LEARNING SYMMETRIC LOCOMOTION USING CUMULATIVE FATIGUE FOR REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Modern deep reinforcement learning (DRL) methods allow simulated characters to learn complex skills such as locomotion from scratch. However, without further exploitation of domain-specific knowledge, such as motion capture data, finite state machines or morphological specifications, physics-based locomotion generation with DRL often results in unrealistic motions. One explanation for this is that present RL models do not estimate biomechanical effort; instead, they minimize instantaneous squared joint actuation torques as a proxy for the actual subjective cost of actions. To mitigate this discrepancy in a computationally efficient manner, we propose a method for mapping actuation torques to subjective effort without simulating muscles and their energy expenditure. Our approach is based on the Three Compartment Controller model, in which the relationships of variables such as maximum voluntary joint torques, recovery, and cumulative fatigue are present. We extend this method for sustained symmetric locomotion tasks for deep reinforcement learning using a Normalized Cumulative Fatigue (NCF) model. In summary, in this paper we present the first RL model to use biomechanical cumulative effort for full-body movement generation without the use of any finite state machines, morphological specification or motion capture data. Our results show that the learned policies are more symmetric, periodic and robust compared to methods found in previous literature.

1 INTRODUCTION

It is a long standing task in computer animation to make characters walk on their own. In this context, Deep Reinfocement Learning (DRL) has become a promising method for automatic generation of movement controls for interactive, physics-based characters. However, in many cases the resulting motions are still not perceived as natural (Schulman et al., 2017). A common approach to mitigate this is to use motion capture or animation data (Peng et al., 2018; Bergamin et al., 2019; Won et al., 2020; Peng et al., 2021). Nevertheless, such approaches are limited to characters and movements to which data is readily available. Furthermore, obtaining qualitatively good data is oftentimes expensive, and many biomechanical constrains that are implicit in captured motions are not preserved during editing and retargeting – which is often required when data is limited. Another method for improving motion quality is to optimize for movement characteristics that shape the motion such as symmetric gait properties (Yu et al., 2018; Abdolhosseini et al., 2019) or minimal energy consumption and task goals. While such methods overcome the need of motion capture data, the absence of biomechanical constraints still may lead to unwanted behaviour and unnatural torque patterns. Another group of methods that have emerged come from bio-mechanical literature, which include musculoskeletal models and other forms of biological constraints. Previous works (Wang et al., 2012; Geijtenbeek et al., 2013; Lee et al., 2014) in this direction have explored biomimetic muscles and tendons to simulate a variety of human and animal motions. However, such muscle-based methods are usually computationally expensive, especially under a reinforcement learning framework (Kidziński et al., 2018). In this research we work towards developing a cumulative fatigue reward based on biomechanical literature to account for a computationally efficient way to include motion constraints that are implicit in articulated figures driven by musculotendon units, in the context of locomotion. To improve on quality we further incorporate movement characteristics, such as gait symmetry enforcement methods by Abdolhosseini et al. (2019) and Yu et al. (2018).

Contributions. In this paper we present the first RL model to use biomechanical cumulative effort for full-body movement generation. We derive a Normalized Cumulative Fatigue (NFC) model suitable for reinforcement learning based on the Three Compartment Controller (3CC) model by Xia & Frey Law (2008) and show that both models are equivalent under the assumption of sustained dynamic load conditions but that the 3CC model fails when applied to pre-existing benchmark environments without further hyper-parameter-tuning of the environment itself. Furthermore, the fatigue reward derived from our model more accurately reflects the embodied biomechanical nature of a simulated character when compared to a reward based on instantaneous torque (Yu et al., 2018; Abdolhosseini et al., 2019; Schulman et al., 2017). We apply the cumulative fatigue model to a simulated humanoid for learning sustained symmetric locomotion and show that our method is robust and can generate more relaxed, natural and symmetric locomotion especially in complex environments – without the need of motion capture data, finite state machines or morphological specifications, as well as no further hyper-parameter-tuning of possible pre-existing environments. Additionally, the simplification from 3CC to NCF allows the method to be more easily adaptable to arbitrary characters that may not exhibit biologically accurate properties.

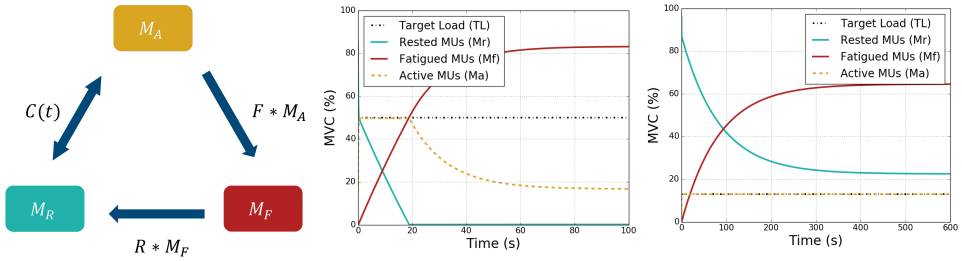
2 Related Work

Recent developments in DRL have seen significant progress in solving high-dimensional continuous control problems. For example, Schulman et al. (2015a) have proposed Trust Region Policy Optimization (TRPO) and show that this method can be used to generate biped locomotion in a 2D planar space. Later, by combining TRPO with Generalized Advantage Estimation, Schulman et al. (2015b) have extended their work for their humanoid locomotion task to three dimensions. Afterwards, they have proposed Proximal Policy Optimization (PPO), which further improves the data efficiency of the algorithm (Schulman et al., 2017). However, the resulting movements oftentimes still look jerky and unnatural. A common way to overcome these issues, is to exploit domain specific knowledge in various forms (Ramamurthy et al., 2019):

Imitation Learning. Oftentimes, reference motion is used in this regard. Peng et al. (2017) introduce a two-level hierarchical controller to generate locomotion: the low-level controller is learned by mimicking the reference locomotion data; the high-level controller is acting as a planner in order to respond to environment changes. However, this method is not capable of highly dynamic motions. Peng et al. (2018) address this issue and achieve significantly more natural-looking motions using imitation learning. More recently, they have extended their method with generative adversarial imitation learning (Peng et al., 2021). Other works in this direction include mimicking various features over a large dataset of movements with RL (Bergamin et al., 2019) or learning a mixture of experts models for various movements (Won et al., 2020). However, all these methods require readily available motion capture data as a prerequisite for training.

