Human Uncertainty-Aware MPC for Enhanced Human-Robot Collaborative Manipulation

Al Jaber Mahmud
Electrical and Computer Engineering
George Mason University
Virginia, USA
amahmud5@gmu.edu

Duc M. Nguyen
Computer Science
George Mason University
Virginia, USA
mnguy21@gmu.edu

Filipe Veiga
Electrical and Computer Engineering
George Mason University
Virginia, USA
ffveiga@gmu.edu

Xuesu Xiao
Computer Science
George Mason University
Virginia, USA
xiao@gmu.edu

Xuan Wang
Electrical and Computer Engineering
George Mason University
Virginia, USA
xwang64@gmu.edu

2024 IEEE 7th International Conference on Industrial Cyber-Physical Systems (ICPS) | 979-8-3503-6301-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICPS59941.2024.10640020

***Abstract*—This paper presents the development of a novel control algorithm designed for tasks involving human-robot collaboration. By using an 8-DOF robotic arm, our approach aims to counteract human-induced uncertainties added to the robot's nominal trajectory. To address this challenge, we incorporate a variable within the regular Model Predictive Control (MPC) framework to account for human uncertainties, which are modeled as following a normal distribution with a non-zero mean and variance. Our solution involves formulating and solving an uncertainty-aware Discrete Algebraic Ricatti Equation (ua-DARE), which yields the optimal control law for all joints to mitigate the impact of these uncertainties. We validate our methodology through theoretical analysis, demonstrating the effectiveness of the ua-DARE in providing an optimal control strategy. Our approach is further validated through simulation experiments using a Fetch robot model, where the results highlight a significant improvement in performance over a baseline algorithm that does not consider human uncertainty while solving for optimal control law.**

***Index Terms*—Model Predictive Control, Human-Robot Interaction, Human Uncertainty.**

I. Introduction

Systems involving collaboration between humans and robots have the potential to reduce the workload on human operators through robotic assistance significantly. The ability of such human-assistive autonomous robots to adapt to unpredictable human preferences while optimizing energy usage is a critical factor in achieving operational efficiency and ensuring safety [1]–[3]. In Human-Robot Collaboration (HRC), a common task is object co-manipulation [4], [5]. However, utilizing such co-manipulation introduces significant challenges that include the unpredictability of human actions [6], which may deviate from established paths, and the increased complexity in controlling the system, because of the high degree of freedom of a robotic arm [7].

To address these issues, this paper develops a control algorithm by formulating and solving a human uncertainty-aware Model Predictive Control (MPC). The modeling of human uncertainty in MPC tracking allows us to estimate its impact on costs in terms of tracking errors and energy consumption when performing manipulation tasks.

Literature review: Tracking problems arise from planning and control, where primary efforts are based on the platform of mobile robots. For ground vehicles, Ostafew et al. [8] developed a learning-based nonlinear MPC designed for navigating off-road terrains. Xiao et al. [9] introduced an imitation learning-based MPC aimed at facilitating navigation in crowded public spaces. Additionally, notable advancements have been made in the field of adaptive planning and control strategies [10]–[13]. For 3D space, various approaches have been developed for UAV platforms [14], where environmental features are less critical, while the vehicle dynamics becomes more sophisticated. These approaches enhance the robot's adaptability and responsiveness to dynamics and environmental changes.

In recent years, using robotic arms for end-effector trajectory tracking has attracted a lot of research interest. Baek et al. [15] presented a practical adaptive time-delay control applied in robot manipulators, Xu et al. [16] developed a learning from demonstration method for robotic arm trajectory tracking, and model predictive path-following control by Faulwasser et al. [17], focusing on precision and efficiency in arm movements. There also exist studies that focus on integrating robotic bases with arms for enhanced tracking performance. Quiroz-Omana et al. [18] proposed a robot kinematic control strategy using linear programming. Giftthaler et al. [19] solved a constrained sequential linear quadratic control problem to design a time-varying feedback controller. Osman et al. [20] used the whole body dynamics to obtain a task-space stabilization controller utilizing nonlinear MPC.

However, a common oversight in these studies is neglecting external uncertainties, especially those arising from human involvement in collaborative tasks. While robust MPC addresses uncertainties within the robot's control framework [21], [22], the focus is mainly on the robot's internal dynamics [23], [24], which are not directly applicable to the characterization of

This project is supported by NSF Grant #2332210.

