Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 56-3 (2023) 337–342

# Model Predictive Control for Tremor Suppressing Exoskeleton

**Subham Samal** * **Oumar Barry** **

* *Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061 USA (e-mail: subhamsamal@vt.edu)*

** *Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061 USA (e-mail: obarry@vt.edu)*

**Abstract:** This paper focuses on the application of model-based predictive control (MPC) for a full wrist exoskeleton designed for the alleviation of tremors in patients suffering from Parkinson's Disease and Essential Tremor. The main motivation for using MPC here relies on its ability to incorporate state and input constraints, which are crucial for the user's safety. The forearm-exoskeleton model is successively linearized at each time sample to obtain a linear state space model. The optimal input is then generated by minimizing a convex quadratic cost function. Finally, simulation cases are provided to demonstrate the effectiveness of the control scheme.



*Keywords:* Optimal Control, Robotics

## NOMENCLATURE

The mathematical notations used are listed as following:

| | |
|---|---|
| $\|\mathbf{Z}\|^n$ | $n$-norm of a matrix $\mathbf{Z}$ |
| $\mathbf{A} \succ \mathbf{0}$ | Square matrix $\mathbf{A}$ is positive definite |
| $\mathbf{I}_n$ | Identity Matrix of dimension $n$ |
| $\text{col}(a_1...a_n)$ | Column vector with elements $a_1...a_n$ |
| $\text{diag}(a_1...a_n)$ | Diagonal matrix with diagonal entries $a_1...a_n$ |

## 1. INTRODUCTION

Controlling an exoskeleton system is a challenging problem due to the complexity and nonlinearity of the exoskeleton dynamics. Various control algorithms have been introduced in the last few years to overcome these challenges, notably sliding mode controllers and adaptive control approaches. In recent times, there has been more attention towards model based optimal control schemes. These methods aim to reduce future tracking errors while simultaneously satisfying different constraints by solving a finite-horizon optimal control problem. Their inherent robustness and ability to control complex uncertain and disturbed systems successfully has currently made them a popular choice among researchers.

In model-based optimal control, an optimization algorithm uses the system's dynamical model to optimize a suitable cost function and predict the outcome of possible actions to derive an optimal future plan. Model-Predictive control (MPC) is one such realization of model-based optimal control (Camacho and Alba (2013)). At each sampling interval, the behaviour of the system over some future timesteps is predicted by the model. The length of this future timesteps is defined as the prediction horizon. Based on these predictions, an objective function is minimized with respect to the future sequence of inputs. This minimization function is a constrained optimization problem, which is solved for each sampling interval. Although prediction and optimization are performed over the complete prediction horizon, only the first input sequence, or inputs for the current sampling interval are used. This same procedure is repeated for the next sampling time, by sliding the time window to one sampling time forward. This mechanism is known as moving or receding horizon strategy.

The model of the coupled hand-exoskeleton system is complex and nonlinear. In recent times, nonlinear model predictive control (NMPC) (Allgöwer and Zheng (2012)) has been well developed for robotic applications (Wilson et al. (2016);Vougioukas (2007)), but at the same time, it has been observed that the computational effort required is significantly higher compared to the linear version. NMPC requires a nonlinear programming problem to be solved online, which is non-convex, has a larger number of decision variables, and a global minimum that is in general, impossible to find (Henson (1998);Kuhne et al. (2004)). In this paper, a successive linearization approach of the hand-exoskeleton dynamical model is used to yield a linear, time-varying description of the system, which is then solved through linear MPC. Then, considering the control inputs as the decision variables, the optimization problem is transformed into a convex quadratic programming (QP) problem, which is solved by numerically robust solvers to obtain global optimal solutions.

This study proposes to implement a model predictive controller to the exoskeleton developed in Wang et al. (2019) to suppress pathological tremors, i.e. Essential tremors (Louis (2001)) and Parkinson's disease (Jankovic (2008)). Tremor suppression orthoses can be classified into active, semi-active, and passive. Passive orthoses aren't tuneable to the user's or the environment's needs, and also offer a resistive force for the voluntary motions of the user along with tremor suppression, making them less effective (Fromme et al. (2019)). While many active wrist orthoses


10.1016/j.ifacol.2023.12.047

(Herrnstadt and Menon (2016); Wang (2023);Rocon et al. (2007)) have successfully alleviated tremors by applying appropriate forces to the wrist joint, they cannot conveniently incorporate input force/torque and range of motion limitations, which are crucial to the safety of the user. This is possible in model predictive controller, hence motivating us to explore the same. While MPC has been successfully implemented in the past for exoskeleton control, this study aims to extend its use for tremor suppression.

The rest of the paper is arranged as follows: In Section 2, the linearization formulation of the human-exoskeleton model is discussed. In Section 3, the Model Predictive Controller algorithm is set up with a quadratic cost function for minimization based on the linearized model. In Section 4, the performance of the controller is demonstrated by MATLAB simulations of the wearable exoskeleton, followed by Section 5, where active tremor suppression with BMFLC(Veluvolu and Ang (2011)) is explored. Finally, Section 6 summarizes the findings and proposes future work.

