2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
September 24–28, 2017, Vancouver, BC, Canada

Nonlinear Model Predictive Control of an Upper Extremity Rehabilitation Robot Using a Two-Dimensional Human-Robot Interaction Model

Borna Ghannadi, Naser Mehrabi, Reza Sharif Razavian and John McPhee

Abstract— **Stroke rehabilitation technologies have focused on reducing treatment cost while improving effectiveness. Rehabilitation robots are generally developed for home and clinical usage to: 1) deliver repetitive practice to post-stroke patients, 2) minimize therapist interventions, and 3) increase the number of patients per therapist, thereby decreasing the associated cost. The control of rehabilitation robots is often limited to black- or gray-box approaches; thus, safety issues regarding the human-robot interaction are not easily considered. To overcome this issue, controllers working with physics-based models gain more importance. In this study, we have developed an efficient two dimensional (2D) human-robot interaction model to implement a model-based controller on a planar end-effector-type rehabilitation robot. The developed model was used within a nonlinear model predictive control (NMPC) structure to control the rehabilitation robot. The GPOPS-II optimal control package was used to implement the proposed NMPC structure. The controller performance was evaluated by simulating the human-robot rehabilitation system, modeled in MapleSim®. In this system, a musculoskeletal model of the arm interacting with the robot is used to predict movement and muscle activation patterns, which are used by the controller to provide optimal assistance to the patient. In simulations, the controller achieved desired performance and predicted muscular activities of the dysfunctional subject with a good accuracy. In our future work, a structure exploiting the NMPC framework will be developed for the real-time control of the rehabilitation robot.**

I. INTRODUCTION

Stroke is the leading cause of chronic upper extremity impairments in older adults [1], [2]. Intense and motivating rehabilitation therapy has shown promising results in stimulating the neural plasticity to treat post-stroke movement impairments [3]. Hence, upper extremity rehabilitation robots are developed targeting this kind of therapy [4].

To stimulate neural plasticity, three modes of high-level control scenarios are used in rehabilitation robots: assistive, corrective, and resistive (see Fig. 1) [5], [6]. Consequently, low-level control scenarios are needed to implement those high-level control modes. Assistive control mode is the most common in these robots, and the following approaches are used for this mode: 1) Passive control–passive trajectory tracking, passive mirroring and passive stretching, 2) Triggered passive control, and 3) Partially assistive control–admittance/impedance control, attractive force field control, model-based assistive control, and offline adaptive control [6] (refer to Fig. 1).

In robotic rehabilitation, although the patient is physically interacting with a mechanical device, these robots mostly use black-box approaches (Fig. 1), which do not consider human-robot interaction. Thus, safety issues may become a concern. For this reason, gray-box (triggered passive), kinematic-based (attractive force field), model-based and artificial-intelligence (AI) (offline adaptive mode) controllers have been developed to reduce the safety risks [6] (see Fig. 1). Though gray-box controllers incorporate feedback from the human body using biosignals [7], they require intensive preparation and are therefore not practical for daily usage with multiple post-stroke patients. Among other controllers, model-based control is more advanced, since: 1) it can provide more information regarding the human-robot dynamic interaction, 2) it can be used for dynamic parameter identification of the human body, 3) in contrast to black/gray-box controllers, a physics-based model can leverage the inherent dynamics of the system and facilitate implementation of special control techniques [8], [9], that can optimize a specific performance criteria while meeting stringent system constraints.

Studies on model-based control of rehabilitation robots are limited in the literature. Ding et. al. [10] used a musculoskeletal upper extremity model (without including muscle dynamics) to implement a model-based assistive controller for an upper extremity rehabilitation exoskeleton.

The objective of the current work is to predict muscle activities of a post-stroke patient during interaction with a rehabilitation robot, and provide quantitative therapy evaluations, thereby improving the quality of therapy in terms of safety and motor function improvement. To achieve this goal, we integrated a dynamic model of an end-effector-type robot with a 2D upper extremity musculoskeletal model to develop the human-robot interaction model, which can then be used with a proper controller to achieve this goal. For example, it can be used with gray-box controllers, in which the surface elecromyography (sEMG) signals can estimate the muscle activation levels of the patient, or be integrated with AI and any other type of controller that can predict muscle activations.