Optimizing Movement Characteristics. Instead of using reference data as prior knowledge, another option is to exploit the characteristics of specific types of motions that shall be generated using hand-crafted features. In this regard, Yu et al. (2018) exploit symmetry property of locomotion and propose the mirror symmetry loss. They combine it with energy optimization and add an external force that acts as a virtual assistant to learn symmetric locomotion from scratch. Abdolhosseini et al. (2019) emphasize the core idea behind the mirror symmetry loss, called *symmetric policy*, and analyze its performance in terms of locomotion symmetry by combining it with DRL in a variety of ways to produce symmetric gaits. They also show that symmetry enforcement methods improve gait symmetry in general, but cannot guarantee a symmetric gait. Furthermore, while such methods overcome the need for motion capture data, the absence of bio-mechanical constraints still leads to unwanted behaviour and less natural torque patterns.

Musculoskeletal Models. A more biologically-accurate approach for movement synthesis involves musculoskeletal models. Previous works (Taga, 1995; Anderson & Pandy, 2001; Geyer & Herr, 2010; Ackermann & van den Bogert, 2012; Ijspeert et al., 2007; Maufroy et al., 2008; Thelen et al., 2003) in biomechanics have developed musculoskeletal models that use biomimetic muscles and tendons to simulate a variety of human and animal motions. Controlling a muscle-based virtual characters has also been explored in computer animation – from upper body movements (Lee & Terzopoulos, 2006; Lee et al., 2009; 2018), to hand manipulation (Tsang et al., 2005; Sueda et al.,



(a) 3CC Model (b) $TL = 0.5$ (c) $TL = 0.13 < \frac{R}{F+R}$

Figure 1: (a) Behavior of the 3CC model at (b) 50% Maximum Voluntary Contraction (MVC) and at (c) 13% MVC $< \frac{R}{F+R}$. Note how the full load cannot be held any longer after 20s in (b) (yellow dashed line), while the load in (c) can be held indefinitely. $C(t)$ denotes the feedback-controller term, R and F the rest and fatigue coefficients, respectively.

2008), and full-body locomotion (Wang et al., 2012; Geijtenbeek et al., 2013; Lee et al., 2014). However, such muscle-based methods are usually computationally expensive, especially under a reinforcement learning framework (Kidziński et al., 2018). To address this issue, Jiang et al. (2019) have proposed a technique to transform an optimal control problem formulated in the muscle actuation space to an equivalent problem in the joint-actuation space by training the model with control signals obtained from the muscle actuation space. The result shows that as long as the model can reflect the underlying biomechanical properties, it is not necessary to model muscle and tendon details explicitly in order to generate more realistic motions. However, a disadvantage of this work is that they need to learn the mapping from the muscle-based actuation space to the torque-based space using reference data.

Biomechanical Cumulative Fatigue. Muscle fatigue is the failure to maintain the required or expected force (Edwards, 1981). In contrast to instantaneous fatigue, which does not take the endurance time into account, biomechanical fatigue assumes the fatigue to accumulate over time – i.e. the longer a task is done, the more fatiguing it becomes. Muscle fatigue is task-related and can vary across muscles and joints (Xia & Frey Law, 2008; Imbeau et al., 2006; Enoka & Duchateau, 2008; Jang et al., 2017; Frey Law & Avin, 2010; Frey-Law et al., 2012), which partially explains the challenging nature of representing muscle fatigue analytically. In this regard, Liu et al. (2002b) have proposed a motor unit (MU)-based fatigue model which uses three muscle activation states to estimate perceived biomechanical fatigeu: resting, activated and fatigued. The model is able to predict fatigue at static load conditions but fails at submaximal or dynamic conditions. Xia & Frey Law (2008) have proposed a Three-Compartment Controller (3CC) model which improves upon the model of Liu et al. (2002b) for dynamic load conditions by introducing a feed-back controller term between the active and rest MU-states based on torque without the need of explicit modeling of muscle actuators. The 3CC model, as a torque-based model for modeling muscle fatigue and recovery, has already shown effectiveness in motion analysis (Jang et al., 2017) and synthesis (Cheema et al., 2020). The follow-up work (Looft et al., 2018) has been successfully used in upper body motion synthesis under a DLR framework by Cheema et al. (2020) without any motion capture data for mid-air interaction analysis and synthesis. Inspired by their work, we extend it to full-body locomotion generation on arbitrary pre-existing characters and environments.

3 Preliminaries: Fatigue Modeling

Previous works in computer animation, robotics and standard RL (Yu et al., 2018; Abdolhosseini et al., 2019; Rajamäki & Hämäläinen, 2017; Peng et al., 2017) use instantaneous squared joint torques as a simple measurement to minimize the effort of a given task. However, such a measure is not very biologically accurate as it does not take the endurance time into account and thus the increasing amount of perceived fatigue the longer the task is sustained.

The Three-Compartment Controller (3CC) model (Xia & Frey Law, 2008) is a cumulative fatigue model that assumes motor units (MUs) to be in one of three possible states: 1) *active* (M_A) – MUs

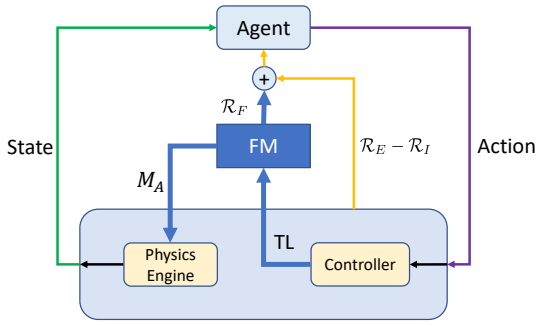


Figure 2: System overview.

contributing to the task, 2) *fatigued* (M_F) – MUs without activation, and 3) *resting* (M_R) – inactive MUs not required for the task. These are usually expressed as a percentage of *maximum voluntary contraction* (*%MVC*), which can practically be expressed as percentage of maximum voluntary force (*%MVF*) or torque (*%MVT*). Additionally, control theory is used to obtain behaviour matching muscle physiology, i.e. active MUs' force production should decay (fatigue) over time when enough constant force is used. Such a cumulative fatigue model thus gives us a more accurate representation of perceived fatigue. This is expressed by the following system of equations:

$$\frac{\partial M_A}{\partial t} = C(t) - F \cdot M_A \tag{1a}$$

$$\frac{\partial M_R}{\partial t} = -C(t) + R \cdot M_F \tag{1b}$$

$$\frac{\partial M_F}{\partial t} = F \cdot M_A - R \cdot M_F \tag{1c}$$

Where F and R denote the fatigue and recovery coefficients. $C(t)$ is a bounded proportional controller in order to produce the required force, i.e. target load (TL) by controlling the size of M_A and M_R:

$$C(t) = \begin{cases} L_D \cdot (TL - M_A) & \text{if } M_A < TL \text{ and } M_R > TL - M_A \\ L_D \cdot M_R & \text{if } M_A < TL \text{ and } M_R \leq TL - M_A \\ L_R \cdot (TL - M_A) & \text{if } M_A \geq TL \end{cases} \tag{2}$$