## 2. MODEL AND LINEARIZATION

In this study, the proposed controller is designed on the TAWE exoskeleton (Wang and Barry (2021)), which follows a generic human-exoskeleton multibody model structure:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} = -\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{h}(\mathbf{q},\dot{\mathbf{q}},t) + \mathbf{J_u^T}(\mathbf{q})\mathbf{u} \tag{1}$$

where $\mathbf{q} \in \mathbf{R}^{n_q}$is the generalized coordinate; $t \in \mathbf{R}_+$ is the time variable and $\mathbf{u} \in \mathbf{R}^{n_u}$ is the control input from the exoskeleton. $\mathbf{M} \in \mathbf{R}^{n_q \times n_q}$ is the inertia matrix and satisfies $\mathbf{M} = \mathbf{M^T} \succ \mathbf{0}$; $\mathbf{C} \in \mathbf{R}^{n_q \times n_q}$ is the Coriolis and centripetal matrix; $\mathbf{h} \in \mathbf{R}^{n_q}$ is the generalized force vector, and includes potential energy forces and energy dissipation forces, and time-dependent excitations; $\mathbf{J_u} \in \mathbf{R}^{n_u \times n_q}$ is the control input Jacobian matrix.

The state is defined as $\mathbf{x} = \mathrm{col}(\mathbf{q}, \dot{\mathbf{q}})$, and thus the state equation can be written as:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{M^{-1}}(-\mathbf{C}\dot{\mathbf{q}} - \mathbf{h} + \mathbf{J_u^T}\mathbf{u}) \end{bmatrix} \tag{2}$$

which is of the form: $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u})$

The system can be linearized about the point $(\mathbf{x_r}, \mathbf{u_r})$ by taking the Taylor expansion and discarding the higher order terms:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u}) = \mathbf{f}(\mathbf{x_r},\mathbf{u_r}) + \mathbf{f_{x,r}}(\mathbf{x}-\mathbf{x_r}) + \mathbf{f_{u,r}}(\mathbf{u}-\mathbf{u_r}) \tag{3}$$

where

$$\mathbf{f_{x,r}} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\substack{x=x_r \\ u=u_r}} \quad \text{and} \quad \mathbf{f_{u,r}} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\substack{x=x_r \\ u=u_r}}$$

Using Euler's method, for sample time $\Delta t$, Eqn. (3) can be discretized as:

$$\begin{aligned} \mathbf{x(t+1)} &= \mathbf{x(t)} + \mathbf{f(x_r,u_r)\Delta t} \\ &\quad + \mathbf{f_{x,r}(x(t)-x_r)\Delta t} + \mathbf{f_{u,r}(u(t)-u_r)\Delta t} \\ \implies \mathbf{x(t+1)} &= (\mathbf{I} + \mathbf{f_{x,r}\Delta t})\mathbf{x(t)} + (\mathbf{f_{u,r}\Delta t})\mathbf{u(t)} \\ &\quad + (\mathbf{f(x_r,u_r)} - \mathbf{f_{x,r}x_r} - \mathbf{f_{u,r}u_r})\mathbf{\Delta t} \end{aligned} \tag{4}$$

The Jacobian matrices $\mathbf{f_{x,r}}$ and $\mathbf{f_{u,r}}$ are calculated as:

$$\mathbf{f_{x,r}} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{n_q} \\ \mathbf{A1} & \mathbf{A2} \end{bmatrix} \qquad \mathbf{f_{u,r}} = \begin{bmatrix} \mathbf{0} \\ \mathbf{B} \end{bmatrix}$$

where

$$\begin{aligned} \mathbf{A1} &= \frac{\partial}{\partial \mathbf{q}}\mathbf{M^{-1}}(-\mathbf{C}\dot{\mathbf{q}} - \mathbf{h} + \mathbf{J_u^T}\mathbf{u}) \\ &= \frac{\partial \mathbf{M^{-1}}}{\partial \mathbf{q}}(-\mathbf{C}\dot{\mathbf{q}} - \mathbf{h} + \mathbf{J_u^T}\mathbf{u}) \\ &\quad + \mathbf{M^{-1}}(-\frac{\partial \mathbf{C}}{\partial \mathbf{q}}(\dot{\mathbf{q}}) - \frac{\partial \mathbf{h}}{\partial \mathbf{q}} + \frac{\partial \mathbf{J_u^T}}{\partial \mathbf{q}}(\mathbf{u})) \end{aligned} \tag{5a}$$

$$\begin{aligned} \mathbf{A2} &= \frac{\partial}{\partial \dot{\mathbf{q}}}\left(\mathbf{M^{-1}}\left(-\mathbf{C}\dot{\mathbf{q}} - \mathbf{h} + \mathbf{J_u^T}\mathbf{u}\right)\right) \\ &= \mathbf{M^{-1}}\left(-\frac{\partial \mathbf{C}}{\partial \dot{\mathbf{q}}}\dot{\mathbf{q}} - \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{q}}}\right) \end{aligned} \tag{5b}$$

$$\mathbf{B} = \frac{\partial}{\partial \mathbf{u}}\mathbf{M^{-1}}(-\mathbf{C}\dot{\mathbf{q}} - \mathbf{h} + \mathbf{J_u^T}\mathbf{u}) = \mathbf{M^{-1}}\mathbf{J_u^T} \tag{5c}$$

The modeling of the system has been done using the symbolic toolbox in MATLAB. Hence, first calculating $\mathbf{M^{-1}}$ symbolically and then calculating $\frac{\partial \mathbf{M^{-1}}}{\partial \mathbf{q}}$ isn't numerically efficient. Hence $\frac{\partial \mathbf{M^{-1}}}{\partial \mathbf{q}}$ is instead calculated as:

$$\frac{\partial \mathbf{M^{-1}}}{\partial \mathbf{q}} = \frac{\partial(\mathbf{M^{-1}}\dot{\mathbf{M}}\mathbf{M^{-1}})}{\partial \dot{\mathbf{q}}} \tag{6}$$

where $\mathbf{M^{-1}}$ is calculated numerically.

