributions.

- 1) The design of a self-rolling soft wheel with integrated hardware for electrohydraulic actuators.
- 2) The mathematical hybrid-dynamic model to describe the behavior of a family of cylindrical-shaped shell-bulging rolling wheels.
- 3) The demonstration of inherent acceleration and deceleration functions through electromechanical design.
- 4) The speed-regulating controller for shell-bulging rolling wheels with continuous outputs and discrete inputs.

The electromechanical and software integration in this soft wheel is a fundamental building block toward a four-wheel platform that provides several alternative benefits to traditional vehicles.

## II. ELECTROMECHANICAL DESIGN

### A. Mechanical Design

The electrohydraulic rolling soft wheel consists of 16 folded-HASEL actuators [43], [44], a modular cylindrical frame, an on-board printed circuit board (PCB), and an experimental platform. We describe the detailed design of the wheel in the following sections.

The folded-HASEL actuator uses a 20- $\mu\text{m}$ -thick polyester film (L0WS, multiplastics) with an inner heat-sealing layer and corona-treated outer layer for ink adhesion. The actuator pouches are sealed using a heated tip temperature of 195 °C with a tip sealing speed of 400 mm/min. A small fill port is left unsealed in each pouch to provide a means to fill the pouches with a liquid dielectric during the last fabrication step. Electrodes are made from a carbon ink (CI-2051, Engineered Materials Systems, Inc.) and were deposited on the outer layer of the films using a screen-printing method. Finally, the pouches are filled with a dielectric silicone fluid (PSF-5cSt, Clearco) using a syringe and needle inserted into the fill port of the pouch. After filling, a soldering iron (WE1010NA, Weller) heated to 176 °C is used to seal the fill port. The complete fabrication process is described in detail in [43]. Each actuator has 12 pouches and folded into the dimension of 50 mm x 55 mm x 12 mm using 5 mm x 1 mm transfer tapes (924, 3M). Under the application of high voltage (HV) in 1–10 kV range on one electrode and a low voltage (LV) connection on the other, the actuator's electrodes zip together, displacing the liquid dielectric to the surrounding volumes inside the actuators, resulting in the overall change in the actuators' strokes [see Fig. 2(a)].

As can be seen from Fig. 2(c), the cylindrical frame consists of two circular rims with a diameter of 313.54 mm and 16 actuator pads (55 mm x 50 mm x 13 mm) and therefore is fully modular. The frame is three-dimensional (3-D) printed using 1.75 mm

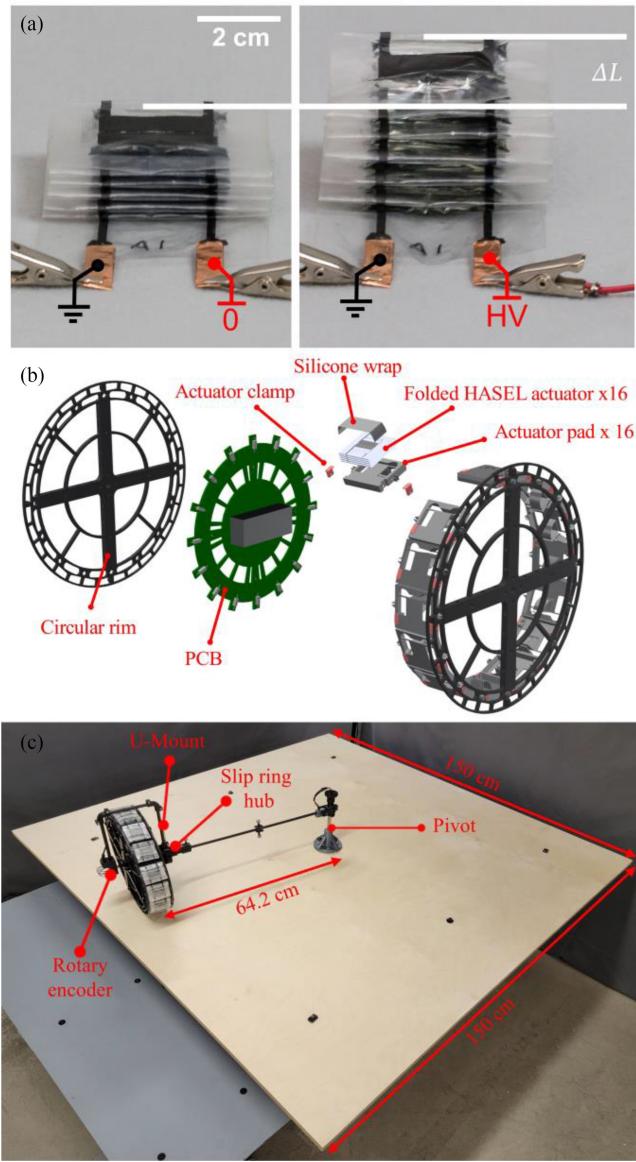


Fig. 2. (a) Folded-HASEL actuator activates under HV application. The actuators represent the soft components of the wheel. (b) 3-D render of the electrohydraulic rolling soft system driven by 16 folded-HASEL actuators. The wheel rolls on the cylindrical frame, not on the actuators. (c) Wheel platform has the rolling soft wheel mounted to a pivot arm via a U-mount. An absolute rotary encoder (EMS22A50, Bourns) and a motion capture system (not shown) are used for both characterization and feedback control of the wheel. The wheel rolls on a circular path with a radius 64.2 cm on a 150 cm x 150 cm square wooden platform. The platform is elevated 700 cm from the ground.

PRUSA PLA filament and a 3-D printer (I3 MK3S, PRUSA). A total of 16 actuator pads are placed radially around the edge of the rims such that they both hold the two rims together and allow the relaxed actuators to be 1 mm away from being in contact with the ground. This design allows the wheel to roll on the rigid frame and minimize the contact friction between the actuators and the ground. Each actuator pad surface lies orthogonally to the radial line of the cylindrical frame. Since there are 16 actuator pads, the angle between the two vectors that are orthogonal to two adjacent actuator pads' surfaces is  $360^\circ/16 = 22.5^\circ = \pi/8$  rad.

The PCB that commands and drives the actuators is sandwiched between the two circular rims of the wheel.

To simplify the experiments, the wheel is mounted to an axle that can pivot around the center of a platform [see Fig. 2(c)] This setup provides stability to the wheel and allows it to roll as long as required for experiments. The platform is a 1.52 m x 1.52 m x 0.019 m (60 in x 60 in x 0.75 in) Baltic-birch wood table elevated by 0.7 m from the ground using four 80/20 support columns (1515, 80/20 Inc.). The axle consists of an 8.5-mm-diameter 547-mm-long carbon-fiber rod (Rock West Composites), whose other end is mounted to the center of the platform but free to rotate both horizontally and vertically thanks to a pair of ceramic ball bearings (B15R8TH9C3, Boca Bearings) on the 12-mm aluminum pivot shaft and a pair of steel ball bearings (625ZZ, uxcell) at a 3-D-printed elbow (not shown). The axle is connected to the wheel's frame via a U-mount made of carbon fiber. One side of the U-mount connects to a 12-wire slip ring (ROB-13065, Sparkfun) (not shown) that provides electrical interfaces between the USB, power, ground, and the encoder's wires and the wheel's PCB. Without the slip ring, the wheel under rolling motion would cause the wires to twist and get tangled. The slip ring is connected to a second set of ceramic ball bearings (B15R8TH9C3, Boca Bearings) that allow the wheel to rotate about its geometric center (GC) with minimal friction. The other side of the U-mount aligns the rotary encoder's shaft with the wheel's GC (not shown).

### B. Electrical Design

The onboard PCB that drives 16 HASEL actuators contains three groups: the power delivery group, the driving and control group, and the sensing group. The block diagram of the three groups is shown in Fig. 3(a). The red arrows show the power line for all onboard electrical components, while the black arrows show the input-output relationships between every two components. To fit inside the frame of the wheel, the PCB has diameter of 250 mm, and its front and back 3-D models can be seen in Fig. 3(b) and (c).

The power delivery group consists of an 8-to-24 V step-up dc-dc converter (U3V50F24, Pololu), an 8-to-5 V step-down linear dropout regulator (REG113NA-5/3K, Texas Instruments), and an 8-to-3.3 V step-down linear dropout regulator (REG113NA-3.3/3K, Texas Instruments). The step-up dc-dc converter provides the required 24 V to the HV dc-dc converter (10A24-P30, Advanced Energy), which generates a constant 9 kV output. The 5 and 3.3 V regulators provide power to both the driving and control group and the sensing group.

The driving and control group consists of a microcontroller (Teensy 3.6, PJRC) that communicates with a personal computer via a full-speed Universal Serial Bus (USB 1.1). The microcontroller processes sensing data and transfer them to a personal computer, which, depending on the data, generates either open-loop or closed-loop commands. The commands are then transferred to the microcontroller for HV switching of the actuators.

The HV switching circuit (see Fig. 4) has 16 units of active charge-passive drain circuit to independently drive 16 actuators.

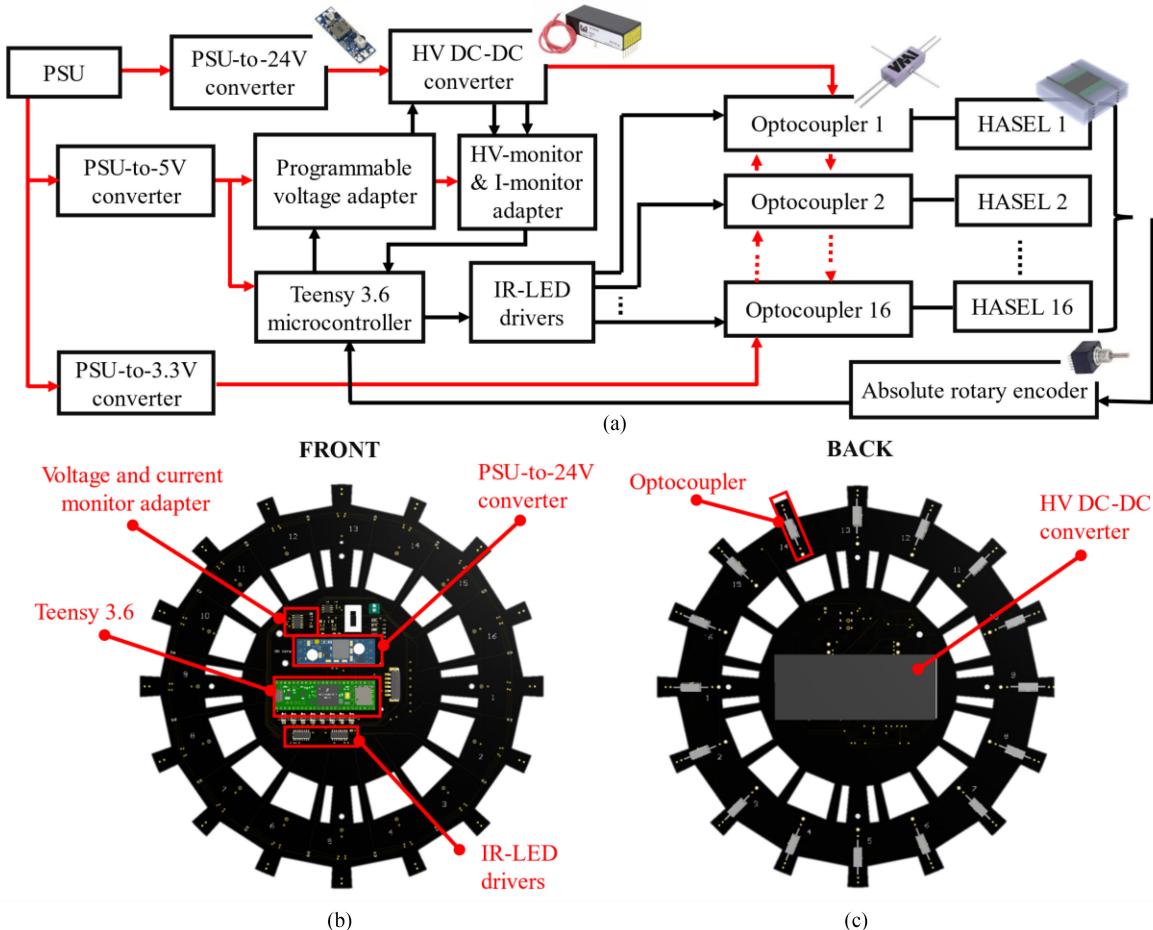


Fig. 3. Embedded system overview. (a) Block diagram of the electrical system, in which the power supply unit is not integrated. The onboard HV dc-dc converter can output up to 10 kV at 3 mA. The HV-tolerated optocouplers, each of which contains an optodiode and a 940 nm IR-LED, act as switches that connect or disconnect the HV dc-dc converter from the actuators depending on the amount of light received from the IR-LEDs. These LEDs are toggled based on a microcontroller and two IR-LED drivers. The red arrows indicate the power line for all electrical components, and the black arrows represent the input-output relationships between every two components. (b) Front view of the PCB contains all LV components. (c) Back view of the PCB contains all HV components. This design prevents HV components from damaging LV ones.

Each unit contains an optocoupler (shown in gray box) that consists of a HV-tolerant optodiode (OZ100SG, Voltage Multiplier Inc.) and an HV tolerated drain resistor ( $R_d$ ). To charge the actuators, the optodiodes change their impedance depending on the corresponding infrared light emitting diodes (IR-LEDs) (L1IZ-0940000000000, Lumileds). The 16 IR-LEDs are toggled by digital pins on the microcontroller (Teensy 3.6, PJRC) via two IR-LED drivers (TPIC6C596, Texas Instrument). When the IR-LEDs are turned OFF, depending on the value, the resistor can influence the rate at which charges on the electrodes neutralize, thus controlling how quickly the actuator drains themselves. The value of the drain resistor is determined in Section II-C. The optocouplers are placed radially at the edge of the PCB [see Fig. 3(c)] to facilitate physical connection to the HV electrodes of the actuators.

Finally, the sensing group consists of a single-supply operational amplifier (TLV4172IDR, Texas Instruments) for HV and current monitoring, and an absolute rotary encoder (EMS22A50, Bourns). While the HV and current monitors are critical for HV fail-safe design, the absolute rotary encoder is used for the wheel's speed regulation. The encoder sends the digital bytes

corresponding to the wheel's absolute angular position to the microcontroller, which then forwards the converted data via the USB to the computer for further processing.

Due to the overall constraints of the PCB space and the placements of electrical components (to guarantee appropriate distances from the HV points), the PCB has asymmetric component placements, resulting in the wheel's COM and GC not the same. The wheel's frame contains four 3D printed pins on which custom weights can be added, allowing the wheel's COM to be the same as its GC; from trial-and-error, we select a combination of 0.88 and 3.58 g of weights on the appropriate pins (not shown).

