2022 IEEE International Conference on Robotics and Automation (ICRA)
May 23-27, 2022. Philadelphia, PA, USA

MyoSim: Fast and physiologically realistic MuJoCo models for musculoskeletal and exoskeletal studies

Huawei Wang*[1], Vittorio Caggiano*[2], Guillaume Durandau[1], Massimo Sartori[1], Vikash Kumar[2]

***Abstract*— Owing to the restrictions of live experimentation, musculoskeletal simulation models play a key role in biological motor control studies and investigations. Successful results of which are then tried on live subjects to develop treatments as well as robot aided rehabilitation procedures for addressing neuromusculoskeletal anomalies ranging from limb loss, to tendinitis, from sarcopenia to brain and spinal injuries. Despite its significance, current musculoskeletal models are computationally expensive, and provide limited support for contact-rich interactions which are essential for studying motor behaviors in activities of daily living, during rehabilitation treatments, or in assistive robotic devices. To bridge this gap, this work proposes an automatic pipeline to generate physiologically accurate musculoskeletal, as well as hybrid musculoskeletal-exoskeletal models. Leveraging this pipeline we present *MyoSim* – a set of computationally efficient (over 2 orders of magnitude faster than state of the art) musculoskeletal models that support fully interactive contact rich simulation. We further extend MyoSim to support additional features that help simulate various real-life changes/diseases, such as muscle fatigue, and sarcopenia. To demonstrate the potential applications, several use cases, including interactive rehabilitation movements, tendon-reaffirmation, and the co-simulation with an exoskeleton, were developed and investigated for physiological correctness. Webpage: https://sites.google.com/view/myosuite**

I. Introduction

The agility and adaptability of the human musculoskeletal system and its underlying neural control have long inspired research in understanding biological motor control [1]. This research is important to identify both the neural mechanism implemented by the central nervous system and how the actuation of movements is implemented in the musculoskeletal system. Indeed, injuries, diseases and aging can cause changes both in the central control and in the peripheral musculoskeletal, which affect the capacity to support body weight, maintain posture, move, and manipulate objects [2]. All of those have far reaching consequences in maintaining human health.

The study of the musculoskeletal actuation is also critical to develop robotic rehabilitation strategies via wearable assistive devices [3] as well to enable equivalent capabilities in robotics system such as cobot [4] or bioinspired controllers for legged robots [5]. Progress in these areas is often constrained by the scarcity of detailed human-robot experiments which

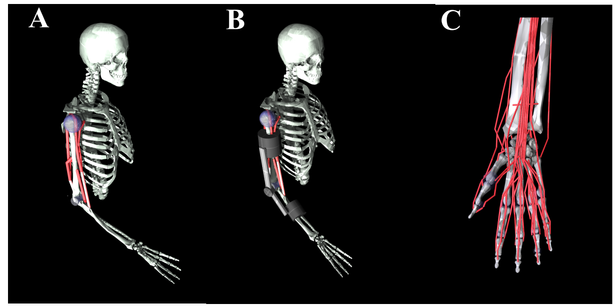

Fig. 1: Neuromusculoskeletal model generated using the conversion tool presented in Section II-A. A) a two degrees of freedom (DOF) elbow model with 6 muscle-tendon units, B) previous elbow model with an one DOF exoskeleton, and C) Hand model with 23 Dofs and 38 muscle-tendon units..

limits our understanding of how different robotic design and control strategies lead to a diverse repertoire of human-robot motor skills. Accurate and computationally efficient computer simulations of human-robot interaction can enable devising optimal robotic technologies completely *in silico*, thereby speeding up the process of translating already-tested technologies to *in vivo* real-world situations. This can be especially impactful in the neuromuscular conditions (e.g. fatigue, sacopenia, spinal cord injury, etc), where human-robot interaction studies are quite limited.

Human musculoskeletal system investigations have been limited by the invasiveness of the available techniques and by the restricted manipulations possible [6]. Computational methods have been used to fill the gap to simulate how muscle, bone, and joints interact for the generation of movements. This has allowed a more detailed understanding of the neural mechanisms to generate movements commands [7], [8] and how disease [9], traumas [10] and perturbations affect movement control [11]. Physics-based musculoskeletal simulation engines such as OpenSim [12], AnyBody [13] and SIMM [14] have focused on making detailed and physiologically accurate models of the musculoskeletal system. They are widely used in human neural-mechanical control, human robot interaction, and rehabilitation, among others fields. However such engines are computationally expensive and provide limited support for contact rich interactions. Alternatively, physics engines such as PyBullet [15], MuJoCo [16] and Dart [17] are relatively more efficient and support contact interactions but lack adequate support for muscle modeling (PyBullet and Dart) or lack functionally validated musculoskeletal models. Even though attempts have been made in adding physiological

*These authors contributed equally to this manuscript.
[1]Huawei Wang, Guillaume Durandau and Massimo Sartori are with Faculty of Biomechancial Engineering, Engineering Technology, University of Twente, 7500 AE Enschede, The Netherlands h.wang-2@utwente.nl; g.durandau@utwente.nl; m.sartori@utwente.nl
[2] Vittorio Caggiano and Vikash Kumar are with Facebook AI Research, USA caggiano@gmail.com; vikashplus@gmail.com

models in MuJoCo [18]–[20], they were limited in their validation.

