MyoSuite:
A contact-rich simulation suite for musculoskeletal motor control

Vittorio Caggiano*
Meta AI Research, New York, NY, USA
CAGGIANO@GMAIL.COM

Huawei Wang*
and Guillaume Durandau
and Massimo Sartori
University of Twente, the Netherlands

Vikash Kumar
Meta AI Research, Pittsburgh, PA, USA

Editors: R. Firoozi, N. Mehr, E. Yel, R. Antonova, J. Bohg, M. Schwager, M. Kochenderfer

Abstract
Embodied agents in continuous control domains have been traditionally exposed to tasks with limited opportunity to explore musculoskeletal details that enable agile and nimble behaviors in biological beings. The sophistication behind bio-musculoskeletal control not only poses new challenges for the learning community but realizing agents embedded in the same perception-action loop that the human sensory-motor system solves can also have a far-reaching impact in fields of neuro-motor disorders, rehabilitation, assistive technologies, as well as collaborative-robotics.

Human biomechanics is a complex multi-joint-multi-actuator musculoskeletal system. The sensory-motor system relies on a range of sensory-contact rich and proprioceptive inputs that define and condition motor actuation required to exhibit intelligent behaviors in the physical world. Current frameworks for studying musculoskeletal control do not include at the same time the needed physiological sophistication of the musculoskeletal systems and support physical world interaction capabilities. In addition, they are neither embedded in complex and skillful motor tasks nor are computationally effective and scalable to study motor learning in the timescale that current learning paradigms require.

To realize a platform where physiological detail and challenges behind human motor control can be investigated, we present a suite of physiologically accurate biomechanical models of elbow, wrist, and hand, with physical contact capabilities which allow complex and skillful contact-rich real-world tasks. The implemented motor tasks provide a great variability of control challenges: from simple postural control to skilled hand-object interactions involving tasks like turning a key, twirling a pen, rotating two balls in one hand, etc. Finally, by supporting physiological alterations in musculoskeletal geometry (tendon transfer), assistive devices (exoskeleton assistance), and muscle contraction dynamics (muscle fatigue, sarcopenia), we present real-life tasks with temporal changes, thereby exposing realistic non-stationary conditions in our tasks which most continuous control benchmarks lack.

Project Webpage: https://sites.google.com/view/myosuite
Keywords: Musculoskeletal model, motor control, neuro-AI

© 2022 V. Caggiano*, H. Wang*, G. Durandau, M. Sartori & V. Kumar.
Figure 1: MyoSuite tasks – **Finger**: joint pose (A), tip reach (B). **Elbow**: joint pose (C), exo-assisted (D). **Hand**: key turn (E), joint pose (F), tips reach (G), object reposition (H), pen-twirl (I), baoding balls (J).

1. Introduction

Data-driven learning paradigms have enabled rapid progress in multiple fields such as vision (Deng et al. (2009)), natural language processing (Wang et al. (2018)), speech (Panayotov et al. (2015)), among others. While there has been progress, the field of embodied AI (EAI) is still awaiting its breakthrough moments and real-world impacts. Traditionally, challenges in the EAI domain have been proposed by exposing agents to simplified stationary problems that, although have pushed the field forward, do not translate into real-world advancements. One of the root causes is that real-world problems are complex and non-stationary. The human central nervous system, on the other side, can easily handle complex tasks in non-stationary conditions (Latash (2012)) by building control schemes that integrate and model proprioceptive as well as sensory information and transform them into optimal (Shadmehr and Krakauer (2008); Scott (2012)) and adaptable (Wolpert and Ghahramani (2000)) control policies. Such neuro-control challenges that the human central nervous system can seamlessly handle can pose as the next frontier for algorithmic paradigms in the EAI. Furthermore, any development in such a critical problem space can translate into impactful real-world advancements in critical fields like neuromechanics, physiotherapy, rehabilitation, as well as robotics.