Recent progress in the development of the NMPC moti-

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

Borna Ghannadi, Naser Mehrabi and Reza Sharif Razavian are with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L3G1, Canada bghannad@uwaterloo.ca, nmehrabi@uwaterloo.ca, rsharifr@uwaterloo.ca

John McPhee is a professor in the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L3G1, Canada mcphee@uwaterloo.ca

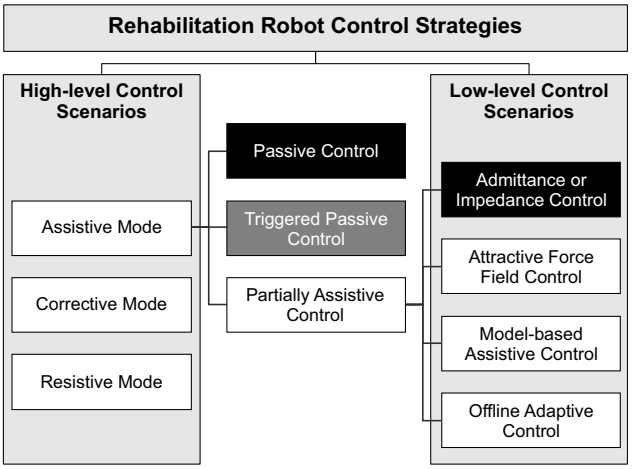

Fig. 1. Rehabilitation robot control strategies. Gray- and black-box control methods are colored in dark gray and black, respectively.

vates us to use the developed non-linear human-robot interaction model to control the rehabilitation robot. In our research, the human-robot interaction model is confined within an NMPC of the stroke rehabilitation robot. The proposed controller uses the musculoskeletal model of the upper extremity to predict the movements and muscle activations [11], thereby providing optimal assistance to the patient.

II. METHOD

A. Modeling

To develop a human-robot interaction model, both systems are modeled separately and then integrated into a single model. To model these systems, the MapleSim® software package is used for the following reasons:

1) Multi-domain capabilities: mechanical system support for modeling a parallelogram linkage, electrical system support to model DC motors, and biological system support for modeling the muscles and musculoskeletal system,
2) Symbolic processing,
3) Optimized code generation,
4) CAD support.

In the following subsections, each model will be presented individually. Then, their integration will be discussed.

1) Upper Extremity Rehabilitation Robot Model: Our rehabilitation robot is designed and developed by Quanser Inc. and the Toronto Rehabilitation Institute (TRI). It is a two degree of freedom (DOF) parallelogram linkage (see Fig. 2), that a post-stroke holds the end-efector (see Fig. 2) and the robot performs reaching movements in the horizontal plane for the rehabilitation of movement impairments in the upper extremity. This 2DOF robot has two DC motors and two optical encoders connected to the motors. The motors drive the parallelogram base joints using timing belts. The timing belts increase the stiffness of the driving joints. The robot has friction in its joints and on the surface supporting the end-effector. The frictions are modeled using

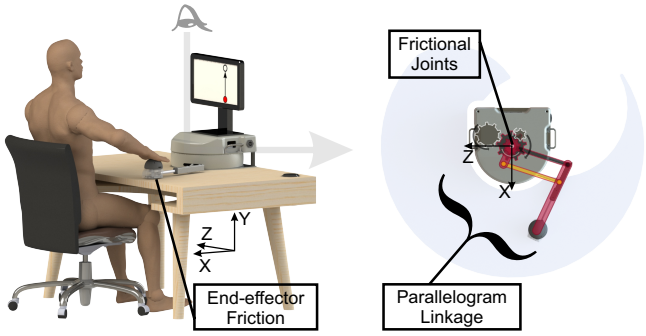

Fig. 2. Studied rehabilitation robot.

a continuous velocity-based frictional model [12]. The end-effector kinematics are:

$$\begin{cases} \mathbf{r}_e = \mathbf{\Phi}_R(\mathbf{q}) \\ \mathbf{v}_e = \frac{\partial \mathbf{\Phi}_R}{\partial \mathbf{q}}\dot{\mathbf{q}} = \mathbf{J}_R\dot{\mathbf{q}} \\ \mathbf{a}_e = \dot{\mathbf{J}}_R\dot{\mathbf{q}} + \mathbf{J}_R\ddot{\mathbf{q}} \end{cases} \tag{1}$$

where $\mathbf{r}_e$, $\mathbf{v}_e$ and $\mathbf{a}_e$ are end-effector position, velocity and acceleration in the global coordinates (ZX), respectively, and $\mathbf{J}_R$ is the robot geometric Jacobian. The dynamic model of the system is:

$$\begin{aligned} \boldsymbol{\tau}_R - {\mathbf{J}_R}^T\mathbf{F}_{int} &= \mathbf{M}_R(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{V}_R(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{f}_T \\ &\quad + \mathbf{K}_P(\mathbf{q}-\mathbf{q}_0) + {\mathbf{J}_R}^T\mathbf{f}_F \\ &= \mathbf{\Gamma}_R(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}) \end{aligned} \tag{2}$$

where $\boldsymbol{\tau}_R$ is the vector of robot motor torques, $\mathbf{F}_{int}$ and $\mathbf{f}_F$ are robot to human interaction force and friction force under the end-effector in the global coordinates, respectively. $\mathbf{M}_R$ is the robot inertia (mass) matrix, and $\mathbf{V}_R$ is the robot Coriolis-centrifugal torque vector. $\mathbf{f}_T$ is the friction torque vector at the joints. $\mathbf{K}_P$ is a 2×2 symmetric joint stiffness matrix, and $\mathbf{q}_0$ is the equilibrium position of the driving joint angles.

