2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

Assistive Sliding Mode Control of a Rehabilitation Robot with Automatic Weight Adjustment

Arash Hashemi[1] and John McPhee[2]

Abstract— **There are approximately 13 million new stroke cases worldwide each year. Research has shown that robotics can provide practical and efficient solutions for expediting post-stroke patient recovery. This simulation study aimed to design a sliding mode controller (SMC) for an end-effector-based rehabilitation robot. A genetic algorithm (GA) was designed for automatic controller weight adjustment. The optimal weights were obtained by minimizing a cost function comprising the end-effector position error, robot input, robot input-rate, and patient input. To promote safe tuner optimization, a model of the human arm was incorporated to generate the human joint torque. A computed-torque proportional derivative controller (CTPD) was designed for the human arm to approximate the central nervous system (CNS) motor control. This controller was adjusted to simulate rehabilitation effects and patient adaptation. The tuner was optimized for a trajectory tracking task with an assistive high-level control scheme. The simulation results showed lower cost compared to seven manual weight settings. The optimal weights provided good tracking performance and suitable robot inputs. This research provides a framework to conduct various simulations before testing our controller on human subjects. The preliminary results of this study will be used as the starting point for online adaptive controller tuning, which will be examined in our future research.**

I. INTRODUCTION

As the leading cause of adult disability [1], and one of the prevalent causes of global death, stroke has brought about physiological, psychological, social, and economic impacts [2]. Post-stroke patients deal with a myriad of neurological difficulties including, but not limited to, hemispheric behavioral difference, perceptual dysfunction, and ostoeporosis and fracture risk [3]. Hence, finding solutions for the rehabilitation of post-stroke patients is of utmost importance. Fortunately, robot-assisted rehabilitation has proven helpful in improving the patients' motor recovery. Robots can show utility by automating the repetitive rehabilitation practices, hence saving a considerable time and effort; they reduce the workload of physical therapists and increase the number of patients treated by each therapist [4]. In addition, robots can be equipped with sensors and biomarkers to quantitatively assess the patients' progress [5].

End-effector-based rehabilitation robots are a subbranch of assistive devices that have been beneficial for restoring the patients' motor functions. As a result, researchers have been studying and developing new designs. In this research, we took advantage of a planar manipulundum rehabilitation robot, designed by Quanser Inc., Toronto Rehabilitation Institute (TRI), and the University of Waterloo. This robot provides unilateral training of the upper extremity; the patient grasps the end-effector and the robot automates the repetitive practices in the horizontal plane of motion.

Various control strategies have been proposed for rehabilitation robots, assistive devices, and exoskeletons [6]. Previous studies have investigated the use of Proportional-Integral-Derivative (PID) controllers [7]. These controllers, albeit simple to implement, face performance degradation when operating under post-stroke patient disturbance. Model predictive controllers (MPC) have also been investigated on rehabilitation robots [8]; however, MPC has high computational cost due to its online optimization, and its performance depends on the hardware update frequency and processing power. Sliding mode control (SMC) was selected in our research as the low-level controller because of its high performance on nonlinear dynamical systems, relatively low computational cost, and its ability to handle disturbances. This controller has been previously examined on rehabilitation robots [4], as well as exoskeletons [9]. On top of the low-level sliding mode controller, in this study, we adopted a high-level assistive control scheme. We targeted early stages of active rehabilitation, when the patient can exert force but still requires assistance from the robot to finish the task; with this regard, the patient benefits both from staying on the desired trajectory and engaging in the practice; i.e. full assistance from the robot with no engagement from the patient is not desirable.

Regardless of the choice of the strategy, each controller contains weights that require meticulous tuning. Manual weight adjustment is a time-consuming and exhaustive process. Thus, automatic weight tuning approaches have been proposed in the literature. A stream of research has focused on method-based strategies [10] for automotive applications. However, these approaches do not acquire optimal weight values. Other studies have utilized reinforcement learning (RL) for automatic weight adjustment [14], [15]. While RL generalizes better to high-space problems than other methods [14], it is more difficult to implement and adds more computational burden than many of the existing approaches; moreover, most of the deep RL methods are prone to getting stuck at a local optima. As a result, other solutions like gradient-descent [11], Bayesian optimization [13], and evolutionary algorithms [12] are preferred for problems with less than five states and fixed-weight calculations; having said that, RL methods are a good choice for adaptive weight

*This work was funded by the Canada Research Chairs Program and the Natural Sciences and Engineering Research Council of Canada (NSERC).

