A Predictive Model of Human Movements based on Model Predictive Control for Human-Robot Interaction

Aeden G. Gillam[1] and Reza Sharif Razavian[2], Member, IEEE

***Abstract*— Predicting human movements is vital to safely control robots that physically interact with humans. However, predictive neuromuscular models that are fast enough for real-time control applications have proven challenging, due to the complexity of the neural and musculoskeletal systems. Nonlinear optimization-based prediction of movements in a musculoskeletal model is prohibitively slow. On the other hand, highly simplified models based on linear control theory cannot handle complexities of the human musculoskeletal system. Model Predictive Control (MPC) can potentially fill the gap between these two modeling extremes, by taking into account physiological nonlinearities, constraints, and redundancies while keeping computations fast through its receding horizon formulation. This study presents a new predictive model for the human movements based on MPC, which can control activity of four muscles acting on an inertia in a two-dimensional space to generate movements. The MPC results are compared to that of the prominent human motor control model in the neuroscience literature, which is based on linear quadratic regulator. The predicted movements are similar between the two controllers and are qualitatively similar to human behavior. MPC achieves these results while satisfying physiological constraints on muscle activities and ranges of motion – features that are not present in the existing models. These results demonstrate promise and potential for MPC controllers to accurately predict human neuro-muscular activities for the next generation controllers for human-robot interaction.**

I. INTRODUCTION

Human-robot interactions are becoming increasingly popular in industrial and medical settings [1]. With advancements in robotic hardware, there is special interest in using such devices to aid in neuromuscular rehabilitation [1,2]. However, for a robot to physically interact with a human in a safe and intuitive manner, it is necessary to construct mathematical representations for the human neuro-muscular control to enable the robot to predict the user's behavior and facilitate natural user interactions.

One of the biggest challenges in this area is that only a few accurate models for the neuromuscular control of human movements exist. One major hurdle in building these models is the complexity, high-dimensionality, and nonlinearity of the human neural musculoskeletal systems. While there has been promising development in large-scale nonlinear optimization-based musculoskeletal models to predict movement and muscle activities [3-6], these models are too slow to be useful in real-time control of robots. These models tend to take 30 min to 1 $hour$ or longer to run [3].

Unlike biomechanists who focus on detailed musculoskeletal models to predict movements, neuroscientists often dedicate their attention to isolating the fundamental principles of movement control, which has led to significant discoveries. A prominent model of human motor control is the application of linear optimal control to a highly simplified model of human neuro-muscular systems [7,8]. In these models, which is based on a modified Linear Quadratic Gaussian (LQG) controller, patterns of human movements, e.g., that of the hand, is well predicted in a variety of settings [9-15], including hand motion along a flat 2D surface [10] and virtual cursor control via 1D hand dynamics [12]. However, this implementation of the LQG controller for predicting human behavior is only applicable to linear models of the human musculoskeletal systems—an assumption that is far from reality. The neural and musculoskeletal systems are highly nonlinear and are subject to major physiological constraints. Existing LQG-based models cannot handle nonlinearities nor constraints, which play a major role in shaping human behavior.

To overcome these limitations, this study introduces the implementation of a Model Predictive Controller (MPC) to model neuromuscular control. Unlike the LQG, the MPC controller has the ability to handle nonlinear system dynamics and can implement constraints onto the state and input variables. Examples of constraints present in the musculoskeletal system include the maximum and minimum force that a muscle can produce, physiological ranges of joint positions, and maximum velocities—parameters that could not be included in the linear optimal control models.

With an MPC controller, optimal input variables are determined by the minimization of a cost function over a prediction horizon while satisfying the constraints on control and state variables. Through these horizons, the predicted trajectory of the system is used to determine the best input values that will achieve the desired outcomes. This predictive methodology is analogous to preplanning a course of action prior to executing that behavior, such as planning the path taken to grab an object [16].

This study develops a human motor control model based on an MPC algorithm. While MPC can handle nonlinear systems, a linear model is implemented due to its relative simplicity and its ability to be compared with current LQG models [10,17]. Both the MPC and LQG models simulate motion of a point mass in two-dimensional space actuated by linear, independent muscle models. A two-dimensional

*Research supported by Northern Arizona University.

[1]A. G. Gillam is with the Northern Arizona University Department of Mechanical Engineering, Flagstaff, AZ 86011 USA (e-mail: agg256@nau.edu).