2) 2D Musculoskeletal Upper Extremity Model: We have used a musculoskeletal model to represent the human body. To build such a model, one should consider the tradeoff between the model complexity and computation costs. To be used within the NMPC scheme, the musculoskeletal upper extremity model should have a simple structure. Since the robot performs 2D movements, the musculoskeletal arm is modeled as a 2D double-pendulum operating in the horizontal plane. The model parameters are mapped from a high-fidelity 3D musculoskeletal model presented in [13], [14]. The mapping procedure has been discussed in [15]. The double-pendulum model is driven by 6 muscle groups, that are lumped from 29 muscles in the 3D musculoskeletal arm model. These muscles include shoulder and elbow mono/bi-articular flexors and extensors. The 3D arm model is placed in front of the robot at different heights (sitting positions); then the mapping is done on a plane that has maximal interference between the robot and 3D arm model workspaces (see Fig. 3). Muscle lengths are approximated by a 5th-order polynomial function of arm joint angles

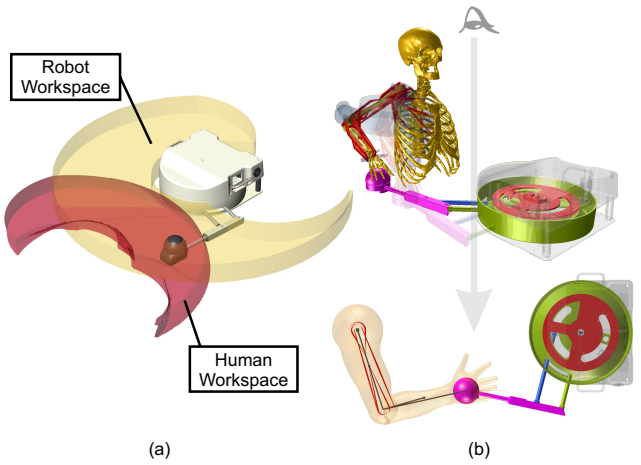

Fig. 3. (a) Rehabilitation robot workspace and 3D musculoskeletal model workspace evaluated at different chair heights and a single hand orientation with respect to the end-effector. (b) Mapping 3D musculoskeletal model to 2D model while interacting with the rehabilitation robot.

($\boldsymbol{\theta} = [\theta_1, \theta_2]^T$). Then, the virtual work principle is used to evaluate muscle moment arms [16]. In the arm model, the tendons are assumed to be rigid elements, since the tendon compliance is proportional to its slack length and tendon slack lengths in the upper extremity are not large [17]. Hence, only the contractile and passive elements of the Hill-type muscle model are used to simulate muscle contraction dynamics. Considering the dynamic properties of human body joints [18], damping coefficients are assigned to the shoulder and elbow joints of the 2D arm model. The hand kinematics are:

$$\begin{cases} \mathbf{r}_h = \boldsymbol{\Phi}_H(\boldsymbol{\theta}) \Rightarrow \boldsymbol{\theta} = \boldsymbol{\Psi}_H(\mathbf{r}_h) \\ \mathbf{v}_h = \frac{\partial \boldsymbol{\Phi}_H}{\partial \boldsymbol{\theta}} \dot{\boldsymbol{\theta}} = \mathbf{J}_H \dot{\boldsymbol{\theta}} \\ \mathbf{a}_h = \dot{\mathbf{J}}_H \dot{\boldsymbol{\theta}} + \mathbf{J}_H \ddot{\boldsymbol{\theta}} \end{cases} \tag{3}$$

where $\mathbf{r}_h$, $\mathbf{v}_h$ and $\mathbf{a}_h$ are hand position, velocity and acceleration in the global coordinates, respectively. $\mathbf{J}_H$ is the 2D arm geometric Jacobian, and $\boldsymbol{\Psi}_H$, which has a closed-form expression, is the inverse of vector function $\boldsymbol{\Phi}_H$. Finally, the dynamic equation of this 2DOF arm model is as follows:

$$\begin{aligned} \boldsymbol{\tau}_H + {\mathbf{J}_H}^T \mathbf{F}_{int} &= \mathbf{M}_H(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{V}_H(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{B}\dot{\boldsymbol{\theta}} \\ &= \boldsymbol{\Gamma}_H(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}}) \end{aligned} \tag{4}$$

where $\boldsymbol{\tau}_H$ is the vector of 2D arm joint torques, which are functions of muscle lengths ($l^M_{1..6}$), velocities ($v^M_{1..6}$), moment arms ($r^M_{1..6}$) and activations ($a^M_{1..6}$). $\mathbf{M}_H$ and $\mathbf{V}_H$ are the 2D arm inertia (mass) matrix and Coriolis-centrifugal torques vector, respectively. $\mathbf{B}$ is a diagonal damping matrix.

