ReGAIL: Toward Agile Character Control From a Single Reference Motion

Paul Marius Boursin, Yannis Kedadry, Victor Zordan, Paul Kry, Marie-Paule Cani

ReGAIL: Toward Agile Character Control from a Single Reference Motion

Paul Boursin
boursin@lix.polytechnique.fr
LIX - Ecole Polytechnique/CNRS
IP Paris, France

Yannis Kedadry
yannis.kedadry@polytechnique.edu
LIX - Ecole Polytechnique/CNRS
IP Paris, France

Victor Zordan
vbz@clemson.edu
Clemson University
Clemson, SC, U.S.A.

Paul Kry
kry@cs.mcgill.ca
McGill University
Montréal, QC, Canada

Marie-Paule Cani
LIX - Ecole Polytechnique/CNRS
IP Paris, France

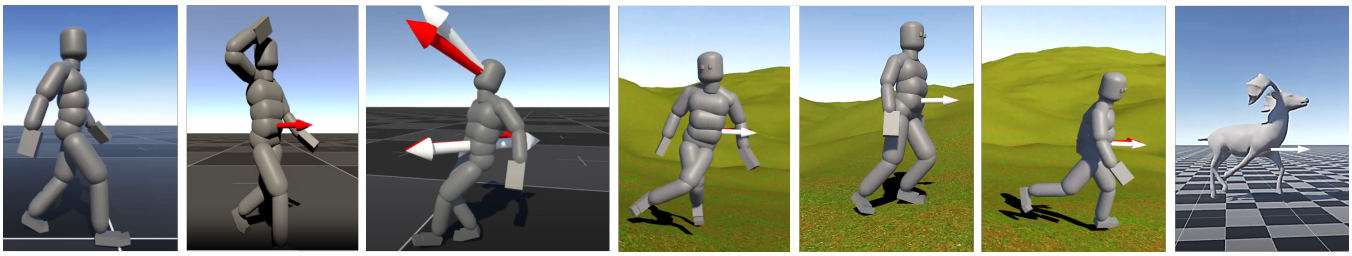

Figure 1: From a single walking cycle (shown at left), our agile character discovers a series of new motor skills such as waving, walking backwards while looking up, moving sideways, walking up and down slopes, which requires adapting joint stiffness over time. Learning this variety of control from a single motion clip is possible thanks to relative pose and velocity observations of state, and augmentation of the observation to include the deviation from each goal (differences between red and white arrows) expressed in a local frame. We also demonstrate our method on a quadruped character (shown at right).

ABSTRACT

We present an approach for training "agile" character control policies, able to produce a wide variety of motor skills from a single reference motion cycle. Our technique builds off of generative adversarial imitation learning (GAIL), with a key novelty of our approach being to provide modification to the observation map in order to improve agility and robustness. Namely, to support more agile behavior, we adjust the value measurements of the training discriminator through relative features - hence the name ReGAIL. Our state observations include both task relevant relative velocities and poses, as well as relative goal deviation information. In addition, to increase robustness of the resulting gaits, servo gains and damping values are included as part of the policy action to let the controller learn how to best combine tension and relaxation during motion. From a policy informed by a single reference motion, our resulting agent is able to maneuver as needed, at runtime, from walking forward to walking backward or sideways, turning and stepping nimbly. We demonstrate our approach for a humanoid and a quadruped, on both flat and sloped terrains, as well as provide ablation studies to validate the design choices of our framework.

CCS CONCEPTS

• **Computing methodologies** → **Physical simulation**; **Procedural animation**; **Reinforcement learning**; **Adversarial learning**; **Learning from demonstrations**.

KEYWORDS

character animation, physically-based simulation, motion controllers, reinforcement learning, generative adversarial imitation learning

1 INTRODUCTION

The synthesis of motion controllers for physical characters has attracted much attention in recent years, driven by the need to generate autonomous characters for video games and immersive virtual environments. A main reason for the appeal of such characters is their ability to both interact and adapt to a changing, interactive physical world. Although significant progress has been made through the use of deep reinforcement learning (DRL) for training control policies, a major weakness of current solutions is that the resulting characters exhibit limited abilities to adapt in the presence of interactions, which can lead to unnatural reactions and a breakdown of visual fidelity overall. Our goal is to expand the controller's capabilities, by making characters more agile, without increasing the burden of developing control policies.

To date, much of the realism of DRL-based controllers for human characters is derived from imitation. While DRL can generate motion controllers without the need for any input data [Heess et al. 2017; Yu et al. 2018], controllers achieve more human-likeness when reference motion is provided as input [Peng et al. 2018]. However, precise imitation does not allow deviation from the reference movement to support the production of a plausible response during an interaction (i.e. one that is compliant to the specific disturbance

forces present). In addition, while DRL combined with generative models, such as adversarial neural networks [Peng et al. 2021a; Xu and Karamouzas 2021], allows learning from input data without exact reproduction of an input motion, these frameworks often require a diverse set of actions and/or a large dataset, which can be cumbersome, especially for characters that are not humanoid, and for which example data is unavailable.

In our work, we investigate the synthesis of more *agile* controllers – both in their ability to cope with a wider variety of scenarios, but also to introduce more realism in their responses – without the need for exhaustive data examples. In particular, real-time video game characters should exhibit agility in everyday actions such as walking, including climbing steep slopes, moving backwards or sideways, turning and stopping quickly, taking stutter steps and skip steps, and so on. Relying on example data can quickly become prohibitive. Instead, we seek to train motion controllers to generate new, visually plausible and diverse motor skills from sparse motion data, such as a single walking clip.

We call our solution ReGAIL, for Relative Generative Adversarial Imitation Learning. Indeed, the key insight is the use of relative pose, relative velocity, and relative goal deviation. At the same time, we suppress absolute information usually used within discriminators of alternative systems, such as features like absolute position and angular velocity. These changes permit the input motion clip to give good guidance during learning under a larger variety of conditions, exposing a wider distribution of example states during training, thus greatly improving the capacity of generalization of the learned policy despite very limited input data. Further, we augment the policy action space to include the capacity to select its own servo values during control. While our goal for this is to increase agility for robustness, recent findings support that such stiffness modulation is also more humanlike [Xie et al. 2023]. During training, we purposely exercise the character to perform under a wide array of conditions, including a rich variety of environmental settings (e.g. terrain variation), differing target directions, and a collection of secondary goals such as hand location, head look-at orientations, and root facing conditions (e.g. facing forward while moving backwards).