In comparison to virtual and robotic problems, musculoskeletal control bolsters complexity (task, non-stationarity, as well as dimensionality) that EAI algorithms need to handle. Muscles have third order dynamics and can only generate forces in one direction (pulling). They undergo changes in their force-generating ability depending on their operating length, contractile velocity, or fatigue state (Arnold and Delp (2011)) leading to non-stationarity behaviors. Moreover, muscles may undergo structural changes in response to exercises, aging (e.g., sarcopenia) which further alters their contractile properties over time (Kirkendall and Garrett (1998)). Surgical interventions and assistive devices e.g., exoskeleton, require re-adjustment of their nominal behaviors. In addition, the muscle control space is high dimensional (the number of muscles exceeds the number of human joints - about 600 muscles to control about 300 joints), redundant (multiple muscles act on the same joint), and multi-articular (muscles very often act on multiple joints) (Hirashima and Oya (2016));
As a result, in addition to the non-stationary challenges, the overall system also suffers from the ‘curse of dimensionality’ (Bernstein (1966)), which is less common in joint level control, which is typical in robotics (Wolpert and Ghahramani (2000)).

The machine learning community has made major advancements by defining and competing on established benchmarks. In the EAI domain, OpenAI-Gym (Brockman et al. (2016)) and DmControl (Tassa et al. (2018)) are the de-facto benchmarks for behavior synthesis. These benchmarks however are not suitable for real-world problems. First, they consist of mostly simple problems and are largely already solved. Second, those benchmarks have very limited capabilities to test the adaptability of an agent in response to non-stationary environment changes. Usually, non-stationary settings have been investigated via contrived examples - removing links (Nagabandi et al. (2018)) or adding spurious joints (Gupta et al. (2017); Devin et al. (2017)). There is a dire need for new benchmarks in EAI that expose realistic challenges to the algorithmic paradigms, are embedded closely in the real world, and can translate into real-world impact.

Previous attempts at establishing realistic musculoskeletal tasks as benchmarks (Song et al. (2020)) were narrowly defined. They favored bigger and functionally relevant muscle groups e.g. legs (Hamner et al. (2010); Sartori et al. (2013); White et al. (1989); Ackermann and Van den Bogert (2010)) and arms (Delp et al. (2007); Seth et al. (2018); Saul et al. (2015a); McFarland et al. (2019); Lee et al. (2015); Saul et al. (2015b)). These attempts relied on physics-based musculoskeletal simulation engines such as OpenSim (Seth et al. (2018)), AnyBody (Damsgaard et al. (2006)) and SIM that although are widely used for simulating human neural-mechanical control, human robot interaction, and rehabilitation are computationally expensive (simulating large number of muscle is intractable) and provide limited support for contact rich interactions with their environments (see Table 1). While, physics engines used in the robotic field (PyBullet Coumans and Bai (2016), MuJoCo Todorov et al. (2012), IsaacGym Makoviychuk et al. (2021), RaiSim Hwangbo et al. (2018), and Dart Lee et al. (2018)) are relatively more efficient and support contact interactions, but lack adequate support for modeling anatomical and functionally validated musculoskeletal models (see Table 1).

In order to expose the community to the exciting challenges presented by the musculoskeletal control, we present a physiologically realistic and computationally efficient framework: MyoSuite.

**Our contributions:**

- We developed a set of musculoskeletal models (from simple one joint to complex full hand) that are physiologically accurate, several orders of magnitude faster than the state of art musculoskeletal simulators (Ikkala and Hämäläinen (2020); Erez et al. (2015)), and support full contact dynamics.

Table 1: Physic simulators. MyoSuite provides physiologically realistic and computationally efficient musculoskeletal models in the MuJoCo physics engine.