[2]R. Sharif Razavian, is with the Northern Arizona University Department of Mechanical Engineering, Flagstaff, AZ 86011 USA (e-mail: Reza.Sharif-Razavian@nau.edu).

space represents the task space in point-to-point reaching movements [18]. The LQG model does not include any constraints, while the MPC model takes into account the bounds on muscle activities and ranges of motion, more indicative to physiological muscles.

II. METHODS

A. Physical Model

The plant model under consideration for the MPC-based and LQG-based human motor control simulation is depicted in Fig. 1(a) and 1(b), respectively. The MPC model is a simplified version of two pairs of antagonistic muscles working together to move a point mass in both the vertical (y) and horizontal (x) directions. One end of each muscle is attached to a fixed support for while the other end is directly attached to the mass. This model is restricted to two-dimensional motion.

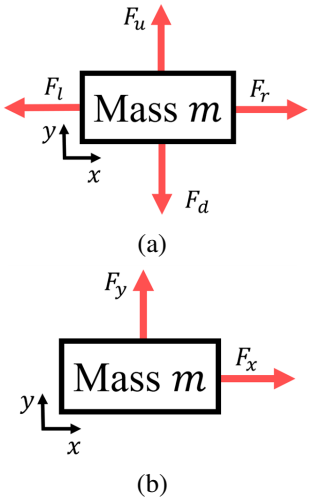

Fig. 1. The 2D plant model for motor control design. (a) The 4-muscle plant model with constraints for the MPC model. (b) The unconstrained 2-muscle plant model for the LQG model.

It is assumed that all muscles are only able to produce a pull force on the mass, consistent with physiological muscles. Further, each muscle has a maximum force that it can produce on the mass. Gravitational effects and any other external forces that may be applied to the system are neglected. The interaction between tendon and muscles are also neglected.

The dynamics of this system can be expressed through a summation of forces in the horizontal and vertical direction. These dynamics can be expressed as:

$$\ddot{x} = \frac{F_r}{m} - \frac{F_l}{m} \tag{1}$$

$$\ddot{y} = \frac{F_u}{m} - \frac{F_d}{m} \tag{2}$$

Here, x and y are the positions along the horizontal and vertical axes respectively, m is the mass of the arm, and F are the muscle forces in the left, right, up, and down directions. All muscle forces can be expressed as a proportion of the maximum force that each muscle could exert. Let muscle activation, a, with appropriate subscripts represent the ratio of the muscle force F to its maximum force capacity F_{max}:

$$F_r = a_r F_{r,max} \tag{3}$$

$$F_l = a_l F_{l,max} \tag{4}$$

$$F_u = a_u F_{u,max} \tag{5}$$

$$F_d = a_d F_{d,max} \tag{6}$$

The activation of muscle force is not instantaneous. It takes time for the muscle force to change. For simplicity, a time constant τ is utilized to represent the activation dynamics as a first-order filter:

$$\dot{a}_r = \frac{1}{\tau}u_r - \frac{1}{\tau}a_r \tag{7}$$

$$\dot{a}_l = \frac{1}{\tau}u_l - \frac{1}{\tau}a_l \tag{8}$$

$$\dot{a}_u = \frac{1}{\tau}u_u - \frac{1}{\tau}a_u \tag{9}$$

$$\dot{a}_d = \frac{1}{\tau}u_d - \frac{1}{\tau}a_d \tag{10}$$

Here, u represents the neural excitation to the muscle, and is the input to the plant model that is directly controlled by the MPC (see section II.B).

This model is converted to a standard state-space form for control purposes:

$$\dot{\mathbf{X}}(\mathbf{t}) = \mathbf{A}\mathbf{X}(\mathbf{t}) + \mathbf{B}\mathbf{u}(\mathbf{t}) \tag{11}$$

with

$$\mathbf{X}(\mathbf{t}) = \begin{bmatrix} x(t) & \dot{x}(t) & a_r(t) & a_l(t) & y(t) & \dot{y}(t) & a_u(t) & a_d(t) \end{bmatrix}^T \tag{12}$$

$$\mathbf{u}(\mathbf{t}) = \begin{bmatrix} u_r(t) & u_l(t) & u_u(t) & u_d(t) \end{bmatrix}^T \tag{13}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{F_{r,max}}{m} & -\frac{F_{l,max}}{m} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{\tau} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{F_{u,max}}{m} & -\frac{F_{d,max}}{m} \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} \end{bmatrix} \tag{14}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & \frac{1}{\tau} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\tau} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\tau} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\tau} \end{bmatrix}^T \tag{15}$$

It is further assumed that the plant is fully observed, accomplished through vision (positions and velocities) and proprioception (positions, velocities, and muscle forces).

