Proceedings of Machine Learning Research 220:233–250, 2023 NeurIPS 2022 Competition Track

MyoChallenge 2022: Learning contact-rich manipulation using a musculoskeletal hand

Vittorio Caggiano* *Meta AI, USA*
Guillaume Durandau* *McGill University, Canada*
Huwawei Wang *University of Twente, Netherlands*
Alberto Chiappa, Alexander Mathis *EPFL, Switzerland*
Pablo Tano, Nisheet Patel, Alexandre Pouget *University of Geneva, Switzerland*
Pierre Schumacher, Georg Martius *Max Planck Institute for Intelligent Systems, Germany*
Daniel F.B. Haeufle *Hertie Institute for Clinical Brain Research, Germany*
Yiran Geng, Boshi An, Yifan Zhong, Jiaming Ji, Yuanpei Chen, Hao Dong, Yaodong Yang *Peking University, China*
Rahul Siripurapu, Luis Eduardo Ferro Diez Michael Kopp *IARAI, Austria*
Vihang Patil *JKU, Austria*
Sepp Hochreiter *JKU & IARAI, Austria*
Yuval Tassa *Deepmind, UK*
Josh Merel *Meta Reality Labs, USA*
Randy Schultheis *Meta AI, USA*
Seungmoon Song *Northeastern University, USA*
Massimo Sartori *University of Twente, Netherlands*
Vikash Kumar *Meta AI, USA*

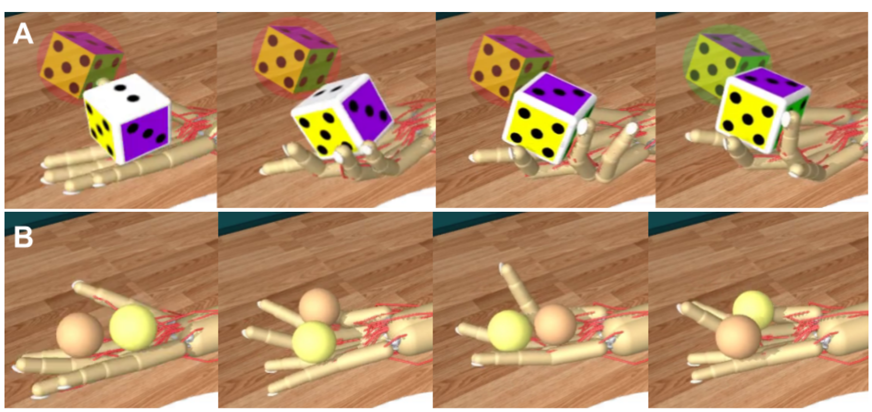


Figure 1: Tasks for *MyoChallenge*:: Track-A: **Die reorientation**, Track-B: **Baoding ball**.

* co-first

Abstract

Manual dexterity has been considered one of the critical components for human evolution. The ability to perform movements as simple as holding and rotating an object in the hand without dropping it needs the coordination of more than 35 muscles which act synergistically or antagonistically on multiple joints. This complexity in control is markedly different from typical pre-specified movements or torque based controls used in robotics. In the *MyoChallenge* at the NeurIPS 2022 competition track, we challenged the community to develop controllers for a realistic hand to solve a series of dexterous manipulation tasks. The MyoSuite framework was used to train and test controllers on realistic, contact rich and computation efficient virtual neuromusculoskeletal model of the hand and wrist. Two tasks were proposed: a die re-orientation and a boading ball (rotation of two spheres respect to each other) tasks. More than 40 teams participated to the challenge and submitted more than 340 solutions. The challenge was split in two phases. In the first phase, where a limited set of objectives and randomization were proposed, teams managed to achieve high performance, in particular in the boading-ball task. In the second phase as the focus shifted towards generalization of task solutions to extensive variations of object and task properties, teams saw significant performance drop. This shows that there is still a large gap in developing agents capable of generalizable skilled manipulation. In future challenges, we will continue pursuing the generalizability both in skills and agility of the tasks exploring additional realistic neuromusculoskeletal models.
Challenge Webpage: https://sites.google.com/view/myochallenge

Keywords: Reinforcement learning, Neuromusculoskeletal control, hand and wrist, manipulation

1. Introduction

Intelligence is thought to be linked to movements. Humans drew a competitive edge over other species when they started effectively changing the environment to their advantage. Manual dexterity and prehension, which are linked to the effective production and use of tools, are one of the most marked capabilities of humans (Karakostis et al., 2021). The effectiveness of humans to realize complex yet agile movement is deeply rooted in the specifics of simultaneous control of muscle subgroups. Musculoskeletal systems consist of bones of various lengths connected together via redundant numbers of skeletal muscles and tendons. The neuromuscular control of the human body is characterized by a high-dimensional space (involving about 600 muscles to control around 300 joints), as well as being both redundant (i.e., multiple muscles act on the same joint) and multi-articular (i.e., a single muscle often acts on multiple joints). Tendons and muscles in the musculoskeletal systems act via contraction (pull-only). Tendons transfer muscle forces to bones while serving as temporary energy storage units for motion efficiency. This complexity of the musculoskeletal system conceals the principles of effective motor control in biological systems.

The biomechanics community has been utilizing simulations of the musculoskeletal system to reproduce mechanical and functional movements. Still, controlling simulated musculoskeletal models to perform skilled and agile movements has been challenging (Valero-Cuevas, 2005; Song and Geyer, 2015; McFarland et al., 2022). This control problem has been mostly approached with the use of controllers that do not use data-driven methods e.g. optimal control(Hansen et al., 2003) or reflex-based control(Song and Geyer, 2013), with limitations on the generalizability of the solution.