<table>
<thead>
<tr>
<th>Simulator</th>
<th>Contact Dynamics</th>
<th>Muscle Model</th>
<th>Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>MuJoCo(MyoSuite)</td>
<td>✓</td>
<td>✓</td>
<td>↑</td>
</tr>
<tr>
<td>OpenSim</td>
<td>✓</td>
<td>✓</td>
<td>↓</td>
</tr>
<tr>
<td>AnyBody</td>
<td>✓</td>
<td>✓</td>
<td>↓</td>
</tr>
<tr>
<td>SIMM</td>
<td>✓</td>
<td>✓</td>
<td>↓</td>
</tr>
<tr>
<td>PyBullet</td>
<td>✓</td>
<td>✓</td>
<td>↑</td>
</tr>
<tr>
<td>Dart</td>
<td>✓</td>
<td>3</td>
<td>↑</td>
</tr>
<tr>
<td>RaiSim</td>
<td>✓</td>
<td>3</td>
<td>↑</td>
</tr>
</tbody>
</table>

1. With some exceptions like HandManipulateEgg and HandManipulatePen (Plappert et al. (2018)) or in-hand manipulation of real-world objects (OpenAI et al. (2019); Huang et al. (2021); Chen et al. (2021))
2. Contains a basic implementation of a muscle model that MyoSuite builds upon
3. Efforts have been made to add muscle models in both DART Lee et al. (2019) and RaiSim Younguk et al. (2018)
• We designed a family of 9 realistic dexterous manipulation tasks (from simple posing to simultaneous manipulation of two Baoding balls) using these models. These task families take inspiration from the state of art robotic dexterous manipulation results (Rajeswaran et al. (2018a); Nagabandi et al. (2020); Andrychowicz et al. (2020)).

• MyoSuite supports physiological alterations in musculoskeletal geometry (tendon transfer, exoskeleton assistance) and muscle contraction dynamics (muscle fatigue, sarcopenia) to expose real-life tasks with temporal changes, thereby subjecting self-adapting algorithms to the realistic non-stationary challenges under continuous control settings.

• In its present form MyoSuite consists of 204 tasks: 9 task-families x 2 difficulty level (easy, hard) x 3 reset conditions (fixed, random, and none) x 8 (or 4) combinations of non-stationarity variations. We present baselines on MyoSuite tasks outlining its features and complexities.

2. Preliminaries

Behavior synthesis for embodied agents can be formulated as a sequential decision-making problem where the goal is to generate a coordinated sequence of trajectories that allows the agent to achieve desired outcomes. Next we outline how behavior synthesis, for musculoskeletal systems in particular, can be formulated using Markov Decision Processes (MDP) formulation under the Reinforcement Learning (RL) paradigm.

2.1. MDP formulation

Per usual notation MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, T, \mathcal{R}, \rho, \gamma)$. $\mathcal{S} \subseteq \mathbb{R}^n$ and $\mathcal{A} \subseteq \mathbb{R}^m$ represent the continuous state and action spaces respectively. The unknown transition dynamics is described by $s' \sim T(.|s, a)$. $\mathcal{R} : \mathcal{S} \rightarrow [0, R_{\text{max}}]$, $\gamma \in [0, 1)$, and $\rho$ represents the reward function, discount factor, and initial state distribution respectively. Policy is a mapping from states to a probability distribution over actions, i.e. $\pi : \mathcal{S} \rightarrow P(A)$, which is parameterized by $\theta$. The goal of the agent is to learn a policy $\pi^*_\theta(a|s) = \arg\max_{a} \mathcal{J}(\pi, \mathcal{M})$, where $\mathcal{J} = \max_{\theta} \mathbb{E}_{s_0 \sim \rho(s), a \sim \pi(\cdot|s)} \left[ \sum_t R(s_t, a_t) \right]$ i.e. the expected sum of discounted rewards in an episodic setting. Policy gradient algorithms (Silver et al. (2014)), TD-learning based algorithms, such as Q-learning (Watkins and Dayan (1992)), SARSA (Sutton (1996)), actor-critic based methods (Konda and Tsitsiklis (2000)), etc. can be leveraged to optimize $\mathcal{J}$ to generate behaviors.

