

ARTEMIS: An Open-Source, Full-sized Humanoid Robot for Dynamic Locomotion

Taoyuanmin Zhu[†], Min Sung Ahn[†], and Dennis Hong

Abstract—This paper presents ARTEMIS, an full-sized humanoid robot designed for dynamic motions. With 20 active degrees of freedom using custom proprioceptive actuators, ARTEMIS is capable of walking up to 2.1 m/s using a model-based control approach, making it one of the fastest humanoid robots at the time. It can also seamlessly transition between walking and running, making it the first platform entirely developed in academia to demonstrate such capabilities. This paper explains the details of the platform as well as the controller. ARTEMIS’s performance and robustness are validated on various outdoor terrains as well as by winning a global robotics soccer competition. Having validated the platform, we open-source it to the wider community, starting from its actuation approach to the robot model with baseline controllers, to provide an accessible foundation for making custom humanoids.

I. INTRODUCTION

Humanoids are increasingly becoming a viable solution to provide additional productivity in the coming age of workforce shortage. We are seeing multiple companies from around the world developing humanoids with the goal of making them do useful tasks such as manipulation in factories, within homes, and ultimately at any place humans are at [1]–[5]. We are still at the very beginning stages of this effort however, with the strategy on how to do it still unknown and many different approaches being explored. There are different facets to the challenge of achieving a general purpose humanoid capable of manipulating diverse objects similar to humans, as well as traversing all terrains that humans can go. While the immediate obstacle may be the availability of data to train on, the other is the accessibility to a reliable, proven hardware.

To this end, we are seeing an increase in humanoids developed not only by companies for internal use, but also for distribution to other groups [6], [7]. This makes it easy for researchers and developers unfamiliar with hardware to gain access to them and solely focus on software. While these off-the-shelf solutions provide a working platform to jumpstart development, simultaneously they offer little customization, limiting the work beyond what is possible on the platform.

This work introduces ARTEMIS, an full-sized humanoid robot with 20 active degrees of freedom. At a time when off-the-shelf solutions were unavailable, ARTEMIS was created for robust locomotion on various terrains, achieve running, and ultimately to demonstrate that the use of proprioceptive



Fig. 1: ARTEMIS walking outdoors.

actuators, which had previously been validated only on quadrupeds, is a viable solution for full-sized humanoids as well. In that regard, through this paper, we present:

- 1) An overview of a custom full-sized humanoid ARTEMIS.
- 2) Model-based locomotion controller used to validate the hardware.
- 3) Open-source of the above to provide a working example for the humanoid robotics community.

This paper itself is organized by first providing an overview of ARTEMIS’s hardware and in-depth going into the motivations behind the mechanical design and the system design. Afterwards, the dynamic locomotion controller used to validate the platform is explained. Results on the locomotion performance and its robustness are presented prior to concluding the paper.

II. RELATED WORKS

A. Humanoid Design

Earlier humanoid platforms predominately used high gear reduction actuators coupled with force/torque sensors for feedback control. ASIMO, HRP series robots, and HUBO are some of the most successful robots that used such an approach, which delivered high torque density but suffered from limited impact resistance and diminished backdrivability [8]–[11]. These platforms could all achieve stable locomotion, but typically exhibited slow, quasi-static gaits that lacked the dynamic qualities in locomotion that nature demonstrates. Series Elastic Actuators [12] are another alternative which addresses these limitations by incorporating a physical compliant element between the gearbox and the load. Demonstrated in platforms like ATRIAS, THOR, and

[†]Equal contribution.

All authors are with the Robotics and Mechanisms Laboratory (RoMeLa) at the University of California, Los Angeles (UCLA), Los Angeles, CA 90095, USA. {tymzhu, aminsung, dennishong}@ucla.edu

WALK-MAN, impact absorption and force controllability was improved, unlocking more energy efficient locomotion, but control bandwidth was severely impacted [13]–[15].

Only recently have proprioceptive actuators [16] gained prominence in legged robots, starting from their application in quadrupeds [17]–[19]. These pair high-torque motors with low-reduction, high-efficiency transmissions to achieve torque control directly through motor current at the cost of torque density. The reduced reflected inertia also significantly improves dealing with impacts and unlocking dynamic motions, allow a good compromise between the traditional high torque density actuators and the compliant SEAs.

The kinematics of the lower body for a humanoid can significantly affect its locomotion capabilities. Most platforms approximate the human hip’s spherical joint using three intersecting rotary joints. This allows for compact packaging around the hip while maintaining the necessary range of motion for dynamic motions. The knee joint, which often bears the most power requirements, often utilizes a single revolute joint while some incorporate parallel springs to improve energy efficiency during stance phases [20]. For the ankles, typically two degrees of freedom controlling the foot’s pitch and roll motion are designed for to provide stability on uneven terrain. The foot design itself varies from rigid, flat structures to more complex designs that include passive/active toe segments for more human-like gaits.

B. Humanoid Control

Control frameworks have also evolved significantly over the past decade to achieve increasingly dynamic behaviors. Early approaches relied heavily on Zero Moment Point (ZMP) [21] as the stability criteria, planning long horizon trajectories and trying to track those trajectories. However, more recent advances have moved towards dynamic control paradigms such as Model Predictive Control [22]–[24] which lack a formal stability guarantee, but have shown to be more resilient to unexpected disturbances. These often result in continuous adaptation of footstep placement and center of mass trajectories using either a very simplified model such as the linear inverted pendulum model (LIPM), or the full order model at the cost of compute to achieve highly dynamic motions and recovery behaviors. A fine balance between model complexity, re-planning frequency, and compute budget is required to achieve dynamic locomotion.