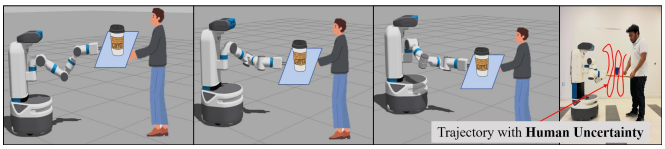

Fig. 1: Human robot co-transportation. The robot needs to adapt to human uncertainties added to a nominal trajectory .

external uncertainty.

Statement of contribution: In this paper, we study human-robot co-manipulation using a model of a robot manipulator. The main contributions of this work are as follows: First, we model human uncertainties that follow a normal distribution with non-zero mean and variance. We then formulate a new human uncertainty-aware MPC tracking problem. Then, we solve this problem to get the optimal control input sequence, i.e., the manipulator's joint angle velocities. Finally, we verify our approach's effectiveness using theoretical derivation and simulated experiments. We compare its performance with a baseline algorithm that does not consider human uncertainty while solving for optimal control law.

Notations: Let I_r denote the $r \times r$ identity matrix. Let $\text{diag}\{a_1, a_2, \cdots, a_r\}$ denote a diagonal matrix with a_i being the ith diagonal entry. For a vector x, $|x|_2$ denotes its 2-norm. For a square matrix M, $\text{Tr}(M)$ denotes its trace. We use $M \succ 0$, $M \succeq 0$ to denote the matrix is positive definite and positive semi-definite, respectively. We let $\|x\|_M^2 = x^\top M x$ with $M \succeq 0$, which represents a quadratic function of the vector evaluated by the matrix M.

II. Problem Statement and Formulation

This section formulates our problem of interest. We introduce a trajectory around a fetch robot with human uncertainties that the robot needs to track and adapt to. Then, we describe how we model human uncertainty and the robot's motion dynamics. Finally, we formulate a human uncertainty-aware MPC tracking problem to describe the control objective mathematically.

A. Nominal Trajectory and Trajectory with Human Uncertainty

We aim to perform a co-manipulation task between a robot and a human by balancing a board, as illustrated in Fig. 1. To this end, we define a nominal trajectory that represents the robot's end-effector's expected position and orientation in the world frame, which is defined as $r_t = [r_t^x, r_t^y, r_t^z, r_t^\alpha, r_t^\beta, r_t^\gamma]^\top \in \mathbb{R}^6$ for $t \in \{0, 1, 2, \cdots T\}$, where superscripts x, y, z represent the 3D positions and α, β, γ represent the roll, pitch, and yaw orientations, respectively. As we are balancing the board, the pitch (β) and yaw (γ) should always be kept zero. We assume that human attempts to follow the nominal trajectory, but their actions are subject to uncertainties. This will lead to a deviation of the board from the nominal trajectory, necessitating the robot to adjust accordingly. To model this, we consider a receding horizon tracking problem [25]. At each time step t, the robot seeks to track a segment of the trajectory for future H steps:

$$\tilde{r}(k) = r(k) + D(h + \sum_{\tau=0}^{k} \varpi(\tau)), \tag{1}$$

with $k \in \{0, 1, \cdots, H\}$ and $\varpi \in \mathbb{R}^3$ represents the human uncertainties modeling discussed in section II-B. $D = \begin{bmatrix} I_3 & \mathbf{0}_{3\times 3} \end{bmatrix}^\top \in \mathbb{R}^{6\times 3}$ is a matrix that maps human uncertainties onto the nominal trajectory. We assume that the human only causes positional uncertainties, without affecting the desired roll, pitch, and yaw of the reference trajectory. Therefore, the last three rows of D are kept as zeros. The vector $h = [h^x, h^y, h^z]^\top \in \mathbb{R}^3$ represents the positional uncertainty created by the human up to time step t, which can be directly observed by the robot and should be added to the nominal trajectory.

B. Human Uncertainty Modelling

In our approach, we model human behavior and preference-based uncertainty as $\varpi(t) \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$. $\boldsymbol{\mu} \in \mathbb{R}^3$ is the tendency or the average expected positional uncertainty and the covariance matrix $\Sigma \in \mathbb{R}^{3\times 3}$, oriented in the x, y, z directions, captures the variability and individual differences in human behavior, providing insight into how uncertainties might deviate from the mean in three-dimensional space. To quantify such uncertainty, suppose a human trajectory can be measured as $\tilde{r}(t)$. Comparing it with the nominal trajectory, the positional difference is defined as

$$\varpi(t) = D(\tilde{r}(t) - r(t)) \in \mathbb{R}^3 \tag{2}$$

It is easy to formulate the following minimization problem to estimate $\mathbb{E}(\varpi) = \boldsymbol{\mu}$,

$$\min_{\boldsymbol{\mu}} \sum_{t=0}^{T} \|D(\tilde{r}(t) - r(t)) - \boldsymbol{\mu}\|^2 \tag{3}$$