Machine learning and data driven methods have been shown to be particularly suitable to address complex, over-redundant and highly nonlinear dynamics (Wochner et al., 2022). Indeed, Deep Reinforcement Learning applied to physiologically realistic models allowed point reaching (Fischer et al., 2021; Joos et al., 2020; Caggiano et al., 2022b), simple locomotion (Song et al., 2021; Kidziński et al., 2018, 2020), to reproduce or mimic naturalistic behaviours (Peng et al., 2018; Lee et al., 2019) and to generate gaits (Park et al., 2022; Barbera et al., 2021). Nevertheless, the complexity of hand object manipulation offers a complete new set of challenges. In addition to the high-dimensional control space and the complex dynamic of the muscle-tendons, hand dexterity requires coordination of multiple fingers and wrist while maintaining a discontinuous contact with the object(s) being handled.

The question that we want to address with this challenge is: *Can we achieve human level dexterity and generalization in physiological digital twins?*

MyoChallenge proposes to endow control over a realistic musculoskeletal hand model to solve complex dexterous tasks, pushing its dexterity to the limits by requiring policies to learn simultaneous coordination of up to two objects. Previous NeurIPS competition on musculoskeletal control – NeurIPS 2019: Learn to Move - Walk Around (Song et al., 2021), NeurIPS 2018: AI for prosthetics (Kidziński et al., 2020), and NeurIPS 2017: Learning to Run (Kidziński et al., 2018) – focused on controlling legs and locomotion with models with at most 22 muscles. *MyoChallenge*, exploits the unique features of a human hand to manipulate objects that, as opposed to legs models used for walking, offer very different challenges: a greater number of muscles in the hand i.e. 39 muscles in the hand vs 22 (11 per each leg) muscles for the legs, greater number of degrees of freedom i.e. 23 in the hand vs 8 for the legs (4 per each leg), finger muscles span more joints than leg muscles i.e. up to 4 joints for each individual finger, continuous monitoring of objects with their own dynamics, and multiple and discontinuous hand-object interaction.

In order to handle the above complexity, *MyoChallenge* leverages MyoSuite[1] - an open-source framework which implements computational biomechanical models and allows muscle-driven simulations of these models to solve skilled tasks (Caggiano et al., 2022a). MyoSuite offers physiologically accurate musculoskeletal full hand models (Wang et al., 2022a) in a framework that is several orders of magnitude (from 70x to 4000x) (see Figure 7 in (Caggiano et al., 2022b)) faster than the state of art musculoskeletal simulators (Ikkala and Hämäläinen, 2022; Erez et al., 2015) used in previous challenges. MyoSuite also support full contact dynamics, which most competing alternatives lack, to enable contact rich manipulation behaviors.

2. MyoChallenge: Task and evaluations

The dexterity of the human hand facilitates an expressive array of manipulation skills required for almost every activity of daily living. Such complex behaviors are also notoriously difficult to synthesize: They require finely balancing contact forces, breaking and reestablishing contacts repeatedly, and maintaining control of unactuated objects. *MyoChallenge*

1. https://sites.google.com/view/myosuite

meant to provide a series of tasks to challenge skilled in-hand manipulation. Here, we present the rationale behind the tasks (Sec. 2.1), the actuation and observation space of the hand model (Sec. 2.2) and finally the tasks proposed (Sec. 2.3).

2.1. Design philosophy

In-hand manipulation skills offer a testbed for dexterity as it is necessary for the hand to continuously interact and handle one or multiple objects. For this, we explored two different aspect of in hand manipulation: complete 3D rotation which requires delicate coordination of various muscles to reorient an object across different axis without dropping it and, simultaneous rotation of two object over the palm, which requires both dexterity and coordination to achieve relative rotation of the balls around each other without dropping them. Furthermore, we also seek generalization in presence of randomization of the initial conditions or goal, changes in physical attributes or in presence of non-stationarity. This was done with increasing difficulty over time over two distinctive phases.

2.2. MyoHand: hand and wrist model

The neuromusculoskeletal model *MyoHand* used in *MyoChallenge* consists of 29 bones, 23 joints, and 39 muscles-tendon units (see table 5.6). This forearm-wrist-hand model was based on two widely used OpenSim models: the MoBL human upper extremity model (Saul et al., 2015) (McFarland et al., 2019) and the 2nd-Hand (for hand and fingers) model (Lee et al., 2015). Both OpenSim models were converted using a converter to preserve accurate moment arm, muscletendon length and muscle force (Wang et al., 2022b). A more detailed description of this model can be found in (Caggiano et al., 2022a).

2.3. Tasks

Participants were asked to build controllers to solve two different tasks. The first task, a die reorientation, required a rotation of a die over the palm to match a desired goal orientation without dropping it (Fig. 1:A). The second task, inspired by the chinese practise of baoding ball for increasing finger flexibility and for hand rehabilitation, necessitated simultaneous rotation of two balls respect to each other over the palm without dropping them (Fig. 1:B).

Action and observation space. The action space to control the hand actuation was a 39-dimensional vector of continuous values between 0 and 1 (corresponding to minimum and maximum muscle contraction, respectively). The state space consisted of a vector containing MyoHand's joint angle position, velocity and muscle activations. In addition, positions and orientations of the die and position of the balls were also appended respectively for each task.

Task parameters and randomization Tasks parameters and the different randomized parameters between phases for each task can be found in Table 1.

2.4. Submissions and Evaluation

In order to succeed, participants needed to obtain the highest success (in terms of goal achievement) with the minimum effort (in terms of lowest overall muscle activation). The

Task - Phase	position [mm]	orientation [rad]	size [mm]	mass [g]	friction coefficient
Die reor. - 1	± 10	± 1.57	26	108	± (1.0, 0.005, 0.0001)
Die reor. - 2	± 20	± 3.14	± 7	50 - 250	± (0.2, 0.001, 0.00002)
	completion [s]	radius (x, y) [mm]	size [mm]	mass [g]	friction coefficient
Bao. ball - 1	5	(25, 28)	22	43	± (1.0, 0.005, 0.0001)
Bao. ball - 2	± 1	(20-30, 22-32)	± 3	30 - 300	± (0.2, 0.001, 0.00002)

Table 1: Summary of task variations

EvalAI platform (https://eval.ai) was used for hosting the challenge and to run the evaluation.