L_D and L_R are muscle force development, and relaxation factors, which describe the sensitivity towards the target load (Xia & Frey Law, 2008). The behavior of the 3CC model at different load conditions (TL) can be seen in Fig. 1. If the conditions cannot be matched any longer due to high fatigue and not enough rested MUs available, M_A starts to decline (Fig. 1b). From Eq. 2 we can conclude that the target load can be held iff $M_A + M_R \geq TL$. Liu et al. (2002a) have shown that the lower bound of $M_A + M_R$ is $\frac{R}{F+R}$, i.e. the target load TL can be held indefinitely iff $TL \leq \frac{R}{F+R}$ (Fig. 1c) which results in $TL = M_A$.

4 Method

In this section, we present our approach for biomechanical fatigue-based locomotion synthesis. An overview of our system can be seen in Fig. 2. A reward based on cumulative fatigue $\mathcal{R}_F$ is added in addition to the existing environment rewards $\mathcal{R}_E$ in benchmark models. If a reward $\mathcal{R}_I$ based on instantaneous joint squared torques exists, it is replaced with the fatigue reward.

4.1 Cumulative Fatigue for Sustained Locomotion

The 3CC model describes the whole dynamic of motor activation, fatigue and rest. However, we are mostly interested in the fatigue compartment, which is described by Eq. 1c, for constructing the fatigue reward function of the locomotion tasks. For that we first discretize and rearrange Eq. 1c:

$$\frac{M_F^{(i+1)}}{\Delta t F} = (1 - \Delta t R) \cdot \frac{M_F^{(i)}}{\Delta t F} + M_A^{(i)}, (i \in \mathcal{N}) \tag{3}$$

With $\Delta t = t_{i+1} - t_i$ and $M_F^{(0)} = 0$. (i) and $(i+1)$ are the abbreviations of consecutive timestamps t_i and t_{i+1}, respectively. The fatigue coefficient F can be eliminated by substituting $f^{(i)} = \frac{R}{F} M_F^{(i)}$, and M_A can be set to $M_A = TL$ due to formulating locomotion as a sustained task, which results in:

$$f^{(i+1)} = (1 - \Delta t R) \cdot f^{(i)} + \Delta t R \cdot TL^{(i)} \tag{4}$$

We call $f^{(i)}$ Normalized Cumulative Fatigue (NCF). Since the difference between original cumulative fatigue $M_F^{(i)}$ and NCF $f^{(i)}$ is just a scaling factor $\frac{R}{F}$. The advantage of this is that F does not need to be defined explicitly reducing the hyper-parameter space. Furthermore, since locomotion can be executed for a considerably long time without fatiguing, we assume the extreme case that there are always sufficient non-fatigued motor units ($M_A + M_R$) in the 3CC model to produce the desired target load (TL). In this case TL can be set to $TL = M_A$.

To make sure that $TL \leq \frac{R}{F+R}$, we assume $\boldsymbol{\tau}_{max} \cdot \frac{F+R}{R} \leq MVT$, where $\boldsymbol{\tau}_{max}$ is the maximum torque $\boldsymbol{\tau}$ allowed at a joint in the simulation environment with $TL = \frac{\tau}{\tau_{max}} \geq \frac{\tau}{MVT}$. This is in contrast to the original 3CC model and Cheema et al. (2020), where $TL = \frac{\tau}{\tau_{max}} = \frac{\tau}{MVT}$. This change allows us to plug-in our method to pre-existing environments without explicitly having to fine-tune specific MVTs for each joint or check for specific L_D, L_R and F values for the system to function and hold true, as these are canceled out. In the following we describe how the NCF is used to compute the fatigue reward for RL.

4.2 Fatigue Reward for Reinforcement Learning

In contrast to Cheema et al. (2020) who use the difference between M_A and TL for the reward signal, we directly take the reformulation of the fatigue function (NCF) as a reward signal, since we assume $TL = M_A$ in our reformulation. Akin to them, we model two fatigue functions for each degree-of-freedom (DoF) roughly corresponding to antagonistic muscle pairs. We adopt this approach for all joints of the simulated character. Given a simulated character with n DoF, the magnitude of torque at axis j in "positive" and "negative" directions are denoted as $\tau_{j,+}$ and $\tau_{j,-}$ ($j \in [1, n]$), respectively. When $\tau_{j,+} \geq 0$, $\tau_{j,-} = 0$, and vice versa. The target load at axis j can be expressed by $TL_{j,+} = \frac{\tau_{j,+}}{\tau_{j,max}}$ and $TL_{j,-} = \frac{\tau_{j,-}}{\tau_{j,max}}$, where $\tau_{j,max}$ is maximum torque magnitude at axis j. Note that $\tau_{j,max}$ does not necessarily equal to MVT at axis j. It is just the maximum torque magnitude that actuator can apply on axis j. Then the NCF at axis j at time t_{i+1} in "positive" and "negative" directions are

$$f_{j,+}^{(i+1)} = (1 - \Delta t R_{j,+}) \cdot f_{j,+}^{(i)} + \Delta t R_{j,+} \cdot TL_{j,+}^{(i)} \tag{5a}$$

$$f_{j,-}^{(i+1)} = (1 - \Delta t R_{j,-}) \cdot f_{j,-}^{(i)} + \Delta t R_{j,-} \cdot TL_{j,-}^{(i)} \tag{5b}$$

respectively. The NCF of the simulated character with n DoF at time t_i can be expressed by a vector:

$$\boldsymbol{f}^{(i)} = [f_{1,+}^{(i)}, f_{1,-}^{(i)}, f_{2,+}^{(i)}, f_{2,-}^{(i)}, \cdots, f_{n,+}^{(i)}, f_{n,-}^{(i)}]^T \tag{6}$$

To penalize the use of excessive strength we use the L2 norm to formulate our fatigue reward:

$$\mathcal{R}_F = -w_F \|\boldsymbol{f}\|_2, \tag{7}$$

where $w_F \geq 0$ is the fatigue reward weight. This reward is then used to replace the instantaneous squared joint torque effort of previous work. We set $w_F = 1$ and the recovery coefficient to $R = 0.2$ for all joints. R was chosen such that M_F does not saturate by the end of the training episode. With this we can take advantage of the fatigue accumulation.

