Efficient and Versatile Quadrupedal Skating: Optimal Co-design via Reinforcement Learning and Bayesian Optimization

Abstract— **In this paper, we propose an effective mechanical and algorithmic solution to enabling skating motion with passive wheels on state-of-the-art quadrupedal robots. The skating locomotion enables a hybrid combination of wheeled and legged mobility without the necessity of motorization at the feet, which simultaneously promote efficiency, speed, and mechanical simplicity. To realize these potential advantage of skating, we employ a bilevel optimization approach with *an upper level optimization via Bayesian Optimization (BO)* to search for the best mechanical design and *a lower level Reinforcement Learning (RL)* to find an optimal motor policy. The end results not only provide optimal mechanical and control designs but also show versatile locomotion behaviors such as *hockey stop* (a rapid braking maneuver by turning sideways to maximize friction) and *self-aligning behavior* (automatically adjusts its orientation to move more efficiently in the commanded direction), providing the first comprehensive study on quadrupedal robotic skating.**

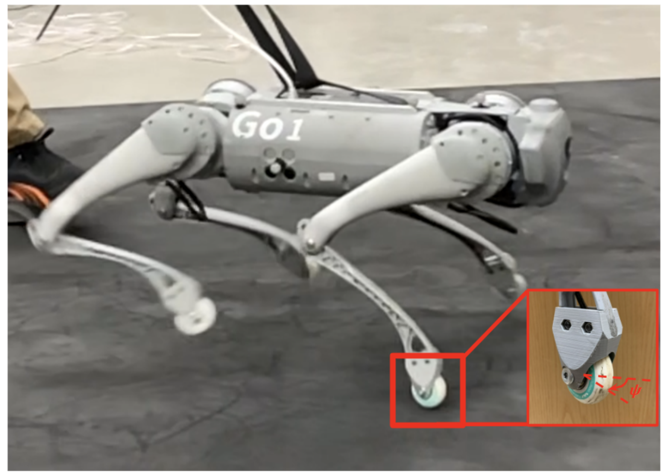

Fig. 1: Quadrupedal skating setup with passive wheels. Each foot of the Unitree Go1 is equipped with a 3D-printed roller support holding a passive wheel. The yaw installation angle ψ determines the rolling direction and is treated as a key design parameter in our co-design framework.

I. Introduction

Legged robots have recently demonstrated remarkable agility, adaptability, and robustness across diverse terrains, making them increasingly capable of real-world deployment [1], [2], [3]. Despite these advancements, locomotion efficiency and speed remain critical challenges compared to wheeled platforms [4], [5]. Wheels offer inherent energetic advantages and smooth ground interaction [4], but they typically sacrifice versatility when confronted with unstructured environments. Hybrid approaches, such as wheeled-legged robots [6], [7], attempt to reconcile these trade-offs, yet most existing solutions rely on actively motorized wheels, introducing mechanical complexity, energy costs, and control overhead.

In this work, we explore an alternative mode of hybrid mobility—*quadrupedal skating with passive wheels*. By equipping each foot with a passive wheel, a quadruped can combine the stability and maneuverability of legged locomotion with the efficiency and speed of wheeled motion, without requiring motorization at the feet. As illustrated in Fig. 1, we implement this idea by attaching custom 3D-printed roller supports with adjustable yaw angle ψ to the feet of a Unitree Go1. The mechanical simplicity of passive wheels makes the approach attractive for lightweight, energy-efficient robotic design.

However, realizing effective quadrupedal skating introduces significant challenges. Unlike traditional walking or rolling, skating requires precise coordination of ground contact forces, body posture, and momentum exchange to achieve stable and controllable motion. The absence of active wheel control also shifts greater burden onto the robot's mechanical design and high-level motor policy. This coupling between morphology and control makes *co-design* essential: poor mechanical configurations can severely limit the range of feasible skating behaviors, while suboptimal control policies can fail to exploit even well-designed morphologies [8].

To address this, we present a *bilevel optimization framework* for quadrupedal skating that jointly optimizes mechanical design and motor policy. At the upper level, we employ *Bayesian Optimization (BO)* to efficiently explore the mechanical design space [9]. At the lower level, we leverage *Reinforcement Learning (RL)* to discover optimal motor policies tailored to each candidate design [10], [11]. Through this co-design process, we identify design–policy pairs that enable both efficient and versatile skating behaviors.

Our main contributions are as follows:

- **A novel locomotion paradigm:** We introduce quadrupedal skating with passive wheels, demonstrating its potential for combining speed, efficiency, and maneuverability.
- **A bilevel co-design framework:** We propose a BO–RL approach that jointly optimizes mechanical and control parameters for skating.
- **Versatile locomotion demonstrations:** We validate our method on a state-of-the-art quadruped, showcasing diverse behaviors such as *hockey stop*(a rapid braking maneuver by turning sideways to maximize friction), and *self-aligning behavior*(automatically adjusts its orientation to move more efficiently in the commanded

direction).

- **First systematic study:** To our knowledge, this work presents the first comprehensive study of dynamic skating locomotion on quadrupedal robots.

Overall, our results show that co-design of morphology and control is a practical and effective way to realize skating in quadrupedal robots, achieving behaviors that conventional designs cannot easily support.