2.2. MDP formulation for Robotic systems vs. Musculoskeletal systems

**Robotic systems**: Under our MDP characterization $\mathcal{M}$, state space $\mathcal{S}$ consists of $\{\text{joint position, joint velocity}\}$, action space $\mathcal{A}$ consists of actuator’s $\{\text{position/velocity/torque demands}\}$ and samples for policy optimization, are gathered either directly from the real world transition $T_{\text{real}}$, or via physics simulation engines $T_{\text{sim}}$ (Todorov et al. (2012); Coumans and Bai (2016); Lee et al. (2018)).

**Musculoskeletal systems**: Under the MDP characterization $\mathcal{M}$, state space $\mathcal{S}$ consists of $\{\text{muscle–tendon length, muscle–tendon velocity, muscle activations}\}$, action space $\mathcal{A}$ consists of ac-

---

4. To facilitate investigation in reset-free algorithms such as Gupta et al. (2021)
tor’s \{ \alpha - \textit{Motoneurons signals} \}^5 \) samples for policy optimization, are gathered from the physics simulation engines \( T_{sim} \) controlled via muscle actuators.

Unlike robotic systems, which can be viably investigated by drawing samples from \( T_{sim} \) or \( T_{real} \), owing to the limitations of live experimentation, musculoskeletal simulations \( T_{sim} \) have been the cornerstone of most investigation and understanding behind biological motor control. In contrast to robotics where system identification for comprehensive simulations can be solved comprehensively, mathematical models for biological systems are based on sparse data available from cadavers (e.g. not all parameters can be identified leading to inherent uncertainty).

3. MyoSuite

Unlike robotic counterparts, which are double acting and usually have one-to-one relationships between joints and actuators, tendons and muscles in musculoskeletal systems act via contraction (pull-only) and span multiple joints inducing strong coupling between them. Muscles become fatigued with extended usage. Tendons transfer muscle forces to bones while serving as temporary energy storage units for motion efficiency. These details, while complex, conceal within themselves the ingredients of effective motor control in biological systems. To facilitate investigations in these details, we present the “MyoSuite” – a collection of physiologically accurate musculoskeletal models (Section 3.1) and physically realistic contact-rich tasks (Section 3.2) of varying complexity.

3.1. Musculoskeletal models

Musculoskeletal models are commonly modeled as a 3rd order system which contains first (or second-order) muscle activation and contractile dynamics (full detail in Online Appendix/Models) as well as the second-order skeletal dynamics. Our models are developed using MuJoCo physics engine Todorov et al. (2012) via a thorough investigation of well studied existing models (Delp et al. (2007); Seth et al. (2018); Saul et al. (2015a); McFarland et al. (2019); Lee et al. (2015); Saul et al. (2015b)) and functional studies (Wu et al. (2008)). We started from OpenSim models (Lee et al. (2015); McFarland et al. (2019); Saul et al. (2015a)) of the arm and hand which are widely used in fields of human neural-mechanical control, human-robot interaction, and rehabilitation. In order to implement those models in MuJoCo, we developed a pipeline Wang et al. (2022) to perform geometry transformations of bones and muscles attachment, moment arm optimization, and muscle force optimization. After rigorous modeling and validation (Section 4.1), we built three physiologically realistic models (Figure 2) of varying complexities.

**MyoFinger:** A simplified and intuitive model (based on Xu et al. (2012)) of a 4 Degree of Freedom (DoF) finger (MyoFinger, Figure 2A), which is actuated through a series of 5 simplified antagonistic muscle-tendon units. We also provide its robotic counterpart with simple torque actuators to facilitate comparative investigations.

5. Motorneurons are the final neuronal stage that connects the central nervous system to the muscles.
**MyoElbow**: A model of 1 DoF human elbow joint (Figure 2B). MyoElbow model is based on OpenSim’s default testing arm model (Delp et al. (2007); Seth et al. (2018)) and actuated using multiple agonist/antagonist pairs (3 flexors and 3 extensors).