Numerical simulations can be leveraged to investigate, understand, and modulate physical interactions between the neuromusculoskeletal system and wearable robotic devices (e.g. exoskeletons): they enable predictions on how the human motor system reacts to robot assistance. This step is crucial for the development of technologies that could effectively enhance or restore movement capacity [21]. In contrast, current robotic exoskeletons are currently built and tested with limited knowledge on how the human body would respond and adapt over time during motor interaction tasks. This is one of the major reasons why despite progress in mechatronics and materials, current robotic solutions have shown only modest results and have limited clinical impact [22], [23].

Most exoskeleton/human controllers are based on biomechanical principles [24] or are optimized using human in the optimization loop [25], which require first the creation of the exoskeleton followed by user studies, leading to long iteration times. Additionally we have a minimal understating of the composite human-exoskeleton system properties because only limited experimental data are available [26]. Alternatively, simulation of neuromusculoskeletal and robotic systems can be combined with behavior synthesis frameworks [27]–[30] to drastically reduce the design iteration cycles. However, these frameworks need realistic and computationally efficient musculoskeletal models capable of simulating detailed contact rich musculoskeletal-exoskeletal-environment interactions to learn effective and physiologically accurate behaviors.

Here, we propose physiologically accurate (with respect to current state of the art, e.g. OpenSim [12]) neuromusculoskeletal models in a novel framework - *MyoSim* (A suite for musculoskeletal motor control simulation) - based on an efficient physic engine (MuJoCo [16]) suitable to exploit data driven Reinforcement Learning (RL) based neural command computation to study human machine interactions. **Specifically, our contributions are:**

- A fully automated model-agnostic pipeline which reduce the tedious manual effort often required in creating new models by optimizing existing and validated (OpenSim) musculoskeletal models. The resulting models are numerically equivalent to the original and can be simulated two orders of magnitude faster than the original models.
- A set of computationally efficient and physiologically accurate musculoskeletal *MyoSim* models generated (Figure 1A) and rigorously validated against the state of the art, e.g., OpenSim models.
- Embedding the *MyoSim* models in a framework that supports full rigid body physics and enables contact rich interaction with passive objects (such as household objects, tools, etc) as well as actuated devices (such as exoskeletons, automated doors, etc).
- Human-Robot simulations were generated as a use-case to study musculoskeletal responses to exoskeletal assistance. Support for biological phenomenons such as muscle-fatigue, sarcopenia, etc were also developed to study their effects on the overall musculoskeletal-exoskeletal system. The resulting system is demonstrated to show physiological correctness.

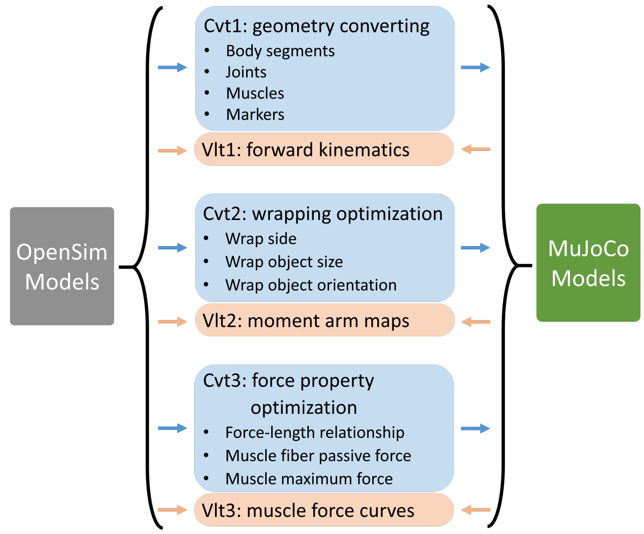

Fig. 2: Three-steps conversion pipeline

II. METHODS

In this section, the musculoskeletal model conversion tool is explained in detail first. Then the converted *MyoSim* models are outlined. Lastly, additional biological features supported in *MyoSim* , such as muscle fatigue and sarcopenia, are presented.

A. Musculoskeletal model conversion tool

We used physiologically validated and widely adopted OpenSim musculoskeletal models as reference. Then, a layer-by-layer automatic conversion pipeline was developed to convert those OpenSim models into MuJoCo. The pipeline consisted of three major conversion (Cvt) steps, as shown in Figure 2. A validation (Vlt) module followed each conversion step to match the converted MuJoCo model and the reference OpenSim model. A short description of them is listed below, followed by the detail explanations in the subsections.

1) **Cvt1: geometry conversion.** Parse kinematics details (body segment lengths, joints, muscles-tendon pathways, etc)[19] and reflective markers from OpenSim model into MuJoCo model.
   Vlt1: forward kinematics. Check that the end point positions along the kinematic chain are matched between the two models at different joint configurations.
2) **Cvt2: wrapping optimization.** Optimize wrapping constraints to guarantee that the muscle pathways are physiologically reasonable.
   Vlt2: moment arm maps. Check that moment arms of each muscle at each joint angle are matched between models.
3) **Cvt3: force property optimization.** Optimize muscle force parameters in MuJoCo to approximate OpenSim model's force curves.

Vlt3: muscle force curves. Check force-length-activation differences between the two models.

Geometry Conversion:
OpenSim and MuJoCo geometry structures are simply defined based on their relative position in a similar hierarchical (XML) format. An open-access conversion code was created by Ikkala et al [19] that parsed the bodies, joints, and muscles into the MuJoCo modelling format. We extended their code by adding in the conversion 1) markers and 2) wrapping surfaces for muscles to avoid collisions with bones during movements. The forward kinematics module (*Vlt1*) checks the differences of the selected marker locations at different joint configurations (see the Results section III-A).