II. Related Work

A. Quadrupedal Locomotion

Quadrupedal robots have advanced significantly in recent decades, achieving dynamic mobility, robustness, and versatility across diverse terrains. Early hydraulic systems such as BigDog demonstrated the feasibility of rough-terrain locomotion [1], while subsequent platforms like MIT Cheetah [12] showcased model-based controllers leveraging whole-body dynamics and model predictive control to achieve agile running and jumping [3]. More recently, reinforcement learning (RL) [13], [14] has enabled quadrupeds to autonomously acquire highly dynamic skills, producing controllers that rival or surpass expert-designed strategies [2], [10]. These methods have empowered robots such as ANYmal [15] and Cheetah to traverse unstructured environments with agility and resilience. Despite these advances, pure legged locomotion remains energetically less efficient and slower than wheeled locomotion, motivating hybrid approaches such as wheel-legged robots [16], [17], [18], [19] that combine the strengths of both. However, such wheel-legged often has heavy actuation located as the toes, which increases the hardware complexity and reducing the overall reliability of the system.

B. Robot Skating

Compared to walking and wheeled driving, skating locomotion for robots has received limited attention, even though it offers a promising trade-off between efficiency and maneuverability. The key distinction of skating is that propulsion and gliding are decoupled: robots push off with legs to generate momentum while passive or low-friction contacts sustain long glides, which introduces challenges such as nonholonomic rolling constraints and reduced ground friction. A few studies in the literature has been trying to address this problem. Bjelonic et al. demonstrated quadrupedal skating with ANYmal by attaching passive wheel end-effectors and employing force control to generate glide-and-push sequences, showing that skating can reduce cost of transport compared to trotting [20]. The SlidBot platform further advanced this concept by adding passive wheels at the knees and lower legs, achieving roller-skating gaits with improved speed and energy efficiency over diagonal trot on flat and sloped terrains [21]. Similarly, Chen et al. analyzed quadrupeds with passive wheels, highlighting challenges in gait transitions and internal force coordination during walking-to-skating switches [22]. Other studies have examined external skateboard interaction [23] and biped robots with inline skates [24], confirming that skating can boost efficiency on smooth surfaces but requires specialized control. Overall, these works establish skating as a feasible but underdeveloped locomotion mode, with open challenges in maneuverability, terrain generalization, and integration with learning-based controllers. Our work addresses these gaps by co-designing morphology and control to enable quadrupedal skating with cool behaviors.

C. Robot Hardware and Motion Policy Co-Design

Co-design of robot morphology and control has emerged as an effective paradigm to unlock new locomotion capabilities. Existing approaches can be broadly categorized into three strategies.

Universal-policy methods decouple design and control: a single design-conditioned policy is trained across a distribution of morphologies and later used as a fast evaluator. Examples include morphology-randomization for general locomotion controllers [25], [26] and design-conditioned policies for actuator optimization [27]. While versatile, these methods incur substantial upfront training costs since policies must learn to handle many suboptimal morphologies.

Joint optimization methods integrate morphology parameters directly into the RL loop, simultaneously updating design and control. Such strategies can discover novel morphologies [28], [29], but often suffer from instability due to the non-stationarity of changing robot bodies and limited simulator support for online morphology changes.

Bilevel optimization methods treat co-design as a nested problem, where an outer-loop optimizer proposes candidate designs and an inner-loop RL agent trains a specialized policy for each. This approach has been applied to legged robots and manipulators [30], [31], producing policies that are experts for their respective morphologies. However, the high cost of repeated RL training makes sample efficiency crucial. Bayesian Optimization (BO) is particularly well suited in this context due to its ability to optimize expensive black-box functions in low-dimensional spaces with relatively few evaluations [32], [9].

Our work adopts this bilevel framework, combining RL for policy learning with BO for design search, and extends it to the novel setting of quadrupedal skating locomotion with passive wheels.

III. Quadrupedal Skating via Co-Design

Skating with passive wheels requires both hardware adaptation and tailored control. Neither design nor control alone is sufficient: the wheel orientation strongly affects the feasibility of skating gaits, while the policy determines whether the robot can exploit the design for stable and agile motion. This motivates a co-design formulation where hardware and control are optimized jointly. In this section, we formalize the bilevel optimization problem, describe the mechanical design space of roller supports, and introduce the optimization framework that integrates Bayesian Optimization (BO) with Reinforcement Learning (RL).

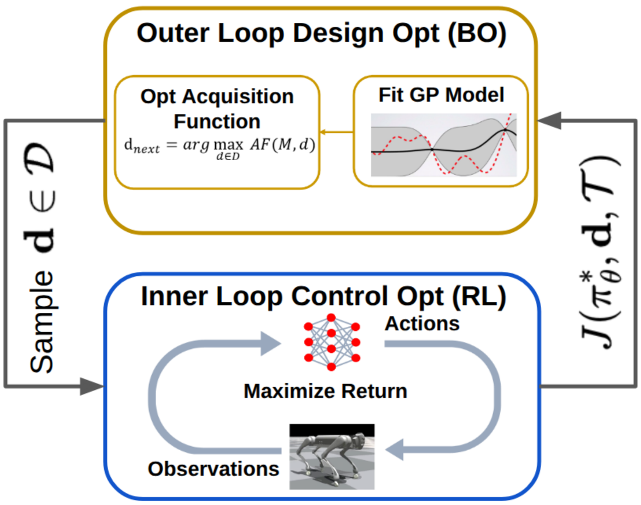

Fig. 2: Bilevel co-design framework for quadrupedal skating. The outer loop uses BO to explore design parameters. For each candidate design, the inner loop employs RL to optimize the policy π_θ. The policy performance $J(\pi_\theta^*, \mathbf{d}, \mathcal{T})$ is fed back to BO, enabling efficient search for design–policy pairs that maximize skating performance.