We show results for both humanoid and quadruped characters, showing that a variety of walking-related behaviors that can be generated from a single straight-line walking motion clip (see Figure 1). We compare our results with the state of the art. Finally, we report ablation studies to support the components that empower our technique.

2 RELATED WORK

Following seminal work exploring the manual design of motion controllers, deep reinforcement learning (DRL) has emerged as the most effective method for synthesizing control to date. The set of possible actions of a physically-based character model being continuous, most methods use policy gradient algorithms, such as PPO [Schulman et al. 2017], to learn the optimal action policy given state observations (see the full survey of Kwiatkowski et al. [2022]). While DRL methods are able to generate motion controllers directly, for instance by favoring symmetric and low energy motion [Yu et al. 2018], generated gaits for humanoid agents remain far from humanlike without the presence of reference motion. Therefore, most recent works incorporate different ways to train controllers from example motion. Early methods either aimed to reproduce the specific motion provided in an input clip through direct imitation, such as DeepMimic [Peng et al. 2018], or proposed the generation of more diverse motion controllers, such as DeepLoco [Peng et al. 2017], by introducing a hierarchical approach to enable learning of a high-level skills for foot placement, used in conjunction with low-level motor actions learned from reference data.

Greater diversity is achieved by combining DRL with generative models. For example, AMP [Peng et al. 2021b] and ICCGAN [Xu and Karamouzas 2021] combine generative adversarial neural network (GAN) [Ho and Ermon 2016] using a discriminator to calculate similarity of motion features generated by the controllers with those in unstructured motion clips. While AMP and ICCGAN are functionally similar, AMP shows control over more complex tasks, such as dribbling while ICCGAN uses multiple discriminators to avoid model collapse, a common problem plaguing GAIL motion controllers. Follow-on work continues to appear broadening capabilities of each [Tessler et al. 2023; Xu et al. 2023a,b].

Generating a rich set of motion controllers for a given character in these frameworks may require a growing set of reference data (e.g., 30 minutes of video [Tessler et al. 2023]), which is both cumbersome and may result in longer training time. In contrast, Lee et al. [2021] demonstrates the generation of motion variations from a single input clip, where the motion space is progressively expanded to include a full family of motor skills embedding the demonstrated action. Our work follows a similar stream, relying on GAIL and PPO while aiming to generalize imitation learning from sparse input, such as a single motion clip or a small set of manual keyframe data. In contrast, we do not grow a single family of motor skills, but discover controllers for varied gaits with controllable directions. We accomplish this by reformulating both the observation and input clip. We note Lee et al. [2021] and our approach are complementary and could be combined to meet both variation within a motion controller, and diversifying the type of control possible from a single clip. While other work pre-train policies with latent spaces and train networks to control them [Peng et al. 2022; Tessler et al. 2023], or rely on VAEs [Feng et al. 2023; Ling et al. 2020; Won et al. 2022; Yao et al. 2022], our technique permits generalization of motor skills from a single input clip through rewriting observations. Thus our character only needs to be trained once to learn how to diversify an input walking gait.

Beyond generalization, we are also keen to improve character animation through more humanlike control and responsivity. We are inspired by papers like Peng and van de Panne [2017] which asks if the action space matters and we believe the action space plays a key roll in compliance and agility through stiffness modulation. Thus, we equip our policy with active control over joint stiffness, so it can learn to employ stiffness as necessary for the current task. Muscle tendon units are one way to introduce varying stiffness through antagonistic muscles [Geijtenbeek et al. 2013], while other work includes PD gains in the action space [Chentanez et al. 2018]. Another strategy is to displace the original reference motion to minimize interaction forces without modifying an existing imitation controller to animate compliant interactions [Lee et al. 2022]. In general, in contrast to muscles and tendons or other

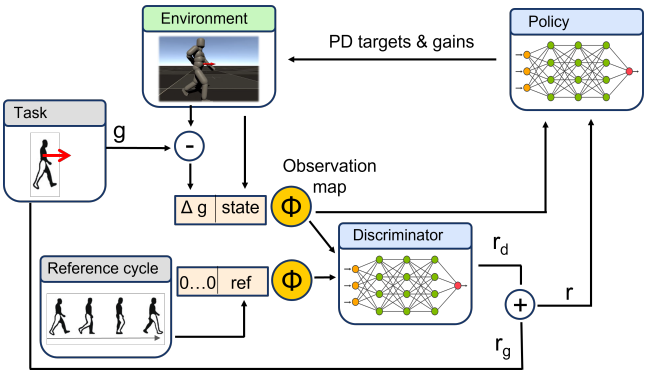

Figure 2: The ReGAIL training framework is shown. A key distinction of our approach is that the observation map Φ transforms the state information and goal deviation Δ to provide only relative information to the discriminator and the policy. Also, the policy provides both targets and gain information. Further, during training, the policy is set to maximize the sum of goal and discriminator rewards. Here, the discriminator receives relative information for the reference motion with zero goal deviation.

approaches, the simplicity of PD controllers remains appealing for learning character controllers that can interact with the environment. Likewise, regardless the choice, recent work suggests that we must take care to avoid naive character controllers that are simply too stiff [Xie et al. 2023].

Along with the question about the choice of action space, Kim and Ha [2021] address the natural follow up question, confirming that choices made about the observation space are also critical to the success (and speed) of learning. This is also related to our work in that we choose observations to be relative pose and velocity combined with goal deviations, with this choice being helpful to the overall objectives (i.e., expand the variety of motions that can be controlled while being rewarded by a discriminator trained with only a single motion clip). Early work of Ding et al. [2015] let optimization create low rank feedback policies which can ultimately ignore parts of the state while selecting other parts as important. In our work we purposefully select the local relative velocity to be important, while we omit information that would hinder performance of the larger collection of motions we would like the controller to learn (e.g., hand and foot positions in the local root, world-up aligned, coordinate frame).

3 OVERVIEW

Our characters are modeled as a hierarchy of rigid links, each connected by three degree of freedom (DOF) rotational joints. Although nothing specifically limits us to these, a human and a quadruped models are showcased in this paper, with mass and joint positions set based on known weight and bone size for their real-world counterparts. Following current practices, we attach PD controllers (servos) to all the joints and use them to calculate torques, rather than directly defining torque values. One reference motion cycle is provided for each character.

