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, complicated manipulation tasks, it is desired that a multi-fingered robotic hand can robustly manipulate objects without knowing the full dynamics of objects (i.e. mass, moment of 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 using force optimization and torque control regulates the contact force and torque 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 at least 40% mass uncertainty and 50% moment of inertia uncertainty. Moreover, it does not require velocity information of the object. Simulation results on Mujoco verify the efficacy of the proposed dual-stage grasp controller. 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 moment of inertia (MoI). In many applications (e.g. fruit harvesting), the accurate models of the object dynamics are usually unknown in advance. They are estimated by 3D sensing, as well as prior knowledge such as density and statistical model. 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 contacts, which are complex to model because of various surface properties. 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, which is challenging for robust control. Moreover, the potential sliding between the fingertips and the object would degrade the object motion tracking performance.

[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

As a result, robust dexterous manipulation for nonlinear systems has received significant attention. A robust controller using linear matrix inequality for contact uncertainties was proposed in [2]. The controller is designed for a LTI system linearized around an equilibrium point based on full state feedback. A robust force-position controller using 6D tactile sensors was implemented to realize adaptive grasping [3]. Nonlinearities were ignored in [3] due to its constant-pose grasping property. μ-synthesis with descriptor form was used to control magnetic bearing systems [4], active magnetic levitation systems [5], and satellite attitude control systems [6]. These applications work on linearized systems without stability guarantee under the influence of nonlinearities. In order to consider parameter variations caused by nonlinearities, a LPV control with smooth scheduling was applied in [7], with an assumption that the nonlinearities can be approximated through linear varying parameters.

To deal with dynamics uncertainties, a disturbance observer (DOB) was proposed in [8] for tracking control. The nonlinearities and parameter uncertainties are lumped into a disturbance term, which loses structures of both the nominal models and the uncertainties. Moreover, it assumes full state feedback, while in dexterous hand, the velocity feedback is difficult due to the size constraints, backlash error and cost issue. Feedback linearization was applied to control an unmanned aerial vehicle [9]. A linear state observer and a DOB are combined to observe the state and lumped disturbance. Similar to [8], the structures of the parameter uncertainties are ignored, and the linear state observer assumes a perfect model information for state estimation.

This paper proposes a dual-stage grasp controller which consists of a robust controller and a manipulation controller. The proposed dual-stage grasp controller achieves 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 parameter variations, 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 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. At each time step, an Cartesian space force is computed to drive the object based on the designed robust controller. The proposed manipulation controller consists of a contact force optimizer

and a joint-level torque controller, which provides reasonable contact force on fingertips and prevent slippage between the fingertips and the object. 4) Moreover, the dual-stage grasp controller does not require expensive 3D/6D tactile sensors nor velocity measurements of objects/joints.

The remaining of this paper is organized as follows. Section II introduces the overall dual-stage grasp controller framework. Section III describes the system dynamics and the combination of the feedback linearization and the modeling. The μ-synthesis based robust controller, and the manipulation controller that consists of a force optimizer and a torque controller, are presented in Section IV and Section V, respectively. Section VI shows the simulation results on a robotic hand manipulating an object with 40% mass and 50% MoI uncertainties. Section VII concludes the paper.

II. DUAL-STAGE PLANNER 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 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 design goal of this grasp controller is to:

1) track the desired pose r of the object,
2) be robust to object dynamics uncertainties (i.e. mass and MoI uncertainties) and external disturbances u_{dis},
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 of the object. The robust controller is applied on a linearized nominal plant with nonlinear uncertainties. The linearized nominal plant is obtained by feedback linearization on an augmented nonlinear plant. The desired Cartesian force F obtained from feedback linearization and robust controller is converted into torque command τ to the hand by the manipulation controller.

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

The following sections focus on the modeling of the uncertain dynamical systems, robust controller design, and the manipulation controller, respectively.

III. MODELING OF UNCERTAIN DYNAMICAL SYSTEMS

A. State-Space Dynamics

The hand and object dynamics are described by:

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

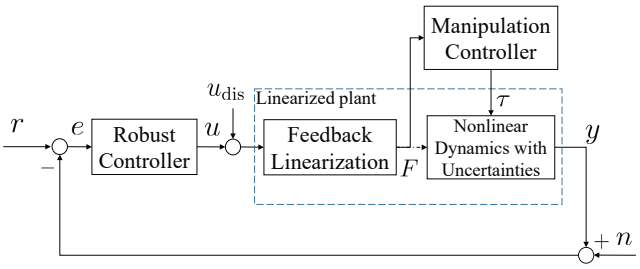

Fig. 1. The general framework of the proposed dual-stage grasp controller.