Wrapping Optimization:
OpenSim and MuJoCo have different ways of defining how muscles wrap over wrapping surfaces. In OpenSim, three wrapping methods were defined: MidPoint, Axial, Hybrid [31], [32]. Nevertheless, in MuJoCo, muscles can wrap over or can be forced to pass through a surface. In order to guarantee similar forces produced by a muscle at each joint, an optimization was developed. We adapted both the location where a muscle wraps on the surface in MuJoCo i.e. *'side site'* and the dimensions of the wrapping objects (surfaces such as ellipsoids and torus, which are missing in MuJoCo, were approximated with other surfaces).

The following optimization was developed:

Find: wrapping sides & wrapping object sizes:
$X = \{[L_1, ..., L_n], [S_1, ..., S_p]\}$

To minimize: objective function:

$$F = \sum_{m=1,j=1}^{M,J} (d_{m,j}^{osim}(Q) - d_{m,j}^{mjc}(Q, X))^2 \tag{1}$$

When: iterate all joint angle meshes

Subject to: model kinematics; parameter bounds

where, $L = (x, y, z)$ represents the location of side site; $S = (r, l)$ represents the size of wrapping e.g. cylinder dimensions; d represents the moment arms; $Q = [q_1, ...q_J]$ represents joint angles; M represents the total number of muscles; J represents the total number of joints

Particle swarm optimization (PSO) [33] was used as the optimization method to minimize the differences of moment arms i.e. errors, between the converted MuJoCo model and the referencing OpenSim model for all muscles at all joints angles. Details of this can be found in the Results section III-A.

Force Property Optimization:
The OpenSim and the MuJoCo platforms also use different methods for modelling muscle force properties. MuJoCo does not consider elastic tendons, fiber pennation angles, while OpenSim has both of them. OpenSim uses physical lengths for the definition of optimal fiber and tendon slack lengths, whereas, MuJoCo uses normalized value [34]. Even though the MuJoCo muscle model is a simplified version of the OpenSim muscle model, it is still possible to generate similar force-length-velocity relationships. However, the muscle property parameters needs to be optimized, instead of directly mapping from the OpenSim model. The following optimization was developed to match the muscle force properties:

Find: muscle force parameters:
$X = \{l_{min}, l_{max}, fp_{max}, f_{max}\}$

To minimize: objective function:

$$F = \sum_{m=1}^{M} (f_m^{osim}(l_m, a) - f_m^{mjc}(l_m, a, X))^2 \tag{2}$$

When: iterate all muscle length l_m and activation a

Subject to: muscle dynamic; parameter bounds

where, l_{min} is the minimal normalized muscle length when active fiber force reaches zero; l_{max} is the maximum normalized muscle length when active fiber force reaches zero; fp_{max} is the normalized passive fiber force when muscle length equal to l_{max}; f_{max} is the absolute value of maximum muscle force; $f(l_m, a)$ represents the muscle forces when the muscle has an activation of a and at the length of l_m.

Differential evolution (DE) [35] was used as optimization method. Muscle activation dynamics and the force-velocity relationship are identical between the OpenSim and MuJoCo models, as a result, they do not need to be optimized, but can be directly transferred. Validation of this step is the errors of force-length-activation maps between the OpenSim and MuJoCo models. More details of it can be found in the Results section III-A.

B. Neuromusculoskeletal systems in MyoSim

1) Elbow model: This model is based on the *arm26* OpenSim model and has two degrees of freedom (shoulder rotation and elbow flexion-extension) and 6 muscles-tendon units (3 flexors and 3 extensors) (Fig. 1-A).

2) Exo-skeletal model: Human-Machine interaction simulations were generated as a use-case to illustrate how musculoskeletal model response to the exoskeleton assistance can be studied. An elbow soft exoskeleton (Cable driven for example) was modeled as an ideal torque actuator perfectly aligned with the elbow joint with a weight of 0.101 Kg for the upper arm and 0.111 Kg on the forearm (Fig. 1-B). The control of the actuator was done using a neuromusculoskeletal based controller, which represents a ideal version of the controller presented in [36], [37]. In this controller, a percentage of the joint torque is given back as assistance.

3) Full hand model: This model (Fig. 1-C) is comprised of 29 bones, with 23 Dofs, and 38 muscle-tendon units and captures the full complexity of an adult human hand.

C. Biological phenomena supported in MyoSim

To challenge our models and to validate the results two realistic conditions were simulated, muscle fatigue and Sarcopenia.

1) Muscle Fatigue: Fatigue is a muscle condition that is inherent to physical activity. Reduction of fatigue in factory worker could decrease work related musculoskeletal injuries as it is a contributing factor to Cumulative trauma disorder [38]. Nevertheless, experimental tests on the effect that exoskeletons can have in reduction of muscle fatigue have rarely been done [39]. This is due to the difficulty of experimentally recording the effect of fatigue at the muscle level. Offering the possibility to test the effect of robotics design and controller on muscle fatigue reduction would be a game changer for the development of cobot or exoskeleton in industrial setting.

A model of Dynamic muscle fatigue based on a dynamic muscle fatigue model [40] was implemented in the *MyoSim* . The following equation was implemented in the muscle model:

$$F^{Max}_{upd}(t) = F^{Max} \exp\left(k_{fatigue} * \int_0^t -\frac{F^{Act}_m(u)}{F^{Max}} du\right) \quad (3)$$

With $F^{Max}_{upd}(t)$ the current updated maximal muscle force, F^{Max} the maximal force without fatigue, F^{Act}_m the active part of the muscle force and $k_{fatigue}$ a fatigue coefficient with a value of 1.

