2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Adaptive Motion Priors with Constrained Optimization

Tuchapong Sangthatworn and Bawornsak Sakulkueakulsuk

Abstract— **Choosing locomotion learning paradigm in high-DOF system like humanoid robot faces several challenges. Free exploration creates complex reward surfaces that resist efficient exploration, while human motion priors cannot be directly copied due to different mechanical constraints. We present Adaptive Motion Priors with Constrained Optimization (AMPCO), a novel framework that transitions from human reference motions to task-focused optimization within learned behavioral bounds. AMPCO employs a two-phase optimization strategy: (1) Adaptive Imitation Guidance that prioritizes human motion, and (2) Adaptive Reward Weighting for Constrained Optimization that optimizes task objectives while maintaining motion quality within statistically-guaranteed bounds from Phase I. The transition between phases is automatically detected through percentile-based breakout detection from discriminator convergence. AMPCO introduces adaptive weighting mechanisms that smoothly adjust the importance of human imitation based on learning progress. Our experiments on the Unitree G1 humanoid robot simulation demonstrate that AMPCO reduces energy consumption variance by 67-90% across all baseline methods while achieving 70% lower energy consumption than task-focused baseline while maintaining velocity tracking accuracy comparable to the best-performing methods, with minimal computational overhead (<0.012% per training cycle).**

I. INTRODUCTION

Humanoid robots present a fundamental challenge: their high-DOF systems create complex reward surfaces that are difficult to explore efficiently [1], [2]. This complexity manifests in reward function design, where balancing primary task objectives (velocity tracking, energy efficiency) with auxiliary rewards (motion smoothness, balance) creates an inherent trade-off—auxiliary rewards that guide "how" to behave often compromise achieving real objective [3], [4]. Recent work demonstrates that auxiliary rewards require significant engineering effort and cannot adapt to the robot's evolving capabilities [3], while improperly weighted shaping rewards lead agents to optimize for the shaped rewards rather than the true rewards [5]. This choice becomes particularly critical in high-DOF systems, where the vast action-state space makes efficient exploration a fundamental challenge.

The exploration challenge in humanoid robots stems from their vast action-state spaces. While curiosity-driven methods [6], [7] achieve 2-3x efficiency gains in quadrupeds, bipedal systems require more structured guidance. Human motion provides valuable prior knowledge, with frameworks like AMP [8] and GMP [9] demonstrating that discriminator-based style rewards can guide exploration toward energy-efficient gaits evolved through millions of years of human evolution. However, these approaches face a critical limitation: robots have different mechanical constraints, joint limits, and actuator dynamics than biological systems.

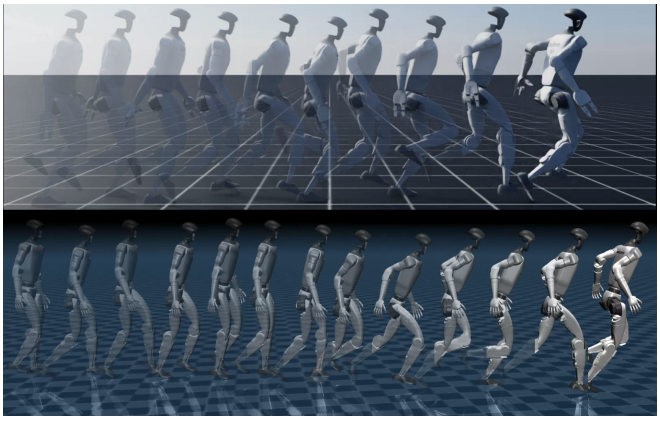

Fig. 1. AMPCO policy transfer from IsaacLab (top) to MuJoCo (bottom) demonstrating gait transition from walking (1.0 m/s) to running (2.0 m/s). Velocity command increases at frame midpoint. Both simulators exhibit similar locomotion patterns, validating successful cross-platform transfer.

Traditional approaches to balancing imitation and task objectives employ fixed weights throughout training, forcing practitioners into suboptimal compromises. While sophisticated adaptive methods exist—ORSO [10] for continuous control, Reward Training Wheels [3] for navigation—they require additional neural networks and complex optimization mechanisms. The recent shift toward constraint-based formulations [5] dramatically reduces this complexity, replacing 10+ reward terms with 3 primary objectives and 11 constraints, yet still requires manual constraint specification.

Dynamic reward scheduling emerges as a middle ground. Beyond simple weighted sums, approaches like the Reward Fusion Module [11] establish task hierarchies, while Versatile Motion Prior [12] adapts style weights based on discriminator performance. However, these methods lack principled mechanisms to detect phase transitions or maintain behavioral during aggressive optimization.

The transition from reference-based to reference-free learning represents a critical capability for discovering robot-specific optimal behaviors. Imitation-relaxation frameworks [13] achieve beyond-human performance (5.0 m/s on MIT-MiniCheetah) through discrete stage transitions, while Adaptive Mimic [14] employs continuous performance-based weighting. These approaches demonstrate that robots can surpass reference limitations, but require either predetermined transition schedules or lack explicit behavioral constraints.

Tuchapong Sangthaworn and Bawornsak Sakulkueakulsuk are at the Institute of Field Robotics (FIBO), King Mongkut's University of Technology Thonburi (KMUTT), Bang Mod, Thung Khru, Bangkok, Thailand tuchapong.sangt@kmutt.ac.th, bawornsak.sak@kmutt.ac.th

While existing methods address the transition from reference-based to task-focused learning, they rely on either static weights, predetermined schedules, or performance metrics. Fixed-weight approaches maintain constant weights throughout training, forcing them to pre-select a single point on the naturalness-performance trade-off. Schedule-based methods like SRW [15] use time-based transitions that change weights according to training episodes rather than learning progress. Performance-based methods like Adaptive Mimic [14] modulate weights based on capability improvements but lack principled mechanisms for detecting learning plateaus or ensuring behavioral bounds during optimization. This leads to either premature transitions that underutilized reference demonstrations or delayed transitions that waste computation on converged objectives. This gap motivates the need for principled convergence detection that indicates when policies have extracted maximum value from limited demonstrations before transitioning to task optimization.