3) 2D Human-Robot Model: The developed robot and 2D musculoskeletal arm models are connected to each other by a revolute joint (along the Y-axis), which is equipped with a force sensor. This integrated model is a 2DOF closed-chain linkage with 8 inputs (2 robot motor torques, $\tau_{R1,2} \in [-10, 10]$ N.m, and 6 muscle activations, $a^M_{1..6} \in [0, 1]$), and 4 outputs (2 robot joint angles, $q_{1,2}$, and 2 force sensor outputs, $F_{ext\,Z,X}$). For this 2DOF mechanism, an optimized dynamic equation can be derived, if the number of generalized coordinates is reduced to 2 (the number of DOFs). In this subsection, we will provide a procedure to obtain an efficient number of equations for the system dynamics. In this closed-chain linkage, translational kinematics of the hand and end-effector should be the same (i.e. $\mathbf{r}_h = \mathbf{r}_e$, $\mathbf{v}_h = \mathbf{v}_e$, $\mathbf{v}_h = \mathbf{v}_e$), and from (1) and (3) we get:

$$\begin{cases} \boldsymbol{\theta} = \boldsymbol{\Psi}_H(\boldsymbol{\Phi}_R(\mathbf{q})) \\ \dot{\boldsymbol{\theta}} = {\mathbf{J}_H}^{-1} \mathbf{J}_R \dot{\mathbf{q}} \\ \ddot{\boldsymbol{\theta}} = {\mathbf{J}_H}^{-1} \left(\dot{\mathbf{J}}_R \dot{\mathbf{q}} + \mathbf{J}_R \ddot{\mathbf{q}} - \dot{\mathbf{J}}_H \dot{\boldsymbol{\theta}} \right) \end{cases} \tag{5}$$

Thus, the 2D arm kinematics can be written in terms of robot kinematics. Combining (2) and (4), the 2D human-robot dynamics will be:

$${\boldsymbol{\tau}_H} + {\mathbf{J}_H}^T {\mathbf{J}_R}^{-T} \left(\boldsymbol{\tau}_R - \boldsymbol{\Gamma}_R(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \right) - \boldsymbol{\Gamma}_H(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}}) = \mathbf{0} \tag{6}$$

The left-hand-side of (6) is a function ($\boldsymbol{\Pi}$) of control inputs ($\tau_{R1,2}$ and $a^M_{1..6}$), $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$, in which $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, and $\ddot{\boldsymbol{\theta}}$ are substituted from (5). The rearranged dynamic equation and its corresponding state-space equation is:

$$\ddot{\mathbf{q}} = \left(\frac{\partial \boldsymbol{\Pi}}{\partial \ddot{\mathbf{q}}} \right)^{-1} \left(\boldsymbol{\Pi} - \frac{\partial \boldsymbol{\Pi}}{\partial \ddot{\mathbf{q}}} \ddot{\mathbf{q}} \right) = \mathbf{M}^{-1}\mathbf{F} \tag{7}$$

$$\dot{\mathbf{x}}(t) = \begin{Bmatrix} \dot{\mathbf{q}} \\ \mathbf{M}^{-1}\mathbf{F} \end{Bmatrix} = \mathbf{f}\left(\mathbf{x}(t), \mathbf{u}(t)\right) \tag{8}$$

where $\mathbf{x} = [q_{1,2}, \dot{q}_{1,2}]^T \in \mathbb{R}^4$ and $\mathbf{u} = [\tau_{R1,2}, a^M_{1..6}]^T \in \mathbb{R}^6$ are the system state and control input vectors, respectively. $\mathbf{M} \in \mathbb{R}^{2\times 2}$ is the system mass matrix, and $\mathbf{F} \in \mathbb{R}^2$ is the right-hand-side force vector of the system dynamics.

B. Controller Structure

In the human body, the arm motion is controlled by the central nervous system using a combination of feed-forward and feedback control commands [11]. The feed-forward commands are estimated using an internal representation of the arm, and the feedback control is a set of corrective commands resulting from sensory organs in the arm. This control structure is analogous to the NMPC with receding horizon. The NMPC uses: 1) a forward dynamics (feed-forward) model of the system to predict optimal movements, and 2) feedback information for error correction. During a prediction horizon (t_{ph}), optimal movements are determined by minimizing a cost functional (J) subject to the system dynamics (i.e. (8)) and some constraints.