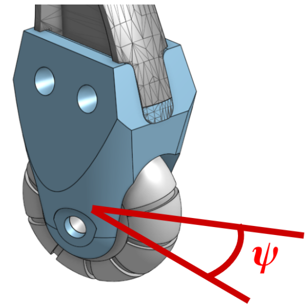

Fig. 3: The model of the roller support with yaw angle definition. The yaw angle ψ is defined as the angle between the wheel rolling direction and the sagittal plane of the robot when the robot is standing still at the desired base height using joint PD. This parameter determines the effective rolling direction of the passive wheel and serves as the key design variable in our co-design framework.

A. Co-Design via Bilevel Optimization

As illustrated in Fig. 2 We formulate quadrupedal skating as a bilevel optimization problem, where the outer loop optimizes design and the inner loop optimizes control.

For a given design $\mathbf{d}$, the inner loop determines the optimal control policy $\pi_\theta^*(\mathbf{d}, \mathcal{T})$ parameterized by θ that maximizes the expected discounted return, given the design and a task set $\mathcal{T}$, which can be formulated as

$$\pi_\theta^*(\mathbf{d}, \mathcal{T}) = \arg\max_{\pi_\theta} \mathbb{E}_{\tau \sim p(\tau|\pi_\theta, \mathbf{d}, \mathcal{T})} \left[\sum_{t=0}^{T} \gamma^t r_t \right], \qquad (1)$$

where τ denotes robot trajectories sampled by controlling the robot with design $\mathbf{d}$ using policy π_θ to perform tasks in $\mathcal{T}$, r_t are instantaneous rewards and γ is the discount factor. We use Reinforcement Learning(RL) for this inner loop control optimization because it scales much better than model-based counterparts when handling complex task sets like a distribution of commands, and complex operating environments like the one in roller skating which involves non-holonomic contacts and non-trivial gaits.

The performance of $\pi_\theta^*(\mathbf{d}, \mathcal{T})$ is evaluated using a metric $J(\pi_\theta^*, \mathbf{d}, \mathcal{T})$ using the design d and the set of tasks $\mathcal{T}$ that π_θ^* was optimized on. The outer loop design optimization problem can then be formulated as

$$\mathbf{d}^* = \arg\min_{\mathbf{d} \in \mathcal{D}} J(\pi_\theta^*, \mathbf{d}, \mathcal{T}) \qquad (2)$$

Since evaluating $J(\pi_\theta^*, \mathbf{d}, \mathcal{T})$ requires training a full policy and is non-differentiable due to stochastic RL and discrete contact dynamics, the outer loop is naturally framed as a black-box optimization problem. We therefore adopt BO as the outer-loop optimizer, coupled with RL for the inner loop. This bilevel formulation enables systematic discovery of design–policy pairs that maximize skating performance.

B. Mechanical Design of Roller Support

A critical element of our approach is the roller support hardware. We replace the toes of Unitree Go1 with 3D-printed supports that connect the legs to passive wheels, creating a lightweight skating module. The key design variable is the yaw angle ψ, which defines the rolling direction of each wheel (see Fig. 3).

A straightforward yet naive design is the "Parallel Configuration" (P Configuration), where all wheels are aligned with the x-axis ($\psi = 0°$). While this may seem optimal for forward skating, Go1's leg kinematics prevent the wheels from producing controllable forward velocity v_x when $\psi = 0°$(see Fig. 4). This limitation is not unique to Go1, but also applies to other quadrupedal robots such as MIT Cheetah [12] and Unitree B2 [33], since they share similar leg configurations. In this case, the wheel remains confined to the sagittal plane, producing little friction along its rolling direction and leaving v_x uncontrollable. Therefore, at least one of ψ_1 or ψ_2 must be nonzero, motivating a systematic co-design approach.

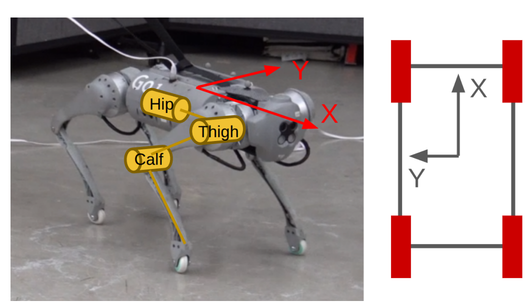

Fig. 4: Naive P Configuration ($\psi = 0°$) results in uncontrollable forward velocity v_x due to kinematic constraints of the leg design.

IV. Algorithmic Implementation

A. Inner Loop Motion Policy Optimization via Reinforcement Learning (RL)

1) RL formulation: The motion policy for each candidate design is trained using reinforcement learning within the massively parallelized *IsaacLab* simulator We formulate this learning problem as a Markov Decision Process (MDP), which is defined by the tuple ($\mathcal{O}$, $\mathcal{A}$, p, r), representing the observation space, action space, state transition dynamics, and reward function, respectively. The state transition $p(s_{t+1}|s_t, a_t)$ is governed by the rigid body dynamics of the simulated robot.

The goal is to learn an optimal policy $\pi_\theta(a_t|o_t)$ with parameters θ that maps an observation $o_t \in \mathcal{O}$ to a distribution over actions $a_t \in \mathcal{A}$. The objective is to maximize the expected discounted sum of future rewards:

$$\pi_\theta^* = \arg\max_\theta \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right] \tag{3}$$

where $\gamma \in [0, 1)$ is the discount factor. We utilize the Proximal Policy Optimization (PPO) algorithm to train the policy network. The specifics of the observation space, action space, and reward function are detailed below.