Therefore, the outputs $\mathbf{Y(t)}$ are equivalent to the state variables:

$$\mathbf{Y(t)} = \mathbf{X(t)} \tag{16}$$

Finally, this continuous-time state-space is transformed into discrete-time with a time step of $0.02\ s$.

B. The MPC Formulation

The MPC Controller is based on the formulations given by Maciejowski [19]. The goal of the MPC controller is to determine the optimal values of control increments, $\Delta\hat{\mathbf{u}}$ that minimizes the quadratic cost function:

$$V(k) = \sum_{i=1}^{H_p} \left(\hat{\mathbf{X}}\left(k+i|k\right) - \mathbf{r}\left(k+i\right)\right)^T \mathbf{Q} \left(\hat{\mathbf{X}}\left(k+i|k\right) - \mathbf{r}\left(k+i\right)\right) + \sum_{i=0}^{H_u-1} \left(\Delta\hat{\mathbf{u}}\left(k+i|k\right)\right)^T \mathbf{R}\left(\Delta\hat{\mathbf{u}}\left(k+i|k\right)\right) \tag{17}$$

where H_p represents the length of the prediction horizon (in number of time steps), H_u is the length of the control horizon, $\hat{\mathbf{X}}\left(k+i|k\right)$ is the predicted state variable at time $(k+i)$ given the observation of the outputs at time k, $\mathbf{r}\left(k+i\right)$ is the reference trajectory (i.e. the desired setpoint) at time $k+i$, $\Delta\hat{\mathbf{u}}\left(k+i|k\right)$ are the unknown change in input values at time $k+i$ given the current states at time k, and $\mathbf{Q}$ and $\mathbf{R}$ are weight matrices associated with the state error and input terms, respectively. The $\mathbf{Q}$ and $\mathbf{R}$ matrices are diagonal matrices with elements on the diagonal. $\mathbf{Q}$ is defined as $3 \times 10^{-5+i/20}$ for positions, $3 \times 10^{-7+i/20}$ for velocities, and 3×10^{-2} for muscle activations, where i increases by 1 at each time step across the prediction horizon. $\mathbf{R}$ is defined as 1 across its diagonal. Through mathematical manipulation [8], this cost function is transformed into the cost function of (18), which is only in terms of controls and can be solved using Quadratic Programming.

$$V(k) = \frac{1}{2}\Delta\mathbf{U}(k)^T(2\mathbf{H})\Delta\mathbf{U}(\mathbf{k}) + \mathbf{G}^T\Delta\mathbf{U}(\mathbf{k}) \tag{18}$$

Here $\Delta\mathbf{U}\left(k\right)$ is defined as all predicted $\Delta\hat{u}$ values across the control horizon H_u, concatenated vertically. $\mathbf{H}$ and $\mathbf{G}$ can be defined as follows:

$$\mathbf{H} = \Theta^T\mathcal{Q} + \mathcal{R} \tag{19}$$

$$\mathbf{G} = -2\Theta^T\mathcal{Q}\mathcal{E}(k) \tag{20}$$

where $\mathcal{E}\left(k\right)$ represents the error associated with each of the state output values at time k relative to the setpoint r [19]. Θ is defined by the coefficients of the unknown control terms $\Delta\hat{\mathbf{u}}$, defined in (21):

$$\Theta = \begin{bmatrix} B_d & A_dB_d + B_d & \cdots & \sum_{i=0}^{H_u-1} A_d^iB_d & \sum_{i=0}^{H_u} A_d^iB_d & \cdots & \sum_{i=0}^{H_p-1} A_d^iB_d \\ \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & B_d & A_dB_d + B_d & \cdots & \sum_{i=0}^{H_p-H_u} A_d^iB_d \end{bmatrix}^T \tag{21}$$

Here A_d and B_d are discretized versions of the A and B matrices presented in (14) and (15), respectively.