Given the desired task/joint space trajectories, the whole body control (WBC) framework is often used to enable coordinated motion across all degrees of freedom [25], [26]. Often formulated as a quadratic program, the optimization solves for joint torques, joint accelerations, and ground reaction forces. Modern day compute allows these optimizations to be solved online when paired with the right optimization solver [27], [28]. If the full order model is used instead or non-convex constraints are included, which makes the problem nonlinear, a slower update frequency may be required [29], [30].

More recently, we are seeing the rise of learning-based approaches for locomotion. The locomotion objective is

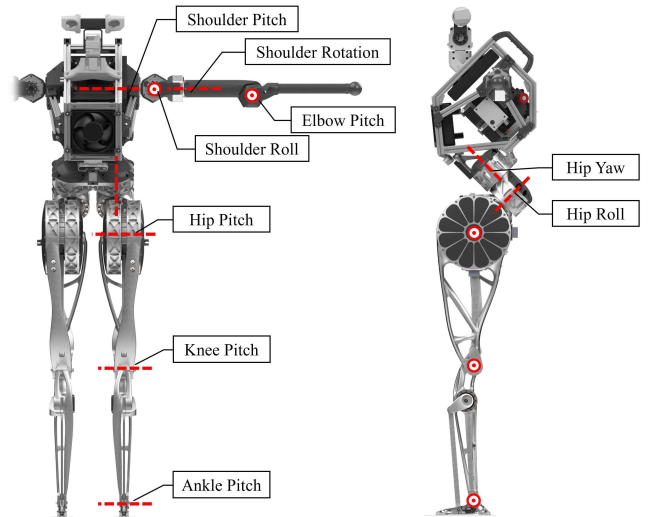


Fig. 2: ARTEMIS’s joint configuration.

achieved in a reinforcement learning framework where primarily, the successful approaches have purely trained in simulation to leverage the efficient, parallel training pipelines that are readily available [31]. Few works have also tried to primarily learn on hardware with sample efficient approaches [32], [33], while more recently, we are seeing motion capture data being utilized to further provide a stronger, direct signal to locomotion [34]–[37]. However, the critical shortcoming and active research area in this field is the sim-to-real gap that learning-based approaches must overcome to successfully realize the policies on hardware [38].

Based on this, ARTEMIS takes the state-of-the-art approaches in both hardware and software. Utilizing proprioceptive actuators for their impact mitigation and control bandwidth, along with an innovative joint architecture and model-based control, we seek to provide a validated approach and platform that other researchers could build off of.

III. PLATFORM OVERVIEW

In this section, we explain the hardware and software behind ARTEMIS. We first introduce an overview of the motivation behind the design and its capabilities, and then discuss the primary software stack that is used for dynamic locomotion.

A. Hardware Overview

1) *Mechanical Design*: ARTEMIS is a 37 kg, electrically actuated humanoid standing 140 cm tall with 20 degrees of freedom. It represents a significant departure from traditional humanoids by prioritizing dynamic motions through strategic distribution of mass, innovative configuration of the joints, and a focus on lightweight structures. The integration of these principles result in a platform specifically targeted for dynamic, robust, and fast locomotion and maneuvers.

Most notably, ARTEMIS features a 5 DoF leg as seen in Figure 2, which differs from the conventional 6 DoF leg seen in other humanoids. It consists of a 45° tilted hip

TABLE I: Lower-Body Actuator Specifications

	Hip roll and yaw	Hip and ankle pitch	Ankle pitch
Approximate weight (g)	850	3000	350
Gear reduction	14.5	5.9	50.3
Peak torque (Nm)	85	250	25
Continuous torque (Nm)	30	80	8
Velocity (rad/s)	25	23	30
Reflected inertia (kgm ²)	0.049	0.088	0.020

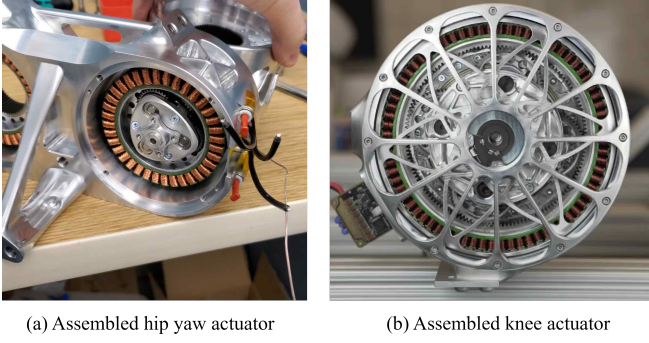


Fig. 3: Assembled actuators.

yaw and hip roll axes configuration to equalize the torque distribution between the two joints while providing better femur clearance for the hip pitch motion. This approach results in a substantial range of motion and allows for diverse poses and movements. Following the knee pitch is the ankle pitch, with ARTEMIS notably omitting the ankle roll joint, a design choice that sacrifices single-leg static balancing capability.

Additionally, ARTEMIS’s mass distribution is strategically distributed for dynamic motions. As shown in Figure 4, Over 60% of each leg’s mass is concentrated around the hip area, significantly reducing distal mass compared to traditional designs, enabling faster leg acceleration during swing phase for fast locomotion as well as robust recoveries. This is achieved in multiple ways. Firstly, ARTEMIS employs linkage-based transmission systems to relocate actuators proximally, reducing the distal mass as much as possible. For example, the knee actuator is positioned coaxially with the hip pitch actuator while the ankle pitch actuator is located closer to the knee. Secondly, the structural components were designed using topology optimization to maximize rigidity while minimizing the weight. This process involved defining boundary conditions, conducting topology studies to maximize rigidity and minimize deflection, and then creating a parametric model for further refinement through FEA analysis.