[1] Arash Hashemi is with Systems Design Engineering, 200 University Ave. W., Waterloo, Canada a8hashem@uwaterloo.ca

[2] John McPhee is a professor in Systems Design Engineering, 200 University Ave. W., Waterloo, Canada mcphee@uwaterloo.ca

978-1-7281-1178-0/21/$31.00 ©2021 IEEE

adjustment.

We took advantage of the genetic algorithm (GA) method for automatic controller weight adjustment. This meta-heuristic approach uses search-based global optimization techniques based on natural selection and biological evolution [16] and has been previously utilized for controller tuning in robotic applications [17], hence is a good candidate for our purpose. In general, most research on controller tuning for clinical purposes has either focused on impedance/admittance controllers [14], while manually tuning the low-level controllers, or has predefined an approximate torque profile for the controller [12]. In our study, we examined the weight adjustment of low-level controllers, which has a major effect on the performance of the robot. In addition, the tuning algorithms in the aforementioned studies either run online with the human in the loop, or with offline human data gathered from interaction with an untrained agent. While this approach leads to subject-specific adjustment on exoskeletons and prostheses, it is not recommended for rehabilitation practices, especially for early active stages, due to safety considerations. We conducted the tuning process offline in simulation. To bypass the need for actual human data, a realistic model of the patient is required. To this end, we leveraged a planar model of the human arm. We applied a computed-torque proportional-derivative (CTPD) controller as the human motor controller. A passive torque term was added to the CTPD to consider muscle viscosity and stiffness. Also, human adaptation to the robot was modelled by adding a slacking term. The coefficients of CTPD were changed at each rehabilitation cycle to simulate the patient progress.

In this simulation study, we used a GA for automatic SMC weight tuning on a rehabilitation manipulundum. It should be noted that compared to other controllers, less intuition exists for choosing SMC weights. Hence, an automatic tuner scheme would help the design of this specific controller. A human arm model with a motor controller was utilized in the loop to simulate the effect of post-stroke patients. The GA-SMC structure can potentially benefit other assistive devices, prostheses, and exoskeletons. Furthermore, the same tuner scheme can be used with other low-level controllers. The results of this work provide the initial weights for the online tuning schemes; instead of initiating the tuner randomly, the first parameters would be set as values that are validated in simulation. To the best of the authors' knowledge, this is the first work on realistic human-robot simulations with automatic weight adjustment for rehabilitation and clinical purposes.

The remainder of this paper is structured as follows: Section II describes the modeling of the robot and the subject arm. Section III explains the design of the human motor controller. Section IV highlight the SMC controller design. Section V discusses the GA tuner and its integration with the controller. Section VI presents the simulation results and the discussion about the findings and finally, Section VII draws conclusions from this work.

II. Modeling

A. Robotic Arm

The robot is a 2 degrees of freedom (DOF) 4-linkage fully actuated planar parallelogram manipulator. The dynamic model of the robot was obtained using the Lagrange energy method in terms of $\mathbf{q} = [q_1, q_2]^T \in \mathbb{R}^2$ generalized coordinates, the vector of actuated robot joint angles. The following are the robot equations of motion:

$$\mathbf{M}_R(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_R(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} = \tau_R + \tau_H \tag{1}$$

$\tau_R \in \mathbb{R}^2$ is the vector of robot motor torques. $\tau_H \in \mathbb{R}^2$ represents the torque resulting from the applied human force to the end-effector. $\mathbf{M}_R(\mathbf{q}) \in \mathbb{R}^{2\times2}$ is the symmetric positive definite inertia matrix. $\mathbf{C}_R(\mathbf{q},\dot{\mathbf{q}}) \in \mathbb{R}^{2\times2}$ is the Coriolis matrix, which contains centrifugal forces. Setting joint angles and angular velocities as states $\mathbf{x} = [\mathbf{q}, \dot{\mathbf{q}}]^{\mathbf{T}} \in \mathbb{R}^{4\times1}$, the state-space model of the robot is derived as follows:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{M}_R(\mathbf{q})^{-1}\tau_R + \Gamma(\mathbf{q},\dot{\mathbf{q}}) + \Phi(\mathbf{q}) \end{bmatrix}_{4\times1} \tag{2}$$

where $\Gamma(\mathbf{q},\dot{\mathbf{q}}) = -\mathbf{M}_R(\mathbf{q})^{-1}\mathbf{C}_R(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}$ and $\Phi(\mathbf{q}) = \mathbf{M}_R(\mathbf{q})^{-1}\tau_H$.