To further quantify Σ, assuming the solution of Equation (3) to be $\hat{\boldsymbol{\mu}}$, then we can estimate the covariance matrix as,

$$\hat{\Sigma} = \frac{1}{T-1} \sum_{t=0}^{T} \left[(D(\tilde{r}(t) - r(t)) - \hat{\boldsymbol{\mu}})(D(\tilde{r}(t) - r(t)) - \hat{\boldsymbol{\mu}})^\top \right] \tag{4}$$

Here, the factor of $\frac{1}{T-1}$ is due to the concept of Bessel's correction [26]. Now, to track the trajectory with such uncertainty, i.e., Equation (1), in a receding horizon manner, we introduce the dynamics of the mobile manipulator in the following subsection.

C. 8-DOF Robotic Arm Dynamics

The Fetch robot features a 7-DOF (degrees of freedom) robotic arm built on a robotic base [27]. We treat the base's heading angle as an additional degree of freedom for the arm. With this adjustment, we equivalently have an 8-DOF robotic arm, as depicted in Figure 2 [28]. To get the state transition of the 8-DOF arm, we use forward kinematics using Jacobian

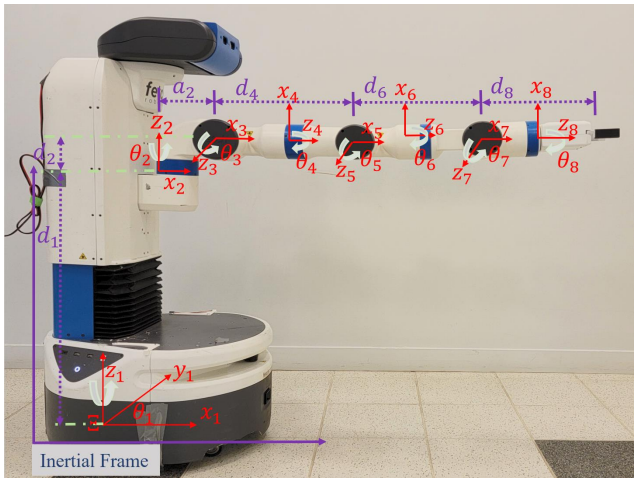

Fig. 2: Joint axes configurations of the 8-DOF configuration with DH parameters in body frame Ξ within the inertial frame.

matrix $\mathbf{J}(\cdot) \in \mathbb{R}^{6\times 8}$ derived based on the DH-parameters [29] of the robotic arm configured in Fig. 2:

$$\begin{aligned} s(k+1) &= f(\theta(k) + \tau u(k)) \\ &\approx s(k) + \tau \mathbf{J}(\theta(k))u(k) \\ &= s(k) + B(\theta(k))u(k) \end{aligned} \tag{5}$$

where s represents the end-effector pose, $\theta = [\theta_1,\ \theta_2,\ \theta_3, \cdots, \theta_8]^\top \in \mathbb{R}^8$ represents the joint angles of the 8-DOF robotic arm and $u = [v_a, v_b, \dot{\theta}_2, \cdots,\ \dot{\theta}_8]^\top \in \mathbb{R}^9$ represents the corresponding joint angle velocities. Particularly, since θ_1 is the rotation of the base, its actuation is based on a differential signal on its wheels v_a and v_b. $\mathbf{J}(\theta(k)) = \frac{\partial f(\theta)}{\partial \theta} \in \mathbb{R}^{6\times 9}$ is the Jacobian matrix. $B(\theta(k)) = \tau \mathbf{J}(\theta(k))$ is the input matrix which depends on the joint angle combinations $\theta(k)$.

D. Problem Formulation

Our problem of interest is to track the trajectory (1) by the end-effector of the robot with the 8-DOF robotic arm dynamics (5). We solve this by introducing a unique MPC formulation, where we add an extra variable ϖ that represents human uncertainties as discussed in section II-B and formulated as Equation (6). We solve this equation for optimal control input sequence over the planning horizon. Note that, we do not consider uncertainties coming from the robot dynamics, rather our formulation considers human-induced uncertainties.

