Robust Dexterous Manipulation under Object Dynamics Uncertainties

Yongxiang Fan[1], Liting Sun[1], Minghui Zheng[1], Wei Gao[2], Masayoshi Tomizuka[1]

***Abstract*— Dexterous manipulation has broad applications in assembly lines, warehouses and agriculture. To perform broad-scale manipulation tasks, it is desired that a multi-fingered robotic hand can robustly manipulate objects without knowing the exact objects dynamics (i.e. mass and inertia) in advance. However, realizing robust manipulation is challenging due to the complex contact dynamics, the nonlinearities of the system, and the potential sliding during manipulation. In this paper, a dual-stage grasp controller is proposed to handle these challenges. In the first stage, feedback linearization is utilized to linearize the nonlinear uncertain system. Considering the structures of uncertainties, a robust controller is designed for such a linearized system to obtain the desired Cartesian force on the object. In the second stage, a manipulation controller regulates the contact force based on the Cartesian force from the first stage. The dual-stage grasp controller is able to realize robust manipulation without contact modeling, prevent the slippage, and withstand 40% mass and 50% inertia uncertainties. Moreover, it does not require velocity measurement or 3D/6D tactile sensor. Simulation results on Mujoco verify the efficacy of the proposed method. The simulation video is available at [1].**

I. INTRODUCTION

Dexterous manipulation is essential for manipulators to execute complicated tasks, such as circuit assembly, commodity organizing and fruit harvesting. To perform broad-scale manipulations, a robotic hand usually has to manipulate objects with various shapes and dynamics properties such as mass and inertia. In many applications, accurate models of the object dynamics are unknown in advance. They are estimated from 3D sensing, as well as prior knowledge such as density and statistical models. Consequently, uncertainties are introduced into the system. It is difficult to deal with such uncertainties in dexterous manipulation. First, the object is not directly controlled by actuators. Alternatively, energy is transferred from the fingertips to the object through unknown contact dynamics. Second, the robotic hand for dexterous manipulation can be a high degree-of-freedom (DOF) nonlinear system and can not be directly written into linear time-invariant (LTI) or linear parametric-varying (LPV) form. Moreover, the potential sliding between the fingertips and the object would degrade the object motion tracking performance.

As a result, robust dexterous manipulation for nonlinear systems has received significant attention. A robust controller for contact uncertainties was proposed in [2]. The controller is designed for a LTI system linearized around an equilibrium point. A force-position controller using 6D tactile sensors was implemented to realize adaptive grasping [3]. Nonlinearities were ignored due to its constant-pose grasping property. In order to consider parameter variations caused by nonlinearities, a LPV control with smooth scheduling was applied in [4], assuming that the nonlinearities can be approximated through linear varying parameters. To deal with dynamics uncertainties, a disturbance observer (DOB) was proposed in [5] for tracking control. The nonlinearities and parameter uncertainties are lumped into a disturbance term. It assumes full state feedback, while in dexterous hand, the velocity feedback is difficult due to the size constraints and cost issue. Feedback linearization was applied to control an unmanned aerial vehicle [6]. A linear state observer and a DOB are combined to observe the state and the lumped disturbance. Similar to [5], the structures of the parameter uncertainties are ignored, and the linear state observer assumes a perfect model for state estimation.

This paper proposes a dual-stage grasp controller for dexterous manipulation under object dynamics uncertainties and external disturbances. Distinctive features of this paper include: 1) The nonlinearities are reduced by feedback linearization on a nominal model. Compared with LPV that assumes linear variations of parameters, the proposed method is more computationally efficient for broad-scale manipulations. 2) The robust controller is formulated as a μ-synthesis problem, and the structures of the uncertainties are considered by descriptor form, instead of treating uncertainties as a lumped disturbance, which results in information loss and a larger disturbance to resist. 3) By the dual-stage formulation, the complicated contact modeling is bypassed, and the contact force is regulated and the slippage is prevented. 4) Moreover, the dual-stage grasp controller does not require expensive 3D/6D tactile sensors or velocity measurements of objects/joints.

The remaining of this paper is organized as follows. Section II introduces the dual-stage grasp controller framework. Section III describes the system dynamics and its combination with the feedback linearization. The robust controller and the manipulation controller are presented in Section IV and Section V, respectively. Section VI shows the simulation results. Section VII concludes the paper.

II. DUAL-STAGE GRASP CONTROLLER FRAMEWORK

Figure 1 shows the proposed framework of the dual-stage grasp controller. In this figure, r, y, n and e denote the reference pose, the actual pose, the measurement noise

[1]Yongxiang Fan, Liting Sun, Minghui Zheng, and Masayoshi Tomizuka are with Department of Mechanical Engineering, University of California, Berkeley, Berkeley, CA 94720, USA yongxiang_fan, litingsun, minghuizheng, tomizuka@berkeley.edu

[2]Wei Gao is with School of Aerospace, Tsinghua University, Beijing, 100084, P. R. China gaow13@mails.tsinghua.edu.cn

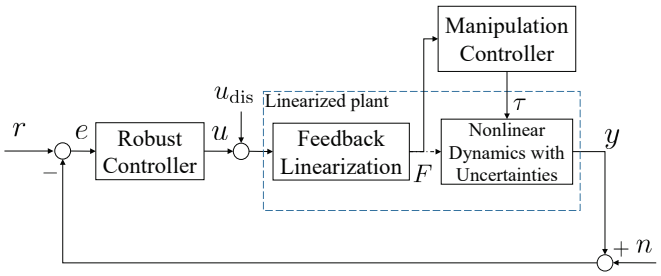

Fig. 1: The general framework of the proposed dual-stage grasp controller.

and the pose error of the object, respectively. The signal u denotes the control input to the linearized plant. The signal u_{dis} is the external disturbance to the plant. F is the desired Cartesian space force on the object. The signal τ is the torque command to the hand in order to realize F. The objective is to: 1) track the desired pose r of the object, 2) be robust to object dynamics uncertainties (i.e. mass and inertia uncertainties) and external disturbances u_{dis}, and 3) realize firm contact without violating the friction cone constraints.