2) Sarcopenia: As we age, muscle mass reduces, which create a reduction of maximal force that muscles can produce. This condition is commonly referred as Sarcopenia [41]. Between the age of 20-40 to older people a reduction of up to 50% of grip force can be observed [42].

In the presented simulation tool, advanced sarcopenia was modelled as a global reduction of 50% of the maximal muscle force.

III. Experiments and Results

A. Validation of Conversion Pipeline

A 2 joints 6 muscles (TRILong - Triceps Long, TRILat - Triceps Lateral, TRImed - TricepsMedial, BIClong - Biceps Long, BICshort - Biceps Short , BRA - Brachialis) OpenSim elbow model [43] is used to validate the conversion pipeline. The three steps validation is shown in Figure 3. The first column shows the model appearances in OpenSim (top) and in MuJoCo (bottom). Second column indicates the matches of markers in forward kinematics checks. As shown in the plot, two selected markers have identical locations in both models, which means the match of geometry and joint definitions. The third column shows the results of moment arm validation. Compared to the OpenSim model, the converted MuJoCo model after Cvt1 (Mjc_Cvt1) has large differences (the orange dash line with triangle markers), however after the second conversion step Cvt2, the moment arm differences between the OpenSim model and the MuJoCo model (Mjc_Cvt2) is greately reduced (the green dashed lines are overlapping on the blue lines). The fourth column shows the validation of muscle forces. A large reduction of differences of muscle force was achieved also by applying the third step of the pipeline. Results of this last step are indicated by the matches between the blue solid line (OpenSim model), and the green dash line (converted MuJoCo model after the third step Mjc_Cvt3). The yellow dash line (converted MuJoCo model shows the intermediate step Mjc_Cvt2). Similar validation metrics and trends were observed in the full hand model (Fig. 1-C) as well, results of this model are not shown in this manuscript because of space constraints. Please visit https://sites.google.com/view/myosuite for full details.

B. Speed Comparison

Simulation speeds between the converted MuJoCo and referencing OpenSim models were tested also. All models were initialized from the neutral posture (all joint angles equal to 0). Muscle activation were set as the square waves switching between 0 and 1, with the duration width of 1 second. For the elbow models, ten repetitions of 5000 seconds simulations were conducted and execution time were averaged. Long forward simulations showed that MuJoCo elbow model is around 170 times more efficient than the OpenSim model. Adding more complex models with more muscles execution times and gap increases. For example, for a full arm model with 50 muscles, MuJoCo performs on average 900 times faster than OpenSim (similar results were reported already by Ikkala et al [19] who showed 600 times increase in speed using MuJoCo as apposed to OpenSim).

C. Human Robot interaction

In the previous validated elbow model, we locked shoulder movements and we added an exoskeleton (see II-B.2). Simulation of human robot interaction were realized for two experiments: I) flexion extension (between 0 - 30, 0 - 60 and 0 - 80 degrees with 2 sec hold time) with an healthy model with different weights on the hand (task inspired by [44]) and II) static with the elbow at 90 degrees with an healthy, Sarcopenia (see section II-C.2) and fatigue (see section II-C.1) conditions. The trials were repeated for the standard arm model with and without the exoskeleton support presented in II-B.2.

We trained a joint policy of exo and muscles by means of a Natural Policy Gradient (NPG) [45] algorithm in the natural (healthy) condition without any weight on the hand to reach random targets in the range of elbow flexions between 0 and 130 degrees of the elbow.

Figure 4 shows the effects of the reaching task (Experiment I) without and with exoskeleton assistance when a load is applied on the hand. As expected the reaching is minimally impacted when the exoskeleton is not functional but there is no additional perturbation. When a weight is applied the exoskeleton is able to recover the original position with about 5 degrees error (distance of joint angles, mean $\pm$ std, 0 Kg: 4.95 $\pm$ 3.42, 1 Kg: 4.54 $\pm$ 3.12, 2 Kg: 4.34 $\pm$ 3.43). This is obtained both with a 60% reduction of both the muscle activation (Figure 4A, BIClong 60%, BICshort 54%, BRA 66%) and forces (Figure 4B, BIClong 60%, BICshort 58%, BRA 67%) at the weigth of 2 Kg.

Figure 5 and Figure 6 show the effects of holding a static position without and with exoskeleton in the presence of