### C. Optimizing Actuator's Performance With Respect to Silicone Wrap and Drain Resistor

When an actuator is placed upright and when the HV is switched OFF, the actuator relies on its own inertia to return to its contracted state. However, an actuator attached on a rolling wheel takes significantly longer time to contract to its original state, since gravity keeps the actuators in the expanded state beyond the wheel's rim when they are upside down. This

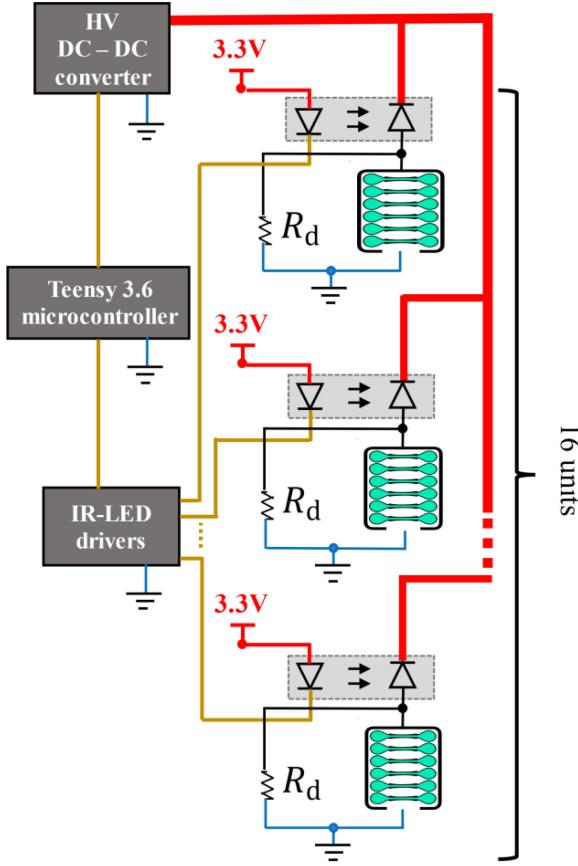


Fig. 4. Simplified schematic of 16 active charge–passive drain HV units for 16 actuators. The term “active charge” comes from the ability to toggle the IR-LEDs to adjust the optocoupler’s impedance to power the actuators. The term “passive drain” describes the use of HV-tolerated resistors  $R_d$  to passively discharge the corresponding actuators when the optocouplers are turned OFF. The 16 units are connected in parallel to a single HV dc-dc converter and controlled by two IR-LED drivers.

behavior prevents the actuator from exerting sufficient output forces in the next activation-deactivation cycle and the wheel from rolling smoothly on the platform.

To prevent the contracted actuators from extending beyond the wheel’s rim, and to help the actuator contract faster when HV is switched OFF, a 130 mm x 24 mm rectangular silicone wrap (00-30 Ecoflex, Smooth-On) is wrapped (with 50% pre-stretch) around each actuator. Since the choice of a drain resistor also affects how quickly each actuator relaxes, we evaluate an actuator’s performance with respect to both different silicone wrap’s thickness and drain resistor [see Fig. 5]. Fig. 5(a) and (b) shows the experiment setup that quantifies the displacement’s fall (contraction) time of an actuator. The inverted actuator is initially activated under 8 kV. When the HV optocoupler is switched OFF, the neutralization of electrostatic charges thanks to the drain resistor and the restoring force from the silicone wrap cause the actuator to contract, removing the blockage of the laser beam and allowing it to shine light to a photoresistor (SEN-09088, Sparkfun), whose change of resistance is sensed by a microcontroller (Arduino Uno Rev3, Arduino). The process is repeated five times for each combination of silicone wrap and drain resistor. The average duration from the time the HV

optocoupler is switched OFF to the time the laser beam signal is registered into the microcontroller is recorded in Fig. 5(c). As expected, for the same drain resistor, thinner silicone wraps produce less restoring force on the actuator, resulting larger fall time. For the same silicone wrap, larger drain resistors restrict flow of charges more, causing the actuator to relax slower. Therefore, the combination of a resistor of  $200\text{ M}\Omega$  and silicone wrap with  $800\text{ }\mu\text{m}$  thickness results in the best fall time of 0.53 s, while the combination of a resistor of  $500\text{ M}\Omega$  and a wrap with thickness of  $400\text{ }\mu\text{m}$  results in the worst fall time of 2.27 s.

We also evaluate how the choice of silicone wrap and drain resistor affects the wheel’s ability to roll by quantifying the change of angle due to the reaction force between the ground and the actuator as it expands. The wheel is initially at a resting angle such that the actuator third is placed  $11.25^\circ$  away from its vertical position. The actuator is then activated for 3 s and the wheel’s change in angle between its initial angle and final angle is recorded by the absolute rotary encoder. The process is repeated five times for each combination of drain resistor and wrap [illustration shown in Fig. 5(d) and (e)]. Fig. 5(f) shows the average change in angle of the wheel with respect to different drain resistors and wrap thickness. From Fig. 5(f), the results between the wrap thicknesses of  $400$  and  $600\text{ }\mu\text{m}$  are significantly close to each other, and they are better than those from the wrap thickness of  $800\text{ }\mu\text{m}$ . Fig. 5(f) also suggests that the influence of the wrap’s thickness on the angle traveled of the wheel is nonlinear. Since the actuators’ top surface are placed very close (1mm) to the edge of the rim, a  $600\text{-}\mu\text{m}$ -thick silicone wrap is more likely to protrude beyond the wheel’s rim after repeated cycles of expansion and contraction than a  $400\text{-}\mu\text{m}$ -thick silicone wrap. A  $400\text{-}\mu\text{m}$ -thick silicone wrap is preferred since the wrap’s protrusion creates significant frictions and reduce the performance of the wheel. From the results in both Fig. 5(c) and (f), it is therefore optimal to select a resistance value of  $200\text{ M}\Omega$  for shortest fall time while not sacrificing good actuator’s force and stroke. The silicone wrap thickness of  $400\text{ }\mu\text{m}$  is selected for a reasonably fast fall time but with much better actuator’s output force and stroke.

### III. HYBRID DYNAMIC MODEL

#### A. Dynamics Modeling of the Rolling Soft Wheel

The dynamic modeling of the wheel of radius  $r$  and pivot arm of length  $L$  is based on [45]. Fig. 6(a) shows the three reference frames required for the dynamic modeling of the rolling motion. The notation convention can be briefly described as follows: a variable with an arrow is a vector, a variable with a hat is a unit vector, a variable without a hat or an arrow is a scalar value, a variable with a single dot is a derivative of that variable, and a variable with two dots is a second derivative of that variable.

The platform-fixed frame  $\mathcal{F}$  has a coordinate system whose origin is  $O$ . Its orthogonal basis  $\{\hat{E}_x, \hat{E}_y, \hat{E}_z\}$  is defined as follows:  $\hat{E}_x$  points along the first edge of the platform,  $\hat{E}_z$  points in the direction of gravity, and  $\hat{E}_y = \hat{E}_z \times \hat{E}_x$ . The reference

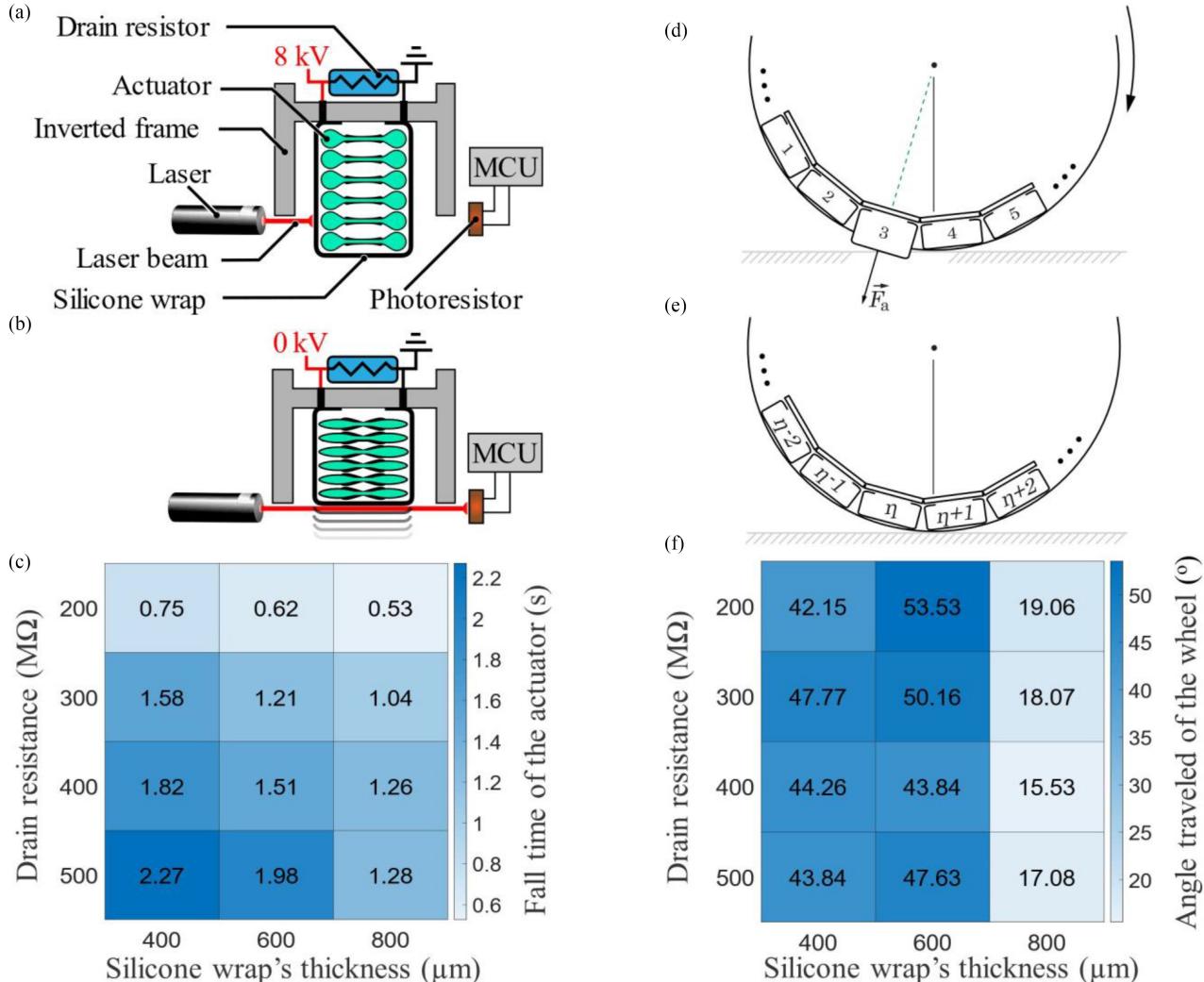


Fig. 5. Influences of silicone wrap and the HV-tolerated drain resistors on the actuator's fall time and how much force an actuator exerts on the ground. The amount of force applied is determined indirectly by the amount of angle traveled by the wheel from a predetermined initial angular position. (a) Experimental setup begins with the actuator fully activated under 8 kV, and the activated actuator blocks the beam from a laser pointer. (b) When HV is switched OFF, the drain resistor discharges the actuator, and the silicone wrap pushes the actuator back to its contracted state. A microcontroller (Arduino Uno Rev3, Arduino) determines the fall time as the duration between the HV switching OFF and the instant the laser beam hits the photoresistor. (c) Fall time (s) of the actuator with respect to different pair of drain resistors ( $M\Omega$ ) and silicone wrap thickness ( $\mu m$ ). (d) Wheel activates the actuator third from rest and begins rolling. This actuator is initially positioned at  $11.25^\circ$  away from its vertical position. (e) Wheel rolls to a complete stop. (f) Amount of angle traveled of the wheel ( $^\circ$ ) due to the force generated by actuator third with respect to different pairs of drain resistances and thicknesses of silicone wraps.

frame  $\mathcal{P}$  is fixed to the pivot has a coordinate system whose origin is A. We denote B as the point on the wheel in contact with the platform. Its orthogonal basis  $\{\hat{p}_1, \hat{p}_2, \hat{p}_3\}$  is defined as follow:  $\hat{p}_1$  points along the direction of  $\overrightarrow{AB}$ ,  $\hat{p}_3$  points along the direction of  $\hat{E}_z$ , and  $\hat{p}_2 = \hat{p}_3 \times \hat{p}_1$ . The coordinate system fixed to reference frame  $\mathcal{P}$  rotates an angle of  $\psi$  with respect to the translated coordinate system fixed to reference frame  $\mathcal{F}$ , with  $\psi$  is the angle between  $\hat{p}_1$  and  $\hat{E}_x$  [or between  $\hat{p}_2$  and  $\hat{E}_y$  as seen in Fig. 6(a)]. Finally, the reference frame  $\mathcal{R}$  is fixed to the HASEL rolling wheel with a coordinate system of origin C. We denote P as an arbitrary point on the rim of the wheel. Its orthogonal basis  $\{\hat{e}_1, \hat{e}_2, \hat{e}_3\}$  is defined as follow:  $\hat{e}_1$  points along the direction of  $\hat{p}_1$ ;  $\hat{e}_2$  points along the direction of  $\overrightarrow{CP}$ ; and  $\hat{e}_3 := \hat{e}_1 \times \hat{e}_2$ . The coordinate system fixed to reference frame  $\mathcal{R}$  rotates an angle of  $\varphi$  with respect to the coordinate system

fixed to reference frame  $\mathcal{P}$  as the wheel rotates around point C, with  $\varphi$  is the angle between  $\hat{e}_2$  and  $\hat{p}_2$  (shown as dashed line pointed from C).

${}^{\mathcal{F}}\vec{\omega}^{\mathcal{P}}$  is the angular velocity of frame  $\mathcal{P}$  with respect to an observer in frame  $\mathcal{F}$ .  ${}^{\mathcal{F}}\vec{\omega}^{\mathcal{R}}$  is the angular velocity of frame  $\mathcal{R}$  with respect to an observer in frame  $\mathcal{F}$ .  $\vec{r}_A$  is the position of point A relative to point O in  $\mathcal{F}$ .  $\vec{r}_C - \vec{r}_B$  is the position of point C (COM of the wheel) relative to point B (contact point between the wheel and the platform) in  $\mathcal{P}$ .

Assuming the wheel rolls without slip, the velocity of contact point B of rigid body as observed in fixed frame  $\mathcal{F}$

$${}^{\mathcal{F}}\vec{v}_B^{\mathcal{R}} = \vec{0}. \quad (1)$$

The velocity  ${}^{\mathcal{F}}\vec{v}_C$  of the wheel's COM (point C) as viewed from fixed frame  $\mathcal{F}$  is related to the velocity  ${}^{\mathcal{F}}\vec{v}_B^{\mathcal{R}}$  of point B of

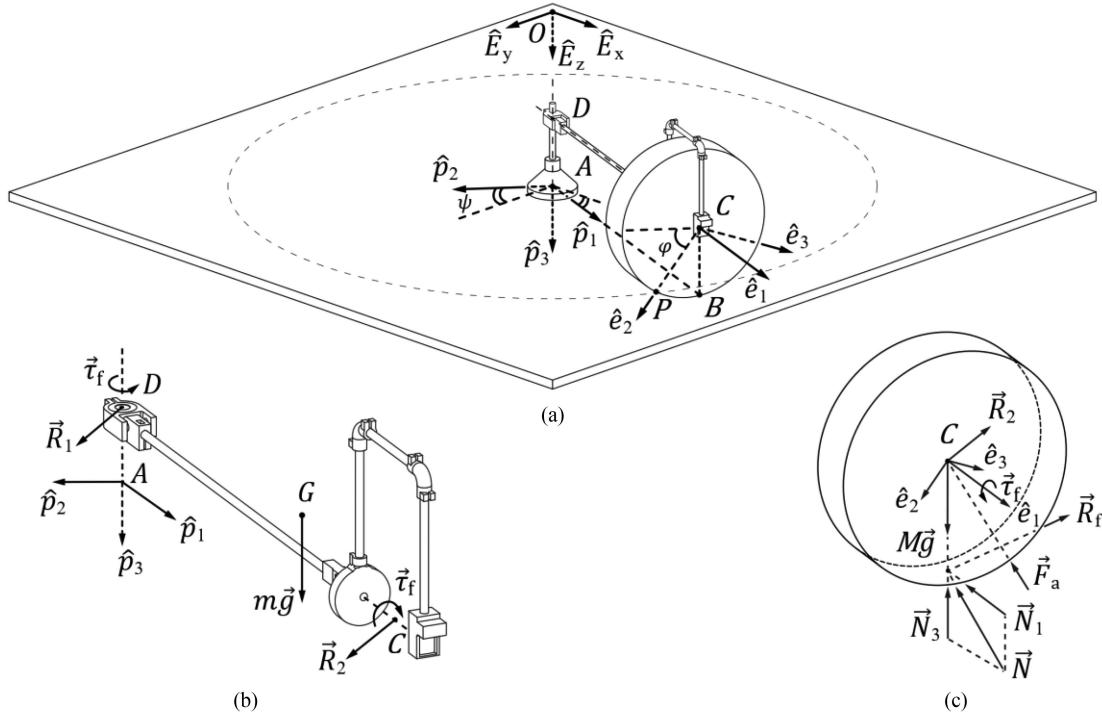


Fig. 6. Kinematic and dynamic presentations of the rolling platform. (a) Reference frames of the platform, the pivot arm with U-mount, and the electrohydraulic rolling soft wheel are presented to describe the wheel's rolling motion on a circular path. (b) FBD of the pivot arm with U-mount that secures and guides the wheel's rolling motion. The pivot arm with U-mount has only rotational motion around axis  $\hat{p}_3$ . (c) FBD of the wheel, which has both translational and rotational motions.