## 3. THE MPC ALGORITHM

The essence of a Model Predictive Controller is to optimize predictions of a system's behaviour over a sequence of future control inputs. At each sampling time, a minimization function is solved to generate an optimal control sequence, and its first element is applied to the plant. At the next sampling time, with the updated system states, the problem is solved again to obtain the optimal control sequence. The minimization function is generally formulated to be a quadratic function of the states and control inputs(Kuhne et al. (2004)).

The objective here is that when provided a reference trajectory $x^{des}$, the controller calculates the optimal actuation input basis the dynamics and defined cost function, and provides these inputs to the human-exoskeleton dynamic system. The cost function is defined such that the calculated actuation inputs can track the exoskeleton motion along a given reference.

Taking the prediction horizon as $\mathbf{N}$, the cost function is defined as:

$$\begin{aligned} \mathbf{J_{0\to N}}(\mathbf{x_0},\mathbf{U_{0\to N}}) &:= \|\mathbf{x_N} - \mathbf{x_N^{des}}\|_{\mathbf{P}}^{\mathbf{2}} \\ &\quad + \sum_{\mathbf{k=0}}^{\mathbf{N-1}} (\|\mathbf{x_k} - \mathbf{x_k^{des}}\|_{\mathbf{Q}}^{\mathbf{2}} + \|\mathbf{u_k}\|_{\mathbf{R}}^{\mathbf{2}}) \end{aligned} \tag{7}$$

subject to the constraints :

$$\mathbf{x_{k+1}} = \mathbf{A x_k} + \mathbf{B u_k} + \mathbf{K} \quad \text{for k = 0,1,..N-1} \tag{8a}$$

$$\mathbf{X_l} \le \mathbf{x_k} \le \mathbf{X_u} \quad \text{for k = 0,1,..N-1} \tag{8b}$$

$$\mathbf{U_l} \le \mathbf{u_k} \le \mathbf{U_u} \quad \text{for k = 0,1,..N-1} \tag{8c}$$

Here, $\mathbf{P} = \mathbf{P^T} \succ \mathbf{0}$, $\mathbf{Q} = \mathbf{Q^T} \succ \mathbf{0}$ and $\mathbf{R} = \mathbf{R^T} \succ \mathbf{0}$ are the cost matrices associated with the terminal state, non-terminal states and inputs respectively. Equation (8a) is a

simplified version of (4), representing the state equation while (8b), (8c) represents the state and input constraints.

This cost function can be written as:

$$\mathbf{Z}_{\mathbf{0}\to\mathbf{N}}^{\mathbf{T}} \begin{bmatrix} \bar{\mathbf{Q}} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{R}} \end{bmatrix} \mathbf{Z}_{\mathbf{0}\to\mathbf{N}} + \mathbf{g}\mathbf{Z}_{\mathbf{0}\to\mathbf{N}} \tag{9}$$

where

$$\mathbf{Z}_{\mathbf{0}\to\mathbf{N}} = \mathrm{col}(\mathbf{x_1}, \mathbf{x_2}, ...\mathbf{x_N}, \mathbf{u_0}, \mathbf{u_1}, ...\mathbf{u_{N-1}})$$
$$\bar{\mathbf{Q}} = \mathrm{diag}(\mathbf{Q}, \mathbf{Q}..., \mathbf{P})$$
$$\bar{\mathbf{R}} = \mathrm{diag}(\mathbf{R}, \mathbf{R}..., \mathbf{R})$$
$$\mathbf{g} = [-\mathbf{2}(\mathbf{x_1^{des}})^{\mathbf{T}}\mathbf{Q}, -\mathbf{2}(\mathbf{x_2^{des}})^{\mathbf{T}}\mathbf{Q}, ... - \mathbf{2}(\mathbf{x_N^{des}})^{\mathbf{T}}\mathbf{P}, \mathbf{0}, \mathbf{0}...\mathbf{0}]$$

The state equations and the constraint equations can be written in matrix form as

$$\begin{bmatrix} \mathbf{I}_{n_x} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & -\mathbf{B} & \mathbf{0} & \dots & \mathbf{0} \\ -\mathbf{A} & \mathbf{I}_{n_x} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & -\mathbf{A} & \mathbf{I}_{n_x} & \mathbf{0} & \mathbf{0} & \dots & -\mathbf{B} \end{bmatrix} [\mathbf{Z}_{\mathbf{0}\to\mathbf{N}}] = \begin{bmatrix} \mathbf{A}\mathbf{x_0} + \mathbf{K} \\ \mathbf{K} \\ \vdots \\ \mathbf{K} \end{bmatrix} \tag{10a}$$

$$\begin{bmatrix} \mathbf{I}_{2\mathrm{N}\times(n_x+n_u)} \\ -\mathbf{I}_{2\mathrm{N}\times(n_x+n_u)} \end{bmatrix} [\mathbf{Z}_{\mathbf{0}\to\mathbf{N}}] \leq \begin{bmatrix} (\mathbf{X_u})_{\mathbf{N}\times\mathbf{1}} \\ (\mathbf{U_u})_{\mathbf{N}\times\mathbf{1}} \\ (-\mathbf{X_l})_{\mathbf{N}\times\mathbf{1}} \\ (-\mathbf{U_l})_{\mathbf{N}\times\mathbf{1}} \end{bmatrix} \tag{10b}$$

The cost function in (9) is quadratic, and various quadratic solvers can be used to obtain $\mathbf{Z}_{\mathbf{0}\to\mathbf{N}}$ that minimizes the cost function. In our simulations, this quadratic optimization problem has been solved using the qpSWIFT solver as it has been found to be faster than most other solvers(Pandala et al. (2019)).