Above the legs, as seen in Figure 5, the torso houses the compute unit, electronics, and batteries within a flexible T-slotted aluminum framing structure, which allows flexibility for additional modifications in the future, whether it is for aesthetic reasons or for additional compute. The torso also includes a dedicated cooling system with a front-mounted fan to properly dissipate the heat from the computer and peripheral electronics. Additionally, the battery cage is me-

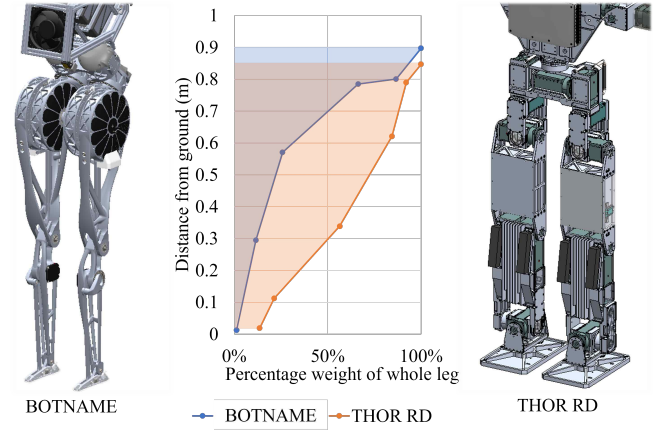


Fig. 4: Mass distribution of the lower body compared to a traditional design (THOR-RD).

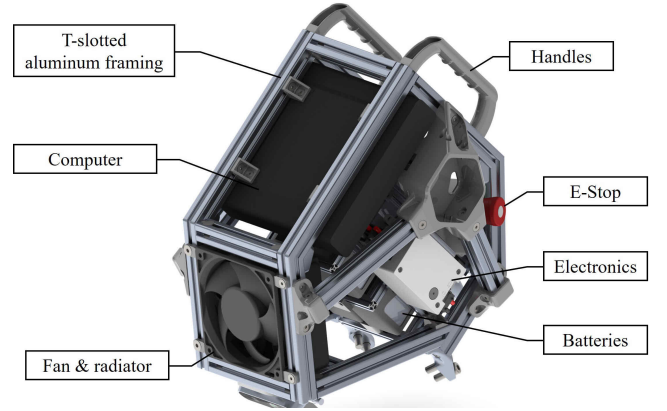


Fig. 5: ARTEMIS’s torso.

chanically isolated within the torso to protect the batteries during potential impacts.

Since ARTEMIS is designed primarily for dynamic motions, the upper body includes 4 DoF arms seen in Figure 2, which is used for counterbalancing during locomotion rather than manipulation. Each arm has three joints at the shoulder (pitch-roll-pitch configuration) and one joint at the elbow. Carbon fiber is used as the structural component to minimize weight while retaining strength. The arms can then be swung fast enough to generate counteracting momentum for balance while also being strong enough to support the robot’s weight during get-up maneuvers.

The head contains the last two degrees of freedom for the pan and tilt motion of the neck. An integrated stereo camera with an inertial measurement unit enables visual simultaneous localization and mapping (vSLAM).

2) *System Overview*: ARTEMIS is integrated with a comprehensive suite of sensors, computing resources, and custom electronics that are optimized for dynamic locomotion through real-time state estimation and control. To meet the demands of real-time model-based control, ARTEMIS uses an HP Elite Mini 800 G9 which has a 12th Gen Intel Core i7-12700T processor (12 cores/20 threads) that is paired with an NVIDIA GeForce RTX 3050 Ti GPU on-board. This config-

uration provides sufficient processing power in a small form factor while ensuring compatibility with future upgrades and CUDA capabilities for learning-based approaches, stereo vision processing, and visual SLAM.

The main perception system on ARTEMIS is the Stereolabs ZED 2 stereo camera that is mounted on the neck, providing learning-based depth sensing and object detection capabilities. For balancing, ARTEMIS relies on the MicroStrain 3DM-CV7 inertial measurement unit, selected for its distinct $1.5^\circ/\text{h}$ gyroscope bias instability and 1000 Hz data rate, which significantly outperformed its previous generations and approached the quality of the more expensive fiber optic gyroscope systems.

ARTEMIS also has custom-designed foot sensors that prioritize robustness over precision (provided by the traditional force-torque sensors). The sensors use hall-effect linear encoders with $0.4\ \mu\text{m}$ resolution to measure the deflection of the cantilever beams inside the aluminum foot structure. This approach provides reliable contact detection while providing mechanical overload protection through retention features that prevent beam failure during high-impact events such as running or jumping. Each foot's PCB also has a 6-axis IMU to support multi IMU based state estimation.

The compute and the sensors are powered through custom 99.9 Wh lithium battery packs designed to be below the FAA limit for travel. It delivers a 22.2V nominal voltage and weighs 650g per module. ARTEMIS typically operates with four battery modules, with separate modules dedicated to compute and actuators. Safe operation of the robot is achieved through a wireless emergency stop system that comprises of a receiver integrated into the robot and a handheld transmitter for remote operation. It allows immediate power cut off via high-side MOSFET switches while offering a secondary function that commands the actuators to go into a damped mode through three-phase shorts across the motor windings, preventing potential damages during emergency shutdowns.

This integration of compute, sensors, and custom electronics allows ARTEMIS to be a robust, safe, and high-performance yet quickly serviceable robot. These careful selection of components allows ARTEMIS to be a dependable research platform capable of supporting advanced control algorithms for dynamic motions, while maintaining the reliability needed for practical operation and uptime.