3.1 Including PD Gains in the action space

Diverging from most current work, we include servo gains as part of the values (actions) to be set by the policy which permits the control to modulate its stiffness and gives it further capabilities such as increased compliance as it executes tasks. As a counterpart, we supply joint limits to all rotational DOFs, to discover these extended capabilities in a reduced posture space. The policy network π is set to output five scalars for each joint, each in the range $[-1, 1]$, namely three Euler angles as target for the joint and two servo values for the proportional derivative (PD) control. The former are linearly mapped for the preset lowest and highest Euler angle limits at each joint. Similarly, the servo values are mapped between zero and the maximal stiffness and damping values set for each joint.

Because the policy selects both the target positions and low-level servo gains, it has greater control over the character through each of its articulations, giving the neural network the capacity to not only move however it wants, given its configuration, but also use more or less compliance to do so. Conversely through joint limits, the policy is prevented from accessing or exploring invalid regions by limiting the articulation space of the DoFs. Not only does this force the characters to respect realistic limits, but it also reduces the action space and thus simplifies the control problem.

We experimentally observed that this limited action space improves training performance. We also observed that it does not prevent training agents to imitate motion data that goes outside the limits of their DoFs.

3.2 Training framework

ReGAIL's training framework is summarized in Figure 2. As was done in [Peng et al. 2021b], we pair PPO with GAIL to train motion controllers. However, we propose a number of changes to the framework, described next, to align with our goal of increasing the agent's agility, through its ability to use a variety of motor skills based on context.

First, to allow the discriminator to value a broader set of motions rather than strict imitation of the reference clip, we carefully design a novel observation map that re-frames the global simulation (and environment) features to be meaningful in the context of agile control and extended gait possibilities. As detailed in Section 4, we replace absolute observations for position and velocity by relative joint pose and velocity information.

Second, to extend the conditions appropriate for our repertoire of high-level goals (reach, look-at, etc.), information about how closely important aspects of the state match those of the desired goals is added both as part of the observation map for the discriminator as well as to the policy. Section 5 describes our goals in more depth.

3.3 Training process

While the discriminator and policy are trained simultaneously, the agent is assigned goals that change every few seconds. For example, it is asked to face, look and move in different directions at once.

The combination of such different tasks forces the policy to adapt the gait to accomplish them simultaneously: for example, walking while facing perpendicular to the direction of motion leads the agent to discover side steps in order to satisfy both objectives. Thus, the agent learns to generalize the input walking gait and in

particular to make transitions between gaits and directions. He also learns to turn over at any time.

4 RELATIVE OBSERVATIONS

The selection of observations is a key part of the success for DRL control problems. In our work, we are only working with a short clip (or even a single cycle) or reference motion, yet we would like to successfully train a variety of tasks that correspond to the reference motion style. Our choice of features for observation are therefore quantities that hide, in a way, or otherwise de-emphasize information that would limit the motions rewarded by the discriminator.

We propose the observation state mapped through Φ to include the following features:

- the rotation of each joint relative to the parent link;
- the relative linear velocity of each link with respect to its parent link's velocity, expressed in the parent frame;
- the current deviation from goals, as detailed in Section 5;
- and the height h of the agent, defined as the vertical distance between the lowest foot and the top of the head.

We provide more details on each of these features subsequently below. However, an important aspect of this observation map is that the information is provided in a relative form as consistently as possible. For example, in contrast with previous work, we do not include any joint or end effector position (globally or relative to the root) in the observations. Indeed, our experiments show that root-relative information prevented generalization to new gaits and behaviors.

Purposefully missing in the list of features above are observations about speed or direction of the root frame's motion. Indeed, the latter are treated as goals, as opposed to being directly included in observations for imitation. However, as stated above, deviation from each goal is included within the relative observation provided to the discriminator, in addition to providing it to the policy. From our experimentation we found that during training, combining the state trajectories from policy evaluations with goal deviation in the manner described permits the discriminator to combine valuation of the aggregate task and goals simultaneously.

4.1 Relative Pose

There are several options for the coordinate frame describing the pose. We choose to use the parent link frame, providing local information about the *relative* pose only. This is in contrast to most approaches that favor more global imitation observations, i.e. describing the entire pose in the character's root frame.

Our choice for pose to be relative to the parent link naturally supports greater deviation from the reference - as a bend at the elbow, for instance, can be seen as locally matching the reference pose regardless the state of the lower arm relative to the root. Thus, for example, leaning into an upward slope can be more readily permissible to the discriminator when it is provided as local relative information. This extends throughout the pose as the character performs activities farther from the original motion embedded in the reference. Furthermore, when legs deviate from the reference, each individual leg joint will have a small deviation with respect to its parent, while if expressed in the root frame, small deviations of parented joints would stack up to a much larger deviation for the foot.

We provide the pose of a link, or joint, by using two axes of a rotation matrix, as the normal and tangent vector of the link. This is convenient both because it makes use of less data and avoids the wrapping problem.

4.2 Relative linear Velocity

Including a velocity observation for each body is known to help produce successful controllers with GAIL [Peng et al. 2021b]. However, unlike previous work, our observation for velocity is a relative measure, namely the difference between the linear velocity of a link and the linear velocity of the parent link, expressed in the parent's coordinate frame. This is consistent with our use of relative pose, in that the relative velocity is agnostic to the more global context of the root, and also supports greater variation in the discriminator.

Thanks to this choice we avoid expressing velocity information in the character's root local frame which would provide full egocentric information about the direction and magnitude of the link velocities. To make this clear, consider that backwards walking is easy to discriminate from forward walking in the root frame. However, in contrast, when using velocity relative to the parent link, knee and hip velocities closely resemble one another (both are contracting the leg) during forward and backwards walking. We observe similar benefits for walking sideways and on different slopes. While one could mitigate this problem by adding additional reference motion, our approach enables the control to discover more versatility within a single policy derived from a single reference clip.

4.3 Relative to Gravity Direction

Despite our efforts to limit information provided to the discriminator, we recognize the importance of providing some information relative to the direction of gravity. While our investigations show that we can train policies for walking on flat terrain without gravity direction relative information, we observe that including this extra information is necessary to learn controllers that successfully walk on uneven terrain.

The gravity direction features we compute are the projections of the feature vectors describing each limb's rotation and linear velocity onto the world up direction, which we evaluate using a dot product. In practice, we augment the coordinates of the previous features with this extra coordinate.

5 GOALS AND REWARDS

The agent's main objective is to walk in a user-specified target direction at some prescribed speed. During training, the agent is tasked to move in random directions changing every few seconds. The agent thus has to learn how to generalize the walking gait and in particular to transition between directions, as well as to learn to turn.