Evaluation Metrics. Die-Reorientation task used negative orientation error $M_{t=H} = -|R_t - R_{goal}|$ as a performance metric, where R_t is the orientation of the at the end of the task horizon and R_{goal} is the target orientation. Baoding balls task, on the other hand, used number of rotations $M_{t=H} = |R_t^z|$ as the performance metric, where R_t^z is the relative rotation of the sphere with respect to each other. For quantitative evaluations of the submissions, participants were asked to upload their behavior policies to our online platform which automatically evaluated the solution and updated results on a score-board. Final score were averaged over multiple seeds and task variations.

3. Solution strategies

In the following section the first two teams for each tasks describer their solution and the results obtained.

3.1. Die rotation tasks approach

3.1.1. Team PKU-MARL (First ex-aequo)

Team PKU-MARL used a traditional reinforcement learning framework with a Natural Policy Gradient (Kakade, 2001) algorithm. For training, Reward Shaping (Laud, 2004), Curriculum Learning (Bengio et al., 2009), and Multi-target Training (Chen et al., 2022a) were used to improve the performance of the policy (Fig. 2). The baseline reward function was extended by using three additional reward terms encouraging the agent to touch the die, rotate the die and move the die to the target position. The curriculum training affected both the magnitude of domain randomization and the range of possible goals. Multi-target training was another modification done to the original environment: the target was refreshed after success, so that the agent has to learn to follow the changing goal.

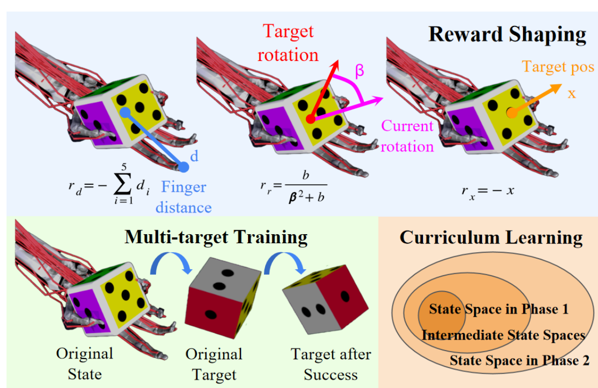


Figure 2: Training methods (PKU-MARL).

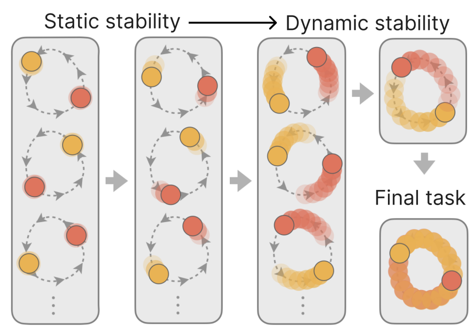
3.1.2. Team IARAI-JKU (First ex-aequo)

The evolution of the high jump technique (Dapena, 2002) shows how essential curriculum learning is for musculoskeletal skills (Skinner, 1958; Selfridge et al., 1985; Schmidhuber, 1991; Elman, 1993; Schmidhuber, 2002; Bengio et al., 2009; Schmidhuber, 2012; Portelas et al., 2020). It further suggests that exploration in such high-dimensional spaces is challenging. Hence, Team IARAI-JKU considered three ways to reduce task-irrelevant exploration (Siripurapu et al., 2022). First, a curriculum that automatically increases the task difficulty based on the agent's performance (Wang et al., 2019) by adapting the starting state distribution (Florensa et al., 2018). Second, potential-based shaping rewards (Ng et al., 1999; Arjona-Medina et al., 2019). And third, episodes are terminated early upon failure. The solution was built on top of the evotorch (Toklu et al., 2023) baseline using PGPE (Sehnke et al., 2010) and the ClipUp optimizer (Toklu et al., 2020).

3.2. Baoding ball tasks approach

3.2.1. Team stiff_fingers (first)

Combining ideas from stochastic optimal control Todorov and Jordan (2002) and reinforcement learning, Team stiff_fingers developed a training curriculum called Static to Dynamic Stabilization (SDS). The SDS curriculum first learns stable static solutions at several intermediate points along the desired object trajectory and gradually relaxes them to yield dynamically stable movement motifs. Akin to coaching techniques for skill-learning in humans, SDS allows the agent to experience intermediate configurations *before* learning a policy that reaches those configurations from the default initial state of the task at hand.



Ablation Study

Curriculum	Phase 1	Phase 2
None	41%	0%
Location only	42%	4%
Speed only	45%	0%
SDS *(ours)*	**100%**	**55%**

Figure 3: Training procedure and Ablation study (stiff_fingers).

In the first task, the balls are initialized at random phases along the desired rotation cycle, and the task of the agent is simply to hold them still at the initial position (see Fig. 3, first panel). In the following tasks, the balls are also initialized randomly, but now the task of the agent is to move them in the desired trajectory, gradually increasing the target speed. As the curriculum advances and the targets speed up, at one point it is not beneficial to use random initialization anymore, as the policy can benefit from exploiting the inertia of the balls. At this point, SDS initializes the balls at the original initial position of the task (see Fig. 3, second-to-last panel) and continues speeding up the targets until it reaches the final task. Stable-baselines3 (Raffin et al., 2019) was used to deploy SDS using a popular on-policy reinforcement learning algorithm PPO; (Schulman et al., 2017) with a recurrent architecture that has LSTM layers (Hochreiter and Schmidhuber, 1997) in both the actor and critic, which allows to deal with the partially observed environment. The agent receives positive binary rewards for each timestep when the balls overlap with the targets and negative scalar rewards proportional to the distance between each ball and its target. Our observation and action space were not modified from the ones given by the challenge designers.