## 4. SIMULATIONS AND DISCUSSION

This section demonstrates the performance of the developed model predictive controller through simulations of the forearm model and wearable wrist exoskeleton (Wang et al. (2018)) carried out using the ANDY Toolbox (Wang et al. (2018)), an analytical multibody toolbox in MATLAB, validated in Wang and Barry (2020). Fig. 1 shows the conceptual design of the TAWE exoskeleton, while the 3D visualization of simulation is shown in Fig. 2. For some of the simulations, the references $\mathbf{r}$ are selected as bounded periodic and quasiperiodic trajectories with multiple harmonic components (Wang and Barry (2021)). The simulation and control sampling rate is taken as 500 Hz, and the frequency at which the linearization is done is 100 Hz.

The weighing matrices, defined in section 3 are taken to be: $\mathbf{P} = \mathbf{Q} = \mathrm{diag}(200, 1000, 0.1, 0.1)$ and $\mathbf{R} = 0.4\mathbf{I_2}$. The prediction horizon $\mathbf{N}$ is taken to be 20.

### *4.1 Forearm model*

Initially, the developed model was tested on a forearm (wrist) model, with the MPC algorithm implemented to obtain the optimal control inputs in Flexion-Extension and Radial-Ulnar Deviation directions to follow a reference trajectory.

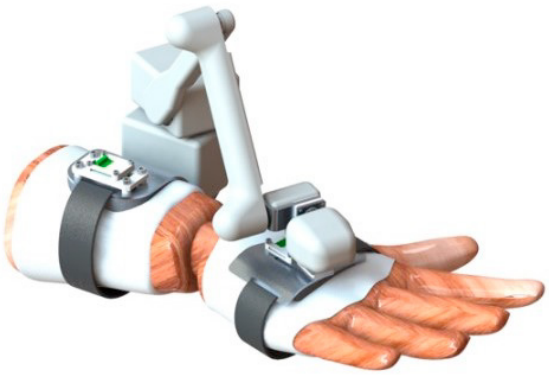

Fig. 1. The CAD model of the conceptual design of TAWE attached to a right human forearm mannequin (Wang and Barry (2020))

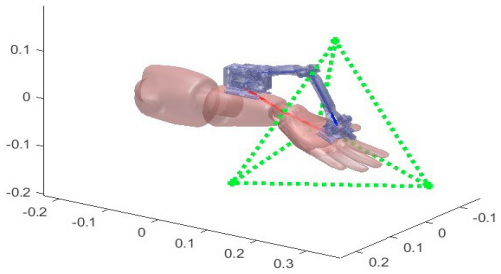


Fig. 2. 3D model of the TAWE simulation in MATLAB environment using ANDY Toolbox

The wrist is a constrained 3D rotational joint and the 3 DOFs are given as radial-ulnar deviation (RUD), flexion-extension (FE), and supination-pronation (SP). Here, the generalized coordinate is considered as $\mathbf{q_1} = [\theta_{\mathrm{RUD}}, \theta_{\mathrm{FE}}, \theta_{\mathrm{SP}}]^T$. The rotation $\theta_{\mathrm{SP}}$ on the SP direction is constrained to $[\theta_{\mathrm{RUD}}, \theta_{\mathrm{FE}}]^T$ (Li et al. (2005)). To formulate this, a kinematic constraint is defined, and is given as $\mathbf{r}_c(\mathbf{q_1}) \in \mathbf{R}$, which constrains the $\theta_{\mathrm{SP}}$ DOF of the system. The time derivative of $\mathbf{r}_c$ is given as

$$\dot{\mathbf{r}}_c = \mathbf{0} = \mathbf{J}_{c,q}(\mathbf{q}, \rho)\dot{\mathbf{q}} + \mathbf{J}_{c,\rho}(\mathbf{q}, \rho)\dot{\rho}; \qquad \rho = [\theta_{\mathrm{SP}}] \tag{11}$$

where $\mathbf{J}_{c,q} \in \mathbf{R}^{1\times2}$ and $\mathbf{J}_{c,\rho} \in \mathbf{R}^{1\times1}$ are Jacobian matrices. With $\dot{\rho} = -\mathbf{J}_{c,\rho}^{-1}\mathbf{J}_{c,q}$, a 2-DOF assembled dynamical model is obtained which follows the dynamical model structure given in Eq. (1) (Wang and Barry (2021)).