The dual-stage grasp controller consists of a robust controller and a manipulation controller, as shown in Fig. 1. The robust controller takes e as input, and generate F to the object. It applies on a linearized nominal plant with nonlinear uncertainties. The linearized nominal plant is obtained by feedback linearization on a nonlinear dynamics. The F obtained from feedback linearization and robust controller is converted into τ by the manipulation controller.

In robust controller design, the feedback linearization is directly connected to the nonlinear dynamics by the assumption that the actual force on the object after executing τ is close to F. The gap between these two forces is treated as part of u_{dis}.

III. MODELING OF UNCERTAIN DYNAMICAL SYSTEMS

A. State-Space Dynamics

The hand and object dynamics are described by:

$$\begin{aligned} &M_h(q)\ddot{q} + C_h(q,\dot{q})\dot{q} + N_h(q,\dot{q}) + J_h^T(q,x_o)f_c = \tau \\ &M_o(x_o)\ddot{x}_o + C_o(x_o,\dot{x}_o)\dot{x}_o + N_o = G(q,x_o)f_c \end{aligned} \tag{1}$$

where $M_{h/o}, C_{h/o}$ and $N_{h/o}$ are inertia matrices, Coriolis matrices and gravities for the hand/object. $q, \dot{q}$ and $\ddot{q} \in \mathbb{R}^{n_q}$ are joint angle, velocity and acceleration, with n_q as the total DOFs of the hand. $x_o, \dot{x}_o$ and $\ddot{x}_o \in \mathbb{R}^{n_x}$ are a local parameterization of object position, velocity and acceleration, where n_x is the dimension of the pose of the object, with $n_x = 6$ for 3D manipulation ($n_x = 3$ for 2D manipulation). $f_c \in \mathbb{R}^{d_c n_c}$ and $\tau \in \mathbb{R}^{n_q}$ are contact force vector and joint torque vector, where d_c is the dimension of each contact, and n_c is the contact number. $J_h \in \mathbb{R}^{(d_c n_c)\times n_q}$ is the hand Jacobian and $G \in \mathbb{R}^{n_x \times (d_c n_c)}$ is the grasp map [7].

If the contacts are fixed w.r.t both object and fingertips, then

$$J_h(q,x_o)\dot{q} = G^T(q,x_o)\dot{x}_o \tag{2}$$

holds. Equation 2 assumes the contact forces remain in the friction cone.

The object and hand dynamics in (1) can be connected by (2):

$$M(q,x_o)\ddot{x}_o + C(q,\dot{q},x_o,\dot{x}_o)\dot{x}_o + N(q,x_o) = GJ_h^{-T}\tau \tag{3}$$

where:

$$\begin{aligned} M &= M_o + GJ_h^{-T}M_hJ_h^{-1}G^T \\ C &= C_o + GJ_h^{-T}C_hJ_h^{-1}G^T + GJ_h^{-T}M_h\frac{d(J_h^{-1}G^T)}{dt} \\ N &= N_o + GJ_h^{-T}N_h \end{aligned} \tag{4}$$

In some applications such as fruit harvesting, only the rough values of the mass m_o and the inertia $\mathcal{I}_o$ of the object can be estimated. Therefore, M_o, C_o, N_o cannot be exactly known and would exhibit some uncertainties. Suppose that the inertia, Coriolis and gravity can be represented as:

$$M = \bar{M} + \tilde{M}_o, \quad C = \bar{C} + \tilde{C}_o, \quad N = \bar{N} + \tilde{N}_o \tag{5}$$

with nominal values:

$$\begin{aligned} \bar{M} &= \bar{M}_o + GJ_h^{-T}M_hJ_h^{-1}G^T \\ \bar{C} &= \bar{C}_o + GJ_h^{-T}C_hJ_h^{-1}G^T + GJ_h^{-T}M_h\frac{d(J_h^{-1}G^T)}{dt} \\ \bar{N} &= \bar{N}_o + GJ_h^{-T}N_h \end{aligned}$$

where $\bar{M}_o, \bar{C}_o, \bar{N}_o$ are nominal object inertia, Coriolis, gravity, and $\tilde{M}_o, \tilde{C}_o, \tilde{N}_o$ are corresponding uncertainties. The torque command τ can be related to the object-centered force F:

$$\tau = J_h^T(G^\dagger F + N_G\lambda) \tag{6}$$

where N_G is the matrix composed by the basis of the null space of G, and λ is a free variable to control the magnitude and direction of the contact force.