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, and $n_x = 3$ 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, see [10] for more details.

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 vision sensors are available for dynamics parameter estimation. Hence, only the rough values of the mass m_o and the MoI $\mathcal{I}_o$ 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):

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

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

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

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

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

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

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

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 MoI 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 $(\bullet)^\wedge$ denotes the matrix representation of cross product.

With (9), the uncertainty term $-M_{\text{aug}}^{-1}(\tilde{M}_{\text{aug}}\dot{x} + \tilde{C}_{\text{aug}}x)$ 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

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

In general 3D manipulation, the Coriolis term is typically ignored due to the low-speed operation condition, as shown in [11]. The tracking performance neglecting Coriolis term in 3D manipulation is shown in Fig. 12.

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:

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

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:

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

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

$$\begin{aligned}\dot{x} &= Ax + L_1w + B_F(u + d)\\ z &= R_1Ax + R_1L_1w + R_1B_F(u + d)\\ y &= [I_{3\times3}, 0_{3\times3}]x \qquad w = \mathbf{\Delta}z\end{aligned} \tag{15}$$

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

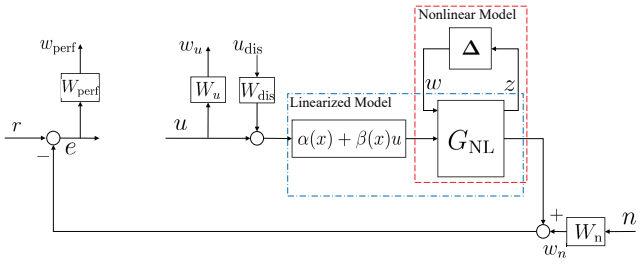

Fig. 2. Generalized plant with weighting functions.

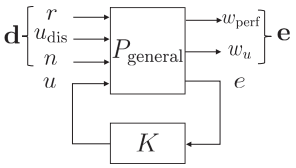

Fig. 3. Illustration of the closed-loop system. The closed-loop system is defined by a lower LFT between P_{general} and K w.r.t. K. The input and output of the system are concatenated as $\mathbf{d}$ and $\mathbf{e}$, respectively.

is approximated by a LTI system linearized around an equilibrium point. The resultant system after LTI approximation 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

The goal of the robust controller is to obtain desired Cartesian force of the object for motion tracking with guaranteed stability and performance robustness. 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. The structures of the weighting functions will be described in Section IV-B.

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 [12] for more 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:

$$\begin{aligned} W(s) &= \text{diag}([a_1 W_{1,1}(s), a_2 W_{2,2}(s), a_3 W_{3,3}(s)]) \\ W_{i,i}(s) &= \left(\frac{G_h s + G_l \sqrt{\frac{G_h^2 - 1}{1 - G_l^2}} \omega_c}{s + \sqrt{\frac{G_h^2 - 1}{1 - G_l^2}} \omega_c} \right)^n \end{aligned} \tag{18}$$

where a_i is the weight to the i-th channel. G_h is the high-frequency gain, G_l is the low-frequency gain, ω_c is the cross-over frequency, and n is the order for the weighting function. 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 crossover 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

Given the contacts between the fingertips and the object, the task of the manipulation controller is to generate torque commands for the hand to drive the object and achieve desired force on the object. The desired force is provided by the robust controller in Section IV. The manipulation controller consists of a force optimizer, which compute a 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 optimization is formulated into a quadratic programming (QP):

subject to:

$$\min_{\beta} \quad \alpha_1\|f\|_2^2 + \alpha_2\|f - f_{\text{prev}}\|_2^2 + \alpha_3\|\Psi\|_2^2 \tag{19a}$$

$$s.t. \quad \Psi = F - G(q, x_o)f \tag{19b}$$

$$f = B\beta \tag{19c}$$

$$\beta \geq 0 \tag{19d}$$

$$\tau_{\min} \leq J_h^T(q, x_o)f \leq \tau_{\max} \tag{19e}$$

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 [13]. $\beta \geq 0$ is the non-negative linear coefficients of columns of B. The weights $\alpha_1, \alpha_2, \alpha_3$ are used to balance different cost terms.