Since the control torques are applied at the wrist joint in FE and RUD directions, the state and input constraints were considered basis the average range of motion and strength of the wrist joint. The torque limits for flexion and extension were considered to be 12 Nm and 7 Nm respectively, and for radial and ulnar deviations, the limits were 11 Nm and 9.5 Nm respectively (Delp et al. (1996);Yoshii et al. (2015)). In FE and RUD directions, the limits are $-75°$ to $75°$ (-1.31 rad to 1.31 rad) and $-45°$ to $25°$ (-0.79 rad to 0.44 rad) respectively (Wang (2023)). According to a study (Rosen et al. (2005)) of the kinematics and dynamics of the arm in various daily activities, the max angular velocities in flexion and extension were recorded

to be 232.9 deg/s (4.06 rad/s)and 141.2 deg/s (2.46 rad/s). Similarly for radial and ulnar deviation, it was recorded to be 203.9 deg/s (3.56 rad/s) and 180.4 deg/s (3.14 rad/s) respectively. Keeping some buffer, the angular velocity limits were kept at 270 deg/s ($3\pi/2$ rad/s) for both FE and RUD.

The tracking performance is presented in figure 3. From figures 3(a) and 3(b), it can be observed that the trajectory of $\mathbf{q}$ is accurately following the set references, and the errors in 3(c) and 3(d) are close to zeros, confirming the efficacy of the controller.

The forearm model example allows us to observe the suitability and performance behaviors of the controller. The MPC Controller is used in the wearable exoskeleton for the next case.

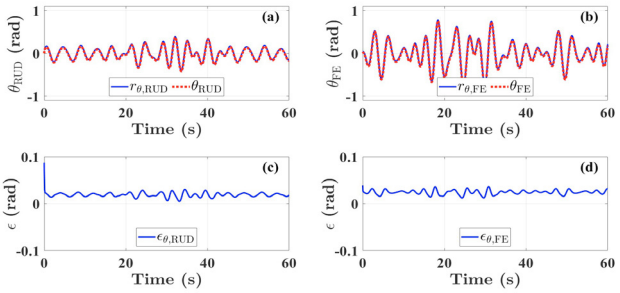


Fig. 3. The control system performance when applied only on forearm for trajectory tracking, where (a–b) shows the trajectories followed ($\theta$) and reference ($r_\theta$) from the simulation; (c-d) shows the tracking errors ($\epsilon = r_\theta - \theta$)

### *4.2 Forearm with TAWE Exoskeleton*

TAWE (Wang and Barry (2020)) - a wearable wrist exoskeleton has been developed by our team for active pathological tremor alleviation and movement assistance, the conceptual design of which is shown in Figure 1. TAWE constitutes a 6-DOF rigid linkage mechanism that allows unconstrained wrist movement, while its kinematics is defined by the biomechanism of the wrist. The detailed modeling of forearm-TAWE dynamics has been explained and validated in (Wang and Barry (2020)). Here, a simpler case is considered, where the forearm is fixed. This leads to the forearm being a 3-DOF system with $\mathbf{q_1} = [\theta_{\mathrm{RUD}}, \theta_{\mathrm{FE}}, \theta_{\mathrm{SP}}]^T$ as the wrist 3D rotation angles, and TAWE being a 6-DOF system with $\mathbf{q_2} \in R^6$ as its six joints. The closed kinematic chain formed between the forearm and TAWE results in $\mathbf{q_2}$ being fully constrained to $\mathbf{q_1}$, and as described in section 4.1, the rotation $\theta_{\mathrm{SP}}$ is constrained to $[\theta_{\mathrm{RUD}}, \theta_{\mathrm{FE}}]^T$. Similar to section 4.1 these configurations lead to a set of kinematic constraints defined as $\mathbf{r}_c(\mathbf{q_1}, \mathbf{q_2}) \in \mathbf{R^7}$, which constrains seven DOFs of the system in total, and the time derivative of $\mathbf{r}_c$ is calculated as:

$$\dot{\mathbf{r}}_c = \mathbf{0} = \mathbf{J}_{c,q}(\mathbf{q}, \rho)\dot{\mathbf{q}} + \mathbf{J}_{c,\rho}(\mathbf{q}, \rho)\dot{\rho}; \qquad \rho = [\theta_{\mathrm{SP}} \quad \mathbf{q_2^T}]^\mathbf{T} \tag{12}$$

$\mathbf{J}_{c,q} \in \mathbf{R}^{7\times 2}$ and $\mathbf{J}_{c,\rho} \in \mathbf{R}^{7\times 7}$ are Jacobian matrices, and with $\dot{\rho} = -\mathbf{J}_{c,\rho}^{-1}\mathbf{J}_{c,q}$, a 2-DOF assembled dynamical model can be obtained. The control inputs are provided at the first two joints of TAWE Wang and Barry (2021).

First, the tracking performance is tested as shown in Fig. 4. It can be observed from figure 4 that the trajectory followed for the TAWE-Forearm model closely matches the trajectory followed by the Forearm model, and thus, closely follows the set trajectory. Figure 5 shows the control inputs obtained for trajectory tracking for the Forearm-TAWE system. These results confirm that the MPC controller can be extended to the forearm-TAWE system as well.