The state space equation can be derived by plugging (5) and (6) into (3):

$$\begin{aligned} &\left(\underbrace{\begin{bmatrix}\mathbb{I} & \mathbb{O}\\ \mathbb{O} & \bar{M}\end{bmatrix}}_{\bar{M}_{\text{aug}}} + \underbrace{\begin{bmatrix}\mathbb{O} & \mathbb{O}\\ \mathbb{O} & \tilde{M}_o\end{bmatrix}}_{\tilde{M}_{\text{aug}}}\right)\underbrace{\begin{bmatrix}\dot{x}_o\\ \ddot{x}_o\end{bmatrix}}_{\dot{x}} + \left(\underbrace{\begin{bmatrix}\mathbb{O}\\ \bar{N}\end{bmatrix}}_{\bar{N}_{\text{aug}}} + \underbrace{\begin{bmatrix}\mathbb{O}\\ \tilde{N}_o\end{bmatrix}}_{\tilde{N}_{\text{aug}}}\right) + \\ &\left(\underbrace{\begin{bmatrix}\mathbb{O} & -\mathbb{I}\\ \mathbb{O} & \bar{C}\end{bmatrix}}_{\bar{C}_{\text{aug}}} + \underbrace{\begin{bmatrix}\mathbb{O} & \mathbb{O}\\ \mathbb{O} & \tilde{C}_o\end{bmatrix}}_{\tilde{C}_{\text{aug}}}\right)\underbrace{\begin{bmatrix}x_o\\ \dot{x}_o\end{bmatrix}}_{x} = \underbrace{\begin{bmatrix}\mathbb{O}\\ \mathbb{I}\end{bmatrix}}_{B_F}F \end{aligned} \tag{7}$$

where $\mathbb{I}, \mathbb{O} \in \mathbb{R}^{n_x \times n_x}$. Equation (7) can be rewritten as:

$$\begin{aligned} \dot{x} = &-\bar{M}_{\text{aug}}^{-1}\bar{C}_{\text{aug}}x - \bar{M}_{\text{aug}}^{-1}\bar{N}_{\text{aug}} + \bar{M}_{\text{aug}}^{-1}B_FF - \\ &\bar{M}_{\text{aug}}^{-1}\tilde{M}_{\text{aug}}\dot{x} - \bar{M}_{\text{aug}}^{-1}\tilde{C}_{\text{aug}}x - \bar{M}_{\text{aug}}^{-1}\tilde{N}_{\text{aug}} \end{aligned} \tag{8}$$

In 3D manipulation, the parameters of (8) can be decomposed as:

$$-\bar{M}_{\text{aug}}^{-1}\tilde{M}_{\text{aug}} = L_1\boldsymbol{\Delta}R_1, \quad -\bar{M}_{\text{aug}}^{-1}\tilde{C}_{\text{aug}} = \sum_{j=1}^{2}L_{2j}\boldsymbol{\Delta}R_{2j} \tag{9}$$

when parameterizing the rotation matrix R of the object by Z-Y-X Euler angles E, with

$$\begin{aligned}
&L_1 = L_{21} = [0_{6\times6}; \bar{M}^{-1}] \times \text{diag}(I_{3\times3}, Q_E^T) \\
&\mathbf{\Delta} = \text{diag}(\delta_m I_{3\times3}, \delta_{\mathcal{I}_1}, ...\delta_{\mathcal{I}_3}) \quad \text{with} \|\mathbf{\Delta}\|_\infty \leq 1 \\
&R_1 = -\text{diag}(\Delta m I_{3\times3}, \Delta\mathcal{I}) \times [0_{6\times6}, \text{diag}(I_{3\times3}, Q_E)] \\
&R_{21} = -\text{diag}(\Delta m I_{3\times3}, \Delta\mathcal{I}) \times [0_{6\times6}, \text{diag}(0_{3\times3}, \dot{Q}_E)] \\
&L_{22} = [0_{6\times6}; \bar{M}^{-1}] \times \text{diag}(I_{3\times3}, R(Q_E\dot{E})\hat{}) \\
&R_{22} = -\text{diag}(\Delta m I_{3\times3}, \Delta\mathcal{I}) \times [0_{6\times6}, \text{diag}(0_{3\times3}, Q_E)]
\end{aligned}$$

where $\Delta m \in \mathbb{R}$ and $\Delta\mathcal{I} = \text{diag}(\Delta\mathcal{I}_1, ...\Delta\mathcal{I}_3)$ are the maximal mass and inertia uncertainties, $Q_E \in \mathbb{R}^{3\times3}$ is a Jacobian matrix from Euler angle rate $\dot{E}$ to angular velocity of the object in body frame, and $(\bullet)\hat{}$ denotes the matrix representation of cross product.

With (9), the uncertainty term $-M_{\text{aug}}^{-1}(\tilde{M}_{\text{aug}}\dot{x} + \tilde{C}_{\text{aug}}x)$ in (8) can be represented by:

$$\begin{aligned}
&L_1\mathbf{\Delta}\underbrace{(R_1\dot{x} + R_{21}x)}_{z_1} + L_{22}\mathbf{\Delta}\underbrace{R_{22}x}_{z_2} \\
&= L_1\underbrace{\mathbf{\Delta}z_1}_{w_1} + L_{22}\underbrace{\mathbf{\Delta}z_2}_{w_2} = L_1w_1 + L_{22}w_2
\end{aligned} \tag{10}$$

2D manipulation is used for illustration and comparison purpose. The Coriolis uncertainty can be eliminated by choosing the local parameterization as body frame translation and rotation angle. Thus L_{21}, R_{21} and L_{22}, R_{22} are removed, and

$$\begin{aligned}
&L_1 = [0_{3\times3}; \bar{M}^{-1}] \\
&\mathbf{\Delta} = \text{diag}(\delta_m I_{2\times2}, \delta_{\mathcal{I}_3}) \quad \text{with} \|\mathbf{\Delta}\|_\infty \leq 1 \\
&R_1 = [0_{3\times3}, -\text{diag}(\Delta m I_{2\times2}, \Delta\mathcal{I}_3)]
\end{aligned} \tag{11}$$

In general 3D manipulation, the Coriolis term is typically ignored due to the low-speed operation condition, as shown in [8].