Prior to discussing task specific goals and rewards, it is important to highlight a key difference between previous work and our approach. Namely, we propose the use of *goal deviation* describing relative goal satisfaction from the difference of a goal quantity

defined by the state, and the desired goal set by the user. These deviations, as detailed below, become the primary state observations provided to both the discriminator and policy by which goals are achieved.

When using GAIL to produce a reward signal from expert demonstrations, the goal defined by an extrinsic reward given by the programmer may conflict with the reward signal given by the GAIL discriminator. This challenge is overcome by our use of the goal deviation to inform the policy and discriminator of deviations that require correction.

As the goal deviation is defined as the difference between state and target, the desired value is always 0. As such, all goal deviations observations are explicitly set to 0 when converting the input motion clip through the relative observations map. Thus, the expert demonstration is considered to always be following the tasks perfectly, forcing the policy to also follow the task, as well as the demonstration. Therefore, it is crucial for the policy to keep the goal deviation as close to 0 as possible in order to avoid being discriminated. This means that the GAIL reward signal now encourages imitation and the goals, at the same time.

In other words, we use the discriminator's observation to indirectly resolve an equation during training :

$$\begin{aligned} \Delta g &= s_g - g, \\ \Delta g &\to 0 \\ \Leftrightarrow s_g - g &\to 0 \\ \Leftrightarrow s_g &\to g. \end{aligned} \tag{1}$$

where s_g is the state related to the goal g, and the task to accomplish is $s_g = g$. We write $\Delta g \to 0$ as the discriminator will reward the policy for keeping Δg close to 0. Consequently, the same will happen for keeping s_g close to g, which is precisely the task at hand.

In the following subsections we provide more details on goals and rewards. The total rewards r_g includes the sum of rewards defined for each task. The goal deviation vector Δg is assembled by concatenating the deviation for each task (which is likewise then concatenated with the observations).

5.1 Speed, Direction, and Turning

In our experiments, the reference clip involves walking forwards without turning. In the standard AMP framework, the discriminator would bias the policy against learning to turn. When using a goal reward signal that trains the policy to learn to turn, it would conflict with the GAIL reward signal that learns to walk straight. Scaling the rewards to work together then becomes a complex problem because the goal reward reduces the GAIL reward and vice versa.

To control horizontal speed and direction a 2D vector, $\overrightarrow{v_g}$ is provided as part of goal vector g. $\overrightarrow{v_g}$ is obtained by scaling the desired horizontal direction by the desired speed, expressed the character's root local frame.

The goal deviation Δg includes $\overrightarrow{v_r} - \overrightarrow{v_g}$, where $\overrightarrow{v_r}$ is the character's root velocity expressed in the root local frame. Thus, Δg informs the policy of how the current velocity needs to be corrected, and the discriminator will incite the policy to match $\overrightarrow{v_r} - \overrightarrow{v_g} \to 0 \Leftrightarrow \overrightarrow{v_r} \to \overrightarrow{v_g}$.

We likewise define a reward associated to the task of speed and direction control:

$$r_v = \min\left(\frac{\overrightarrow{v_r} \cdot \overrightarrow{v_g}}{\|\overrightarrow{v_g}\|^2}, 1\right). \tag{2}$$

We normalize $\overrightarrow{v_g}$ to get the root speed in the target direction, then normalize that again by $\|\overrightarrow{v_g}\|$ so that the reward for going at the target speed is exactly 1. We do not give extra rewards for going faster than the target speed.

In this case of controlling speed and direction, we present ablations in the results section that show that not only does the goal deviation observation help the agent perform the task, but that the additional extrinsic reward is not absolutely necessary.

5.2 Root-body facing direction

In the reference motion clips used in our experiments, the character always faces forward. To generalize motion to more diverse gaits, we include a goal for the character to face in any horizontal direction while moving in another prescribed direction; for example, it may be asked to face orthogonal to the direction of motion, so as to produce a sideways walk.

Here, we supply the 2D vector $\overrightarrow{d_g}$ as the desired forward direction of the character's root.

We add $\overrightarrow{d_r} - \overrightarrow{d_g}$ to Δg, where $\overrightarrow{d_r}$ is the current forward direction of the character's root. This results in a secondary imitation objective $\overrightarrow{d_r} \to \overrightarrow{d_g}$.

The reward for this task is computed by a simple dot product:

$$r_d = \overrightarrow{d_r} \cdot \overrightarrow{d_g}. \tag{3}$$

Since $\overrightarrow{d_r}$ and $\overrightarrow{d_g}$ are both normalized, the reward is naturally capped at 1 being the optimal case.

We show in the ablation studies provided in the supplementary video that the relative goal observation is necessary to combine walking and facing direction tasks in our scenario. Indeed, when we remove Δg from the discriminator observations but keep it in the policy observations, the policy learns to walk and imitate perfectly while completely ignoring the facing direction, forfeiting the secondary task and its reward. This shows that the goal deviation is necessary to teach the policy to face in the desired direction.

5.3 Gaze Direction

Much like the root-body facing direction, we provide a head gaze direction as the 3D vector $\overrightarrow{d_g}$. While the user may set this direction in different frames, e.g., relative to the desired motion direction or facing direction, we express this goal in the local frame of the character's neck. Likewise, we include $\overrightarrow{d_h} - \overrightarrow{d_g}$ into Δg, with the current direction of the head given as $\overrightarrow{d_h}$; and define a reward as the dot product between $\overrightarrow{d_h}$ and $\overrightarrow{d_g}$.

Having the controller learn to orient the head is preferable to superimposing a kinematic fix to the head pose at run time because the policy and simulator will ensure that the goal is achieved in a physically valid manner. For example, the chest will naturally turn to accommodate the neck when it is approaching its joint limits.

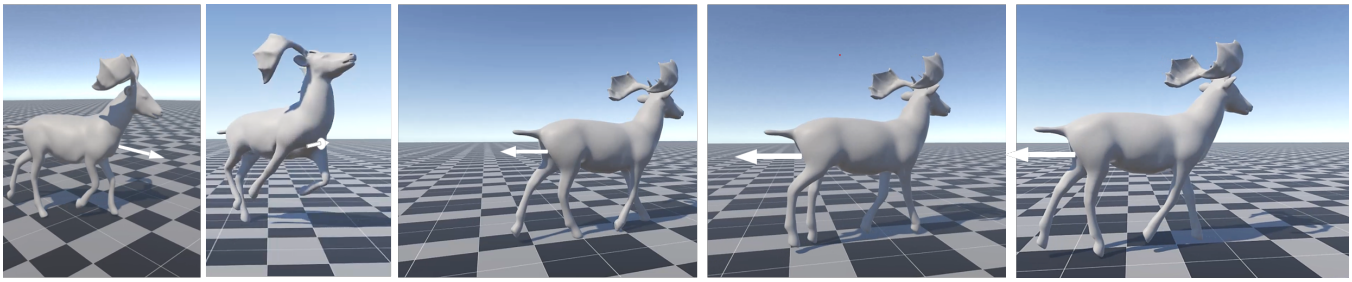

Figure 3: In addition to learning to imitate the input walk cycle (left) our quadruped discovers a fast way to turn by buckling on two legs (second picture), as well as a new backwards gait (middle to right).