Our proposed human uncertainty-aware MPC tracking can be formulated as follows:

$$\begin{aligned} &\min_{\boldsymbol{u}(0:H-1)} \ \mathcal{J}(\boldsymbol{u}(0:H-1)) \\ &\triangleq \mathbb{E}_{\varpi}\left[\sum_{k=0}^{H} \left[s(k)-\tilde{r}(k)\right]^\top Q \left[s(k)-\tilde{r}(k)\right]\right] \\ &\quad + \sum_{k=0}^{H-1} u(k)^\top R u(k) \\ &\text{s.t.} \quad s(k+1) = s(k) + B(\theta)u(k), \quad s(0) = s_0 \\ &\qquad \tilde{r}(k) = r(k) + D(h + \sum_{\tau=0}^{k} \varpi(\tau)), \quad \varpi \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma) \end{aligned} \tag{6}$$

where $\boldsymbol{u}(0:H-1) = \{u(0), \cdots, u(H-1)\}$, $Q \in \mathbb{R}^{6\times 6} \succeq 0$, $R \in \mathbb{R}^{9\times 9} \succ 0$ are the weighting matrices for tracking and input costs, respectively. s_0 is the starting end-effector pose of a planning horizon. Without considering human uncertainty, i.e., $\boldsymbol{\mu} = 0$ and $\Sigma = 0$, problem (6) becomes a regular MPC tracking problem.

III. Main Result

While addressing the problem (6), it is obvious that the optimal control input sequence $\boldsymbol{u}(0:H-1)$ depends on the input matrix $B(\theta)$. This section presents a solution to obtain the optimal control input sequence $\boldsymbol{u}(0:H-1)$. Our approach is summarized in Algorithm 1. We define the following error dynamics for (5) by subtracting $\tilde{r}(k+1)$ from both sides of the equation:

$$\begin{aligned} e(k+1) &= e(k) + B(\theta(k))u(k) + \tilde{r}(k) - \tilde{r}(k+1) \\ &= e(k) + B(\theta(k))u(k) + r(k) - r(k+1) \\ &\quad - D\varpi(k+1) \end{aligned} \tag{7}$$

with $e(k) = s(k) - \tilde{r}(k)$ being the tracking error. We assume that the optimal cost-to-go function follows:

$$\mathcal{J}^*(e(k), k) = e(k)^\top P(k) e(k) + 2e(k)^\top p(k) + c(k) \tag{8}$$

where, $P(k) \in \mathbb{R}^{6\times 6}, p(k) \in \mathbb{R}^6, c(k) \in \mathbb{R}$, are unknown matrices, vectors, and scalars to be determined. The optimal control input sequence u^* and Theorem (1) can verify the assumed solution.

Theorem 1. *Assuming the optimal solution $\boldsymbol{u}^*$ of (6) yields an optimal cost $\mathcal{J}^*$ with the form of (8). Then $P(k)$, $p(k)$, and $c(k)$ can be computed by the following ua-DARE:*

$$P(k) = Q + P(k+1) - P(k+1)BMP(k+1) \tag{9}$$

$$\begin{aligned} p(k) &= p(k+1) + P(k+1)(r(k) - r(k+1) - D\boldsymbol{\mu}) \\ &\quad - P(k+1)BMP(k+1)(r(k) - r(k+1) - D\boldsymbol{\mu}) \\ &\quad - P(k+1)BMp(k+1) \end{aligned} \tag{10}$$

$$\begin{aligned} c(k) &= c(k+1) + \|r(k) - r(k+1)\|_{P(k+1)} \\ &\quad + Tr(\Sigma D^\top P(k+1) D) \\ &\quad - 2(r(k) - r(k+1))^\top P(k+1) D\boldsymbol{\mu}) \\ &\quad - \|P(k+1)(r(k) - r(k+1) - D\boldsymbol{\mu}) + p(k+1))\|^2_{BM} \\ &\quad + 2(r(k) - r(k+1) - D\boldsymbol{\mu})^\top p(k+1) \end{aligned} \tag{11}$$

with $M = (R + B^\top P(k+1)B)^{-1}B^\top$*, terminal conditions:*

$$P(H) = Q, \quad p(H) = 0, \quad c(H) = 0 \tag{12}$$

Based on Theorem (1), for each time-step k, the optimal control inputs follow:

$$u^*(k) = -M(P(k+1)(e(k)+r(k)-r(k+1)-D\boldsymbol{\mu})+p(k+1)) \tag{13}$$

Proof. For derivation simplicity, let $B = B(\theta)$. Integrating Equation (7) into the optimal cost Equation (8),