AMPCO addresses the interconnected challenges in bipedal locomotion through a unified framework. Rather than manually engineering rewards to navigate complex reward surfaces [2], we leverage **Adaptive Imitation Guidance (Phase I)** with discriminator-based adaptive weights for guided exploration. This phase constrains exploration to human-like motions, avoiding sample inefficiency in high-DOF spaces. When the discriminator can no longer distinguish between policy and reference behaviors, indicating the policy has learned all available motion patterns, we automatically detect this convergence through **percentile-based breakout detection**, ensuring complete reference utilization without prior knowledge of the convergence value. During **Adaptive Reward Weighting for Constrained Optimization (Phase II)**, we compute constraints automatically derived from the Phase I convergence distribution, allowing the policy to optimize task objectives while maintaining behaviors. [8], [9], [14], [15], our combination eliminated manual tuning effort bridges reference-based stability with reference-free flexibility, transforming high-DOF humanoid learning systems, where high variance across seeds into a reliable and repeatable procedure.

The main contributions of this paper are:

- Adaptive imitation guidance (Phase I) that prioritizes human-like motion acquisition in early training phases, preventing catastrophic failures and unstable behaviors by ensuring that policies develop foundational locomotion skills before pursuing task-specific objectives.
- Automated phase detection through discriminator convergence analysis, eliminating manual design while ensuring optimal transition timing based on actual learning progress rather than predetermined schedules.
- Automatic behavioral reward bound constructed via Chebyshev's inequality from Phase I imitation reward distribution, resolving ROGER's [16] manually specified reward boundaries, which is impractical for adversarial motion priors where discriminator reward convergence value are unknown and shift throughout training.
- Adaptive Reward Weighting for Constrained Optimization (Phase II) that bounds discriminator reward within automatic behavioral reward bound along task optimization process, allowing the policy to push toward optimal task performance without breaking energy-efficient and stable locomotion, achieving what neither pure imitation nor pure task optimization can reach alone.

The remainder of this paper is organized as follows: **Section II** presents the AMPCO methodology, detailing our two-phase optimization strategy with automatic transition detection. **Section III** describes the experimental setup, including the simulation environments, domain randomization parameters, and evaluation metrics. **Section IV** analyzes results across comparison studies, ablation experiments, and sim-to-sim validation. **Section V** examines limitations and future work, and **Section VI** concludes the paper.

II. METHODOLOGY

AMPCO a two-phase optimization strategy: (1) Adaptive Imitation Guidance (Phase I) for reference-based learning; (2) Percentile-Based Breakout Detection to identify convergence (automatic transition); (3) Adaptive Reward Weighting for Constrained Optimization (Phase II). We first review the base AMP formulation, then detail each component.

A. AMP: Adversarial Motion Priors for Human-Like Behavior Imitation

We adopt Adversarial Motion Priors (AMP) [8] for its key advantage: instead of explicitly tracking reference motions, it uses adversarial learning to incorporate expert demonstrations, relaxing motion constraints while enabling generalization beyond demonstrated behaviors while preserving human-like motion characteristics.

The AMP framework consists of two key components: a task-specific reward r_t^G and a style reward r_t^S.

$$r_t = w_t^G r_t^G + w_t^S r_t^S \tag{1}$$

$$w_t^G = 1 - w_t^S \tag{2}$$

Where w_t^G and w_t^S are the weight coefficient that balances the importance of the two reward components.

The AMP framework employs a discriminator network $D(o_t, o_{t+1})$ distinguishes between state transitions from expert demonstrations (+1) and policy-generated behaviors (-1).

Expert Loss: Trains the discriminator to output +1 for expert transitions from reference motion data.

$$L_{expert} = \mathbb{E}_{d^M(o_t,o_{t+1})}[(D(o_t,o_{t+1})-1)^2] \tag{3}$$

Policy Loss: Trains the discriminator to output -1 for transitions generated by the learning policy.

$$L_{policy} = \mathbb{E}_{d^\pi(o_t,o_{t+1})}[(D(o_t,o_{t+1})+1)^2] \tag{4}$$

Gradient Penalty: Regularizes discriminator's gradients to ensure smooth and improve training stability.