B. Software Overview

Humanoid robots represent the pinnacle of integration challenges because they require seamless coordination between hardware, sensors, and the appropriate software to fully utilize their potential. They need to simultaneously coordinate multiple degrees of freedom while maintaining balance, responding to environmental changes, and performing complex maneuvers. ARTEMIS is an epitome of this challenge, demanding a carefully designed software and control approach to achieve safe, stable operation and dynamic locomotion.

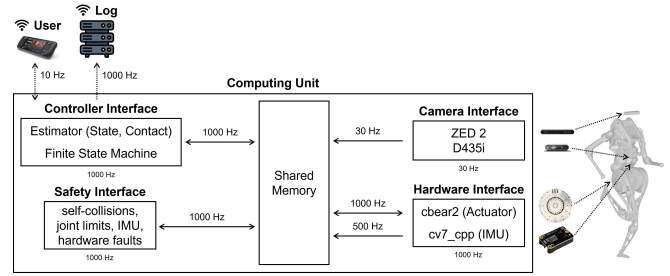


Fig. 6: Software overview.

1) Architecture Overview: The design philosophy behind the software stack is focused around modularity, reusability, and extensibility. The emphasis on these ideas is not merely a software decision, but necessary to safely develop a high degree of freedom, full-sized robot. By choosing clear interfaces between different components, the approach allows incremental testing from individual modules to the complete system, enables component-level replacement, and simplifies combining different controllers to operate collectively. This also simplifies development because researchers can focus only on their contributions while reusing the existing, validated stack. Finally, the different components can then be swapped with minimal additional integration efforts.

This approach is inspired by [17]. The stack is divided into three primary interfaces which are 1. robot interface, which is a common interface to either the simulation environment or the physical hardware, 2. controller interface, which runs the different controller-specific algorithms, and 3. safety interface, which catches abnormal behaviors. These components run concurrently and data is exchanged through shared memory.

In the robot interface, essential hardware/simulation communication and software to operate the robot are run. The hardware part of the interface facilitates communication with the actuators and the IMU. For actuators, ARTEMIS uses a custom library that communicates over dual RS-485 chains operating at 1,000 Hz with each chain responsible for the lower and the upper body. Each communication packet contains position, velocity, and torque commands and the joint state and error flags are returned. The interface to the IMU operates through a separate thread running Microstrain's SDK, except rather than using raw acceleration and angular rates, ARTEMIS samples filtered data from the 3DM-CV7 at 500 Hz. This choice negates the need for explicit acceleration and angular rate bias estimation in state estimation. The IMU signals are piped into shared memory, where it is taken both by the state estimator as well as the safety layer. The simulation interface mirrors that of the hardware, in that the same commands can be sent to the simulators with the same state information returning to the software stack. ARTEMIS can be simulated using both Gazebo and MuJoCo, providing flexibility for testing across different physics engines and supporting effortless sim-to-sim transfer and testing.

The controller interface receives the state information from shared memory and executes controller specific algorithms

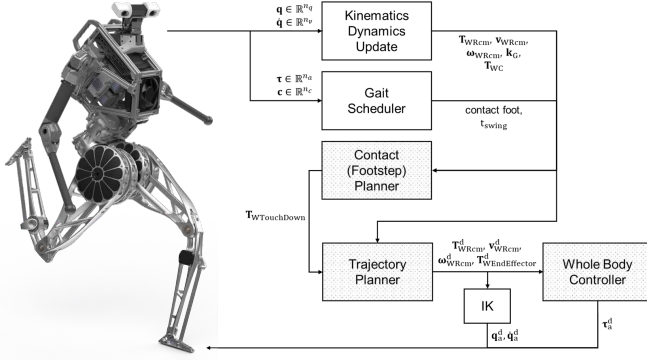


Fig. 7: ARTEMIS's locomotion controller overview.

encapsulated in their own states. A state machine manager facilitates the different states the robot should be in, which in this scope of work is primarily the balancing, walking, and running states. This modular state machine approach allows different controllers (e.g. different learning-based approaches, hardware unittests) to be added and activated depending on the robot's operational mode.

Lastly, the safety interface is a critical component for protecting the robot and the environment. Running as a separate process at 1,000 Hz, it monitors for erroneous behaviors and can immediately halt the robot from operation. The layer checks for self-collisions and monitors joint velocities against both present thresholds and expected velocities from the controller. Additionally, abnormal acceleration and angular rates from the IMU, high-frequency oscillations in joint states that could suggest instability, and error codes from the actuators are monitored. When triggered, the actuators are commanded to go into the damping mode.

2) *Dynamic Locomotion:* The default locomotion stack on ARTEMIS is a model-based approach capable of traversing uneven, discontinuous terrain by adapting its trajectories in real-time. This is achieved by sequentially estimating the state of the robot, planning footsteps to maintain a desired velocity, generating trajectories on the fly, and then tracking those plans, all within 2 ms, as seen in Figure 7. Because ARTEMIS has 5 DoF legs, dynamic balancing becomes necessary and computing the control input in real-time becomes necessary for stability. In this section, we briefly explain each of the four components.

For a floating base system, the ability to estimate the pose and the pose rates is required to calculate the kinematics and the dynamics of the robot. ARTEMIS uses the invariant Extended Kalman Filter (InEKF) to estimate the floating base pose and pose rates [39]. The InEKF propagates the dynamics forward using the IMU model and then corrects the estimation through kinematic odometry by using the contact information from the foot sensors. Unlike previous approaches, to enhance the accuracy of the estimation, two contact points (toe and heel) are independently considered as contacts to capture early toe/heel contacts.