3.2.2. Team AL4Muscles (second, honorable-mention)

Team AL4Muscles solved the first phase with a combination of MPO (Abdolmaleki et al., 2018), implemented in TonicRL (Pardo, 2020), and DEP-RL (Schumacher et al., 2023). The original dense reward signal was maintained, and only added a negative cost for dropping the balls, states and action spaces were kept at default values. While pure MPO would often get stuck in the first half of the rotation, DEP-RL was able to more reliably explore around this bottleneck.

In the second phase, the training regime was changed, as the policies did not cope well with the multi-task aspect. Instead of the original task, the policy was trained with randomized terminations and static goals. The goals were randomly initialized along the rotation ellipse and the episode terminated with a success if the center of the goal was achieved. Crucially, the success only counts after a randomly sampled time interval, which is not part of the state. The policy now needs to stabilize the goal, but is sometimes rewarded for just shortly achieving it. A binary reward that incentives staying closer to the center of the goal was also used. In addition, the balls were initialized in a position recorded from a previous episode at random times. With this procedure, the policy has to reach any goal position from any prior position and stabilize it, which generalizes well to the original task with moving goals. The joint angle velocity was not part of the original state and was approximated by the difference of successive time steps.

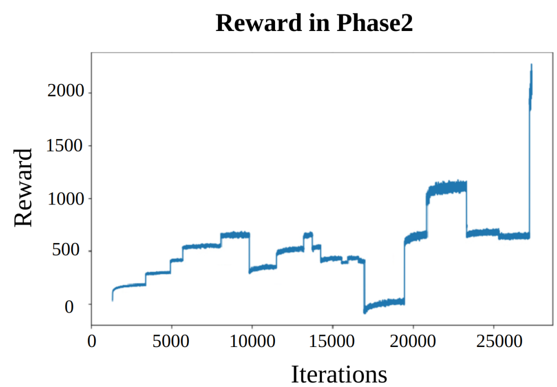


Figure 4: Training curve (PKU-MARL).

4. Results

4.1. Winner die rotation tasks results

4.1.1. Team PKU-MARL (First ex-aequo)

The best solution of the team PKU-MARL used all the three methods mentioned in 3.1.1. Specifically, the curriculum learning method along with the reward shaping technique improved the success rate in phase 1 to 71%. In phase 2, the agent did not perform as expected during the multi-target training, so the focus shifted on reaching a high success rate for rotations within 90 degrees and giving up those large rotations, resulting in an 11% success rate. Fig. 4 shows the reward curve during training. The large leaps in the curve is due to reward tuning during the training. The code describing this solution is available at [2].

2. https://github.com/PKU-MARL/MyoChallenge

4.1.2. Team IARAI-JKU (First ex-aequo)

Team IARAI-JKU's solution achieved the highest success rate of 13% with the lowest effort of 0.03. A 3 layer RNN with leakyReLU activation (22k parameters) was used. A 256 cpu ray cluster, with a population of 256 environments, with each environment updating its difficulty based on the past performance of the population of 20000 agents was used. Training lasted for a total of 4 days which corresponded to 4000 parameter updates or 12 billion samples and submitted our best performing model. Fig. 5 shows the performance curves with respect to updates (3 million samples per update). Note that however, due to the use of PGPE, this solution solution consumes more samples, but due to the lack of backpropagation and lower variance gradient estimates, required less wall time. Code is available in [3].

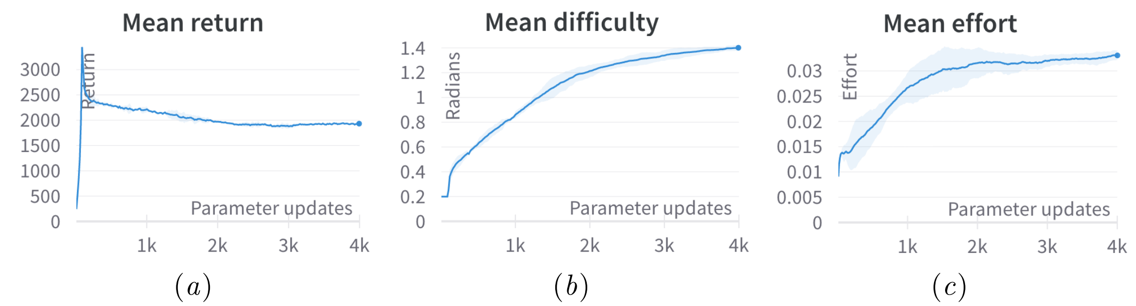


Figure 5: Average over two runs. (IARAI-JKU) (a) depicts the mean return, which starts decreasing slowly as env difficulty increases in (b). (c) shows the mean effort

4.2. Winner baoding ball tasks results

4.2.1. Team stiff_fingers (first)

SDS (section 3.2.1) achieved perfect performance in Phase 1 (100%) and 55% in Phase 2. Note that it is impossible to reach 100% performance in the full task since the targets and the balls are initiated independently. By ablating SDS from the training, we show how the pre-training phase in which the model learns how to achieve static stability at multiple intermediate positions is fundamental for the effective learning of the dynamic motor control task. A comparison included three other training procedures using the same architecture. Training directly on the final task (Fig. 3: *None*) fails in the Full Task and quickly falls into a local minimum in the simpler Fixed task. Furthermore, the same curriculum used to obtain the winning policy for the challenge fails when starting all the episodes with the balls at the same initial position (Fig. 3: *Speed only*), or when starting at multiple initial positions but with a rapidly moving target (Fig. 3: *Location only*). Code is available at [4].

4.2.2. Team AL4Muscles (second)

Team AL4Muscles adopted DEP-RL that, with modified task conditions, achieved 98% in phase 1 and 41% in phase 2. While the additional DEP (Der and Martius, 2015; Schumacher

3. https://github.com/iarai/MyoChallgenge-IARAI-JKU
4. https://github.com/amathislab/myochallenge

et al., 2023) exploration was able to escape certain bottleneck states in phase 1, this effect was smaller in phase 2 due to the task variability. A modified surrogate task allowed to achieve good performance in phase 2, despite randomized physics and task variations. Fig. 6 shows the score in the challenge task over training for 10 random seeds. Only 10 rollouts were evaluated at each point, which is why the shown scores can be larger than the final task score, evaluated online. Considering that in (Caggiano et al., 2022b) the baoding ball task was strongly seed dependent, our approach achieves low variance across runs. Code is available in [5].