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 (19c) and (19d) together ensure that the contact force remains within positive $colspan(B)$ (i.e. friction cone). Constraint (19e) guarantees that the contact force f is realizable.

The joint level torque control takes the optimal contact force f^* from the force optimization as input, and yields the control torque:

$$\tau = J_h^T(q, x_o)f^*$$

$J_h^T(q, x_o)$ maps the contact force on fingertips to the joint torque vector.

VI. SIMULATION RESULTS

Simulation results are provided in this section to verify the effectiveness of the proposed dual-stage grasp controller. The simulation video is available at [1].

A. Simulation Setup

The controller is implemented in the Mujoco physics engine [14]. 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, 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 purposes of illustration, a planar hand with two identical fingers and 4 DOFs is set up, as shown in Fig. 4(a). Two hands are equipped with joint encoders, motor torque sensors, one-dimensional distributive tactile sensors. The manipulated 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 MoI are assumed to have 40% and 50% uncertainties in the robust controller design.

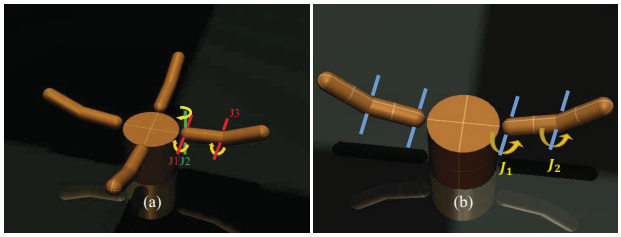

Fig. 4. Two hand models used in the simulation. The hand on the left side is used for 3D manipulation. It has four identical fingers and 12 DOFs. Each finger has three revolute joints J1, J2 and J3. The hand on the right side is developed for 2D manipulation. It has two identical fingers and 4 DOFs. Each finger has two revolute joints J1 and J2. The motion of the object is constrained in 2D space.

B. Comparison Between Different Methods

For comparison, the proposed robust controller is compared with other methods. One of these methods is DOB proposed in [8]. Another is modified impedance control (MIC) based on the previous work of the authors. The two approaches are briefly reviewed in the following.

1) Disturbance Observer Based Tracking Control: This method assumes the dynamics of the system as $M\ddot{x}_o + C\dot{x}_o + N + d = F$. The dynamics can be rewritten as $\ddot{x}_o = u + w$, where $u = F - \bar{N}$ and $\bar{N}$ denotes nominal gravity value, and $w = \ddot{x}_o - (M\ddot{x}_o + C\dot{x}_o + \tilde{N} + d)$ is the lumped disturbance. Similar to feedback linearization, the controller is set as: $u = \ddot{r} - k_1\dot{e} - k_2 e - \hat{w}$, where $e = x_o - r$. In this manner, the error dynamics are:

$$\begin{bmatrix}\dot{e}\\ \ddot{e}\end{bmatrix} = \underbrace{\begin{bmatrix}0_{3\times3} & I_{3\times3}\\ -k_2 & -k_1\end{bmatrix}}_{A}\begin{bmatrix}e\\ \dot{e}\end{bmatrix} + \underbrace{\begin{bmatrix}0_{3\times3}\\ I_{3\times3}\end{bmatrix}}_{B} e_w$$

where $e_w = w - \hat{w}$, $\hat{w} = \hat{w}_0 + K_o E$, and $\dot{\hat{w}}_0 = B^T P E$. P can be found by: $A^T P + PA = -Q$, and K_o can be found by: $K_o A + K_o B K_o + B^T P = 0$, with $E = [e^T, \dot{e}^T]^T$. The desired force on the object is $F = u + \bar{N}$.

2) Modified Impedance Control: In impedance control, the desired motion of the object is transformed into the desired force on the object [11], [15]. An additional integral term has been added in our previous work [16] to address the object mass uncertainty: $F_{\text{imp}} = M^d\ddot{x}_o + B^d(\dot{x}_o - \dot{r}) + K^d(x_o - r) + I^d\left(\int_0^t (x_o - r)dt\right)$, where M^d, B^d and K^d are the desired inertia, damping and stiffness, respectively. I^d is a gain matrix for additional integral term. To remove the requirement of acceleration measurement, we set $M^d = M$, as shown in [11]. In this manner, the desired force F can be obtained by $M\ddot{x} + \bar{N} = F + F_{\text{imp}}$. The unknown inertia matrix poses certain difficulties in guaranteeing the stability of the controller.