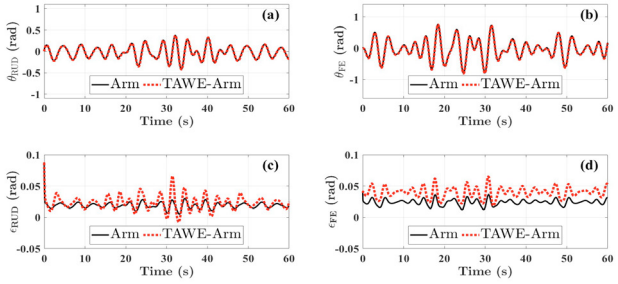


Fig. 4. The control system performance applied to Forearm-TAWE and Forearm model for trajectory tracking, where (a–b) shows the trajectories followed ($\theta$) and (c-d) shows the tracking errors ($\epsilon = r_\theta - \theta$)

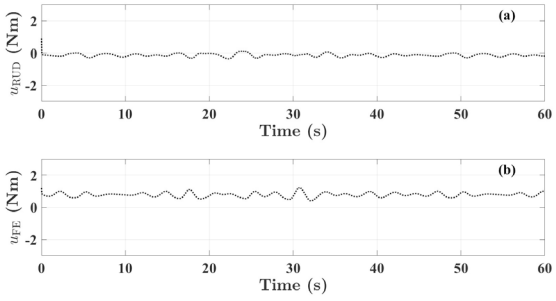


Fig. 5. The inputs for trajectory tracking with TAWE; where (a–b) shows the control inputs in FE and RUD

Then, tremors are introduced as a model uncertainty and the performance for tremor suppression with MPC is observed. For this, the Forearm-exoskeleton is kept stationary, i.e. the set trajectory to follow is kept as 0 in both FE and RUD directions. The results of the MPC Controller is compared with a proportional-derivative controller (PD), with a feed-forward term. The feed-forward input is constant, and is given to compensate for the torque due to gravity, while the PD term drives back the system to [0,0] as the tremors act to move the system away from [0,0]. Figure 6 shows that when the PD controller is applied, the deviation is quite large, while in figures 7(a) and 7(b), it can be observed that the error oscillation amplitudes are significantly reduced when MPC is applied, thus confirming that MPC is applicable for tremor suppression as well.

## 5. EXPLORING MPC WITH TREMOR MODELING

In the previous sections, it was observed that the developed MPC model can provide passive tremor suppression. In this section, implementing active tremor suppression using BMFLC was explored.

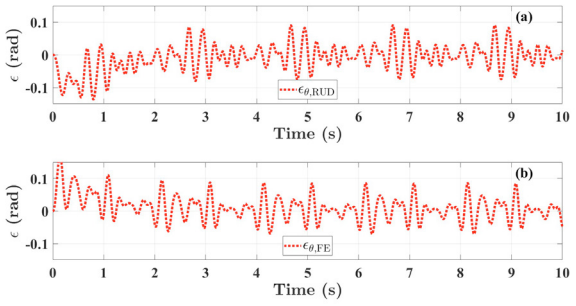


Fig. 6. The deviation with a PD Controller; where (a–b) shows the tracking errors in RUD and FE ($\epsilon = \mathrm{r}_\theta - \theta$)

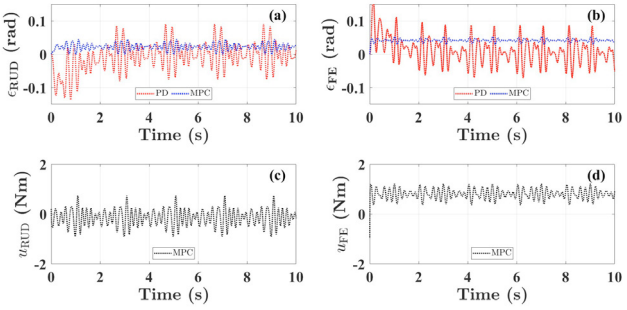


Fig. 7. Comparison of the stationary exoskeleton-forearm tracking controls between PD and MPC Controllers (a-b) ($\epsilon = \mathrm{r}_\theta - \theta$). The control input generated from MPC is shown in (c-d)

The synthetic tremor excitation introduced to the model is assumed to be a combination of harmonic waves with different frequencies within a certain bandwidth. For example, for Parkinsonian tremor (Rocon et al. (2007))), this range is 3–6 Hz. A band-limited multi-frequency Fourier linear combiner (BMFLC) (Veluvolu and Ang (2011)) was designed to model the tremor, which is included as a part of the cost function for the model predictive controller to compensate for the synthetic tremor, and the compensating torque was included as part of the constraints. A BMFLC model with $n_{\mathrm{BMFLC}}$ frequency components can be structured as:

$$\mu_{\mathrm{BMFLC}}(t) = \sum_{i=1}^{n_{\mathrm{BMFLC}}} (p_{\mu,\mathrm{BMFLC},i} sin(c_{\mu,\mathrm{BMFLC},i} t) + p_{\mu,\mathrm{BMFLC},i+n} cos(c_{\mu,\mathrm{BMFLC},i} t)) \quad (13)$$

where $p_{\mu,\mathrm{BMFLC},i}$ is the $i^{th}$ uncertain amplitude parameter and $c_{\mu,\mathrm{BMFLC},i}$ is the $i^{th}$ constant frequency. The frequencies $c_{\mu,\mathrm{BMFLC},1}$ and $c_{\mu,\mathrm{BMFLC},n_{\mathrm{BMFLC}}}$ determines the bandwidth, and $n_{\mathrm{BMFLC}}$ determines the frequency-domain resolution of model (Wang (2023)).