In our human-robot rehabilitation system, the goal is to optimally control the robot while estimating the human body muscular activities. Thus, the system can be controlled by an NMPC with feed-forward predictions for human body and robot optimal movements, and feedback commands for prediction improvement. To minimize the robot energy consumption, the human's physiological effort, and end-effector tracking error, our proposed cost functional is:

$$J = \int_{t_0}^{t_0+t_{ph}} \left(\mathbf{u}^T \mathbf{R} \mathbf{u} + \mathbf{e}^T \mathbf{Q} \mathbf{e} \right) dt \tag{9}$$

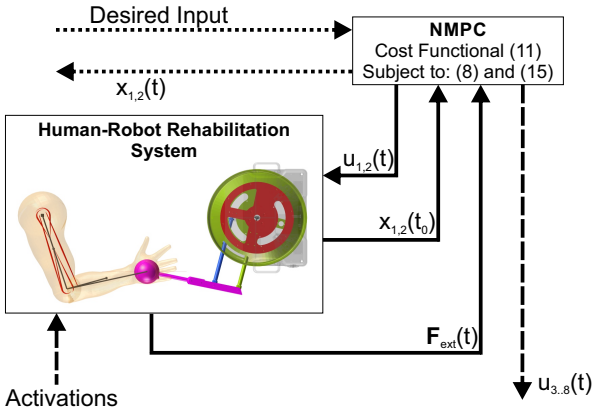

Fig. 4. Controller structure of the human-robot rehabilitation system for model-in-loop testing.

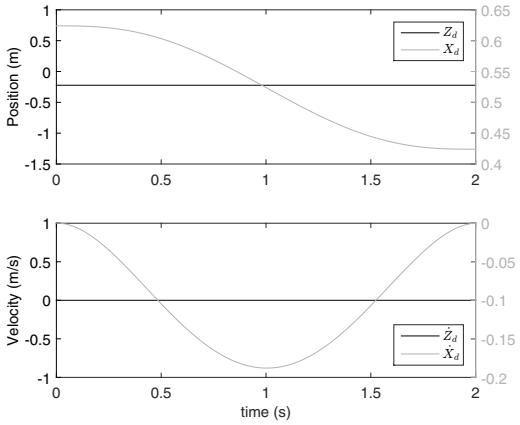

Fig. 5. Desired trajectory for point-to-point reaching movement.

where $\mathbf{R}$ and $\mathbf{Q}$ are positive-definite diagonal weighting matrices. These weights are chosen such that the objective functional results in the allowable error associated with the state or effort. Thus, the end-effector compliance can be controlled by adjusting the $\mathbf{Q}$. $\mathbf{e} = [e_1, e_2, e_3, e_4]^T$ is the state vector error in the global coordinates, i.e.:

$$\mathbf{e} = \begin{Bmatrix} \mathbf{\Phi}_R(\mathbf{q}) - \mathbf{r}_d \\ \mathbf{J}_R\dot{\mathbf{q}} - \dot{\mathbf{r}}_d \end{Bmatrix} \tag{10}$$

where $\mathbf{r}_d$ is the desired end-effecor (hand) trajectory in the global coordinates.

Considering the human-robot dynamics (8), interaction force is assumed to be an internal force. Nevertheless, if the human's muscle activation patterns are different from the human-robot model activations, the modeled interaction force ($\mathbf{F}_{int}$) will not be equal to the measured values by the force sensor ($\mathbf{F}_{ext}$). Consequently, an internal force constraint is introduced to the system dynamics:

$$\boldsymbol{\psi}\left(\mathbf{x}(t), \mathbf{u}(t), t\right) = \mathbf{F}_{int} - \mathbf{F}_{ext} = \mathbf{0} \tag{11}$$

where considering (4), the modeled interaction force can be written as:

$$\mathbf{F}_{int} = {\mathbf{J}_R}^{-T}\left(\boldsymbol{\tau}_R - \boldsymbol{\Gamma}_R\right) \tag{12}$$

Then, the human-robot constraint vector is:

$$\begin{Bmatrix} \mathbf{x}_{min} \\ \mathbf{u}_{min} \\ \mathbf{0} \end{Bmatrix} \leq \begin{Bmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \\ |\boldsymbol{\psi}\left(\mathbf{x}(t), \mathbf{u}(t), t\right)| \end{Bmatrix} \leq \begin{Bmatrix} \mathbf{x}_{max} \\ \mathbf{u}_{max} \\ \boldsymbol{\epsilon} \end{Bmatrix} \tag{13}$$

Since the dynamic properties of the human body may be different from the 2DOF model's dynamic properties, the internal force constraint (11) is relaxed to reduce the effect of unwanted dynamics due to the mentioned difference. Thus, in (13), a relaxation parameter ($\boldsymbol{\epsilon}$) is added to the internal force constraint.

The controller schematic is presented in Fig. 4. The human-robot rehabilitation system, modeled in MapleSim®, receives two sets of inputs: 1) the robot motors are driven by the NMPC output for joint torques, 2) muscle activations of the musculoskeletal model are either produced by a forward static optimization (FSO), in which a musculoskeletal system behavior is modeled through static minimization of physiological cost functional (see [19]), or measured sEMG data in real patient experiments. The NMPC controller optimizes the cost functional (9) subject to the system dynamics (8) and constraints (13). Current (at time $t = t_0$) robot joint angles and end-effector force are sent to the controller. The controller performance is evaluated by comparing the muscle activation inputs/outputs (dashed lines in Fig. 4) and position inputs/outputs (dotted lines in Fig. 4).