C. Parameter Lists

1) Parameters for Proposed Dual-Stage Planner: The parameters of weighting functions are shown in the following table:

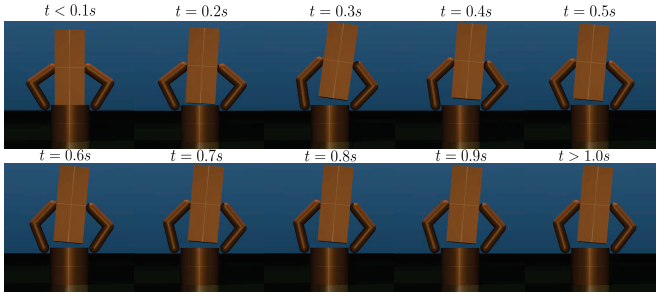

Fig. 5. A simulation on 2D manipulation under object dynamics uncertainties using the proposed dual-stage grasp controller. The task is to realize the desired object motion within the workspace of the hand. The object is subject to -20% mass and 50% MoI uncertainties. A robust controller that can resist $\pm 40\%$ mass and $\pm 50\%$ MoI uncertainties is implemented.

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

As for the parameters for manipulation controller, 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 are $\alpha_1 = 0.1, \alpha_2 = 0.1, \alpha_3 = 1000$. The actual friction coefficient in Mujoco simulation is set to be 1, while in manipulation controller design, we use a conservative coefficient 0.5774.

2) Parameters for DOB and MIC: The parameters for DOB based control are: $k_2 = 95 \times \text{diag}([1, 1, 0.01])$, $k_1 = 95 \times \text{diag}([1, 1, 0.008])$, and Q matrix is chosen as: $Q = \text{diag}([51000, 51000, 6, 51000, 51000, 1])$. The parameters for MIC are: $K^d = 50 \times \text{diag}([1, 1, 0.2])$, $B^d = 5 \times \text{diag}([1, 1, 0.02])$, $I^d = 50 \times \text{diag}([1, 1, 0.2])$.

D. Simulation Results

A general manipulation task is employed to verify the efficacy of the proposed dual-stage grasp controller. The desired object motion is to track to (150 mm, -10 mm, 5°) from (139 mm, 0 mm, 0°). 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 such as [17]. 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% MoI 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 uncertainties than are specified in the uncertain elements without going unstable.

The simulation results of the proposed method under -20% mass uncertainty and 50% MoI uncertainty are shown in Fig. 5 and Fig. 6. Fig. 6(a) shows the position and orientation errors of the object. The maximum settling time[1] of all channels is within 0.91 seconds. Fig. 6(b) shows the desired

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

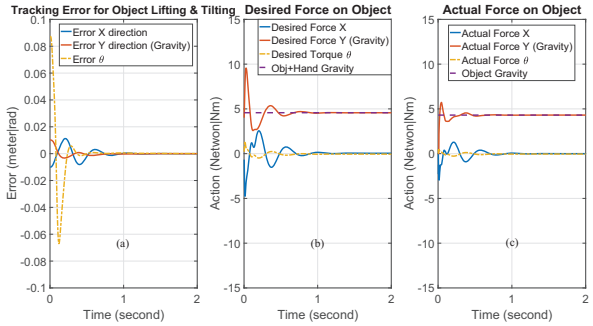

Fig. 6. Illustration of the performance for the dual-stage grasp controller with robust controller and feedback linearization. The manipulated object is subject to -20% mass and 50% of MoI uncertainties. The maximum settling time of all channels is less than 0.91 seconds.

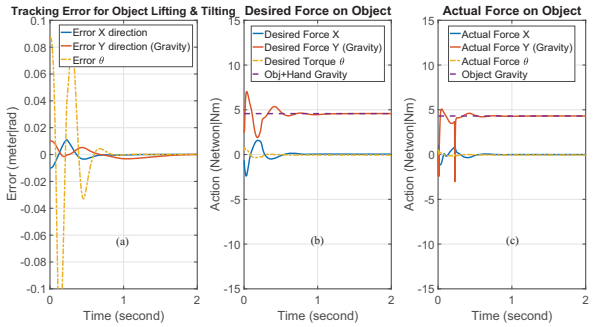

Fig. 7. Illustration of the performance of the best-tuned robust controller without feedback linearization. The settling time is 1.6836 seconds.