reference frame  $\mathcal{R}$  as observed in  $\mathcal{F}$

$$\mathcal{F}\vec{v}_C = \mathcal{F}\vec{v}_B^{\mathcal{R}} + \mathcal{F}\vec{\omega}^{\mathcal{R}} \times (\vec{r}_C - \vec{r}_B) \quad (2)$$

$$\mathcal{F}\vec{v}_C = r\dot{\varphi}\hat{p}_2. \quad (3)$$

We denote point  $G$  as the COM of the pivot arm. Due to its geometry, point  $G$  does not locate on the pivot arm [see Fig. 7(b)]. Since we assume the wheel rolls without slip, we can establish the kinematic constraints as follows:

$$\vec{r}_C = L \cdot \hat{p}_1 + z_{DA} \cdot \hat{p}_3 \quad (4)$$

$$\mathcal{F}\frac{d}{dt}(\vec{r}_C) = \mathcal{P}\frac{d}{dt}(\vec{r}_C) + \mathcal{F}\vec{\omega}^{\mathcal{P}} \times \vec{r}_C \quad (5)$$

$$\mathcal{F}\vec{v}_C = \dot{\psi} L \cdot \hat{p}_2 \quad (6)$$

where  $z_{DA}$  is the distance between point  $D$  and  $A$ , both of which are located on the pivot. We know that  $\mathcal{F}\vec{v}_C = r\dot{\varphi}\hat{p}_2$ , therefore

$$\dot{\psi} L \cdot \hat{p}_2 = r\dot{\varphi}\hat{p}_2 \quad (7)$$

$$\dot{\psi} = \frac{r}{L}\dot{\varphi}. \quad (8)$$

a) *Kinetics of the pivot arm with U-mount:* The forces and torques associated with the free body diagram (FBD) of the U-mount [shown in Fig. 7(b)] are as follows:

$\vec{R}_1$ : Reaction force at the pivot elbow

$$\vec{R}_1 = R_{11} \cdot \hat{p}_1 + R_{12} \cdot \hat{p}_2 \quad (9)$$

$m\vec{g}$ : Weight of the pivot arm (with U-mount) along  $\hat{p}_3$

$\vec{R}_2$ : Resultant reaction force from the wheel on the arm

$$\vec{R}_2 = R_{21} \cdot \hat{p}_1 + R_{22} \cdot \hat{P}_2 + R_{23} \cdot \hat{p}_3 \quad (10)$$

$\vec{\tau}_f$ : Pure bearing friction torque opposite to  $\hat{p}_3$ .

$$\vec{\tau}_f = \tau_f \cdot \hat{p}_3 \quad (11)$$

Since there are no translational motion for the pivot arm, we can ignore the Euler's first law. According to Euler second law, we have the following relationship between the moment  $\vec{M}_D$  relative to point  $D$  and the angular momentum  $\mathcal{F}\vec{H}_D$  relative to point  $D$  in reference frame  $\mathcal{F}$

$$\vec{M}_D = \mathcal{F}\frac{d}{dt}(\mathcal{F}\vec{H}_D) \quad (12)$$

$$\vec{M}_D = (\vec{r}_G - \vec{r}_D) \times mg \cdot \hat{p}_3 + (\vec{r}_C - \vec{r}_D) \times \vec{R}_2 - \tau_f \cdot \hat{p}_3 \quad (13)$$

where  $\vec{r}_G - \vec{r}_D$  is the position of point  $G$  relative to point  $D$  in  $\mathcal{P}$ ,  $\vec{r}_C - \vec{r}_D$  is the position of point  $C$  relative to point  $D$  in  $\mathcal{P}$ . Let us consider the angular momentum of the pivot arm with U-mount

$$\mathcal{F}\vec{H}_D = \mathcal{P}I_D \cdot \mathcal{F}\vec{\omega}^{\mathcal{P}} \quad (14)$$

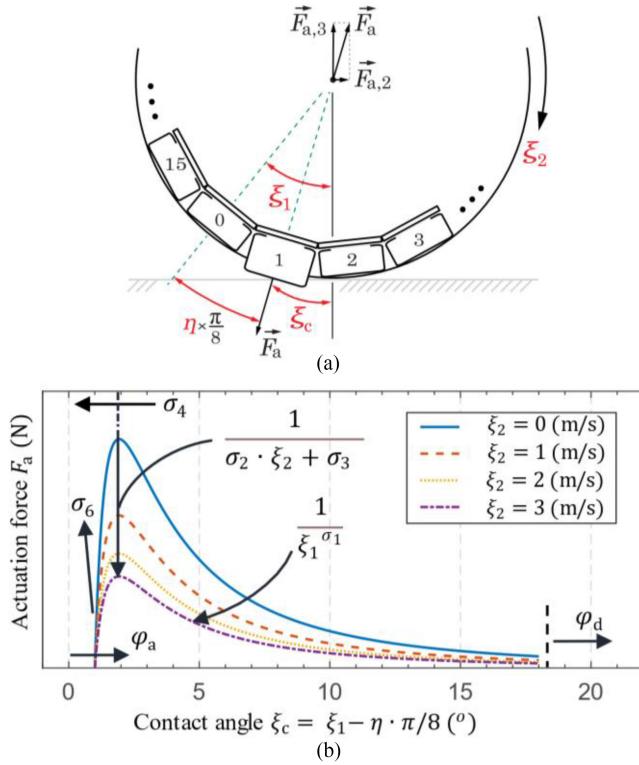


Fig. 7. Description of the actuation force  $F_a$ , which is assumed to traverse through the COM of the rolling soft wheel. (a) Illustration of the contact angle  $\xi_c$  of actuator first and how it decomposes the force to vertical and horizontal components. (b) Generic profile of the actuation force  $F_a$  with respect to the contact angle  $\xi_c$  and angular speed  $\xi_2$  of an arbitrary actuator  $\eta$ th. The force function is assumed to have an exponential form and its profile depends on six parameters  $\{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6\}$ .

where  ${}^P I_D$  is the moment of inertia tensor of the pivot arm (with U-mount) relative to point D. By expanding (14), we have

$${}^F \vec{H}_D = \begin{bmatrix} I_{11} & I_{12} & I_{13} \\ I_{21} & I_{22} & I_{23} \\ I_{31} & I_{32} & I_{33} \end{bmatrix} \cdot \begin{bmatrix} 0 \hat{p}_1 \\ 0 \hat{p}_2 \\ \psi \hat{p}_3 \end{bmatrix}. \quad (15)$$

Applying rate of change transport theorem to (15), we have

$$\frac{d}{dt} ({}^F \vec{H}_D) = {}^P \frac{d}{dt} ({}^F \vec{H}_D) + {}^F \vec{\omega}^P \times {}^F \vec{H}_D. \quad (16)$$

Equating (13) and (16) based on (12) and without the detailed derivation, we obtain the following:

$$0 = I_{13} \ddot{\psi} - I_{23} \dot{\psi}^2 \quad (17)$$

$$-m g x_{GA} - L R_{23} = I_{23} \ddot{\psi} + I_{13} \dot{\psi}^2 \quad (18)$$

$$L R_{22} - \tau_f = I_{33} \ddot{\psi}. \quad (19)$$

b) *Kinetics of the HASEL rolling wheel:* For simplicity, the wheel's geometry can be treated as a cylinder. Based on the FBD [see Fig. 7(c)], the following forces and torques are involved:

$M\vec{g}$ : Weight of the wheel along  $\hat{p}_3$

$\vec{N}$ : Reaction force of the ground on the wheel

$$\vec{N} = N_1 \cdot \hat{p}_1 + N_3 \cdot \hat{p}_3 \quad (20)$$

$\vec{R}_2$ : Resultant, simplified reaction force from the pivot arm on the wheel

$$-\vec{R}_2 = -R_{21} \cdot \hat{p}_1 - R_{22} \cdot \hat{p}_2 - R_{23} \cdot \hat{p}_3 \quad (21)$$

$\vec{\tau}_f$ : Pure bearing friction torque opposite to  $\hat{e}_1$

$R_f$ : Static friction force between the wheel and the ground opposite to direction  $\hat{p}_2$

$F_a$ : Actuator force exerting on the wheel

$$\vec{F}_a = F_{a,2} \cdot \hat{p}_2 + F_{a,3} \cdot \hat{p}_3. \quad (22)$$

Since the soft wheel involves both translational and rotational motion, two Euler's laws are required to describe its dynamics. Applying Euler first law, we have the net forces equals the wheel's mass  $M$  times the acceleration  ${}^F \vec{a}_C$  of point C in the reference frame  $\mathcal{F}$

$$\sum \vec{F} = M \cdot {}^F \vec{a}_C. \quad (23)$$

Without detailed derivation, (23) can be expanded as follows:

$$-R_{21} + N_1 = M r \dot{\psi} \dot{\varphi} \quad (24)$$

$$-R_{22} - R_f + F_{a,2} = M r \ddot{\varphi} \quad (25)$$

$$Mg - R_{23} + N_3 + F_{a,3} = 0. \quad (26)$$

According to Euler second law, we have the following relationship between the moment  $\vec{M}_C$  relative to point C and the angular momentum  ${}^F \vec{H}_C$  relative to point C in reference frame  $\mathcal{F}$

$$\vec{M}_C = {}^F \frac{d}{dt} ({}^F \vec{H}_C). \quad (27)$$

First, we consider the left side of (27)

$$\vec{M}_C = (\vec{r}_B - \vec{r}_C) \times (\vec{N} + \vec{R}_f) + \vec{\tau}_f. \quad (28)$$

Then, we consider the right side of (27)

$${}^F \vec{H}_C = {}^R I_C \cdot {}^F \vec{\omega}^R \quad (29)$$

where  ${}^R I_C$  is the moment of inertia tensor of the pivot arm with U-mount relative to the COM (point C) of the wheel. While the 3-D modeling of the HASEL soft wheel using SolidWorks 2019 (SolidWorks) show that the product of inertias (POI) for  ${}^R I_C$  are nonzero, these values are negligible compared to the diagonal components. For this reason, we assume that  $\{\hat{e}_1, \hat{e}_2, \hat{e}_3\}$  are the principal axis basis vectors and COM is the reference point and the POI's are treated as zeros. Given

$${}^F \vec{\omega}^R = \dot{\psi} \cdot \hat{P}_3 + \dot{\varphi} \cdot \hat{p}_1 \quad (30)$$

$${}^F \vec{\omega}^R = \dot{\varphi} \cdot \hat{e}_1 + \dot{\psi} \sin(\varphi) \cdot \hat{e}_2 + \dot{\psi} \cos(\varphi) \cdot \hat{e}_3. \quad (31)$$

We have

$${}^F \vec{H}_C = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix} \cdot \begin{bmatrix} \dot{\varphi} \hat{e}_1 \\ \dot{\psi} \sin(\varphi) \hat{e}_2 \\ \dot{\psi} \cos(\varphi) \hat{e}_3 \end{bmatrix}. \quad (32)$$

Applying rate of change transport theorem, we have

$$\mathcal{F} \frac{d}{dt} (\mathcal{F} \vec{H}_C) = \mathcal{P} \frac{d}{dt} (\mathcal{F} \vec{H}_C) + \mathcal{F} \vec{\omega}^R \times \mathcal{F} \vec{H}_C. \quad (33)$$

Equating (28) and (33) with respect to the orthogonal basis vectors, we have

$$r R_f - \tau_f = I_1 \ddot{\varphi} + (I_3 - I_2) \psi^2 \sin(\varphi) \cos(\varphi) \quad (34)$$

$$r N_1 \cos(\varphi)$$

$$= I_2 (\ddot{\psi} \sin(\varphi) + \dot{\psi} \dot{\varphi} \cos(\varphi)) \\ + (I_1 - I_3) \dot{\psi} \dot{\varphi} \cos(\varphi) \quad (35)$$

$$- r N_1 \sin(\varphi) \\ = I_3 (\ddot{\psi} \cos(\varphi) - \dot{\psi} \dot{\varphi} \sin(\varphi)) \\ + (I_2 - I_1) \dot{\psi} \dot{\varphi} \sin(\varphi). \quad (36)$$

From (17), (19), (25), and (34), and the kinematic constraint (8), we obtain the following differential equation:

$$L r F_2 - r \tau_f - L \tau_f \\ = L I_1 \ddot{\varphi} + r (I_3 - I_2) \frac{I_{13}}{I_{23}} \ddot{\varphi} \sin(\varphi) \cos(\varphi) \quad (37) \\ + \left( L M r^2 + I_{33} \frac{r^2}{L} \right) \ddot{\varphi}.$$

The equation contains only one variable  $\varphi$  but is nonlinear due to the term  $\ddot{\varphi} \sin(\varphi) \cos(\varphi)$ . However, since  $I_3 - I_2 \approx 0$ , we can drop this term. The friction torques of the bearing can be treated as constants and its direction depend on the direction of the wheel's angular velocity. Therefore, (37) can be simplified as

$$\left( L I_1 + L M r^2 + I_{33} \frac{r^2}{L} \right) \ddot{\varphi} \\ = L r F_{a,2} - (r \tau_f + L \tau_f) \operatorname{sign}(\dot{\varphi}). \quad (38)$$

From (38), the continuous-time state-space model during activation periods can be written as follows:

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} = \begin{bmatrix} \xi_2 \\ \frac{L r F_{a,2} - (r \tau_f + L \tau_f) \operatorname{sign}(\xi_2)}{L I_1 + L M r^2 + I_{33} \frac{r^2}{L}} \end{bmatrix} \quad (39)$$

where  $\xi_1 := \varphi \in \mathbb{R}$  is the angular position of the wheel, and  $\xi_2 := \dot{\varphi} \in \mathbb{R}$  is the angular velocity of the wheel.