$\mathcal{Q}$ and $\mathcal{R}$ are concatenated versions of the $\mathbf{Q}$ and $\mathbf{R}$ weighted matrices associated with the error of the outputs and inputs and are defined in (22) and (23), respectively.

$$\mathcal{Q} = \begin{bmatrix} \mathbf{Q}\left(\mathbf{k}\right) & 0 & \cdots & 0 \\ 0 & \mathbf{Q}\left(\mathbf{k+1}\right) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{Q}\left(\mathbf{H_p}\right) \end{bmatrix} \tag{22}$$

$$\mathcal{R} = \begin{bmatrix} \mathbf{R}\left(\mathbf{k-1}\right) & 0 & \cdots & 0 \\ 0 & \mathbf{R}\left(\mathbf{k}\right) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{R}\left(\mathbf{H_u-1}\right) \end{bmatrix} \tag{23}$$

The advantage of the MPC model over the existing LQG counterpart (see section II.C) is its ability to satisfy constraints on the state and input variables. The muscle activations, a, must be between 0 (not activated) and 1 (fully activated) to be consistent with the physiological muscle. Since the dynamics between control input u and activation a is first-order, this constraint is implemented onto the input $\Delta\mathbf{u}$.

Constraints are enforced in the MPC controller to prevent muscle activations to exceed 1 or fall below 0, which take the form of:

$$\Omega\Delta\mathbf{U} \leq \omega \tag{24}$$

where Ω and ω are defined as:

$$\Omega = \mathcal{F} \tag{25}$$

$$\omega = -\mathcal{F}_1\mathbf{u}\left(k-1\right) - f \tag{26}$$

where $\mathbf{u}\left(k-1\right)$ is the previous time step inputs and f is a 8×1 array of constraints on the inputs, defined as:

$$\begin{aligned} f &= \begin{bmatrix} u_{r,1} & -u_{r,2} & u_{l,1} & -u_{l,2} & u_{u,1} & -u_{u,2} & u_{d,1} & -u_{d,2} \end{bmatrix}^T \\ &= \begin{bmatrix} 0 & -1 & 0 & -1 & 0 & -1 & 0 & -1 \end{bmatrix}^T \end{aligned} \tag{27}$$

Here, $u_{r,1}$, $u_{l,1}$, $u_{u,1}$, and $u_{d,1}$ represent lower bounds constraints and $u_{r,2}$, $u_{l,2}$, $u_{u,2}$, and $u_{d,2}$ indicate upper bound constraints. $\mathcal{F}$ is a concatenated version of the coefficients (termed F_j) of $\Delta\hat{\mathbf{u}}\left(k+i|k\right)$ in the constraint equations across the control horizon H_u, as defined in (28):

$$\mathcal{F}_i = \sum_{j=i}^{H_u} F_j \tag{28}$$

where i represents the time step index within a control horizon. $\mathcal{F}_1$ can be calculated by substituting $i = 1$ into (24). Values of F_j are 1 for an upper-bound inequality constraint and -1 for a lower-bound inequality constraint. For details the reader is referred to [19]. The constrained MPC controller is a quadratic program, which is solved using MATLAB's built-in 'quadprog' function.

C. The LQG Formulation

The LQG-based motor control model is developed following the formulation of [10,17]. The plant model for this LQG controller is shown in Fig. 1(b) This plant model is similar to the one used by the MPC controller; however, since the LQG cannot consider the pull-only constraint on the muscle forces, the pairs of antagonist muscles are replaced with just one muscle that can push and pull. The rest of the plant model, including the muscle dynamics and state-space formulations are the same as the one used with the MPC. The LQG seeks to minimize the quadratic cost function:

$$V = \sum_{i=1}^{N} \left(\hat{\mathbf{X}}(i)^T Q_i \hat{\mathbf{X}}(i) + \mathbf{u}(i)^T R_i \mathbf{u}(i) \right) \tag{29}$$

where N is the total number of time-steps in the simulation, and $\mathbf{Q}$ and $\mathbf{R}$ are the weight terms for the states and controls, respectively. In this model, the $\mathbf{Q}$ and $\mathbf{R}$ matrices are constant diagonal matrices; $\mathbf{R}$ has its diagonal elements equal to 1. The $\mathbf{Q}$ matrix has penalty terms for the positions 3×10^{-3} and the velocities 3×10^{-5} along the diagonal, and zeros otherwise. The $\mathbf{Q}$ and $\mathbf{R}$ matrices are tuned such that the predicted movement becomes closer to human behavior.