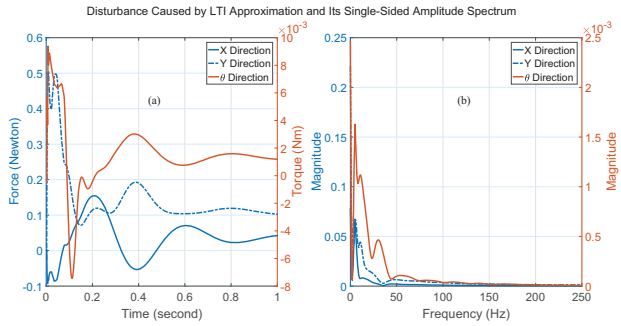

Fig. 8. Illustration of the disturbance caused by LTI approximation after feedback linearization. The disturbance caused by equilibrium approximation mainly lies in low-frequency region(<12 Hz).

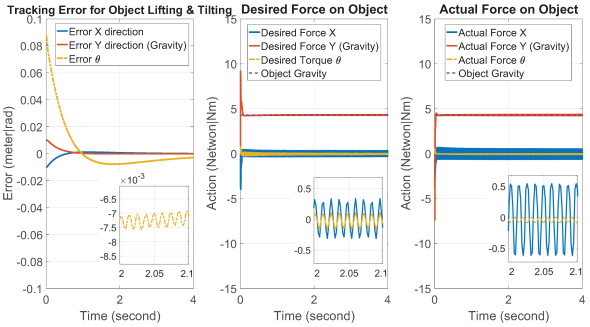

Fig. 9. Illustration of the performance of DOB proposed by [8] in Mujoco. The manipulated object is subject to -20% mass uncertainty. The settling time is 3.3321 seconds. The oscillation is caused by a large Q matrix.

force on the composite hand-object system. The force in Y direction (shown in red) can rapidly converge to actual gravity of the composite system (purple dash line), though there exists 20% gap between the nominal and the actual object mass. The desired force is converted into joint torque command by the manipulation controller. Fig. 6(c) shows the actual force on the object detected by the force sensor (force sensor is used for result analysis only).

The tracking performance of the robust controller without feedback linearization is shown in Fig. 7. The robust controller is well-tuned by following the criteria introduced in Section IV-B.1. Compared with Fig. 6, the robust controller without feedback linearization has more severe oscillation and longer settling time. The oscillation is caused by the nonlinearities of the system.

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

$$d_{\text{LTI}} = (I - \bar{M}_{\text{eq}}\bar{M}^{-1})(\tilde{M}_o\ddot{x}_o + \tilde{N}_o) - \bar{M}_{\text{eq}}\bar{M}^{-1}\tilde{C}_o\dot{x}_o \quad (20)$$

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. 8. Figure 8(a) shows the magnitude of the disturbance, and Fig. 8(b) shows the spectrum of the d_{LTI}. The disturbance caused by equilibrium approximation mainly lies in low-frequency region(<12 Hz), which can be suppressed by the proposed robust controller.

In comparison, The simulation results of the DOB and MIC under the same amount of mass and MoI uncertainties are shown in Fig. 9 and Fig. 10. The convergence speed of DOB is slightly faster than MIC. Both of these two methods make the desired and actual force in y direction converges to actual gravity. Compared with MIC, the DOB has more intuitive way to guarantee stability. However, the tuning process of DOB takes longer time, which poses a potential challenge to use this method. On one hand, the Q matrix in Section VI-B.1 influences the disturbance estimation gain, and increasing Q will speedup the convergence, while a large Q will result in severe oscillation. On the other hand, the gain matrices k_1 and k_2 tuned in absence of disturbance does not work since they result in a slow disturbance compensation. The best tuned result is shown in Fig. 9.

The tracking performance for -40% mass uncertainty are shown in Fig. 11. The settling time for the proposed dual-stage grasp controller is 1.6804 seconds.