The control input u is F, and the augmented gravity $\bar{N}_{\text{aug}}$ can be compensated by an additional control input $u_0 = \bar{N}_{\text{aug}}$. The gravity uncertainty $\tilde{N}_{\text{aug}}$ is considered as part of the disturbance u_{dis}. Then the uncertain state space model is represented as:

$$\begin{aligned}
\dot{x} &= \underbrace{-\bar{M}_{\text{aug}}^{-1}\bar{C}_{\text{aug}}}_{A}x + \underbrace{L_1}_{B_1}w_1 + \underbrace{\bar{M}_{\text{aug}}^{-1}B_F}_{B_2}(u - u_0 + u_{\text{dis}}) \\
z_1 &= C_1x + \underbrace{R_1L_1}_{D_{11}}w_1 + \underbrace{R_1\bar{M}_{\text{aug}}^{-1}B_F}_{D_{12}}(u - u_0 + u_{\text{dis}}) \\
y &= \underbrace{[I_{3\times3}, 0_{3\times3}]}_{C_2}x \qquad w_1 = \mathbf{\Delta}z_1
\end{aligned} \tag{12}$$

where $C_1 = -R_1\bar{M}_{\text{aug}}^{-1}\bar{C}_{\text{aug}}$. Equation (12) describes uncertainties by linear fractional transformation (LFT). Notice though the system is nonlinear, due to the state dependencies of the dynamics parameters.

B. Combining Feedback Linearization with Modeling

A challenge in robust control is the implementation on nonlinear systems. Although some extensions have been done for LPV systems, the application of robust control to a general nonlinear system is still challenging.

To reduce the influence of nonlinearities, feedback linearization is applied to linearize the model. More specifically, for (8), the command force may be:

$$F = \left(\bar{M}_{\text{aug}}^{-1}B_F\right)^\dagger \left[\bar{M}_{\text{aug}}^{-1}\bar{C}_{\text{aug}}x + \bar{M}_{\text{aug}}^{-1}\bar{N}_{\text{aug}} + B_Fu\right] \tag{13}$$

Notice $\left(\bar{M}_{\text{aug}}^{-1}B_F\right)\left(\bar{M}_{\text{aug}}^{-1}B_F\right)^\dagger = [0_{3\times3}, 0_{3\times3}; 0_{3\times3}, I_{3\times3}]$, rather than identity. Therefore, (8) becomes:

$$\begin{aligned}
&\dot{x} = Ax + B_Fu - \bar{M}_{\text{aug}}^{-1}\tilde{M}_{\text{aug}}\dot{x} - \bar{M}_{\text{aug}}^{-1}\tilde{C}_{\text{aug}}x - \bar{M}_{\text{aug}}^{-1}\tilde{N}_{\text{aug}} \\
&A = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}
\end{aligned} \tag{14}$$

Following the similar procedure as (9) and (11):

$$\begin{aligned}
&\dot{x} = Ax + L_1w + B_F(u + d) \\
&z_1 = R_1Ax + R_1L_1w_1 + R_1B_F(u + d) \\
&y = [I_{3\times3}, 0_{3\times3}]x \qquad w_1 = \mathbf{\Delta}z_1
\end{aligned} \tag{15}$$

The model would be a LTI system if there were no uncertainties. However, due to the parametric uncertainties, the feedback linearization based on nominal parameters will not be able to eliminate all the nonlinearities. Therefore, the remaining nonlinear uncertainties after feedback linearization are approximated by a LTI system evaluated around an equilibrium point. The resultant system has the same form as (15), except that the L_1 and R_1 are evaluated at the equilibrium point. The feasibility of this approximation is validated in Section VI.

The linearized plant described by (15) is controllable and observable. The robust controller will be designed based on this linearized plant.

IV. ROBUST CONTROLLER DESIGN

A. Design Scheme

Robust controller is to obtain desired Cartesian force of the object for motion tracking with guaranteed robust stability and performance. The generalized plant P_{general} that the robust controller will work on is shown in Fig. 2. G_{NL} and $\mathbf{\Delta}$ define an upper LFT w.r.t. $\mathbf{\Delta}$ (denoted as $F_u(G_{NL}, \mathbf{\Delta})$) to represent the nonlinear uncertain dynamics, as shown in red dash box. The feedback linearization described by $\alpha(x)+\beta(x)u$ is connected with the nonlinear uncertain plant to linearize the nominal model, as shown in the blue dash-dot box. Equation (15) is the combination of two boxes.

The inputs to the generalized plant P_{general} are $\{r, u_{\text{dis}}, n, u\}$, which denote the pose reference, the input disturbance, the noise and the control input to the plant. The outputs of the plant are $\{w_{\text{perf}}, w_u, e\}$, which denote the tracking performance, the action magnitude and the pose error. W_{perf} is to suppress tracking errors at different frequencies. W_u is to regulate the control input. W_{dis} is to shape the input disturbance. W_n is to shape the measurement noise. The structures of the weighting functions will be described in Section IV-B.

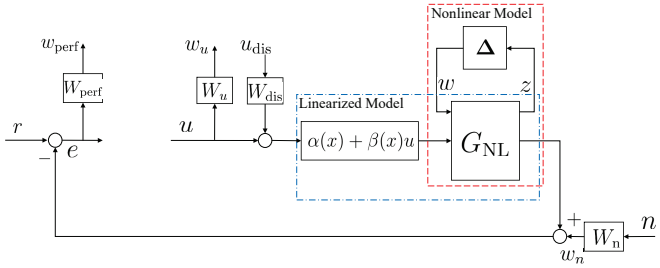

Fig. 2: Generalized plant with weighting functions.

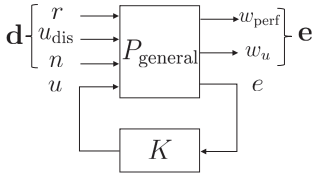

Fig. 3: Illustration of the closed-loop system.