This LQG controller has an optimal Kalman filter gains for state estimation, and an optimal control gain for control. For details the reader is referred to [17,18].

III. RESULTS

All simulations are performed in MATLAB 2024a with numerical values of the parameters as shown in TABLE I.

TABLE I
PARAMETER NUMERICAL VALUES USED IN SIMULATIONS

Parameter	Numeric Value	Reference
System Inertia m	$2\ kg$	[20]
Maximum Muscle Force F_{max}	$1,000\ N$	[21]
Muscle Dynamics Time Constant τ	$60\ ms$	[11]
Discretization Time Step Δt	$20\ ms$	[22]
Number of Time Steps N	76	

For both the MPC and LQG simulations, the point masses start at either the (x, y) coordinate of $(-0.45, -0.20)\ m$ or $(0.45, -0.20)\ m$ from rest, and must reach the location of $(0, 0)\ m$ with zero velocity after roughly $1\ s$. These distances were utilized in a previous experiment [23]. All muscle activations initially start at 0. The MPC controller uses a prediction and control horizon of 76 time-steps. The results of the first and second set of distance simulations are shown in Figs. 2 and 3, respectively. The experimentally observed velocity profiles [23] are also compared with the simulated ones in Figs. 2(c) and 3(c).

As shown in both distance simulations, the MPC and LQG controllers determined that the optimal path from the initial to final position was a straight connection between the two locations in approximately $1\ s$, consistent with experimental data [23]. Both controllers overshot the set point position at

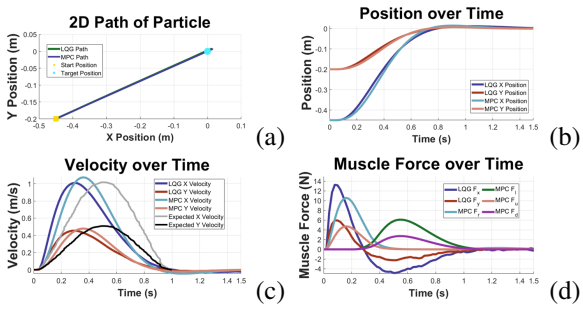

Fig. 2. LQG and MPC Simulation of point mass travelling from $(-0.45, -0.20)\ m$ to $(0, 0)\ m$. (a) 2D path of point mass. (b) Position curves of LQG and MPC controllers. (c) Velocity curves of LQG, MPC, and experimental data from [23]. (d) Muscle forces predicted by LQG and MPC.

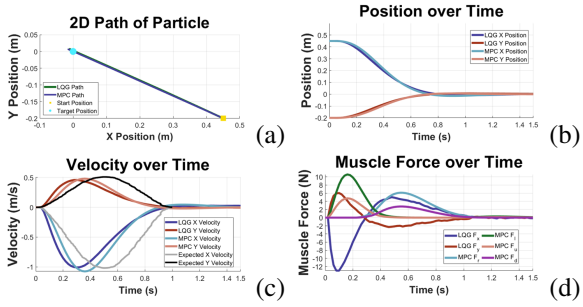

Fig. 3. LQG and MPC Simulation of point mass travelling from $(0.45, -0.20)\ m$ to $(0, 0)\ m$. (a) 2D path of point mass. (b) Position curves of LQG and MPC controllers. (c) Velocity curves of LQG, MPC, and experimental data from [23]. (d) Muscle forces predicted by LQG and MPC.

approximately $0.75\ s$, then steadily approach the targeted position. The masses' positions overlapped considerably between the LQG and MPC controllers from both initial start positions, demonstrating the similarity between the two controllers.

The velocity and muscle force profiles between the LQG and MPC controllers for both initial positions are similar, with small discrepancies. These parameters for the first and second simulation are tabulated in TABLE II and III, respectively. In all simulations, the maximum velocity values of the MPC simulation tended to be slightly larger in magnitude and occur slightly later than the LQG controller. These velocity profiles are also similar to the experimental data [18,23], though a slight forward skew in velocity profiles is present in the LQG and MPC simulations. A change in weights in the $\mathbf{Q}$ and $\mathbf{R}$ matrices have the potential to change the slew in mass velocity profiles.