5.4 Hand Position

Reaching is a useful goal that is also straightforward to include by modifying discriminator observations. We provide $\bar{p}$ as target position for the hand expressed in the local coordinate frame, and concatenate the vector to the goal g. The goal deviation Δg, again includes the difference between the desired and current position. The reward is computed as

$$r_h = \|p_{t-1} - \bar{p}_{t-1}\| - \|p_t - \bar{p}_{t-1}\|, \tag{4}$$

where p_{t-1} is the position of the hand and $\bar{p}_{t-1}$ is the target hand position at time step $t-1$. It may seem counter-intuitive to use only the target hand position from the last frame; however, since p_t depends directly on the action taken at frame $t-1$, which was calculated from a network inference that had as input $\bar{p}_{t-1}$, there is no way for the model to attain $\bar{p}_t$, except by predicting it. To avoid forcing the model to learn to predict the target, we reward the network for getting closer to the last target it saw.

Empirically, providing a goal deviation to the discriminator for hand pose tasks directly foils training, as Δg is often large, resulting in model collapse, as the discriminator will always give very low scores to the policy. A solution to this would be to rewrite the goal deviation to be a velocity difference, instead of a position difference, to encourage the hand to always move towards the target at a specific speed. This would significantly reduce Δg while still explicitly containing the task; although this has not yet been tested.

In addition, the poses required to satisfy the hand task are too different from the reference motion data to receive a high value by the discriminator. To solve this, we *remove* features from the discriminator, specifically pose and velocity features for the arm joints (shoulder to hand), as well as Δg for the hand. All are still passed to the policy. This involves retraining the controller for the reaching task, while an interesting avenue for future work would be to learn a discriminator and policy that can selectively suppress features as needed, much like [Zolna et al. 2021].

5.5 Fall prevention

We explicitly teach the agent not to fall by including a height tracking reward. Specifically, we give a negative reward to penalize the current policy and terminate the episode early if the character fails to maintain a minimum height, i.e. if:

$$h < 0.7 \cdot h_{\text{max}}, \tag{5}$$

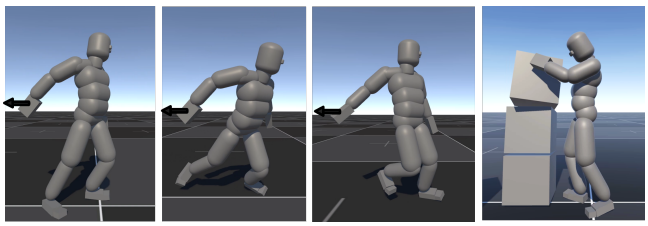

Figure 4: When strongly pulled-back by the hand (left), the character struggles to walk, and may need to make a few back steps to keep the equilibrium (middle); With the hand reaching task, the agent can interact with other objects (right).

where h_{max} is the height of the character when standing upright.

6 RESULTS, VALIDATION, AND DISCUSSION

Our implementation relies on the Unity ML-Agents library [Juliani et al. 2020] for the deep reinforcement learning framework, including PPO and GAIL. We use two character models in our experiments, a humanoid and quadruped, both provided with a single kinematic animation clip constituting a single walk cycle each.

Interactive control. After training, the trained controller can be interactively given with new tasks, under the user's control. Example animations appear in the accompanying video clips of various results, and we guide the viewer's eye to the nimbleness and fluidity of the character's responses, including the 'fancy' footwork in which the character engages during the the various scenarios. We see that the character employs sidesteps, back steps, makes small hops or skip steps as necessary for the context in Figures 1 and 4.

Likewise, the quadruped bucks backward onto two legs to make faster turns in a plausible and emergent fashion (see Figure 3). Notably, the creature did not receive any input of two-legged motion and found this *improvisation* in response to the ReGAIL control.

Rough terrain. We also trained the characters on rough terrain with varying slopes (see Figure 1 and the companion video). To this end, we crafted a terrain with a height map generated with octaves of Perlin noise. During training, we generate random terrains and distribute the agents on the terrain, instructing them to walk around. We pass to the policy the distance and normal vector resulting from a single ray cast in front of its feet. Resulting motion reveals that

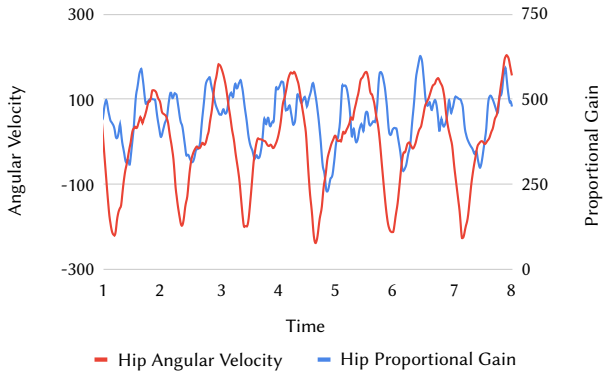

Figure 5: The policy exhibits gain modulation (blue) of the hip joint in the swing direction, visibly increasing and decreasing during swing and stance phases, as can be inferred from the angular velocity of the hip swinging back and forth (red). Note the maximum proportional gain for the hip is 1000.

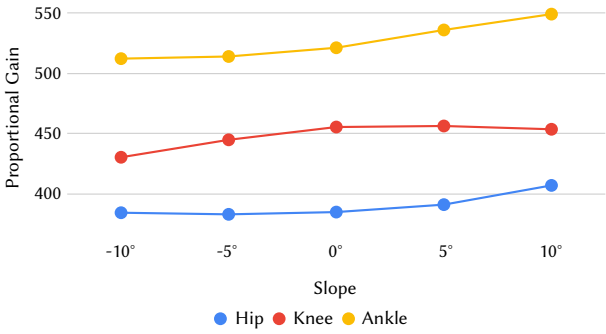

Figure 6: Trained policy reveals that the average stiffness of leg joints correlates with the slope of the terrain. The policy directly sets larger gains when walking uphill and selects lower gains in descent.

the humanoid agents learn to lean forward when climbing and lean backward when descending in an expected manner.

