Advanced Skills through Multiple Adversarial Motion Priors in Reinforcement Learning

Nevò Mirzai Hamadani
Tifaine Mezencev
Zoé Frottier

Paper by:
Eric Vollenweider
Marko Bjelonic
Victor Klemm
Nikita Rudin
Joonho Lee
Marco Hutter

École polytechnique fédérale de Lausanne

November 2024

Main Idea/Contributions

Multi-AMP – Extension of AMP

RL method to perform multiple, complex motion styles by dynamically switching between them

Executive summary

Robot Type:

- Wheeled quadruped robot

Control Method:

- Position and velocity control
- Torque control (joint limits)

Design Method:

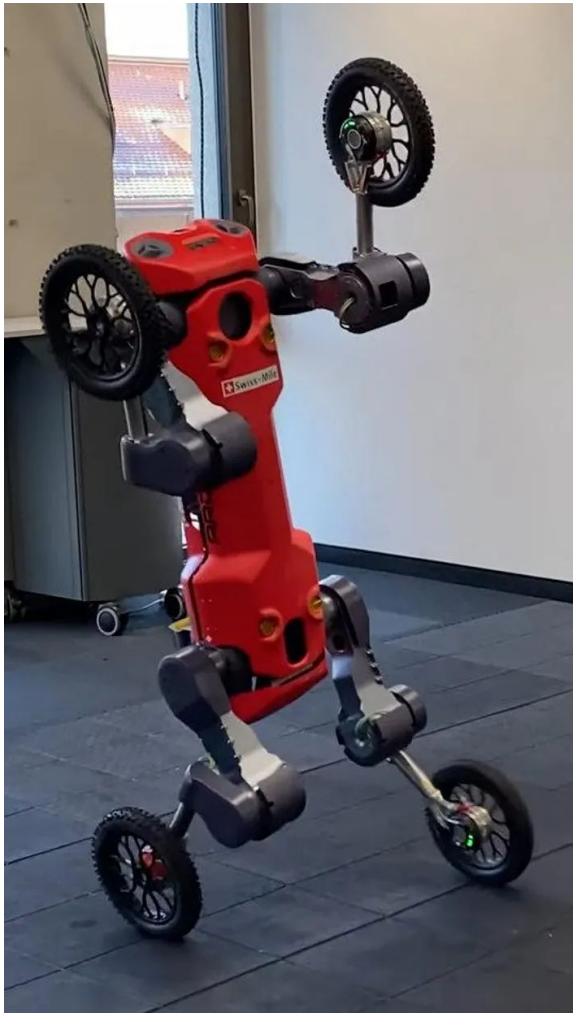
- Multi-AMP (RL)
- Trajectory optimization

Gait Types:

- Multi-gait adaptation (with wheels):
  - 4 legged walking/navigating
  - Upright 2 legged walking/navigating
  - Wheels allow for Hybrid gaits

Sensors:

- Proprioceptive Sensors
  - IMU
  - Wheel/Joint Encoders



Speaker

Adversarial Motion Prior (AMP)

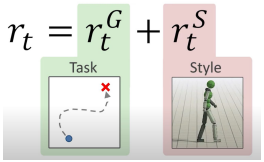
Purpose: train agents to perform tasks with a specific motion style.

Style Guidance: Adversarial discriminator trained on example motion data (e.g., walking, running) to evaluate and enforce style without exact motion tracking.

Dual Reward Structure:

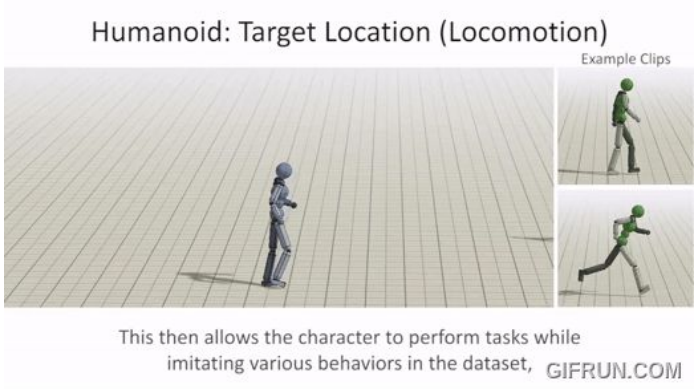
- **Task Reward**: e.g., position
- **Style Reward**: e.g zombie style