The connection between the generalized plant P_{general} and the controller K is described by Fig. 3. P_{general} and K define a lower LFT w.r.t. K as $F_l(P_{\text{general}}, K)$, to denote the closed-loop system. The closed-loop system concatenates the inputs $\{r, u_{\text{dis}}, n\}$ as $\mathbf{d}$ and the outputs $\{w_{\text{perf}}, w_u\}$ as $\mathbf{e}$. The objective of the robust controller design is to synthesize K to keep $\mathbf{e}$ small for all reasonable $\mathbf{d}$. The small is in the sense of infinity norm, i.e.

$$\begin{aligned} &K = \underset{K}{\operatorname{argmin}} \|F_L(P_{\text{general}}, K)\|_\infty \\ &\text{with:} \\ &\mathbf{e} = F_L(P_{\text{general}}, K)\mathbf{d} \\ &\|F_L\|_\infty := \max_{\omega\in\mathbb{R}} \bar{\sigma}(F_L(j\omega)) \end{aligned} \tag{16}$$

The D-K iteration is applied to solve (16):

$$\min_K \inf_D \|DF_L(P_{\text{general}}, K)D^{-1}\|_\infty < 1 \tag{17}$$

Readers can refer [9] for implementation details.

The designed controller K will be used to calculate u based on the pose error e. Then the output of the controller is combined with feedback linearization (13) to obtain the desired Cartesian space force F for the object.

B. Design of Weighting Functions

The general form of a weighting function $W(s)$ in P_{general} can be written as:

$$W(s) = \text{diag}([a_1 W_{1,1}(s), a_2 W_{2,2}(s), a_3 W_{3,3}(s)])$$

where a_i is the weight to the i-th channel. $W_{i,i}$ is a SISO filter determined by high-frequency gain G_h, low-frequency gain G_l, cross-over frequency ω_c, and order n. In this section, the principle for choosing parameters are introduced. The concrete values for these parameters will be shown in Section VI.

1) Design of Performance Weighting Function W_{perf}: W_{perf} penalizes the tracking error caused by the general disturbance $\mathbf{d}$. High cross-over frequency w_c penalizes the disturbance with large bandwidth. With larger w_c, the system takes shorter time to settle down, and the desired force tends to change at higher frequencies. Consequently, the error oscillates at higher frequencies. The low-frequency gain G_l penalizes the magnitude of low-frequency disturbance. When G_l is very small, the low-frequency error is large, but the high-frequency error is small, which means that the system takes shorter time to converge. On the other hand, the high-frequency gain G_h penalizes the magnitude of high-frequency disturbances. Increasing G_h will speed-up the convergence. However, the oscillation will be enlarged, and the low-frequency performance will be compromised. As for n, large order n makes the system have more freedom to choose the best controller, while an excessive large n increases the order of the controller. The motivation for tuning a_i is the fact that the behavior in translation directions and rotation direction are usually different because of different parameter scales.

2) Design of Action Weighting Function W_u: The actions at different frequencies are penalized equally. This is a special case when $G_l = G_h$, which means the weighting function is a constant. Similar as before, large $G_{l/h}$ penalizes the magnitude of action. A larger $G_{l/h}$ results in more penalty to control effort, thus the force generated by controller is smaller. The smaller force can result in slower convergence speed and poor disturbance rejection. On the contrary, a small $G_{l/h}$ can make the controller generate excessive large force and damage the object. The influences of a_i and n can be reflected into changing $G_{l/h}$.

3) Design of Disturbance Weighting Function W_{dis}: The disturbance weighting function is used to shape the exogenous disturbance in the generalized plant P_{general}. The cross-over frequency ω_c indicates the shaping bandwidth. Generally, it enlarges the magnitude of low-frequency disturbances and shrink the magnitude of high-frequency disturbances. A large G_l will create a virtual disturbance with large low-frequency gain. Therefore, the controller would concentrate on reducing the low-frequency disturbance. In our implementation, the gravity is treated as static disturbance. Therefore, increasing G_l makes the actual system response faster by using the larger control effort. The high-frequency gain G_h specifies the shaping factor to high-frequency disturbances. A large value makes the system consider the disturbance rejection in full scale, and the low-frequency disturbance response will be compromised. Similar with W_{perf}, a_i specifies the scales of shaping for different channels, and n specifies the freedom of designing W_{dis}.

4) Design of Noise Weighting Function W_n: The W_n is designed to be a high-pass filter to shape the noise to the generalized plant P_{general}. The reasons are twofolds. First, the vision sensor used for object pose detection has high-frequency noises. Second, the manipulation controller used for desired force approximation is a low-pass filter, which may result in additional high-frequency approximation error.

The tuning of noise weighting is similar with disturbance weighting tuning.

V. MANIPULATION CONTROLLER DESIGN

The manipulation controller is utilized to generate torque commands for the hand to track the desired force generated by the robust controller in Section IV. The manipulation controller consists of a force optimizer, which searches desired contact force f on fingertips from the desired force F on the object, and a joint-level torque controller, which generates an appropriate joint torque vector τ to reproduce f. The force optimizer is formulated into a quadratic programming (QP):

$$\min_{\beta} \quad \alpha_1\|f\|_2^2 + \alpha_2\|f - f_{\text{prev}}\|_2^2 + \alpha_3\|\Psi\|_2^2 \tag{18a}$$

$$s.t. \quad \Psi = F - G(q, x_o)f \tag{18b}$$

$$f = B\beta \tag{18c}$$

$$\beta \geq 0 \tag{18d}$$

$$\tau_{\text{min}} \leq J_h^T(q, x_o)f \leq \tau_{\text{max}} \tag{18e}$$

where $f = [f_1^T, ..., f_{n_c}^T]^T$ is a concatenated contact force vector in contact frame. f_{prev} is the contact force of the previous time step. $B = \text{diag}\{B_1, ..., B_{n_c}\}$ and B_i is a conservative pyramid approximation of friction cone [10]. $\beta \geq 0$ is the non-negative coefficients of columns of B. A slack variable Ψ is introduced to relax the hard constraint $F = Gf$, since $F = Gf$ might result in an infeasible solution, and the location measurements of contact points might be noisy. The constraints (18c) and (18d) together ensure that the contact force remains within positive $colspan(B)$ (i.e. friction cone). Constraint (18e) guarantees that the contact force f is realizable.