4.3 Symmetry Enforcement using Mirror Symmetry Loss

We adopt the mirror symmetry loss $\mathcal{L}_{sym}$ first proposed by Yu et al. (2018) to enforce a more symmetric gait:

$$\mathcal{L}_{sym}(\theta) = \sum_{t=1}^{T} \|\pi_\theta(s_t) - \mathcal{M}_a(\pi_\theta(\mathcal{M}_s(s_t)))\|^2, \tag{8}$$

T denotes the episode length. Given a state space S and an action space A, a policy $\pi_\theta : S \to A$ is considered symmetric iff $\forall s \in S, \pi_\theta(\mathcal{M}_s(s)) = \mathcal{M}_a(\pi_\theta(s))$, with $s \in S$ and $a \in A$. $\mathcal{M}_s : S \to S$

and $\mathcal{M}_a : A \rightarrow A$ are here state and action mirroring functions with $\mathcal{M}_a(a)$ being the mirror action of action a, and $\mathcal{M}_s$ the mirror state of state s, respectively. Yu et al. (2018) optimize this as an auxiliary loss in addition to the default PPO by Schulman et al. (2017):

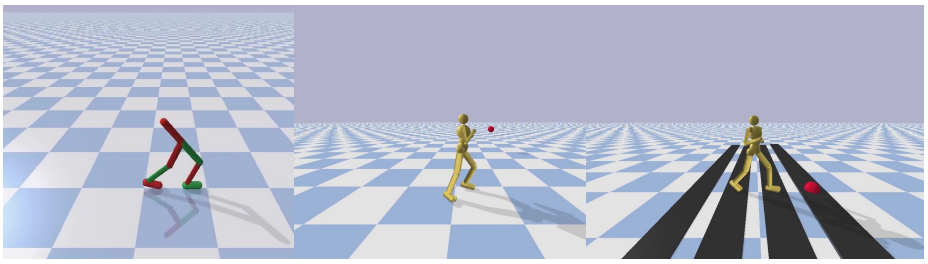
$$\pi_\theta = \arg\min_\theta \mathcal{L}_{PPO}(\theta) + w_{sym}\mathcal{L}_{sym}(\theta), \tag{9}$$

where w_{sym} is a scalar hyper-parameter used to balance the gait symmetry loss with the standard policy optimization loss which aims to maximize the original objective.

5 Experiments

We build our experiments upon the open-source implementation of Abdolhosseini et al. (2019) who use PPO (Schulman et al., 2017) as their base algorithm in addition to their environment rewards ($\mathcal{R}_E$ in Fig. 2), as well as the symmetry loss proposed by Yu et al. (2018) as an additional symmetry enforcement method (Abdolhosseini et al., 2019). All hyper-parameters, existing rewards and losses are kept the same except that the instantaneous squared torque "low-energy reward" (Yu et al., 2018; Abdolhosseini et al., 2019) ($\mathcal{R}_I$ in Fig. 2) is replaced with our cumulative fatigue reward $\mathcal{R}_F$. Note, that we assume that Abdolhosseini et al. (2019) already did a hyper-parameter search for the symmetry enforcement methods and have published their best results in their comparative symmetry enforcement study (Abdolhosseini et al., 2019), which we build upon. For the comparison against Cheema et al. (2020), we assume $F = 10 \cdot R$ and $L_D = L_R = 10$, based on average values from biomechanical literature (Looft et al., 2018; Xia & Frey Law, 2008).

5.1 Environments



(a) Walker2D (b) Walker3D (c) Stepper

Figure 3: Locomotion controller trained for different environments. (a) Biped walking. (b) Humanoid running. (c) Humanoid stepping on stairs.

In our experiments we evaluate our method in three different locomotion environments, adopted from Abdolhosseini et al. (2019) (Fig. 3). We use PyBullet as the physics engine and PyTorch as the learning framework.

Walker2D The character contains 6 joints (hips, knees, ankles) with one DoF. Hence, the action space consists of a 6D vector, where each element represents the normalized torque (between -1 and 1 per DoF) applied to a specific joint. The observation space is 22D and consists of root information (root z-coordinate, x and y direction vector, root velocity, roll, and pitch), joint angles, joint angular velocities, and binary foot contact information.

Walker3D The character has 21-DoF corresponding to abdomen (3 DoF), hips (3 DoF), knees (1 DoF), ankles (1 DoF), shoulders (3 DoF), and elbows (1 DoF). The observation space is 52D, consisting of root information (roll, pitch and root velocity), joint angles (21D), joint angular velocities (21D), binary foot contact information (2D), and target position (3D).

Stepper The stepper environment uses the same character as Walker3D, having the same action space as Walker3D, i.e. 21D. The task for the character is to navigate terrain consisting of a sequence of stepping blocks, which are generated consecutively. The character receives the positions of two upcoming blocks as an offset (x,y,z) of the character root frame. The observation space is 55D, similar to Walker3D with additional 3D for the second stepping block target position.

5.2 EVALUATION METRICS

In order to evaluate the effect of our method on locomotion symmetry, we utilize four gait symmetry metrics:

Number of successful models. To determine whether a model was successful or not, we compute if the trained model can walk a certain number of strides, akin to Abdolhosseini et al. (2019). For that we use 20 strides. We train 20 models from 20 different random seeds for each environment and benchmark method. The results can be seen in Table 1. The following metrics are then computed over these successful models:

Normalized Squared Torques (ST). A simple measure for effort is the squared instantaneous torque applied at each time step. The Normalized Squared Torque for one episode is defined as:

$$ST = \frac{1}{T}\sum_{i=1}^{T}\frac{1}{n}\sum_{j=1}^{n}\|\frac{\tau_j^i}{\tau_{j,max}^i}\|^2 \tag{10}$$

In our evaluation, we evaluate at most 21 episodes. For each episode, the agent needs to at least produce 20 strides and we choose the first 1500 steps. Then we average squared torque cost for each episode as the squared torque cost for the model.