$$L_{GP} = \mathbb{E}_{d^M(o_t,o_{t+1})}[||\nabla D(o_t,o_{t+1})||^2] \tag{5}$$

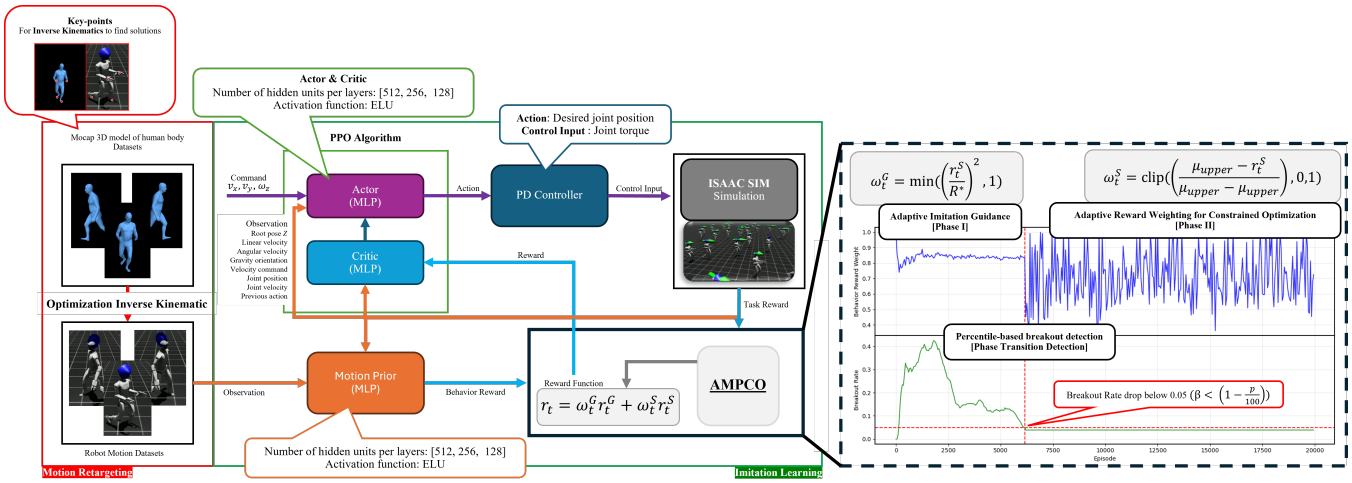

Fig. 2. AMPCO framework overview. Left (red box): Motion retargeting converts human motion capture to robot datasets. Center: Imitation learning (green box) using AMP framework with PPO and Motion Prior (orange box) to generate PD controller actions. Right (dashed box): Our contributions: (1) Adaptive Imitation Guidance (Phase I), (2) Percentile-based breakout detection for phase transition, (3) Adaptive Reward Weighting for Constrained Optimization (Phase II). The framework adaptively balances task and behavior rewards, initially focusing on imitation before transitioning to task optimization within behavioral constraints.

Combined AMP Loss:

$$L_{AMP} = \frac{1}{2}L_{expert} + \frac{1}{2}L_{policy} + \lambda_{GP}L_{GP} \tag{6}$$

Where $d^M(o_t, o_{t+1})$ and $d^\pi(o_t, o_{t+1})$ represent the distributions over state transitions from the expert demonstrations and policy, respectively.

The discriminator score is transformed into a reward signal that encourages expert-like behavior:

$$r_t^S = \max\left[0, 1 - 0.25(D(o_t, o_{t+1}) - 1)^2\right] \tag{7}$$

This function maps discriminator outputs to the range [0, 1], providing higher rewards when transitions resemble expert behavior.

B. AMPCO: Adaptive Motion Priors with Constrained Optimization

Adaptive Imitation Guidance (Phase I): We propose conditioning w_t^G on discriminator reward r_t^S, inverting the task-reward focus of Adaptive Mimic [14], ensuring task weight grows only as the policy genuinely deceives the discriminator rather than as task performance incidentally improves. The rationale is twofold: purely fixing $w^S = 1$ is insufficient, as locomotion requires a small degree of task guidance to direct the robot toward meaningful movement during early exploration. However, introducing too much task pressure too early causes the robot to prioritize reaching target velocities through any available strategy before stable motion patterns are established. By growing w_t^G with the robot's current motion quality, our method ensures that energy-efficient and stable motion is established during early exploration, before transition to task optimization phase. Initially, we focus on the discriminator reward r^S as the primary optimization criterion by setting $w^S = 1,\ w^G = 0$,

$$\max r_t^G = \max r_t^S = R^* \tag{8}$$

$$w_t^G = min((\frac{r_t^S}{R^*})^2, 1) \tag{9}$$

where R^* represents the current expected maximum reward value (either task or discriminator), ensuring the computed weights remain normalized within [0,1]. As the style reward r_t^S approaches R^*, the task weight w_t^G gradually increases, triggering a smooth transition from imitation-focused to task-focused optimization. As the policy converges toward expert behavior, the discriminator rewards eventually converge when the policy can no longer deceive the discriminator. We then use discriminator rewards convergence behavior to design automatic Percentile-based breakout detection.

Percentile-based breakout detection (Phase Transition Detection): To detect convergence without prior knowledge of the convergence value, fixed thresholds therefore fail to generalize. We propose a percentile-based breakout detection on imitation reward method adapted from statistical process control principles [17]. We test whether new rewards continue to exceed the historical performance distribution. We compute the p-th percentile $P_t = \text{Percentile}_p(\mathscr{B}_{t-1})$ and define the breakout indicator b_t and breakout rate β_t:

$$b_t = \mathbb{1}[\bar{r}_t^S > P_t] \tag{10}$$

$$\beta_t = \frac{1}{W}\sum_{i=t-W+1}^{t} b_i \tag{11}$$

where $\bar{r}_t^S$ is the mean discriminator reward per step, $\mathscr{B}_t = \{\bar{r}_{t-W+1}^S, ..., \bar{r}_t^S\}$ is a sliding window of size W, and b_t indicates whether the current reward exceeds the historical performance boundary. The breakout rate β_t measures how frequently recent rewards continue to exceed their own history. Convergence is detected when $\beta_t < (1 - p/100)$, indicating the reward distribution has stopped improving relative to its own history.

Algorithm 1: Percentile-based breakout detection

input: $r_t^S,\ \sigma^S$
$P_t \leftarrow Percentile_p(\mathscr{B}_{t-1})$
$b_t \leftarrow$ calculate break out indicator according to Equation 10 using $r_t^S,\ P_t$
$\beta_t \leftarrow$ calculate break out indicator according to Equation 11 using b_t
$\mathscr{B}_t$.append(r_t^S)
if $\beta_t < (1-p/100)$ *and* $|\mathscr{B}_t| = W$ **then**
  $\mu_{\text{upper}},\ \mu_{\text{lower}} \leftarrow$ calculate boundary according to Equation 13 using $r_t^S,\ \sigma^S$
  return true, $\mu_{\text{upper}}, \mu_{\text{lower}}$
end
return false, 0, 0

Adaptive Reward Weighting for Constrained Optimization [16] (Phase II):

Although AMP is an effective framework for style-guided policy learning, maintaining balance between style and task rewards remains a challenge, particularly as the agent's capabilities evolve throughout training. To address this challenge, we incorporate the Reward Online Gain Estimation in Reinforcement Learning [16] that offers an online adaptive reward-weighting mechanism based on penalty thresholds.