B. Human Arm

Online training with subjects in the loop can cause major safety issues in rehabilitation robotics and careful simulation studies are required before testing with human subjects. To consider the patient in simulation, we developed a two DOF planar human arm model consisting of two rigid links to be integrated with the robot. This model was used to simulate elbow flexion-extension and shoulder abduction-adduction in upper-extremity planar movements. A male subject (height= 1.71 m, weight=85 kg) was considered and the Winter anthropometric tables [20] were used to scale the segment lengths, masses, center of mass (COM) positions, and moments of inertia. The equations of motion of the arm are similar to the robot and hence, are not presented. The human-robot interaction is depicted in Fig. 1.

III. Human Motor Controller

Various studies have verified that the central nervous system (CNS) commands a combination of feedforward and feedback terms for human motor control [21]. The feedforward term is generated by approximating an internal model of the arm, and the feedback term is used to correct errors by making use of sensory organs. To simulate the CNS neural command, a computed-torque proportional-derivative (CTPD) controller was designed for the human arm. The dynamics of the arm were utilized to calculate the feedforward term and the PD signal produced the feedback portion of the controller. Similar to [21], it was assumed that the sensory data was fully available to the CNS. The motor controller is formulated as follows:

$$\tau_C = \mathbf{M}_A(\theta)(\ddot{\theta}_d + k_p\mathbf{e}_p + k_d\dot{\mathbf{e}}_p) + \mathbf{C}_A(\theta,\dot{\theta})\dot{\theta} \tag{3}$$

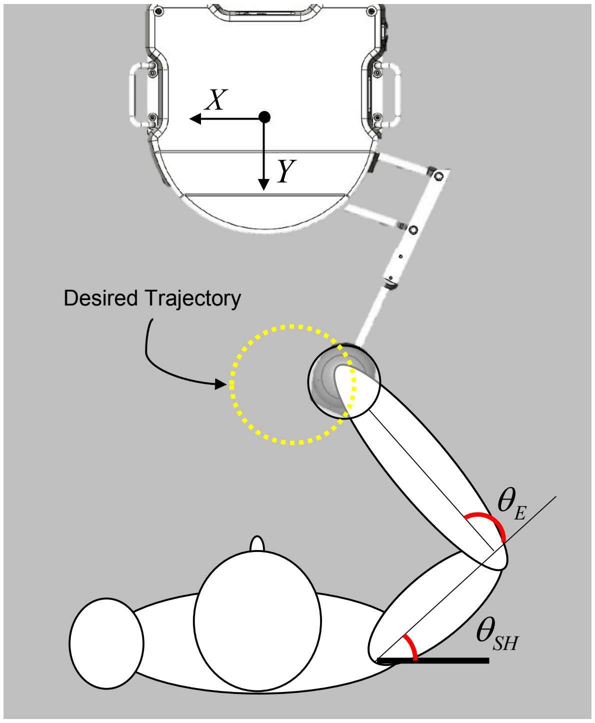

Fig. 1. Human-robot interaction.