Cumulative Fatigue (CF). The cumulative fatigue is calculated using the fatigue term derived from NCF, which is described in Section 4 Equation 4 and 7 (as a qualitative measure, we omit the weight term $-w_f$). The Cumulative Fatigue for one episode is defined as:

$$CF = \frac{1}{T}\sum_{i=1}^{T}\|\boldsymbol{f}^{(i)}\| \tag{11}$$

Vectorized Symmetry Index (VSI). A widely-used metric in the biomechanics literature is the Robinson et al. (1987) Symmetry Index (SI), which uses scalar features of the left and right side of the body to determine symmetry. Yu et al. (2018) choose to use the average torques as a feature. However, their metric does not consider torque differences between two sides of the same joint, and neglects the direction of torque. To overcome these drawbacks, we propose VSI, which accepts vectorized features:

$$VSI = \frac{2\|\boldsymbol{X}_r - \boldsymbol{X}_l\|_2}{\|\boldsymbol{X}_l\|_2 + \|\boldsymbol{X}_r\|_2} \times 100\% \tag{12}$$

With $\boldsymbol{X}_r = \frac{1}{T}\sum_{t=1}^{T}(\boldsymbol{\tau}_t^{r+}, \boldsymbol{\tau}_t^{r-})$ and $\boldsymbol{X}_l = \frac{1}{T}\sum_{t=1}^{T}(\boldsymbol{\tau}_t^{l+}, \boldsymbol{\tau}_t^{l-})$, respectively, where $\boldsymbol{\tau}_t^+$ represent the positive torque directions and $\boldsymbol{\tau}_t^-$ the negative at time t. r and l denote the right and the left side, respectively.

Phase-Portrait Index (PPI). VSI only measures the torque symmetry. However, phase-portraits can be used to investigate gait symmetry. A phase-portrait is a 2D scatter plot of which the x and the y axis represent rotation angle and angular velocity of a given joint, usually over a single gait cycle. If the gait is asymmetric the phase-portraits of the two sides will not fully overlap (see Fig. **??**). To quantify this information, we adopt the phase-portrait index (PPI) proposed by Abdolhosseini et al. (2019), which is defined as:

$$PPI = \frac{1}{c}\min_s\sum_{t=0}^{c-1}\|q_t^r - q_{t+s}^l\|_1 + \|\dot{q}_t^r - \dot{q}_{t+s}^l\|_1 \tag{13}$$

Where c is the length of a gait cycle, q_t^r and $\dot{q}_t^r$ are the normalized right joint position and velocity at time t. Similarly, q_{t+s}^l is the normalized left joint position at time $t+s$ mod c, as the elements that are shifted beyond the last position are reintroduced at the beginning.

Spectral Entropy (SE). Spectral entropy is commonly used tool to measure the uncertainty of the frequency of a given signal. In this work, we use spectral entropy to measure the periodicity of locomotion:

$$SE = -\sum_{f=0}^{N-1} f \ln \left(\frac{\|\hat{x}(f)\|^2}{\sum_{f'=0}^{N-1} \|\hat{x}(f')\|^2} \right) \tag{14}$$

$\hat{x}(f)$ denotes the Fourier transform of a signal. The SE can be used to measure the periodicity of a given signal since, in general, the more periodicity the signal has, the less uncertainty the signal frequency has. We compute the SE of the left and right hip joint rotation angle and average between the two signals.

5.3 Ablation Studies

Table 1: Number of models that successfully learned locomotion (could walk 20 or more strides) per method and environment.

Method	Environment		
	Walker2D	Walker3D	Stepper
PPO (Schulman et al., 2017)	20/20	20/20	11/20
PPO+3CC (Cheema et al., 2020)	20/20	0/10	0/10
PPO+NCF	20/20	18/20	13/20
SYMM (Yu et al., 2018)	20/20	20/20	17/20
SYMM+3CC	20/20	0/10	2/10
SYMM+NCF (Ours)	20/20	20/20	18/20

The following results are computed over these successful models. Note, how in contrast to ours, the 3CC models fail to produce locomotion movements for the majority of models in the more difficult environments, due to them fatiguing too much. This is shown in Tables 3 and 2, where the 3CC models produce the least amount of torque but the number of successful models (Table 1). This is because the set MVTs in the pre-defined environments are set too low for the model to work properly without changing the environments themselves. Due to this we were unable to add the results in the given plots but included them in the tables in the appendix and in the supplementary video for qualitative results. Our overall results show that our proposed model produces more relaxed movements (Fig. 4) in terms of torque actuation than the baseline models, while still finishing the tasks competitively with the baselines or even better in terms of symmetry and periodicity (Fig. 5).

Table 2: Normalized Squared Torques (median[std]): Lower numbers are better.

Method	Environment		
	Walker2D	Walker3D	Stepper
PPO (Schulman et al., 2017)	0.423 [0.026]	0.295 [0.026]	0.309 [0.040]
PPO+3CC	0.220 [0.010]	NaN	NaN
PPO+NCF	0.398 [0.059]	0.243 [0.032]	0.291 [0.044]
SYMM (Yu et al., 2018)	0.377 [0.054]	0.2254 [0.034]	0.202 [0.027]
SYMM+3CC	**0.194** [0.030]	NaN	**0.078** [0.003]
SYMM+NCF (Ours)	0.362 [0.028]	**0.179** [0.025]	0.188 [0.031]

6 Limitations and Future Work

While our cumulative fatigue reward improves on existing benchmark methods in terms of relaxedness, symmetry and periodicity, the 3CC model, which it is based on, is not task- and joint-agnostic (Imbeau et al., 2006; Frey Law & Avin, 2010; Frey-Law et al., 2012) and thus our model can be

Table 3: Cumulative Fatigue (median[std]): Lower numbers are better.

Method	Environment		
	Walker2D	Walker3D	Stepper
PPO (Schulman et al., 2017)	0.882 [0.040]	1.639[0.091]	1.715 [0.156]
PPO+3CC	0.623 [0.025]	NaN	NaN
PPO+NCF	0.847 [0.080]	1.403 [0.146]	1.606 [0.188]
SYMM (Yu et al., 2018)	0.847 [0.076]	1.448 [0.134]	1.390 [0.137]
SYMM+3CC	**0.587** [0.043]	NaN	**0.814** [0.017]
SYMM+NCF (Ours)	0.817 [0.040]	**1.257** [0.121]	1.301 [0.161]

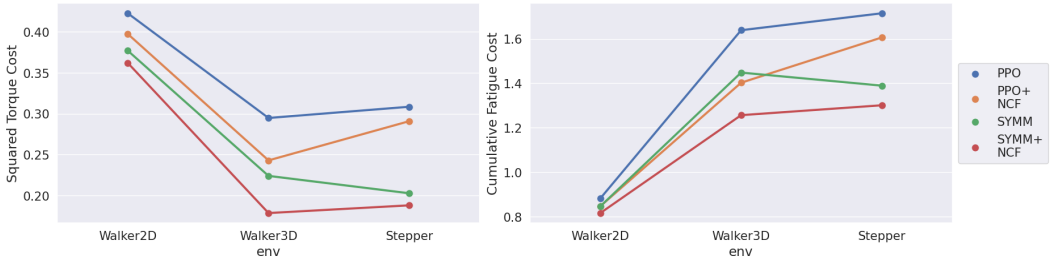


Figure 4: Median torque and cumulative fatigue cost in different environments.

improved by optimizing R for each joint based on a sensitivity analysis for various tasks, including locomotion. Additionally, the 3CC model has initially been designed for isometric tasks and is not verified for isokinetic tasks such as locomotion from the viewpoint of biomechanics (Xia & Frey Law, 2008; Looft et al., 2018). To conclude if NCF is valid for locomotion from a biomechanical perspective, a separate study with real participants needs to be conducted. Furthermore, in this paper we mostly focus on symmetry, however extensions to other methods and features have yet to be done.