An important contrast between the LQG and MPC models are the muscle activation forces. Unlike the LQG case, the MPC model applies an active minimum-force constraint of $0\ N$, enabling the system to behave similar to physiological human muscles. In Fig. 2(d), the constraints are shown to be active on the down and left muscles when the up and right muscles are active. The down and left muscles in the MPC simulation become active when the LQG vertical and horizontal muscle forces both become zero at $0.28\ s$. This

TABLE II

SUMMARY OF LQG AND MPC SIMULATION RESULTS; START POSITION $(-0.45, -0.20)\ m$

	LQG		MPC	
Parameter	**Time**	**Value**	**Time**	**Value**
$V_{x,max}$[a]	$0.30\ s$	$1.02\ m/s$	$0.36\ s$	$1.07\ m/s$
$V_{y,max}$[b]	$0.28\ s$	$0.45\ m/s$	$0.36\ s$	$0.48\ m/s$
$F_{r,max}$[c]	$0.10\ s$	$13.12\ N$	$0.16\ s$	$10.59\ N$
$F_{l,max}$[d]	$0.54\ s$	$-4.83\ N$	$0.54\ s$	$6.10\ N$
$F_{u,max}$[e]	$0.08\ s$	$5.96\ N$	$0.16\ s$	$4.74\ N$
$F_{d,max}$[f]	$0.58\ s$	$-2.26\ N$	$0.54\ s$	$2.73\ N$

[a]Maximum horizontal velocity
[b]Maximum vertical velocity
[c]MPC - Maximum right force; LQG - Maximum positive horizontal force
[d]MPC - Maximum left force; LQG - Maximum negative horizontal force
[e]MPC - Maximum up force; LQG - Maximum positive vertical force
[f]MPC - Maximum down force; LQG - Maximum negative vertical force

TABLE III

SUMMARY OF LQG AND MPC SIMULATION RESULTS; START POSITION $(0.45, -0.20)\ m$

	LQG		MPC	
Parameter	**Time**	**Value**	**Time**	**Value**
$V_{x,max}$	$0.28\ s$	$-1.00\ m/s$	$0.36\ s$	$-1.07\ m/s$
$V_{y,max}$	$0.30\ s$	$0.47\ m/s$	$0.36\ s$	$0.48\ m/s$
$F_{r,max}$	$0.48\ s$	$4.76\ N$	$0.54\ s$	$6.10\ N$
$F_{l,max}$	$0.10\ s$	$-12.99\ N$	$0.16\ s$	$10.59\ N$
$F_{u,max}$	$0.10\ s$	$5.83\ N$	$0.16\ s$	$4.74\ N$
$F_{d,max}$	$0.52\ s$	$-2.07\ N$	$0.54\ s$	$2.73\ N$

intersections signifies the correlation of muscle activations between the MPC and LQG controllers, with the added benefit of non-negative muscle forces produced by the MPC. Between $0.28\ s$ and $0.50\ s$, the up and right muscles deactivate while the down and left muscles activate for the MPC controller. This region signifies the time-dependency of antagonist muscles working in opposition of one another, a feature that the LQG controller omits. From $0.28\ s$ to $1.1\ s$ the LQG and MPC muscle forces correlate, with activity in the MPC left and down muscles and negative activity in the LQG horizontal and vertical muscle forces.

The muscle force peaks are also similar between MPC and LQG (Figs. 2(d), 3(d) and TABLES II and III). Generally, the LQG simulation will reach peak muscle forces a few cs sooner than the MPC simulation. For the first distance simulations, LQG and MPC had peak right and upper muscle activations between $0.08\ s$ and $0.16\ s$, and had peak left and down muscle activations between $0.48\ s$ and $0.58\ s$. The same peak muscle activation times occurred for the second distance simulation. LQG muscle forces tended to be greater in magnitude than the MPC case for the first muscle force peak. However, MPC muscle forces tended to be greater in magnitude than the LQG case for the second muscle force peak. These differences in muscle force peaks may be indicative of the agonist-antagonist interaction that is at play for the MPC simulation. The LQG case has only one muscle to actuate the system while the MPC has two muscles that can interact with one another to reduce the initial peak force required to move to the targeted location. Larger second muscle force peaks in the MPC case may compensate for the lower muscle activity during the first phase of the motion, reducing the overall largest peak muscle forces of the system.