**MyoHand**: The dexterous human hand is comprised of 29 bones, 23 joints, and 38 muscles-tendon units (see Online Appendix/Models for a detailed description of the hand muscles). This forearm-wrist-hand model (MyoHand, Figure 2C) combined and extended popular OpenSim models: MoBL - human upper extremity model (Saul et al. (2015a) McFarland et al. (2019)) - and the 2nd-Hand - for hand and fingers models (Lee et al. (2015)).

### 3.2. Tasks

Leveraging these musculoskeletal models, we built a series of tasks (see Figure 1) of varying difficulty. The task difficulty is varied along two axes: task-complexity, and task-non-stationarity. Task-complexity has two variations (difficulty - easy/hard, and Reset - Fix/Random/None), and task-non-stationarity has 8 (or 4 if tendon-transfer and exoskeleton assistance are not possible) variations. In total, in its current form MyoSuite consists of 204 tasks (Table 2): 9 task-families x 2 difficulties-levels x 3 Resets x 8 (or 4) combinations of non-stationarity variations. We provide a complete description of the tasks and task-complexities in Online Appendix/Tasks. Next we detail various non-stationarities that are supported in MyoSuite tasks.

<table>
<thead>
<tr>
<th>Complexity</th>
<th>Non-Stationarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Easy/Hard</td>
<td>Reset</td>
</tr>
<tr>
<td>Finger Joint Pose</td>
<td>✓ ✓ F/R/N ✓ ✓ ✓ ✓ ✓</td>
</tr>
<tr>
<td>Finger Tip Reach</td>
<td>✓ ✓ F/R/N ✓ ✓ ✓ ✓</td>
</tr>
<tr>
<td>Elbow Joint Pose</td>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
</tr>
<tr>
<td>Hand Key Turn</td>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
</tr>
<tr>
<td>Hand Joints Pose</td>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
</tr>
<tr>
<td>Hand Tips Reach</td>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
</tr>
<tr>
<td>Hand Object Hold</td>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
</tr>
<tr>
<td>Hand Pen Twirl</td>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
</tr>
<tr>
<td>Hand Baoding Balls</td>
<td>✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓</td>
</tr>
</tbody>
</table>

Table 2: MyoSuite tasks with (a) complexity variations (easy/hard and Reset - Fix, Random, None), and (b) non-stationarities variations (None, Sarcopenia, Fatigue, Tendon-transfer and exoskeleton).

### 3.3. Realistic non-stationary task-variations

Muscle properties are constantly changing. These changes can be instantaneous - like for musculoskeletal injury or surgery - or can vary over a short time frame - like muscle fatigue or exoskeleton assistance. To study neuromuscular adaptation to non-stationarities due to these changes during the real work-life scenario, four different variations in muscle properties have been included: Sarcopenia, Fatigue, Tendon Transfer, and Exoskeleton assistance.

**Sarcopenia**: Sarcopenia is a muscle disorder that occurs commonly in the elderly population (Cruz-Jentoft and Sayer (2019)) and characterized by a reduction in muscle mass or volume. The peak in grip strength can be reduced up to 50% from age 20 to 40 (Dodds et al. (2016)). We modeled sarcopenia for each muscle as a reduction of 50% of its maximal isometric force.

6. We also include an Opponens Pollicis muscle for the critical role it has in manual dexterity (Karakostis et al. (2021))
Fatigue: Muscle Fatigue is a short-term (second to minutes) effect that happens after sustained or repetitive voluntary movement. It has also been linked to traumas e.g. cumulative trauma disorder (Chaffin et al. (2006)). A dynamic muscle fatigue model (Ma et al. (2009)), that builds on the idea that different types of muscle fiber have different contributions and resistances to fatigue (Vøllestad (1997)), was developed and integrated into the modeling framework. See Online Appendix/Non-Stationary for details on the implementation of the fatigue model.