C. Simulations

In this study, the proposed NMPC problem is solved by the GPOPS-II optimal control package [20]. GPOPS-II uses orthogonal collocation, which is a direct optimization method. In this method, both state and control input variables are approximated by a series of polynomials and fed to a nonlinear programming (NLP) problem. An interior-point optimizer (IPOPT) is used to solve this NLP problem.

Two modes of simulations for the assessment of the proposed controller on the developed human-robot interaction model are performed. 1) Healthy human interaction: the controller weights are adjusted for the system shown in Fig. 4 by connecting the controller activation outputs to the activation inputs of the human-robot rehabilitation system (i.e. dashed lines are connected to each other). The adjustment is done by trial and error while checking the controller performance inside the desired criteria (less interaction force with good tracking). This mode simulates the robot interaction with a healthy human subject. 2) Dysfunctional patient interaction: the controller performance is tested by driving the human-robot rehabilitation system with zero muscular activations. In this mode, a totally dysfunctional patient is interacting with the robot. If the activations from the controller are at the same level as the input activations (i.e. zero activations), the controller will be successful in estimating the human's behavior while interacting with the robot. The desired trajectory for the simulations is selected based on the coordination of arm movements, in which the best performance in an unconstrained point-to-point reaching task is to generate smoothest motion, and this objective is

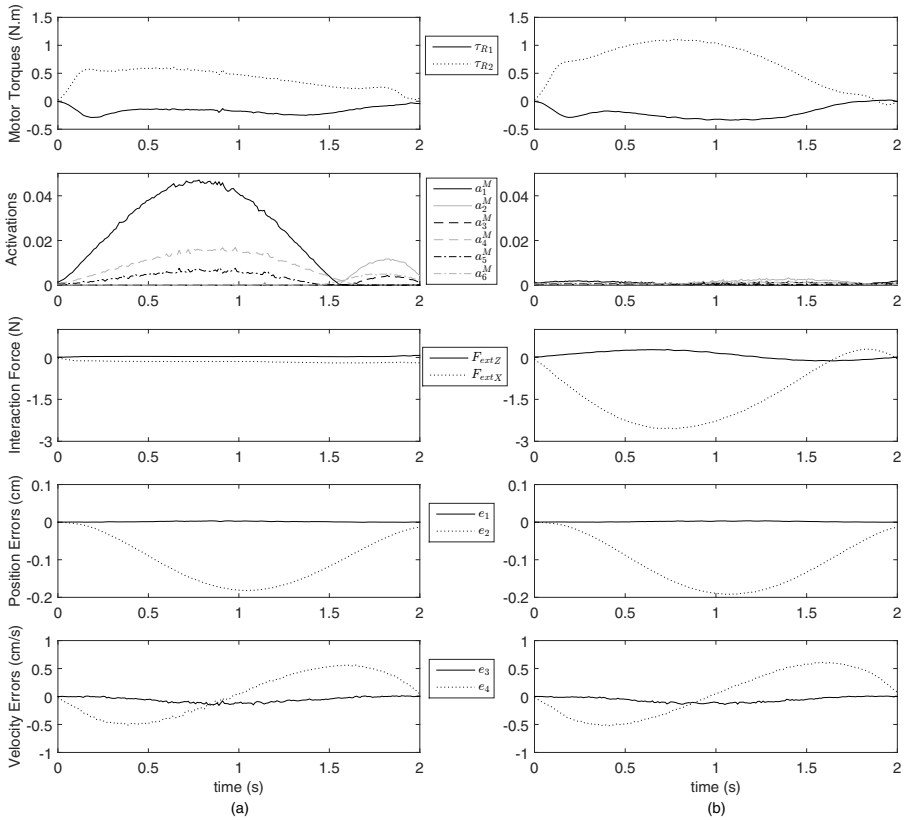

Fig. 6. Human-robot rehabilitation system simulation results using NMPC in two modes: (a) Activations of the human-robot system are provided by the NMPC output (mode-I), and (b) Activations of the human-robot system are zero to simulate a dysfunctional patient interaction with the robot (mode-II). Position and velocity errors are in the global coordinates (ZX).

determined by minimizing the square of jerk magnitude [21]. This trajectory ($\mathbf{r}_d = [Z(t), X(t)]^T$, see Fig. 5) is a straight line with a bell-shaped tangential speed profile inside the human-robot workspace and defined as:

$$\begin{cases} Z(t) = Z_0 \\ X(t) = X_0 + (15\delta - 6\delta^2 - 10)S\delta^3 \end{cases} \tag{14}$$

where S is the path length, and $\delta = t/t_f$. The path length and simulation time are set to $S = 20$ cm and $t_f = 2$ s, respectively. The simulation time step is 10 ms, and the prediction horizon for the NMPC is $t_{ph} = 100$ ms.

