# Learning Quadrupedal Locomotion with Gait Priors and Trust Region Policy Optimization

Ankit Sinha (Group 1)
MS in Computer Science
Georgia Institute of Technology
Email: asinha389@gatech.edu

***Abstract*—Learning robust locomotion policies for quadrupedal robots remains a fundamental challenge in reinforcement learning and robotics. While model-free deep RL algorithms like Trust Region Policy Optimization (TRPO) offer promise for continuous control tasks, vanilla policy learning often fails to discover natural, symmetric, and periodic gaits characteristic of biological quadrupeds. This paper investigates the application of TRPO to quadrupedal locomotion on the Unitree Go1 robot in MuJoCo simulation. We demonstrate that vanilla TRPO struggles to learn effective locomotion, resulting in unstable and asymmetric gaits. To address this limitation, we propose a residual learning framework that combines a parametric trotting gait prior with a TRPO-learned residual controller. Our approach successfully produces natural, rhythmic, and smooth locomotion while maintaining forward movement and balance. We provide detailed mathematical formulations of TRPO, the gait prior design, and the residual policy architecture. Experimental results validate that incorporating domain-specific structure through gait priors significantly improves learning efficiency and locomotion quality compared to learning from scratch. The code repository can be found here DRL_Project_TRPO. This is the google drive link to my project materials project materials.**

## I. Introduction

Quadrupedal locomotion represents a cornerstone challenge in legged robotics, requiring the coordination of twelve or more degrees of freedom to achieve stable, efficient, and adaptive movement. Recent advances in deep reinforcement learning (RL) have enabled end-to-end learning of complex motor behaviors, with algorithms like Trust Region Policy Optimization (TRPO) [1] and Proximal Policy Optimization (PPO) [2] demonstrating success in continuous control domains. However, applying vanilla model-free RL to quadrupedal locomotion presents significant challenges. The high-dimensional action space, under-specified reward signals, and the need to discover periodic gait patterns make the learning problem exceptionally difficult. Biological quadrupeds exhibit well-structured gaits—walk, trot, pace, and gallop—that emerge from millions of years of evolution and are encoded in both morphology and neural circuitry. In contrast, RL agents learning from scratch must discover these patterns through random exploration, often resulting in unnatural, inefficient, or unstable behaviors.

### A. Motivation and Contributions

This work investigates the following research question: *Can incorporating prior knowledge about natural quadrupedal gaits improve the learning of locomotion controllers using TRPO?* We hypothesize that providing a structured gait prior as a reference trajectory, and learning only residual corrections via TRPO, will lead to more natural and stable locomotion compared to learning policies from scratch.

Our contributions are threefold:

1) We implement and evaluate vanilla TRPO for quadrupedal locomotion on the Unitree Go1 robot, documenting its limitations in discovering natural gaits.
2) We design a parametric trotting gait prior based on rhythmic sinusoidal trajectories that capture the essential coordination patterns of quadrupedal trotting.
3) We propose a residual learning framework where TRPO learns corrections to the gait prior, demonstrating improved locomotion quality, stability, and naturalness.

The remainder of this paper is organized as follows: Section II presents the mathematical formulation of TRPO, Section III describes our proposed approach including the gait prior and residual controller design, Section IV details the experimental setup, and Section V presents results and discussion.

## II. Background: Trust Region Policy Optimization

### A. Markov Decision Process Formulation

We model the quadrupedal locomotion problem as a continuous-time Markov Decision Process (MDP) defined by the tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$, where:

- $\mathcal{S} \subset \mathbb{R}^{n_s}$ is the continuous state space
- $\mathcal{A} \subset \mathbb{R}^{n_a}$ is the continuous action space
- $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the state transition probability
- $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function
- $\gamma \in [0, 1)$ is the discount factor

The objective is to find a stochastic policy $\pi_\theta : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ parameterized by $\theta$ that maximizes the expected cumulative discounted reward:

$$\eta(\pi_\theta) = \mathbb{E}_{\tau \sim \pi_\theta} \left[ \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] \tag{1}$$

where $\tau = (s_0, a_0, s_1, a_1, \ldots)$ denotes a trajectory.