7 Conclusion

In this paper we presented, to the best of our knowledge, the first work to use biomechanical cumulative fatigue for full-body continuous control in complex environments. We derived a Normalized Cumulative Fatigue (NCF) model from the 3CC model and applied it to a humanoid locomotion in pre-existing simulated environments. The agents are trained on a locomotion task. Our results show that our method generally performs better than existing methods in terms of relaxedness, locomotion symmetry and periodicity. Compared to the 3CC model, our NCF model is easier to use, because it does not need additional hyper-parameter tuning of the environment and cancels out hyper-parameters needed in the original 3CC implementation, such as F, L_D, L_R and MVC for every joint.

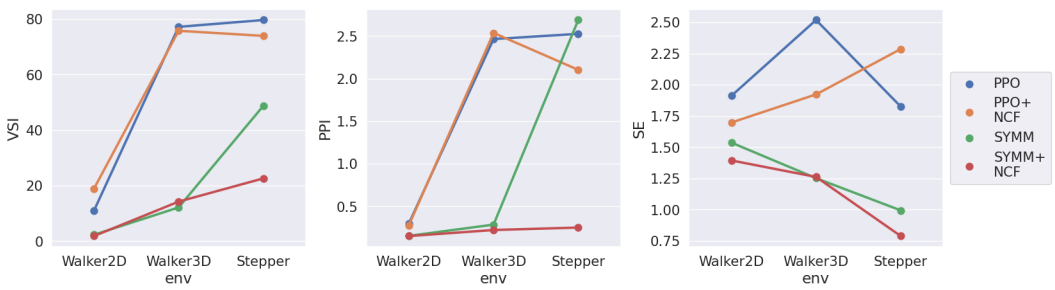


Figure 5: Median VSI, PPI and SE in different environments.

REFERENCES

Farzad Abdolhosseini, Hung Yu Ling, Zhaoming Xie, Xue Bin Peng, and Michiel Van de Panne. On learning symmetric locomotion. *Proceedings - MIG 2019: ACM Conference on Motion, Interaction, and Games*, 2019. doi: 10.1145/3359566.3360070.

Marko Ackermann and Antonie J van den Bogert. Predictive simulation of gait at low gravity reveals skipping as the preferred locomotion strategy. *Journal of biomechanics*, 45(7):1293–1298, 2012.

Frank C Anderson and Marcus G Pandy. Dynamic optimization of human walking. *J. Biomech. Eng.*, 123(5):381–390, 2001.

Kevin Bergamin, Simon Clavet, Daniel Holden, and James Richard Forbes. Drecon: data-driven responsive control of physics-based characters. *ACM Transactions On Graphics (TOG)*, 38(6): 1–11, 2019.

Noshaba Cheema, Laura A. Frey-Law, Kourosh Naderi, Jaakko Lehtinen, Philipp Slusallek, and Perttu Hämäläinen. Predicting Mid-Air Interaction Movements and Fatigue Using Deep Reinforcement Learning. pp. 1–13, 2020. doi: 10.1145/3313831.3376701.

Richard HT Edwards. Human muscle function and fatigue. In *Ciba Found Symp*, volume 82, pp. 1–18. Wiley Online Library, 1981.

Roger M Enoka and Jacques Duchateau. Muscle fatigue: what, why and how it influences muscle function. *The Journal of physiology*, 586(1):11–23, 2008.

Laura A Frey Law and Keith G Avin. Endurance time is joint-specific: a modelling and meta-analysis investigation. *Ergonomics*, 53(1):109–129, 2010.

Laura A Frey-Law, John M Looft, and Jesse Heitsman. A three-compartment muscle fatigue model accurately predicts joint-specific maximum endurance times for sustained isometric tasks. *Journal of biomechanics*, 45(10):1803–1808, 2012.

Thomas Geijtenbeek, Michiel Van De Panne, and A Frank Van Der Stappen. Flexible muscle-based locomotion for bipedal creatures. *ACM Transactions on Graphics (TOG)*, 32(6):1–11, 2013.

Hartmut Geyer and Hugh Herr. A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on neural systems and rehabilitation engineering*, 18(3):263–273, 2010.

Auke Jan Ijspeert, Alessandro Crespi, Dimitri Ryczko, and Jean-Marie Cabelguen. From swimming to walking with a salamander robot driven by a spinal cord model. *science*, 315(5817):1416–1420, 2007.

Daniel Imbeau, Bruno Farbos, et al. Percentile values for determining maximum endurance times for static muscular work. *International Journal of Industrial Ergonomics*, 36(2):99–108, 2006.

Sujin Jang, Wolfgang Stuerzlinger, Satyajit Ambike, and Karthik Ramani. Modeling cumulative arm fatigue in mid-air interaction based on perceived exertion and kinetics of arm motion. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 3328–3339, 2017.

Yifeng Jiang, Tom Van Wouwe, Friedl De Groote, and C Karen Liu. Synthesis of biologically realistic human motion using joint torque actuation. *ACM Transactions On Graphics (TOG)*, 38 (4):1–12, 2019.

Łukasz Kidziński, Sharada Prasanna Mohanty, Carmichael F Ong, Zhewei Huang, Shuchang Zhou, Anton Pechenko, Adam Stelmaszczyk, Piotr Jarosik, Mikhail Pavlov, Sergey Kolesnikov, et al. Learning to run challenge solutions: Adapting reinforcement learning methods for neuromusculoskeletal environments. In *The NIPS'17 Competition: Building Intelligent Systems*, pp. 121–153. Springer, 2018.