To obtain these uncertain amplitude parameters, a cost term is introduced to (7) which updates those parameters at each time sample. Thus, the modified cost function can be written as:

$$\mathbf{J_{0\to N}}(\mathbf{x_0}, \mathbf{U_{0\to N}}) := \|\mathbf{x_N} - \mathbf{x_N^{des}}\|_{\mathbf{P}}^{\mathbf{2}} + \sum_{\mathbf{k=0}}^{\mathbf{N-1}} (\|\mathbf{x_k} - \mathbf{x_k^{des}}\|_{\mathbf{Q}}^{\mathbf{2}} + \|\mathbf{u_k}\|_{\mathbf{R}}^{\mathbf{2}}) + \phi^{\mathbf{T}} \mathbf{S} \phi \quad (14)$$

where $\phi \in \mathbf{R}^{4n_{\mathrm{BMFLC}} \times 1}$ is the uncertain amplitude parameter vector given as :

$$\phi = [p_{\mu,1,\mathrm{RUD}}, .. \, p_{\mu,2n_{\mathrm{BMFLC}},\mathrm{RUD}}, p_{\mu,1,\mathrm{FE}}, .. \, p_{\mu,2n_{\mathrm{BMFLC}},\mathrm{FE}}]^T \quad (15)$$

and $\mathbf{S} = \mathbf{S^T} \succ \mathbf{0}$ is the cost matrix associated with the amplitude vector. The modified control input is given as $\mathbf{u_k} + \mathbf{u_{tr}}(t_\mathbf{k}, \phi)$ where $\mathbf{u_{tr}}(t_\mathbf{k}, \phi)$ is the additional virtual input that causes tremor, and is given as:

$$\mathbf{u_{tr}}(t_\mathbf{k}, \phi) = \mathbf{J^T_{\phi,\mathrm{BMFLC}}} \times \phi \quad (16)$$

$$\mathbf{J}_{\phi,\mathrm{BMFLC}} = \begin{bmatrix} sin(\mathrm{c}_{\mu,1,\mathrm{RUD}} t) & 0 \\ \vdots & \vdots \\ sin(\mathrm{c}_{\mu,n,\mathrm{RUD}} t) & 0 \\ cos(\mathrm{c}_{\mu,n+1,\mathrm{RUD}} t) & 0 \\ \vdots & \vdots \\ cos(\mathrm{c}_{\mu,2n,\mathrm{RUD}} t) & 0 \\ 0 & sin(\mathrm{c}_{\mu,1,\mathrm{FE}} t) \\ \vdots & \vdots \\ 0 & sin(\mathrm{c}_{\mu,n,\mathrm{FE}} t) \\ 0 & cos(\mathrm{c}_{\mu,n+1,\mathrm{FE}} t) \\ \vdots & \vdots \\ 0 & cos(\mathrm{c}_{\mu,2n,\mathrm{FE}} t) \end{bmatrix} \quad (17)$$

With the above modifications, the new decision variables $\mathbf{Z_{0\to N}}$ is given as $\mathrm{col}(\mathbf{x_1}, \mathbf{x_2}, ...\mathbf{x_N}, \mathbf{u_0}, \mathbf{u_1}, ...\mathbf{u_{N-1}}, \phi)$, and the state equation and input constraint are modified as:

$$\mathbf{x_{k+1}} = \mathbf{A x_k} + \mathbf{B}(\mathbf{u_k} + \mathbf{u_{tr}}(t_\mathbf{k}, \phi)) + \mathbf{K} \quad (18a)$$

$$\mathbf{U_l} \le \mathbf{u_k} + \mathbf{u_{tr}}(t_\mathbf{k}, \phi) \le \mathbf{U_u} \quad \text{for k = 0 to N-1} \quad (18b)$$

Considering these modified constraint and state equations, the equations (10a and 10b) are changed accordingly. It should be noted that the input to the exoskeleton is still $\mathbf{u_k}$, and $\mathbf{u_{tr}}$ is the estimated virtual input.

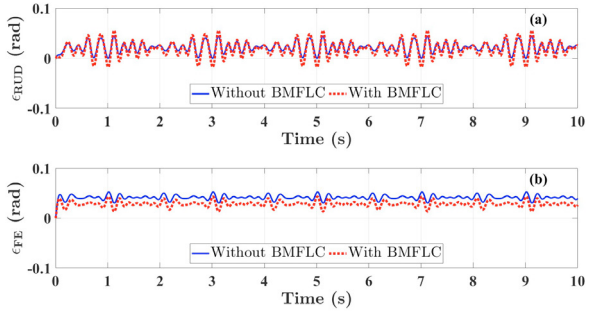


Fig. 8. The comparison for performances for tremor suppression using tracking errors ($\epsilon = \mathrm{r}_\theta - \theta$) for stationary exoskeleton for passive suppression (without BMFLC) and active suppression (with BMFLC)

From figure 8, it was observed that adding the BMFLC component leads to only slight improvement in the performance in the FE direction, but no improvement in the RUD direction. This is likely because the tremor should be modeled using past data, which has not been considered here, leading to an incomplete model.

Thus, in our future works, the tremor information of the past will be considered for estimating the tremor using the BMFLC model. Once the BMFLC component $\mathbf{u_{tr}}$ is updated, this updated model component will be used for model predictive control for active tremor suppression.

## 6. CONCLUSION AND FUTURE WORK

This paper presents a model predictive controller for an exoskeleton aimed for tremor suppression. The non-linear multibody system was linearized at each time sample to obtain the state space form, and a convex quadratic cost function with constraints was minimized to obtain the control inputs. The devised controller was then validated by the control simulations of the forearm and TAWE systems. The controller demonstrated good tracking performance as the errors between set references and followed trajectory was always close to zero ($\leq 0.07$ rad). Further, with added tremors, the error oscillation amplitudes were significantly smaller for the MPC controller compared to the PD controller, thus demonstrating passive tremor suppression.