Stiffness control. With respect to gain modulation, we observe that the servo gains of the trained policy reveal desirable nuances with respect to how they change both within individual walk cycles and in response to walking on different slopes. Notably this is without careful shaping or guidance, merely training for imitation of the same motion cycle under a collection of different tasks. Figure 5 shows modulation of stiffness during swing and stance phases of the walk cycle, while Figure 6 shows the average stiffness of hip knee and ankle while walking on terrain of different grades. Through gain selection they exhibit higher gains during the uphill climbs and lower their gains in descent.

Computational time. We train policies with 20 environments in parallel each containing 50 agents for a total of 1000 concurrent

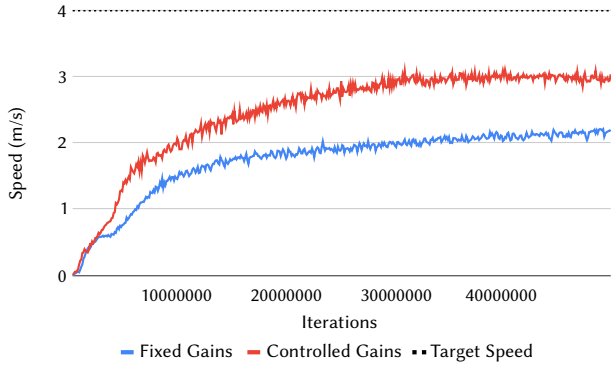

Figure 7: An agent that has PD gains in its action space learns to match a target speed better than an agent with fixed gains.

agents, for 50 million steps; however more steps would result in better quality motion. It takes 16 hours to train a policy on a machine with an Intel Xeon W-2255 CPU and an NVIDIA RTX A5000. During inference, the same computer can simulate the physics, infer the policies and render 30 agents at the same time, at 100 frames per second, in the Unity editor which adds considerable overhead. Since the policy is a neural network that has 3 hidden layers of 512 neurons each, it is very fast to infer.

6.1 Ablation studies

Ablation of servo gains from the action space: Figure 7 shows that an agent with fixed gains does not perform as well at a fast target speed goal in comparison to our actions that includes gains. In this ablation the fixed gains we use are half of the maximum gains described in Section 3.1. Figure 5 motivates the decision to fixing gains at half of the maximum, i.e., we observe that agents that can control their gains tend to use this value on average.

Ablation of the relative terms from the observations: By ablating the relative nature of velocity observations, we show that the agent will learn to turn around when it is instructed to go behind itself, while in the exact same scenario with the relative velocity, it learns a new gait, namely walking backwards. In this ablation experiment, the observation includes the linear velocity of joints instead of their difference with the parent's velocity. The discriminator thus sees the global heading direction and speed of each joint. Since the reference motion data only includes velocities that are pointing forwards, it makes it difficult for the policy to learn walking backwards. Furthermore, when we add a secondary objective forcing the agent to face opposite of the target walking direction, the agent completely ignores imitation in this ablation. These results are presented in the supplementary video.

Ablation of goal-related terms: Removing the goal deviation from the discriminator's inputs demonstrates that this term is useful for optimizing the goal. Figure 8 shows our evaluation of this, in the case of an agent asked to walk in a prescribed direction, at a prescribed, constant speed (dotted line). We compare the speed reached by the character in the target direction throughout training

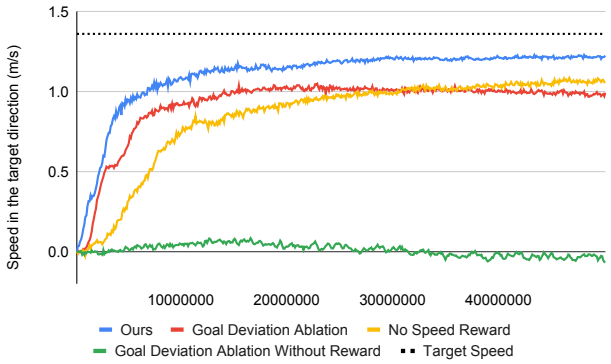

Figure 8: We show the average root speed in the target direction (scalar product between root velocity and target direction) of the agent from beginning to the end of training (steps). Providing the goal deviation to the discriminator teaches the policy to walk at the target speed and direction, with or without the help of an extrinsic reward. This chart shows that goal deviation and the reward work together to learn the task efficiently, and that the reward is not absolutely necessary, by selectively removing the goal deviation from the discriminator, and/or removing the speed reward.

in the four following cases: Our solution (blue curve); With the ablation of the deviation from goal in the observation provided to the discriminator (red curve); With the ablation of the extrinsic goal-related term in the reward (yellow curve); and with ablation of both terms (green curve). In the goal deviation ablations, we remove the goal deviation term from the observations provided to the discriminator, but still keep it in the observations provided to the policy. This shows the learning curve when the discriminator does not help to achieve the goal.

We then compare this to our solution without any extrinsic goal-related reward (yellow curve), which shows that while it leads to slower learning, the use of the deviation from the goal in the discriminator's reward alone, still outperforms its use in extrinsic reward. By combining both ablations (green curve), we show that the policy never learns to walk in the correct direction, because neither the discriminator nor the reward encourages it to achieve the goal.

6.2 Comparisons with AMP

In the following, we compare our model to a pretrained AMP model. The AMP model used is the *Target Heading* model from the original paper [Peng et al. 2021b] which was trained using walking, running and turning animation clips for around 60 millions samples. Results are available in the companion video.

Reaction to projectiles: We first compare the reaction of the models to cubes of different sizes and masses thrown at them. Results show that the AMP model barely reacts even when the cubes get quite heavy, and often falls. In contrast, our model is more agile and tries to adapt its gait to avoid falling.