### B. Hybrid State-Space Model of the Electrohydraulic Rolling Soft Wheel

The variable  $\xi_1$  both represents the angular position of the rolling soft wheel with respect to time and tracks the relative position of 16 actuators with respect to the ground. We index 16 actuators from 0 to 15. The angle  $\xi_1$  with the value between 0 and  $\pi/8$  rad corresponds to the actuator 0th; the angle  $\xi_1$  with the value between  $i \cdot \pi/8$  and  $(i + 1) \cdot \pi/8$  rad corresponds to the actuator  $i$ th. Based on the value of  $\xi_1$ , we can determine current the actuator index,  $\eta$ , associated with the actuator that is within 0 and  $\pi/8$  rad from vertical plane. This plane is defined by three points A, B, and C as seen in Fig. 8(a). In other words,  $\eta = i$  when the actuator  $i$ th is in contact with the ground. Based on this mapping, we can determine the actuator that is in contact with

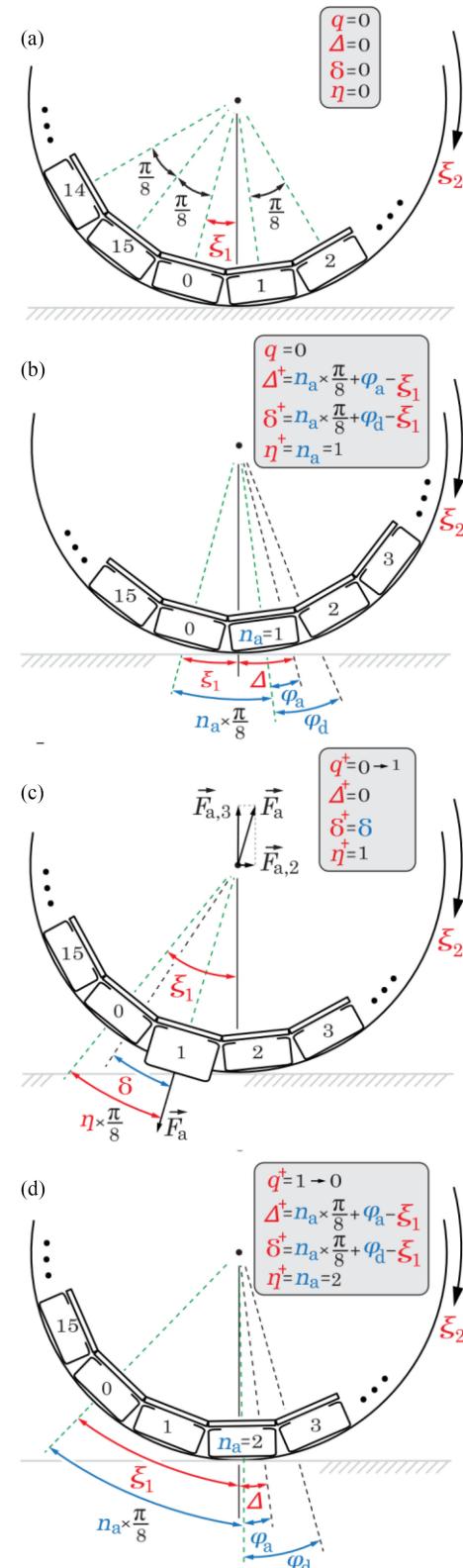


Fig. 8. Typical operation of electrohydraulic rolling soft wheel. (a) Wheel has initial angular position  $\xi_1$  and angular speed  $\xi_2$ . (b) Current actuator  $\eta$ th, the activation- and deactivation-angle down counters ( $\Delta$  and  $\delta$ , respectively) are updated. (c) Actuator oneth activates when  $\Delta$  reaches 0, while  $\delta$  continues to count down. (d) Actuator oneth deactivates when  $\delta$  reaches 0. The current actuator  $\eta$ th, the activation- and deactivation-angle down counters ( $\Delta$  and  $\delta$ , respectively) are updated. The process then repeats.

the ground by rounding down the result of the division between the wheel angle  $\xi_1$  and  $\pi/8$  rad

$$\eta = \xi_1 / (\pi/8) . \quad (40)$$

Based on the wheel angle  $\xi_1$  and the contact actuator index  $\eta$ , we define the contact angle as the current angle of the actuator  $\eta$ th with respect to the vertical plane that is defined by three points  $A$ ,  $B$ , and  $C$  in Fig. 6. Therefore,  $\xi_c$  ranges from 0 to  $\pi/8$  rad

$$\xi_c = \xi_1 - \eta \cdot \pi / 8. \quad (41)$$

When and for how long the actuation force is applied to the wheel by each actuator depend on three inputs. The activation angle,  $\varphi_a$ , is the contact angle  $\xi_c$  of the actuator  $\eta$ th that the actuator  $\eta$ th activates. Based on this definition,  $\varphi_a \in [0, \pi/8]$ . The deactivation angle of actuator  $\eta$ th,  $\varphi_d$ , is the contact angle  $\xi_c$  of the actuator  $\eta$ th that the actuator  $\eta$ th deactivates. Therefore, we have  $\varphi_d \in [0, \pi/8]$  and  $\varphi_d > \varphi_a$ . Finally, the target actuator,  $n_a$  is the next actuator to be activated. Therefore, we have the following input:

$$u = \begin{bmatrix} n_a \\ \varphi_a \\ \varphi_d \end{bmatrix}. \quad (42)$$

To conveniently track the status of the actuator  $\eta$ th, we introduce a state  $q \in \{0, 1\}$ , with  $q = 0$  means the actuator  $\eta$ th is not activated, and  $q = 1$  means the actuator  $\eta$ th is activated. Since the wheel's performance depends on the contact angles at which the HV optocoupler is turned ON and at which the HV optocoupler is turned OFF, we introduce two additional nonphysical states to track these angles: activation-angle down counter  $\Delta$  and deactivation-angle down counter  $\delta$ . The activation-angle down counter decrements from the last moment  $q$  switches from 1 to 0 to the next moment  $q$  switches from 0 to 1. In other words, this variable decreases its value, at which the HV optocoupler of the current actuator  $\eta$ th just switches OFF, to zero. As soon as zero is reached, the HV optocoupler of the next actuator, determined by input  $n_a$ , is turned ON. The deactivation-angle down counter begins decreasing when the last time  $q$  switches from 1 to 0 to zero. At this moment, the zero-valued deactivation-angle down counter signals the wheel to switch  $q$  from 1 to 0 again, and the process repeats. These down counters decrement at the same rate as the angular velocity as follow:  $\dot{\Delta} = -(1 - q) \xi_2$  and  $\dot{\delta} = -\xi_2$ .

The force applied between the wheel and the ground is neither instantaneous nor constant over its activation duration. We assume that the actuation force  $F_a$  travels through the COM of the wheel, as can be seen from Fig. 7(a); the contact angle  $\xi_c$  therefore can decompose this force to the vertical and horizontal components. Besides, the force is also dependent on the actuation status  $q$ , the activation angle  $\varphi_a$ , and the angular speed  $\xi_2$ . We assume the actuation force function to have the following form:

$$F_a(\xi_c, \xi_2, q, \varphi_a) = \frac{1}{\xi_c^{\sigma_1}} \frac{1}{\sigma_2 \xi_2 + \sigma_3} e^{-\frac{(\ln(\xi_c) - \sigma_4)^2}{\sigma_5^2}} q \sigma_6 (\xi_c - \varphi_a). \quad (43)$$

The profile of the actuation force  $F_a$  depends on the six parameters  $\sigma_1$ ,  $\sigma_2$ ,  $\sigma_3$ ,  $\sigma_4$ ,  $\sigma_5$ , and  $\sigma_6$ , four states  $\xi_1$ ,  $\xi_2$ ,  $q$ , and  $\eta$ , and one input  $\varphi_a$ . It can be seen from Fig. 7 that the profile of the actuator force can be adjusted by changing the values of the parameters and the states. The arrows indicate the modification of the profile of the force function due to the increases of the corresponding parameters, states, or inputs. Specifically, increasing  $\varphi_a$  causes the actuator to activate a larger contact angle, increasing  $\sigma_6$  increases the left-side slope of the force function, increasing  $\sigma_1$  makes the right-side slope of the force function more curvy, increasing  $\sigma_2$ ,  $\xi_2$ , or  $\sigma_3$  reduces the peak force, and increasing the deactivation angle  $\varphi_d$  increases the angle at which the generated force is cut OFF. The general equation of the force function is assumed from the observation that the force must increase rapidly from 0 N at the activation angle, reach its peak, and diminish gradually as the wheel's angle increases. Since the rate of the actuator's stroke is finite, the peak force it generates should decrease as the wheel's speed increases, hence the exponential function. While the activation angle  $\varphi_a$  determines at which angle the actuator force to rise from 0 N, the deactivation angle  $\varphi_d$  determines the angle at which the force output is cut OFF (see Fig. 7). The input  $\varphi_d$  is not explicitly shown in the force function  $F_a(\xi_c, \xi_2, q, \varphi_a)$ , yet it determines when  $q$  switches from 1 to 0, as it is used to update the deactivation-angle down counter shown in (50).

Since  $F_{a,2}$  is the magnitude of  $\hat{p}_2$  component of  $\vec{F}_a$ , we have

$$F_{a,2} = F_a \cdot \sin(\xi_c). \quad (44)$$

The state-space model is continuous in the entire durations of activation or deactivation of an actuator. Therefore, during these periods, the actuation status  $q$  and the actuator of interest  $\eta$  do not change their states ( $\dot{q} = 0$ ,  $\dot{\eta} = 0$ ). We can re-write the continuous state-space model (62) of the HASEL wheel as follows:

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{q} \\ \dot{\Delta} \\ \dot{\delta} \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} \xi_2 \\ \frac{L r (F_a \sin(\xi_c) - (r \tau_f + L \tau_f) \text{sign}(\xi_2))}{L I_1 + L M r^2 + I_{33} \frac{r^2}{L}} \\ 0 \\ -(1 - q) \xi_2 \\ -\xi_2 \\ 0 \end{bmatrix}. \quad (45)$$

What happen at the moments  $q$  switches between 0 and 1 can be modeled using the hybrid dynamic framework introduced in [46] and [47]. Given state  $x \in \mathbb{R}^n$ , input  $u \in \mathbb{R}^m$ , and output  $y \in \mathbb{R}^p$ , the framework has the following form:

$$\mathcal{H} = (\mathbf{C}, f, \mathbf{D}, g) \quad (46)$$

$$\mathcal{H} = \begin{cases} \dot{x} = f(x) & (x, u) \in \mathbf{C} \\ x^+ = g(x, u) & (x, u) \in \mathbf{D} \\ y = h(x) \end{cases} \quad (47)$$

where  $f$  is the flow map,  $\mathbf{C}$  is the flow set,  $g$  is the jump map, and  $\mathbf{D}$  is the jump set. The flow component of the model,  $\dot{x} = f(x)$ , is already introduced in (45). The jump component,  $x^+ = g(x, u)$  represents a discrete jump of the state  $x$  to  $x^+$  based on the function  $g(x, u)$ . The superscripts + indicates that

the states  $x$  change instantly into  $x^+$ . The function  $g(x, u)$  serves two purposes: It is in which the inputs are implemented to update the states of the wheel and toggle the actuator status  $q$ . The observation function  $y = h(x)$  determines how six states can be constructed from sensor data. The following section explains what happen to all six states during jumps.

First, the jump map cannot discretely change the state of the wheel's angular position and angular velocity; therefore  $\xi_1$  and  $\xi_2$  remain constant during jumps. Second, the status of the actuator of interest  $q$  is always binary (0 or 1). Therefore,

$$q^+ = 1 - q. \quad (48)$$

Third, the updated activation-angle down counter is calculated based on the sum of the activation angle of the next actuator and its actuator index multiplied by  $\pi/8$  rad (the angle between every two adjacent actuators), minus the current angular position of the wheel as follows:

$$\Delta^+ = q \left( \varphi_a + n_a \left( \frac{\pi}{8} \right) - \xi_1 \right). \quad (49)$$

Fourth, the updated deactivation-angle down counter is calculated based on the addition of the deactivation angle of the next actuator and its actuator index multiplied by  $\pi/8$  rad, minus the current angular position of the wheel as follows:

$$\delta^+ = \varphi_d + n_a \left( \frac{\pi}{8} \right) - \xi_1. \quad (50)$$

The main difference between the jump equations for the activation-angle down counter and the deactivation-angle down counter is that the activation-angle down counter only decrements during the period of  $q = 0$ . As soon as the counter becomes zero,  $q = 0$  turns into  $q^+ = 1 - q = 1$ , thus activating the actuator  $\eta$ . However, the deactivation-angle down counter only becomes zero when  $\xi_1 = \varphi_d + n_a(\pi/8)$ , causing  $q = 1$  to turn into  $q^+ = 0$ , thus deactivating the actuator  $\eta$ . Finally, the switching rule for the current actuator index  $\eta$  depends on the actuation status  $q$  right before the jump. In the first scenario where  $q = 1$  becomes  $q^+ = 0$ , the current actuator  $\eta^{th}$  has just deactivated, the jump rule determines that the next actuator index  $\eta$  equals the target actuator index  $n_a$  obtained from the input  $u$ :  $\eta^+ = n_a$ . In the second scenario when  $q = 0$  becomes  $q^+ = 1$ , the current actuator  $\eta^{th}$  starts to activate. Therefore, the jump rule determines that there should be no changes in the actuator index:  $\eta^+ = \eta$ . Combining the two scenarios, we have

$$\eta^+ = q n_a + (1 - q) \eta. \quad (51)$$

From equations (71)–(74), we have

$$\begin{bmatrix} \xi_1^+ \\ \xi_2^+ \\ q^+ \\ \Delta^+ \\ \delta^+ \\ \eta^+ \end{bmatrix} = \begin{bmatrix} \xi_1 \\ \xi_2 \\ 1 - q \\ q(n_a(\frac{\pi}{8}) + \varphi_a - \xi_1) \\ n_a(\frac{\pi}{8}) + \varphi_d - \xi_1 \\ q n_a + (1 - q) \eta \end{bmatrix}. \quad (52)$$

Finally, the flow and jump sets describe the flow and jump conditions. Since flow and jump cannot happen simultaneously, these sets are exclusive from one another. Specifically, jump set is satisfied as soon as  $(\Delta = 0 \wedge \delta \neq 0 \wedge q = 0) \vee$

$(\Delta = 0 \wedge \delta = 0 \wedge q = 1)$ . When the jump set is satisfied, the model follows the jump map  $x^+ = g(x, u)$ . When flow set is satisfied, the hybrid dynamic model follows the flow map  $\dot{x} = f(x)$ . These sets also describe the feasible values of states  $x$  during flow and jump, respectively. Therefore, we have

$$\begin{aligned} \mathcal{D} := & \{x \in \mathbb{R}^2 \times \{0, 1\} \times \mathbb{R}^2 \times \mathbb{N}_{\geq 0} | \\ & (\Delta = 0 \wedge \delta \neq 0 \wedge q = 0) \vee (\Delta = 0 \wedge \delta = 0 \wedge q = 1)\} \end{aligned} \quad (53)$$

$$\begin{aligned} \mathcal{C} := & \{x \in \mathbb{R}^2 \times \{0, 1\} \times \mathbb{R}^2 \times \mathbb{N}_{\geq 0} | \\ & \sim \pi (\Delta = 0 \wedge \delta \neq 0 \wedge q = 0) \\ & \wedge \sim (\Delta = 0 \wedge \delta = 0 \wedge q = 1)\}. \end{aligned} \quad (54)$$

### C. Operation of the Electrohydraulic Rolling Soft Wheel

Initially, the HASEL rolling wheel has activation status  $q = 0$ , activation-angle down counter  $\Delta = 0$ , deactivation-angle down counter  $\delta = 0$ , and current actuator index  $\eta = 0$  [see Fig. 8(a)]. Given the measurement of angular position  $\xi_1$ , the wheel can determine the current actuator index using (40). The wheel enters the jump map and update the values of  $q$ ,  $\Delta$ ,  $\delta$ , and  $\eta$  according to the input  $u$  [see Fig. 8(b)]. With an initial angular speed  $\xi_2$ , the wheel rolls, increments  $\xi_1$ , and decrements both  $\Delta$  and  $\delta$ . As soon as the activation-angle down counter reaches zero, the jump condition  $\Delta = 0 \wedge \delta \neq 0 \wedge q = 0$  is satisfied, which triggers the hybrid dynamic model to enter the jump map. The jump map determines that the activation status switches from 0 to 1, signaling the optocoupler first to close and activate the actuator first, as shown in Fig. 8(c). The deactivation-angle down counter remains the same after the jump and continue to decrease in the flow ( $\delta^+ = \delta$ ), and the current actuator index  $\eta$  remains 1. The generated force component  $F_{a,2}$  causes the wheel to accelerate and increase the angular speed  $\xi_2$ , which further decreases  $\delta$ . As soon as the deactivation-angle down counter reaches zero, the jump condition  $\Delta = 0 \wedge \delta = 0 \wedge q = 1$  is satisfied, which triggers the hybrid dynamics model to enter the jump map again. This time, the jump map determines that the activation status switches from 1 to 0, signaling the optocoupler first to open and deactivate the actuator first, as shown in Fig. 8(d). At this moment, the jump map also updates the new values of  $q$ ,  $\Delta$ ,  $\delta$ , and  $\eta$  according to the new input  $u$ , which determines the next target actuator (second in this example) to activate.