5. Discussions

5.1. Impact

5.2. Participation

MyoChallenge had a consistent participation of 40 teams coming from more than 10 countries. Across the 2 phases, we had more than 340 submissions. MyoSuite (Caggiano et al., 2022a) powered the competition and was downloaded more than 4000 times, with a clear peak before the end of the first phase/start of phase 2. This made the *MyoChallenge* one of the most successful challenge of NeurIPS'22. We also got the opportunity to organize a workshop[6] as part of the NeurIPS 22 conference. In this workshop, we were able to bring together experts from the neuroscience, biomechanics, and machine learning fields, which created an unique opportunity to start a discussion on the crossroads of these disciplines. After the end of the challenge, the Myosuite framework was downloaded more than 2000 times. This indicates that the *MyoChallenge* further catalyzed the community around solving those problems.

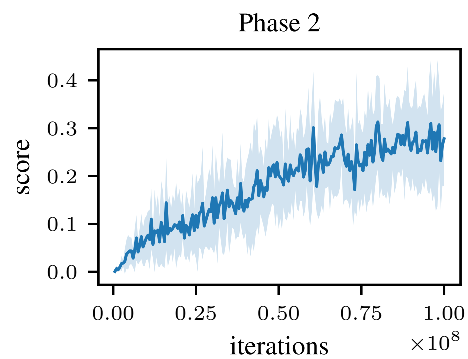


Figure 6: Training curve (AL4Muscles).

5.3. Lesson learned

Most winning solutions used curriculum learning. Some solutions used gradual domain randomization or an ensemble of networks for task specialization.

Team PKU-MARL achieved good results using a simple model, demonstrating the feasibility of reinforcement learning in complex muscle control tasks. However, at the same time, this model is still not comparable to human or animal control systems. Here are some realistic ideas that may be worth looking into: first, constructing a mechanism that combines multiple policy networks, while the networks can be separately trained in different states, will result in diverse behaviors. Second, by increasing the parallelism to provide more samples for training. The team believes that reinforcement learning will become

5. https://github.com/martius-lab/AL4MyoChallenge
6. https://sites.google.com/view/myochallenge#h.t3275626vjox

increasingly versatile and applicable in operational skill training due to advancements in training techniques (Kuba et al., 2021) and the development of more realistic environments. These improvements will enable the acquisition of skills with higher flexibility (Chen et al., 2022b) and generalization capabilities (Li et al., 2022; Geng et al., 2023).

Team IARAI-JKU learned that reducing task-irrelevant exploration (Siripurapu et al., 2022) is key to learning in such high-dimensional environments. Second, it was found that although evolutionary methods are worse than standard RL methods in sample efficiency (Majid et al., 2021) (despite the lower variance gradient estimates in our case), since they avoid backpropagation through the recurrent policy, they take advantage of the high throughput of the MyoSuite environments. Although the use of distributed evolutionary method impacts reproducibility, the relatively few and easily explainable hyperparameters ensure the variance in the results are low (Salimans et al., 2017). This shows that it is general and can be applied to other such tasks. Third, the use of a recurrent policy. Nevertheless, recent methods have shown that transformers may be better (Li et al., 2023) as their attention mechanism bypasses the sequential flow of information and allows for better credit assignment and recollection of past actions.

Team stiff_fingers found that the task posed a very complex exploration problem due to the instability of the balls, which can easily fall from the hand, the high dimensional observation and action spaces, and the long timescale at which muscular actions produce their effect in the environment. Their result highlights the usefulness of the SDS curriculum to develop realistic, high-performance sensorimotor models. The team believes that developing such training procedures for realistic musculoskeletal models will bring new advances in studying motor control by allowing exploration of the core principles of biological skill learning and help in reverse-engineering sensorimotor circuits (Hausmann et al., 2021). They also note that their model was trained for 6 weeks. Given that MyoSuite is about 1000 times faster than OpenSim, this also illustrates the new possibilities.

Team AL4Muscles found that reward design, state initialization and termination conditions had the strongest effect on performance. In their case, the agent had to stabilize the objects from many different starting conditions while the reward specification makes the goal configuration discoverable, without prescribing a trajectory. As for unsuccessful approaches, the team experimented with state space design, hindsight experience replay, curriculum learning for physics and task variations, and different action cost formulations. A bigger state space often renders learning unfeasible, while the original state space induces partial observability. It was found that MPO performed well with default parameters, which could not significantly be improved upon with tuning. The reusable action representations (Leo et al., 2016) for muscle-control is believed to be a promising direction, while current RL methods also lack the capability to produce truly energy efficient solutions.

5.4. Limitations

All proposed solutions were based on reinforcement learning, while powerful they have shown limitations in this challenge. Solutions inspired from other field that machine learning could also help in solving musculoskeletal control tasks for example in lower-limb control, reflex (Song and Geyer, 2015), sensorimotor connectivity priors Chiappa et al. (2022) or central pattern generator (Ijspeert et al., 2007) are simple yet extremely powerful solution (simple rule or neurons) and they have been shown to be able to create stable locomotion. In future challenges, the creation of cross-disciplinary teams (biomechanics, neuroscience and machine learning) could facilitate the development of hybrid solutions.

Large file to upload solutions. A big blocker experienced by some teams was the need of submitting solutions via heavy i.e. > 1Gb, docker files, which are sometimes tricky to set-up and compile. The large size of those files might require stable and high-speed connection which might have limited participation. We offered helper scripts and documentation which reduced the docker set-up and compilation pain-point. In a questionnaire we run after the challenge, we found that the winners were not blocked by the Dockers submission but they would have welcomed simpler alternatives, for example, colab or Jupyter notebook, to submit their solutions. This potential participation selection needs to be more carefully investigated for future editions of this challenge.