As mentioned in section 5, methods to estimate tremor using past tremor information for active tremor suppression will be investigated. Also, in this study, disturbances were not considered and the inertia of the human arm was considered to be known with certainty. Further, it is assumed here that wrist kinematics is known, which would not be the case in practical applications. Thus, in future works, investigation would be done to implement this controller with uncertain inertias and disturbances. A Wrist Kinematics Identification (WKI) algorithm developed in Wang (2023) would be implemented to obtain an approximated wrist model and wrist angles using the inertia measurement units (IMU) and encoders on TAWE. The performance of the controller would be further tested through experiments, including testing on human subjects and the performance would be investigated with different motion planning algorithms.

## ACKNOWLEDGEMENTS

The authors would like to thank Dr. Jiamin Wang for their expertise and guidance throughout all aspects of this study.

## REFERENCES

Allgöwer, F. and Zheng, A. (2012). *Nonlinear model predictive control*, volume 26. Birkhäuser.

Camacho, E.F. and Alba, C.B. (2013). *Model predictive control*. Springer science & business media.

Delp, S.L., Grierson, A.E., and Buchanan, T.S. (1996). Maximumisometric moments generated by the wrist muscles in flexion-extension and radial-ulnar deviation. *Journal of Biomechanics*, 29(10), 1371–1375.

Fromme, N.P., Camenzind, M., Riener, R., and Rossi, R.M. (2019). Need for mechanically and ergonomically enhanced tremor-suppression orthoses for the upper limb: a systematic review. *Journal of neuroengineering and rehabilitation*, 16, 1–15.

Henson, M.A. (1998). Nonlinear model predictive control: current status and future directions. *Computers & Chemical Engineering*, 23(2), 187–202.

Herrnstadt, G. and Menon, C. (2016). Admittance-based voluntary-driven motion with speed-controlled tremor rejection. *IEEE/ASME Transactions on Mechatronics*, 21(4), 2108–2119.

Jankovic, J. (2008). Parkinson's disease: clinical features and diagnosis. *Journal of neurology, neurosurgery & psychiatry*, 79(4), 368–376.

Kuhne, F., Lages, W.F., and da Silva Jr, J.G. (2004). Model predictive control of a mobile robot using linearization. In *Proceedings of mechatronics and robotics*, 525–530. Citeseer.

Li, Z.M., Kuxhaus, L., Fisk, J.A., and Christophel, T.H. (2005). Coupling between wrist flexion–extension and radial–ulnar deviation. *Clinical biomechanics*, 20(2), 177–183.

Louis, E.D. (2001). Essential tremor. *New England Journal of Medicine*, 345(12), 887–891.

Pandala, A.G., Ding, Y., and Park, H.W. (2019). qpswift: A real-time sparse quadratic program solver for robotic applications. *IEEE Robotics and Automation Letters*, 4(4), 3355–3362.

Rocon, E., Belda-Lois, J.M., Ruiz, A., Manto, M., Moreno, J.C., and Pons, J.L. (2007). Design and validation of a rehabilitation robotic exoskeleton for tremor assessment and suppression. *IEEE Transactions on neural systems and rehabilitation engineering*, 15(3), 367–378.

Rosen, J., Perry, J.C., Manning, N., Burns, S., and Hannaford, B. (2005). The human arm kinematics and dynamics during daily activities-toward a 7 dof upper limb powered exoskeleton. In *ICAR'05. Proceedings., 12th International Conference on Advanced Robotics, 2005.*, 532–539. IEEE.

Veluvolu, K.C. and Ang, W.T. (2011). Estimation of physiological tremor from accelerometers for real-time applications. *Sensors*, 11(3), 3020–3036.

Vougioukas, S.G. (2007). Reactive trajectory tracking for mobile robots based on non linear model predictive control. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 3074–3079. IEEE.

Wang, J. (2023). *Design and Control of an Ergonomic Wearable Full-Wrist Exoskeleton for Pathological Tremor Alleviation.* Ph.D. thesis, Virginia Tech.

Wang, J., Barry, O., Kurdila, A.J., and Vijayan, S. (2019). On the dynamics and control of a full wrist exoskeleton for tremor alleviation. In *Dynamic Systems and Control Conference*, volume 59155, V002T27A008. American Society of Mechanical Engineers.

Wang, J. and Barry, O.R. (2020). Multibody analysis and control of a full-wrist exoskeleton for tremor alleviation. *Journal of Biomechanical Engineering*, 142(12).

Wang, J. and Barry, O.R. (2021). Inverse optimal robust adaptive controller for upper limb rehabilitation exoskeletons with inertia and load uncertainties. *IEEE Robotics and Automation Letters*, 6(2), 2171–2178.

Wang, J., Kamidi, V.R., and Ben-Tzvi, P. (2018). A multibody toolbox for hybrid dynamic system modeling based on nonholonomic symbolic formalism. In *Dynamic Systems and Control Conference*, volume 51913, V003T29A003. American Society of Mechanical Engineers.

Wilson, J., Charest, M., and Dubay, R. (2016). Non-linear model predictive control schemes with application on a 2 link vertical robot manipulator. *Robotics and Computer-Integrated Manufacturing*, 41, 23–30.

Yoshii, Y., Yuine, H., Kazuki, O., Tung, W.l., and Ishii, T. (2015). Measurement of wrist flexion and extension torques in different forearm positions. *Biomedical engineering online*, 14, 1–10.