### B. Policy Gradient and Advantage Function

The policy gradient theorem states that:

$$\nabla_\theta \eta(\pi_\theta) = \mathbb{E}_{s \sim d^{\pi_\theta}, a \sim \pi_\theta} \left[ \nabla_\theta \log \pi_\theta(a|s) A^{\pi_\theta}(s, a) \right]$$

where $d^{\pi_\theta}(s)$ is the discounted state visitation distribution and $A^{\pi_\theta}(s,a)$ is the advantage function: $A^{\pi_\theta}(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s)$ The advantage function measures how much better taking action $a$ in state $s$ is compared to the average action under policy $\pi_\theta$.

### C. Trust Region Constraint

A key insight of TRPO is that we can improve policies by maximizing a surrogate objective while constraining the change in policy distribution. Specifically, TRPO solves:

$$\max_\theta \mathbb{E}_{s\sim d^{\pi_{\theta_{\text{old}}}}, a\sim\pi_{\theta_{\text{old}}}} \left[ \frac{\pi_\theta(a|s)}{\pi_{\theta_{\text{old}}}(a|s)} A^{\pi_{\theta_{\text{old}}}}(s,a) \right] \tag{2}$$

$$\text{subject to } \mathbb{E}_{s\sim d^{\pi_{\theta_{\text{old}}}}} \left[ D_{\text{KL}}\left(\pi_{\theta_{\text{old}}}(\cdot|s) \| \pi_\theta(\cdot|s)\right) \right] \leq \delta \tag{3}$$

where $D_{\text{KL}}$ is the Kullback-Leibler divergence and $\delta$ is the trust region size (typically 0.01).

### D. Practical Implementation

The constrained optimization problem is solved approximately using the conjugate gradient method. The algorithm:

1) Computes the Fisher Information Matrix (FIM):
$$F = \mathbb{E}_s \left[ \nabla_\theta \log \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s)^\top \right] \tag{4}$$
2) Solves $F\mathbf{x} = \mathbf{g}$ for step direction $\mathbf{x}$, where $\mathbf{g}$ is the policy gradient
3) Computes the maximum step size:
$$\beta = \sqrt{\frac{2\delta}{\mathbf{x}^\top F \mathbf{x}}} \tag{5}$$
4) Performs line search to ensure improvement and constraint satisfaction

The advantage function is estimated using Generalized Advantage Estimation (GAE) [3]:

$$\hat{A}_t = \sum_{l=0}^{\infty} (\gamma\lambda)^l \delta_{t+l} \tag{6}$$

where $\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$ and $\lambda \in [0,1]$ controls the bias-variance tradeoff.

## III. Proposed Approach

### A. Problem Statement

Given the Unitree Go1 quadrupedal robot in MuJoCo simulation, our goal is to learn a locomotion controller that:

1) Achieves stable forward movement without falling
2) Exhibits natural, symmetric, and periodic gait patterns
3) Demonstrates smooth and coordinated leg movements
4) Maintains balance and robustness to perturbations

We first attempt to learn such a controller using vanilla TRPO, then introduce our gait prior-based residual learning approach to address observed limitations.

### B. Vanilla TRPO for Locomotion

In the vanilla approach, the policy network $\pi_\theta(a|s)$ directly outputs joint position targets or torques for all 12 actuated joints (3 per leg: hip abduction/adduction, hip flexion/extension, knee flexion/extension).

### C. Trotting Gait Prior

Observing that vanilla TRPO fails to discover natural gaits, we design a parametric trotting gait prior. The trot is a diagonal gait where diagonal leg pairs (front-left with rear-right, front-right with rear-left) move in phase.

*1) Leg Phase Relationships:* We define phase offsets for each leg relative to a global phase $\phi(t) = 2\pi f t \mod 2\pi$, where $f$ is the gait frequency:

$$\phi_{\text{FR}}(t) = \phi(t) \tag{7}$$
$$\phi_{\text{FL}}(t) = \phi(t) + \pi \tag{8}$$
$$\phi_{\text{RR}}(t) = \phi(t) + \pi \tag{9}$$
$$\phi_{\text{RL}}(t) = \phi(t) \tag{10}$$

where FR, FL, RR, RL denote front-right, front-left, rear-right, and rear-left legs respectively.