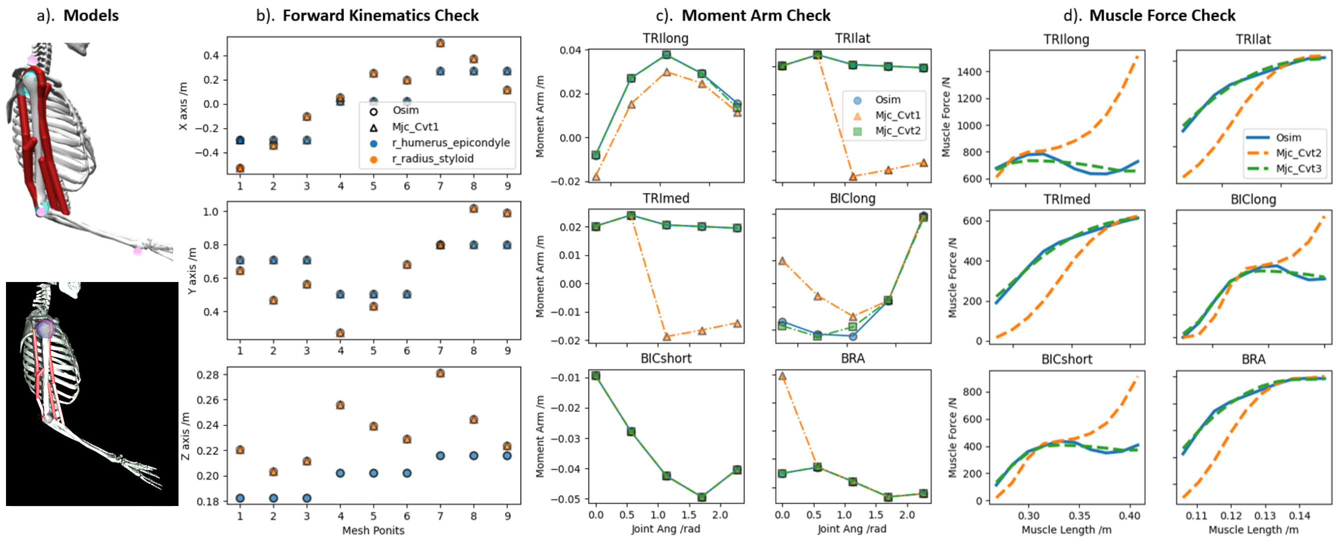

Fig. 3: Validation between the converted MuJoCo and reference OpenSim models. a). The two models, top: OpenSim; bottom: MuJoCo. b). The forward kinematics validation. c). The moment arm validation. d). The muscle force validation. Muscle acronyms TRIlong - Triceps Long, TRIlat - Triceps Lateral, TRImed - TricepsMedial, BIClong - Biceps Long, BICshort - Biceps Short , BRA - Brachialis

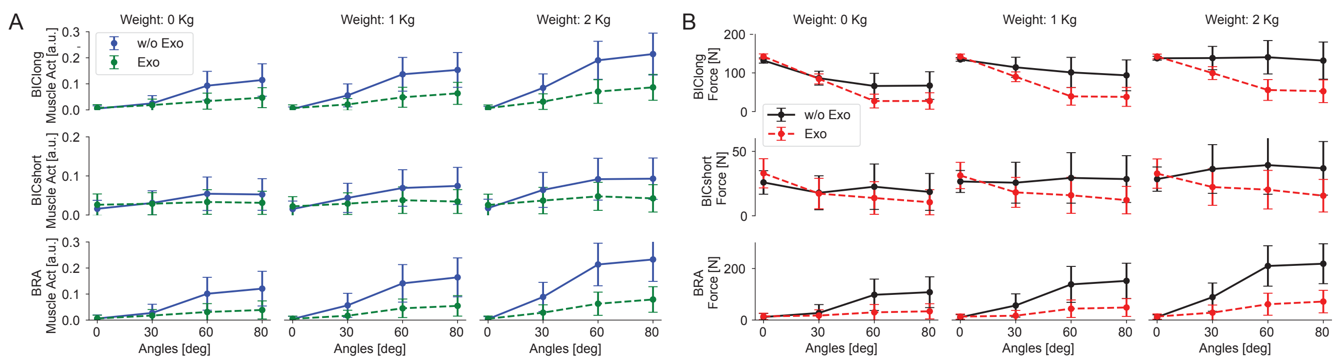

Fig. 4: Effects of loads on Exoskeleton assisted reaching. A policy was trained in a reaching task of exoskeleton assisted elbow flexion to reach targets at 0 deg, 30 deg, 60 deg, 80 deg and hold the position for 2 seconds. A - Flexor muscles activations without (blue) and with (green) exoskeleton assistance at different target positions. With exoskeleton assistance the muscle activation required to reach the target angles was less. B - Same analysis as for A but on forces.

intrinsic perturbation to the muscle models in the form of sarcopenia (muscle weakness) and fatigue. As expected the holding is minimally impacted when the exoskeleton is not functional (distance of joint angles, mean $\pm$ std: 4.12 $\pm$ 2.97) but there is no additional intrinsic perturbation (first row of Figure 5). When intrinsic perturbation are present, the use of the exoskeleton is able to partially recover for those conditions (mean $\pm$ std, Sarcopenia 6.03 $\pm$ 2.97, Fatigue 11.89 $\pm$ 4.03). This is obtained both with a reduction of the muscle activation (Figure 6A, reduction without vs with exoskeleton of Normal, Sarcopenia and Fatigue for BIClong: 40%, 52%, 61%, BICshort: 30%, 44%, 56%; BRA: 41%, 54%, 62%) and forces (Figure 6B, reduction without vs with exoskeleton of Normal, Sarcopenia and Fatigue for BIClong: 41%, 55%, 98%; BICshort 34%, 50%, 98%, BRA: 43%, 56%, 88%).

IV. Discussion and Conclusions

We presented and tested a new pipeline for the generation of fast and physiologically accurate muscoloskeletal models and simulations of human machine interface. Our results showed that the proposed pipeline allows to generate accurate models i.e. close to the current golden standard (*i.e.* Opensim), while are over 2 order of magnitude faster. We also demonstrated the possibility of using Reinforcement Learning to estimate neural command to solve tasks with sophisticated musculoskeletal-robotic models. This was possible thanks to the efficiency of *MyoSim* achieved via the MuJoCo simulation engine.

