In Silico User Testing for Mid-Air Interactions with Deep Reinforcement Learning

by

Noshaba Cheema

Master Thesis

Faculty of Natural Sciences and Technology I
Department of Computer Science
Saarland University

Supervisor
Prof. Dr.-Ing. Philipp Slusallek

Advisors
Prof. Dr.-Tech. Perttu Hämäläinen
Prof. Dr.-Tech. Jaakko Lehtinen

Reviewers
Prof. Dr.-Ing. Philipp Slusallek
Prof. Dr.-Tech. Perttu Hämäläinen

September 30, 2019

UNIVERSITÄT DES SAARLANDES

Declaration of Authorship

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Statement in Lieu of an Oath

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Datum/Date: _____

Unterschrift/Signature: _____

SAARLAND UNIVERSITY

Department of Computer Science

Abstract

In Silico User Testing for Mid-Air Interactions with Deep Reinforcement Learning

by Noshaba Cheema

User interface design for Virtual Reality and other embodied interaction contexts has to carefully consider ergonomics. A common problem is that mid-air interaction may cause excessive arm fatigue, known as the "Gorilla arm" effect. To predict and prevent such problems at a low cost, this thesis investigates user testing of mid-air interaction without real users, utilizing biomechanically simulated AI agents trained using deep Reinforcement Learning (RL). This is implemented in a pointing task and four experimental conditions, demonstrating that the simulated fatigue data matches ground truth human data. Additionally, two effort models are compared against each other: 1) instantaneous joint torques commonly used in computer animation and robotics, and 2) the recent Three Compartment Controller ($3CC$-r) model from biomechanical literature. $3CC$-r yields movements that are both more efficient and natural, whereas with instantaneous joint torques, the RL agent can easily generate movements that are unnatural or only reach the targets slowly and inaccurately. This thesis demonstrates that deep RL combined with the $3CC$-r provides a viable tool for predicting both interaction movements and user experience *in silico*, without users.

Acknowledgements

CONTENT

Dedicated to PUSCHELCHEN ♡

CHAPTER 1

INTRODUCTION

Interaction design (IxD) is the "practice of designing interactive digital products, environments, systems, and services" [1]. Typical design goals are defined in terms of subjective measures, such as user enjoyment in games or low perceived exertion or effort in gestural interaction, making designing such experiences fundamentally difficult. Hence, IxD is in practice an iterative trial-and-error process where the design needs to be gradually and expensively improved by observing users interacting with prototypes. Designing complex interactive experiences without actual users is therefore even more challenging as the effects of design decisions are nearly impossible to predict.

Despite this, a rising trend in human-computer interaction design is to utilize computational models of users to predict the user experience [2–7]. A major advantage of this is that lengthy and tedious trial-and-error processes can be omitted if sufficient accuracy is reached with such models. Evaluation of alternative solutions to design problems can be done rapidly *in silico*, without users - or at least a pre-selection of the most likely solutions which can later be tested in real life. With the

help of such computational methods a designer can further deploy optimization algorithms to automatically find and propose high-value solutions.

Computational user models have been successfully applied in, e.g., game playing [7–9] and typing [5]. However, many complex interactions are still challenging to model, in particular in the domain of embodied experiences such as Virtual Reality (VR), which require modeling the user's body and biomechanics. Fortunately, new and powerful tools are emerging: Recent advances in deep Reinforcement Learning (RL) [10–13] provide a generic approach to train intelligent agents for any kind of simulated system such as a video game or biomechanical simulation, provided that one can define the agent's goals or tasks as a reward function such as a game score. For user modeling, this means that one needs to make minimal assumptions about user behavior; instead, the agents will explore and discover the behaviors of maximal utility – i.e., cumulative rewards – following a computational rationality model of behavior [14]. Such AI agents can also be extended with models of intrinsic motivation and emotion [15, 16], which can allow prediction of the user experience and behavior beyond simple task-driven behavior and associated metrics like task success rate [4].

1.1 Contribution

This thesis contributes the first user modeling experiment that combines deep RL with a biomechanical arm simulation model that allows both synthesizing mid-air interaction movements and predicting the associated embodied user experience,

with a focus on subjective fatigue. This approach is tested in a mid-air pointing task and four experimental conditions, replicating the experiment design of [17] who tested the same conditions on real humans and analyzed fatigue using motion capture data. The agent is able to efficiently learn the pointing movements needed for the tasks, doing away with the need to capture motions from real humans. Furthermore, the simulation-based fatigue data provides a good fit with the human data of [17]. Compared to the optimization of mid-air pointing movements of Montano Murillo et al. [18], the method described in this thesis does not rely on predetermined effort estimates for different spatial locations. Instead, all resulting data simply emerges from the biomechanical simulation model and rewarding the agent for both accomplishing the task and avoiding fatigue/discomfort.

As an additional contribution, two different fatigue/effort models incorporated as reward function components are compared: 1) instantaneous joint torques common in computer animation, robotics [19], and standard deep RL movement control benchmark tasks (MuJoCo) [20], and 2) the recent Three Compartment Controller (3CC-r) model from biomechanical literature [21]. This work demonstrates that the 3CC-r model yields movements that are both more efficient and natural, whereas with instantaneous joint torques, the RL agent can easily generate movements that are unnatural or only reach the targets slowly and inaccurately. As the 3CC-r model causes no significant increase in computational complexity, this thesis advocates deep RL researchers also incorporating it to their benchmark tasks to increase both the realism of biomechanical effort modeling and naturalness of emerging movements.

CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 Simulating User Behavior

The literature on user modeling features different kinds of models. The most simple ones like Fitt's law [22] allow predicting a quantity like pointing target acquisition time as a function of design variables like target distance using simple mathematical expressions. However, in many cases such mathematical models are not available, and one must instead resort to simulations of how users perceive, things, and act while completing tasks. This was first proposed by Card et al. [5, 23] as early as in 1983 in their GOMS model. As described in [23] GOMS is "a set of **G**oals, a set of **O**perators, a set of **M**ethods for achieving the goals, and a set of **S**elections rules for choosing among competing methods for goals" [23]. Their model was later extended by Adaptive Control of ThoughtRational (ACT-R) and other more sophisticated cognitive architectures [5, 24], which rigorously define basic cognitive and perceptual operations. In theory, each task should consist of a series of these discrete operations.

A limitation of early user models like GOMS is their complexity: Successful application of the model in a design task requires the designer to provide a detailed breakdown of the user's goals and expected behavior. Early cognitive architecture development was also plagued by various cognitive processes hand-crafted in isolation, with difficulties of integrating them to general solutions for, e.g., autonomous skill acquisition [24]. However, this was prior to the recent deep neural network revolution; deep Reinforcement Learning (RL) agents have now been demonstrated to learn a wide variety of skills ranging from video game play [11] to controlling the movement of biomechanically simulated human bodies [13]. Although RL methods can be complex, they are simple to apply , which makes them lucrative for user simulation purposes.

Reinforcement Learning is an approach to discovering the optimal actions for a Markov Decision Process (MDP) [10]. It is assumed that at time t, the agent is in state s_t, takes action a_t, and observes a reward r_t and a next state s_{t+1}. The agent is optimizing utility, i.e., expected cumulative future rewards, in line with the computational rationality view of human behavior [14]. Thus, for user modeling, one only needs to define the states, actions, and rewards. At least in some cases, the reward function can also be inferred from human data [25, 26].

Traditionally, RL user simulation has been limited into simple MDP:s with discrete, enumerable states and actions, e.g., dialogue systems, menus and simple keyboards [27–29]. The discrete states and actions make the MDP:s solvable with classic RL methods like Q-learning [10]. However, recent deep RL methods like Proximal Policy Optimization [30] and Soft Actor Critic [12] also work with

the high-dimensional states and actions required for intelligent control of human biomechanical simulation [13]. The thesis demonstrates that this makes deep RL a viable approach for modeling embodied interaction and predicting user experience outcomes such as fatigue.