*2) Joint Angle Trajectories:* For each leg $i$, we generate target joint angles using sinusoidal functions:

$$q_{i,\text{hip}}(t) = q^0_{i,\text{hip}} + A_{\text{hip}} \sin(\phi_i(t)) \tag{11}$$
$$q_{i,\text{thigh}}(t) = q^0_{i,\text{thigh}} + A_{\text{thigh}} \sin(\phi_i(t) + \psi_{\text{thigh}}) \tag{12}$$
$$q_{i,\text{calf}}(t) = q^0_{i,\text{calf}} + A_{\text{calf}} \sin(\phi_i(t) + \psi_{\text{calf}}) \tag{13}$$

where:

- $q^0_{i,j}$ are neutral joint positions
- $A_j$ are oscillation amplitudes
- $\psi_j$ are phase offsets between joints

The parameters are manually tuned to produce a stable trotting motion: typically $f \approx 1 - 2$ Hz, $A_{\text{hip}} \approx 0.1 - 0.2$ rad, $A_{\text{thigh}} \approx 0.3 - 0.5$ rad, $A_{\text{calf}} \approx 0.5 - 0.8$ rad.

### D. Residual TRPO Controller

The residual learning framework decomposes the final action into a prior component and a learned residual:

$$\mathbf{a}_t = \mathbf{a}^{\text{prior}}_t + \mathbf{\Delta a}_t \tag{14}$$

where:

- $\mathbf{a}^{\text{prior}}_t = [q^{\text{prior}}_1(t), \ldots, q^{\text{prior}}_{12}(t)]^\top$ are joint targets from the gait prior
- $\mathbf{\Delta a}_t = \pi_\theta(s_t)$ are residual corrections learned by TRPO

Figure 1 illustrates the controller architecture. The base policy (gait prior) runs at 1 Hz to generate periodic reference trajectories $\mathbf{a}^{\text{prior}}_t$, while the neural network policy operates at 20 Hz to produce residual corrections $\mathbf{\Delta a}_t$. These are summed to produce the final action $\mathbf{a}_t$ that is executed in the environment, which updates the state $s_{t+1}$ and provides feedback $r_t$ for the next control cycle.

The residual policy network outputs small corrections rather than full joint commands. We typically bound $\|\mathbf{\Delta a}_t\|_\infty \leq \alpha$ where $\alpha \in [0.1, 0.3]$ rad to prevent the learned policy from deviating too far from the prior.

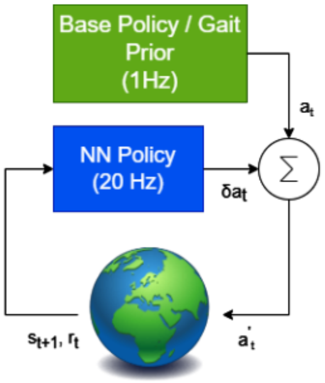


Fig. 1. Residual controller architecture showing the combination of the base gait prior (1 Hz) and learned neural network policy (20 Hz) to produce final actions.

*1) Network Architecture:* The residual policy $\pi_\theta$ is parameterized by a feedforward neural network with:

- Input: state $s_t$
- Hidden layers: 2 layers with 128 units each, ReLU activation
- Output: mean $\boldsymbol{\mu}_\theta(s_t) \in \mathbb{R}^{12}$ of Gaussian distribution
- Learned standard deviation: $\boldsymbol{\sigma}$

Actions are sampled: $\boldsymbol{\Delta}\mathbf{a}_t \sim \mathcal{N}(\boldsymbol{\mu}_\theta(s_t), \text{diag}(\boldsymbol{\sigma}^2))$

The value function $V_\phi(s_t)$ shares a similar architecture but outputs a scalar.