The pipeline introduced in this paper extended Ikkala and Hamalainen's work [19] in which a conversion process of anatomical information – geometry structures, joint definitions, as well as muscle paths – from the OpenSim to the MuJoCo format was presented. In addition, we developed

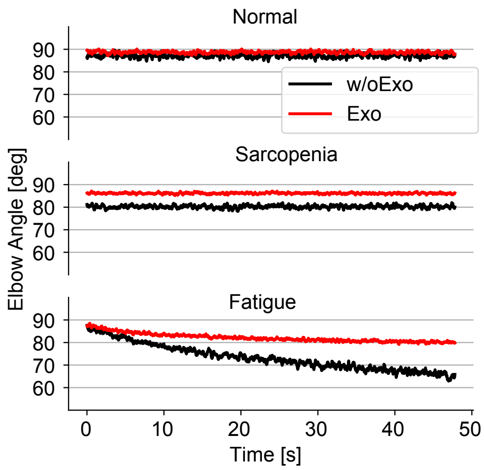

Fig. 5: Effects of Sarcopenia and Fatigue on Exoskeleton assisted reaching. The same policy trained for exoskeleton assisted reaching was tested on holding a 90 deg angle elbow posture. The three rows show the effect of Normal, Sarcopenia and Fatigue respectively on holding the position. During Sarcopenia the muscle loss of force (weakness) is almost completely recovered with the exoskeleton. Likewise for fatigue, the exoskeleton compensate partially for the loss of force. Nevertheless, since the policy was not trained to compensate for those conditions, as results it is not able to drive the exoskeleton to a complete recovery of the function.

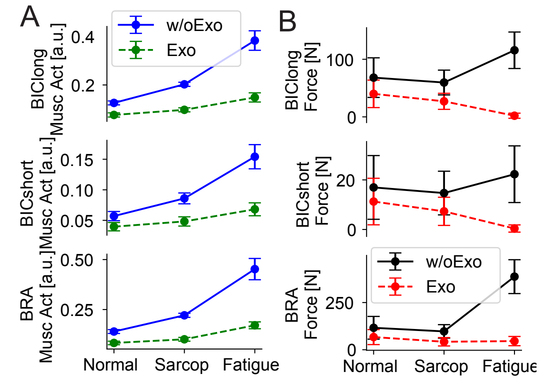

Fig. 6: Effects of Sarcopenia and Fatigue on Exoskeleton assisted holding. Same task and policy as for 5. A - Flexor muscles activations without (blue) and with (green) exoskeleton assistance with the different conditions. With exoskeleton assistance the muscle activation required to compensate for the functional deficit is decreased. B - Same analysis as for A but on the forces.

two new optimization steps to make sure that converted musculoskeletal models have the closest moment arms and muscle force properties to the reference OpenSim models. Differences between models and formats are caused by the different ways of determining the minimal wrapping paths as well the muscle force-length curves. Without these two additional steps, the converted MuJoCo models will diverge from the physiologically accurate behaviour.

One of the most critical aspect of the proposed pipeline was the matching of the moment arm and the force-length-activation data from a reference OpenSim musculoskeletal models. This step can be easily adapted to personalize the model when subject specific data is available. However, the required data include the explicit muscle moment arms, force-length, force-velocity maps, which are difficult to obtain from human subjects *in vivo*. Another popular personalizing approach is based on functional tasks and joint torque fitting that can be easily measured [46]. Whereas, this is out of the scope of this paper, we would like to explore this direction in the future studies.

In this manuscript, we showed examples of how by controlling the simulated human musculoskeletal elbow-exoskeleton we are able to produce realistic results. Namely, the ideal exoskeleton was able to reduce the effect of fatigue and sarcopenia in static trials. In dynamic trials, the exoskeleton was able to reduce EMGs and muscle force levels for different weight ranges. Still, the proposed models of sarcopenia and fatigue are relatively simple models that do not capture the complexity and variety of the physiological systems e.g. differential effect of muscle size and composition. Those models will need to be further extended to better reflect real conditions. Their use in the context of this paper was to show how realistic non-stationarities have an effect on the control of the exoskeleton. Indeed, the trained policy was not made adaptable to the new conditions (weight, sarcopenia and fatigue), which provide further insights into how the control might need to change also in presence of the exoskeleton to adapt to those conditions. A validation of the control policy against experimentally recorded EMGs will be needed.

Finally, real exoskeleton should be simulated and experimental results of human-machine interaction should be used to compare and validate the prediction of the simulation. This is central to further develop tools to predict the effect of robotics device on the human body or even on the central nervous system.

Future steps of this work will involve the extension of the *MyoSim* to include additional musculoskeletal models for other body parts. Also, we will study complex motor tasks that can be validated by experiments, with/without machine interactions. Finally, we will explore new pipelines for development of subject specific musculoskeletal models based on *in vivo* measurements e.g. via MRI scans, ultrasound imaging, EMG and motion recordings.

ACKNOWLEDGMENT

Huawei Wang's, Guillaume Durandau's and Massimo Sartori's contributions in part received funding support from 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

[1] H. Westerblad, J. D. Bruton, and A. Katz, "Skeletal muscle: Energy metabolism, fiber types, fatigue and adaptability," *Experimental cell research*, vol. 316, no. 18, pp. 3093–3099, 2010.

[2] M. Porter, A. Vandervoort, and J. Lexell, "Aging of human muscle: Structure, function and adaptability," *Scandinavian journal of medicine & science in sports*, vol. 5, no. 3, pp. 129–142, 1995.