A recent result that further motivates this work is that biomechanically realistic movement can be synthesized efficiently through simplified skeletal simulation without muscle and tendon detail, as long as the actuation effort minimized by an RL agent is computed with a higher degree of biomechanical realism through a machine learning model that predicts muscle activations from joint actuation torques [31]. Extending this approach, the presented model actuates with joint torques and increases biomechanical realism through a fatigue model incorporated as an extra reward function component.

It should be noted that modern neural-network based RL also generalizes to Partially Observable Markov Decision Processes (POMDP:s) where the agent cannot access the full environment or simulation state [32]. This is often the case in user modeling that incorporates realistic perception models.

2.2 Quantifying Mid-Air Interaction Fatigue

Muscle fatigue is the failure to maintain the required or expected force [33]. Fatigue depends on a multitude of simultaneous physiological and neurological processes, making it difficult to pinpoint a single mechanism responsible for the loss of force

[34–37]. It is also task-related and can vary across muscles and joints [17, 34, 38–41], which partially explains the challenging nature of representing muscle fatigue analytically [34].

A widely used empirical model to estimate the effect of fatigue on the task endurance time (ET) of static load conditions is the Rohmert's curve [38, 42]. Hincapié et al. [43] developed Consumed Endurance (CE), a metric to quantify arm fatigue of mid-air interactions, based on the Rohmert's curve. Although straightforward, the approach lacks the ability to generalize to dynamic load conditions or recovery during rest periods [17, 34] as it is based on the Rohmert's curve, which is only valid for static load conditions. Furthermore, CE is assumed to be zero at exertion levels below 15%. This limits the use of the model for evaluating mid-air interaction with low exertion levels [17, 40].

Liu et al. [44] have proposed a motor unit (MU)-based fatigue model which uses three muscle activation states: resting (M_R), activated (M_A) and fatigued (M_F). The model is able to predict fatigue at static load conditions but fails at sub-maximal or dynamic conditions [17, 34]. Xia et al. [34] have proposed a Three-Compartment Controller (3CC) model which improves upon the model of Liu et al. for dynamic load conditions by introducing a feed-back controller term between the active (M_A) and rest (M_R) muscle states. Frey et al. [41] validated the 3CC model in estimating ET under static load conditions and have obtained joint-specific parameters. Later, Jang et al. [17] have optimized the 3CC model for mid-air interaction tasks [17]. They further show that the 3CC model can be used to estimate ground truth human perceived fatigue ratings based on the Borg

CR10 scale [45]. While their findings are based on kinematic data only, they still require motion data captured from real humans.

More recently, Looft et al. [21] have published an improved 3CC-r model for intermittent tasks by introducing an additional rest recovery multiplier r and validated their results based on perceived fatigue from participants for specific joints.

CHAPTER 3

PRELIMINARIES: FATIGUE MODELING

Two different fatigue models are investigated in this thesis: 1) instantaneous joint torques as a measure of instantaneous effort, and 2) the recently developed Three-Compartment Controller (3CC-r) Model by Looft et al. [21].

3.1 Instantaneous Joint Torque Effort

Instantaneous joint torque is a simple measure used in computer animation, robotics, and standard RL benchmark problems [46–49] to measure and minimize the instantaneous effort of a given task a simulated agent is performing. When defining movement optimization objective functions, the torques are usually squared to make the optimization avoid using excessive strength.

3.2 Three-Compartment Controller (3CC-r) Model

An instantaneous effort model gives a simple measure to determine the difficulty of a given task. However, it is not very biologically accurate as a simple task can

11
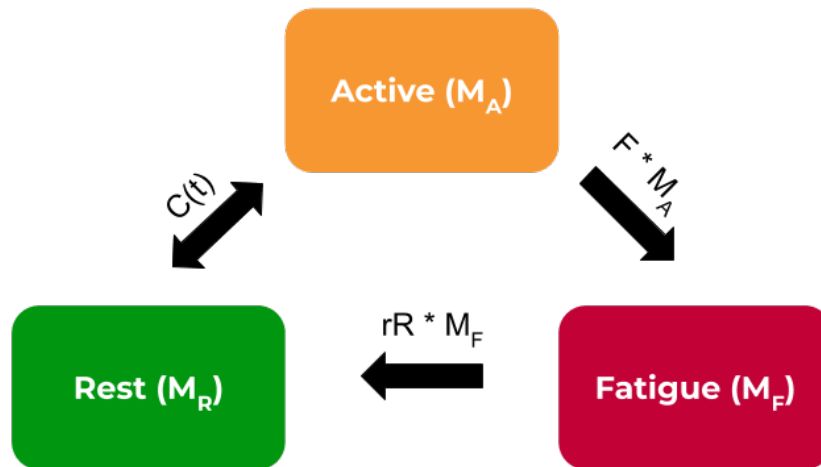
FIGURE 3.1: Three compartment controller model.

become difficult when done long enough. A cumulative effort function, such as the 3CC-r model, thus gives a more accurate representation of perceived fatigue.

Akin to Liu et al. [44], the 3CC-r model [21] assumes motor units (MUs) to be in one of the three possible states:

- *active* - MUs contributing to the task

- *fatigued* - fatigued MUs without activation

- *resting* - inactive MUs not required for the task

Fig. 3.1 shows the relationship between these states. $M_A(\%)$ is the compartment of active MUs, $M_F(\%)$ the compartment of fatigued, and $M_R(\%)$ the compartment of resting MUs. Each compartment is expressed as a percentage of the *maximum voluntary contraction* ($\%MVC$) [17, 34].

In addition to that, the compartment theory is combined with control theory to define system behaviour which matches muscle physiology [34], i.e. active MUs'

force production should begin to decay (fatigue) over time. This is expressed by the following equations:

$$\frac{\partial M_R}{\partial t} = -C(t) + rR \cdot M_F \tag{3.1}$$

$$\frac{\partial M_A}{\partial t} = C(t) - F \cdot M_A \tag{3.2}$$

$$\frac{\partial M_F}{\partial t} = F \cdot M_A - rR \cdot M_F \tag{3.3}$$

Where F and rR are the model parameters defining at which rate the motor units fatigue, and which rate they recover and enter the rest period, respectively. In contrast to the traditional 3CC model [34], the 3CC-r model introduces an additional rest recovery factor r, which enhances the recovery when the required force, i.e. target load (TL), is zero to better represent perceived fatigue estimates from user studies [21]:

$$rR = \begin{cases} r \cdot R & \text{if } M_A \geq TL \\ \\ R & \text{else} \end{cases} \tag{3.4}$$

The 3CC-r [21] is equivalent to the 3CC [34] model when $r = 1$. Based on a sensitivity analysis r is set to 7.5. F is set to 0.0146, and R to 0.0022 based on Jang et al. [17] 3CC-model optimization for mid-air interactions.
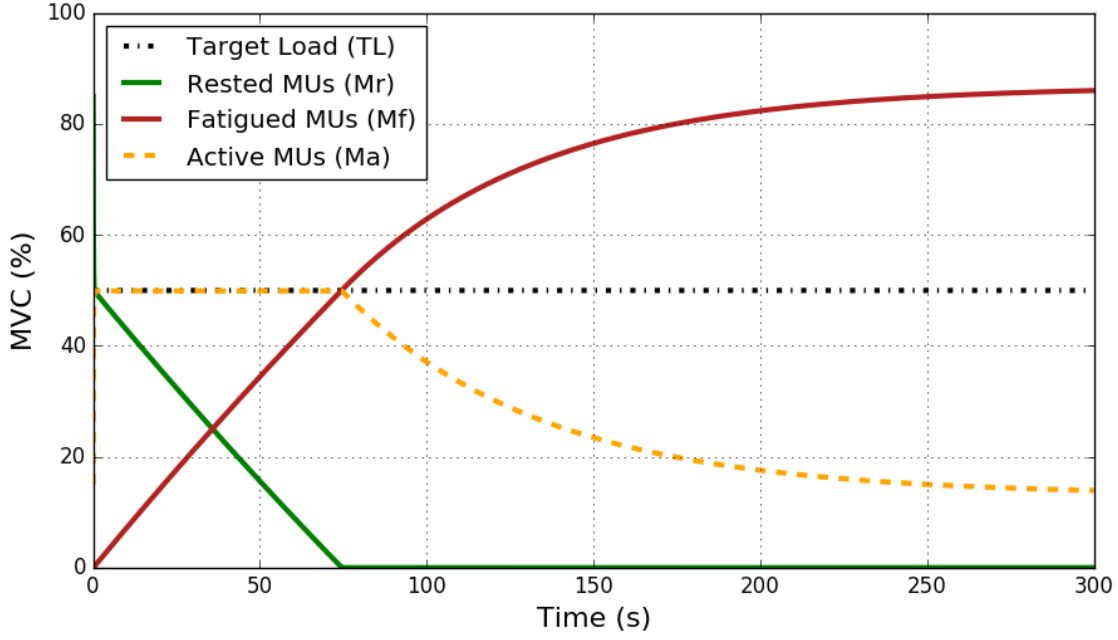
FIGURE 3.2: Behaviour of 3CC-*r* model at target load of 50% MVC (black dotted line). The load cannot be held after around 70 s (yellow dashed line).