### D. Braking Rules

In the control problem of speed regulation, the wheel can lower its speed by passively decelerating thanks to the system's frictions; however, active decelerations via braking allows the wheel to react more quickly to the environment and reach target angular speed more effectively. The rolling soft wheel achieves braking by activating the actuator in the direction such that its force component  $F_{a,2}$  opposes the rolling direction (shown in Supplementary Video).

When the wheel is in motion, the current actuator  $i^{th}$  in contact with the ground is described in the hybrid dynamic model as  $\eta = i$ . The braking behavior is achieved by selecting the next actuator to be activated as  $n_a = i + 2$  instead of  $i + 1$ , which

TABLE I  
MECHANICAL CHARACTERISTICS OF THE ELECTROHYDRAULIC ROLLING SOFT WHEEL

Parameters	Values	Unit
Weight ( $M$ ) of robot	0.979	kg
Distance ( $L$ ) from the wheel's COM to pivot center	0.642	m
Radius ( $r$ ) of the wheel	0.157	m
POI ( $I_{13}$ ) of pivot arm	-0.00176	kg · m <sup>2</sup>
POI ( $I_{23}$ ) of pivot arm	$-4.97 \times 10^{-6}$	kg · m <sup>2</sup>
POI ( $I_{21}$ ) of pivot arm	0.0759	kg · m <sup>2</sup>
MOI ( $I_1$ ) of the wheel	0.00717	kg · m <sup>2</sup>
MOI ( $I_2$ ) of the wheel	0.00402	kg · m <sup>2</sup>
MOI ( $I_3$ ) of the wheel	0.00391	kg · m <sup>2</sup>

results in a negative contact angle since  $\xi_c = \xi_1 - (i + 2) \pi/8 < 0$ . As the wheel continues to roll,  $\xi_1$  increases, causing the contact angle to increase from a negative value to zero. The activation and deactivation angles corresponding to the actuator  $(i + 2)$ th must therefore be negative. Specifically, the activation angle and the deactivation angles are between  $-22.5^\circ$  and  $0^\circ$  with the activation angle closer to  $-22.5^\circ$  and the deactivation angle closer to  $0^\circ$ . The negative activation-angle down counter  $\Delta$  and deactivation-angle down counter  $\delta$ , when multiplied with  $-\xi_2$ , as seen in (49) and (50), become up counters from negative values to zero. With this setup, the jump map, flow map, jump set, and flow set for active deceleration are identical to those for acceleration. The braking behavior is therefore achieved by simply selecting the appropriate inputs  $n_a$ ,  $\varphi_a$ , and  $\varphi_d$ . In this article, we opt not to estimate the force function during braking, which we explain in the Section VI (“discussion”). However, we still show how braking is implemented as part of the controller in the Section V (“model predictive controller design”).

#### IV. CHARACTERIZATION AND PARAMETER ESTIMATION

To quantify the location and orientation of the electrohydraulic rolling soft wheel on the platform, we use a motion capture system (OptiTrack, NaturalPoint Inc.) that consists of eight infrared cameras (Prime<sup>x</sup> 22, NaturalPoint Inc.) with a 3-D accuracy of  $\pm 0.15$  mm and a sampling rate of 360 Hz. The data are saved to CSV files and then postprocessed using MATLAB (MATLAB R2020a, MathWorks). Furthermore, the continuous dynamic equation  $\dot{x} = f(x, u)$  is discretized using Runge–Kutta fourth-order method with a step size  $h = 1/300$  (s).

##### A. Estimation of Friction Torques

Most mechanical parameters of the wheel are given in Table I, except for the friction torque’s of the two set of bearings on the pivot and the wheel that are unavailable from the supplier. For simplicity, the friction torques are treated as constants. Their values can be estimated by conducting an optimization procedure.

First, 30 set of angular position and angular velocity of the wheel, with arbitrary initial angular velocity and angular position, are captured using the motion capture system. We follow the MATLAB procedure “Fit ODE, Problem-Based” [48]

TABLE II  
INFLUENCES OF THE ACTIVATION ANGLE AND DEACTIVATION ANGLE ON THE RISE TIME AND STEADY-STATE LINEAR SPEED

$\delta_d$ (°)	$\delta_a$ (°)	1	3	5	7	9	11
		$t_r$ (s)	$\xi_{2,ss}$ (m/s)	$t_r$ (s)	$\xi_{2,ss}$ (m/s)	$t_r$ (s)	$\xi_{2,ss}$ (m/s)
22	$t_r$ (s)	29.17	26.23	27.04	23.83	12.57	
	$\xi_{2,ss}$ (m/s)	0.57	0.43	0.39	0.32	0.21	
20	$t_r$ (s)	31.86	25.20	31.37	16.63	19.45	
	$\xi_{2,ss}$ (m/s)	0.48	0.44	0.5	0.31	0.24	
18	$t_r$ (s)	32.76	25.98	27.17	18.30		
	$\xi_{2,ss}$ (m/s)	0.63	0.45	0.38	0.36		
16	$t_r$ (s)	36.22	30.96	34.34			
	$\xi_{2,ss}$ (m/s)	0.47	0.38	0.31			
14	$t_r$ (s)	38.37	37.34				
	$\xi_{2,ss}$ (m/s)	0.42	0.42				
12	$t_r$ (s)	32.19					
	$\xi_{2,ss}$ (m/s)	0.53					
11							

for the first twenty training datasets to estimate the friction torque by minimizing the sum of squared of the error between the estimated states and the measured states (angular positions and velocities). The result is  $\tau_f = 0.0104$  (kg m<sup>2</sup>/s<sup>2</sup>). We then simulate the wheel’s dynamics using this  $\tau_f$  value and compare the R-squared results with the remaining ten test sets. The goodness of fit values for the angular position of the wheel range between 0.962 and 0.999 with an average of 0.987. The goodness of fit values for the angular velocity of the wheel range between 0.793 and 0.9890 with an average of 0.938.

##### B. Estimation of Actuation Force Function

We observe the effect of the actuation forces indirectly based on how the wheel accelerates. As previously mentioned, the actuation forces depend on the wheel’s dynamic states, including the contact angle  $\xi_c$ , the current angular speed  $\xi_2$ , and the actuator’s status  $q$ , and the control input  $u$ , which includes the activation angle  $\varphi_a$ , and the deactivation angle  $\varphi_d$ . Since we assume that all actuators are identical, the actuation forces should not depend on which actuator ( $n_a$ ) being activated.

To estimate the function of the actuation force, we have the first six actuators to activate at the activation angle  $\varphi_a = 3^\circ$  to ensure the wheel has sufficient momentum for continuous rolling. In the subsequent actuations, we look at the wheel’s angular speed with respect to a combination of activation angle  $\varphi_a$ , ranging from  $1^\circ$  to  $11^\circ$ , and the deactivation angle  $\varphi_d$ , ranging from  $11^\circ$  to  $22^\circ$ . All actuators past the six initial actuators are activated sequentially, thus removing the influence of input  $n_a$  from the experiment, and all actuators have the same predetermined activation and deactivation angles throughout the 60-s-long trials. The activation and deactivation angles are the same for all actuators in each trial and are only varied across different trials. The influences of the activation angles and deactivation angles on the linear speed’s rise times  $t_r$  (s) and steady-state values  $\xi_{2,ss}$  (m/s) are documented in Table II. Any combination of activation angle and deactivation angles shown

in the gray region of Table II result in the wheel's discrete rolling behavior at steady state.

Based on this table, for the same activation angle, varying the deactivation angle (specifically in the range from 11° to 22°) has little influences on the wheel's steady-state linear speed, as shown in the unclear patterns of the steady-state linear speed with respect to increasing the deactivation angle. This behavior agrees with the general form of the force function as the force diminishes rapidly as the distance between the actuator and ground increases.

The fluctuation of the linear speed for the same activation can be explained by the actuator's charge retention, which occurs after applying high electric fields to the actuators for a long period of time and causes the actuators' performance to worsen. While this effect is not well understood, it may be related to space charge creation in the thin-film dielectrics, caused by ionization of low-molecular-weight impurities under application of high electric fields [49], [50]. Since the order of experiences for the Table II is randomized, the experiments that are conducted earlier result in greater steady-state linear speed. Charge retention also result unclear pattern for the speed's rise time, especially when the experiment trials are randomized.

For the same deactivation angle, increasing the activation angle results in smaller actuation force, which in turn results in smaller steady-state linear speed. This is consistent with the profile of the force function. However, there are no clear trends for the rise time of the angular speed of the wheel with respect to either the activation angle or the de-activation angle. Since the influence of the deactivation angles is not obvious, we can simplify the process of estimating the force function's parameters by fixing the deactivation angle to its maximum value of 22° and only varying the activation angles. With the input  $n_a$  simply incrementing based on  $n_a = 16 \bmod (\eta + 1)$  and the input  $\varphi_d = 22^\circ$ , the three-input-six-state hybrid dynamics model becomes one-input-six-state model.

Due to the nonlinear problem of the force function's parameter estimation that contains both continuous and discrete components, we choose particle swarm optimization (PSO) algorithm [51]–[55]. The algorithm solves an optimization problem by maintaining a population of candidate solutions, dubbed particles, and moving these particles around in the search-space according to simple mathematical formula over the particle's position and velocity. Since PSO makes no assumptions about the problem that is being optimized, can search very large spaces of candidate solutions, and does not use the gradient of the problem being optimized, the algorithm is ideal for the estimation of actuation force's parameters, which is a nonlinear, noncontinuous optimization problem with speed-dependent continuous and discrete dynamics.

In our problem, the candidate solutions are the set of force function's parameters  $\{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6\}$ , as shown in (43). The force function is generated based on these parameters and applied to the hybrid dynamics model to simulate the wheel's angular position and angular speed. The objective function is to minimize the sum of squared errors between the simulated linear speed and the measured linear speed for all trials with different activation angles from 1° to 9° with an increment of

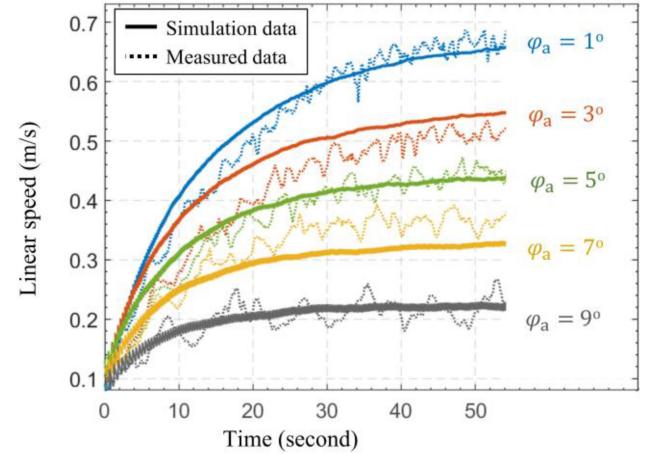


Fig. 9. Evaluation of the estimated force function  $F_a$  with six numerical parameters  $\{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6\}$ . Visual comparison between the simulated linear speed of the rolling system using hybrid dynamic model with the estimated force function and the measured linear speed using motion capture system. The two datasets are generated based on five trials with five values of  $\varphi_a$  ranging from 1° to 9° with an increment of 2°, a prescribed  $n_a = 16 \bmod (\eta + 1)$ , and a constant  $\varphi_d = 22^\circ$ .

2° and constant deactivation angle of 22°. Specifically, we call  $x_p^w = \{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6\}_p^w$  as a set of parameters in the search space  $\mathbb{R}^6$ .  $p = 1, \dots, P$  with  $P$  is the total number of candidate solutions, or particles, in the search space.  $w = 1, \dots, W$  represents the increment of time step with  $W$  is the final time at which either an optimal solution is reached (its corresponding sum of squared errors is lower than a predetermined threshold) or the number of search iterations of the PSO algorithm is reached.

The new position  $x_p^{w+1}$  and velocity  $v_p^{w+1}$  of each particle after each search iteration are updated by the following rules:

$$x_p^{w+1} = x_p^w + v_p^{w+1} \quad (55)$$

$$v_p^{w+1} = c_1 v_p^w + c_2 r_1 (p_p^w - x_p^w) + c_3 r_2 (p_g^w - x_p^w) \quad (56)$$

where  $c_1$  is the inertia term,  $r_1$  and  $r_2$  are random values between 0 and 1, and  $c_2$  and  $c_3$  represent the cognitive and social scaling parameters. Here, we select  $c_1$  to be 0.5,  $c_2$  and  $c_3$  to be 1 and 2, respectively. To limit computation time, we select the, possibly sub-optimal, solution after ten search iterations.

### C. Actuation Force Model Evaluation

The result of the PSO algorithm described in the previous section is as follow:  $\{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6\} = \{1.214, 1.997, 3.836, -1.004, -0.1009, 22.357\}$ , for five trials (see Fig. 9), each with approximately 20000 data points sampled at 360 Hz. Fig. 9 shows how well the linear angular speed generated by the hybrid dynamic model with the chosen force parameters fit with the linear angular speed obtained from the measured data in five experiments with activation angles  $\varphi_a$  ranging from 1° to 9° and constant deactivation angle  $\varphi_d = 22^\circ$ . Table III gives the fitting performance between the simulated linear speed using the hybrid dynamic model with the estimated force function and the measured linear speed of the wheel using motion capture system.

TABLE III  
FITTING PERFORMANCE BETWEEN THE SIMULATED LINEAR SPEED AND THE MEASURED LINEAR SPEED OF THE WHEEL

$\varphi_a$ (°)	RMSE (m/s)	R-squared	$e_{ss}$ (m/s)
1	0.17	0.97	0.12
3	0.32	0.82	0.18
5	0.16	0.93	0.09
7	0.24	0.73	0.27
9	0.13	0.72	0.10

The performance results are based on three metrics: root-mean-square errors (mm), goodness of fit (R-squared), and steady-state errors (m/s) between the simulated and experimental linear speed.

From Fig. 9 and Table III, the simulated data matches well with the measured data for experiments with activation angles  $\varphi_a = 1^\circ$ ,  $\varphi_a = 5^\circ$  (R-squared values of 0.97 and 0.93, respectively). For the experiments with activation angles  $\varphi_a = 3^\circ$  and  $\varphi_a = 7^\circ$ , there are deviations in both transient and steady-state velocities, resulting in poor R-squared values (0.82 and 0.73, respectively). The steady-state errors of 0.18 (m/s) and 0.27 (m/s) also reflect the poor fit in these two experiments. Based on Fig. 9, the fitting result in the experiment with activation angle  $\varphi_a = 9^\circ$  is more acceptable than what R-squared values of 0.72 and steady-state error of 0.10 (m/s) indicate.

There are several reasons for the imperfect R-squared values and steady-state errors. First, as the wheel rolls more slowly due to larger activation angles, the wheel's linear speed becomes more discrete, resulting more fluctuation in the speed profile. Second, the order of trials is random: the trial with activation angle  $\varphi_a = 7^\circ$  is conducted before the trials with activation angles  $\varphi_a = 1^\circ$ ,  $\varphi_a = 5^\circ$ , and  $\varphi_a = 9^\circ$ , which are conducted before the trial with activation angle  $\varphi_a = 3^\circ$ . The later experiments result in more noticeable charge retention on the actuators that cause the actuation forces to be weaker than that of the same experiment, but if conducted earlier. Finally, the optimization solution is suboptimal, and the form of the force function may be more complex, especially in the sophisticated hybrid dynamic interaction between the actuators and the ground while the wheel continuously changes its position and speed.