Building on ROGER's principles, our framework dynamically adjusts reward weights based on performance relative to behavioral constraints. This ensures task-focused optimization without deviating from human-like motion boundaries.

$$w_t^S = \text{clip}\left(\frac{\mu_{\text{upper}} - r_t^S}{\mu_{\text{upper}} - \mu_{\text{lower}}}, 0, 1\right),\ w_t^G = 1 - w_t^S \tag{12}$$

Where r_t^S is discriminator mean reward, $R_{threshold}$ discriminator reward threshold and $R_{boundary}$ discriminator boundary.

Since the discriminator is a learned generative model with dynamic and unknown reward boundaries, we cannot intuitively predefine appropriate constraint ranges for ROGER. To address this, we employ Chebyshev's inequality to construct data-driven behavioral constraint boundaries from the convergence distribution. Given the mean μ_{sat} and standard deviation σ_{sat} of rewards at convergence:

$$\mu_{\text{upper}} = \mu_{\text{sat}} + k\sigma_{\text{sat}}, \quad \mu_{\text{lower}} = \mu_{\text{sat}} - k\sigma_{\text{sat}} \tag{13}$$

where $k = \sqrt{1/(1-\alpha)}$ guaranties that at least α proportion of rewards fall within these bounds. For $\alpha = 0.95$, we obtain $k \approx 4.47$, providing reward bound that allow the policy to deviate from reference motions when necessary for task optimization while maintaining acceptable motion quality.

Our approach leverages AMP's diverse motion generation capabilities while incorporating ROGER's constraint-based adaptation, creating a framework that maintains behavioral consistency throughout task optimization.

AMPCO introduces minimal computational overhead to the standard AMP pipeline. We measured the additional processing time for each component across our 24-timestep episodes: Phase I adaptive weighting requires 2.22±0.22 μs per timestep (53.28 μs per episode), Phase II constrained optimization requires 3.20±0.75 μs per timestep (76.8 μs per episode), and percentile-based breakout detection requires 166.95±27.95 μs once per episode. The total overhead is therefore approximately 220 μs during Phase I and 244 μs during Phase II. Compared to the standard training cycle of 2.1 seconds (1.45s data collection + 0.65s learning updates), AMPCO adds less than 0.012% computational overhead. This negligible cost stems from our design choices: (1) simple arithmetic operations for adaptive weighting executed in parallel across environments, (2) maintaining only a sliding window of 1000 scalar values for percentile detection, and (3) no additional neural networks or gradient computations.

III. EXPERIMENT DETAILS

A. Training and Implementation Details

The implementation uses IsaacLab 4.2.0, training 4096 parallel environments at 200Hz simulation frequency with policy queries at 50Hz. The Unitree G1 humanoid robot features 23 DOF: 1 root/pelvis, 5 per leg, and 6 per arm. Commands $c_t = (v_x^*,\ v_y^*,\ w_{yaw}^*)$ specify desired velocities, while actions $a_t \in \mathbb{R}^{23}$ represent target joint positions for PD controllers on actuated joints.

Three observation models are employed: Actor/Critic models use $o_t \in \mathbb{R}^{82}$ including root pose (z-position), linear, angular velocity, gravity orientation $o_t^{root} \in \mathbb{R}^{10}$, velocity commands $o_t^{command} \in \mathbb{R}^3$, joint position, velocity $o_t^{joint} \in \mathbb{R}^{46}$, and previous actions $o_t^{a_{t-1}} \in \mathbb{R}^{23}$. The discriminator uses $o_t \in \mathbb{R}^{53}$ containing root pose (z-position), linear, angular velocity $o_t^{root} \in \mathbb{R}^7$, and joint position, velocity $o_t^{joint} \in \mathbb{R}^{46}$. Motion datasets are sourced from the AMASS Dataset [18].

Policies are trained using PPO-Clip over 20,000 episodes with 24-timestep trajectories. Training completes in 12-14 hours on Intel i5-14400F and NVIDIA RTX 4070 Ti SUPER.

B. Sim-to-Sim cross validation

We validate AMPCO's generalization capability by transferring policies trained in IsaacLab to MuJoCo simulation. To ensure robustness, we apply domain randomization including mass variations, sensor noise, and external disturbances, with parameters detailed in Table I.

TABLE I
DOMAIN RANDOMIZATION RANGES

Randomization Item	Range	Unit
Mass variation	$[-0.5, 0.5]$	kg
Friction coefficient	$[0.5, 1.0]$	(unitless)
Center of mass offset	$[-0.1, 0.1]$	m
Impulse (x, y)	$[0, 0.5]$	m/s
Linear velocity noise	$[-0.1, 0.1]$	m/s
Angular velocity noise	$[-0.2, 0.2]$	rad/s
Projected gravity noise	$[-0.05, 0.05]$	m/s^2
Joint position noise	$[-0.01, 0.01]$	rad
Joint velocity noise	$[-1.5, 1.5]$	rad/s

C. Hyperparameter Selection Guide

All RL hyperparameters follow the original AMP implementation [8]. AMPCO approach uses parameters from Table II with a modified task reward function:

$$r_t^G = \exp(-0.25||v^* - v_t^{base}||^2) \cdot r_{upright} \tag{14}$$

where $r_{uprigt} \in [0,1]$ represents the projected gravity vector along the robot's z-axis, ensuring stable upright posture an essential foundation for locomotion tasks as demonstrated in [19]. The velocity terms v^* and v_t^{base} denote the commanded velocity and actual robot base velocity in the local hip coordinate frame, respectively, as 2-D velocity vectors.

AMPCO sensitivity experiments varying $\alpha \in \{0.90, 0.95, 0.99\}$ and p $\in \{0.90, 0.95, 0.99\}$ over N=10 seeds confirm that all achieve statistically equivalent energy consumption (p>0.05), with total variation across settings remaining smaller than variance of any single baseline. Detailed results are available in the supplementary video.