$C(t)$ is the time-varying muscle activation-deactivation drive, which can produce the target load TL in percent by controlling the size of M_A and the availability of M_R. The following equations describe $C(t)$ mathematically:

$$C(t) = \begin{cases} L_D \cdot (TL - M_A) & \text{if } M_A < TL \text{ and } M_R > (TL - M_A) \\ L_D \cdot M_R & \text{if } M_A < TL \text{ and } M_R \leq (TL - M_A) \\ L_R \cdot (TL - M_A) & \text{if } M_A \geq TL \end{cases} \quad (3.5)$$

L_D is the muscle force development factor, and L_R is the relaxation factor. Based on the analysis by [34], these are set to 10.

CHAPTER 4

SYSTEM

The system is implemented using the Unity game engine and their ML Agents Toolkit v0.8.2 [50] implementation of the Proximal Policy Optimization (PPO) [30] RL algorithm. The following details our effort model, the simulated pointing task, and the RL problem formulation and training settings.

4.1 Fatigue Model

4.1.1 Instantaneous Fatigue Model

In this thesis, instantaneous torques are compared against to the more advanced 3CC modeling. More specifically, instantaneous torques normalized with respect to maximum torque T_{max} are used:

$$Effort_I(\vec{T}) = \left(\frac{\|\vec{T}\|}{T_{max}} \right) \tag{4.1}$$

4.1.2 Cumulative Fatigue Model

Since the 3CC model is a relative unit-less system [34], TL can be described in a variety of options, e.g. the percentage of maximum voluntary torques (MVT) or forces (MVF).

Previous studies [17, 21, 41] set TL as the ratio of the magnitude of torque $\vec{T}$ and the maximum voluntary torque T_{max} at a joint: $\frac{\|\vec{T}\|}{T_{max}} \cdot 100\%$. While this is a valid approach to measure the load at a given joint, it is more accurate to model two 3CC-r models per degree-of-freedom (DOF) (one for the "positive", and one for the "negative" direction)). Each DOF roughly corresponds to a muscle group. The presented method avoids modeling at the level of individual muscles to reduce simulation and RL training time.

The target load can then be expressed as a vector of torque ratios for each direction:

$$\vec{TL}(\vec{T}) = \left[\frac{T_1^+}{T_{max}}, \frac{T_1^-}{-T_{max}}, \frac{T_2^+}{T_{max}}, \frac{T_2^-}{-T_{max}}, \frac{T_3^+}{T_{max}}, \frac{T_3^-}{-T_{max}} \right]^\top \qquad (4.2)$$

Where $\frac{T_i^+}{T_{max}}$ is the ratio of the torque at axis i in the "positive" direction, and $\frac{T_i^-}{-T_{max}}$ in the "negative" direction, respectively. When $\frac{T_i^+}{T_{max}} \geq 0$, then $\frac{T_i^-}{-T_{max}} = 0$, and vice versa. Each value in $\vec{T}L$ is used as input for a separate 3CC-r model.

For simulated agents the cumulative effort $Effort_C$ is described as the difference between the actual target load $\vec{T}L(\vec{T})$ at the joint and the desired muscle activation $\vec{M}_A = \left[M_{A_1}^+, M_{A_1}^-, M_{A_2}^+, M_{A_2}^-, M_{A_3}^+, M_{A_3}^- \right]^\top$ given by the 3CC-r model:
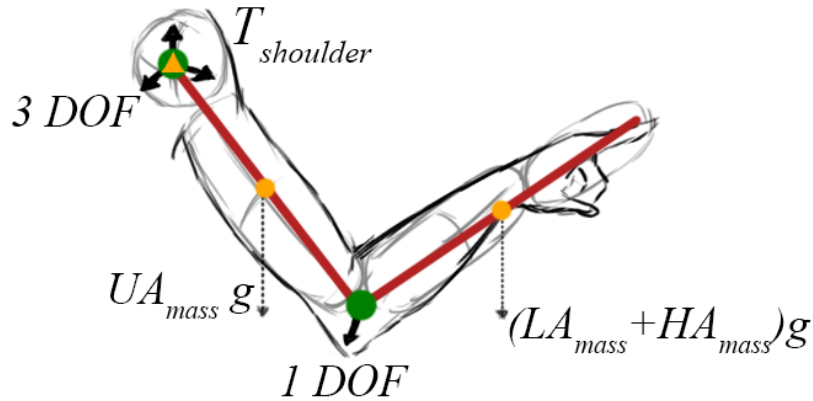
FIGURE 4.1: Forces acting on our biomechanical upper limb model. The limbs are modeled as rigid bodies (red) connected with joints (green). The degrees of freedom (DOF) of each joint are denoted by the arrows at the respective joint.

$$Effort_C(\vec{T}) = \left\| \frac{\vec{M}_A}{100} - \vec{T}L(\vec{T}) \right\| \tag{4.3}$$

The advantage of this over using M_F directly is that the cumulative fatigue given by the 3CC model stagnates after some time (Fig. 3.2 *red line*). Furthermore, it is not clear from M_F alone when a target load could not be held. Using the difference between the actual active motor units and the desired load tells when the load is okay to be held, and when it becomes a burden. This is shown in Fig. 3.2, where the active motor units (*yellow*) start to decline after 70 s because the target load (*black*) was not sustainable.

4.2 Simulated Upper Limb Model

Similar to [17], arm fatigue is mostly assumed to be attributed to shoulder-joint fatigue, due to the shoulder fatiguing faster than the elbow or wrist during arm
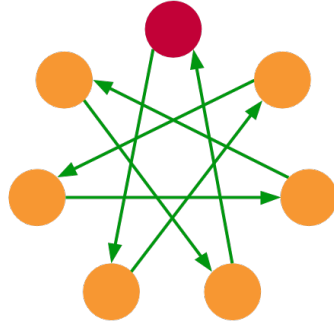
FIGURE 4.2: ISO 9241-9 reciprocal pointing task with 7 targets. Agents point to highlighted (red) target. Targets highlight in the pattern indicated by the arrows.

movement [40]. Thus, only the shoulder joint torques are used for either effort model. The arm is simulated using a 4 DOF serial chain - 3 for the shoulder and 1 for the elbow (Fig. 4.1), where the limbs are modeled as rigid bodies connected by joints [51]. To estimate the shoulder joint the method described in [43] is used.