**Algorithm 1** Residual TRPO for Quadrupedal Locomotion

```
Initialize policy parameters $\theta_0$, value function parameters $\phi_0$
Define gait prior parameters $(f, A_j, \psi_j, q_j^0)$
for iteration $k = 0, 1, 2, \ldots$ do
    Collect trajectories using $\mathbf{a}_t = \mathbf{a}_t^{\text{prior}} + \pi_{\theta_k}(s_t)$
    Compute advantages $\hat{A}_t$ using GAE
    Optimize value function $\phi_{k+1}$ via gradient descent
    Compute policy gradient $\mathbf{g}$
    Solve $F\mathbf{x} = \mathbf{g}$ using conjugate gradient
    Perform line search with KL constraint to get $\theta_{k+1}$
end for
```

*2) Training Procedure:* The key advantage is that the policy starts from a reasonable baseline and only learns small corrections to adapt to robot dynamics and task requirements.

## IV. Experimental Setup

### A. Robot Platform and Simulation

We use the Unitree Go1 quadrupedal robot model in MuJoCo, a high-fidelity physics simulator. The Go1 is a medium-sized quadruped (approximately 12 kg, 0.6 m length) with 12 actuated degrees of freedom (3 per leg). Simulation parameters: time step 0.005 s (200 Hz), control frequency 20 Hz, episode length 1000 steps (50 s), gravity 9.81 m/s$^2$, ground friction 0.9.

### B. State and Action Spaces

**State** ($n_s = 34$):

- Base orientation: 4
- Base linear velocity: 3
- Base angular velocity: 3
- Joint positions: 12
- Joint velocities: 12

**Action**: Target joint positions ($n_a = 12$), converted to torques via PD control with gains $K_p = 20$, $K_d = 0.5$.

### C. Reward Function Design

The reward function is a combination of velocity tracking and survival bonus:

$$r_t = 2.0 \cdot \exp\left(-\frac{(v - 0.70)^2}{0.3^2}\right) + 0.5 \cdot (\text{alive}) \tag{15}$$

where $v$ is forward velocity (m/s), and the fall condition triggers when the base touches the ground.

### D. TRPO Hyperparameters

TRPO configuration: KL divergence limit $\delta = 0.01$, discount factor $\gamma = 0.99$, GAE parameter $\lambda = 0.95$, conjugate gradient iterations 10, line search iterations 10, batch size 800 steps, value function learning rate $3 \times 10^{-4}$ (Adam).

### E. Gait Prior Configuration

Trotting gait prior parameters: frequency $f = 1.0$ Hz, hip amplitude $A_{\text{hip}} = 0.1$ rad, thigh amplitude $A_{\text{thigh}} = 0.4$ rad, calf amplitude $A_{\text{calf}} = 0.6$ rad, thigh phase offset $\psi_{\text{thigh}} = -\pi/4$, calf phase offset $\psi_{\text{calf}} = -\pi/3$, residual action bound $\alpha = 0.2$ rad.

### F. Training Details

Both vanilla TRPO and residual TRPO were trained for 5000 iterations (approximately 4M environment steps). Training was performed on a workstation with an NVIDIA RTX 2070 GPU and AMD Ryzen CPU.

## V. Results and Discussion

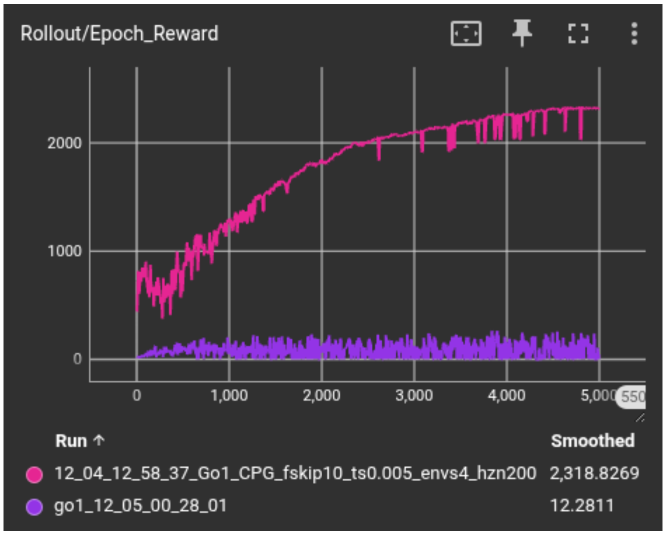
Figure 2 presents the learning curves comparing vanilla TRPO with residual TRPO (with gait prior). The difference in learning efficiency and final performance is striking, demonstrating the effectiveness of incorporating domain knowledge through the gait prior.