Name	Expression
Base Velocity	${}^{\mathcal{B}}\mathbf{v}$, ${}^{\mathcal{B}}\boldsymbol{\omega}$
Commanded Base Velocity	${}^{\mathcal{B}}\mathbf{v}_{xy}^d$, ${}^{\mathcal{B}}\omega_z^d$
Projected Gravity	${}^{\mathcal{B}}\mathbf{u}_g$
Joint Position and Velocity	$\mathbf{q}, \dot{\mathbf{q}}$
Last Actions	$\mathbf{a}_{\text{prev}}$

TABLE I: Observation Space.

2) Observation Space: As shown in Table I, the observation space includes the commanded and actual base linear and angular velocities, base orientation represented using projected gravity ${}^{\mathcal{B}}\mathbf{u}_g = \mathbf{R}_{\mathcal{WB}}[0, 0, -1]^T$ where $\mathbf{R}_{\mathcal{WB}}$ is the rotation matrix of body frame with respect to the world frame, joint positions and velocities, and last actions. An important aspect specific to roller skating is how linear velocity commands are given and observed.

As shown in (4), we always observe linear velocity command in base frame, but the command can be given either in base frame or in world frame. In the latter case, the command is first transformed to body frame and then observed. These two ways of giving commands will also motivate different reward functions to track the commands. We refer to the two cases as "Base Frame Command" and "World Frame Command":

$${}^{\mathcal{B}}\mathbf{v}_{xy}^d = \begin{cases} {}^{\mathcal{B}}\mathbf{v}_{xy}^d & \text{(Base Frame Command)} \\ \mathbf{R}_{\mathcal{WB}}^T\, {}^{\mathcal{W}}\mathbf{v}_{xy}^d & \text{(World Frame Command)} \end{cases} \tag{4}$$

A comparison between them will be shown in Sec. V-C.

Remark on Desired Linear Velocity: The motivation for World Frame Command lies in the nature of roller skating. While roller wheels improve energy efficiency when moving along the rolling direction, the amount of friction force we can get along the rolling direction is close to zero, which sabotages the velocity tracking performance. Fortunately, the available friction perpendicular to the rolling direction is unaffected, which means if the robot strategically change its body yaw angle such that the rolling direction is used for energy efficiency, and its perpendicular direction is used for abrupt change in linear velocity, it can achieve a good balance between energy efficiency and velocity tracking. Tracking linear velocity in base frame constrains the body relative orientation with respect to the velocity commands and hinders this potential, while tracking linear velocity in world frame does not have this limit.

3) Action Space: The actions $\mathbf{a}$ outputted from the neural network are scaled and biased to get the desired joint positions $\mathbf{q}^d$, which is then fed to the low level joint PD controller together with the PD gains k_p, k_d and a zero desired joint velocity. The desired joint torques are

$$\boldsymbol{\tau} = k_p(\mathbf{q}^d - \mathbf{q}) - k_d\dot{\mathbf{q}}. \tag{5}$$

Name	Expression
Base Linear Velocity xy	$r_{\mathbf{v}_{xy}}$ using (6) or (8)
Base Angular Velocity z	r_{ω_z} using (7) or (9)
Base Height	$(z - z^d)^2$
Base Orientation	$\left\|{}^{\mathcal{B}}\mathbf{u}_{g,xy}\right\|_2^2$
Base Linear Velocity z	${}^{\mathcal{B}}v_z^2$
Base Angular Velocity xy	$\left\|{}^{\mathcal{B}}\boldsymbol{\omega}_{xy}\right\|_2^2$
Action Rate Minimization	$\|\mathbf{a} - \mathbf{a}_{\text{prev}}\|_2^2$
Joint Acceleration Minimization	$\|\ddot{\mathbf{q}}\|_2^2$
Vertical Virtual Leg	$\left\|\mathbf{p}_{\text{virt leg, xy}}\right\|_2^2$
Collision Avoidance	$\sum \mathbf{1}_{\{\|\mathbf{f}\|_2 > f_0\}}$
Joint Position Limit	$\sum \|\mathbf{q}_{\text{exceed}}\|_2^2$

TABLE II: Reward Terms for Skating.

4) Rewards: The reward function includes tracking terms on base velocity, height, and orientation, smoothing terms on action rate and joint acceleration, heuristics that encourages the virtual legs to be vertical, and penalties for approaching physical limits like self-collision or joint position going out of range. While most rewards remain the same in this work, we need to define different rewards for the aforementioned Base Frame Command and World Frame Command. The Base Frame Command tracking reward is the common practice to track velocity commands in base frame, as shown in (6)-(7).

$$r_{\mathbf{v}_{xy}} = r_{\exp}\left(\left\|{}^{\mathcal{B}}\mathbf{v}_{xy} - {}^{\mathcal{B}}\mathbf{v}_{xy}^d\right\|_2\right) \tag{6}$$

$$r_{\omega_z} = r_{\exp}({}^{\mathcal{B}}\omega_z - {}^{\mathcal{B}}\omega_z^d) \tag{7}$$

where $r_{\exp}(e) = \exp(e^2/\sigma)$ is the exponential tracking reward for any scalar error e. The World Frame Command tracking reward is used to enable the strategic body turning mentioned in Sec. IV-A.2, where base linear velocity should be tracked in world frame, and should be prioritized over base angular velocity tracking, as encoded as (8)-(9).