Seunghwan Lee, Ri Yu, Jungnam Park, Mridul Aanjaneya, Eftychios Sifakis, and Jehee Lee. Dexterous manipulation and control with volumetric muscles. *ACM Transactions on Graphics (TOG)*, 37(4):1–13, 2018.

Sung-Hee Lee and Demetri Terzopoulos. Heads up! biomechanical modeling and neuromuscular control of the neck. In *ACM SIGGRAPH 2006 Papers*, pp. 1188–1198. 2006.

Sung-Hee Lee, Eftychios Sifakis, and Demetri Terzopoulos. Comprehensive biomechanical modeling and simulation of the upper body. *ACM Transactions on Graphics (TOG)*, 28(4):1–17, 2009.

Yoonsang Lee, Moon Seok Park, Taesoo Kwon, and Jehee Lee. Locomotion control for many-muscle humanoids. *ACM Transactions on Graphics (TOG)*, 33(6):1–11, 2014.

Jing Z. Liu, Robert W. Brown, and Guang H. Yue. A dynamical model of muscle activation, fatigue, and recovery. *Biophysical Journal*, 82(5):2344–2359, 2002a. ISSN 00063495. doi: 10.1016/S0006-3495(02)75580-X. URL http://dx.doi.org/10.1016/S0006-3495(02)75580-X.

Jing Z Liu, Robert W Brown, and Guang H Yue. A dynamical model of muscle activation, fatigue, and recovery. *Biophysical journal*, 82(5):2344–2359, 2002b.

John M. Looft, Nicole Herkert, and Laura Frey-Law. Modification of a three-compartment muscle fatigue model to predict peak torque decline during intermittent tasks. *Journal of Biomechanics*, 77:16–25, 2018. ISSN 0021-9290. doi: https://doi.org/10.1016/j.jbiomech.2018.06.005. URL https://www.sciencedirect.com/science/article/pii/S0021929018304342.

Christophe Maufroy, Hiroshi Kimura, and Kunikatsu Takase. Towards a general neural controller for quadrupedal locomotion. *Neural Networks*, 21(4):667–681, 2008.

Xue Bin Peng, Glen Berseth, KangKang Yin, and Michiel Van De Panne. Deeploco: Dynamic locomotion skills using hierarchical deep reinforcement learning. *ACM Transactions on Graphics (TOG)*, 36(4):1–13, 2017.

Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics (TOG)*, 37(4):1–14, 2018.

Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. Amp: Adversarial motion priors for stylized physics-based character control. *arXiv preprint arXiv:2104.02180*, 2021.

Joose Rajamäki and Perttu Hämäläinen. Augmenting sampling based controllers with machine learning. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 1–9, 2017.

Rajkumar Ramamurthy, Christian Bauckhage, Rafet Sifa, Jannis Schücker, and Stefan Wrobel. Leveraging domain knowledge for reinforcement learning using mmc architectures. In *International Conference on Artificial Neural Networks*, pp. 595–607. Springer, 2019.

RO Robinson, Walter Herzog, and Benno M Nigg. Use of force platform variables to quantify the effects of chiropractic manipulation on gait symmetry. *Journal of manipulative and physiological therapeutics*, 10(4):172–176, 1987.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. PMLR, 2015a.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015b.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Shinjiro Sueda, Andrew Kaufman, and Dinesh K Pai. Musculotendon simulation for hand animation. In *ACM SIGGRAPH 2008 papers*, pp. 1–8. 2008.

Gentaro Taga. A model of the neuro-musculo-skeletal system for human locomotion. *Biological cybernetics*, 73(2):97–111, 1995.

Darryl G Thelen, Frank C Anderson, and Scott L Delp. Generating dynamic simulations of movement using computed muscle control. *Journal of biomechanics*, 36(3):321–328, 2003.

Winnie Tsang, Karan Singh, and Eugene Fiume. Helping hand: an anatomically accurate inverse dynamics solution for unconstrained hand motion. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 319–328, 2005.

Jack M Wang, Samuel R Hamner, Scott L Delp, and Vladlen Koltun. Optimizing locomotion controllers using biologically-based actuators and objectives. *ACM Transactions on Graphics (TOG)*, 31(4):1–11, 2012.

Jungdam Won, Deepak Gopinath, and Jessica Hodgins. A scalable approach to control diverse behaviors for physically simulated characters. *ACM Transactions on Graphics (TOG)*, 39(4): 33–1, 2020.

Ting Xia and Laura A. Frey Law. A theoretical approach for modeling peripheral muscle fatigue and recovery. *Journal of Biomechanics*, 41(14):3046–3052, 2008. ISSN 00219290. doi: 10.1016/j.jbiomech.2008.07.013.

Wenhao Yu, Greg Turk, and C. Karen Liu. Learning symmetric and low-energy locomotion. *ACM Transactions on Graphics*, 37(4), 2018. ISSN 15577368. doi: 10.1145/3197517.3201397.

A Appendix

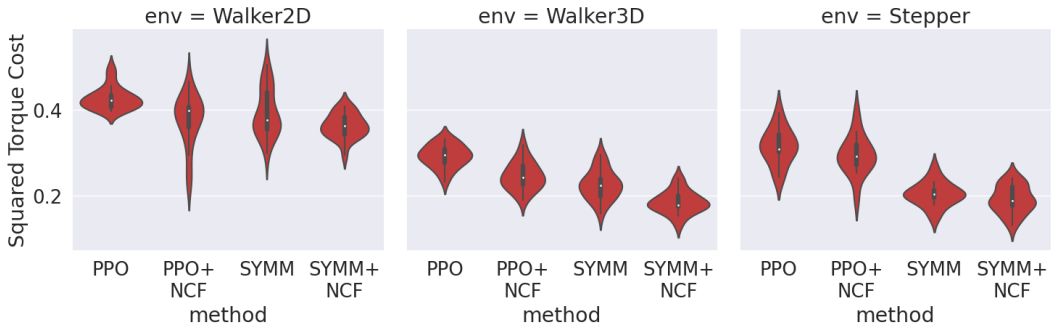


Figure 6: Violin plots using the Squared Torque Cost.