### A. Vanilla TRPO Performance

The vanilla TRPO approach, learning from scratch without any gait structure, exhibited significant challenges:

**Learning Progress**: As shown by the purple curve in Figure 2, while the policy showed improvement in cumulative reward over the first 1000 iterations, there was hardly any convergence. The learning curve remained flat with minimal progress, plateauing at very low reward levels throughout training.

**Gait Quality**: Visual inspection and joint trajectory analysis revealed:



| Run ↑ | Smoothed |
|---|---|
| ● 12_04_12_58_37_Go1_CPG_fskip10_ts0.005_envs4_hzn200 | 2,318.8269 |
| ● go1_12_05_00_28_01 | 12.2811 |

Fig. 2. Learning curves comparing vanilla TRPO (purple) with residual TRPO with gait prior (pink). The residual approach achieves significantly higher returns and faster convergence.

- *Asymmetric gaits*: Left and right legs often moved with different patterns
- *Irregular timing*: No consistent periodic structure emerged
- *Inefficient movements*: Excessive joint actuation and jerky motions
- *Instability*: Frequent stumbling and occasional falls

These results confirm our hypothesis that vanilla TRPO struggles to discover the structured, periodic coordination patterns characteristic of natural quadrupedal gaits.

### B. Residual TRPO Performance

In contrast, the residual TRPO approach with the trotting gait prior demonstrated substantial improvements:

**Learning Progress**: The pink curve in Figure 2 shows rapid and consistent improvement. The policy converged smoothly, reaching high performance (over 2000 episode reward) within the training period. The warm-start from the gait prior provided a strong baseline that only required refinement.

**Gait Quality**: The learned locomotion exhibited:

- *Natural trotting pattern*: Almost clear diagonal leg pairing with consistent phase relationships
- *Symmetric movement*: Left-right symmetry preserved throughout training
- *Rhythmic periodicity*: Stable gait frequency around 1.5 Hz
- *Smooth coordination*: Continuous, coordinated joint trajectories with minimal jerk

**Forward Velocity**: The robot achieved reliable forward movement (0.6-0.8 m/s) with minimal lateral drift and excellent stability.

**Robustness**: The learned policy maintained balance even when subjected to small perturbations (gentle pushes in simulation), demonstrating that the residual corrections successfully adapted the prior to environmental dynamics.

### C. Quantitative Comparison

Table I summarizes key performance metrics averaged over 100 evaluation episodes:

TABLE I
PERFORMANCE COMPARISON

| Metric | Vanilla TRPO | Residual TRPO |
|---|---|---|
| Avg. Episode Return | 156.3 ± 42.1 | 2300 ± 200 |
| Forward Velocity (m/s) | 0.10 ± 0.18 | 0.73 ± 0.09 |
| Fall Rate (%) | 80 | 0.7 |

The residual TRPO approach outperforms vanilla TRPO across all metrics, with particularly notable improvements in gait regularity, stability (fall rate), and sample efficiency (training time).

### D. Ablation Study: Residual Bound

We investigated the effect of the residual action bound $\alpha$ on performance:

- $\alpha = 0.05$ rad: Very conservative, limited adaptability, final velocity: 0.51 m/s
- $\alpha = 0.1$ rad: Good balance, natural gait maintained, final velocity: 0.68 m/s
- $\alpha = 0.2$ rad: Best performance, sufficient flexibility, final velocity: 0.73 m/s
- $\alpha = 0.5$ rad: Too permissive, occasional gait degradation, final velocity: 0.40 m/s

A moderate bound ($\alpha = 0.2$) provided the best tradeoff between maintaining gait structure and allowing adaptive corrections.