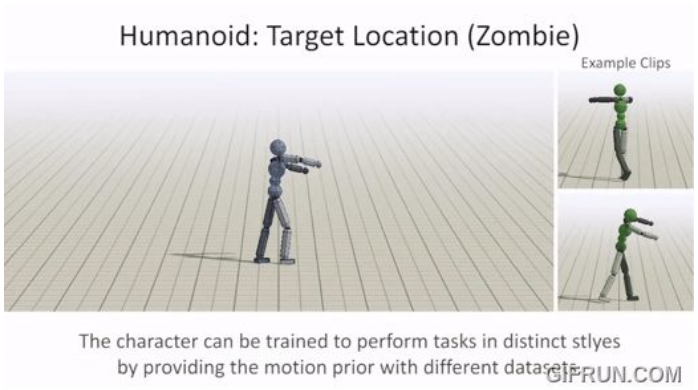
$$r_t = r_t^G + r_t^S$$



Benefits: Produces natural behaviors in simulated environments, simplifying style control without manual motion tracking.

Video Example: Human Style vs Zombie Style





Multi AMP

Goal of AMP:

- Switching multiple different style-reward
- Learn several task simultaneously

Definition:

- States of time-steps where policy applies to the i style B^i_π
- Motion data for each style M^i
- Adversarial setup with n discriminator D^i
- Every trainable style is defined by : $\{D^i, B^i_\pi, M^i\}$

Discriminator:

- Predict the differences between motion database M^i and agent's transition sampled B^i_π
- Least square problem :

$$\begin{aligned} L^i = & \mathbb{E}_{d^{M^i}(s,s')}\left[(D^i(\phi(s),\phi(s'))-1)^2\right] \\ & + \mathbb{E}_{d^{B^i_\pi}(s,s')}\left[(D^i(\phi(s),\phi(s'))+1)^2\right] \\ & + \frac{w^{gp}}{2}\mathbb{E}_{d^{M^i}(s,s')}\left[\|\nabla_\phi D^i(\phi)|_{\phi=(\phi(s),\phi(s'))}\|^2\right], \end{aligned} \quad (1)$$

Task Reward:

Like a standard RL-cycle
policy π(at|st)→predicts an action at
environment → new state s_{t+1} and reward function

$$r_t^{task} = R(c_t, s_t, s_{t-1})$$

Style reward:

command c_t →augmented with c_s:
one-hot-encoded style selector.
c_s: 0 everywhere except the index of the active style i.
Construct the style-descriptor transition dt = [ϕ(st), ϕ(st+1)] and style-reward function:

$$r_t^{style} = -log\left(1 - \frac{1}{1+\exp^{-D^i([\phi(s_t),\phi(s_{t+1})])}}\right)$$

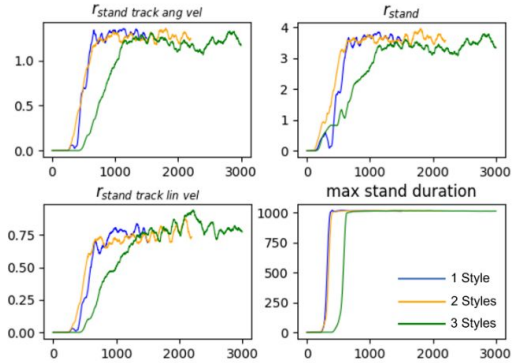
Results

3 Tasks:

- 4 leg locomotion
- ducking under a table
- Stand up, 2 wheel locomotion, sitting down