The weights $\alpha_1, \alpha_2, \alpha_3$ are used to balance different cost terms. They are tuned to penalize the magnitude of the contact force, the change of the contact force and the force tracking error, respectively. The tuning process considers the response speed of the real-world hand actuators and the force tracking performance.

The joint level torque control takes the optimal contact force f^* from the force optimization as input, and yields the control torque by $\tau = J_h^T(q, x_o)f^*$.

VI. SIMULATION RESULTS

A. Simulation Setup

The controller is implemented in the Mujoco physics engine [11]. The simulation time step is set to be 2 ms. Our platform is a desktop with 4.0 GHz Intel Core Quad CPU, 32GB RAM, and running Windows10 operating system.

The hand models used in the simulation are shown in Fig. 4. The general hand model used in 3D manipulation is shown in Fig. 4(a). It has four identical fingers and 12 DOFs. Each finger has three revolute joints. For illustration purpose, a planar hand with two identical fingers and 4 DOFs is set up, as shown in Fig. 4(b). Two hands are equipped with joint encoders, motor torque sensors, and one-dimensional distributive tactile sensors. The manipulated

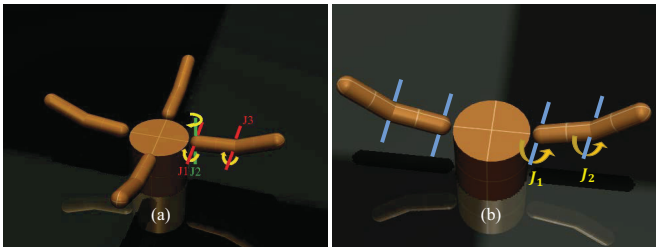

Fig. 4: Two hand models used in the simulation.

object is approximately 0.5 kg. The dynamics parameters of the object are assumed to be unknown to the controller. A vision system can be employed to estimate the dynamics parameters, and obtain the motion by tracking the features on it. Currently, the object pose is obtained from the simulator, and the mass and inertia are assumed to have 40% and 50% uncertainties in the robust controller design.

B. Parameter Lists

The parameters of the weighting functions in Section IV-B are shown in Table I:

TABLE I: Parameters of Weighting Functions

Weightings	ω_c	G_l	G_h	(a_1, a_2, a_3)	n
W_{perf}	2π	1100	0.9	(1,1,2)	2
W_u	N/A	0.0001	0.0001	(1,1,0.5)	1
W_{dis}	200π	80	0.1	(1,1,10)	2
W_n	20π	0.1	10	(1,1,1)	1

The parameters of manipulation controller are set as follows: the joint torques are constrained by $\tau_{\text{min}} = -0.5$ Nm and $\tau_{\text{max}} = 0.5$ Nm. 0.5 Nm is twice the joint torque in static case. The weights for different cost terms in (18) are $\alpha_1 = 0.1, \alpha_2 = 0.1, \alpha_3 = 1000$. The dimensionality of the contact space in the simulation is 6, with sliding, torsional and rolling friction coefficients 1, 0.005 and 0.0001, respectively. In manipulation controller design, we use the point contact with friction model [7] and assume a conservative sliding friction coefficient 0.5774. The algorithm has been verified to be robust to various contact surface parameters (e.g. stiffness and damping).

C. Simulation Results

In this section, the proposed controller is compared with disturbance observer (DOB) based tracking [5] and modified impedance control (MIC) [12]. For illustration purpose, we first show the comparison results in 2D manipulation.

The desired object motion is to move from (139 mm, 0 mm, 0°) to (150 mm, -10 mm, 5°). The equilibrium point is chosen at the beginning of the contact. The configuration of both the hand and the object at the beginning of the contact can be planned by grasp planning [13]. In this paper, the equilibrium point is prerecorded for simplicity. Therefore, we can calculate all the nominal parameters that are required for modeling. The controller is designed based on 40% mass and 50% inertia uncertainties. The order of the controller is 29 after model reduction. The robust stability margin is 1.62, which means that the system can withstand about 62% more

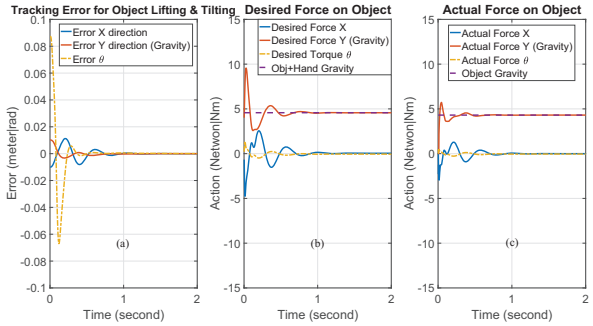

Fig. 5: Performance of the proposed method in Mujoco.