From this experiment, we can also see that the maximum and minimum controllable range of linear speed is 0.2 to 0.7 m/s. Below a speed of 0.2 m/s, the wheel's motion becomes discrete, and the speed has a saw-tooth profile (not shown). Exceeding speed above 0.7 m/s will require a lighter wheel or stronger actuators.

## V. MODEL PREDICTIVE CONTROLLER DESIGN

### A. Controller Description

The electrohydraulic rolling soft wheel has state jumps that are state-dependent and discrete input  $u$  that remains constant for the entire duration of activation-deactivation of a target actuator  $n_a$ . These two characteristics of the wheel make it challenging to use conventional controllers. In this section we describe the

model predictive controller (MPC) and the PSO algorithm to implement optimal inputs  $u$  to regulate the wheel's linear speed.

The controlled system [see Fig. 10(a)] begins with a reference speed that the wheel attempts to track. The controller, based on the hybrid dynamic model of the plant and the PSO optimizer, generates an input  $u$  to minimize the errors between the predicted speed and the reference speed. The wheel then changes its angular position and speed due to the actuation force of the target actuator. Both the motion capture system and the absolute rotary encoder quantify the angular position of the wheel. Based on the measurement of the first physical state (angular position  $\xi_1$ ), the state observer is simply based on (45) to compute its derivative and obtain the second physical state (angular speed  $\dot{\xi}_1$ ). From the two known physical states, the state observer simply computes the remaining four non-physical states based on the flow map (45), jump map (47), jump set (53), and flow set (54). The outputs of the state observer consist of full six states that are used for the next MPC iteration and the estimated speed that is used in the disturbance estimator.

The disturbance estimator [56] is implemented to account for the real-time error in the dynamic modeling, unmodeled external disturbances, and fabrication and computation errors that result in deviations between the measured speed and the simulated speed generated by the hybrid dynamic equations. Since we demonstrate only the control of the steady-state angular speed, the disturbance estimator is not used during the transient phase of wheel's speed. When the predicted speed of the wheel is less than approximately 0.1 m/s (0.64 rad/s) of the reference speed, the estimator is switched ON, allowing the controller to gradually correct for the speed's offset. As mentioned in [46] and [47], a pair  $(t, j)$  represent the flow of events in the hybrid dynamic system, with times  $t$  accounts for the flow of the continuous segment of the system and jump steps  $j$  represents the events that the system switches from the flow map to the jump map. Since the controller is digital, the time  $t$  is discretized and represented as variable  $k$ . The detailed explanation of the discretization is shown in Fig. 10(b). The disturbance estimator has the follow discrete form:

$$\hat{d}(k+1, j) = \hat{d}(k, j) + K e_{ss}(k, j) \quad (57)$$

$$e_{ss}(k, j) = \xi_{2,\text{meas}}(k, j) - \xi_{2,\text{prdt}}(k, j) \quad (58)$$

where

- $\hat{d}(k+1, j)$ : Updated estimated disturbance term.
- $\hat{d}(k, j)$ : Current estimated disturbance term.
- $e_{ss}(k, j)$ : Steady-state error between the measured and predicted speeds.
- $\xi_{2,\text{meas}}(k, j)$ : Measured speed.
- $\xi_{2,\text{prdt}}(k, j)$ : Current predicted speed based on the MPC algorithm.
- $K$ : Positive valued tuning constant.

A larger  $K$  results in faster error build-up and more responsive correction for the control offset. However, too large  $K$  also amplifies noises in the measured speed and cause the controller to generate an oscillatory speed around the reference speed.

The controller has two operating modes: acceleration and active deceleration. In the acceleration mode, the controller

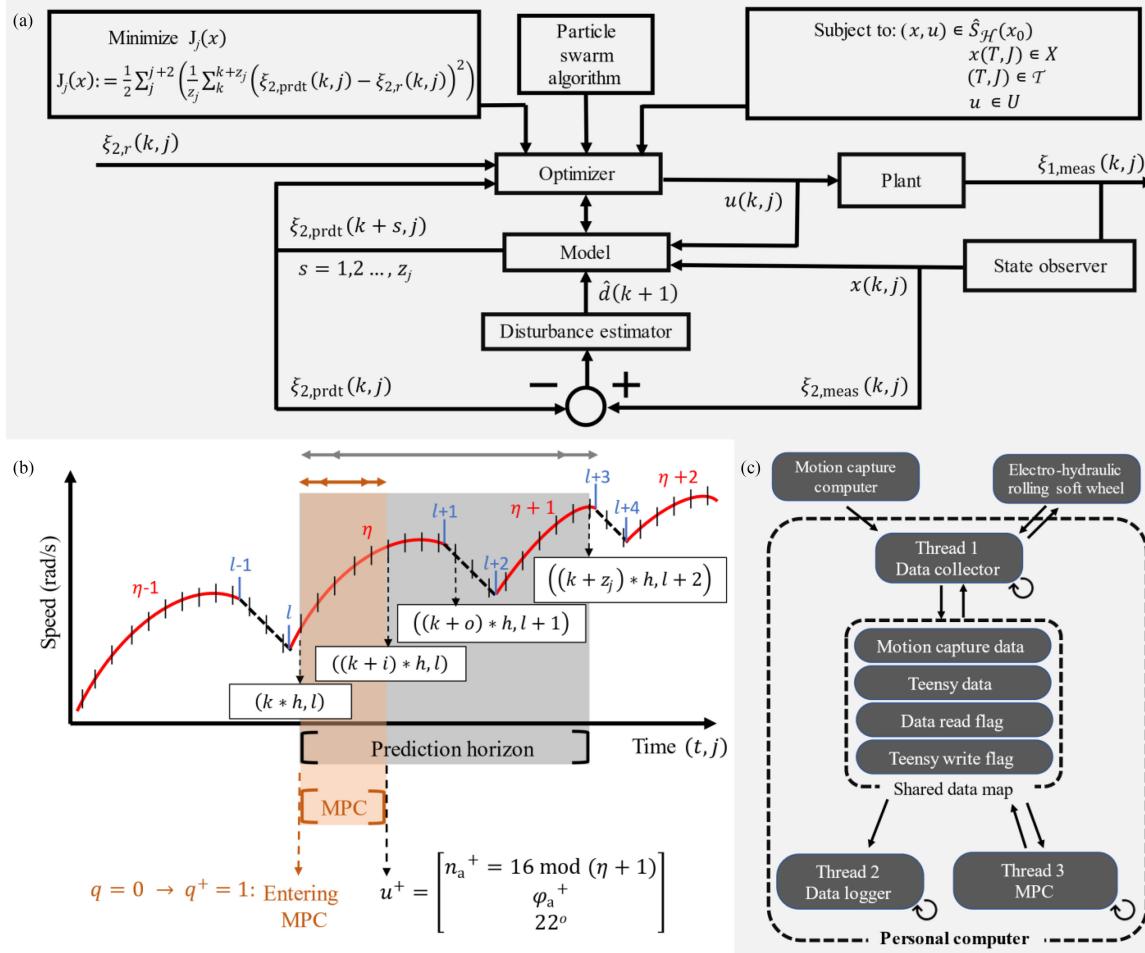


Fig. 10. Description of the MPC for the electrohydraulic rolling soft wheel. (a) The MPC block diagram with a particle swarm optimizer that minimizes the cost function  $J_j(x)$  and is subjected to state and input constraints. The MPC also employs a disturbance estimator to correct for the discrepancies between the predicted and measure speeds due to modeling simplification and unmodeled disturbances. (b) Illustrative snapshot of a typical MPC iteration for wheel's speed (m/s) with respect to times  $t$ , jump steps  $j$ . The red line on the plot represents the wheel's status  $q = 1$ , whereas the black line represents the wheel's status  $q = 0$ . The snapshot illustrates the speed-dependent behavior of the continuous and discrete dynamics. At time  $(t, j) = (k * h, l)$ , the wheel applies HV to the actuator  $\eta^{\text{th}}$ , and the MPC computes the optimal  $u$  for the target actuator  $n_a$ . The prediction horizon spans  $(t, j) = (k * h, l)$  to  $(t, j) = ((k + z_j) * h, l + 2)$ . The MPC computation must complete before the wheel deactivates the actuator  $\eta^{\text{th}}$ ; that is  $(t, j) = ((k + i) * h, l) < ((k + o) * h, l)$ , with  $i$  is the time instant at which the MPC completes its computation and  $o$  is the time instant at which the wheel deactivates its actuator  $\eta^{\text{th}}$ . The orange and gray arrows indicate the adaptive, nonperiodic nature of the controller as the wheel's speed changes. (c) Real-time, C++-based, and multi-threaded implementation of the MPC controller for the speed regulation of the wheel. The multithreaded program allows for integration of nonperiodic thread (the MPC) and the periodic threads (the data collector and the data logger). The arrows indicate the direction of data transmission.

uses MPC algorithm with PSO algorithm to determine the optimal input to activate the next actuator such that the difference between predicted speed  $\xi_{2,\text{prdt}}(k, j)$  and the reference speed  $\xi_{2,r}(k, j)$  is minimized. If  $\xi_{2,\text{prdt}}(k, j) < \xi_{2,r}(k, j) + 0.3$  (m/s), the controller will maintain the acceleration mode. If  $\xi_{2,\text{prdt}}(k, j) \geq \xi_{2,r}(k, j) + 0.3$  (m/s), the controller will switch to the active deceleration mode. In this mode, the controller applies braking inputs with a target actuator  $n_a = 16 \bmod (i + 2)$  and predetermined activation angle of  $-22^\circ$  and deactivation angle of  $-6^\circ$ .

The MPC involves a cost function  $J_j(x)$  that can be defined as

$$J_j(x) := \frac{1}{2} \sum_{j=1}^{j+2} \left( \frac{1}{z_j} \sum_{k=1}^{k+z_j} (\xi_{2,\text{prdt}}(k, j) - \xi_{2,r}(k, j))^2 \right). \quad (59)$$

Since we do not know the reference speed in the future, we assume that it is a constant throughout each MPC iteration:  $\xi_{2,r}(k, j) = \xi_{2,r}(k+1, j) = \dots = \xi_{2,r}(k+z_j, j)$ . The cost function  $J_j(x)$  is the mean squared error between the predicted speed and the reference speed during the entire activation-deactivation cycle (duration of  $q = 0$  and  $q = 1$ ) which corresponds to the jump step incrementing twice from right after  $j = l$  to right before  $j = l + 3$  [see Fig. 10(b)], with  $l$  is an integer that represent the current jump step of the hybrid dynamic model.  $z_j$  is the total number of time steps spanning these jumps with step size  $h = 1/300$  (s). The PSO optimization goal is: Given  $x_0 \in \mathbb{R}^n$

$$\begin{aligned} &\text{Minimize } J_j(x) \\ &\text{Subject to: } x \in \hat{S}_{\mathcal{H}}(x_0) \\ &x(T, J) \in X \end{aligned}$$

$$(T, J) \in \mathcal{T}$$

$$u \in U$$

where  $\hat{S}_H(x_0)$  is the set of solution pairs  $(x, u)$  of  $H$  satisfying  $x_0 = x(0, 0)$ , and  $(T, J)$  denotes the terminal time of  $x$ .

We allow the terminal times of the feasible solution pairs to belong to a set  $\mathcal{T} \subset \mathbb{R}_{\geq 0} \times \mathbb{N}_{\geq 0}$ , called the prediction horizon

$$\mathcal{T} = \left\{ (t, j) \in \mathbb{R}_{\geq 0} \times \mathbb{N}_{\geq 0} : \max \left\{ \frac{T}{f_s}, j \right\} = \tau \right\} \quad (60)$$

for some integer  $\tau \in \{1, 2, \dots\}$  and  $f_s > 0$ , where  $f_s$  is the sampling rate of the controller, so that the terminal time  $(T, J)$  of any feasible solution pair satisfies  $\max \{T/f_s, j\} = \tau$ . In our problem, the prediction horizon for a given control input is the entire duration of activation-deactivation cycle that is previously mentioned.

The optimal input  $u$  is computed at the beginning of every period  $q = 0$  for the target actuator  $n_a$  which is always the adjacent actuator of the current actuator  $\eta^{\text{th}}$  ( $n_a = 16 \bmod (\eta + 1)$ ). Each input  $u$  is fixed for the entire activation-deactivation cycle of the target actuator. Since the input is discrete and remain constant for the entire activation-deactivation cycle, the concept of a control horizon does not apply.

Fig. 10(b) shows a snapshot of a typical MPC iteration for acceleration with respect to times  $t$ , jump steps  $j$ , and the wheel's speed. From the figure, the current actuator to be activated is  $\eta$ . The vertical, equally spaced marks represent time  $t$  that is discretized into steps  $k$  by Runge–Kutta fourth order method with step size  $h$ . Whenever there is switch of status  $q$ , the jump step  $j$  increments. MPC then computes an optimal control input  $u^+$  which is used by the embedded microcontroller to update the next control command. The MPC prediction horizon takes two jump steps from  $J_k = l$  to  $J_{k+z_j} = l + 2$ , which corresponds to the instant the target actuator  $n_a$  starts to deactivate. Because the change in jump steps based on the wheel's activation and deactivation angle down counters, which in turn decrement at the same rate as the wheel's speed  $\xi_2$ , the total number of time steps required spanning  $J_k = l$  to  $J_{k+z_j} = l + 2$  decreases as the wheel's speed increases. Therefore, the prediction horizon is adaptive (decreasing), causing the MPC computation time to be also adaptive (decreasing). Here, we define  $i$  as the time instance at which the MPC completes its computation and  $o$  as the time instance at which the wheel deactivates its actuator  $\eta^{\text{th}}$ . The instant  $(t, j) = (T_{k+i}, J_{k+i}) = ((k + i) * h, l)$  at which the MPC completes its computation must happen before  $J_{k+o-1} = l$  turns into  $J_{k+o} = l + 1$ , since the actuator  $n_a$  requires the updated activation angle down counter to begin decrementing at the instant  $J_{k+o-1} = l$  turning into  $J_{k+o} = l + 1$  and reach zero at the instant  $J_{k+z_j-1} = l + 1$  turning into  $J_{k+z_j} = l + 2$ . Based on this description, we have  $0 < i < o < z_j$ .

Optimization-based controllers have difficulties with both nonlinear, nonconvex, and discrete cost functions and their real-time computation capability. To solve for the optimal input  $u(k, j)$  that minimizes the cost function  $J_j(x)$  in real-time, we employ the PSO algorithm. The algorithm has the same structure as that mentioned in the previous section. Specifically, we call

$x_p^w = \varphi_{a_p}^w$  as a set of activation angles in the search space  $\mathbb{R}$ .  $p = 1, \dots, P$  with  $P$  is the total number of candidate solutions, or particles, in the search space.  $w = 1, \dots, W$  represents the increment of time step.  $W$  is the final time at which either an optimal solution is reached (its corresponding sum of squared errors is lower than a predetermined threshold) or the number of search iterations of the PSO algorithm is reached. Due to the computational cost and the constraint of MPC duration, we choose  $P = 15$  and  $W = 1$ . This means that there is only one iteration of PSO with 15 possible activation angles  $\varphi_a$  ranging from  $1^\circ$  to  $15^\circ$  to choose from. Beyond  $15^\circ$ , the actuation force is too weak to influence the wheel's speed. The goal of the PSO algorithm is to pick one out of 15 values of  $\varphi_a$  that can minimize the errors between the predicted speed generated by the target actuator  $n_a$  and the reference speed. The deactivation angle  $\varphi_d$  is fixed at  $22^\circ$ . Finally, since there is only one iteration of PSO, (55) and (56) do not apply. The next section describes how the controller is implemented in real-time.