4.3 Mid-Air Pointing Task

The mid-air pointing task is modeled after the ISO 9241-9 standard [17, 52, 53] based on Fitts' law [22, 54]. The standard is extensively used for evaluating 2D pointing devices, such as mice, pens, and touch screens [53]. The task has participants point at a circle of targets, with a given width and distance to each other, in a given order. Akin to [17], this work uses seven targets with a width of 10 cm and a distance of 30 cm to each other, corresponding to an index of difficulty ID [22, 54] of $ID = \log_2\left(\frac{30}{10} + 1\right) = 2$. Fig. 4.2 shows the target sequence.

Previous studies [17, 43, 55] have shown that the height and distance from the arm's resting position affect the perceived fatigue ratings of participants, making them fatigue more rapidly the higher and further away the arm is. Additionally,

rest periods are also a decisive factor of how fatigued we are [17]. These two factors are investigated in the presented experiments and compared to ground truth human data from prior studies [17].

Similar to Jang et al. [17], the task switches between pointing and resting periods. Like them, the following four different rest periods are used: [5s, 10s, 15s, 20s].

4.4 RL Problem Formulation

To be able to apply RL methods, the MDP needs to be defined, i.e. states, actions, and rewards. Additionally, the PPO method used in this thesis requires the definition of a policy network for sampling an action given the current state. The MDP transition model $s_{t+1} = f(s_t, a_t)$ is implemented by Unity's ML-Agents Toolkit: After sampling an action, it actuates the simulation and queries the next state.

4.4.1 State and Action Space

The state vector $\vec{s}$ of the agent, comprises a concatination of the following features:

- limb positions with respect to the shoulder [9 values]

- linear velocities of upper and lower arm limbs [6 values]

- angular velocities of upper and lower arm limbs [6 values]

- direction vector from finger tip to target (not normalized) [3 values]

- target switch time [1 value]

- rest period Boolean [1 value]

The RL actions are defined as actuation torques applied through Unity's *Ad-dTorque()* method at the center of mass of the upper and lower arm limbs. The action vector $\vec{a}$ from the policy specifies how much of the maximum voluntary torques (MVT) to apply to the limbs. It is comprised of 4 values, denoting the three actuation torque values for the upper, and one for the lower arm. The torques that are applied cannot overshoot shoulder MVT and elbow MVT values, respectively. The shoulder MVT is furthermore used as T_{max} in the effort calculation, described in Section 4.1.

4.4.2   Network

A policy π is represented by a neural network which maps a given state s to a distribution over action $\pi(a|s)$. The action distribution is modeled as a Gaussian, where the state dependant mean $\mu(s)$ and the diagonal covariance matrix Σ are specified by the network output:

$$\pi(a|s) = \mathcal{N}(\mu(s), \Sigma) \tag{4.4}$$

The inputs s to the network is processed by two fully-connected layers with 128 hidden units, each, using the Swish [56] activation function. During training the

network adapts the mean and covariance matrix such that the actions become less noisy as the agent gradually starts to exploit instead of explore.

4.4.3 Reward

The reward $\rho(t)$ at each step t consists of two terms that encourage the character to pointing towards the target when there is one, while using minimal effort:

$$\rho(t) = (\omega_P \rho(t)_P + \omega_F \rho(t)_F) \cdot 0.01 \tag{4.5}$$

$\rho(t)_P$ and $\rho(t)_F$ are defined as the pointing and fatigue objectives, respectively, with ω_P and ω_F being their respective weights. The pointing objective encourages the agent to point towards the current target, while the fatigue objective encourages it to make use of actions which require less effort than others. Setting $\omega_P = 100$ and $\omega_F = 0.01$ results in the desired behavior for various settings.

The pointing reward $\rho(t)_P$ depends on the distance between the target and the finger tip, and is defined with:

$$\rho(t)_P = \begin{cases} 1 & \text{target has been hit} \\ \exp\left(-\frac{\|p_{target}(t) - p_{finger}(t)\|^2}{\tau_P^2}\right) & \text{else} \end{cases} \tag{4.6}$$

where $p_{target}(t)$ is the target's position, and $p_{finger}(t)$ the position of the finger tip at step t, respectively. τ_P the tolerance distance in meters for when the reward

becomes $\rho(t)_P \approx 0.3679$, when the finger does not hit the target. τ_P is set to $\tau_P = 0.15$. A minimal reward of approx. 0.001 is obtained at around 40 cm from the target with this objective function.

The fatigue reward $\rho(t)_F$ is defined using the $Effort$ functions described in Section 4.1:

$$\rho(t)_F = \exp\left(-\frac{Effort(\vec{T}(t))^2}{\tau_F^2}\right) \tag{4.7}$$

τ_F is the tolerance in percentage of how much the torque ratio is allowed to deviate from the desired torque ratio, while obtaining a reward of at least approx. 0.3679. In the case of the instantaneous effort function this means how much percent is the shoulder muscle allowed to deviate from zero torque. However, in the case of the cumulative effort function based on the 3CC-r model, this means how much is it allowed to deviate from the allowed motor unit activation $\vec{M}_A$ given by the 3CC-r model. The best tolerance value τ_F for each effort model is determined in Section 5.1.

Note that in some movement optimization cases, squared cost terms are used without the exponentiation [47, 57]. The exponentiation in this thesis follows Peng et al. [49]; it converts minimized costs to maximized rewards and also limits the reward to a predefined range, which makes it easier to train PPO's value function predictor network.

4.5 Training

The policy training is done using the Proximal Policy Optimization (PPO) algorithm [30, 50]. Standard hyperparameters defined in [50] are used, with the following adjustments: batch size = 2024, buffer size = 20240, $\gamma = 0.995$, max steps = $1.0e^6$, normalization = True, number of epochs = 3, time horizon = 1000, summary of frequency = 3000.

4.5.1 Initial State Distribution

PPO training proceeds in episodes, where at the start of each episode the agents and the environment are reset to an initial state s_0. Each episode is simulated to a fixed time horizon with actions sampled from the policy, after which the agents and the environment are reset again. In total, $1e^6$ time steps, or 1000 episodes with a time horizon of 1000 for each episode is used in this thesis.

Many RL benchmark problems such as the MuJoCo locomotion environments use a fixed initial state s_0 or add only small random perturbations to it [46]. However, as demonstrated by [49], a diverse enough initial state distribution can greatly improve movement learning. To implement this, multiple settings of the pointing task are randomly sampled from a uniform distribution. Table 4.1 shows an overview of these settings. *Target Height* and *Target Distance* are in relation to the shoulder position and the center point of the target circle. *Pointing Period* describes the duration of the pointing period before the user is supposed to rest,

Training Settings for Pointing Task

Setting	Distribution	Lower Bound	Upper Bound
Target Height	Uniform	−40 cm	+20 cm
Target Distance	Uniform	+10 cm	+70 cm
Switch Time	Uniform	1 s	2 s
Pointing Period	Uniform	30 s	90 s
Rest Period Index	Uniform	1	4
Initial Target Index	Uniform	1	7

TABLE 4.1: Settings of the pointing and resting task during training. Target height and distance are in relation to the shoulder position.

while *Rest Period Index* defines which of the four rest periods [5s, 10s, 15s, 20s] to use. The initial target index of the seven targets sequence is chosen randomly.

During training five agents are used whose attributes are also sampled randomly for each episode. However, instead of sampling from a uniform distribution they are sampled from a Gaussian distribution to more accurately represent male and female strength properties. Table 4.2 shows which features are sampled from what kind of distribution. For the MVT of shoulder, average values found in biomechanical literature [58] are used. Based on [58, 59] the average elbow MVT is estimated between 1 Nm to 10 Nm higher to the person's shoulder MVT. Hence, the elbow MVT is sampled from a uniform distribution, where the lower bound (LB) is the agent's shoulder MVT and the upper bound (UB) is an additional +10 Nm. Once the body weight is sampled, the actual arm weight is set according to average body weight percentages for the upper limbs (Table 4.3) [60].