[3] F. Tamburella, N. Tagliamonte, I. Pisotta, M. Masciullo, M. Arquilla, E. Van Asseldonk, H. Van Der Kooij, A. Wu, F. Dzeladini, A. Ijspeert, *et al.*, "Neuromuscular controller embedded in a powered ankle exoskeleton: Effects on gait, clinical features and subjective perspective of incomplete spinal cord injured subjects," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 28, no. 5, pp. 1157–1167, 2020.

[4] A. Ranavolo, G. Chini, F. Draicchio, A. Silvetti, T. Varrecchia, L. Fiori, A. Tatarelli, P. H. Rosen, S. Wischniewski, P. Albrecht, *et al.*, "Human-robot collaboration (hrc) technologies for reducing work-related musculoskeletal diseases in industry 4.0," in *Congress of the International Ergonomics Association*, Springer, 2021, pp. 335–342.

[5] A. J. Ijspeert, A. Crespi, D. Ryczko, and J.-M. Cabelguen, "From swimming to walking with a salamander robot driven by a spinal cord model," *science*, vol. 315, no. 5817, pp. 1416–1420, 2007.

[6] F. Schuind, M. Garcia-Elias, W. P. Cooney III, and K.-N. An, "Flexor tendon forces: In vivo measurements," *The Journal of hand surgery*, vol. 17, no. 2, pp. 291–298, 1992.

[7] M. Sartori, L. Gizzi, D. G. Lloyd, and D. Farina, "A musculoskeletal model of human locomotion driven by a low dimensional set of impulsive excitation primitives," *Frontiers in computational neuroscience*, vol. 7, p. 79, 2013.

[8] S. Aoi, T. Ohashi, R. Bamba, S. Fujiki, D. Tamura, T. Funato, K. Senda, Y. Ivanenko, and K. Tsuchiya, "Neuromusculoskeletal model that walks and runs across a speed range with a few motor control parameter changes based on the muscle synergy hypothesis," *Scientific reports*, vol. 9, no. 1, pp. 1–13, 2019.

[9] B. S. Gardiner, F. G. Woodhouse, T. F. Besier, A. J. Grodzinsky, D. G. Lloyd, L. Zhang, and D. W. Smith, "Predicting knee osteoarthritis," *Annals of biomedical engineering*, vol. 44, no. 1, pp. 222–233, 2016.

[10] T. F. Besier, M. Fredericson, G. E. Gold, G. S. Beaupré, and S. L. Delp, "Knee muscle forces during walking and running in patellofemoral pain patients and pain-free controls," *Journal of biomechanics*, vol. 42, no. 7, pp. 898–905, 2009.

[11] N. Thatte and H. Geyer, "Toward balance recovery with leg prostheses using neuromuscular model control," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 5, pp. 904–913, 2015.

[12] A. Seth, J. L. Hicks, T. K. Uchida, A. Habib, C. L. Dembia, J. J. Dunne, C. F. Ong, M. S. DeMers, A. Rajagopal, M. Millard, S. R. Hamner, E. M. Arnold, J. R. Yong, S. K. Lakshmikanth, M. A. Sherman, J. P. Ku, and S. L. Delp, "Opensim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement," *PLOS Computational Biology*, vol. 14, pp. 1–20, Jul. 2018.

[13] M. Damsgaard, J. Rasmussen, S. T. Christensen, E. Surma, and M. De Zee, "Analysis of musculoskeletal systems in the anybody modeling system," *Simulation Modelling Practice and Theory*, vol. 14, no. 8, pp. 1100–1111, 2006.

[14] *Software for interactive musculoskeletal modeling*, `https://motionanalysis.com/simm/`.

[15] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," 2016.

[16] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2012, pp. 5026–5033.

[17] J. Lee, M. X. Grey, S. Ha, T. Kunz, S. Jain, Y. Ye, S. S. Srinivasa, M. Stilman, and C. K. Liu, "Dart: Dynamic animation and robotics toolkit," *Journal of Open Source Software*, vol. 3, no. 22, p. 500, 2018.

[18] C. T. Richards and E. A. Eberhard, "In vitro virtual reality: An anatomically explicit musculoskeletal simulation powered by in vitro muscle using closed-loop tissue–software interaction," *Journal of Experimental Biology*, vol. 223, no. 10, jeb210054, 2020.

[19] A. Ikkala and P. Hämäläinen, *Converting biomechanical models from opensim to mujoco*, 2020. arXiv: `2006.10618 [q-bio.QM]`.

[20] F. Fischer, M. Bachinski, M. Klar, A. Fleig, and J. Müller, *Reinforcement learning control of a biomechanical model of the upper extremity*, 2020. arXiv: `2011.07105 [q-bio.QM]`.

[21] M. Sartori and G. S. Sawicki, "Closing the loop between wearable technology and human biology: A new paradigm for steering neuromuscular form and function," *Progress in Biomedical Engineering*, vol. 3, no. 2, p. 023 001, 2021.

[22] T. G. Hornby, D. D. Campbell, J. H. Kahn, T. Demott, J. L. Moore, and H. R. Roth, "Enhanced gait-related improvements after therapist-versus robotic-assisted locomotor training in subjects with chronic stroke: A randomized controlled study," *Stroke*, vol. 39, no. 6, pp. 1786–1792, 2008.

[23] J. Hidler, D. Nichols, M. Pelliccio, K. Brady, D. D. Campbell, J. H. Kahn, and T. G. Hornby, "Multicenter randomized clinical trial evaluating the effectiveness of the lokomat in subacute stroke," *Neurorehabilitation and neural repair*, vol. 23, no. 1, pp. 5–13, 2009.