$$r_{\mathbf{v}_{xy}} = r_{\exp}({}^{\mathcal{W}}v_{xy,\text{err}}) = r_{\exp}\left(\left\|{}^{\mathcal{W}}\mathbf{v}_{xy} - {}^{\mathcal{W}}\mathbf{v}_{xy}^d\right\|_2\right) \tag{8}$$

$$r_{\omega_z} = \begin{cases} r_{\exp}({}^{\mathcal{B}}\omega_z - {}^{\mathcal{B}}\omega_z^d) & ({}^{\mathcal{W}}v_{xy,\text{err}} \leq e_0) \\ \frac{e_1 - v_{xy,\text{err}}}{e_1 - e_0} r_{\omega_z}(e_0) & (e_0 < {}^{\mathcal{W}}v_{xy,\text{err}} \leq e_1) \\ 0 & ({}^{\mathcal{W}}v_{xy,\text{err}} > e_1) \end{cases} \tag{9}$$

The idea behind (9) is to only track angular velocity when the linear velocity error is small, where e_0, e_1 are two error thresholds representing small and large linear velocity tracking errors.

B. Outer Loop Design Optimization via Bayesian Optimization (BO)

1) BO Formulation: Given the metric values $J(\pi_\theta^*, \mathbf{d}, \mathcal{T})$ for all the previously tried designs, BO minimizes the metric value through two steps repeated iteratively: (i) fitting a Gaussian Process (GP) surrogate model to predict mean and uncertainty of $J(\pi_\theta^*, \mathbf{d}, \mathcal{T})$, and (ii) optimizing an acquisition function to propose the next candidate design. We adopt a phased acquisition schedule: Upper Confidence Bound(UCB) with a large exploration coefficient is used initially for broad search, while an annealed UCB focuses the search on promising regions, and finally Expected Improvement(EI) refines the best candidates.

2) Design Space: The largest possible design space include the wheel yaw angle for four legs separately $\mathbf{d} = [\psi_{FR}, \psi_{FL}, \psi_{RR}, \psi_{RL}]$. Taking into account the left-right symmetry of quadrupedal robots, $\psi_{FR} = -\psi_{FL}$ and $\psi_{RR} = -\psi_{RL}$, and thus the design space can be reduced to 2D including $\psi_{\text{front}} = \psi_{FL}$, and $\psi_{\text{rear}} = \psi_{RL}$. One can further argue that the front rear asymmetry is not significant and couple all four legs together, making $[\psi_{FR}, \psi_{FL}, \psi_{RR}, \psi_{RL}] = [\psi, -\psi, -\psi, \psi]$, as illustrated in Fig. 7. We will detail about 1D case in Sec. V-A-Sec. V-B, and the 2D case in Sec. V-D

3) Design Metric: To evaluate the quality of a design $\mathbf{d}$ in face of multiple commands sampled from $\mathcal{T}$ and the stochasticity of RL, we first define an intuitive instantaneous metric $f(s, a, c)$ which is a function of the state s, action a, and command c of a specific robot (environment), at a specific time step. As the policy is being trained in IsaacLab and a large amount of trajectory samples are readily available, we evaluate the averaged metric $J(\pi_\theta^*, \mathbf{d}, \mathcal{T})$ using a Monte Carlo estimate of the objective:

$$J(\pi_\theta^*, \mathbf{d}, \mathcal{T}) = \frac{1}{N_{\text{step}} N_{\text{env}}} \sum_{k=1}^{N_{\text{step}}} \sum_{i=1}^{N_{\text{env}}} f(s_{i,k}, a_{i,k}, c_{i,k}), \quad (10)$$

computed as the average over N_{env} parallel environments and N_{step} time steps. The commands gets sampled from $\mathcal{T}$ multiple times during N_{step} time steps. Each simulator step corresponds to $0.02\,\text{s}$. Metrics are aggregated over windows of $N_{\text{step}} = 1000$ steps ($\approx 20\,\text{s}$) across $N_{\text{env}} = 4096$ parallel environments. Velocity commands are resampled every 200 steps, and environments reset asynchronously. Although PPO updates occur every 24 steps ($\approx 0.48\,\text{s}$), metrics are aggregated over 1000-step windows to obtain stable estimates.

The instantaneous metric $f(\cdot)$ is defined as the energy efficiency, measured by the cost of transport (CoT) defined as the sum of the square torque, which is proportional to the electric power of motors, normalized by the gravitational force times the L2 norm of the actual twist $\boldsymbol{\xi} = [v_x, v_y, \omega_z]$.

$$\text{CoT} = \frac{\|\boldsymbol{\tau}\|_2^2}{mg\,\|\boldsymbol{\xi}\|_2}, \quad (11)$$

where $\boldsymbol{\tau}$ is the joint torque vector, m is the robot mass, and g is gravity. In case of Body Frame Command, $\boldsymbol{\xi} = [{}^{\mathcal{B}}v_x, {}^{\mathcal{B}}v_y, {}^{\mathcal{B}}\omega_z]$. In case of World Frame Command, $\boldsymbol{\xi} = [{}^{\mathcal{W}}v_x, {}^{\mathcal{W}}v_y, {}^{\mathcal{B}}\omega_z]$.