### B. Real-Time Implementation

The controller's varying computation time requires concurrent computing to separate the nonperiodic MPC thread from the periodic data collector thread that collects the motion capture data, the embedded microcontroller data, and data logging [see Fig. 10(c)]. Besides, since each MPC iteration that involves intensive real-time computation must complete before  $j$  increments to  $j + 1$  (mentioned in the previous section), we require powerful computation hardware to meet this real-time demand.

A computer with a Ryzen 5 5600X (Advanced Micro Devices) processor and 32 GB of 3600 MHz DDR4 RAM is used for the C++ multi-threaded implementation. The program is designed around a shared data map that all threads have access to concurrently. This data map contains the latest update of the motion capture data and embedded microcontroller (Teensy 3.6, PJRC) data. The first thread, the data collector, continuously samples the motion capture and microcontroller data and write them to the shared data map at 600 Hz. The data logger thread obtains a copy of the shared data map and write it to a text file at the rate of 300 Hz for post processing. The MPC thread polls the content of the shared data map; when the wheel is activated ( $q^+ = 1$ ), MPC computes an optimal control command input  $u$  and send it to the embedded microcontroller via the data collector thread. The embedded microcontroller sends angular position from the rotary encoder, the actuation status  $q$ , and the current actuator index  $\eta$  over serial and check for new control command input  $u$  at 300 Hz. If there are updated input, the controller waits until the current actuator  $\eta$  finishes activating then applies the command to the target actuator accordingly.

### C. Controller's Preliminary Result

Fig. 11 shows real-time steady-state control results for the electrohydraulic rolling soft wheels. Fig. 11(a) and (b) demonstrates the improvement of the steady-state results of the wheel's angular speed that tracks four different referenced signals (1, 1.5, 2, and 2.5 rad/s) due to the inclusion of the disturbance estimator. For the referenced signals of 1, 1.5, and 2 rad/s, there

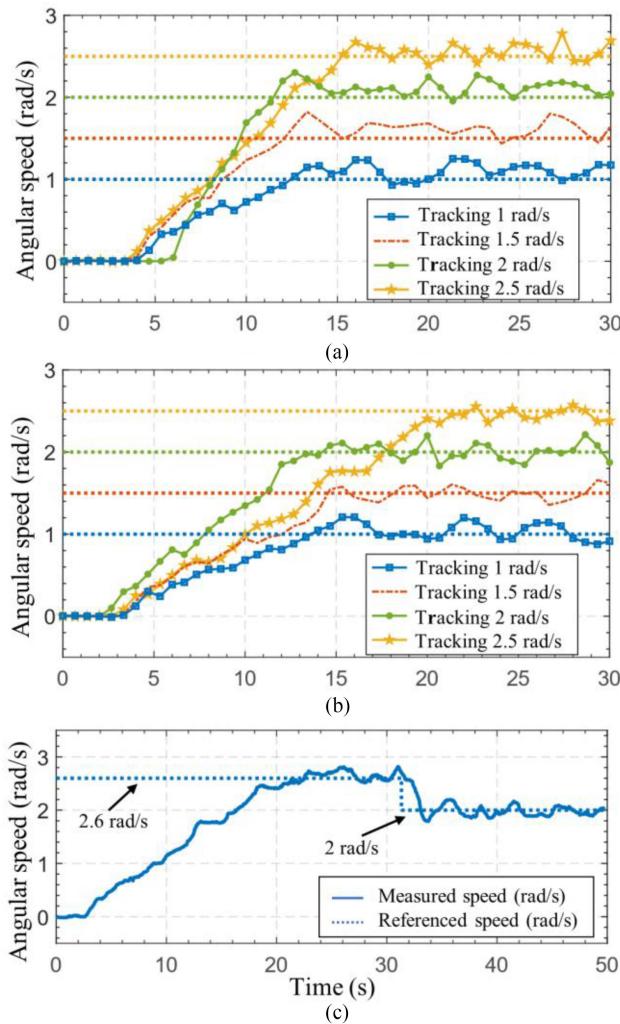


Fig. 11. Real-time steady-state control result for the electrohydraulic rolling soft system. (a) Control result without the disturbance estimator. There are significant steady-state errors between the measured referenced angular speeds. (b) Control result with the disturbance estimator. The steady-state errors between the measured and referenced speeds are minimized. (c) Control results with the referenced speed stepping down from 2.6 to 2.0 rad/s.

is an overall decrease in percent of steady-state error between the measured angular speeds and referenced speed: 9.80% to 0.80%, 7.82% to 0.96%, and 5.37% to 0.52% respectively. There is a slight increase in steady-state error in the case of 2.5 rad/s due to the charge retention on the actuators. Similarly, the disturbance estimator also improves the normalized root-mean-square error (NRMSE) for the first three cases of reference speeds: 0.133 to 0.090; 0.098 to 0.053; and 0.067 to 0.046, respectively. However, the increased steady-state error for the case of 2.5 rad/s also results in a slight increase in NRMSE (from 0.039 to 0.042). These results indicate that the disturbance estimator is important to improve the MPC's performance.

It is interesting to note that, in both figures, the rise times of the wheel's speed to track 2.5 rad/s are noticeably slower than those to track 2 rad/s, and, for the same referenced speed, the rise times in Fig. 11(b) are slower than those in Fig. 11(a). This behavior can be explained by the charge retention on the actuator. The charge retention not only affects the actuator's

maximum stroke, but also how quickly the actuator expands, thus influencing the wheel's acceleration.

Fig. 11(c) shows how the MPC controller with disturbance estimator regulates the wheel's angular speed (rad/s) that tracks a referenced speed of 2.6 rad/s for 31 s, then actively decelerate to reach a new referenced speed of 2 rad/s and maintain at this speed for 19 s. The braking fall-time from 2.6 to 2 rad/s is approximately 1.2 s. While there are some steady-state speed fluctuations due to the fabrication and controller design process, these results demonstrate the feasibility that the wheel can adjust its speed by either accelerating or decelerating in real-time.

## VI. DISCUSSION

Rolling soft systems have the following three design tradeoffs: level of integration (hardware and software); performance (such as speed and acceleration); and system's weight and size. Table IV illustrates how the HASEL-driven rolling soft wheel fares against existing rolling systems driven by soft actuators. While it is often true that light weight is an advantage for performance and efficiency, weight and performance alone are insufficient to describe all aspects of the system. Specifically, several existing soft systems ([27], [33]) achieved high performance such as rolling speed (1.25 and 1.43 blps, respectively) and small weight (11.27 and 12.2 g, respectively), at the cost of any level of hardware integration and useful functionalities such as deceleration and speed control. As given in Table IV, only the wheeled robot [27], the soft wheel robot [35], and the spherical robot [37] provide certain levels of self-containment, which result in significant increase in systems' weight (1469 g for [35] and 9100 g for [37]) and/or significant reductions in average or maximum speed (0.9 blps for [27], 0.72 blps for [35], and 0.21 blps for [37]). With the consideration of all design tradeoffs, HASEL shell-bulging soft wheel outperforms existing soft rolling systems; despite its weight of 979 g (to accommodate HV and embedded hardware), the design achieves an excellent maximum speed of 2.2 blps (0.7 m/s), with additional functionalities including deceleration and closed-loop control, which is not demonstrated in virtually any existing rolling soft systems.

With the given rolling performance, the design, modeling, and control limitations mentioned in the previous sections can be addressed in two ways: improving sensing and control and/or redesign of the HASEL actuator. In the first category, the local state of each actuator can be monitored by integrating embedded sensors such as a capacitive self-sensing circuit [57]. Regarding controls, the issue of fabrication inconsistency of 16 actuators can be addressed systematically by a local PID controller [58] for each actuator, which can make the output strokes and rise times consistent across actuators, thus helping the MPC algorithm better compute optimal control commands. Another method to remove the charge retention on the actuators is by means of HV and ground reversal [42], which has been demonstrated in a compact, embedded form [43]. Alternative reversing the polarity of the HV and ground connections to the actuators can forcefully remove all charges accumulated on the actuators, effectively restoring the actuators

TABLE IV  
PERFORMANCE COMPARISON OF ROLLING SYSTEMS DRIVEN BY SOFT ACTUATORS

Rolling systems	Mechanism	Actuation method	Body length (mm)	Weight (g)	Average speed (blps)	Maximum speed (blps)	Hardware integration	Controller
This work	Shell-bulging	HASEL	314	979	Controllable	2.2	Partially	Closed-loop
Distributed robot [39]	Shell-bulging	Pneumatic	—	—	—	—	no	Closed-loop
JSEL [40]	Shell-bulging	Granular jamming	—	—	—	—	no	No
rDEA [37]	Shell-bulging	DEA	90	23	—	~0.95	no	No
Spherical robot [38]	Shell-bulging	Pneumatic	356	9100	—	~0.21	Partially	No
Soft wheel robot [36]	Shell-bulging	Pneumatic	139.7	1469	0.72	—	Fully	No
IRSR [34]	Shell-bulging	DEA	106.0	12.2	1.19	>1.43	No	No
Wheeled robot [27]	Shift of COM	Liquid metal	45.0	11.27	1.1 0.9	1.25	No	No
Spherical rover [28]	Shift of COM	DEA	108	75	—	—	No	No
STAR [31]	GD	SMA	50.0	16.4	>1.00	—	No	No
GRSR [34]	GD	DEA	49.7	0.88	0.77	1.58	No	No
Circular soft robot [11]	GD	SMA	40.0	3.6	0.65	—	No	No
Light motor [32]	GD	Remote light	20	—	~3.00	—	No	No

DEA, dielectric elastomer actuator; EA, electroadhesion actuator; HASEL, hydraulically-amplified self-healing electrostatic; GD, geometric deformation; GRSR, gravity-based rolling soft robot; IRSR, impulse-based rolling soft robot; SMA, shape memory alloy; COM, center of mass.

to the charge-free state. Above all, the mathematical description of the rolling wheel can be significantly improved by considering the physics behind the ground contact force for both acceleration and braking. Due to the sophisticated nature of the wheel's rolling motion, a detailed, physics-based study of the contact force generated by the actuator as the wheel rolls is required to improve the dynamic model of the rolling behavior, which in turn influences how well the controller can regulate the wheel's speed during both transient and steady states.

In terms of actuator redesign, different geometry of folded-HASEL actuators can play an important role in the performance of the rolling soft wheel. The wheel can achieve smoother rolling motion with narrower but longer folded actuators with increased number of pouches. For the same rims' circumference, decreasing the actuator's dimension increases the number of actuators that can be fitted on the same frame. Besides, the decrease in the contact dimension of the actuators and the ground also allows for greater peak forces, which improve the wheel's acceleration. Furthermore, a separate study about the actuator's charge retention effect is critical to design better actuators without this time-dependent behaviors and therefore improve the consistency of the wheel's performance. Importantly, improved materials systems will enable HASEL actuators with drastically improved performance [59].

There are several reasons that influence our design choices regarding the integration of sensors, power supply, and control hardware onto the HASEL-powered rolling wheel. The design and characterization of the wheel in this article is geared toward utilization in a four-wheel platform. Additional wheels can collectively share the added loads from a battery, high-resolution rotary encoders, an inertial measurement unit, and more efficient compact computation hardware (such as FPGA or DSP). Furthermore, a four-wheel platform can enable additional steering function that is not possible with the existing HASEL-driven rolling system. As the wheels are connected to one another via axles, the mathematical model that describes the wheel-pivot arm kinematics and dynamics can be conveniently repurposed to achieve global control on a four-wheel platform. Even though we

have yet to demonstrate the impact absorption of the soft wheel and there are limited capabilities of the controller (such as transient control), future design of a four-wheel platform can provide several inherent advantages over traditional four-wheel vehicles, including reduction of moving parts (braking mechanism and transmission for forward and reverse motions) improvement of serviceability (thanks to the ease of access to the actuators that drive the wheels), and potentially impact absorption. A separate study of impact absorption, terrain traversal, and payload limits of the four-wheel platform, accompanied by redesigns of actuators for surface ruggedness and maximum force and stroke) is a logical and meaningful next step to demonstrate that soft actuators can be a competitive alternative to traditional motors and servos.

## VII. CONCLUSION

In this article, we introduce a novel shell-bulging rolling soft wheel driven by folded-HASEL actuators. The wheel's performance was strongly tied to the interplay between electrical, mechanical, material, and software design choices. With the balance of several design factors, the wheel outperforms existing rolling soft systems in several performance metrics, despite the added weight of the embedded components. Furthermore, this article was the first to describe and analyze the hybrid dynamic model of shell-bulging rolling mechanism that contains both continuous rolling motion and discrete activations and deactivations of the actuators. This model allows us to design an MPC to regulate the wheel's rolling speeds, which enables the wheel to accelerate and leverage its inherent braking capability to reach desired speeds—a critical function that did not exist in previous rolling soft systems. While there remain years for rolling soft wheels to catch up with the matured motor- and servo-based systems, the HASEL-driven rolling wheel presents a right direction toward high performing rolling soft systems that can bring many alternative benefits to traditional, rigid ones. Moreover, the technical discussion in this article may provide guidance and inspiration for creating new, robust, and highly mobile soft robots.

## ACKNOWLEDGMENT

The authors would like to thank V. Sundaram and B. K. Johnson for their discussions regarding the design and characterization of the electrohydraulic rolling soft wheel. The authors also acknowledge funding from the Army Research Office (Grant No. W911NF-18-1-0203), which was used to purchase laboratory equipment to characterize and fabricate transducers.