III. RESULTS AND DISCUSSION

Results of simulations are presented in Fig. 6. Magnitudes of motor torques for healthy human interaction mode (mode-I) are lower than the dysfunctional patient interaction mode (mode-II). In mode-I, the healthy subject tries to minimize his/her muscular activities while performing the task; hence, this reduces robot motor torques compared to mode-II. In the meantime, the robot tries to help the person while minimizing the robot energy consumption, the subject's muscular activities and the end-effector tracking error. in an ideal case, where there is no relaxation of the internal force constraint (see equation (11)), the interaction force should be zero. However, adding relaxing the constraint (see equation (13)) results in less amount of interaction force (see Fig. 6a).

In mode-II, the dysfunctional patient cannot apply any forces (has zero muscular activities). The NMPC with the previous objective tries to find the best motor torques, predict the dysfunctional subject's muscle activations while tracking the desired trajectory. Since the robot should overcome the patient's dynamics while reducing tracking error, the interaction force and motor torques increase compared to mode-I (see Fig. 6b).

In mode-II, the maximum amount of predicted muscle activations is about 0.003, which is very small and may be caused by integration errors, round-off calculations, and relaxation of the internal force constraint. This shows that the NMPC has predicted the dysfunctional patient's muscular activities with a reliable accuracy (the root mean square error is $9.78\text{e}{-4}$). In Fig. 6, the same amount of position and velocity errors for both modes indicates that the tracking term in the cost functional is dominant; thus, the NMPC tries to keep the subject's hand on the trajectory as much as possible. Position and velocity errors in the Z direction is less than the X direction since in the NMPC cost functional, the robot X direction is more compliant than the Z direction.

IV. CONCLUSION AND FUTURE WORK

In this study, we developed an efficient human-robot interaction model, which can be integrated with an NMPC, AI or any type of predictive controller for a human-robot rehabilitation system. The developed model has 2DOF, and

its dynamics has a minimal set of equations (8). This model is used in our proposed NMPC structure (see Fig. 4) to evaluate the controller performance in two modes: healthy subject, and dysfunctional patient interaction with the robot while performing a point-to-point reaching task in the smoothest possible movement path for the hand. The controller can successfully predict the muscular activations for the dysfunctional patient while providing the same therapy (tracking accuracy) to the patient.

In future, controller performance can also be assessed through running simulations with: 1) a spinal cord injury patient, whose muscle activation bounds are altered in a way that the maximum muscle activations in some muscles are less than 1 [22], or 2) a post-stroke patient, whose activation dynamics has changed [23]. These models can be generated through patient-specific FSO simulations.

Our proposed NMPC is implemented using GPOPS-II because of its variable-order adaptive collocation method for solving an optimal control problem. Although the controller performance is excellent in simulations, it is not possible to use it in our real-time experimental setup because: 1) GPOPS-II is not fast enough to run in real-time simulations, 2) the robot's data acquisition card (Q8) which is operable by Quanser's real-time control software driver (QUARC) does not support the GPOPS-II software. For this reason, in our future work, NMPC (i.e. Newton/GMRES method [24]) will be used to control the human-robot rehabilitation system in real-time.

V. ACKNOWLEDGMENTS

This work was funded by the Canada Research Chairs Program, and the Natural Sciences and Engineering Research Council of Canada (NSERC). The authors wish to thank Quanser Inc. for providing the upper limb rehabilitation robot, and TRI for collaborating.

REFERENCES

[1] S. Bansil, N. Prakash, J. Kaye, S. Wrigley, C. Manata, C. Stevens-Haas, and R. Kurlan, "Movement disorders after stroke in adults: a review.," *Tremor and other hyperkinetic movements (New York, N.Y.)*, vol. 2, pp. 1–7, jan 2012.

[2] J. Mehrholz, A. Hädrich, T. Platz, J. Kugler, and M. Pohl, "Electromechanical and robot-assisted arm training for improving generic activities of daily living, arm function, and arm muscle strength after stroke.," *The Cochrane Database of Systematic Reviews*, vol. 6, p. CD006876, jan 2012.

[3] C. L. Richards and F. Malouin, "Chapter 13 Stroke rehabilitation: clinical picture, assessment, and therapeutic challenge," in *Progress in Brain Research*, vol. 218, pp. 253–280, 2015.

[4] A. Turolla, M. Dam, L. Ventura, P. Tonin, M. Agostini, C. Zucconi, P. Kiper, A. Cagnin, and L. Piron, "Virtual reality for the rehabilitation of the upper limb motor function after stroke: a prospective controlled trial.," *Journal of neuroengineering and rehabilitation*, vol. 10, p. 85, jan 2013.

[5] P. Maciejasz, J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Troy, and S. Leonhardt, "A survey on robotic devices for upper limb rehabilitation.," *Journal of Neuroengineering and Rehabilitation*, vol. 11, p. 3, jan 2014.