V. Results and Analysis

We present four experiments to demonstrate the effectiveness of our bilevel BO-RL framework in quadrupedal roller skating. In Sec. V-A, we present the baseline roller skating motion where the design is human-engineered and the control policy is trained using "Base Frame Command" tracking. In Sec. V-B, we emphasize the necessity of outer loop design optimization in the creation of energy-efficient roller skating motions. In Sec. V-C, we showcase interesting transient roller skating behaviors enabled by "World Frame Command" tracking reward. In Sec. V-D, we combine the design optimization in Sec. V-B and the reward engineering in Sec. V-C to achieve both energy efficiency and transient velocity tracking.

A. Baseline via Human Designed Roller Skating

This section introduces the baseline human designed wheel angles. To solve the uncontrollability issue mentioned in Sec. III-B, we introduce a minimal modification in design, where all wheel angles are parameterized by one single yaw angle ψ. As shown in Fig. 7, the resulting gait is trotting where the diagonal legs alternate to contact with the ground. Intuitively, the robot is coupling a diagonal pair of legs as if they form a roller skating shoe.

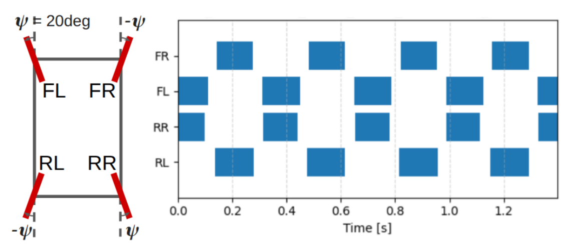

Fig. 7: Baseline Human-Designed Roller Skating.

B. Necessity of Co-Design for Energy Efficiency Improvement

In this section, we showcase the necessity of co-optimizing both design and control. In the inner loop RL used to train the control policies, we use the "Base Frame Command" observations and rewards because we need to compare energy efficiency when tracking commands in different directions, and thus it is desirable to specify the relative orientation between the command and the robot. In the outer loop BO used to optimize designs, we choose the design space to be the minimal 1D wheel yaw angle ψ used to generate the four wheel yaw angles as shown in Fig. 7. The BO objective is the general average of CoT shown in (10)-(11).

Fig. 8 plots the steady state CoT values in polar coordinates when the robot is tracking a body frame linear velocity command of magnitude 1.5 m/s given in different directions. The angle of polar coordinates is determined by the direction

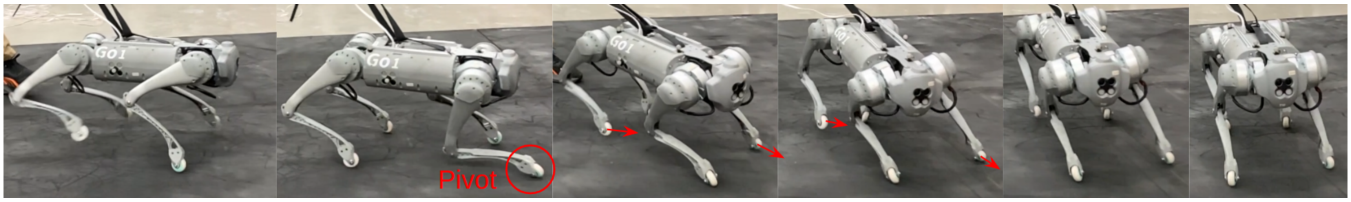

Fig. 5: Hockey Stop

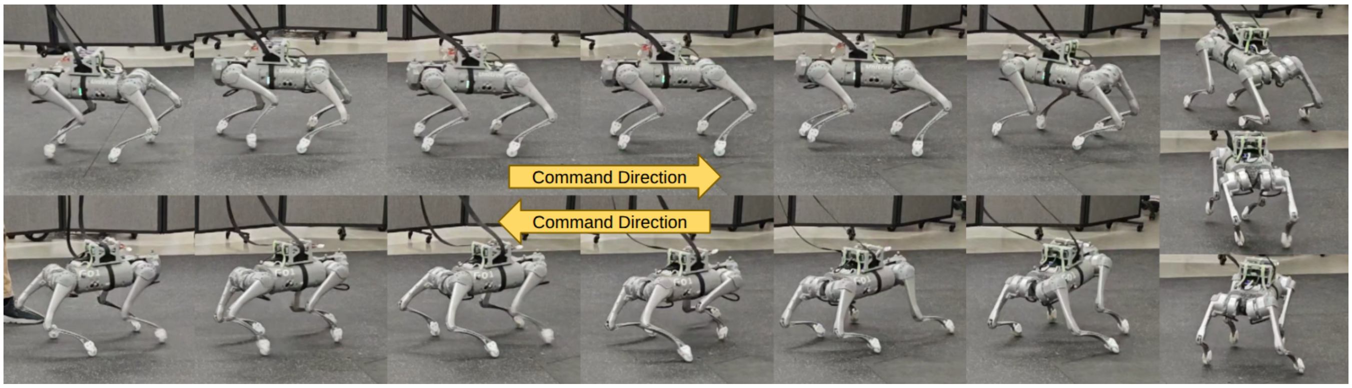

Fig. 6: Self-Aligning Behavior

of the command $\alpha = \arctan 2({}^{\mathcal{B}}v_x^d, {}^{\mathcal{B}}v_y^d)$, and the radius of polar coordinates is the CoT value. Although the BO optimized design can be more energy efficient than walking in almost all directions, the human synthesized design is only more efficient in the forward direction ($\alpha = 0$ deg). This suggests that non-trivial co-optimization of both design and control is needed when synthesizing quadrupedal roller skating behaviors.