For dynamic locomotion and stability, footstep planning is a reactive, event-based plan rather than a fixed, time based

cyclic plan. ARTEMIS's gait generation is innovative as rather than defining steps through the traditional stance and swing times, it defines a gait as a combination of *swing time* and *lift-off percentage (LOP)*. The LOP determines when a stance leg should transition to swing phase relative to the current swing leg's progress. If the LOP is set to 100%, the foot in stance will only go into swing if the current swing foot's stance phase has elapsed or if the swing foot has come in unexpected contact with the environment. Setting this parameter below 100% means the foot in stance will lift off while the current swing foot is also in the air, creating periods where both feet are in swing phase and effectively creating flight phases. This parameterization allows smooth transitions between walking (LOP of 100%) and running gaits (LOP below 100%) simply by adjusting a single parameter.

Given the desired gait timings, where to place the foot is computed. The footstep planner uses the angular linear inverted pendulum (ALIP) model [40] which recently gained popularity for its ability to capture both the linear momentum and the angular momentum of the moving body, which is especially important for bipedal robots that cannot assume a massless leg. The footstep plan is decided based on the desired angular momentum at the end of the subsequent step as in [40].

Next, the feet's trajectories and the center of mass trajectory are generated. ARTEMIS uses a quadratic program to generate trajectories online that have foot height clearance, zero initial and final velocities and accelerations for smooth contact transitions, and continuous acceleration profiles. For the sagittal/lateral directions, the coefficients of quintic splines are solved for while for the vertical motion, a ninth-order polynomial is solved. For the center of mass trajectory, only the vertical (Z) position and velocity profiles are solved for different support phases. During flight, the planner generates ballistic trajectories where the center of mass follows a projectile motion with vertical velocity decreasing linearly due to gravity. When in contact with the ground, the planner generates symmetric, convex arcs to achieve a desired lift-off velocity. The nominal air time, ground time, and lift-off velocity are all computed based on the specified swing time and LOP. This creates smooth transitions between contact and flight phases while the analytical solutions make it simple to solve in sequence with the foot trajectories.

What differentiates our approach compared to existing approaches that utilize the ALIP model is that through a Whole Body Controller (WBC), the desired angular momentum is regularized to 0. Additionally, the limitations of using a reduced-order model (ALIP) can also be mitigated as the WBC includes the full-order dynamics. Our WBC takes in the end-effector positions, torso orientations, center of mass positions, and angular momentum to solve for joint accelerations, joint torques, and ground reaction forces. ARTEMIS uses a weighted approach rather than enforcing strict hierarchies to ensure a solution is found at every control cycle while also saving compute budget.

TABLE II: Whole Body Control Weights

Tasks	Weights		
	Balancing	Walking	Running
Center of Mass Position	[10, 10, 100]	[1, 1, 100]	[1, 1, 100]
Body Orientation	[80, 80, 80]	[20, 20, 40]	[20, 20, 40]
Angular Momentum	[10, 10, 0.1]	[10, 10, 10]	[10, 10, 10]
Stance Leg Position	[1000]	[1000]	[1000]
Swing Leg Position	[10, 10, 10]	[40, 40, 40]	[40, 40, 40]
Swing Leg Orientation	[1, 1, 1]	[1, 1, 1]	[1, 1, 10]
Arm Posture	[1, 1, 1, 1]	[10, 10, 10, 10]	[10, 10, 10, 10]
Head Posture	[1, 1]	[10, 10]	[10, 10]
Force Regularization	0.001	0.001	0.005
$\ddot{q}$ Regularization	0.0001	0.0001	0.0001

IV. RESULTS

This section explains the hardware implementation details required to achieve robust, dynamic locomotion on ARTEMIS. Unlike the widely seen results on quadrupeds, manufacturing and realizing behaviors on humanoid robots are a significant challenge, especially due to the significant sim-to-real gap from imperfect physical models, sensor noise, and computational burdens. Often times, a zero-shot transfer to hardware is difficult and an iterative process of testing in simulation and on hardware is required.

The estimator is one source of potential sim-to-real gap. Although ARTEMIS’s InEKF uses two contact points per foot for improved state estimation, the hyperparameters of the estimator required careful tuning to achieve fast convergence. Even then, in the absence of perception, the X, Y position and heading direction of the robot could significantly drift (13.45m and 4.10m along X and Y directions respectively in 10 minutes of walking). But because the system computes a control input at 500 Hz and relies on the velocity signals, stable locomotion could still be achieved.

The basic kinematics/dynamics are computed using Pinocchio [41], but because the provided model is an approximation from CAD, additional dynamics such as reflected inertia were considered separately. This allowed primarily using feedforward torques as input with minimal joint PD feedback (less than 5% of the final commanded torque). For solving the quadratic programs, off-the-shelf solvers ProxQP [28] and OSQP [27] was used. The solvers parameters were tuned to satisfy the 500 Hz control loop timing constraints. Table II presents the WBC weights that were used depending on the locomotion mode.

Walking and running tests were conducted to validate the dynamic locomotion performance of the platform. To evaluate robustness, ARTEMIS was pushed and pulled from all directions during stationary walking as seen in Figure 8. Without using any perception data, ARTEMIS was also commanded to walk on terrain with random debris on the

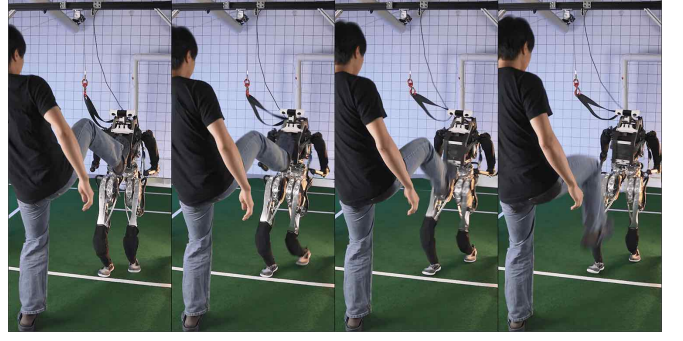


Fig. 8: When ARTEMIS is kicked from the front, it immediately adjusts its footstep (left foot) to recover.