TABLE II
AMPCO PARAMETERS

Parameter	Symbol	Value/Range
Percentile-based breakout detection Parameters		
Sliding window size	W	1000
p-th percentile	p	0.95
Adaptive Reward Weighting for Constrained Optimization Parameters		
Chebyshev confidence level	α	0.95

D. Comparison Study

We evaluate our method in locomotion task, we conduct a comparative analysis against three representative baselines.

- **AMP [8]**: Fixed-weight baseline with constant style weights $w_t^S \in \{0.5, 0.7, 0.8\}$, selected from our prior work evaluation across six weight values [15] to represent task-focused ($w = 0.5$, best tracking), balanced ($w = 0.7$, best multi-objective trade-off), and imitation-focused ($w = 0.8$, lowest energy) operating regimes.
- **AdaMimic [14]**: An adaptive method that dynamically balances imitation and task rewards, originally demonstrated on the DeepMimic [20] formulation, adapted here for adversarial motion priors.
- **AMP [8] + ROGER [16]**: Combines AMP with ROGER, which adaptively adjusts weights during training within constraints optimization.

E. Ablation Study

To validate our design choices, we conducted ablation studies removing each major component. Table IV reveals how each contributes to the final performance, with striking differences in both efficiency and stability.

- **Our w/o AIG**: We excludes **adaptive imitation guidance** as shown in Table IV, the absence of the adaptive imitation guidance lead to unstable and energy-inefficient locomotion patterns, highlighting the critical role of imitation-based exploration in discovering stable, low-energy gaits during early training phases.
- **Our w/o ROGER**: We excludes **adaptive reward weighting for constrained optimization [16]** as shown in Table IV, the absence of the adaptive reward weighting for constrained optimization lead to an inability to balance competing objectives, resulting in overly conservative policies that prioritize energy efficiency and motion quality at the expense of task performance.

F. Metrics for Evaluation

Evaluation considers three metrics:

- **Velocity tracking**: Root-mean-square error between commanded and measured planar velocities (m/s).
- **Energy consumption**: Discrete integration of positive mechanical work (J) with timestep dt=0.02s, computed as $E = \sum_t \sum_i \max(0, \tau_{i,t} \cdot \dot{q}_{i,t}) \cdot dt$
- **Discriminator reward**: Mean style reward from the adversarial discriminator, ranging from 0 to 1.

Each metric is computed over 1,000-timestep windows. Final values represent the mean over the last 1,000 timesteps of training, while peak values indicate worst-case performance after timestep 1,000. Convergence occurs when the 100-step moving average remains within $\pm$10% of the final mean. Statistical comparisons employ wilcoxon signed rank test with exact p-values (N=10 seeds, α=0.05).

IV. RESULTS & ANALYSIS

A. Comparison Study

Figure 3 illustrates the training dynamics over 20,000 episodes across all comparison methods. The trajectories reveal distinct learning patterns: task-focused baselines (AMP [w=0.5] and AdaMimic) exhibit unstable exploration with energy consumption reaching peaks of 44 kJ, while AMPCO demonstrates smooth convergence with the lowest peak energy (17.7 kJ). The discriminator reward curves show AMPCO achieving the highest final values (0.293), indicating superior motion quality compared to all baselines.

While fixed-weight methods show performance throughout training, AMPCO's phase transition (visible around episode 2,500) marks a clear shift from Phase I to Phase II, resulting in consistent improvement across metrics.

Figure 4 presents final performance distributions after complete training, revealing the consistency of each method across 10 seeds. The violin plots demonstrate AMPCO's superior reliability: contrasting sharply with variability of task-focused baselines. Table III quantifies these observations, confirming statistical significance (p<0.05) for AMPCO's improvements in energy efficiency and motion quality.

Velocity Tracking: Final RMSE of 0.288 m/s, statistically equivalent to AdaMimic (0.280 m/s, p=0.77) while consuming 69% less energy. Compared to energy-efficient baselines (AMP w=0.7, 0.8), we achieve significantly better tracking (p<0.05) with comparable energy usage.

Discriminator Reward: The highest final reward of 0.293, outperforming all baselines (p<0.05). Notably, AdaMimic struggles with AMP's discriminator-based rewards (0.008), suggesting poor generalization from its original DeepMimic [20] formulation to adversarial frameworks.

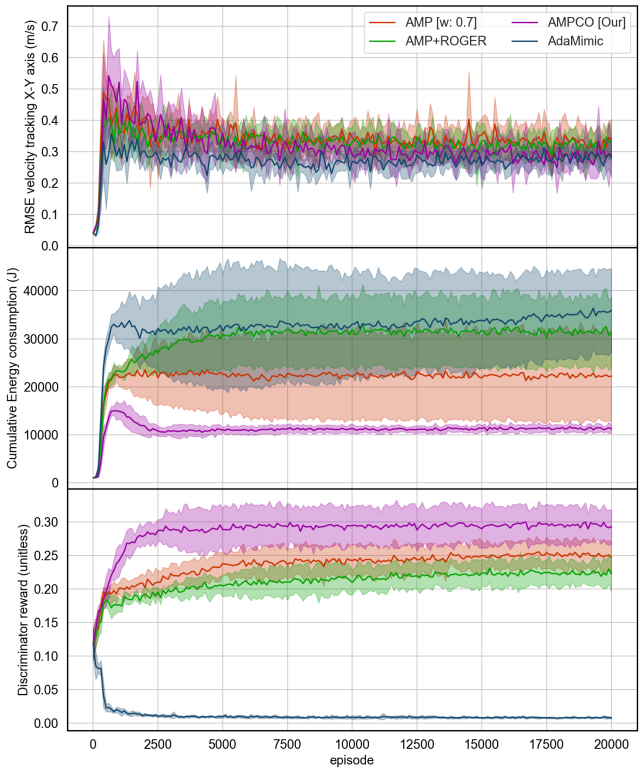

Fig. 3. Training dynamics over 20,000 episodes comparing AMPCO with baseline methods. From top to bottom: RMSE velocity tracking error, cumulative energy consumption, and discriminator reward. Methods shown: AMP [w=0.7] (red), AMP+ROGER (green), AdaMimic (blue), and AMPCO (purple). Lines indicate means with shaded regions showing standard deviation across 10 random seeds.