For the 3CC-*r* model a random initial fatigue is set by sampling uniformly from an initial shoulder load (between $-T_{max}$ and T_{max} for each of the three DOF), which is then applied for a randomly set time (between 0 s and 180 s) onto the

Setting	Distribution	Agent Settings	
		LB / μ	UB / σ
Body Weight (M)	Gaussian	80 kg	1.0
Body Weight (F)	Gaussian	65 kg	1.0
MVT Shoulder (M)	Gaussian	54.9 Nm [58]	1.0
MVT Shoulder (F)	Gaussian	29.2 Nm [58]	1.0
MVT Elbow	Uniform	+0 Nm	+10 Nm

TABLE 4.2: Weight and Maximum Voluntary Torque (MVT) settings for upper body limbs. The elbow MVT is in relation to the agent's shoulder MVT.

Limb	Length	Arm Length & Weight	
		Male	Female
Upper Arm	29.06 cm	2.71%	2.55%
Lower Arm	29.44 cm	1.62%	1.38%
Hand	21 cm	0.61%	0.56%

TABLE 4.3: Length of arm limbs in cm and their corresponding weights in percentage of the total body weight based on [60].

joint. With this experiences for more long-term fatigue effects are gathered in the described experiments.

CHAPTER 5

EVALUATION AND RESULTS

Our system is evaluated in two experiments, with details provided below. The results are visualized in Figures 5.3 and 5.4.

First, the movement synthesis quality of the two effort models in terms of both accuracy of reaching the pointing targets and naturalness of movement are compared. As the presented method defines naturalness as the arm relaxing when possible, there is an obvious accuracy-naturalness trade-off for the agent, which can be adjusted with the reward function parameter τ_F. *Fig. 5.3 indicates that the 3CC model is able to obtain better combinations of naturalness and accuracy, i.e., points near the bottom-right corners of the plots.*

Second, using the model parameters that yield the best accuracy-naturalness trade-off, the modeled fatigue is compared to ground truth human data. Fig. 5.4 shows how the modeled fatigue gives similar fit to the ground truth data as previous work that used motion capture data instead of movement synthesized through RL-controlled simulation. *This indicates that the presented approach is successful in predicting user experience without actual data from users.*

FIGURE 5.1: Four interaction zones used for determining the best model. 1) Target is shoulder height and arm is bent. 2) Target is waist height and arm is bent. 3) Target is shoulder height and arm is straight. 4) Target is waist height and arm is bent.

5.1 Naturalness vs. Accuracy

To determine the best fatigue tolerance value τ_F for each effort model, a grid search on the trained networks is done and the results are plotted in terms of accuracy and naturalness.

Values between $0.0 \leq \tau_F \leq 0.5$ in 0.02 steps were tested. For this experiment 20 agents with different settings for each trained model are used. The parameters of the agents are seeded to have the same settings for each model. To make the models comparable to ground truth human data [17], the targets' switch time is set to 1.3 s, and the pointing period to 60 s. Jang et al. [17] determined that if subjects performed four mid-air interaction periods in a series they had a higher chance of learning and pre-fatigue effects [61]. Hence, the presented experiment is designed in a similar fashion to theirs with the following rest periods in between in this order: [20s, 5s, 15s, 10s]. This setup is akin to group 1 in [17]. In total the pointing task lasts roughly 5 min. In this experiment the targets are placed at four different interaction zones with five agents sharing the same zone: one at shoulder height and having the arm bent, one at waist height and having the arm
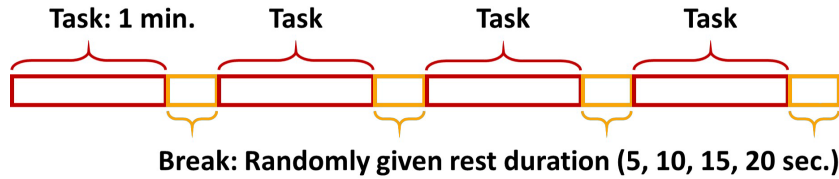
FIGURE 5.2: Experiment protocol of naturalness vs. accuracy measure, as well as during comparison against ground truth human data.

bent, one at shoulder height and the arm straight, and finally one at waist height and having the arm straight. The four interaction zones for this experiment are shown in Fig. 5.1.

For the accuracy of a model this work considers two separate measures: the median of the average distance over time from a target, and the median average time it takes to reach a target. If the agent could not reach the target, the time is set to the switch time. The median is computed over the average value for each agent.

To determine the naturalness η of arm movements, the following equation is used:

$$\eta = 4 - \left(\widetilde{d_E^{point}} + \widetilde{\phi_E^{rest}} + \widetilde{\phi_A^{rest}} + \widetilde{v_A^{rest}} \right) \qquad (5.1)$$

with $d_E^{point}, \phi_E^{rest}, \phi_A^{rest}, v_A^{rest} \in [0, 1]$. The higher the value is, more natural the arm motion is supposed to be. "~" denotes the median value over the agents.

$d_E^{pointing}$ is the average distance over T time steps of the elbow to the plane that is spanned between the shoulder and finger tip and the direction of gravity, when the agent is pointing:

$$d_E^{point} = \frac{1}{T} \sum_t \frac{|\langle \vec{n}, \overrightarrow{ShEl} \rangle|}{\|\overrightarrow{ShEl}\|} \tag{5.2}$$

with $\overrightarrow{ShEl}$ being the vector from the shoulder to the elbow position. The distance is divided by the length of this vector to obtain values between 0 and 1. $\vec{n}$ is the plane normal of the shoulder-finger plane:

$$\vec{n} = \frac{\overrightarrow{ShFi}}{\|\overrightarrow{ShFi}\|} \times \frac{\vec{g}}{\|\vec{g}\|} \tag{5.3}$$

$\overrightarrow{ShFi}$ is the vector from the shoulder to the finger tip. The higher d_E^{point} is, the further away the elbow is from this plane. The idea is that during pointing movements the agent should prefer to keep its elbow down since this position is perceived as less fatiguing than having the elbow point side-ways.

To measure the naturalness of the arm during rest periods, ϕ_E^{rest}, ϕ_A^{rest}, and v_A^{rest} are added into the naturalness equation.

ϕ_E^{rest} is the average elbow angle between the lower and upper arm. The idea is that during rest periods the agent is not supposed to flex its arm much. It is calculated the following way:

$$\phi_E^{rest} = \frac{1}{T}\sum_t \left(\left\langle \frac{\overrightarrow{ElSh}}{\|\overrightarrow{ElSh}\|}, \frac{\overrightarrow{ElHa}}{\|\overrightarrow{ElHa}\|} \right\rangle + 1 \right) \cdot 0.5 \tag{5.4}$$

$\overrightarrow{ElSh}$ is the vector from elbow to shoulder and $\overrightarrow{ElHa}$ from elbow to hand, respectively. When the arm is straight the dot product becomes -1, when the lower arm is perpendicular to the upper arm the dot product is 0, and when the arm is flexing it is close to 1. To keep ϕ_E^{rest} between 0 and 1 (0 being straight and 1 flexing), the dot product is increased by 1 and scaled with 0.5.

While this value determines when the arm is flexing during rest periods, with it alone the naturalness measure would classify holding an arm in front of oneself as more natural than flexing it. Thus, the average angle ϕ_A^{rest} is calculated between arm and the direction of gravity and incorporated in the presented naturalness equation:

$$\phi_A^{rest} = \frac{1}{T}\sum_t \left(\left\langle \frac{\overrightarrow{COMSh}}{\|\overrightarrow{COMSh}\|}, \frac{\vec{g}}{\|\vec{g}\|} \right\rangle + 1 \right) \cdot 0.5 \tag{5.5}$$

$\overrightarrow{COMSh}$ is the vector from the center of mass of the arm to the shoulder.

While this gives a good measure for policies where the arm learns to be static during rest periods, it still sometimes classifies moving arms to be more natural than flexing but resting arms during rest periods. This is because the average of

all frames is taken and if the arm jerks around a lot, this average of that could still be an arm hanging down. To overcome this issue, the average velocity v_A^{rest} of the arm during rest periods is also added:

$$v_A^{rest} = \frac{1}{T} \sum_t \frac{v_{UA} + v_{LA}}{v_{A_{max}}^{rest}} \tag{5.6}$$

v_{UA} and v_{LA} are the upper arm and lower arm velocities obtained by the Unity engine. $v_{A_{max}}^{rest}$ is the maximum velocity value of all parameter settings and agents.