[6] T. Proietti, V. Crocher, A. Roby-Brami, and N. Jarrasse, "Upper-Limb Robotic Exoskeletons for Neurorehabilitation: A Review on Control Strategies," *IEEE Reviews in Biomedical Engineering*, vol. 9, pp. 4–14, 2016.

[7] L. Marchal-Crespo and D. J. Reinkensmeyer, "Review of control strategies for robotic movement training after neurologic injury.," *Journal of Neuroengineering and Rehabilitation*, vol. 6, p. 20, jan 2009.

[8] F. Alambeigi, A. Zamani, G. Vossoughi, and M. R. Zakerzadeh, "Robust shape control of two SMA actuators attached to a flexible beam based on DK iteration," in *2012 12th International Conference on Control, Automation and Systems*, pp. 316–321, 2012.

[9] J. Sovizi, A. Alamdari, M. S. Narayanan, and V. Krovi, "Random Matrix Based Input Shaping Control of Uncertain Parallel Manipulators," in *Volume 7: 2nd Biennial International Conference on Dynamics for Design; 26th International Conference on Design Theory and Methodology*, p. V007T05A008, ASME, aug 2014.

[10] M. Ding, K. Hirasawa, Y. Kurita, H. Takemura, J. Takamatsu, H. Mizoguchi, and T. Ogasawara, "Pinpointed muscle force control in consideration of human motion and external force," in *2010 IEEE International Conference on Robotics and Biomimetics, ROBIO 2010*, pp. 739–744, IEEE, dec 2010.

[11] N. Mehrabi, R. Sharif Razavian, B. Ghannadi, and J. McPhee, "Predictive Simulation of Reaching Moving Targets Using Nonlinear Model Predictive Control," *Frontiers in Computational Neuroscience*, vol. 10, p. 143, jan 2017.

[12] P. Brown and J. McPhee, "A Continuous Velocity-Based Friction Model for Dynamics and Control With Physically Meaningful Parameters," *Journal of Computational and Nonlinear Dynamics*, vol. 11, p. 054502, jun 2016.

[13] B. a. Garner and M. G. Pandy, "A Kinematic Model of the Upper Limb Based on the Visible Human Project (VHP) Image Dataset.," *Computer methods in biomechanics and biomedical engineering*, vol. 2, pp. 107–124, jan 1999.

[14] B. a. Garner and M. G. Pandy, "Musculoskeletal model of the upper limb based on the visible human male dataset.," *Computer methods in biomechanics and biomedical engineering*, vol. 4, pp. 93–126, feb 2001.

[15] B. Ghannadi, N. Mehrabi, and J. McPhee, "Development of a Human-Robot Dynamic Model to Support Model-Based Control Design of an Upper Limb Rehabilitation Robot," in *ECCOMAS Thematic Conference on Multibody Dynamics*, (Barcelona, Spain), 2015.

[16] E. K. Chadwick, D. Blana, R. F. Kirsch, and A. J. van den Bogert, "Real-time simulation of three-dimensional shoulder girdle and arm dynamics.," *IEEE transactions on bio-medical engineering*, vol. 61, pp. 1947–56, jul 2014.

[17] F. E. Zajac, "Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control.," *Critical Reviews in Biomedical Engineering*, vol. 17, pp. 359–411, jan 1989.

[18] M. K. Lebiedowska, "Dynamic properties of human limb segments," in *International Encyclopedia of Ergonomics and Human Factors* (W. Karwowski, ed.), p. 317, CRC Press, 2nd ed., 2006.

[19] N. Mehrabi, R. Sharif Razavian, and J. McPhee, "A physics-based musculoskeletal driver model to study steering tasks," *Journal of Computational and Nonlinear Dynamics*, vol. 10, p. 021012, apr 2014.

[20] M. A. Patterson and A. V. Rao, "GPOPS-II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using hp-Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming," *ACM Transactions on Mathematical Software*, vol. 41, pp. 1–37, oct 2014.

[21] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model.," *The Journal of neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.

[22] D. García-Vallejo, J. M. Font-Llagunes, and W. Schiehlen, "Dynamical analysis and design of active orthoses for spinal cord injured subjects by aesthetic and energetic optimization," *Nonlinear Dynamics*, vol. 84, pp. 559–581, apr 2016.

[23] S. J. Sober, J. M. Stark, D. S. Yamasaki, and W. W. Lytton, "Receptive Field Changes After Strokelike Cortical Ablation : A Role for Activation Dynamics," *Journal of Neurophysiology*, vol. 78, no. 6, pp. 3438–3443, 1997.

[24] N. Mehrabi, S. Tajeddin, N. L. Azad, and J. McPhee, "Application of Newton/GMRES Method to Nonlinear Model Predictive Control of Functional Electrical Stimulation," in *The 3rd International Conference on Control, Dynamic Systems, and Robotics (CDSR'16)*, (Otawa, Ontario, Canada), may 2016.