Tendon tear/ tendon transfer via surgery: While muscle fatigue and sarcopenia affects all muscles, accidents can lead to damage of a subset of muscle-tendon units (termed as tendon-tear). Tendon transfer surgery allows redirecting the application point of muscle forces from one location to another (see Figure 3). It can be used to regain functional control of a joint or limb motion after injury. One such tendon transfer procedure is relocation of Extensor Indicis Proprius (EIP) to replace the Extensor Pollicis Longus (EPL) (Gelb (1995)). Rupture of the EPL can happen after a broken wrist. It results in a loss of control over the thumb extension. We introduce a physical tendon transfer where the EIP application point of the tendon was moved from the index to the thumb and the EPL was removed (see Figure 3).

Exoskeleton assistance: Exoskeleton assisted rehabilitation is becoming common (Jezernik et al. (2003)) and demonstrating significant benefit (Nam et al. (2017)). To study effective modulation of exoskeletal assistance strategies, we modeled an exoskeleton for the elbow using an ideal actuator and the addition of two supports with a weight of 0.101 Kg for the upper arm and 0.111 Kg on the forearm (Figure 1-D, Wang et al. (2022)). The assistance given by the exoskeleton was a percentage of the biological joint torque, this was based on the neuromusculoskeletal controller presented in Durandau et al. (2019).

4. Experiments

4.1. Models validation

We compared the MuJoCo models against the correspondent OpenSim models. Muscle moment arm root mean square (RMS) differences between the MuJoCo model with respect to the OpenSim model were 0.044 ± 0.09% for the elbow and 0.38 ± 0.57% for the hand model. Also, the RMS error in forces was 2.2 ± 1.4% $F_{max}$ (OpenSim peak force) for the elbow model and 4.1 ± 2.0% $F_{max}$ for the hand model. Those errors indicate that the MuJoCo models are anatomically and dynamically similar to the SOTA OpenSim model. Forward simulations showed that MuJoCo models can be several orders of magnitude faster than OpenSim (see Figure 4, from 70x to 4000x faster). By simulating an elbow model (6 muscles) where we iteratively replicated all muscles, it was possible to observe that the OpenSim computing time increased exponentially while the MuJoCo did not (see Figure 4). This increase in efficiency is mostly the result of a simplified implementation of the muscle actuator in
MuJoCo, which allows faster and more stable simulations. In summary, training/studies (Section 4.2) that in OpenSim will take years to simulate can be performed in MuJoCo in a few days!

4.2. Baselines

We present baseline results obtained using Natural Policy Gradient (NPG, Kakade (2001)) for a subset (i.e. stationary, easy and hard, with fixed resets) of the conditions available in MyoSuite. We chose this algorithm owing to its recent successes in solving complex robotic dexterous manipulation tasks (Rajeswaran et al. (2018b)). In Figure 5, we show success rates for the different tasks up to 5M samples. More complex tasks e.g. Baoding balls can be solved, albeit with much higher sample complexity (∼70M samples). In Figure 6A, it is shown a sequence of snapshots of the solution of the key turning task. It is possible to see how the index and thumb activation is functional to the effective rotation of the key. The Pen Twirl task (Figure 6B) requires effective coordination between the wrist and the hand muscles to express the full dexterity of the hand while effectively maneuvering the pen (blue) to the desired goal (green) via a sequences of contacts. Finally, the task of baoding balls (Figure 6C) pushes the dexterity of the hand to its limits by requiring policies to learn simultaneous coordination of not one but two objects. This task is quite challenging to learn as any miss coordination results in catastrophic failures.

4.3. Intrinsic non-stationary perturbations

While extrinsic perturbation can be easily added to each task, we focus our next investigation on behavioral acquisition and adaptation in response to intrinsic non-stationary perturbations (Sec 3.3) that are modelled after various real-life scenarios in MyoSuite.

4.3.1. Sarcopenia (Muscle Degeneration)

We tested a policy trained on a 1D elbow flexion task to reach random targets in its operational space on alternated movements between points A and B (see Figure 7A).