where τ_C is the neural command generated by CNS. $\mathbf{M}_A(\theta)$ and $\mathbf{C}_A(\theta,\dot{\theta})$ are the human arm inertia and Coriolis matrices. $\theta = [\theta_1,\theta_2]^T \in \mathbb{R}^2$ is the vector of arm joint angles, denoting the elbow flexion-extension θ_E and shoulder abduction-adduction θ_{SH}, respectively. τ_A is the human arm joint torques. $\mathbf{e}_p$, $\dot{\mathbf{e}}_p$, are the hand position and velocity errors, respectively. To reduce the computational cost of running hundreds of optimizations, full muscle dynamics were not included in the human arm model; however, to help promote realistic simulations, the passive term of muscle torque generators (MTGs) [23], was added to the neural command to account for the muscle viscosity and nonlinear stiffness properties; One MTG was considered for each degree of freedom, i.e. the elbow and the shoulder. This term was formulated using a double exponential function [23]:

$$\tau_P(\theta,\dot{\theta}) = k_1 e^{-k2(\theta-\theta_{min})} - k_3 e^{-k4(\theta_{max}-\theta)} - c\dot{\theta} \tag{4}$$

where θ_{min}, θ_{max} denote the bounds on the range of motion of arm joints. The parameters k_1, k_2, k_3, k_4, and c were obtained from [24], [25], [26] and adjusted based on planar rehabilitation movements. For instance, the linear damping coefficient c was increased from 0.1 (the reported value in [26]) to account for the higher muscle viscous damping of post-stroke patients. The overall human arm torque is as follows:

$$\tau_A = \tau_C + \tau_P \tag{5}$$

Finally the effect of human joint torque τ_A on the robot, which we previously represented as τ_H, was obtained in (6).

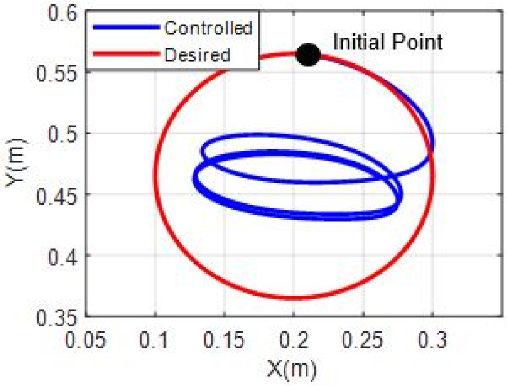

Fig. 2. The hand position in XY for human-only trajectory tracking.

$\mathbf{J}_R$ and $\mathbf{J}_A$ denote the robot and the human arm geometric Jacobian matrices, respectively.

$$\tau_H = \mathbf{J}_R^T \mathbf{J}_A^{-T} \tau_A \tag{6}$$

When interacting with a robotic device, human motor control adapts itself to the robot [27]. One of the adaptation mechanisms in repetitive movements is human-slacking in which the motor control decreases the human engagement (the patient joint torque) when tracking errors are small [28]. This is imperative to model in active rehabilitation settings to avoid too much assistance by the robot and promote patient engagement. To this end, at each simulation cycle, which is the completion of one rehabilitation practice, we compared the tracking error of the robot-assisted movement with the human-only movement. When the former error was smaller, the arm torque was modified as follows [28]:

$$\tau_A^{j+1} = \tau_C^{j+1} + \tau_P^{j+1} - F_f \tau_A^j \tag{7}$$

Where j denotes the simulation cycle, and F_f denotes the human-slacking factor that adjusts the decreased human engagement when tracking errors are small [28].

Also, we simulated the rehabilitation progress by increasing k_p and k_d values at each cycle based on the tracking error and human engagement. Hence, we started from low values of $k_p = 1$ and $k_d = 1$ to account for post-stroke muscle stress scaling [22], and changed the coefficients:

$$\begin{aligned} k_p &= k_p + \delta k_p \\ k_d &= k_d + \delta k_d \end{aligned} \tag{8}$$

The human performance was first simulated without the robot for tracking a circular trajectory. The initial point was selected on the desired trajectory. The human hand in XY is depicted in Fig. 2. As shown, the human arm could not present good tracking without the robot assistance. This is due to the fact that low coefficients were set for the CNS controller to simulate a post-stroke patient.