Documentation. The post challenge survey answers also showed that participants requested more and clearer documentation and additional examples.

Underrepresented participation. A final limitations was the small participation of underrepresented population. For example, no participating came from south America or Africa. The large docker files required for the solution might have prevented participation. Also, winning teams did not contain any woman.

5.5. Future challenges

5.5.1. Organization

Organizing such a large scale event come with numerous challenges requiring both handling technical e.g. setting website, helper code, infrastructure set-up and management, and logistical e.g. advertising and finding sponsors. Future challenges will promote participation of students to help with different aspects of the technical and logistical planning and execution.

As previously discussed, it will be very important to promote more participation of underrepresented population and from under developed country.

5.6. Tasks

Future editions of the *MyoChallenge* will explore more complex manipulations for example by implementing manipulations with bimanual musculoskeletal arms. This would create the possibility to investigate daily living activity manipulation such as tightening the lid onto the jar by twisting it. This would allow further exploration of the task generalization. In addition, we would also like to explore locomotion in complex and rough terrain with lower-limbs musculoskeletal models.

Acknowledgments

We would like to acknowledge support for this competition from the university of Twente Techmed and DSI, Google cloud computing and EU ERC StG interact. Dhruv Batra, Ram Ramrakhya, Deshraj Yadav, and Rishabh Jain for help with the EvalAI platform. A.C. and A.M.: Swiss SNF grant (310030_212516)

References

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=S1ANxQW0b.

Jose A. Arjona-Medina, Michael Gillhofer, Michael Widrich, Thomas Unterthiner, Johannes Brandstetter, and Sepp Hochreiter. Rudder: Return decomposition for delayed rewards. *arXiv:1806.07857 [cs, math, stat]*, September 2019.

Vittorio La Barbera, Fabio Pardo, Yuval Tassa, Monica Daley, Christopher Richards, Petar Kormushev, and John Hutchinson. Ostrichrl: A musculoskeletal ostrich simulation to study bio-mechanical locomotion. *CoRR*, abs/2112.06061, 2021. URL https://arxiv.org/abs/2112.06061.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

Vittorio Caggiano, Huawei Wang, Guillaume Durandau, Kumar Sartori, Massimo, and Vikash. Myosuite: A contact-rich simulation suite for musculoskeletal motor control. In *4th Annual Conference on Learning for Dynamics and Control (L4DC)*. Proceedings of Machine Learning Research, 2022a.

Vittorio Caggiano, Huawei Wang, Guillaume Durandau, Massimo Sartori, and Vikash Kumar. Myosuite – a contact-rich simulation suite for musculoskeletal motor control, 2022b. URL https://arxiv.org/abs/2205.13600.

Tao Chen, Jie Xu, and Pulkit Agrawal. A system for general in-hand object re-orientation. In *Conference on Robot Learning*, pages 297–307. PMLR, 2022a.

Yuanpei Chen, Yaodong Yang, Tianhao Wu, Shengjie Wang, Xidong Feng, Jiechuang Jiang, Stephen Marcus McAleer, Hao Dong, Zongqing Lu, and Song-Chun Zhu. Towards human-level bimanual dexterous manipulation with reinforcement learning. *arXiv preprint arXiv:2206.08686*, 2022b.

Alberto Silvio Chiappa, Alessandro Marin Vargas, and Alexander Mathis. Dmap: a distributed morphological attention policy for learning to locomote with a changing body. *Advances in Neural Information Processing Systems*, 35:37214–37227, 2022.

J. Dapena. The evolution of high jumping technique: Biomechanical analysis. In *ISBS-Conference Proceedings Archive*, 2002.

Ralf Der and Georg Martius. Novel plasticity rule can explain the development of sensorimotor intelligence. *Proceedings of the National Academy of Sciences*, 112(45):E6224–E6232, 2015. doi: 10.1073/pnas.1508400112. URL http://www.pnas.org/content/112/45/E6224.abstract.

Jeffrey L. Elman. *Learning and Development in Neural Networks: The Importance of Starting Small*. 1993.

Tom Erez, Yuval Tassa, and Emanuel Todorov. Simulation tools for model-based robotics: Comparison of bullet, havok, mujoco, ODE and physx. In *IEEE International Conference on Robotics and Automation, ICRA 2015*, pages 4397–4404. IEEE, 2015. doi: 10.1109/ICRA.2015.7139807. URL https://doi.org/10.1109/ICRA.2015.7139807.

Florian Fischer, Miroslav Bachinski, Markus Klar, Arthur Fleig, and Jörg Müller. Reinforcement learning control of a biomechanical model of the upper extremity. *Scientific Reports*, 11(1), jul 2021. doi: 10.1038/s41598-021-93760-1. URL https://doi.org/10.1038%2Fs41598-021-93760-1.

Carlos Florensa, David Held, Markus Wulfmeier, Michael Zhang, and Pieter Abbeel. Reverse curriculum generation for reinforcement learning. *arXiv:1707.05300 [cs]*, July 2018.

Yiran Geng, Boshi An, Haoran Geng, Yuanpei Chen, Yaodong Yang, and Hao Dong. End-to-end affordance learning for robotic manipulation. *International Conference on Robotics and Automation (ICRA)*, 2023.

Nikolaus Hansen, Sibylle D. Müller, and Petros Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evol. Comput.*, 11(1):1–18, mar 2003. ISSN 1063-6560. doi: 10.1162/106365603321828970. URL https://doi.org/10.1162/106365603321828970.