Fig. 5.3 shows the results of the 3CC-r and the instantaneous effort model. Each point denotes a different τ_F value for a model. The models are ordered based on their naturalness measure (lowest first). Models above a median time of 1.2 s learn to hang the arm down due to obtaining more reward from using as little effort as possible compared to the reward obtained from pointing. Models with naturalness values below 0.3 result in unnatural movements and jerky arm behaviour during rest periods. Furthermore, the variance of the results in terms of accuracy increases as is suggested by Fig. 5.3. The sweet spot in which motions are natural, i.e. arm hangs down during rest periods and elbows are kept down during pointing periods, but still accurate lies usually within naturalness values between 3.5 and 3.9 (Fig. 5.3). This is the case for $0.1 \leq \tau_F \leq 0.18$ for the 3CC-r model, and $0.16 \leq \tau_F \leq 0.18$ for the instantaneous effort model. Note how the span of natural but efficient behaviour is larger for the models based on 3CC-r. The plots in Fig. 5.3 show how the 3CC-r models consistently outperform the instantaneous fatigue
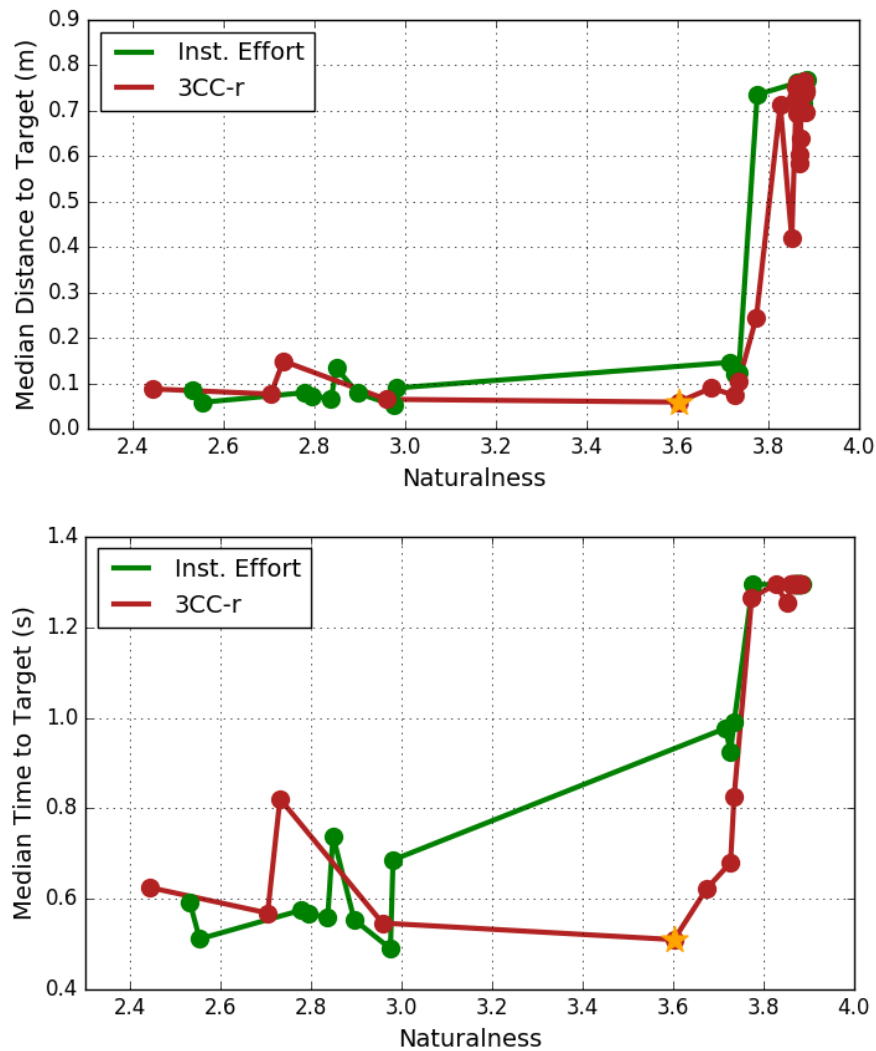
FIGURE 5.3: The evolution of the trade-off between pointing task performance and movement naturalness when sweeping the τ_F parameter in the range $[0, 0.5]$, plotted using both instantaneous torque effort (green) and the 3CC-r model effort (red). For each tested τ_F, the agent is retrained and re-evaluated. The yellow stars indicate the best combinations of both naturalness and pointing task performance. Overall, the 3CC-r yields better combinations of naturalness and pointing task performance, across a range of τ_F.

model in terms of speed and accuracy within this region. Based on the results in Fig. 5.3, we found that $\tau_F = 0.1$ for the 3CC-r model yields the best results in terms of efficiency and naturalness. In the next section this model is used to compare against ground truth perceived fatigue ratings from humans.

5.2   Comparison to Ground Truth Human Data

The synthesized motions obtained from the best presented model are compared against ground truth human data obtained from [17]. Jang et al. [17] use the 3CC model to estimate fatigue ratings based on motion capture data from a Kinect [62] sensor. They compare their results with the subjects' perceived exertion rating using the Borg CR10 [45] scale (see Table 5.1).

To replicate the four conditions in [17], the first two interaction zones shown in Fig. 5.1 are used, and two groups with different rest periods in between the four 60 s pointing periods: [20s, 5s, 15s, 10s] for group 1 and [5s, 10s, 20s, 15s] for group 2. In the following this thesis refers to group 1 and 2 as G1 and G2, and the high and low interaction zones as H and L. Based on the findings of Jang et al. [17], the tasks based on the higher interaction zones should be more fatiguing than the ones based on the lower interaction zones. Furthermore, G1 should feel less fatigued compared to G2, due to a large rest period in the initial period of the task.

Jang et al. [17] use 24 participants in their study of which two were female. Since there was no ground truth data published of each participant's weight and their corresponding maximum torque estimate, this work gauges their subjects in a virtual environment by using average torque and arm weight estimates found in literature. See Table 4.2 and 4.3 for details. Additionally, the presented experiment also uses 22 male, and 2 female (virtual) subjects.
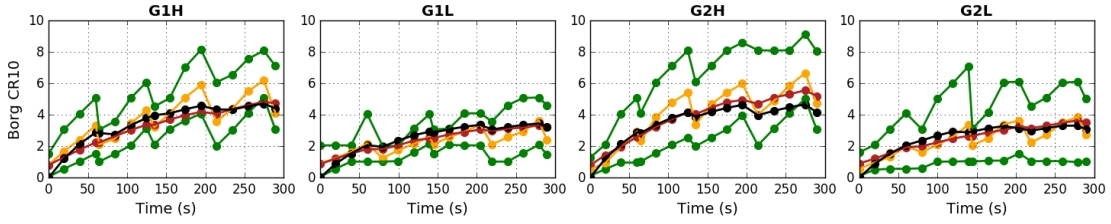
FIGURE 5.4: Results of predicting the Borg CR10 rating. Green: Upper/lower bound of ground truth. Yellow: Average of ground truth. Red: Average 3CC estimate of ground truth computed using motion capture data [17]. Black: Our simulation-based average 3CC-r estimate. Our simulation model yields similar modeling accuracy as [17], but does not require motion capture data.

Similar to [17] a linear relationship between the fatigued motor units M_F obtained from the 3CC-r model and the Borg CR10 scale is assumed, with $\varphi(x) = 0.3 \cdot x$ denoting the linear mapping. To compute M_F for the Borg CR10 estimate, the ratio of magnitude of the torque $\frac{\|T\|}{T_{max}} \cdot 100\%$ is used as a target load for the model.