IV. SMC Robot Controller

The goal of SMC is to drive the error to a stable sliding surface and then move along it to the origin. This algorithm is inherently robust to model uncertainties and/or external disturbances [18] and thus, is a good candidate for rehabilitation robot control under patient disturbance. The sliding surface $\mathbf{s}$ is defined as (10) with $\mathbf{e}$ representing the error between the desired and the controlled robot joint angles, which converges to zero as $\mathbf{s} = 0$ (as shown in (11)). λ is an adjustable weight and determines the speed of convergence.

$$\mathbf{e} = \mathbf{q} - \mathbf{q}_d \tag{9}$$

$$\mathbf{s} = \mathbf{e} + \lambda \dot{\mathbf{e}} \tag{10}$$

$$\dot{\mathbf{e}} = -\lambda \mathbf{e} \tag{11}$$

The control input consists of equivalent control term $\mathbf{u}_{eq}$ and stabilizing control term $\mathbf{u}_s$, the former for forcing the error to the origin and the latter for achieving robustness [19]. The equivalent control term is formulated as follows:

$$\mathbf{u}_{eq} = -\mathbf{M}_R(\mathbf{q})(\hat{\Gamma}(\mathbf{q},\dot{\mathbf{q}}) + \hat{\Phi}(\mathbf{q}) + \ddot{\mathbf{q}}_d - \lambda \dot{\mathbf{e}}) \tag{12}$$

where $\hat{\Gamma}(\mathbf{q},\dot{\mathbf{q}})$ and $\hat{\Phi}(\mathbf{q})$ are the nominal robot and human terms, respectively. The stabilizing control term is presented below:

$$\mathbf{u}_s = -K sign(\mathbf{s}) \tag{13}$$

where K is another tunable weight. The derivation and stability proof of formulations in (12) and (13) are discussed in [19]. The discontinuity in (13) was approximated with a continuous *tanh* function with boundry layer ψ to circumvent the common chattering effect in SMC. The final sliding mode control term was defined as below. This controller includes three tunable parameters λ, K, and ψ:

$$\begin{aligned} \mathbf{u}_s &= -K\ tanh(\mathbf{s}/\psi) \\ \tau_R &= \mathbf{u}_{eq} + \mathbf{u}_s \end{aligned} \tag{14}$$

V. Tuner Structure

The GA toolbox in MATLAB R2021a was utilized for the tuner. It is worthwhile to review that although the tracking performance is an important factor for the controller, it should also promote human engagement. As a result, while the controller is capable of perfect tracking, it is not necessarily desirable in active rehabilitation because it leads to the previously-discussed human-slacking phenomenon. Hence, a trade-off between tracking performance and human contribution should be considered when adjusting the weights. Also, even with the continuous stabilizing term, SMC is still prone to high inputs and input frequency. Overall, a combination of position error, robot torque, robot input-rate, and human torque was set as the GA cost function:

$$\begin{aligned} J = &\frac{1}{N_c}\sum_{j}^{N_c}\sum_{i}^{N_t}(W_1\frac{|P_{ij}-P_{di}|_2}{P_{max}} + W_2\frac{|\tau_{Rij}|_2}{\tau_{Rmax}} \\ &+ W_3\frac{|\dot{\tau}_{Rij}|_2}{\dot{\tau}_{Rmax}} - W_4\frac{|\tau_{Aij}|_2}{\tau_{Amax}}) \end{aligned} \tag{15}$$

N_t is the number of simulation timesteps and N_c is the number of simulation cycles. We set $N_c = 5$ to be able to model human-slacking and rehabilitation progress. P and P_d are the controlled and desired end-effector positions, respectively. $\dot{\tau}_R$ is the input-rate vector. Note that the human torque term has a negative sign in the cost function as high human torques were desirable. We used the human arm model and the CNS controller to calculate the human torque. In the experimental implementation, this value can be estimated by robot sensors, like the force sensor on the end-effector, or by the same model. Recently, there has been studies on using electromyography (EMG) signals and deep neural networks to estimate human torques [29], which can also be useful for the experiments. The cost terms were weighted and normalized. $W_1 = 200, W_2 = 0.01, W_3 = 0.5$, and $W_4 = 1$ were chosen to emphasize high tracking performance, human engagement, and low robot torque and torque frequency. The optimization ran for 20 generations, with 20 population size each. Since very low and high weight values can hurt the stability of the controller, [0.1, 100] were set as the lower and upper optimization bounds.