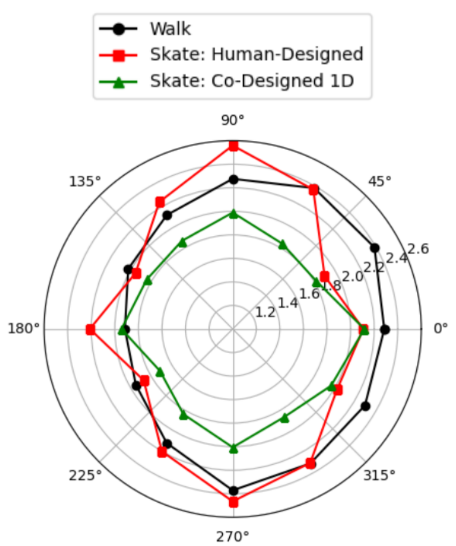

Fig. 8: CoT comparison between walking and skating.

C. Comparison between Base Frame and World Frame Commands

In this section, we investigate the effect of tracking linear velocity in the world frame instead of the body frame, and of prioritizing linear velocity tracking over angular velocity tracking. These two reward designs enable the emergence of the so-called "hockey stop" behavior, well known in the roller skating community, when the robot is constrained to the intuitive design shown in Fig. 7.

As illustrated in Fig. 5, when the robot is moving quickly and receives a sudden stop command, it learns to rotate its body sideways to align with the velocity direction. This orientation corresponds to the direction perpendicular to the wheel rolling axis, where the robot can generate maximum lateral friction force to decelerate. The resulting gait involves using the front right wheel as a pivot while allowing the front left and rear right wheels to roll, effectively turning the body around the pivot wheel. The robot then exploits the sliding friction of all wheels to achieve a rapid stop.

With this hockey stop strategy, the stop time from an initial forward velocity of 2 m/s is reduced by approximately 50%. This demonstrates how world-frame velocity tracking and prioritized linear tracking can elicit emergent strategies that strategically choose the most effective body orientation for stopping.

D. Combining Reward Engineering with Co-Design

If we can further co-optimize wheel orientation together with control, we get the design shown in Fig. 9. This design has a self-aligning behavior illustrated in Fig. 5.

It learns to align its backward direction with the command linear velocity direction, without the need for manually commanding angular velocity targets. This helps reduce the average CoT by 14.6% compared with 1D co-design. To study why this happens, we fix this design and retrain the control policy using body frame linear velocity tracking to see how energy efficiency varies among all directions. As

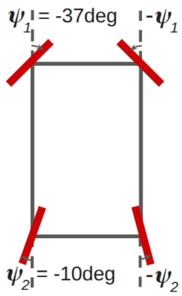

Fig. 9: 2D Co-Design Optimal Design

shown in Fig. 10, although the 2D co-optimized design has worse energy efficiency when skating sideways, it has the best energy efficiency in the backward direction, which is the direction it learns to align with the command direction. This indicates that if the reward engineering can encourage the control policy to strategically pick the best body orientation, the design can focus on improvements in one direction instead of considering all directions, which combined with co-design yields the most efficient skating behavior in this work.

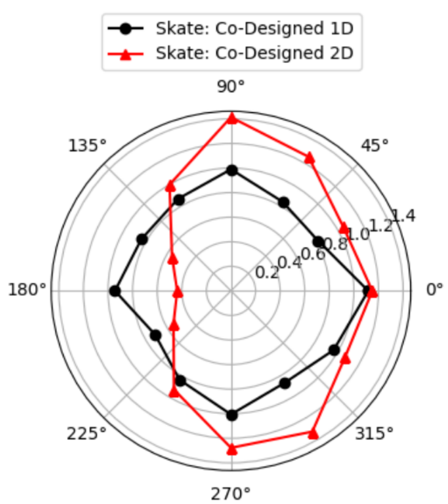

Fig. 10: Directional Analysis of 2D Co-Design Optimal Design when Retrained using Base Frame Command

VI. Conclusion and Future Work

We presented quadrupedal skating with passive wheels as a new locomotion mode that merges the efficiency of wheeled motion with the agility of legged robots. Through a bilevel co-design framework combining Bayesian Optimization and Reinforcement Learning, we identified hardware–policy pairs that enable efficient skating and emergent behaviors such as hockey stops and self-alignment.

Future work will extend this study beyond flat terrain to outdoor and uneven environments, explore lightweight and durable wheel modules, synthesize better reward engineering, and develop more sample-efficient co-design methods. Integrating skating with walking and running modes is another exciting direction toward versatile and multimodal robots.

References

[1] M. Raibert, K. Blankespoor, G. Nelson, and R. Playter, "Bigdog, the rough-terrain quadruped robot," in *IFAC Proceedings Volumes*, vol. 41, no. 2. Elsevier, 2008, pp. 10 822–10 825.

[2] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, p. eaau5872, 2019.

[3] 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 *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2018, pp. 1–9.

[4] S. Kim, J. E. Clark, and M. R. Cutkosky, "isprawl: Design and tuning for high-speed autonomous open-loop running," *Int. J. Robotics Research*, vol. 25, no. 9, pp. 903–912, 2006.

[5] D. Wettergreen, C. Thorpe, and W. L. Whittaker, "Exploring mount erebus by walking robot," *Robotics and Autonomous Systems*, vol. 11, no. 3, pp. 171–185, 1993.