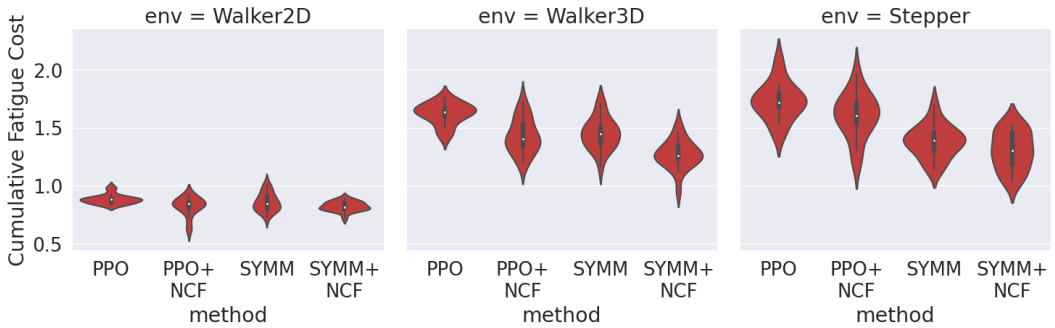


Figure 7: Violin plots using the Cumulative Fatigue Cost.

Table 4: Vectorized Symmetry Index (median[std]): Lower numbers are better.

Method	Environment		
	Walker2D	Walker3D	Stepper
PPO (Schulman et al., 2017)	10.93 [7.804]	77.212 [20.193]	79.678 [20.901]
PPO+3CC	25.116 [10.056]	NaN	NaN
PPO+NCF	18.860 [6.794]	75.790 [23.276]	73.961 [22.581]
SYMM (Yu et al., 2018)	2.258 [1.193]	**12.119** [32.323]	48.753 [32.755]
SYMM+3CC	**1.626** [1.076]	NaN	**18.602** [2.993]
SYMM+NCF (Ours)	1.801 [0.911]	14.217 [15.791]	22.587 [24.557]

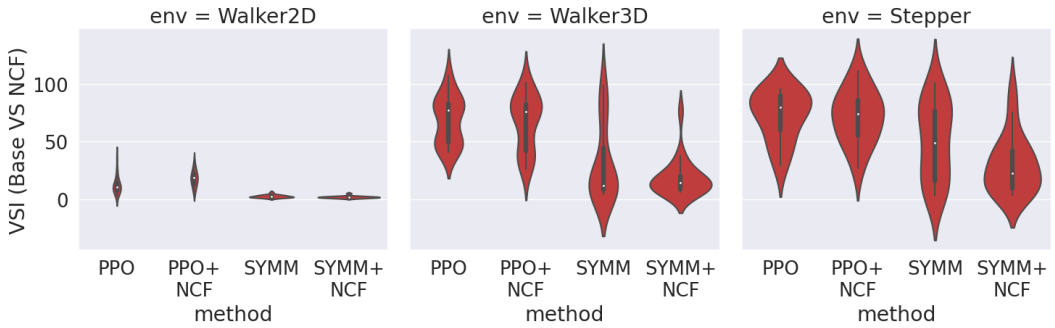


Figure 8: Violin plots using the Vectorized Symmetry Index (VSI).

Table 5: Phase-Portrait Index (median[std]): Lower numbers are better.

Method	Environment		
	Walker2D	Walker3D	Stepper
PPO (Schulman et al., 2017)	0.298 [0.068]	2.464 [0.605]	2.525 [0.801]
PPO+3CC	0.719 [0.372]	NaN	NaN
PPO+NCF	0.277 [0.136]	2.536 [0.812]	2.103 [0.834]
SYMM (Yu et al., 2018)	0.153 [0.022]	0.285 [1.160]	2.687 [1.042]
SYMM+3CC	0.167 [0.050]	NaN	**0.152** [0.009]
SYMM+NCF (Ours)	**0.151** [0.019]	**0.222** [0.911]	0.251 [1.243]

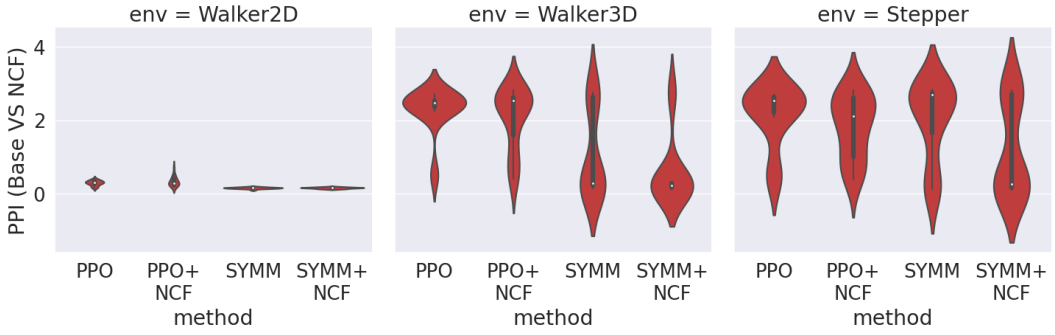


Figure 9: Violin plots using the Phase-Portrait Index (PPI).

Table 6: Spectral Entropy (median[std]) of Hip Angle: Lower numbers are better.

Method	**Environment**		
	Walker2D	**Walker3D**	**Stepper**
PPO (Schulman et al., 2017)	1.911 [0.272]	2.518 [0.695]	1.825 [0.942]
PPO+3CC	1.760 [0.253]	NaN	NaN
PPO+NCF	1.697 [0.275]	1.923 [0.652]	2.286 [0.936]
SYMM (Yu et al., 2018)	1.537 [0.398]	**1.252** [0.361]	0.994 [0.425]
SYMM+3CC	1.427 [0.334]	NaN	**0.561** [0.001]
SYMM+NCF (Ours)	**1.394** [0.347]	1.262 [0.390]	0.788 [0.277]

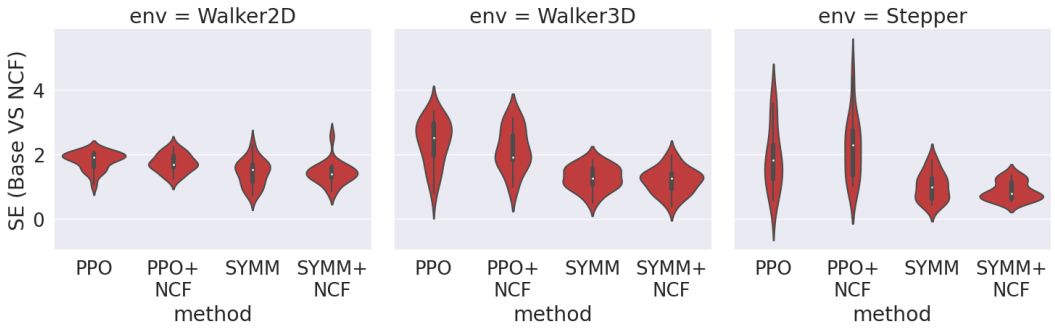


Figure 10: Violin plots using the Spectral Entropy (SE) for the hip angle.