TABLE I

Cost comparison between the optimal weights and seven manual weight settings

Weights	Cost
$[\lambda = 5 \quad K = 5 \quad \psi = 5]$	48000
$[\lambda = 10 \quad K = 10 \quad \psi = 10]$	53389
$[\lambda = 20 \quad K = 20 \quad \psi = 20]$	58961
$[\lambda = 40 \quad K = 40 \quad \psi = 40]$	63508
$[\lambda = 60 \quad K = 60 \quad \psi = 60]$	66323
$[\lambda = 80 \quad K = 80 \quad \psi = 80]$	69118
$[\lambda = 100 \quad K = 100 \quad \psi = 100]$	72618
$[\lambda^* = 1.5749 \quad K^* = 9.989 \quad \psi^* = 4.8699]$	**46103**

VI. Results and Discussion

The optimal weights were found by solving a GA optimization problem for a circular trajectory tracking scenario. During the simulations, we assumed that the added human arm inertia to the robot was negligible due to the fairly low accelerations. The optimization took approximately 1 hour on a typical desktop CPU. As mentioned, patient data was not required for the process as the arm model simulated a post-stroke patient during training. The optimal weight setting was $[\lambda^* = 1.5749 \quad K^* = 9.989 \quad \psi^* = 4.8699]$. These values are difficult to achieve by manual tuning and require meticulous adjustment and multiple simulations, which takes a considerable time and effort. The optimal values were compared with seven manual weight settings in terms of the associated cost in Table I. The adjusted weights presented a lower cost than all the manual settings. Fig. 3 plots the end-effector trajectory

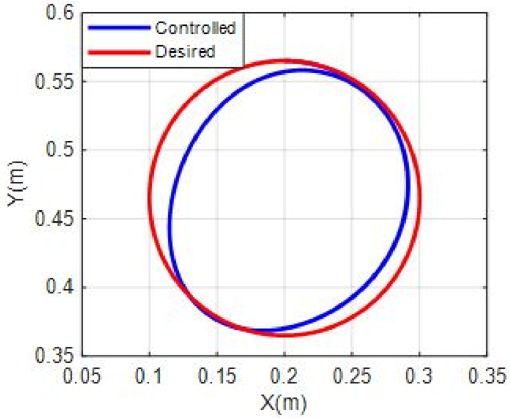

Fig. 3. The end-effector position in XY for robot-assisted trajectory tracking.

with the optimal weights. It is worthwhile to mention that the end-effector and the human arm have the same XY position during rehabilitation. The robot-assisted tracking shows much better performance than the human-only results (See Fig. 2). The root-mean-squared-error (RMSE) along X and Y axes were 8.8 mm and 10.2 mm. The velocity RMSE along X and Y axes were 8.3 mm/s and 10.3 mm/s. The robot torque inputs are shown in Fig. 4. The inputs are well within the robot allowable torque bound ($-10\ N.m < \tau_R < 10\ N.m$). They also have acceptable frequency and low input rate, which makes them suitable for application on the physical hardware.

VII. CONCLUSIONS

In this study, a sliding mode controller (SMC) was designed and applied to an end-effector-based rehabilitation robot in simulation. A human arm model was integrated with the robot, and a computed-torque PD (CTPD) was considered as the central nervous system (CNS) controller. We modified this controller by adding the passive element of muscle-torque generators (MTG) and modelling human adaptation and progress during rehabilitation. The resulting model enables the simulation of “what-if” situations before transferring to experimental implementation. An automatic tuner was designed based on a genetic algorithm (GA) to find the best set of SMC weights for a circular trajectory tracking scenario. The GA cost penalized tracking errors, high input, and rate of input while rewarding patient engagement. The optimal set of weights had a lower cost compared to seven manual weight settings. The full simulation of the controller with the optimal weights resulted in good tracking performance with maximum RMSE of 10.2 mm and 10.3 mm/s for the position and velocity, respectively. The results also showed low inputs and input rates. The obtained values from this simulation study can be used as the starting point for the online training of controller tuners, hence avoiding the use of untrained agents and increasing safety in patient-in-the-loop (PIL) rehabilitation research. It should be noted that patients might change their muscle activation patterns at each trial due to adaption, fatigue, etc. Some of these effects were included in this study. To further explore these elements, we will investigate the application of adaptive online tuners to account for subject-specific muscle activation patterns. To this end, data-driven approaches like model-free reinforcement learning (MFRL) are suggested.