$$\begin{aligned}
&\mathcal{J}^\star(e(k),k) = \min_{u(k)} \mathcal{J}(e(k),k))\\
&= \min_{u(k)}[\mathbb{E}_\varpi[\|e(k)\|_Q^2 + \|u(k)\|_R^2 + \mathcal{J}^\star(e(k+1),k+1)]]\\
&= \min_{u(k)}[\mathbb{E}_\varpi[\|e(k)\|_Q^2 + \|u(k))\|_R^2\\
&+ \|e(k) + Bu(k) + r(k) - r(k+1)\|^2_{P(k+1)}\\
&+ \text{Tr}(\Sigma D^\top P(k+1)D)\\
&- 2(e(k) + Bu(k) + r(k) - r(k+1) - D\boldsymbol{\mu})^\top P(k+1)D\boldsymbol{\mu}\\
&+ 2(e(k) + Bu(k) + r(k) - r(k+1) - D\boldsymbol{\mu})^\top p(k+1)\\
&+ c(k+1)]]
\end{aligned} \tag{14}$$

where $\mathbb{E}(\varpi(k+1)) = \boldsymbol{\mu}$. The optimality condition $\frac{\partial \mathcal{J}^*(e(k),k)}{\partial u(k)} = 0$ yields

$$\begin{aligned}
Ru(k)+B^\top P(k+1)(e(k)+Bu(k)+r(k)-r(k+1)-D\boldsymbol{\mu})\\
+ B^\top p(k+1) = 0
\end{aligned} \tag{15}$$

Thus, for each time-step k, the control inputs follows:

$$u^*(k) = -M(P(k+1)(e(k)+r(k)-r(k+1) - D\boldsymbol{\mu})+p(k+1)) \tag{16}$$

with M defined in Theorem (1). Bringing this back to (14) yields

$$\begin{aligned}
&\mathcal{J}^*(e(k),k)\\
&= \|e(k)\|^2_{(Q+P(k+1))} + \|u(k)\|_R^2\\
&+ 2e(k)^\top P(k+1)Bu(k)\\
&+ 2e(k)^\top P(k+1)(r(k) - r(k+1) - D\boldsymbol{\mu})\\
&+ 2e(k)^\top p(k+1) + 2u(k)^\top B^\top p(k+1)\\
&+ u(k)^\top B^\top P(k+1)Bu(k)\\
&+ 2u(k)^\top B^\top P(k+1)(r(k) - r(k+1) - D\boldsymbol{\mu})\\
&+ \|r(k) - r(k+1)\|_{P(k+1)} + \text{Tr}(\Sigma D^\top P(k+1)D)\\
&- 2(r(k) - r(k+1))^\top P(k+1)D\boldsymbol{\mu})\\
&+ 2(r(k) - r(k+1) - D\boldsymbol{\mu})^\top p(k+1) + c(k+1)
\end{aligned} \tag{17}$$

Reusing condition (15) by left multiplying $u(k)^\top$, using $u(k)$ from (16), and bringing them into (17) yields:

$$\begin{aligned}
&\mathcal{J}^*(e(k),k)\\
&= \|e(k)\|^2_{[Q+P(k+1)-P(k+1)BMP(k+1)]}\\
&+ 2e(k)^\top \big[p(k+1) - P(k+1)B(Mp(k+1)\\
&- P(k+1)BMP(k+1)(r(k) - r(k+1) - D\boldsymbol{\mu})\\
&+ P(k+1)(r(k) - r(k+1) - D\boldsymbol{\mu})\big]\\
&+ c(k+1) + \|r(k) - r(k+1)\|_{P(k+1)}\\
&+ \text{Tr}(\Sigma D^\top P(k+1)D)\\
&- 2(r(k) - r(k+1))^\top P(k+1)D\boldsymbol{\mu})\\
&- \|P(k+1)(r(k) - r(k+1) - D\boldsymbol{\mu}) + p(k+1))\|^2_{BM}\\
&+ 2(r(k) - r(k+1) - D\boldsymbol{\mu})^\top p(k+1)
\end{aligned} \tag{18}$$

Comparing (18) with (8), we have the DARE for $P(k)$, $p(k)$, and $c(k)$. The terminal conditions are obtained by considering $\mathcal{J}^*(e(H), k = H)$ for (6), where $P(H) = Q$ □

Algorithm 1: Biased Human-Uncertainty-Aware MPC Tracking

1. **Input** Nominal trajectory r_t; estimate human uncertainty parameters$\boldsymbol{\mu}$ and Σ; previous human uncertainty h
2. Compute $\tilde{r}(k = 0 : H)$ based on equation (1).
3. Get Jacobian Matrix, $\mathbf{J}(\theta) = \frac{\partial f(\theta)}{\partial \theta}$
4. Compute matrix $B(\theta)$ with (5).
5. Solve the MPC by computing solutions for the ua-DARE in *theorem 1*.
6. Use the results in step 5 in equation (13) to compute the optimal control input sequence $\boldsymbol{u}^*(0 : H-1)$
7. Update the robot's state.
8. Repeat from Step 2

IV. Experiments

This section validates the proposed human uncertainty-aware MPC tracking algorithm using simulated experiments. We build the model of a Fetch robot, shown in Fig. 2, using methods described in Sec. II with DH parameters [30] for link lengths, offset values, and rotation twist angles.