First, we study how sarcopenia (muscle weakness) affects the control of movement. In presence of Sarcopenia (Figure 7B) the Brachioradialis (BRA) - which in normal conditions does not need support from other muscles - needs stronger activations from synergistic muscles (BICLong - biceps longus - and BICShort - biceps short) to solve the task.
4.3.2. Fatigue (Muscle exhaustion)

Second, we investigated the effect of muscle fatigue (introduced in Section 3.3). In this case, the loss of muscle power is progressive over time. In the same alternated movements elbow tasks, we observe gradually increasing contribution of synergistic muscles to compensate for the muscle force loss (Figure 7C).

Figure 7: Control Policy with muscle sarcopenia (weakness) and fatigue during an alternated reaching task between two targets. (A) Experiment design. (B) Effects of Sarcopenia – Top-row: muscle activations for 6 arm muscles (3 agonists and 3 antagonists). Middle-row: muscle activations in presence of sarcopenia. All synergistic flexor muscles (BRA - brachioradialis, BICLong - Biceps Long, BICShort - Biceps short) increase their activation to compensate for the reduced force. Bottom-row: a trace of the activation only for the Biceps Long muscle. (C) Effects of Fatigue – Bottom-row: synergistic muscles progressively increase activations to compensate for Fatigue.

4.3.3. Accidents (Tendon tear)

Next, in order to study the effects of tendon tear, a policy was first trained to solve the Key-Turn task and then challenged with the selective damage (tear) of different thumb muscles (Figure 8). While there are redundancies (not every tear crippled ability), we found that FPL and OP are critical for the correct solution of the key task. In absence of OP the FPL is able to compensate. But when OP is torn FPL cannot compensate resulting in reduced key rotations.

4.3.4. Control in presence of Tendon transfer

Finally, a tendon transfer - EIP muscle was routed to replace the EPL muscle - was performed to test the ability of the policies to compensate for action re-mapping due to tendon surgery. After the surgery, the major thumb muscles needed to compensate for the different activation space (Figure 9 and, a previously trained policy was unable to solve the task. Indeed it was necessary an extensive additional training to control the thumb after tendon transfer (Online Appendix/Solutions). This is typically observed in patient undergoing extensive physiotherapeutical sessions to re-learn the control of the thumb (Wangdell et al. (2016)).

5. Discussion and Conclusions

Here, we have proposed a new framework of physiologically realistic and computationally efficient models and tasks to study human motor control. The proposed models include highly skilled ma-
nipulations and realistic non-stationarities to simulate real-life scenarios such as muscle fatigue, sarcopenia, and tendon transfer. This benchmark will provide biologically relevant problems where task success and physiological representations might differ. Actually, because activation of the musculoskeletal models can be directly related to a (normalized) intramuscular activity recorded in human subjects, it will be possible to validate experimentally the learned solutions.

The basic policies trained already showed physiologically relevant behaviors. First, we observe an automatic adaptation of muscles to leverage antagonistic effects of flexors and extensors. This was clearly visible with the elbow model where those effects could be more easily isolated. Second, co-contractions evolve naturally to compensate for changes in muscle properties e.g. sarcopenia and fatigue. Nevertheless, those compensations are not enough to replace muscles like the Opponens Pollicis which has unique functions for hand manipulations.

Importantly, the implemented models are only the initial iteration on approximating the musculoskeletal system that will need further development and validation. Indeed, both the tasks and physiological changes in muscle properties are a subset of the possible changes that can be considered.

All in all, this work can facilitate cross-pollination and catalyzation of new ideas between different communities. The ML community will have more challenging tasks with 3rd order dynamics (uncommon in ML/RL benchmarks) and physiological realistic non-stationarity provides. The biomechanics community will have an additional platform where contact-rich interactions at scale can be studied. Finally, the robotic community will have a biological system to develop strategies for. Overall, this work will allow more realistic simulation allowing in silico trials of humans, robots, and their interaction.
Acknowledgments

H.W., G.D., and M.S. received funding by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme as part of the ERC Starting Grant INTERACT (Grant No. 803035) as well as part by the Horizon 2020 ICT-10 Project SOPHIA (871237).

References