Finally, the 3D manipulation performance using the hand in Fig. 4(a) are shown in Fig. 12. The Coriolis term is ignored and the velocity measurement is not required. The desired object displaces are 4 mm (X), 10 mm (Z), 0.5 rad (RZ). The maximum settling time of all channels are 1.1 seconds.

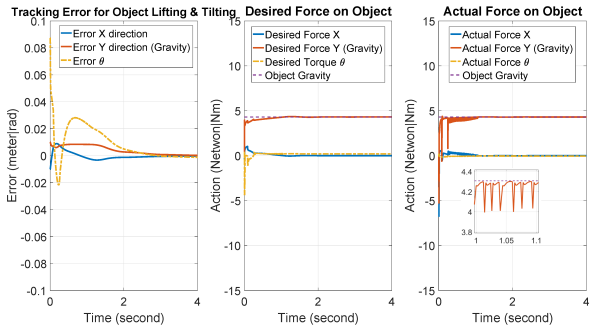

Fig. 10. Illustration of the performance of MIC in Mujoco. The manipulated object is subject to -20% mass uncertainty. The settling time is 3.6097 seconds. The oscillation in actual force is caused by unexpected contacts between the object and the palm.

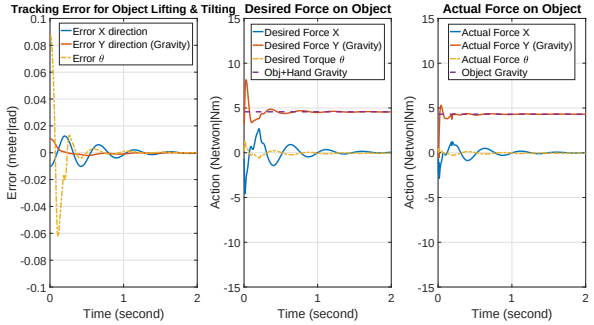

Fig. 11. Illustration of the performance of the proposed dual-stage grasp controller in Mujoco. The manipulated object is subject to -40% mass and 50% of MoI uncertainties. The maximum settling time of all channels are 1.6804 seconds.

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 tolerant 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.

Currently, the proposed method is limited to dynamics uncertainties. In the future, the authors plan to combine this work with finger gaits planning [16] 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.

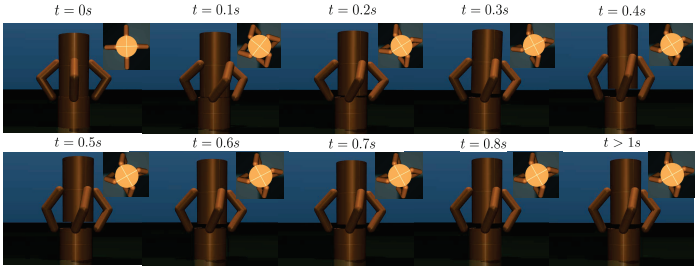

Fig. 12. Illustration of 3D manipulation performance using the proposed algorithm. The manipulated object has -20% mass and 50% of MoI uncertainties. The maximum settling time of all channels are 1.1064 seconds.

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] K. Nonami, N. Ide, and H. Ueyama, "Robust control of magnetic bearing systems using. mu.-synthesis with descriptor form." *JSME International Journal Series C*, vol. 40, no. 4, pp. 688–693, 1997.

[5] A. Piłat and P. Włodarczyk, "The μ-synthesis and analysis of the robust controller for the active magnetic levitation system," *Automatyka/Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie*, vol. 15, no. 1, pp. 85–98, 2011.

[6] T. Ohtani, Y. Hamada, T. Nagashio, T. Kida, S. Mitani, I. Yamaguchi, T. Kasai, and H. Igawa, "Robust attitude control using mu-synthesis for the large flexible satellite ets-viii," *The Journal of Space Technology and Science*, vol. 25, no. 1, pp. 1_27–1_40, 2009.

[7] 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.

[8] 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.

[9] 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.

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

[11] 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.

[12] 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.

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

[14] 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.

[15] T. Wimböck, C. Ott, A. Albu-Schäffer, and G. Hirzinger, "Comparison of object-level grasp controllers for dynamic dexterous manipulation," *The International Journal of Robotics Research*, vol. 31, no. 1, pp. 3–23, 2012.

[16] Y. Fan, W. Gao, 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.

[17] 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.