To randomly generate a human trajectory, then estimate its uncertainty as $\varpi(k) \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, where $\boldsymbol{\mu} = [0.0052,\ 0.0047,\ 0.0021]^\top$ (meters), $\Sigma = \text{diag}(0.014,\ 0.025,\ 0.010)$ (meters). It characterizes a human who tends to create more uncertainties in the y-axis than x, z-axes. For comparison purposes, we articulately increase the magnitude of the covariance matrix by a factor of $q \in \{1,\ 2,\ 4\}$ with $q = 1$ representing the base case, and 2 and 4 increases the strength of the uncertainties.

We validated our proposed algorithm 1 by performing tracking with three different trajectories that are subject to human uncertainties. The decomposed x, y, and z axes for each trajectory are visualized in Fig. 3. Each trajectory is

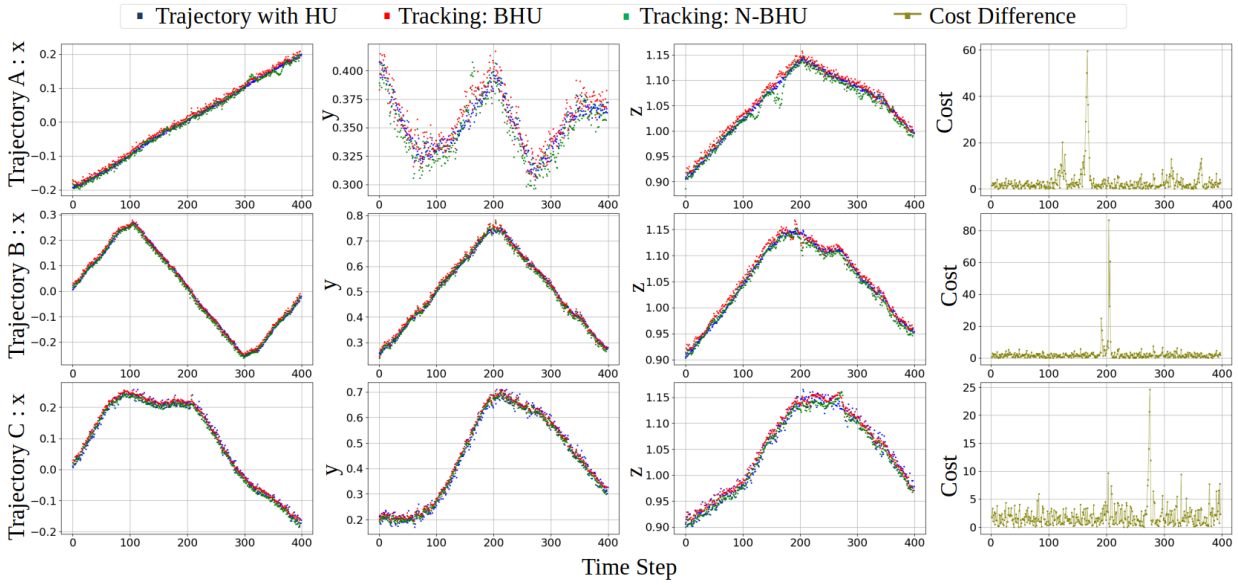

Fig. 3: Comparison of Tracking performance and Cost for each trajectories with $Q = 10000 \cdot I_6$, $R = I_9$, $q = 1$.

TABLE I: Total Cost Comparison of Our Approach with Baseline Algorithm

Traj	Q	q	BHU	N-BHU
A	$10000I_6$	1	**587.945**	1728.750
		2	**1458.788**	2682.087
		4	**2878.625**	4097.953
	$5000I_6$	1	**377.683**	913.213
		2	**842.449**	1482.981
		4	**1462.193**	2138.699
B	$10000I_6$	1	**1289.893**	2010.958
		2	**2145.533**	3086.132
		4	**3712.969**	4912.039
	$5000I_6$	1	**963.029**	1544.563
		2	**1372.532**	2036.649
		4	**2125.284**	2893.756
C	$10000I_6$	1	**736.422**	1294.321
		2	**1649.417**	2430.971
		4	**3243.415**	4311.028
	$5000I_6$	1	**547.486**	944.716
		2	**1009.714**	1501.895
		4	**1797.617**	2420.096

discretized into 400 points with a time step $\tau = 0.1$ (sec). Throughout the simulation, we use real physics to update the robot's states and calculate the end-effector pose with forward kinematics. The parameters are chosen as $H = 5$, $R = I_9$, We set either $Q = 10000 \cdot I_6$ or $Q = 5000 \cdot I_6$