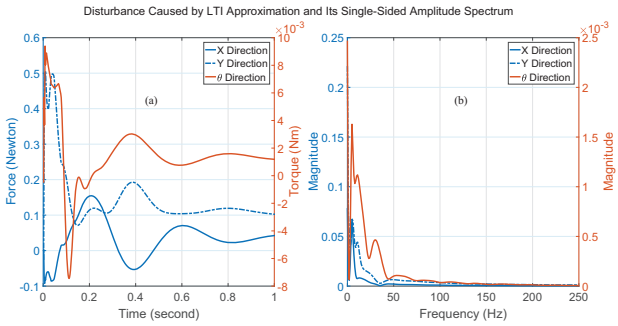

Fig. 7: Disturbance caused by LTI approximation.

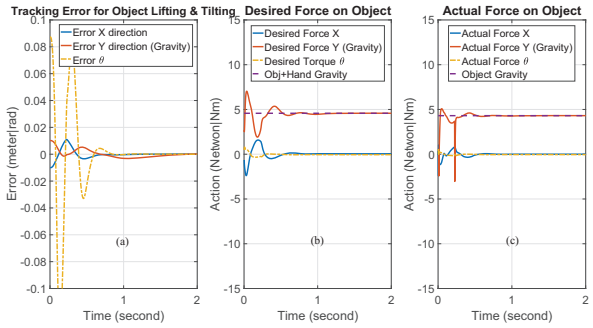

Fig. 6: Performance of the robust controller without feedback linearization.

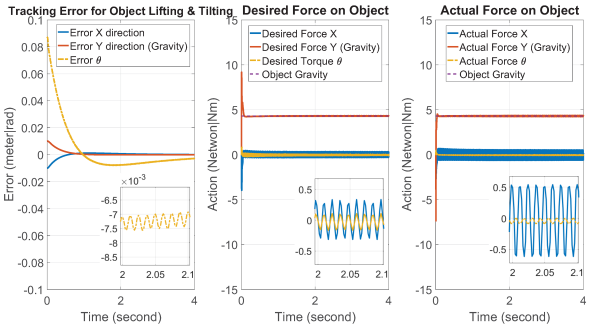

Fig. 8: Performance of DOB [5] in Mujoco.

uncertainties than are specified in the uncertain elements without going unstable.

The simulation results of the proposed method under -20% mass uncertainty and 50% inertia uncertainty are shown in Fig. 5. Fig. 5(a) shows the position and orientation errors of the object. The maximum settling time[1] of all channels is 0.91 seconds. Fig. 5(b) shows the desired force on the composite hand-object system. The force in gravity direction (shown in red) rapidly converges to actual gravity of the composite system (purple dash line), though there exists 20% variations between the nominal and the actual object mass. The desired force is converted into joint torque command by the manipulation controller. Fig. 5(c) shows the actual force on the object detected by the force sensor (force sensor is used for analysis of the results only).

The tracking performance of the robust controller without feedback linearization is shown in Fig. 6. Compared with Fig. 5, the robust controller without feedback linearization has more severe oscillation and longer settling time. The oscillation is caused by the large disturbance introduced by the nonlinearities of the hand-object system.

In Section III-B, we introduce feedback linearization to reduce the nonlinearities of the nominal system. However, the nonlinearities still exist in the uncertain plant, as shown in (15). The model uncertainties in (15) are evaluated at an equilibrium point, which introduces additional disturbance. We call it the disturbance from the LTI approximation, and denote it as d_{LTI}:

$$d_{\text{LTI}} = (I - \bar{M}_{\text{eq}}\bar{M}^{-1})(\tilde{M}_o\ddot{x}_o + \tilde{N}_o) - \bar{M}_{\text{eq}}\bar{M}^{-1}\tilde{C}_o\dot{x}_o \quad (19)$$

where $\bar{M}_{\text{eq}}$ is the nominal inertia matrix at equilibrium point. The magnitudes of d_{LTI} in both time and frequency domain are shown in Fig. 7. Figure 7(a) and (b) show the magnitude and spectrum of d_{LTI}. The disturbance caused by equilibrium approximation mainly lies in low-frequency region($<$12 Hz), which has been suppressed by the proposed robust controller.

In comparison, the simulation results of the DOB and MIC under the same amount of mass and inertia uncertainties are shown in Fig. 8 and 9. The convergence speed of DOB (settling time 3.33 seconds) is slightly faster than MIC (settling time 3.61 seconds). Both of these two methods make the desired and actual force in y direction converge to actual gravity. In DOB method, the nonlinearities and uncertainties are lumped into a disturbance term, which loses structures of both the nominal models and the uncertainties. Therefore, the system using DOB takes longer time to converge. Besides, the tuning process of DOB is time consuming, which poses a potential challenge to using this method. Our best tuned result is shown in Fig. 8. The MIC method uses constant gain matrices, and relies on an integral term to accumulate force to compensate gravity uncertainty, which also results in slow convergence, as shown in Fig. 9. Moreover, the imprecise inertia matrix causes certain difficulties in guaranteeing the stability of the controller.

Finally, the 3D manipulation performance for 20% mass and 50% inertia uncertainties using the 12 DOFs hand is shown in Fig. 10. The Coriolis term is ignored and the

[1]5% threshold is used for all settling time calculations.

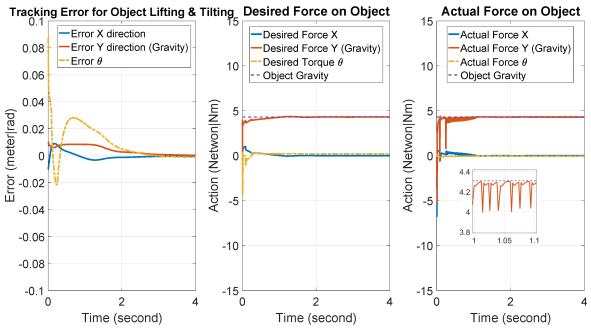

Fig. 9: Performance of MIC [12] in Mujoco.