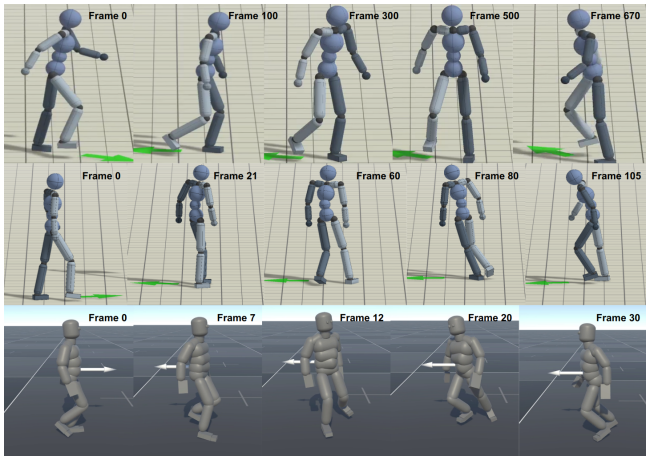

Figure 9: Our model turns around in-place (down) while the AMP model trained with turning motion data has to travel a full arc resulting in a much slower turnaround (middle) but still faster than the AMP model trained without turning motion data (up)

Turning: In a turning task, the AMP model takes up to a few seconds to complete a full turn and must travel a full arc to turn. In contrast, our model, trained without any turning motion data, performs a much faster and in-place turnaround (see Figure 9, and the companion video). We also compared a new AMP model trained without any turning motion data. This new model is even slower at turning around because it needs to travel a wider arc to turn and often falls before the end of the turnaround.

Walking backwards: In this comparison, we test the ability of the models to generalize from a forward walking input to a backward walking gait. For this purpose, we trained the AMP model on a single forward walking motion. We also modified the total rewards of the model by adding a modified version of our facing reward (see Equation 3) where $\overrightarrow{d_g}$ is now the opposite of the target direction to force the agent to walk backwards.

While both models succeed in walking backward, ours is able to alternate his legs in a quite natural way while the AMP model only manages to do tiny jumps (see Figure 10, and the companion video).

We compare our method with AMP's as it is the closest method in the literature, along with ICCGAN [Xu and Karamouzas 2021]. However, there are slight differences in the environments, most of all, different physics simulators: PhysX (ours) and Bullet (AMP), and implementations of PPO. In addition, many parameters are different; reproducing the same environment would not be feasible. However, many parameters are more constraining in our environment, such as the joint rotation limits and maximum torque values.

7 DISCUSSION AND CONCLUSION

This research addresses a critical need in the deployment of physics-based characters. Namely, agents must be robust and agile in their capabilities, and respond plausibly when they deviate from expected

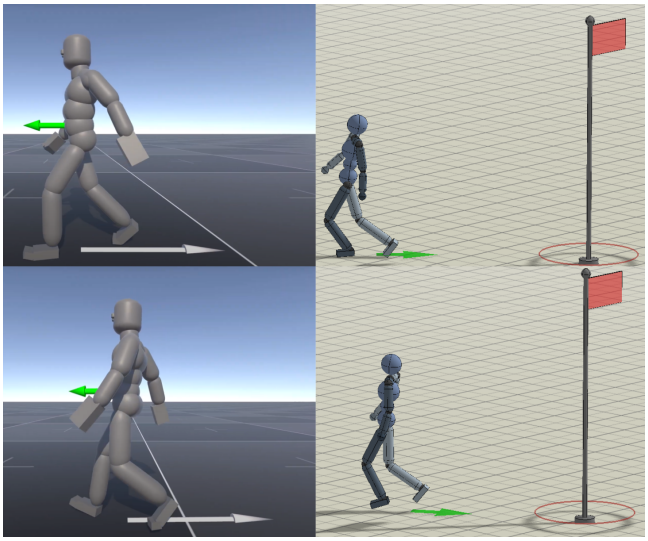

Figure 10: Our model adequately alternates the legs when walking backward (left) while the AMP model jumps with always the same leg positions (right). The arrow on the ground is the prescribed walking direction.

inputs near a given reference motion. ReGAIL is a new way of training controllers where the input motion clip is used as an inspiration which is applied to other contexts, rather than as a strict target motion to be precisely imitated. This is done by re-expressing both the observation of the agent's state and the input data as relative quantities that support variation as the agent moves into novel states. Our investigations reveal nuances and emergent qualities in agile movement that are enabled by these key changes to state of the art systems.

A limitation of our approach is that it does not guarantee natural locomotion gaits. While it broadly generalizes the single motion clip to new improvised behaviors, there is no explicit constraint on the generated gaits and therefore there is no guarantee that the resulting motion will still always be natural. Further, while we perform ablation to breakdown the value of individual components, the diversity of the resulting motion we observe remains the aggregation of parts - the new observation mapping, torque and joint limits, stiffness modulation, and so on. Further studies are be needed to truly isolate the exact contribution of individual features, and assign credit further. However, taken all together, our experiments show that the human and animal characters are both able to learn plausible, novel gaits, such as walking sideways as well as to create unique transition behaviors and to accomplish multiple goals simultaneously.

An important strength of ReGAIL is that by building observation and goal states with key relative features, training in a wider range of (global) conditions does not lead to model collapse, thanks to a greater consistency between disparate episodes. With the right observation, we point the discriminator and policy to the critical information needed for accomplishing the imitation task and a specific set of goals under the current circumstances. The result is a strengthening of the task, or goal's execution, as the agent is exposed to a broader set of conditions.

Our work makes the generation of a rich set of motion controllers possible from little kinematic motion data. In the future, we would like to study more accurate anatomical representations, as well as the addition of more biological reward terms, such as penalizing energy usage, hopeful that our agents would then discover efficient gaits that match their morphology.

Acknowledgments

This work was funded by the 80Prime CNRS project : "PaleoMob3D".

REFERENCES

N. Chentanez, M. Müller, M. Macklin, V. Makoviychuk, and S. Jeschke. 2018. Physics-based motion capture imitation with deep reinforcement learning. In *Proceedings of the 11th ACM SIGGRAPH Conference on Motion, Interaction and Games* (Limassol, Cyprus) *(MIG '18)*. Article 1, 10 pages. https://doi.org/10.1145/3274247.3274506

K. Ding, L. Liu, M. van de Panne, and K. Yin. 2015. Learning reduced-order feedback policies for motion skills. In *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation* (Los Angeles, California) *(SCA '15)*. 83–92. https://doi.org/10.1145/2786784.2786802

Y. Feng, X. Xu, and L. Liu. 2023. MuscleVAE: Model-Based Controllers of Muscle-Actuated Characters. In *SIGGRAPH Asia 2023 Conference Papers (SA '23)*. Article 3, 11 pages. https://doi.org/10.1145/3610548.3618137

T. Geijtenbeek, M. van de Panne, and A. F. van der Stappen. 2013. Flexible muscle-based locomotion for bipedal creatures. *ACM Trans. Graph.* 32, 6, Article 206 (nov 2013), 11 pages. https://doi.org/10.1145/2508363.2508399