Consistency: AMPCO demonstrates the lowest variance across all metrics, with coefficient of variation below 7.2% for energy and 6% for tracking error, indicating robust performance across random seeds.

AMPCO leverages human motion priors to guide initial exploration, preventing the random, energy-wasteful movements seen in task-focused methods. The discriminator-guided exploration naturally discovers efficient gaits similar to human walking, resulting in lower energy consumption while maintaining comparable velocity tracking. This structured exploration through motion priors proves essential for achieving the 70% energy reduction across our experiments.

B. Ablation Study

Table IV reveals contributions of AMPCO components.

Impact of removing AIG (Our w/o AIG): This variant exhibits catastrophic multi-objective failure. While achieving comparable tracking error to AMPCO (0.313 vs 0.288 m/s, p=0.2754), it consumes 154% more energy (28.7k vs 11.3k J, p<0.01) with 239% higher variance (CV=24.4% vs 7.2%). The absence of Adaptive Imitation Guidance leads to unstable, energy-inefficient locomotion patterns, confirming its crucial role in discovering stable gaits during early training.

Impact of removing ROGER (Our w/o ROGER): This variant achieves the lowest energy consumption (9.8 kJ) and highest discriminator reward (0.318), but suffers from 25% worse tracking performance (0.361 vs 0.288 m/s, p<0.05). Without Adaptive Reward Weighting for Constrained Optimization, the policy prioritizing energy efficiency at the expense of task performance.

Convergence analysis: AMPCO demonstrates the fastest discriminator convergence (2.1k episodes) compared to ablation variants (3.1k for w/o AIG, 4.4k for w/o ROGER), confirming that the synergistic combination of AIG and ROGER accelerates learning. This 32-52% faster convergence validates that both components work together to balance exploration and exploitation effectively.

The failures of each ablation variant validate AMPCO's design principle: phase-based optimization that transitions from stable exploration to constrained task optimization.

C. Sim-to-Sim cross validation

Figure 5 results confirm AMPCO's superior multi-objective optimization: Our method achieves the lowest energy consumption (Command 2 m/s: 126.61 J/step). When evaluating normalized velocity tracking error (percentage deviation from commanded velocities), AMPCO demonstrates competitive performance with approximately 15% mean normalized error, comparable to AMP+ROGER (17%) and AMP [w=0.7] (19%), while significantly outperforming AdaMimic (38%). Notably, AMPCO achieves 49% lower energy consumption than fixed-weight AMP [w=0.7], 64% lower than AMP+ROGER, and 69% lower than AdaMimic, while maintaining comparable or superior tracking accuracy. This validates our key contribution that AMPCO successfully balance competing objectives achieving energy efficiency without sacrificing tracking performance, unlike fixed-weight approaches that must compromise between these objectives.

V. LIMITATIONS AND FUTURE WORK

Our current implementation has several limitations. that suggest for improvement. Most notably, AMPCO employs conservative Chebyshev confidence level (α=0.95) that, while ensuring reliable convergence across all random seeds, potentially restrict the solution space. Additionally, our experiment focus on velocity tracking with minimal reward engineering. Though it is effective for validating core principles, AMPCO leaves questions about scalability to complex locomotion tasks such as stair climbing, rough terrain navigation, or dynamic obstacle avoidance.

These limitations motivate a progressive research agenda. With AMPCO's consistent energy-speed optimization providing reliable outcome, future work can safely relax these constraints to discover robot-specific gaits that surpass human demonstrations. This requires adaptive constraint relaxation where reward bounds expand based on gait stability to enable safe exploration of novel behaviors. These would allow robots to discover gaits optimized for their specific morphology. Once these superior behaviors are discovered in simulation, real-world deployment will serve as the final refinement stage, where hardware constraints and physical dynamics fine-tune these policies for practical application.

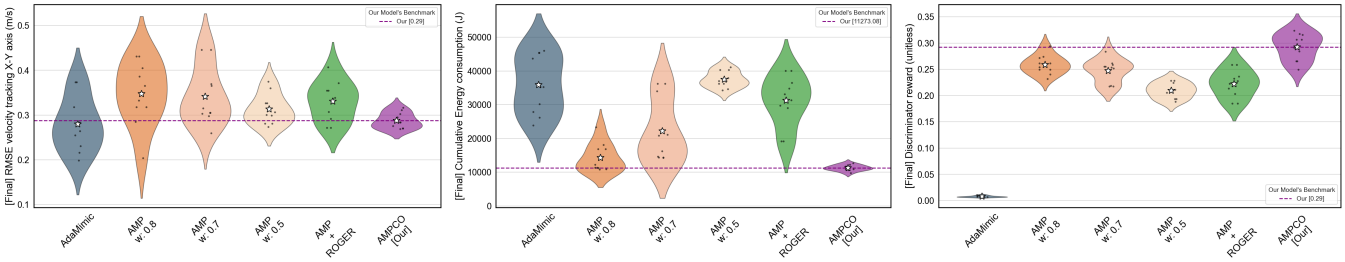

Fig. 4. Final performance distributions after 20,000 episodes. From left to right: RMSE velocity tracking, energy consumption, and discriminator reward. Violin plots show variability across 10 seeds (dots: individual runs, stars: means). Lower is better for tracking and energy; higher is better for discriminator reward. Dashed horizontal lines mark AMPCO's mean performance across each metric for comparison.

TABLE III

COMPARISON STUDY PERFORMANCE EVALUATION (MEAN ± STD) OVER N=10 SEEDS.