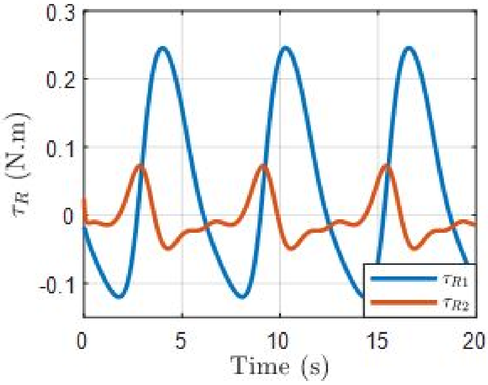

Fig. 4. The robot torque inputs.

ACKNOWLEDGMENT

The authors wish to thank Quanser Inc. for providing the upper limb rehabilitation robot, and Dr. Borna Ghannadi for upgrading the robot hardware.

REFERENCES

[1] F. Pistoia, S. Sacco, C. Tiseo, D. Degan, R. Ornello, and A. Carolei. “The epidemiology of atrial fibrillation and stroke,” in *Cardiology Clinics*, vol. 34, pp. 255-268, 2016, doi: 10.1016/j.ccl.2015.12.002.

[2] “The global burden of disease: 2004 update”. World Health Organization, 2008, [Online]. Available: `https://bit.ly/3khvdNT`.

[3] B. Ghannadi, R. S. Razavian, and J. McPhee, “Upper extremity rehabilitation robots: A survey,” in *Handbook of Biomechatronics*: Elsevier, pp. 319-353, 2019.

[4] J. Niu, Q. Yang, X. Wang, and R. Song, “Sliding mode tracking control of a wire-driven upper-limb rehabilitation robot with nonlinear disturbance observer,” in *Front. Neurol*, 2017, doi: 10.3389/fneur.2017.00646.

[5] S. M. Mostafavi, P. Mousavi, S. P. Dukelow, and S. H. Scott, “Robot-based assessment of motor and proprioceptive function identifies biomarkers for prediction of functional independence measures,” in *J Neuroeng Rehabil.*, vol. 12, no. 105, 2015, doi: 10.1186/s12984-015-0104-7.

[6] P. Maciejasz, J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Troy, and S. Leonhardt, “A survey on robotic devices for upper limb rehabilitation,” in *Journal of Neuroeng. Rehab.*, no. 3, 2014.

[7] W. Yu, and J. Rosen, “A novel linear PID controller for an upper limb exoskeleton,” in *49th IEEE Conf. Decision and Control (CDC)*, Atlanta, GA, USA, 2010.

[8] B. Ghannadi, N. Mehrabi, R. S. Razavian, snd J. McPhee, “Nonlinear model predictive control of an upper extremity rehabilitation robot using a two-dimensional human-robot interaction model,” in *IEEE/RSJ Int. Conf. Intel. Robots. Systems (IROS)*, Vancouver, BC, Canada, 2017.

[9] A. Riani, T. Madani, A. Benallegue, and K. Djouani, “Adaptive integral terminal sliding mode control for upper-limb rehabilitation exoskeleton,” in *Control Eng. Practice*, vol. 75, pp. 108-117, 2018, doi: 10.1016/j.conengprac.2018.02.013.

[10] T. Takahama, and D. Akasaka, “Model predictive control approach to design practical adaptive cruise control for traffic jam,” in *Int. Journal Automotive Eng.*, vol. 9, no. 3, pp. 99-104, 2017, doi: 10.20485/JSAEIJAE.9.399.

[11] J. R. Koller, D. H. Gates, D. P. Ferris, and C. D. Remy, “Body-in-the-Loop optimization of assistive robotic devices: A validation study,” in *Robot. Science and Systems*, pp. 1-10, 2016.

[12] 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,”, in *Science*, vol. 356, no. 6344, pp. 1280-1284, 2017, doi: 10.1126/science.aal5054.

[13] Y. Ding, M. Kim, S. Kuindersma, and C. J. Walsh, “Human-in-the-loop optimization of hip assistance with a soft exosuit during walking,” in *Science Robot.*, 2018, doi: 10.1126/scirobotics.aar5438.