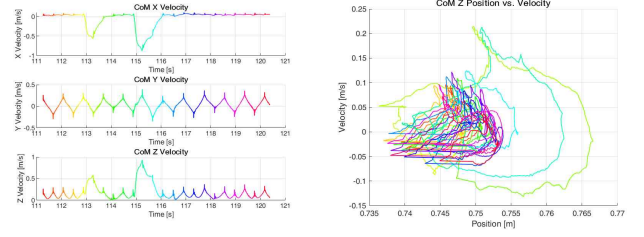


Fig. 9: Left: Center of mass linear velocities. ARTEMIS is kicked twice from the front at timestamps 113 and 115. Right: Phase plot of the center of mass height.

ground simply by detecting contact, adjusting the footstep position, and trying to maintain the center of mass height and body orientation. In both tests, ARTEMIS demonstrated exceptional robustness.

“Stability” of a platform and its controller is hard to formalize but under nominal operating conditions, there should at least exist a limit cycle. When perturbed, the phase plot will deviate from this cycle and if it is able to recover, it will return back to its original cycle. We can see in Figure 8 where ARTEMIS is kicked twice from the front, which is also captured in the CoM linear velocities shown in Section IV. When we look at the phase plot for the vertical component of the CoM in Section IV, we can see that during the kick (timestamps 113 sec and 115 sec corresponding to light green and cyan blue respectively), the cycle diverges from its cycle (0.735m~0.755m and -0.1m/s~0.1 m/s) but recovers in under a second (i.e. one complete step cycle).

Figure 10 and Figure 11 shows the joint positions, velocities, torques, and foot contact sensor force data during a walking experiment. ARTEMIS walks forward around the 9 second mark, moving at 1 m/s. Note the high spike in the joint velocities, especially for the ankle joints, due to the impact when the foot initially touches the ground. Such impacts would be a significant disturbance to traditional humanoids, but for ARTEMIS, this was not an issue because of the highly backdrivable joints and lightweight legs resulting in little impact actually being delivered to the system. Additionally, while the force distribution among the toe and the heel contact sensors are even when stepping in place,

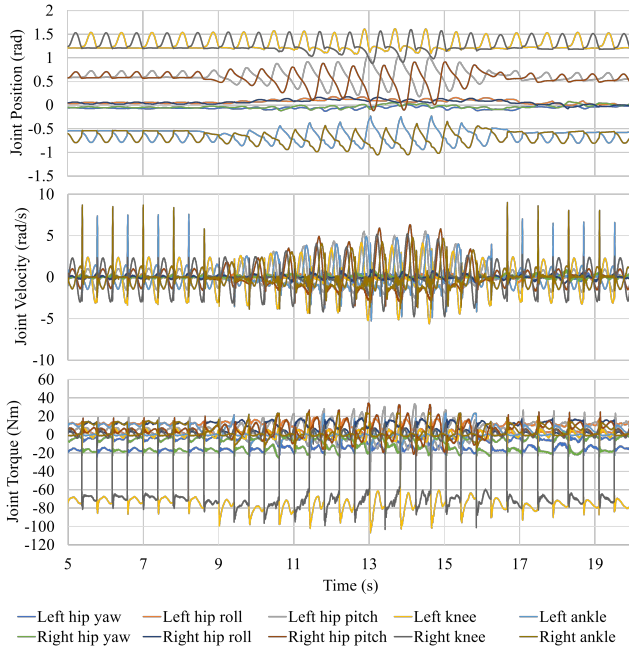


Fig. 10: Joint position, velocity and torque during walking.

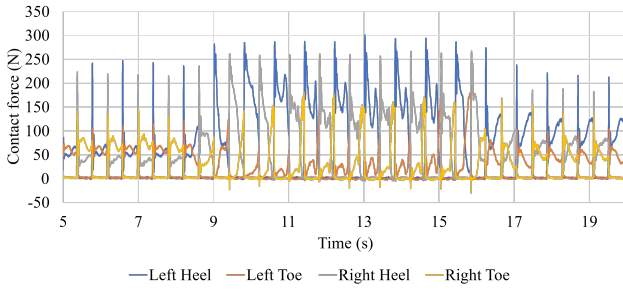


Fig. 11: Foot contact sensor reading during walking.

we can see that when walking forward, the transition of the force from heel to toe can be seen through the stance phase. By only changing the step frequency and the target speed, ARTEMIS is able to reliably walk up to 2.1 m/s.

To demonstrate even more dynamic capabilities of the robot, running was also tested on hardware. This simply requires reducing the LOP below 100%, which results in flight phases in the gait, a parabolic center of mass height, and a corresponding vertical velocity. Figure 12 shows the vertical center of mass position and velocity profile. The profiles show how gravity pulls the robot down during flight phase while during stance phase, the robot pushes onto the ground to reach a desired lift-off velocity to achieve the desired flight phase. Upon landing, the knee pitch jumps up to approximately 150 Nm to catch the robot from falling until it successfully launches the robot again into the air.