An overview of the obtained results is shown in Fig. 5.4. The 3CC-r estimates (*black*) on virtual data mostly follow the trend of the 3CC estimates (*red*) from [17] based on human data, as well as the ground truth average Borg CR 10 data (*yellow*). The average root mean squared error (RMSE) between the 3CC estimates from [17] and their average Borg CR10 ground truth data is 0.58, while ours to ground truth is 0.66. The largest deviation of the presented model is in experiment G2-H. This may be attributed to physiological and psychological factors that play into the role of an individual's perceived fatigue rating [17]. As the minimum and maximum values for G2H in Fig. 5.4 suggest, the variance of inter-individual Borg ratings is high for this case. However, despite using no ground truth human data for the presented calculations this work achieves a similar accuracy to [17] on their data.

Borg CR10 Scale

Score	Definition	Note
0	Nothing at all	No arm fatigue
0.5	Very, very weak	Just noticeable
1	Very weak	As taking a short walk
2	Weak	Light
3	Moderate	Somewhat but not hard to go on
4	Somewhat heavy	
5	Heavy	Tiring, not terribly hard to go on
6		
7	Very strong	Strenuous
8		
9		
10	Extremely strong	Extremely strenuous

TABLE 5.1: Borg CR10 scales with verbal commentary.

CHAPTER 6

DISCUSSION

This thesis makes two main contributions to the HCI and ML community: 1) a cumulative fatigue model for Reinforcement Learning of movement tasks 2) an *in silico* method for virtual user testing.

6.1 Cumulative Fatigue for RL

The presented results show that RL-agents trained on a cumulative fatigue model based on biomechical literature, instead of instantaneous joint torques, learn more efficient and natural policies. This can be utilized for new optimization procedures in computer animation and robotics. The developed model is the first one to use cumulative effort in such a way.

6.2 Reliable In Silico Fatigue Estimate

Furthermore, the results confirmed that ground truth human kinematic data is not necessarily needed to obtain reliable fatigue estimates for the presented pointing

task. This is the first method to achieve this. Hence, the presented model could be utilized for a multitude of HCI applications, where human data is not readily available or expensive to record, and open new pathways to virtual user testing. The advantage of such *in silico* methods is that many physical properties can be reliably modeled with standard games and physics engines [20, 50, 63, 64] making the prediction - in theory - more accurate than with non-invasive *in vivo* methods. With this new environments, e.g. effects of fatigue on the moon or under high pressure, could be easily explored.

6.3 Limitations

While the developed model showed overall good results and performance, the following limitations were identified which should be addressed for future work. Similar to [17], the proposed method is based on the assumption that perceived fatigue can be directly deduced from biomechanical information. In reality how-ever, an individual's perceived fatigue can be attributed to a multitude of different factors [17, 34], e.g. through physiological and psychological changes. Previous studies [17, 43, 45, 59] have shown that individuals may experience fatigue and rest differently than others. However, this could be mitigated with extending the agents with models of intrinsic motivation and emotion [16] in future work.

An additional limitation of this work is that for the trained agent to generalize, it must experience the full variance of environments and tasks during training. In this case, the pointing targets have varying settings, but more variation may

be needed for other applications. Furthermore, the learning process itself is also time-consuming and laborious, and needs to be performed independently for each policy. While it takes around 2 hours on an i7 processor to learn pointing, training can take days for other, more complex tasks [49]. Continous control approaches that combine Monte Carlo Tree Search (MCTS) and machine learning have shown to learn complex tasks online in under 1 minute [19, 65] and could be extended with this method to reduce computing time.

Chapter 7

Conclusion and Future Work

This thesis presents a framework for evaluating subjective fatigue only using virtual embodied AI-agents. The agents have been trained on a pointing task using Reinforcement Learning. For the training we compared two different effort models. First, using instantaneous joint torques; second, using a biomechanical cumulative fatigue model. This work shows that the model trained with cumulative fatigue is able to learn more natural and efficient movements. This is believed to be the first work to use cumulative fatigue in such a way. Finally, the best trained model is used to estimate fatigue ratings under various conditions and the results are compared to ground truth human data obtained from previous studies [17]. Overall, the presented model shows comparable results to ground truth Borg CR10 ratings and 3CC-estimates based on motion capture data, without using any human data. To the best of the author's knowledge this is the first work to achieve this.

There are various possibilities for future work of which some have already been mentioned in Section 6.3, and will be discussed in further detail here. A natural extension of this work is to apply it to multiple joints to obtain a full-body biomechanical model. A recent result by [31] has shown that a machine learning approach

which maps an optimal control problem formulated in the muscle-actuation space to an equivalent problem in the joint-actuation torque results in more natural but computationally more efficient movements. Their approach can be incorporated in the presented model to better model maximum joint voluntary torques. For additional accuracy this work can easily be extended for individual muscle groups.

Another possible path for future work lies in improving the 3CC-r model by either learning the task- and joint-specific parameters from kinematic data and Borg CR 10 ratings, or learning the effort functions directly from data. The latter may generelize the cumulative effort model to various tasks by including possible non-linearities [66] and make predictions of avereage Borg CR 10 ratings more accurate. Furthermore, emotion-driven models [16, 67, 68] might further improve in this direction.

In recent years, computational user models [22–24, 42] have provided a fruitful analysis tool for studying user behaviour and fatigue. While these previous methods were not able to generalize well, this thesis presents a novel model which can be easily extended to various tasks by simply defining a reward, state and action space using RL agents. Additionally, the method described is the first one to both synthesize mid-air movements and accurately predict subjective fatigue ratings without real users. Hence, it has the possibility to open various pathways for inexpensively modeling user behaviour under circumstances that may be difficult - if not impossible - to model *in vivo*.

LIST OF FIGURES

LIST OF TABLES

Bibliography

[1] Alan Cooper, Robert Reimann, and David Cronin. *About face 3: the essentials of interaction design.* John Wiley & Sons, 2007.

[2] Pradipta Biswas, Peter Robinson, and Patrick Langdon. Designing inclusive interfaces through user modeling and simulation. *International Journal of Human-Computer Interaction*, 28(1):1–33, 2012.

[3] Gerhard Fischer. User modeling in human–computer interaction. *User modeling and user-adapted interaction*, 11(1-2):65–86, 2001.

[4] Christian Guckelsberger, Christoph Salge, Jeremy Gow, and Paul Cairns. Predicting player experience without the player.: An exploratory study. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 305–315. ACM, 2017.

[5] Antti Oulasvirta. User interface design with combinatorial optimization. *Computer*, 50(1):40–47, 2017.

[6] Antti Oulasvirta, Xiaojun Bi, and Andrew Howes. *Computational interaction.* Oxford University Press, 2018.

[7] Georgios N Yannakakis, Pieter Spronck, Daniele Loiacono, and Elisabeth André. Player modeling. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

[8] Stefan Freyr Gudmundsson, Philipp Eisen, Erik Poromaa, Alex Nodet, Sami Purmonen, Bartlomiej Kozakowski, Richard Meurling, and Lele Cao. Human-like playtesting with deep learning. In *2018 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8. IEEE, 2018.

[9] Alexander Zook, Brent Harrison, and Mark O Riedl. Monte-carlo tree search for simulation-based strategy analysis. In *Proceedings of the 10th Conference on the Foundations of Digital Games*, 2015.

[10] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction.* MIT press, 2018.

[11] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

[12] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

[13] Seunghwan Lee, Moonseok Park, Kyoungmin Lee, and Jehee Lee. Scalable muscle-actuated human simulation and control. *ACM Transactions on Graphics (TOG)*, 38(4):73, 2019.

[14] Samuel J Gershman, Eric J Horvitz, and Joshua B Tenenbaum. Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, 349(6245):273–278, 2015.

[15] Shaghayegh Roohi, Jari Takatalo, Christian Guckelsberger, and Perttu Hämäläinen. Review of intrinsic motivation in simulation-based game testing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 347. ACM, 2018.