We repeat the MPC planning every step, compute the optimal control inputs for the planning horizon, and update the robot's state with the first control input. To evaluate the control performance, we define the total cost

$$\mathcal{C}_{\text{total}} = \sum_{t=1}^{400} e(t)^\top Q e(t) + u(t)^\top R u(t),$$

taking into account the robot's end effector tracking error cost and control input cost.

Fig. 3 shows the tracking visualization for each trajectory. we compare our proposed approach, MPC tracking with consideration of biased human uncertainty (BHU), with a baseline algorithm that does not consider any human uncertainty while tracking (N-BHU). We observe that the dots representing tracking with BHU (Red) are comparatively closer to the dots of nominal trajectory with human uncertainties (Blue) than the dots of tracking with N-BHU (Green). As we have a relatively larger value for Q, we observe that tracking dots along all three x, y, and z-axes for every trajectory align closely with the trajectory to be followed.

Table I shows the performance in terms of cost for all three trajectories with different Q and q values. The comparison shows that our approach, BHU, outperforms N-BHU. Additionally, we observe a gradual increase in total cost corresponding to the increase in q. This phenomenon occurs as the robot attempts to counteract human uncertainties. However, its ability to fully mitigate these uncertainties is limited due to the small time step, τ. Consequently, the tracking error increases, contributing to the increment in total cost.

The cost plots, also illustrated in Figure 3, show the variation in cost at each time step for all three trajectories. It is worth mentioning that in almost every step, BHU has a lower cost than N-BHU. In some time steps, the cost difference is noticeably higher than in other time steps. This observation can be related to the trajectory itself, meaning that, while navigating complex trajectory segments, BHU is more cost-efficient. This consistent cost advantage translates into the superior overall performance of our proposed method, as detailed in Table I.

V. Conclusions and Future Works

We developed a novel control algorithm for human-robot co-manipulation tasks utilizing an 8-DOF robotic arm. We explicitly model human uncertainty as a normal distribution

with non-zero mean and covariance. We insert that as an extra variable in our formulation, leading to an uncertainty-aware MPC formulation. We solved for the optimal control law from this formulation by computing uncertainty-aware DARE theoretically. We later validated our approach by simulating using a Fetch robot model. Our proposed approach outperforms a baseline algorithm that does not consider human uncertainty in tracking accuracy and energy consumption. Future work will include implementing our algorithm in the Gazebo simulator and on a physical Fetch robot platform.

REFERENCES

[1] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 882–893, 2015.

[2] S. Robla-Gómez, V. M. Becerra, J. R. Llata, E. Gonzalez-Sarabia, C. Torre-Ferrero, and J. Perez-Oria, "Working together: A review on safe human-robot collaboration in industrial environments," *Ieee Access*, vol. 5, pp. 26754–26773, 2017.

[3] C. Brosque, E. Galbally, O. Khatib, and M. Fischer, "Human-robot collaboration in construction: Opportunities and challenges," in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pp. 1–8, IEEE, 2020.

[4] L. Rozo, D. Bruno, S. Calinon, and D. G. Caldwell, "Learning optimal controllers in human-robot cooperative transportation tasks with position and force constraints," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 1024–1030, IEEE, 2015.

[5] A.-N. Sharkawy and P. N. Koustoumpardis, "Human–robot interaction: A review and analysis on variable admittance control, safety, and perspectives," *Machines*, vol. 10, no. 7, p. 591, 2022.

[6] S. Li, H. Wang, and S. Zhang, "Human-robot collaborative manipulation with the suppression of human-caused disturbance," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 4, p. 73, 2021.

[7] S. Erhart, D. Sieber, and S. Hirche, "An impedance-based control architecture for multi-robot cooperative dual-arm mobile manipulation," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 315–322, IEEE, 2013.

[8] C. J. Ostafew, A. P. Schoellig, T. D. Barfoot, and J. Collier, "Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking," *Journal of Field Robotics*, vol. 33, no. 1, pp. 133–152, 2016.