IV. DISCUSSION

This study aimed to simulate neuromuscular control of a point mass actuated by independent muscle forces in a 2D space. This simulation was conducted via an LQG controller, the current model for neural motor control, and the proposed MPC controller, an advanced controller capable of handling anatomically correct constraints that the LQG controller fails to include. Due to the lack of constraints, the LQG-controlled model had two muscles actuating the mass compared to four utilized by the MPC-controlled model.

The MPC approach for the described system resembles much similarity to the LQG model and experimental data. Mass position and velocities express a smooth transition from the start and end positions, correlating to the experimental results of [22,23]. Peak velocities occur slightly sooner than halfway through the task, with both MPC and LQG overshooting the targeted position, then cautiously readjusting. Incorporating multiple muscles in a direction produce similar results, with discrepancies in peak muscle activations and timing. With the linear plant model, MPC correlates strongly to the results produced by the LQG, the leading neuroscientific model for motor control [7], while expanding upon the neuromuscular control framework by adding complexity in multi-muscle relationships and their restrictions to produce more realistic human neuromuscular activity. The simulation time of the MPC controller is approximately $4.2\ s$. This time may be too slow in some human-robotic interactions where actions must be produced in milliseconds [22]. Suggestions to decrease the computational time is to introduce explicit MPC [24] to prevent large-scale optimizations at each time step, especially when a large, nonlinear system is introduced. These results are significant in the framework of human-robotic interactions by verifying the MPC controller as a method to obtain kinematic and muscle activation parameters comparative to experimentally-gained values. Robotic systems can use these results and future applications of the MPC controller to understand how a human will biomechanically behave under certain pretexts. The MPC model opens new possibilities in accurately predicting human neuromuscular control that can establish the framework for next-generation human-interactive robots.

V. CONCLUSION

The MPC formulation of predicting human neuromuscular control enhances the realism of human motion models, providing the tools necessary for the much-needed improvement for human-robotic interactions. When utilized with a heavily simplified linear plant model, the MPC predicted similarly to that of the experimentally verified LQG approach, with the

added benefit of incorporating agonist muscles with active constraints. Further improvements to the MPC formulation and the model design to handle complex, nonlinear dynamics will prove useful in providing the models necessary for robots to appropriately respond to human behavior in more advanced settings.

REFERENCES

[1] B. Ghannadi, R. Sharif Razavian, and J. McPhee, "Upper extremity rehabilitation robots: a survey," in Handbook of Biomechatronics, J. Segil, Ed., Elsevier, 2019, pp. 319–353. doi: 10.1016/B978-0-12-812539-7.00012-X.

[2] M. Shahbazi, S. F. Atashzar, and R. V. Patel, "A Systematic Review of Multilateral Teleoperation Systems," IEEE Transactions on Haptics, vol. 11, no. 3, pp. 338–356, 2018, doi: 10.1109/TOH.2018.2818134.

[3] A. J. Meyer, I. Eskinazi, J. N. Jackson, A. V. Rao, C. Patten, and B. J. Fregly, "Muscle Synergies Facilitate Computational Prediction of Subject-Specific Walking Motions," Frontiers in Bioengineering and Biotechnology, vol. 4, p. 77, Oct. 2016, doi: 10.3389/fbioe.2016.00077.

[4] M. Sharif Shourijeh and J. McPhee, "Optimal Control and Forward Dynamics of Human Periodic Motions Using Fourier Series for Muscle Excitation Patterns," Journal of Computational and Nonlinear Dynamics, vol. 9, no. 2, p. 021005, Sep. 2013, doi: 10.1115/1.4024911.

[5] F. C. Anderson and M. G. Pandy, "Dynamic Optimization of Human Walking," Journal of Biomechanical Engineering, vol. 123, no. 5, p. 381, 2001, doi: 10.1115/1.1392310.

[6] M. Sartori, M. Reggiani, D. Farina, and D. G. Lloyd, "EMG-Driven Forward-Dynamic Estimation of Muscle Force and Joint Moment about Multiple Degrees of Freedom in the Human Lower Extremity," PLoS ONE, vol. 7, no. 12, p. e52618, Dec. 2012, doi: 10.1371/journal.pone.0052618.

[7] E. Todorov, and M. Jordan. "Optimal Feedback Control as a Theory of Motor Coordination" Nature Publishing Group. Published online October 28th, 2002. Doi:10.1038/nn963.