N. Heess, D. TB, S. Sriram, J. Lemmon, J. Merel, G. Wayne, Y. Tassa, T. Erez, Z. Wang, S. M. A. Eslami, M. Riedmiller, and D. Silver. 2017. Emergence of Locomotion Behaviours in Rich Environments. arXiv:1707.02286 [cs.AI]

J. Ho and S. Ermon. 2016. Generative Adversarial Imitation Learning. *CoRR* abs/1606.03476 (2016). arXiv:1606.03476 http://arxiv.org/abs/1606.03476

A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, and D. Lange. 2020. Unity: A general platform for intelligent agents. *arXiv preprint arXiv:1809.02627* (2020). https://arxiv.org/pdf/1809.02627.pdf

J. T. Kim and S. Ha. 2021. Observation Space Matters: Benchmark and Optimization Algorithm. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. 1527–1534. https://doi.org/10.1109/ICRA48506.2021.9561019

A. Kwiatkowski, E. Alvarado, V. Kalogeiton, C. K. Liu, J. Pettré, M. van de Panne, and M.-P. Cani. 2022. A Survey on Reinforcement Learning Methods in Character Animation. *Computer Graphics Forum* (2022), 1–27. https://doi.org/10.1111/cgf.14504

S. Lee, P. S. Chang, and J. Lee. 2022. Deep Compliant Control. In *ACM SIGGRAPH 2022 Conference Proceedings* (Vancouver, BC, Canada) *(SIGGRAPH '22)*. Article 23, 9 pages. https://doi.org/10.1145/3528233.3530719

S. Lee, S. Lee, Y. Lee, and J. Lee. 2021. Learning a family of motor skills from a single motion clip. *ACM Trans. Graph.* 40, 4, Article 93 (jul 2021), 13 pages. https://doi.org/10.1145/3450626.3459774

H. Y. Ling, F. Zinno, G. Cheng, and M. Van De Panne. 2020. Character controllers using motion VAEs. *ACM Transactions on Graphics* 39, 4 (Aug. 2020). https://doi.org/10.1145/3386569.3392422

X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne. 2018. DeepMimic: Example-guided Deep Reinforcement Learning of Physics-based Character Skills. *ACM Trans. Graph.* 37, 4, Article 143 (July 2018), 14 pages. https://doi.org/10.1145/3197517.3201311

X. B. Peng, G. Berseth, K. Yin, and M. Van De Panne. 2017. DeepLoco: dynamic locomotion skills using hierarchical deep reinforcement learning. *ACM Trans. Graph.* 36, 4, Article 41 (jul 2017), 13 pages. https://doi.org/10.1145/3072959.3073602

X. B. Peng, Y. Guo, L. Halper, S. Levine, and S. Fidler. 2022. ASE: Large-scale Reusable Adversarial Skill Embeddings for Physically Simulated Characters. *ACM Trans. Graph.* 41, 4, Article 94 (July 2022).

X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa. 2021a. AMP: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics* 40, 4, Article 1 (July 2021), 15 pages.

X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa. 2021b. AMP: Adversarial Motion Priors for Stylized Physics-Based Character Control. *ACM Trans. Graph.* 40, 4, Article 1 (July 2021), 15 pages. https://doi.org/10.1145/3450626.3459670

X. B. Peng and M. van de Panne. 2017. Learning locomotion skills using DeepRL: does the choice of action space matter?. In *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation* (Los Angeles, California) *(SCA '17)*. Article 12, 13 pages. https://doi.org/10.1145/3099564.3099567

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. 2017. Proximal Policy Optimization Algorithms. *CoRR* (2017). http://arxiv.org/abs/1707.06347

C. Tessler, Y. Kasten, Y. Guo, S. Mannor, G. Chechik, and X. B. Peng. 2023. CALM: Conditional Adversarial Latent Models for Directable Virtual Characters. In *ACM SIGGRAPH 2023 Conference Proceedings (SIGGRAPH '23)*. https://doi.org/10.1145/3588432.3591541

J. Won, D. Gopinath, and J. Hodgins. 2022. Physics-based character controllers using conditional VAEs. *ACM Trans. Graph.* 41, 4, Article 96 (jul 2022), 12 pages. https://doi.org/10.1145/3528223.3530067

K. Xie, P. Xu, S. Andrews, V. B. Zordan, and P. G. Kry. 2023. Too Stiff, Too Strong, Too Smart: Evaluating Fundamental Problems with Motion Control Policies. *Proc. ACM Comput. Graph. Interact. Tech.* 6, 3, Article 34 (aug 2023), 17 pages. https://doi.org/10.1145/3606935

P. Xu and I. Karamouzas. 2021. A GAN-Like Approach for Physics-Based Imitation Learning and Interactive Character Control. *Proc. ACM Comput. Graph. Interact. Tech.* 4, 3, Article 44 (sep 2021), 22 pages. https://doi.org/10.1145/3480148

P. Xu, X. Shang, V. Zordan, and I. Karamouzas. 2023a. Composite Motion Learning with Task Control. *ACM Trans. Graph.* 42, 4, Article 93 (jul 2023), 16 pages. https://doi.org/10.1145/3592447

P. Xu, K. Xie, S. Andrews, P. G. Kry, M. Neff, M. Mcguire, I. Karamouzas, and V. Zordan. 2023b. AdaptNet: Policy Adaptation for Physics-Based Character Control. *ACM Trans. Graph.* 42, 6, Article 177 (dec 2023), 17 pages. https://doi.org/10.1145/3618375

H. Yao, Z. Song, B. Chen, and L. Liu. 2022. ControlVAE: Model-Based Learning of Generative Controllers for Physics-Based Characters. *ACM Transactions on Graphics* 41, 6 (Nov. 2022), 1–16. https://doi.org/10.1145/3550454.3555434

W. Yu, G. Turk, and C. K. Liu. 2018. Learning Symmetric and Low-Energy Locomotion. *ACM Trans. Graph.* 37, 4, Article 144 (jul 2018), 12 pages. https://doi.org/10.1145/3197517.3201397

K. Zolna, S. Reed, A. Novikov, S. G. Colmenarejo, D. Budden, S. Cabi, M. Denil, N. d. Freitas, and Z. Wang. 2021. Task-Relevant Adversarial Imitation Learning. In *Proceedings of the 2020 Conference on Robot Learning (Proceedings of Machine Learning Research, Vol. 155)*, Jens Kober, Fabio Ramos, and Claire Tomlin (Eds.). PMLR, 247–263. https://proceedings.mlr.press/v155/zolna21a.html