[9] X. Xiao, T. Zhang, K. M. Choromanski, T.-W. E. Lee, A. Francis, J. Varley, S. Tu, S. Singh, P. Xu, F. Xia, S. M. Persson, L. Takayama, R. Frostig, J. Tan, C. Parada, and V. Sindhwani, "Learning model predictive controllers with real-time attention for real-world navigation," in *Conference on robot learning*, PMLR, 2022.

[10] X. Xiao, Z. Wang, Z. Xu, B. Liu, G. Warnell, G. Dhamankar, A. Nair, and P. Stone, "Appl: Adaptive planner parameter learning," *Robotics and Autonomous Systems*, vol. 154, p. 104132, 2022.

[11] X. Xiao, B. Liu, G. Warnell, J. Fink, and P. Stone, "Appld: Adaptive planner parameter learning from demonstration," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4541–4547, 2020.

[12] Y. Zhou, X. Wang, and W. Jin, "D3g: Learning multi-robot coordination from demonstrations," *arXiv preprint arXiv:2207.08892*, 2022.

[13] H. Taghavifar and S. Rakheja, "A novel terramechanics-based path-tracking control of terrain-based wheeled robot vehicle with matched-mismatched uncertainties," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 67–77, 2019.

[14] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for uav tracking," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pp. 445–461, Springer, 2016.

[15] J. Baek, S. Cho, and S. Han, "Practical time-delay control with adaptive gains for trajectory tracking of robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5682–5692, 2017.

[16] S. Xu, Y. Ou, J. Duan, X. Wu, W. Feng, and M. Liu, "Robot trajectory tracking control using learning from demonstration method," *Neurocomputing*, vol. 338, pp. 249–261, 2019.

[17] T. Faulwasser, T. Weber, P. Zometa, and R. Findeisen, "Implementation of nonlinear model predictive path-following control for an industrial robot," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 4, pp. 1505–1511, 2016.

[18] J. J. Quiroz-Omaña and B. V. Adorno, "Whole-body kinematic control of nonholonomic mobile manipulators using linear programming," *Journal of Intelligent & Robotic Systems*, vol. 91, pp. 263–278, 2018.

[19] M. Giftthaler, F. Farshidian, T. Sandy, L. Stadelmann, and J. Buchli, "Efficient kinematic planning for mobile manipulators with non-holonomic constraints using optimal control," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3411–3417, IEEE, 2017.

[20] M. Osman, M. W. Mehrez, S. Yang, S. Jeon, and W. Melek, "End-effector stabilization of a 10-dof mobile manipulator using nonlinear model predictive control," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9772–9777, 2020.

[21] M. Zanon and S. Gros, "Safe reinforcement learning using robust mpc," *IEEE Transactions on Automatic Control*, vol. 66, no. 8, pp. 3638–3652, 2020.

[22] M. B. Saltık, L. Özkan, J. H. Ludlage, S. Weiland, and P. M. Van den Hof, "An outlook on robust model predictive control algorithms: Reflections on performance and computational aspects," *Journal of Process Control*, vol. 61, pp. 77–102, 2018.

[23] F. Cursi, V. Modugno, L. Lanari, G. Oriolo, and P. Kormushev, "Bayesian neural network modeling and hierarchical mpc for a tendon-driven surgical robot with uncertainty minimization," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2642–2649, 2021.

[24] G. P. Incremona, A. Ferrara, and L. Magni, "Mpc for robot manipulators with integral sliding modes generation," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 3, pp. 1299–1307, 2017.

[25] T. M. Howard, C. J. Green, and A. Kelly, "Receding horizon model-predictive control for mobile robot navigation of intricate paths," in *Field and Service Robotics: Results of the 7th International Conference*, pp. 69–78, Springer, 2010.

[26] H. Hernandez, "Probability distribution and bias of the sample standard deviation," *ForsChem Research Reports*, vol. 8, pp. 2023–02, 2023.

[27] K. Kreutz-Delgado, M. Long, and H. Seraji, "Kinematic analysis of 7-dof manipulators," *The International journal of robotics research*, vol. 11, no. 5, pp. 469–481, 1992.

[28] J. Sudharsan and L. Karunamoorthy, "Path planning and co-simulation control of 8 dof anthropomorphic robotic arm," *International journal of simulation modelling*, vol. 15, no. 2, pp. 302–312, 2016.

[29] S. Kucuk and Z. Bingul, *Robot kinematics: Forward and inverse kinematics*. INTECH Open Access Publisher London, UK, 2006.

[30] F. R. Inc., "Fetch & freight manual," 2014.