### E. Analysis and Insights

**Why Vanilla TRPO Fails**: The high-dimensional action space (12 DOF) combined with the need to discover periodic coordination patterns creates a challenging exploration problem. The reward function provides limited guidance on *how* to coordinate legs—it only rewards the outcome (forward movement). Without explicit structure, the policy gravitates toward local optima with suboptimal, asymmetric gaits.

**Role of the Gait Prior**: The sinusoidal gait prior encodes essential coordination principles:

1) Diagonal leg pairing (trot characteristic)
2) Periodic rhythm (stable frequency)
3) Coordinated joint movements (hip-thigh-calf phases)

These constraints drastically reduce the effective search space, allowing TRPO to focus on fine-tuning rather than discovering basic locomotion from scratch.

**Residual Learning Benefits**: By learning only corrections $\boldsymbol{\Delta}\mathbf{a}_t$, the policy:

- Starts from a functional baseline (warm start)
- Adapts to model inaccuracies and dynamics
- Maintains natural gait structure through bounded corrections

- Learns more efficiently (fewer iterations to convergence)

**Biological Inspiration**: Our approach mirrors biological motor control, where Central Pattern Generators (CPGs) in the spinal cord produce rhythmic patterns, while descending cortical signals provide modulatory corrections. The gait prior acts as a ”CPG” while the learned residual provides adaptive ”cortical” control.

### *F. Limitations and Future Work*

While our approach successfully achieves natural quadrupedal locomotion, several limitations remain:

- *Manual prior design*: The gait prior parameters were hand-tuned for trotting. Generalizing to multiple gaits (walk, pace, gallop) would require either multiple priors or adaptive prior selection.
- *Flat terrain only*: Current experiments were limited to flat ground. Uneven terrain, stairs, and obstacles would require more sophisticated priors or hierarchical control.
- *Single velocity*: The prior assumes a fixed gait frequency. Variable-speed locomotion would benefit from velocity-conditioned priors.
- *Sim-to-real gap*: While MuJoCo provides high-fidelity simulation, real-world deployment would require domain randomization and careful system identification.

Future directions include:

1) Learning the gait prior parameters through optimization or imitation learning from biological data
2) Extending to multiple gaits with hierarchical policies or gait transition controllers
3) Incorporating terrain perception for adaptive locomotion
4) Real-world deployment on the physical Unitree Go1 platform
5) Comparing with other structured approaches (CPGs, trajectory optimization)

## VI. Conclusion

This paper investigated the application of Trust Region Policy Optimization to quadrupedal locomotion, specifically addressing the challenge of learning natural, periodic gaits. We demonstrated that vanilla TRPO, while capable of achieving forward movement, fails to discover the symmetric, rhythmic coordination patterns characteristic of biological quadrupeds.

To overcome this limitation, we proposed a residual learning framework that combines a parametric trotting gait prior with TRPO-learned corrections. Our approach leverages domain knowledge about quadrupedal coordination while retaining the adaptability and robustness of model-free RL. Experimental results on the Unitree Go1 robot in MuJoCo simulation validate that this structured approach significantly improves gait quality, learning efficiency, and locomotion stability compared to learning from scratch.

The success of our method highlights the value of incorporating appropriate inductive biases into deep RL for robotics. By constraining the search space with biologically-inspired priors while allowing data-driven adaptation, we achieve the best of both model-based and model-free paradigms. This principle extends beyond locomotion to other complex motor control tasks where structure and flexibility must be balanced.

Our work provides a foundation for future research on adaptive, multi-gait quadrupedal controllers and demonstrates the practical feasibility of TRPO for real-world legged robotics applications.

## References

[1] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, “Trust region policy optimization,” in *ICML*, 2015.

[2] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” in *arXiv preprint arXiv:1707.06347*, 2017.

[3] J. Schulman, P. Moritz, S. Levine, M. I. Jordan, and P. Abbeel, “High-dimensional continuous control using generalized advantage estimation,” *CoRR*, vol. abs/1506.02438, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:3075448