ARTEMIS was also tested outdoors on terrains such as concrete, asphalt, grass, mulch, and artificial grass. It could walk on moderate slopes and uneven grounds while completely blind, mitigating unexpected impacts by design and reacting to disturbances at 500 Hz. The extensive locomotion

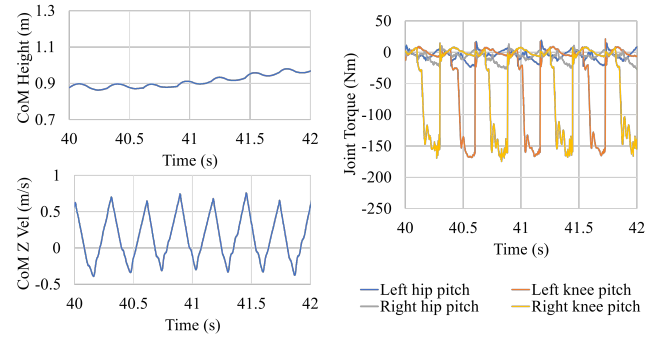


Fig. 12: Center of Mass Height, velocity and joint torque during running.

testing allowed ARTEMIS to be used to compete and win multiple humanoid competitions as seen in Figure 13 (1st in Humanoids 2023 Locomotion Competition, 1st in RoboCup 2024), demonstrating its speed compared to other robots and robust balancing capabilities during robot-to-robot collisions. Please refer to the project page (<https://artemis.romela.org>) for more results.

Given the results and testing since ARTEMIS's bringup in 2023, we open source different components of ARTEMIS to assist others with building their own iterations of actuators, robots, controllers, and peripherals. These implementations could also be adopted when customizing parts of off-the-shelf platforms. Please refer to RoMeLa's repository (<https://www.github.com/RoMeLaUCLA>) for more information.

V. CONCLUSION

This paper presents a brief summary of ARTEMIS, a full-sized humanoid designed and validated for dynamic motions. ARTEMIS deviates from conventional platforms as it uses a combination of compliant actuation and lightweight design to realize state-of-the-art model-based control. This allows ARTEMIS to be one of the fastest and robust humanoids in the world, allowing it to stably traverse in outdoor terrains and also demonstrate its competency against other hu-

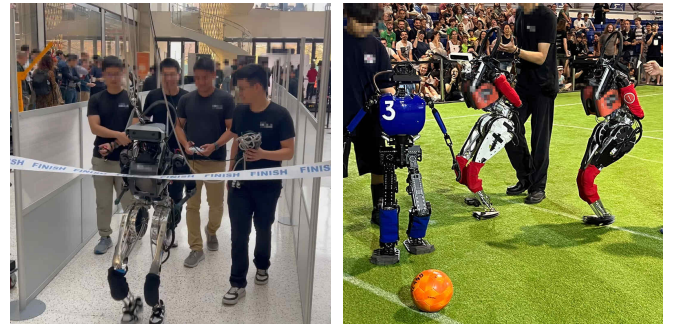


Fig. 13: Left: ARTEMIS competing in Humanoids 2023 Free Walk competition. Right: Multiple ARTEMIS competing in robot soccer competition.

manoids by winning multiple robotics competitions. As one of the few humanoids fully custom-built in academia from actuation to control, we open source parts of ARTEMIS, giving researchers a proven reference to inspire new breakthroughs in humanoid design and control.