[8] E. Todorov. "Stochastic Optimal Control and Estimation Methods Adapted to the Noise Characteristics of the Sensorimotor System." Department of Cognitive Science, University of California San Diego. Neural Comput. 17(5): 1084-1108. May 2005.

[9] J. Izawa, T. Rane, O. Donchin, and R. Shadmehr, "Motor adaptation as a process of reoptimization," Journal of Neuroscience, vol. 28, no. 11, pp. 2883–2891, Mar. 2008, doi: 10.1523/JNEUROSCI.5359-07.2008.

[10] F. Crevecoeur, S. H. Scott, and T. Cluff, "Robust Control in Human Reaching Movements: A Model-Free Strategy to Compensate for Unpredictable Disturbances," The Journal of Neuroscience, vol. 39, no. 41, pp. 8135–8148, Oct. 2019, doi: 10.1523/JNEUROSCI.0770-19.2019.

[11] J. Diedrichsen, R. Shadmehr, and R. B. Ivry, "The coordination of movement: optimal feedback control and beyond," Trends in Cognitive Sciences, vol. 14, no. 1, pp. 31–39, Jan. 2010, doi: 10.1016/j.tics.2009.11.004.

[12] M. Sadeghi et al., "Inferring control objectives in a virtual balancing task in humans and monkeys," eLife, Aug. 2023, doi: 10.7554/eLife.88514.1.

[13] S. H. Yeo, D. W. Franklin, and D. M. Wolpert, "When optimal feedback control is not enough: feedforward strategies are required for optimal control with active sensing," PLOS Computational Biology, vol. 12, no. 12, p. e1005190, Dec. 2016, doi: 10.1371/journal.pcbi.1005190.

[14] S. H. Scott, T. Cluff, C. R. Lowrey, and T. Takei, "Feedback control during voluntary motor actions," Current Opinion in Neurobiology, vol. 33, pp. 85–94, Aug. 2015, doi: 10.1016/j.conb.2015.03.006.

[15] R. Sharif Razavian, M. Sadeghi, S. Bazzi, R. Nayeem, and D. Sternad, "Body Mechanics, Optimality, and Sensory Feedback in the Human Control of Complex Objects," Neural Computation, pp. 1–43, Mar. 2023.

[16] H. R. Sheahan, D. W. Franklin, and D. M. Wolpert, "Motor Planning, Not Execution, Separates Motor Memories," Neuron, vol. 92, no. 4, pp. 773–779, Nov. 2016, doi: https://doi.org/10.1016/j.neuron.2016.10.017.

[17] E. Todorov. "Stochastic Optimal Control and Estimation Methods Adapted to the Noise Characteristics of the Sensorimotor System." Department of Cognitive Science, University of California San Diego. Neural Comput. 17(5): 1084-1108. May 2005.

[18] P. Morasso, "Spatial control of arm movements," Experimental Brain Research, vol. 42, no. 2, pp. 223–227, Apr. 1981, doi: 10.1007/BF00236911.

[19] J. M. Maciejowski. "Predictive Control with Constraints." Prentice Hall. Pearson Education. Published 2002. Print.

[20] R. Hari Krishnan, V. Devanandh, A. Kiran Brahma, S. Pugazhenthi. "Estimation of mass moment of inertia of human body, when bending forward, for the design of a self-transfer robotic facility." Journal of Engineering Science and Technology. 11. 166-176. 2016.

[21] J. H. Challis, "Producing physiologically realistic individual muscle force estimations by imposing constraints when using optimization techniques," Medical Engineering and Physics, vol. 19, no. 3, pp. 253–261, Apr. 1997, doi: 10.1016/s1350-4533(96)00062-8.

[22] R. A. Donatelli, "Sensory Integration and Neuromuscular Control of the Shoulder," in Physical Therapy of the Shoulder, 5th ed. New York, NY, USA: Churchill Livingstone, 2012, ch. 7.

[23] T. Flash, and N. Hogan. "The coordination of arm movements: an experimentally confirmed mathematical model." Journal of Neuroscience. Vol. 5, No 7. pp. 1688-1703. July 1985.

[24] A. Grancharova and T. A. Johansen, "Explicit Nonlinear Model Predictive Control." Springer Science+Business Media, 2012. doi: 10.1007/978-3-642-28780-0.