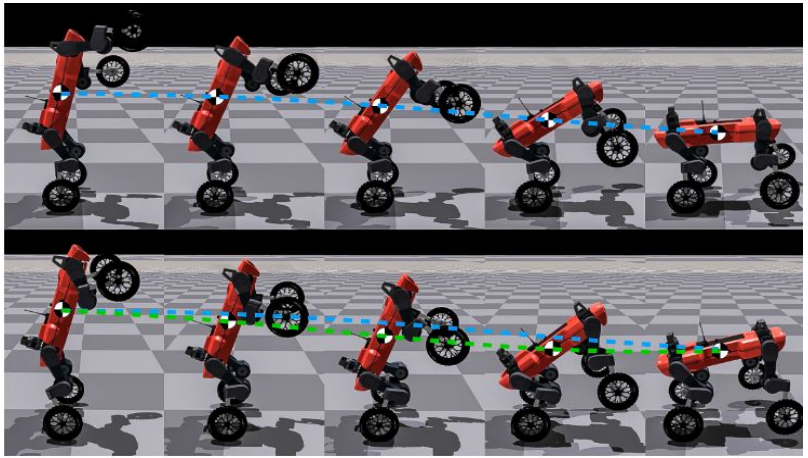
Experiment :

- Spawn 4096 environment in parallel
- Learn 3 tasks simultaneously in a single neural network



Results :

- Learn **equally** well 3 tasks simultaneously than a single task
- Take a bit longer



Sitting down :

- Worked in simulation but default in real
- reverse data of stand up motion recorded

Citations

26 times

... since its publication in october 2022

Applications :

- **Agile locomotion**
- **Path tracking**
- **Robot dynamics**
- **Adaptive gait acquisition** for **quadrupedal** and **bipedal robots**

Areas :

- **Hybrid wheel-legged robots**
- **Autonomous vehicle navigation**
- **Human-like transitions for humanoid robots**

Pros

- Not much tuning needed

-> AMP helps minimize the need for manually calibrating reward functions for each task

- Enable more natural and realistic behaviors

- Improved stability & complex transitions in a variety of dynamic environments

- Learning Complex Behaviors

-> Highly complex movements, difficult to teach using traditional planning or direct control methods (jumps, transitions between different postures, quadrupedal & bipedal (humanoid) configurations,...)

Cons

- Complex reward design & Significant computational power

-> The overall process still requires a enormous amount of data and computational power to stabilize the training resulting in long training times

- Challenges in sim-to-real transitions

-> Can involve long generation times for motion priors, which limits its applicability in real-time scenarios.

- Instability during training

-> Problem with identifying distances between states to assess how well the model is imitating the desired behavior : causes the model to oscillate, fail to converge, or diverge

https://www.mdpi.com/2072-666X/13/8/1252

Ongoing solutions:

- Integration of Wasserstein Distance

-> Increase Stability during training

-> More precise tuning required & More computational complexity

HumanMimic: Learning Natural Locomotion and Transitions for Humanoid Robot via Wasserstein Adversarial Imitation

Offers a more effective gradient and robust measure, which helps the model better capture subtle distinctions between real and generated movements, enhancing stability and reducing the risk of model collapse.

- Conditional AMP

-> Increased Versatility & Smooth Transitions

Conditional Adversarial Motion Priors by a Novel Retargeting Method for Versatile Humanoid Robot Control

More fluid switch between various environment without retraining by allowing a single model to control multiple behaviors or gaits

- Impedance matching frameworks

-> More stability and less Sim-to-Real Transfer problems

Adaptive Gait Acquisition through Learning Dynamic Stimulus Instinct of Bipedal Robot

Adjust the robot's impedance (its response to force) to better match the characteristics of the simulated & real-world environment. The framework minimizes discrepancies between simulation and reality

Possible Exam Questions

1) What is the main Difference between AMP and Multi-AMP?

Answer: *See slides 4-5 :* AMP trains agents to perform tasks in a single, consistent style, e.g. running like a zombie across different objectives. Multi-AMP allows switching between styles, using multiple discriminators to adapt motion for specific tasks. E.g.: it enables a robot to crouch under obstacles and walk normally on open terrain.

2) What are the main advantages of Multi-AMP in comparison to traditional planning or direct control methods ?

Answer: *See slide 8* : Multi-AMP reduces the need for manual tuning of reward functions, enabling more natural and realistic behaviors while improving stability and complex transitions in dynamic environments. It also allows robots to learn highly complex movements, such as jumps or transitions, that are difficult to achieve with traditional planning or direct control methods

Thank you

Nevò
Tifaine
Zoé

École
polytechnique
fédérale
de Lausanne