[6] D. Bellicoso, F. Jenelten, C. Gehring, and M. Hutter, "Dynamic locomotion through online nonlinear motion optimization for wheeled-legged robots," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2018, pp. 1–7.

[7] M. Bjelonic, D. Bellicoso, C. Gehring, F. Jenelten, and M. Hutter, "Keep rollin'—whole-body motion control and planning for wheeled quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 3061–3068, 2017.

[8] A. Tenstad, "Deep evolvable-substrate hyperneat-extending es-hyperneat with multiple substrates in an evolving topology," Master's thesis, NTNU, 2020.

[9] E. Brochu, V. M. Cora, and N. De Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *arXiv preprint arXiv:1012.2599*, 2010.

[10] J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke, "Sim-to-real: Learning agile locomotion for quadruped robots," in *Proc. Robotics: Science and Systems (RSS)*, 2018.

[11] X. B. Peng, G. Berseth, K. Yin, and M. van de Panne, "Deeploco: Dynamic locomotion skills using hierarchical deep reinforcement learning," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 41:1–41:13, 2017.

[12] S. Seok, A. Wang, M. Y. Chuah, D. Otten, J. Lang, and S. Kim, "Design principles for highly efficient quadrupeds and implementation on the mit cheetah robot," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 3307–3312.

[13] L. Bao, J. Humphreys, T. Peng, and C. Zhou, "Deep reinforcement learning for bipedal locomotion: A brief survey," *arXiv preprint arXiv:2404.17070*, 2024.

[14] 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*, vol. 44, no. 8, pp. 1396–1427, 2025.

[15] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch *et al.*, "Anymal-a highly mobile and dynamic quadrupedal robot," in *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2016, pp. 38–44.

[16] L. Cui, S. Wang, J. Zhang, D. Zhang, J. Lai, Y. Zheng, Z. Zhang, and Z.-P. Jiang, "Learning-based balance control of wheel-legged robots," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7667–7674, 2021.

[17] R. Li, H. Wang, Q. Li, Z. Han, Y. Chu, L. Ye, W. Xie, and W. Liao, "Ctbc: Contact-triggered blind climbing for wheeled bipedal robots with instruction learning and reinforcement learning," *arXiv preprint arXiv:2509.02986*, 2025.

[18] K. G. Gim and J. Kim, "Ringbot: Monocycle robot with legs," *IEEE Transactions on Robotics*, vol. 40, pp. 1890–1905, 2024.

[19] M. Bjelonic, V. Klemm, J. Lee, and M. Hutter, "A survey of wheeled-legged robots," in *Climbing and walking robots conference*. Springer, 2022, pp. 83–94.

[20] M. Bjelonic, D. C. Bellicoso, M. E. Tiryaki, and M. Hutter, "Skating with a force controlled quadrupedal robot," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7603–7609.

[21] Z. Wei, J. Ren, J. Guo, Y. Yang, S. Xiang, D. Chen, J. Liu, and A. Song, "Slidbot: A quadruped robot with passive wheels for roller skating," *Journal of Bionic Engineering*, 2025.

[22] J. Chen, K. Xu, X. Ding *et al.*, "Roller-skating of mammalian quadrupedal robot with passive wheels inspired by human motion," *Journal of Bionic Engineering*, vol. 17, no. 5, pp. 981–993, 2020.

[23] H. Liu, S. Teng, B. Liu, W. Zhang, and M. Ghaffari, "Discrete-time hybrid automata learning: Legged locomotion meets skateboarding," *arXiv preprint arXiv:2503.01842*, 2025.

[24] S. W. Jo, T. Takahashi *et al.*, "Motion generation for a humanoid robot with inline skates," in *International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, 2010, pp. 188–195.

[25] G. Feng, H. Zhang, Z. Li, X. B. Peng, B. Basireddy, L. Yue, Z. Song, L. Yang, Y. Liu, K. Sreenath *et al.*, "Genloco: Generalized locomotion controllers for quadrupedal robots," in *Conference on Robot Learning*. PMLR, 2023, pp. 1893–1903.

[26] J. Won and J. Lee, "Learning body shape variation in physics-based characters," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–12, 2019.

[27] F. Bjelonic, J. Lee, P. Arm, D. Sako, D. Tateo, J. Peters, and M. Hutter, "Learning-based design and control for quadrupedal robots with parallel-elastic actuators," *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1611–1618, 2023.

[28] C. Schaff, D. Yunis, A. Chakrabarti, and M. R. Walter, "Jointly learning to construct and control agents using deep reinforcement learning," in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 9798–9805.

[29] D. Ha, "Reinforcement learning for improving agent design," *Artificial life*, vol. 25, no. 4, pp. 352–365, 2019.

[30] C. Chen, P. Xiang, J. Zhang, R. Xiong, Y. Wang, and H. Lu, "Deep reinforcement learning based co-optimization of morphology and gait for small-scale legged robot," *IEEE/ASME Transactions on Mechatronics*, vol. 29, no. 4, pp. 2697–2708, 2023.

[31] Y. Kim, Z. Pan, and K. Hauser, "Mo-bbo: Multi-objective bilevel bayesian optimization for robot and behavior co-design," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 9877–9883.

[32] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2015.

[33] A. Akamisaka and K. Nagao, "Transformable quadruped wheelchairs capable of autonomous stair ascent and descent," *Sensors*, vol. 24, no. 11, p. 3675, 2024.