Sébastien B Hausmann, Alessandro Marin Vargas, Alexander Mathis, and Mackenzie W Mathis. Measuring and modeling the motor system with machine learning. *Current opinion in neurobiology*, 70:11–23, 2021.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 11 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

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

Aleksi Ikkala and Perttu Hämäläinen. Converting biomechanical models from opensim to mujoco. In *Converging Clinical and Engineering Research on Neurorehabilitation IV: Proceedings of the 5th International Conference on Neurorehabilitation (ICNR2020), October 13–16, 2020*, pages 277–281. Springer, 2022.

Emanuel Joos, Fabien Péan, and Orcun Goksel. Reinforcement learning of musculoskeletal control from functional simulations, 2020. URL https://arxiv.org/abs/2007.06669.

Sham M Kakade. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

Fotios Alexandros Karakostis, Daniel Haeufle, Ioanna Anastopoulou, Konstantinos Moraitis, Gerhard Hotz, Vangelis Tourloukis, and Katerina Harvati. Biomechanics of the human thumb and the evolution of dexterity. *Current Biology*, 31(6):1317–1325.e8, 2021. ISSN 0960-9822. doi: https://doi.org/10.1016/j.cub.2020.12.041. URL https://www.sciencedirect.com/science/article/pii/S0960982220318935.

Łukasz Kidziński, Sharada P Mohanty, Carmichael F Ong, Jennifer L Hicks, Sean F Carroll, Sergey Levine, Marcel Salathé, and Scott L Delp. Learning to run challenge: Synthesizing physiologically accurate motion using deep reinforcement learning. In *The NIPS'17 Competition: Building Intelligent Systems*, pages 101–120. Springer, 2018.

Łukasz Kidziński, Carmichael Ong, Sharada Prasanna Mohanty, Jennifer Hicks, Sean Carroll, Bo Zhou, Hongsheng Zeng, Fan Wang, Rongzhong Lian, Hao Tian, et al. Artificial intelligence for prosthetics: Challenge solutions. In *The NeurIPS'18 Competition: From Machine Learning to Intelligent Conversations*, pages 69–128. Springer, 2020.

Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. Trust region policy optimisation in multi-agent reinforcement learning. *arXiv preprint arXiv:2109.11251*, 2021.

Adam Daniel Laud. *Theory and application of reward shaping in reinforcement learning.* University of Illinois at Urbana-Champaign, 2004.

Jong Hwa Lee, Deanna S. Asakawa, Jack T. Dennerlein, and Devin L. Jindrich. Finger muscle attachments for an opensim upper-extremity model. *PLOS ONE*, 10(4):1–28, 04 2015. doi: 10.1371/journal.pone.0121712. URL https://doi.org/10.1371/journal.pone.0121712.

Seunghwan Lee, Moonseok Park, Kyoungmin Lee, and Jehee Lee. Scalable muscle-actuated human simulation and control. *ACM Trans. Graph.*, 38(4), jul 2019. ISSN 0730-0301. doi: 10.1145/3306346.3322972. URL https://doi.org/10.1145/3306346.3322972.

Andrea Leo, Giacomo Handjaras, Matteo Bianchi, Hamal Marino, Marco Gabiccini, Andrea Guidi, Enzo Pasquale Scilingo, Pietro Pietrini, Antonio Bicchi, Marco Santello, and Emiliano Ricciardi. A synergy-based hand control is encoded in human motor cortical areas. *eLife*, 5:e13420, feb 2016. ISSN 2050-084X. doi: 10.7554/eLife.13420. URL https://doi.org/10.7554/eLife.13420.

Puhao Li, Tengyu Liu, Yuyang Li, Yiran Geng, Yixin Zhu, Yaodong Yang, and Siyuan Huang. Gendexgrasp: Generalizable dexterous grasping. *arXiv preprint arXiv:2210.00722*, 2022.

Wenzhe Li, Hao Luo, Zichuan Lin, Chongjie Zhang, Zongqing Lu, and Deheng Ye. A Survey on Transformers in Reinforcement Learning, January 2023.

Amjad Yousef Majid, Serge Saaybi, Tomas van Rietbergen, Vincent Francois-Lavet, R. Venkatesha Prasad, and Chris Verhoeven. Deep Reinforcement Learning Versus Evolution Strategies: A Comparative Survey, September 2021.

Daniel C. McFarland, Emily M. McCain, Michael N. Poppo, and Katherine R. Saul. Spatial dependency of glenohumeral joint stability during dynamic unimanual and bimanual pushing and pulling. *Journal of Biomechanical Engineering-Tansation of the ASME*, 141 (5), 05 2019. doi: 10.1115/1.4043035.

Daniel C McFarland, Benjamin I Binder-Markey, Jennifer A Nichols, Sarah J Wohlman, Marije de Bruin, and Wendy M Murray. A musculoskeletal model of the hand and wrist capable of simulating functional tasks. *IEEE Transactions on Biomedical Engineering*, 2022.

Andrew Y. Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *In Proceedings of the Sixteenth International Conference on Machine Learning*, pages 278–287. Morgan Kaufmann, 1999.

Fabio Pardo. Tonic: A deep reinforcement learning library for fast prototyping and benchmarking. *arXiv preprint arXiv:2011.07537*, 2020.

Jungnam Park, Sehee Min, Phil Sik Chang, Jaedong Lee, Moon Seok Park, and Jehee Lee. Generative gaitnet. *CoRR*, abs/2201.12044, 2022. URL https://arxiv.org/abs/2201.12044.

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

Rémy Portelas, Cédric Colas, Lilian Weng, Katja Hofmann, and Pierre-Yves Oudeyer. Automatic curriculum learning for deep rl: A short survey. *arXiv:2003.04664 [cs, stat]*, May 2020.

Antonin Raffin, Ashley Hill, Maximilian Ernestus, Adam Gleave, Anssi Kanervisto, and Noah Dormann. Stable baselines3. https://github.com/DLR-RM/stable-baselines3, 2019.

Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning, 2017.