	RMSE Velocity Tracking (m/s)		Energy Consumption (J)		Discriminator Reward		Convergence episodes		
Method	Peak↓	Final↓	Peak↓	Final↓	Peak↑	Final↑	RMSE	Energy	Disc
AMP [w = 0.5]	0.700 ± 0.071	0.313 ± 0.032	**44.1k ± 1.8k**	**37.5k ± 2.3k**	0.250 ± 0.012	0.210 ± 0.014	12.9k	**3.2k**	5.2k
AMP [w = 0.7]	0.834 ± 0.062	0.341 ± 0.061	30.8 ± 7.3k	22.2k ± 9.0k	0.277 ± 0.016	0.247 ± 0.021	**3.5k**	5.3k	4.6k
AMP [w = 0.8]	1.155 ± 0.237	**0.347 ± 0.068**	22.0k ± 2.2k	14.3k ± 4.0k	0.289 ± 0.013	0.259 ± 0.017	10.3k	3.5k	**2.1k**
AMP+ROGER	0.818 ± 0.129	0.330 ± 0.042	37.4k ± 6.7k	31.3k ± 7.0k	0.251 ± 0.017	0.222 ± 0.025	9.6k	4.9k	6.8k
AdaMimic	**0.664 ± 0.073**	**0.280 ± 0.061**	43.9k ± 8.3k	35.9k ± 8.2k	0.244 ± 0.135	**0.008 ± 0.001**	9.4k	**12.0k**	**19.5k**
AMPCO [Our]	**1.187 ± 0.098**	0.288 ± 0.017	**17.7k ± 1.7k**	**11.3k ± 0.8k**	**0.323 ± 0.025**	**0.293 ± 0.024**	**14.7k**	3.7k	**2.1k**

↑ Higher is better. ↓ Lower is better. **Green** indicates best value and **Red** indicates worst value. **Bold** indicates extreme values and Non-bold indicates statistical equivalence ($p \geq 0.05$).

TABLE IV

ABLATION STUDY PERFORMANCE EVALUATION (MEAN ± STD) OVER N=10 SEEDS.

	RMSE Velocity Tracking (m/s)		Energy Consumption (J)		Discriminator Reward		Convergence episodes		
Method	Peak↓	Final↓	Peak↓	Final↓	Peak↑	Final↑	RMSE	Energy	Disc
Our w/o AIG	0.833 ± 0.064	0.313 ± 0.056	**34.9k ± 7.1k**	**28.7k ± 7.0k**	**0.239 ± 0.016**	**0.214 ± 0.022**	15.5k	**6.9k**	3.1k
Our w/o ROGER	**1.187 ± 0.098**	**0.361 ± 0.076**	**17.7k ± 1.7k**	**9.8k ± 0.4k**	**0.343 ± 0.023**	**0.318 ± 0.027**	**15.7k**	4.6k	**4.4k**
Our	**1.187 ± 0.098**	**0.288 ± 0.017**	**17.7k ± 1.7k**	11.3k ± 0.8k	0.323 ± 0.025	0.293 ± 0.024	**14.7k**	**3.7k**	**2.1k**

↑ Higher is better. ↓ Lower is better. **Green** indicates best value and **Red** indicates worst value. **Bold** indicates extreme values and Non-bold indicates statistical equivalence ($p \geq 0.05$).

Long-term extensions could explore hierarchical applications where complex whole-body behaviors are acquired through multiple guided exploration phases, ultimately creating a complete pipeline from human-inspired initialization through behavior discovery to real-world deployment.

VI. CONCLUSIONS

We presented AMPCO, a framework that achieves reliable bipedal locomotion through adaptive objective management. Recognizing that different stages of learning require different learning priorities, AMPCO combines the stability of reference-based learning with the flexibility of reference-free learning. This phase-based approach enables policies to develop stable foundations before pursuing performance.

The framework's primary achievement lies in addressing the high variance across random initializations in bipedal locomotion learning. AMPCO reduces energy consumption variance by 67-90% across all baseline methods, transforming bipedal learning from a highly initialization-dependent process into a repeatable procedure where energy and velocity tracking remain within 8% of their mean values regardless of random seed. This reliability comes without sacrificing performance: AMPCO achieves 70% lower energy consumption than task-focused baselines while maintaining velocity tracking accuracy comparable to the best-performing methods. Importantly, these improvements require minimal modifications to existing AMP implementations and add negligible computational overhead (<0.012% per training cycle), making AMPCO immediately accessible to researchers.

With hyperparameter tuning no longer necessary, future research can progressively relax human motion constraints to discover robot-optimal behaviors. AMPCO's framework enables a shift in how researchers approach bipedal learning: instead of manually tuning reward weights, they can now design multi-phase learning curricula that mirror biological development. AMPCO's components provide the foundation to automatically manage such multi-phase transitions, en-

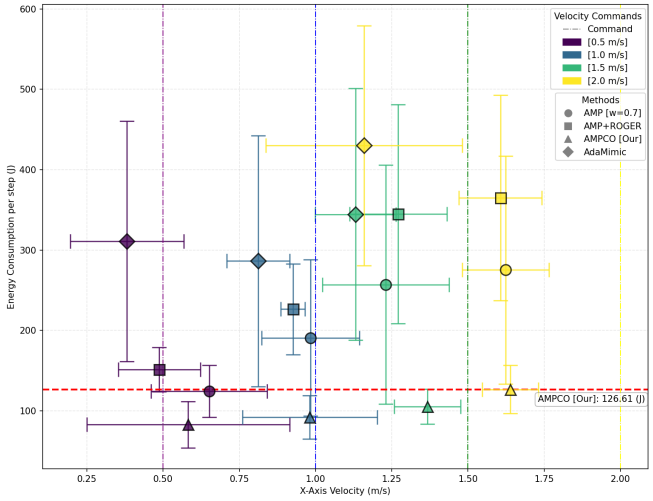

Fig. 5. Cross-simulator validation in MuJoCo showing energy consumption per step across velocity range. Markers indicate: AMPCO (circles), AMP [w=0.7] (squares), AMP+ROGER (triangles), and AdaMimic (diamonds). Colors represent different velocity commands tested (0.5, 1.0, 1.5, 2.0 m/s). Vertical dashed lines mark command velocities; horizontal dashed line shows AMPCO's maximum energy consumption across all velocities.

abling researchers to design developmental learning phases without manual hyperparameter tuning. This shifts the focus from reward weight optimization to defining meaningful skill progressions that mirror natural development.