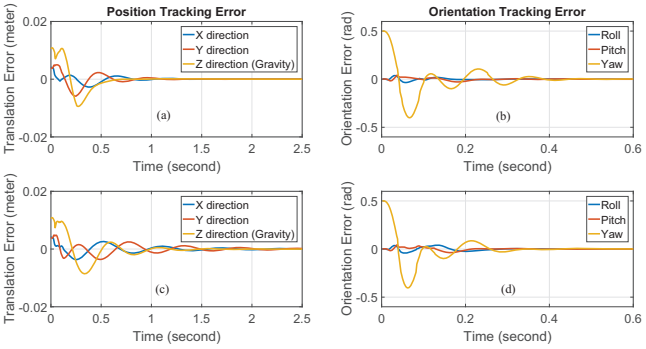

Fig. 11: Performance of the proposed method on 3D manipulation.

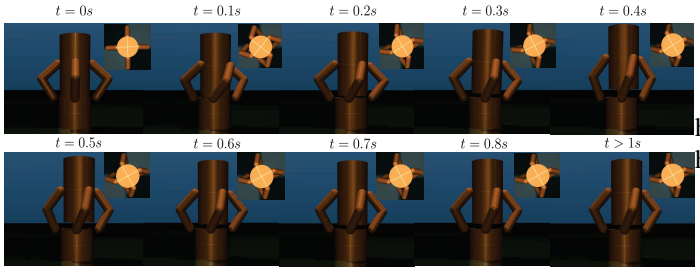

Fig. 10: Performance of 3D manipulation using the proposed algorithm.

velocity measurement is not required. The desired object position and orientation displacements are (4, 4, 11) mm and (0, 0, 0.5) rad. The tracking errors are shown in Fig. 11. Fig. 11(a)(b) shows the position and orientation errors for 20% mass and 50% inertia uncertainties, and the settling time is 1.11 seconds. Fig. 11(c)(d) shows the position and orientation errors for 40% mass and 50% inertia uncertainties, and the settling time is 2.21 seconds.

VII. CONCLUSION

This paper has proposed a dual-stage grasp controller, which includes a robust controller and a manipulation controller, to achieve dexterous manipulation under object dynamics uncertainties and external disturbances. Feedback linearization has been applied to reduce the nonlinearities of the composite hand-object system. By utilizing the structures of the uncertainties, the proposed robust controller can achieve faster convergence and tolerate more uncertainties compared to other methods based on DOB and MIC. The dual-stage formulation skipped complicated contact modeling, and was able to regulate contact force and prevent slippage. Moreover, it did not require joint/object velocity measurement or 3D/6D tactile sensor. Simulations showed that our method can achieve robust manipulation and the fast tracking performance.

In the future, the authors plan to combine this work with finger gaits planning [12] to address the unknown object shape, unexpected slippage issues and realize large-scale object motions.

ACKNOWLEDGMENT

The authors would like to thank Prof. Andrew Packard for his advice on robust control, and Prof. Emanuel Todorov for his help on Mujoco.

REFERENCES

[1] Y. Fan, L. Sun, M. Zheng, W. Gao, and M. Tomizuka. (2017) Robust dexterous manipulation under object dynamics uncertainties. Youtube. [Online]. Available: https://youtu.be/aRSqkSQENY8/

[2] A. Caldas, A. Micaelli, M. Grossard, M. Makarov, P. Rodriguez-Ayerbe, and D. Dumur, "Object-level impedance control for dexterous manipulation with contact uncertainties using an lmi-based approach," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 3668–3674.

[3] T. Takahashi, T. Tsuboi, T. Kishida, Y. Kawanami, S. Shimizu, M. Iribe, T. Fukushima, and M. Fujita, "Adaptive grasping by multi fingered hand with tactile sensor based on robust force and position control," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 264–271.

[4] H. Koc, D. Knittel, M. de Mathelin, and G. Abba, "Modeling and robust control of winding systems for elastic webs," *IEEE Transactions on control systems technology*, vol. 10, no. 2, pp. 197–208, 2002.

[5] C.-S. Liu and H. Peng, "Disturbance observer based tracking control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 122, no. 2, pp. 332–335, 2000.

[6] A. Mokhtari, N. K. M'Sirdi, K. Meghriche, and A. Belaidi, "Feedback linearization and linear observer for a quadrotor unmanned aerial vehicle," *Advanced Robotics*, vol. 20, no. 1, pp. 71–91, 2006.

[7] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[8] M. Li, H. Yin, K. Tahara, and A. Billard, "Learning object-level impedance control for robust grasping and dexterous manipulation," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 6784–6791.

[9] G. J. Balas, J. C. Doyle, K. Glover, A. Packard, and R. Smith, "μ-analysis and synthesis toolbox," *MUSYN Inc. and The MathWorks, Natick MA*, 1993.

[10] C. K. Liu, "Dextrous manipulation from a grasping pose," in *ACM Transactions on Graphics (TOG)*, vol. 28, no. 3. ACM, 2009, p. 59.

[11] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 5026–5033.

[12] Y. Fan, W. Gao, W. Chen, and M. Tomizuka, "Real-time finger gaits planning for dexterous manipulation," *The 20th World Congress of the International Federation of Automatic Control (IFAC)*, 2017, to be presented.

[13] J.-P. Saut and D. Sidobre, "Efficient models for grasp planning with a multi-fingered hand," *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 347–357, 2012.