Katherine R. Saul, Xiao Hu, Craig M. Goehler, Meghan E. Vidt, Melissa Daly, Anca Velisar, and Wendy M. Murray. Benchmarking of dynamic simulation predictions in two software platforms using an upper limb musculoskeletal model. *Computer Methods in Biomechanics and Biomedical Engineering*, 18(13):1445–1458, 2015. doi: 10.1080/10255842.2014.916698. URL https://doi.org/10.1080/10255842.2014.916698. PMID: 24995410.

J. Schmidhuber. Curious model-building control systems. In *[Proceedings] 1991 IEEE International Joint Conference on Neural Networks*, pages 1458–1463 vol.2, November 1991. doi: 10.1109/IJCNN.1991.170605.

Juergen Schmidhuber. Optimal Ordered Problem Solver. *arXiv:cs/0207097*, December 2002.

Jürgen Schmidhuber. POWERPLAY: Training an Increasingly General Problem Solver by Continually Searching for the Simplest Still Unsolvable Problem. *arXiv:1112.5309 [cs]*, November 2012.

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

Pierre Schumacher, Daniel F.B. Haeufle, Dieter Büchler, Syn Schmitt, and Georg Martius. Dep-rl: Embodied exploration for reinforcement learning in overactuated and musculoskeletal systems. In *Proceedings of the Eleventh International Conference on Learning Representations (ICLR)*, May 2023. URL https://openreview.net/forum?id=C-xa_D3oTj6.

Frank Sehnke, Christian Osendorfer, Thomas Ruckstieß, Alex Graves, Jan Peters, and Jurgen Schmidhuber. Policy Gradients with Parameter-Based Exploration for Control. page 10, 2010.

Oliver G Selfridge, Richard S Sutton, and Andrew G Barto. Training and tracking in robotics. pages 670–672, 1985.

Rahul Siripurapu, Vihang Prakash Patil, Kajetan Schweighofer, Marius-Constantin Dinu, Thomas Schmied, Luis Eduardo Ferro Diez, Markus Holzleitner, Hamid Eghbal-zadeh, Michael K. Kopp, and Sepp Hochreiter. InfODist: Online distillation with Informative rewards improves generalization in Curriculum Learning. In *Deep Reinforcement Learning Workshop NeurIPS 2022*, December 2022.

B. F. Skinner. Reinforcement today. *American Psychologist*, 13:94–99, 1958. ISSN 1935-990X. doi: 10.1037/h0049039.

Seungmoon Song and Hartmut Geyer. Generalization of a muscle-reflex control model to 3d walking. In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 7463–7466, 2013. doi: 10.1109/EMBC.2013.6611284.

Seungmoon Song and Hartmut Geyer. A neural circuitry that emphasizes spinal feedback generates diverse behaviours of human locomotion. *The Journal of physiology*, 593(16): 3493–3511, 2015.

Seungmoon Song, Łukasz Kidziński, Xue Bin Peng, Carmichael Ong, Jennifer Hicks, Sergey Levine, Christopher G Atkeson, and Scott L Delp. Deep reinforcement learning for modeling human locomotion control in neuromechanical simulation. *Journal of neuroengineering and rehabilitation*, 18(1):1–17, 2021.

Emanuel Todorov and Michael I Jordan. Optimal feedback control as a theory of motor coordination. *Nature neuroscience*, 5(11):1226–1235, 2002.

Nihat Engin Toklu, Paweł Liskowski, and Rupesh Kumar Srivastava. ClipUp: A Simple and Powerful Optimizer for Distribution-based Policy Evolution, December 2020.

Nihat Engin Toklu, Timothy Atkinson, Vojtěch Micka, Paweł Liskowski, and Rupesh Kumar Srivastava. EvoTorch: Scalable evolutionary computation in Python. *arXiv preprint*, 2023. https://arxiv.org/abs/2302.12600.

Francisco J Valero-Cuevas. An integrative approach to the biomechanical function and neuromuscular control of the fingers. *Journal of biomechanics*, 38(4):673–684, 2005.

Huawei Wang, Vittorio Caggiano, Guillaume Durandau, Kumar Sartori, Massimo, and Vikash. Myosim: Fast and physiologically realistic mujoco models for musculoskeletal and exoskeletal studies. In *2022 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2022a.

Huawei Wang, Vittorio Caggiano, Guillaume Durandau, Massimo Sartori, and Vikash Kumar. Myosim: Fast and physiologically realistic mujoco models for musculoskeletal and exoskeletal studies. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 8104–8111. IEEE, 2022b.

Rui Wang, Joel Lehman, Jeff Clune, and Kenneth O. Stanley. Paired Open-Ended Trailblazer (POET): Endlessly Generating Increasingly Complex and Diverse Learning Environments and Their Solutions. *arXiv:1901.01753 [cs]*, February 2019.

Isabell Wochner, Pierre Schumacher, Georg Martius, Dieter Büchler, Syn Schmitt, and Daniel F. B. Haeufle. Learning with muscles: Benefits for data-efficiency and robustness in anthropomorphic tasks, 2022. URL https://arxiv.org/abs/2207.03952.

Appendix A.

Label	Muscle groups name	Number of muscles
ECRL	Extensor Carpis Radialis Longus	1x
ECRB	Extensor Carpis Radialis Brevis	1x
ECU	Extensor Carpi Ulnaris	1x
FCR	Flexor Carpi Radialis	1x
FCU	Flexor Carpi Ulnaris	1x
PL	Palmaris longus	1x
PT	Pronator teres	1x
PQ	Pronator	1x
EIP	Extensor Indicis Proprius	1x
EPL	Extensor Pollicis Longus	1x
EPB	Extensor Pollicis Brevis	1x
FPL	Flexor Pollicis Longus	1x
APL	Abductor Pollicis Longus	1x
OP	Opponens Pollicis	1x
FDS	Flexor Digitorum Superficialis	4x
FDP	Flexor Digitorum Profundus	4x
EDC	Extensor Digitorum Communis	4x
RI	Radial Interosseous	4x
LU-RB	Lumbrical	4x
UI-UB	Palmar or Ulnar Interosseous	4x