[14] Y. Wen, J. Si, A. Brandt, X. Gao, and H. H.Huang, “Online reinforcement learning control for the personalization of a robotic knee prosthesis,”. in *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2346-2356, 2020, doi: 10.1109/TCYB.2019.2890974.

[15] M. Li, X. Gao, Y. Wen, J. Si, and H. H. Huang, “Offline policy iteration based reinforcement learning controller for online robotic knee prosthesis parameter tuning,” in *2019 International Conference on Robots. Automation (ICRA)*, Montreal, Canada, 2019.

[16] S. Katoch, S. S. Chauhan, and V. Kumar, “A review on genetic algorithm: past, present, and future,” in *Multimedia Tools and Applications*, pp. 1-36, 2020.

[17] T. Salloom, X. Yu, W. He, and O. Kaynak, “Adaptive Neural Network Control of Underwater Robotic Manipulators Tuned by a Genetic Algorithm,” in *Journal of Int. Robot. Systems.*, vol. 97, pp. 657-672, 2020.

[18] A. Pisano, and E. Usai, “Sliding mode control: A survey with applications in math,” in *Mathematics and Computers in Simulation*, vol. 81, no. 5, pp. 954-979, 2011, doi: 10.1016/j.matcom.2010.10.003.

[19] M. Rahmani, H. Komijani, and M. H.Rahman, “New sliding mode control of 2-DOF robot manipulator based on extended grey wolf optimizer,” in *Int. Journal Control Automation Systems*, vol. 18, no. 6, pp. 1572-1580, 2020, doi: 10.1007/s12555-019-0154-x.

[20] D. A. Winter, “Biomechanics and motor control of human movement“, 4th ed., Hoboken, N.J: Wiley, 2009, oCLC: ocn318408191.

[21] N. Mehrabi, R. R. Sharif, B. Ghannadi, and J. McPhee, “Predictive simulation of reaching moving targets using nonlinear model predictive control,” in *Front. Comput. Neurosci.*, 2017, doi: 10.3389/fncom.2016.00143.

[22] M. Asghari, S. Behzadipour, and Gh. Taghizadeh, “A planar neuromusculoskeletal arm model in post-stroke patients,” in *Biological Cybernetics*, 2018, doi: 10.1007/s00422-018-0773-y.

[23] K. Inkol, C. Brown, W. McNally, C. Jansen, and J. McPhee, ”Muscle torque generators in multibody dynamic simulations of optimal sports performance,” in *Multibody System Dynamics*, 2020, doi: 10.1007/s11044-020-09747-9.

[24] W. McNally and J. McPhee, ”Dynamic optimization of the golf swing using a six degree-of-freedom biomechanical model,” in *Multidisciplinary Digital Publishing Institute Proceedings*, vol. 2, no. 6, pp. 243, 2018, doi: 10.3390/proceedings2060243.

[25] Brown, C., McPhee, J.: ”Predictive forward dynamic simulation of manual wheelchair propulsion on a rolling dynamometer,” in *ASME J. Biomech. Eng.* 142(7), 071,008 (2020), doi: 10.1115/1.4046298.

[26] Yamaguchi G.T.: ”Dynamic modeling of musculoskeletal motion: a vectorized approach for biomechanical analysis in three dimensions,” Springer, Berlin (2006). OCLC: 150263239.

[27] Huang. H, Si. J, Brandt. A, and Li. Minhan: ”Taking both sides: seeking symbiosis between intelligent prostheses and human motor control during locomotion,” in *Current Opinion in Biomed. Eng.*, vol. 20, 2021, doi: 10.1016/j.cobme.2021.100314.

[28] Reinkensmeyer, D., Akoner, O., Ferris, D., Gordon, K. (2009): ”Slacking by the human motor system: Computational models and implications for robotic orthoses,” in *2009 Engineering in Medicine and Biology Society (EMBC) Annual International Conference of the IEEE*, Minnesota, USA, 2009.

[29] Nasr A., Bell S., He J., Whittaker R. L., Jiang N., Dickerson C. R., and McPhee J.:”MuscleNET: mapping electromyography to kinematic and dynamic biomechanical variables by machine learning,” in *bioRvix*, 2021, doi: 10.1101/2021.07.07.451532.