## REFERENCES

- [1] V. D. Nguyen, G. S. Soh, S. Foong, and K. Wood, "De-coupled dynamics control of a spherical rolling robot for waypoint navigation," in *Proc. IEEE Int. Conf. Cybern. Intell. Syst. IEEE Conf. Robot., Automat. Mechatron.*, 2017, pp. 562–567, doi: [10.1109/iccis.2017.8274838](https://doi.org/10.1109/iccis.2017.8274838).
- [2] R. M. N. Alexander, "Models and the scaling of energy costs for locomotion," *J. Exp. Biol.*, vol. 208, no. 9, pp. 1645–1652, 2005. [Online]. Available: <https://doi.org/10.1242/jeb.01484>
- [3] S. Bergbreiter, "Effective and efficient locomotion for millimeter-sized microrobots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2008, pp. 4030–4035. [Online]. Available: <https://doi.org/10.1109/iros.2008.4651167>
- [4] A. Behar, J. Matthews, F. Carsey, and J. Jones, "NASA/JPL tumbleweed polar rover," in *Proc. IEEE Aerosp. Conf. Proc.*, 2004, doi: [10.1109/aero.2004.1367622](https://doi.org/10.1109/aero.2004.1367622).
- [5] L. Lu and M. Chu, "Shock absorber design and analysis of a catapult-spherical reconnaissance robot," *Adv. Mater. Res.*, vol. 662, pp. 694–697, 2013.
- [6] L.-H. Chen *et al.*, "Soft spherical tensegrity robot design using rod-centered actuation and control," *J. Mech. Robot.*, vol. 9, no. 2, 2017.
- [7] Y. Masuda and M. Ishikawa, "Development of a deformation-driven rolling robot with a soft outer shell," in *Proc. IEEE Int. Conf. Adv. Intell. Mechatronics*, 2017, pp. 1651–1656, doi: [10.1109/aim.2017.8014255](https://doi.org/10.1109/aim.2017.8014255).
- [8] R. Rotundus, *Durable Mobile Robot for Outdoor Surveillance*, Uppsala, Sweden: Uppsala Univ.. Angstrom Space Technol. Center, 2002.
- [9] R. H. Armour and J. F. V. Vincent, "Rolling in nature and robotics: A review," *J. Bionic Eng.*, vol. 3, no. 4, pp. 195–208, 2006.
- [10] C. Batten and W. Kickbot, "A spherical autonomous robot," Massachusetts Inst. Technol., Cambridge, MA, USA, 2008. [Online]. Available: <http://www.iiit.edu/~cbatten/wor~kickbot-enibint01.pdf>
- [11] Y. Sugiyama and S. Hirai, "Crawling and jumping of deformable soft robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2004, pp. 3276–3281, doi: [10.1109/iros.2004.1389922](https://doi.org/10.1109/iros.2004.1389922).
- [12] F. Carpi, A. Tralli, D. Rossi, and P. Gaudenzi, "Martian jumping rover equipped with electroactive polymer actuators: A preliminary study," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 1, pp. 79–92, Jan. 2007, doi: [10.1109/taes.2007.357156](https://doi.org/10.1109/taes.2007.357156).
- [13] H. Tsukagoshi, M. Sasaki, A. Kitagawa, and T. Tanaka, "Design of a higher jumping rescue robot with the optimized pneumatic drive," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2005, pp. 1276–1283, doi: [10.1109/robot.2005.1570291](https://doi.org/10.1109/robot.2005.1570291).
- [14] M. Seeman, M. Broxvall, A. Saffiotti, and P. Wide, "An autonomous spherical robot for security tasks," in *Proc. IEEE Int. Conf. Comput. Intell. Homeland Secur. Pers. Saf.*, 2006, pp. 51–55, doi: [10.1109/cihsp.2006.313312](https://doi.org/10.1109/cihsp.2006.313312).
- [15] A. Halme, T. Schonberg, and Y. Wang, "Motion control of a spherical mobile robot," in *Proc. 4th IEEE Int. Workshop Adv. Motion Control*, 1996, vol. 1, pp. 259–264.
- [16] H. B. Brown and Y. Xu, "A single-wheel, gyroscopically stabilized robot," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1996, vol. 4, pp. 3658–3663.
- [17] A. Bicchi, A. Balluchi, D. Prattichizzo, and A. Gorelli, "Introducing the SPHERICLE: An experimental testbed for research and teaching in nonholonomy," in *Proc. Int. Conf. Robot. Automat.*, 1997, vol. 3, pp. 2620–2625. [Online]. Available: <https://doi.org/10.1109/robot.1997.619356>
- [18] R. Mukherjee, M. A. Minor, and J. T. Puksrushpan, "Simple motion planning strategies for spherobot: A spherical mobile robot," in *Proc. 38th Conf. Decis. Control Phoenix*, 1999, vol. 3, pp. 2132–2137.
- [19] A. Javadi and P. Mojabi, "Introducing august: A novel strategy for an omnidirectional spherical rolling robot," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2002, pp. 3527–3533.
- [20] I. Kiyoshi, H. Igarashi, and A. Murakami, "Design of a gravitational wheeled robot," *Adv. Robot.*, vol. 16, no. 7 pp. 85–793, 2002.
- [21] G. Reina, M. Foglia, A. Milella, and A. Gentile, "Rough-terrain traversability for a cylindrical shaped mobile robot," in *Proc. IEEE Int. Conf. Mechatron.*, 2004, pp. 148–153, doi: [10.1109/icmech.2004.1364428](https://doi.org/10.1109/icmech.2004.1364428).
- [22] E. Michaud *et al.*, "Autonomous spherical mobile robot to child development studies," *IEEE Trans. Syst., Man, Cybern.*, vol. 35, no. 4, pp. 471–480, Jul. 2005.
- [23] Y. Ming, D. Zongquan, Y. Xinyi, and Y. Weizhen, "Introducing hit spherical robot: Dynamic modeling and analysis based on decoupled subsystem," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2006, pp. 181–186.
- [24] V. A. Joshi, R. N. Banavar, and R. Hippalgaonkar, "Design and analysis of a spherical mobile robot," *Mech. Mach. Theory*, vol. 45, pp. 130–136, 2009.
- [25] X. Niu, A. P. Suherlan, G. S. Soh, S. Foong, K. Wood, and K. Otto, "Mechanical development and control of a miniature nonholonomic spherical rolling robot," in *Proc. 13th Int. Conf. Control Automat. Robot. Vis.*, 2014, pp. 1923–1928, doi: [10.1109/icarv.2014.7064610](https://doi.org/10.1109/icarv.2014.7064610).
- [26] I. Must, T. Kaasik, I. Baranova, U. Johanson, A. Punning, and A. Aabloo, "A power-autonomous self-rolling wheel using ionic and capacitive actuators," *Electroactive Polym. Actuators Devices*, vol. 9430, 2015, Art. no. 9430Q, doi: [10.1117/12.2084252](https://doi.org/10.1117/12.2084252).
- [27] J. Wu *et al.*, "A wheeled robot driven by a liquid-metal droplet," *Adv. Mater.*, vol. 30, 2018, Art. no. 1805039.
- [28] M. Artusi, M. Potz, J. Aristizabal, C. Menon, S. Cocuzza, and S. Debebi, "Electroactive elastomeric actuators for the implementation of a deformable spherical rover," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 1, pp. 50–57, Feb. 2011. [Online]. Available: <https://doi.org/10.1109/tmech.2010.2090163>
- [29] H. C. H. Chiu, M. Rubenstein, and W.-M. Shen, "Deformable wheel-A self-recovering modular rolling track," *Distrib. Auton. Robot. Syst.*, vol. 8, pp. 429–440, 2009.
- [30] J. Sastra, S. Chitta, and M. Yim, "Dynamic rolling for a modular loop robot," *Int. J. Robot. Res.*, vol. 28, no. 6, pp. 758–773, 2009.
- [31] X. Huang *et al.*, "Chasing biomimetic locomotion speeds: Creating untethered soft robots with shape memory alloy actuators," *Sci. Robot.*, vol. 3, 2018, Art. no. eaau7557.
- [32] X. Zhang *et al.*, "Photoactuators and motors based on carbon nanotubes with selective chirality distributions," *Nat. Commun.*, vol. 5, 2014.
- [33] W.-B. Li, W.-M. Zhang, H.-X. Zou, Z.-K. Peng, and G. Meng, "A fast-rolling soft robot driven by dielectric elastomer," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 4, pp. 1630–1640, Aug. 2018, doi: [10.1109/tmech.2018.2840688](https://doi.org/10.1109/tmech.2018.2840688).
- [34] W. Li *et al.*, "Electrically activated soft robots: Speed up by rolling," *Soft Robot.*, vol. 8, no. 5, pp. 611–624, 2021.
- [35] K. Kim, A. K. Agogino, and A. M. Agogino, "Rolling locomotion of cable-driven soft spherical tensegrity robots," *Soft Robot.*, vol. 7, no. 3, pp. 346–361, 2020, doi: [10.1089/soro.2019.0056](https://doi.org/10.1089/soro.2019.0056).
- [36] O. Farias, Jr. *et al.*, "Soft wheel robot," 2022, Accessed on: Jan. 18, 2022 [Online]. Available: <https://softroboticstoolkit.com/soft-wheel-robot>
- [37] M. Potz, M. Artusi, M. Soleimani, C. Menon, S. Cocuzza, and S. Debebi, "Rolling dielectric elastomer actuator with bulged cylindrical shape," *Smart Mater. Struct.*, vol. 19, no. 12, 2010, Art. no. 127001.
- [38] K. W. Wait, P. J. Jackson, and L. S. Smoot, "Self-locomotion of a spherical rolling robot using a novel deformable pneumatic method," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2010, pp. 3757–3762.
- [39] N. Correll, C. D. Onal, H. Liang, E. Schoenfeld, and D. Rus, "Soft autonomous materials- Using Active elasticity and embedded distributed computation," in *Proc. Int. Symp. Exp. Robot.*, 2010, pp. 227–240.
- [40] E. Steltz, A. Mozeika, J. Rembisz, N. Corson, and H. M. Jaeger, "Jamming as an enabling technology for soft robotics," *Electroactive Polym. Actuators Devices*, vol. 7642, 2010, Art. no. 764225.
- [41] S. Hauser, P. Eckert, A. Tuleu, and A. Ijspeert, "Friction and damping of a compliant foot based on granular jamming for legged robots," in *Proc. 6th IEEE Int. Conf. Biomed. Robot. Biomechatron.*, 2016, pp. 1160–1165, doi: [10.1109/biorob.2016.7523788](https://doi.org/10.1109/biorob.2016.7523788).
- [42] E. Acome *et al.*, "Hydraulically amplified self-healing electrostatic actuators with muscle-like performance," *Science*, vol. 359, no. 6371, pp. 61–65, 2018.
- [43] S. Mitchell *et al.*, "An easy-to-implement toolkit to create versatile and high-performance HASEL actuators for untethered soft robots," *Adv. Sci.*, vol. 6, Art. no. 1900178, 2019.
- [44] P. Rothemund, S. Kirkman, and C. Keplinger, "Dynamics of electrohydraulic soft actuators," *Proc. Nat. Acad. Sci.*, vol. 117, no. 28, pp. 16207–16213, 2020. [Online]. Available: <https://doi.org/10.1073/pnas.2006596117>

- [45] A. V. Rao, *Dynamics of Particles and Rigid Bodies: A Systematic Approach*, Cambridge, U.K.: Cambridge Univ. Press, 2006.
- [46] R. Goebel, R. Sanfelice, and A. Teel, "Hybrid dynamical systems," *IEEE Control Syst. Mag.*, vol. 29, no. 2, pp. 28–93, Apr. 2009, doi: [10.1109/MCS.2008.931718](https://doi.org/10.1109/MCS.2008.931718).
- [47] R. Goebel, R. Sanfelice, and A. Teel, *Hybrid Dynamical Systems*, Princeton, NJ, USA: Princeton Univ. Press, 2012, [Online]. Available: <https://doi.org/10.1515/9781400842636>
- [48] "Fit ODE, problem-based," MathWorks, Portola Valley, CA, USA, 2021, [Online]. Available: <https://www.mathworks.com/help/optim/ug/fit-ode-problem-based-least-squares.html>
- [49] S. H. Lee, J.-K. Park, C. R. Lee, and K. S. Suh, "The effect of low-molecular weight species on space charge and conduction in LDPE," *IEEE Trans. Dielect. Electr. Insul.*, vol. 4, no. 4, pp. 425–432, Aug. 1997.
- [50] R. J. Fleming, "Space charge in polymers, particularly polyethylene," *Braz. J. Phys.*, vol. 29, no. 2, pp. 280–294, 1999.
- [51] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw.*, 1995, vol. 4, pp. 1942–1948, doi: [10.1109/ICNN.1995.488968](https://doi.org/10.1109/ICNN.1995.488968).
- [52] R. C. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci.*, 1995, vol. 6, pp. 39–43.
- [53] J. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm Intelligence*, Burlington, MA, USA: Morgan Kaufmann, 2001.
- [54] Y. Shi and R. C. Eberhart, "A modified particle swarm optimizer," in *Proc. IEEE Int. Conf. Evol. Comput.*, 1998, pp. 69–73, doi: [10.1109/ICEC.1998.699146](https://doi.org/10.1109/ICEC.1998.699146).
- [55] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in *Proc. Int. Conf. Evol. Program. VII*, 1998, pp. 591–600, doi: [10.1007/BFb0040810](https://doi.org/10.1007/BFb0040810).
- [56] R. Seyab, "Nonlinear Model Predictive Control Using Automatic Differentiation," Bedford, U.K.: Cranfield Univ., 2006.
- [57] K. Ly *et al.*, "Miniaturized circuitry for capacitive self-sensing and closed-loop control of soft electrostatic transducers," *Soft Robot.*, vol. 6, pp. 673–686, 2019, [Online]. Available: <https://doi.org/10.1089/soro.2020.0048>
- [58] B. K. Johnson *et al.*, "Identification and control of a nonlinear soft actuator and sensor system," *IEEE Robot. Automat. Lett.*, vol. 5, no. 3, pp. 3783–3790, Jul. 2020, doi: [10.1109/LRA.2020.2982056](https://doi.org/10.1109/LRA.2020.2982056).
- [59] N. Kellaris, V. G. Venkata, P. Rothemund, and C. Keplinger, "An analytical model for the design of Peano-HASEL actuators with drastically improved performance," *Extreme Mech. Lett.*, vol. 29, 2019, Art. no. 100449, [Online]. Available: <https://doi.org/10.1016/j.eml.2019.100449>



**Khoi Ly** received the B.S. degree in mechanical engineering from Texas Tech University, Lubbock, TX, USA, in 2015, the M.S. and Ph.D. degrees in mechanical engineering from University of Colorado, Boulder, CO, USA, in 2017 and 2022 respectively..

He is currently a Postdoctoral Research Associate with Organic Robotics Laboratory, Department of Mechanical Engineering, Cornell University, Ithaca, NY, USA. He is interested in developing organic robotic systems that are inspired by nature to bridge the gap of conventional robotics and biological systems.



**Jatin V. Mayekar** received the B.S. degree in mechanical engineering from K.J. Somaiya College of Engineering, Mumbai, India, in 2017, and the M.S. degree in mechanical engineering from the University of Colorado Boulder, CO, USA, in 2019.

He was a Graduate Researcher under Dr. Nikolauas Correll, focused on the development of the HASEL wheel with the University of Colorado Boulder. He is currently a Robotics Applications Engineer with ABB Inc., Auburn Hills, MI, USA, where he is working on developing robotic manipulators for automated factories.



**Sarah Aguasvivas** received the B.S. and M.S. degrees in aerospace engineering from the Pennsylvania State University, PA, USA. He is currently working toward the Ph.D. degree in computer science with the University of Colorado Boulder, CO, USA.

She is currently a Researcher with the Correll Laboratory working on deployed intelligence in low-cost microcontrollers.

Miss Aguasvivas was the recipient awarded with the 2020 Berkeley Rising Stars recognition.



**Christoph Keplinger** received the M.S. and Ph.D. degrees in physics from the Johannes Kepler University, Linz, Austria, in 2008 and 2011, respectively.

He is a Director with the Max Planck Institute for Intelligent Systems (MPI-IS), Stuttgart, Germany, where he heads the Robotic Materials Department, with a current focus on soft robotics, energy capture, and functional polymers. He is an Eminent Visiting Professor of soft robotics with the University of Colorado Boulder, Boulder, CO, USA. Before joining the Max Planck Society in 2020, he was an Assistant Professor of mechanical engineering and a Fellow of the materials science and engineering program with the University of Colorado Boulder, where he also held an endowed appointment as Mollenkopf Faculty Fellow.



**Mark E. Rentschler** (Senior Member, IEEE) received the B.S. degree in mechanical engineering from the University of Nebraska, Lincoln, NE, USA, the M.S. degree in mechanical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, and the Ph.D. degree in biomedical engineering from the University of Nebraska, Lincoln, NE, USA.

He was a National Defense Science and Engineering Graduate Fellow with Massachusetts Institute of Technology. He is currently a Professor and Sylvia Norviel Cancer Research Faculty Fellow in mechanical engineering with the University of Colorado, Boulder, CO, USA. He also holds a secondary appointment with the Department of Surgery, University of Colorado Anschutz Medical Campus, Aurora, CO, USA. His research interests are in medical devices and surgical robotics.

Dr. Rentschler is also a Fellow of ASME and a Senior Member of the National Academy of Inventors.



**Nikolaus Correll** (Senior Member, IEEE) received the Diploma in electrical engineering from the Eidgenössische Technische Hochschule Zürich, Zürich, Switzerland, in 2003, and the Ph.D. degree in computer science from the École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 2007.

He was a Post-doctoral Associate with the Massachusetts Institute of Technology's, Computer Science and Artificial Intelligence Laboratory till 2009. He is currently an Associate Professor of computer science with the University of Colorado, Boulder, CO, USA. His research interests include robotic materials and robotic manipulation.