[24] G. S. Sawicki, O. N. Beck, I. Kang, and A. J. Young, "The exoskeleton expansion: Improving walking and

running economy," *Journal of NeuroEngineering and Rehabilitation*, vol. 17, no. 1, pp. 1–9, 2020.

[25] J. Zhang, P. Fiers, K. A. Witte, R. W. Jackson, K. L. Poggensee, C. G. Atkeson, and S. H. Collins, "Human-in-the-loop optimization of exoskeleton assistance during walking," *Science*, vol. 356, no. 6344, pp. 1280–1284, 2017.

[26] N. A. Bianco, P. W. Franks, J. L. Hicks, and S. L. Delp, "Coupled exoskeleton assistance simplifies control and maintains metabolic benefits: A simulation study," *bioRxiv*, 2021.

[27] S. Song, Ł. Kidziński, X. B. Peng, C. Ong, J. Hicks, S. Levine, C. G. Atkeson, and S. L. Delp, "Deep reinforcement learning for modeling human locomotion control in neuromechanical simulation," *Journal of NeuroEngineering and Rehabilitation*, vol. 18, no. 1, pp. 1–17, 2021.

[28] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine, "Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations," in *Proceedings of Robotics: Science and Systems (RSS)*, 2018.

[29] V. Kumar, E. Todorov, and S. Levine, "Optimal control with learned local models: Application to dexterous manipulation," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 378–383.

[30] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, *et al.*, "Soft actor-critic algorithms and applications," *arXiv preprint arXiv:1812.05905*, 2018.

[31] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen, "Opensim: Open-source software to create and analyze dynamic simulations of movement," *IEEE transactions on biomedical engineering*, vol. 54, no. 11, pp. 1940–1950, 2007.

[32] A. Seth, J. L. Hicks, T. K. Uchida, A. Habib, C. L. Dembia, J. J. Dunne, C. F. Ong, M. S. DeMers, A. Rajagopal, M. Millard, *et al.*, "Opensim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement," *PLoS computational biology*, vol. 14, no. 7, e1006223, 2018.

[33] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, IEEE, vol. 4, 1995, pp. 1942–1948.

[34] E. Todorov. (). "Mujoco modelling documentation," [Online]. Available: `http://www.mujoco.org/book/modeling.html#CMuscle` (visited on 09/14/2021).

[35] K. V. Price, "Differential evolution," in *Handbook of optimization*, Springer, 2013, pp. 187–214.

[36] G. Durandau, D. Farina, G. Asın-Prieto, I. Dimbwadyo-Terrer, S. Lerma-Lara, J. L. Pons, J. C. Moreno, and M. Sartori, "Voluntary control of wearable robotic exoskeletons by patients with paresis via neuromechanical modeling," *Journal of neuroengineering and rehabilitation*, vol. 16, no. 1, pp. 1–18, 2019.

[37] G. Durandau, D. Farina, and M. Sartori, "Robust real-time musculoskeletal modeling driven by electromyograms," *IEEE transactions on biomedical engineering*, vol. 65, no. 3, pp. 556–564, 2017.

[38] D. B. Chaffin, G. B. J. Andersson, and B. J. Martin, *Occupational biomechanics*. John wiley \& sons, 2006.

[39] N. Poon, L. van Engelhoven, H. Kazerooni, and C. Harris, "Evaluation of a trunk supporting exoskeleton for reducing muscle fatigue," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, SAGE Publications Sage CA: Los Angeles, CA, vol. 63, 2019, pp. 980–983.

[40] L. Ma, D. Chablat, F. Bennis, and W. Zhang, "A new simple dynamic muscle fatigue model and its validation," *International Journal of Industrial Ergonomics*, vol. 39, no. 1, pp. 211–220, 2009, ISSN: 01698141.

[41] A. J. Cruz-Jentoft and A. A. Sayer, "Sarcopenia," *The Lancet*, vol. 393, no. 10191, pp. 2636–2646, Jun. 2019, ISSN: 0140-6736.

[42] R. M. Dodds, H. E. Syddall, R. Cooper, D. Kuh, C. Cooper, and A. A. Sayer, "Global variation in grip strength: a systematic review and meta-analysis of normative data," *Age and Ageing*, vol. 45, no. 2, pp. 209–216, Mar. 2016, ISSN: 0002-0729.

[43] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen, "Opensim: Open-source software to create and analyze dynamic simulations of movement," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 11, pp. 1940–1950, 2007.

[44] N. Lotti, M. Xiloyannis, G. Durandau, E. Galofaro, V. Sanguineti, L. Masia, and M. Sartori, "Adaptive model-based myoelectric control for a soft wearable arm exosuit: A new generation of wearable robot control," *IEEE Robotics & Automation Magazine*, vol. 27, no. 1, pp. 43–53, 2020.

[45] S. M. Kakade, "A natural policy gradient," *Advances in neural information processing systems*, vol. 14, 2001.

[46] C. Pizzolato, D. G. Lloyd, M. Sartori, E. Ceseracciu, T. F. Besier, B. J. Fregly, and M. Reggiani, "Ceinms: A toolbox to investigate the influence of different neural control solutions on the prediction of muscle excitation and joint moments during dynamic motor tasks," *Journal of biomechanics*, vol. 48, no. 14, pp. 3929–3936, 2015.