REFERENCES

- [1] “Tesla,” <https://www.tesla.com/AI>.
- [2] “Boston dynamics,” <https://bostondynamics.com/atlas>.
- [3] “1x technologies,” <https://www.1x.tech/>.
- [4] “Agility robotics,” <https://www.agilityrobotics.com/>.
- [5] “Figure,” <https://www.figure.ai/>.
- [6] “Unitree,” <https://www.unitree.com/>.
- [7] “Booster robotics,” <https://www.boosterrobotics.com/>.
- [8] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, “The intelligent asimo: System overview and integration,” in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3. IEEE, 2002, pp. 2478–2483.
- [9] H. Inoue and H. Hirukawa, “Hrp: Humanoid robotics project of miti,” *Journal of the Robotics Society of Japan*, vol. 18, no. 8, pp. 1089–1092, 2000.
- [10] S. Lohmeier, T. Buschmann, M. Schwienbacher, H. Ulbrich, and F. Pfeiffer, “Leg design for a humanoid walking robot,” in *Humanoid Robots, 2006 6th IEEE-RAS International Conference on*. IEEE, 2006, pp. 536–541.
- [11] I.-W. Park, J.-Y. Kim, J. Lee, and J.-H. Oh, “Mechanical design of the humanoid robot platform, hubo,” *Advanced Robotics*, vol. 21, no. 11, pp. 1305–1322, 2007.
- [12] G. A. Pratt and M. M. Williamson, “Series elastic actuators,” in *Intelligent Robots and Systems 95: Human Robot Interaction and Cooperative Robots*, *Proceedings. 1995 IEEE/RSJ International Conference on*, vol. 1. IEEE, 1995, pp. 399–406.
- [13] C. Hubicki, J. Grimes, M. Jones, D. Renjewski, A. Spröwitz, A. Abate, and J. Hurst, “Atrias: Design and validation of a tether-free 3d-capable spring-mass bipedal robot,” *The International Journal of Robotics Research*, vol. 35, no. 12, pp. 1497–1521, 2016.
- [14] N. G. Tsagarakis, D. G. Caldwell, F. Negrello, W. Choi, L. Baccelliere, V. Loc, J. Noorden, L. Muratore, A. Margan, A. Cardellino, *et al.*, “Walk-man: A high-performance humanoid platform for realistic environments,” *Journal of Field Robotics*, vol. 34, no. 7, pp. 1225–1259, 2017.
- [15] S. G. McGill, S.-J. Yi, H. Yi, M. S. Ahn, S. Cho, K. Liu, D. Sun, B. Lee, H. Jeong, J. Huh, *et al.*, “Team thor’s entry in the darpa robotics challenge finals 2015,” *Journal of Field Robotics*, vol. 34, no. 4, pp. 775–801, 2017.
- [16] T. Zhu, M. S. Ahn, and D. Hong, “Design and experimental study of bldc motor immersion cooling for legged robots,” in *2021 20th International Conference on Advanced Robotics (ICAR)*. IEEE, 2021, pp. 1137–1143.
- [17] J. Hooks, M. S. Ahn, J. Yu, X. Zhang, T. Zhu, H. Chae, and D. Hong, “Alfred: A multi-modal operations quadruped robot for package delivery applications,” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5409–5416, 2020.
- [18] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, “Dynamic locomotion in the mit cheetah 3 through convex model-predictive control,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–9.
- [19] B. Katz, J. Di Carlo, and S. Kim, “Mini cheetah: A platform for pushing the limits of dynamic quadruped control,” in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 6295–6301.
- [20] Y. Gong, R. Hartley, X. Da, A. Hereid, O. Harib, J.-K. Huang, and J. Grizzle, “Feedback control of a cassie bipedal robot: Walking, standing, and riding a segway,” in *2019 American control conference (ACC)*. IEEE, 2019, pp. 4559–4566.
- [21] M. Vukobratović and B. Borovac, “Zero-moment point—thirty five years of its life,” *International journal of humanoid robotics*, vol. 1, no. 01, pp. 157–173, 2004.
- [22] F. Farshidian, E. Jelavic, A. Satapathy, M. Gifthalder, and J. Buchli, “Real-time motion planning of legged robots: A model predictive control approach,” in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*. IEEE, 2017, pp. 577–584.
- [23] R. Grandia, F. Farshidian, R. Ranftl, and M. Hutter, “Feedback mpc for torque-controlled legged robots,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 4730–4737.
- [24] G. García, R. Griffin, and J. Pratt, “Mpc-based locomotion control of bipedal robots with line-feet contact using centroidal dynamics,” in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2021, pp. 276–282.
- [25] O. Khatib, L. Sentis, J. Park, and J. Warren, “Whole-body dynamic behavior and control of human-like robots,” *International Journal of Humanoid Robotics*, vol. 1, no. 01, pp. 29–43, 2004.
- [26] L. Sentis and O. Khatib, “A whole-body control framework for humanoids operating in human environments,” in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. IEEE, 2006, pp. 2641–2648.
- [27] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, “OSQP: an operator splitting solver for quadratic programs,” *Mathematical Programming Computation*, vol. 12, no. 4, pp. 637–672, 2020. [Online]. Available: <https://doi.org/10.1007/s12532-020-00179-2>
- [28] A. Bambade, S. El-Kazdadi, A. Taylor, and J. Carpentier, “Prox-qp: Yet another quadratic programming solver for robotics and beyond,” in *RSS 2022-Robotics: Science and Systems*, 2022.
- [29] C. Mastalli, R. Budhiraja, W. Merkt, G. Saurel, B. Hammoud, M. Naveau, J. Carpentier, L. Righetti, S. Vijayakumar, and N. Mansard, “Crocodyl: An efficient and versatile framework for multi-contact optimal control,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2536–2542.
- [30] R. Grandia, F. Jenelten, S. Yang, F. Farshidian, and M. Hutter, “Perceptive locomotion through nonlinear model-predictive control,” *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3402–3421, 2023.
- [31] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, “Learning to walk in minutes using massively parallel deep reinforcement learning,” in *Conference on Robot Learning*. PMLR, 2022, pp. 91–100.
- [32] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, *et al.*, “Soft actor-critic algorithms and applications,” *arXiv preprint arXiv:1812.05905*, 2018.
- [33] L. Smith, J. C. Kew, X. B. Peng, S. Ha, J. Tan, and S. Levine, “Legged robots that keep on learning: Fine-tuning locomotion policies in the real world,” in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 1593–1599.
- [34] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, “Deepmimic: Example-guided deep reinforcement learning of physics-based character skills,” *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.
- [35] X. B. Peng, Y. Guo, L. Halper, S. Levine, and S. Fidler, “Ase: Large-scale reusable adversarial skill embeddings for physically simulated characters,” *ACM Transactions On Graphics (TOG)*, vol. 41, no. 4, pp. 1–17, 2022.
- [36] Z. Fu, Q. Zhao, Q. Wu, G. Wetzstein, and C. Finn, “Humanplus: Humanoid shadowing and imitation from humans,” *arXiv preprint arXiv:2406.10454*, 2024.
- [37] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. Kitani, C. Liu, and G. Shi, “Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning,” *arXiv preprint arXiv:2406.08858*, 2024.
- [38] S. Ha, J. Lee, M. van de Panne, Z. Xie, W. Yu, and M. Khadiv, “Learning-based legged locomotion: State of the art and future perspectives,” *The International Journal of Robotics Research*, p. 02783649241312698, 2024.
- [39] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, “Contact-aided invariant extended kalman filtering for robot state estimation,” *The International Journal of Robotics Research*, vol. 39, no. 4, pp. 402–430, 2020.
- [40] Y. Gong and J. W. Grizzle, “Zero dynamics, pendulum models, and angular momentum in feedback control of bipedal locomotion,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 144, no. 12, p. 121006, 2022.
- [41] J. Carpentier, G. Saurel, G. Buondanno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, “The pinocchio c++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives,” in *IEEE International Symposium on System Integrations (SII)*, 2019.