Supplementary Material

Video demonstrations of the gaits, smooth walk-to-run transitions comparing AMPCO's learning progress against baselines, Sim-to-Sim (Isaac Sim to MuJoCo) evaluation results, and a detailed hyperparameter sensitivity analysis are available at: `https://youtu.be/W4GXNGuupw4`.

Acknowledgment

The authors would like to thanks the financial supports from Institute of Field Robotics (FIBO), King Mongkut's University of Technology Thonburi (KMUTT), Thailand, and the National Science, Research and Innovation Fund (NSRF) (Fundamental Fund).

The authors acknowledge the use of Claude (Anthropic) for editorial assistance in refining academic language, checking redundancy, and improving clarity throughout the manuscript. All scientific content and contributions remain solely the work of the human authors.

References

[1] L. Bao, J. N. Humphreys, T. Peng, and C. Zhou, "Deep reinforcement learning for bipedal locomotion: A brief survey," *ArXiv*, vol. abs/2404.17070, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:269430559

[2] S. Koseki, K. Kutsuzawa, D. Owaki, and M. Hayashibe, "Multimodal bipedal locomotion generation with passive dynamics via deep reinforcement learning," *Frontiers in Neurorobotics*, vol. 16, p. 1054239, 01 2023.

[3] L. Wang, T. Xu, Y. Lu, and X. Xiao, "Reward training wheels: Adaptive auxiliary rewards for robotics reinforcement learning," *ArXiv*, vol. abs/2503.15724, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:276897108

[4] Y. Hu, W. Wang, H. Jia, Y. Wang, Y. Chen, J. Hao, F. Wu, and C. Fan, "Learning to utilize shaping rewards: a new approach of reward shaping," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS '20. Red Hook, NY, USA: Curran Associates Inc., 2020.

[5] Y. Kim, H. S. Oh, J. H. Lee, J. Choi, G. Ji, M. Jung, D. H. Youm, and J. Hwangbo, "Not only rewards but also constraints: Applications on legged robot locomotion," *IEEE Transactions on Robotics*, vol. 40, pp. 2984–3003, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:261100652

[6] C. Lu, L. Shi, Z. Chen, C. Wu, and A. Wierman, "Overcoming the curse of dimensionality in reinforcement learning through approximate factorization," in *Forty-second International Conference on Machine Learning*, 2025. [Online]. Available: https://openreview.net/forum?id=aEsIW59zDm

[7] O. Groth, M. Wulfmeier, G. Vezzani, V. Dasagi, T. Hertweck, R. Hafner, N. M. O. Heess, and M. A. Riedmiller, "Is curiosity all you need? on the utility of emergent behaviours from curious exploration," *ArXiv*, vol. abs/2109.08603, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:237563118

[8] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa, "Amp," *ACM Transactions on Graphics (TOG)*, vol. 40, pp. 1 – 20, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:233033739

[9] H. Zhang, L. Zhang, Z. Chen, L. Chen, Y. Wang, and R. Xiong, "Natural humanoid robot locomotion with generative motion prior," *ArXiv*, vol. abs/2503.09015, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:276937586

[10] C. B. C. Zhang, Z.-W. Hong, A. Pacchiano, and P. Agrawal, "Orso: Accelerating reward design via online reward selection and policy optimization," *ArXiv*, vol. abs/2410.13837, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:273403720

[11] K. Jiang, Z. Fu, J. Guo, W. Zhang, and H. Chen, "Learning whole-body loco-manipulation for omni-directional task space pose tracking with wheeled-quadrupedal-manipulator," *IEEE Robotics and Automation Letters*, 2024.

[12] R. Yang, Z. Chen, J. Ma, C. Zheng, Y. Chen, Q. Nguyen, and X. Wang, "Generalized animal imitator: Agile locomotion with versatile motion prior," in *Conference on Robot Learning*, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:263605512

[13] Y. Jin, X. Liu, Y. Shao, H. Wang, and W. Yang, "High-speed quadrupedal locomotion by imitation-relaxation reinforcement learning," *Nature Machine Intelligence*, vol. 4, pp. 1198–1208, 2022.

[14] C. Zhang, Q. Wu, L. Ma, and H. Su, "Adaptive mimic: Deep reinforcement learning of parameterized bipedal walking from infeasible references," *ArXiv*, vol. abs/2112.03735, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:244920998

[15] T. Sangthaworn and B. Sakulkueakulsuk, "From imitation to task performance: Scheduled reward weighting for energy-efficient bipedal locomotion," in *2025 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2025, pp. 1245–1250.

[16] A. Srisuchinnawong and P. Manoonpong, "Gain Tuning Is Not What You Need: Reward Gain Adaptation for Constrained Locomotion Learning," in *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025.

[17] A. Saghir and Z. T. Kosztyán, "An r package for percentile-based control charts: pbcc," *Software Impacts*, vol. 15, p. 100455, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2665963822001397

[18] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, "AMASS: Archive of motion capture as surface shapes," in *International Conference on Computer Vision*, Oct. 2019, pp. 5442–5451.

[19] Z. Zhuang, D. Shi, R. Suo, X. He, H. Zhang, T. Wang, S. Lyu, and D. Wang, "Tdmpbc: Self-imitative reinforcement learning for humanoid robot control," *ArXiv*, vol. abs/2502.17322, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:276575867

[20] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 143:1–143:14, Jul. 2018. [Online]. Available: http://doi.acm.org/10.1145/3197517.3201311