[16] Thomas M Moerland, Joost Broekens, and Catholijn M Jonker. Emotion in reinforcement learning agents and robots: a survey. *Machine Learning*, 107 (2):443–480, 2018.

[17] 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*, pages 3328–3339. ACM, 2017.

[18] Roberto A. Montano Murillo, Sriram Subramanian, and Diego Martinez Plasencia. Erg-o: Ergonomic optimization of immersive virtual environments. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 759–771, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4981-9. doi: 10.1145/3126594.3126605. URL `http://doi.acm.org/10.1145/3126594.3126605`.

[19] Joose Rajamäki and Perttu Hämäläinen. Augmenting sampling based controllers with machine learning. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, page 11. ACM, 2017.

[20] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.

[21] 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.

[22] Paul M Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47 (6):381, 1954.

[23] Stuart K Card, Allen Newell, and Thomas P Moran. The psychology of human-computer interaction. 1983.

[24] Kristinn Thórisson and Helgi Helgasson. Cognitive architectures and autonomy: A comparative review. *Journal of Artificial General Intelligence*, 3(2): 1–30, 2012.

[25] Senthilkumar Chandramohan, Matthieu Geist, Fabrice Lefevre, and Olivier Pietquin. User simulation in dialogue systems using inverse reinforcement learning. In *Interspeech 2011*, pages 1025–1028, 2011.

[26] Antti Kangasrääsiö, Kumaripaba Athukorala, Andrew Howes, Jukka Corander, Samuel Kaski, and Antti Oulasvirta. Inferring cognitive models from data using approximate bayesian computation. In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pages 1295–1306. ACM, 2017.

[27] Esther Levin, Roberto Pieraccini, and Wieland Eckert. Using markov decision process for learning dialogue strategies. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, volume 1, pages 201–204. IEEE, 1998.

[28] Xiuli Chen, Gilles Bailly, Duncan P Brumby, Antti Oulasvirta, and Andrew Howes. The emergence of interactive behavior: A model of rational menu

search. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*, pages 4217–4226. ACM, 2015.

[29] Katri Leino, Antti Oulasvirta, Mikko Kurimo, et al. Rl-klm: automating keystroke-level modeling with reinforcement learning. In *IUI*, pages 476–480, 2019.

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

[31] Yifeng Jiang, Tom Van Wouwe, Friedl De Groote, and C Karen Liu. Synthesis of biologically realistic human motion using joint torque actuation. *arXiv preprint arXiv:1904.13041*, 2019.

[32] Matthew Hausknecht and Peter Stone. Deep recurrent q-learning for partially observable mdps. In *2015 AAAI Fall Symposium Series*, 2015.

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

[34] 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.

[35] Robert H Fitts. Cellular mechanisms of muscle fatigue. *Physiological reviews*, 74(1):49–94, 1994.

[36] Chris R Abbiss and Paul B Laursen. Models to explain fatigue during prolonged endurance cycling. *Sports medicine*, 35(10):865–898, 2005.

[37] Benjamin K Barry and Roger M Enoka. The neurobiology of muscle fatigue: 15 years later. *Integrative and comparative biology*, 47(4):465–473, 2007.

[38] 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.

[39] 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.

[40] 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.

[41] 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.

[42] Walter Rohmert. Ermittlung von erholungspausen für statische arbeit des menschen. *European Journal of Applied Physiology and Occupational Physiology*, 18(2):123–164, 1960.

[43] Juan David Hincapié-Ramos, Xiang Guo, Paymahn Moghadasian, and Pourang Irani. Consumed endurance: a metric to quantify arm fatigue of mid-air interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1063–1072. ACM, 2014.

[44] 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, 2002.

[45] Gunnar A Borg. Psychophysical bases of perceived exertion. *Med sci sports exerc*, 14(5):377–381, 1982.

[46] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[47] Mazen Al Borno, Martin De Lasa, and Aaron Hertzmann. Trajectory optimization for full-body movements with complex contacts. *IEEE transactions on visualization and computer graphics*, 19(8):1405–1414, 2012.

[48] 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*, 31(4), 2012.

[49] 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):143, 2018.

[50] Arthur Juliani, Vincent-Pierre Berges, Esh Vckay, Yuan Gao, Hunter Henry, Marwan Mattar, and Danny Lange. Unity: A general platform for intelligent agents. *arXiv preprint arXiv:1809.02627*, 2018.

[51] Rositsa Raikova. A general approach for modelling and mathematical investigation of the human upper limb. *Journal of biomechanics*, 25(8):857–867, 1992.

[52] ISO ISO. 9241-9 ergonomic requirements for office work with visual display terminals (vdts)-part 9: Requirements for non-keyboard input devices (fdis-final draft international standard), 2000. *International Organization for Standardization*, 2000.

[53] Robert J Teather and Wolfgang Stuerzlinger. Pointing at 3d targets in a stereo head-tracked virtual environment. In *2011 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 87–94. IEEE, 2011.

[54] I Scott MacKenzie. Fitts' law as a research and design tool in human-computer interaction. *Human-computer interaction*, 7(1):91–139, 1992.

[55] Marina Hofmann, R Brger, Ninja Frost, Julia Karremann, Jule Keller-Bacher, Stefanie Kraft, Gerd Bruder, and Frank Steinicke. Comparing 3d interaction performance in comfortable and uncomfortable regions. In *Proceedings of the GI-Workshop VR/AR*, pages 3–14, 2013.

[56] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[57] Kourosh Naderi, Joose Rajamäki, and Perttu Hämäläinen. Discovering and synthesizing humanoid climbing movements. *ACM Transactions on Graphics (TOG)*, 36(4):43, 2017.

[58] Patricia A Hageman, Debra K Mason, Kelly W Rydlund, and Scott A Humpal. Effects of position and speed on eccentric and concentric isokinetic testing of the shoulder rotators. *Journal of Orthopaedic & Sports Physical Therapy*, 11(2):64–69, 1989.

[59] Laura A Frey-Law, Andrea Laake, Keith G Avin, Jesse Heitsman, Tim Marler, and Karim Abdel-Malek. Knee and elbow 3d strength surfaces: peak torque-angle-velocity relationships. *Journal of applied biomechanics*, 28(6): 726–737, 2012.

[60] Paolo De Leva. Adjustments to zatsiorsky-seluyanov's segment inertia parameters. *Journal of biomechanics*, 29(9):1223–1230, 1996.

[61] James Peter Fisher, Luke Carlson, James Steele, and Dave Smith. The effects of pre-exhaustion, exercise order, and rest intervals in a full-body resistance training intervention. *Applied Physiology, Nutrition, and Metabolism*, 39(11): 1265–1270, 2014.

[62] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10, 2012.

[63] Erwin Coumans. Bullet physics simulation. In *ACM SIGGRAPH 2015 Courses*, SIGGRAPH '15, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3634-5. doi: 10.1145/2776880.2792704. URL `http://doi.acm.org/10.1145/2776880.2792704`.

[64] Scott L Delp, Frank C Anderson, Allison S Arnold, Peter Loan, Ayman Habib, Chand T John, Eran Guendelman, and Darryl G Thelen. Opensim: open-source software to create and analyze dynamic simulations of movement. *IEEE transactions on biomedical engineering*, 54(11):1940–1950, 2007.

[65] Perttu Hämäläinen, Sebastian Eriksson, Esa Tanskanen, Ville Kyrki, and Jaakko Lehtinen. Online motion synthesis using sequential monte carlo. *ACM Transactions on Graphics (TOG)*, 33(4):51, 2014.

[66] Thomas J Barstow and Paul A Molé. Linear and nonlinear characteristics of oxygen uptake kinetics during heavy exercise. *Journal of Applied Physiology*, 71(6):2099–2106, 1991.

[67] Robert P Marinier and John E Laird. Emotion-driven reinforcement learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 30, 2008.

[68] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 16–17, 2017.