Neural Network Control of Multifingered Robot Hands Using Visual Feedback

Yu Zhao and Chien Chern Cheah

Abstract—**It is interesting to observe that humans are able to manipulate an object easily and skillfully without the exact knowledge of the object, contact points, or kinematics of our fingers. However, research so far on multifingered robot control has assumed that the kinematics and contact points of the fingers are known exactly. In many applications of multifingered robot hands, the kinematics and contact points of the fingers are uncertain and structures of the Jacobian matrices are unknown. In this paper, we propose an adaptive neural network (NN) Jacobian controller for multifingered robot hand with uncertainties in kinematics, Jacobian matrices, and dynamics. It is shown that using NNs, the uniform ultimate boundedness of the position error can be achieved in the presence of the uncertainties. Simulation results are presented to illustrate the performance of the proposed controller.**

Index Terms—**Multifinger control, neural network (NN) control, uncertain kinematics and dynamics, visual servoing control.**

I. Introduction

DEXTROUS manipulation has received increasing attention in recent years and considerable effort has been devoted to the development of controllers for multifingered robot hands [1], [2]. In model-based control methods of multifingered robot hands [3]–[5], the exact kinematics and dynamics are assumed to be known. To deal with the uncertain dynamic parameters, several adaptive controllers are proposed [6]–[9]. Recently, Ozawa *et al.* [10] proposed a controller that can achieve stable grasping, position, and relative angle control of the object without any object information. However, these controllers still require the exact kinematics and Jacobian matrix from joint space to task space. In practice, due to the imperfect knowledge of the locations of the contact points, the geometry of the object, etc., the kinematics of the multifingered robot hands cannot be determined exactly. For example, when the robot finger tips are soft and deformable, the kinematics also becomes uncertain due to depression and area contact. To alleviate this problem, Cheah *et al.* [11] propose a task space control law for setpoint control of multifingered robot hands with uncertain Jacobian matrices. In this paper, the exact kinematics and Jacobian of the multifingered robot hand are not required but the result is limited to a class of kinematic uncertainty. In addition, it is assumed that the structures of the Jacobian matrices are known and the uncertain parameters can be linearly parameterized.

It is interesting to observe that humans do not need exact knowledge of the kinematics and dynamics of the fingers but still able to manipulate things easily and skillfully. For example, when we manipulate an object using our fingers, we do not need to know the exact dimension and mass of the object, and the locations of the contact points between the fingers and the object. This is because humans are able to cope with the uncertainties through adaptation and learning. The exploration of a multifingered robot controller to cope with the uncertainties in both kinematics and dynamics is an important step towards understanding the dexterous manipulation by multifingered robot hands.

Neural networks (NNs) are able to approximate arbitrary nonlinear functions by learning through examples; it is therefore a powerful tool for control of systems with unknown structures [12]. There has been a lot of works on NN control of robots [13]–[21]. Some systematic approaches for structured dynamic modeling and adaptive control design for robots using NN can also be found in [22]. These approaches are focusing on manipulator control where the kinematics or Jacobian of the manipulator is assumed to be known exactly. Jagannathan and Galan [23] propose an NN position/force control scheme for a three-finger gripper, in which NNs are used to deal with uncertain dynamics. However, the kinematics are still assumed to be known and the NN control problem for multifingered robot hand with uncertainties in kinematics and dynamics has not been solved so far.

In this paper, NN is applied to multifingered robot control to derive a new vision-based controller that is able to deal with uncertainties in kinematics, dynamics, and Jacobian matrices during object manipulation. The main contribution is the proposal of adaptive NN Jacobian matrices in the multifingered robot controller to cope with parametric and structural uncertainty in the Jacobian matrices. It is shown that the proposed controllers can deal with kinematic and dynamic uncertainties with the introduction of a rotation matrix. Simulation results are presented to illustrate the performance of the proposed control laws.

II. Neural Network Background

An important property of NNs is the ability to approximate an arbitrary nonlinear function $f(x)$ up to a small error. In order to represent the function, an approximating function is chosen first,

Manuscript received July 26, 2007; revised July 18, 2008 and November 11, 2008; accepted December 02, 2008. First published April 14, 2009; current version published May 01, 2009.

Y. Zhao was with School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore. He is now with the OGS Industries, Singapore 189268, Singapore (e-mail: ZhaoYu@pmail.ntu.edu.sg).

C. C. Cheah is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore (e-mail: ECCCheah@ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2008.2012127

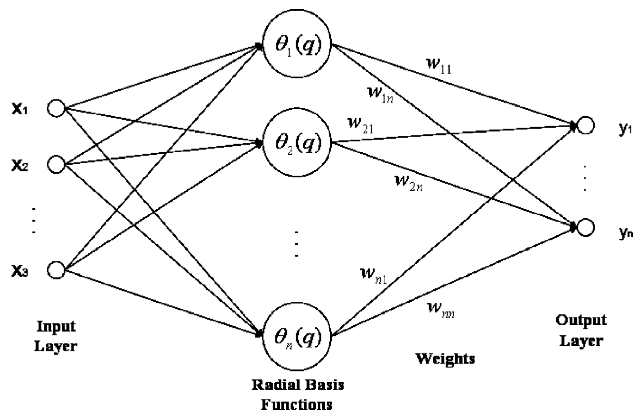

Fig. 1. RBF NN.

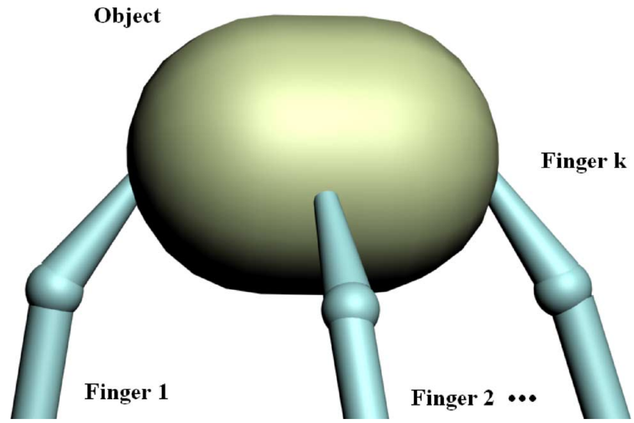

Fig. 2. Multifingered robot holding an object.

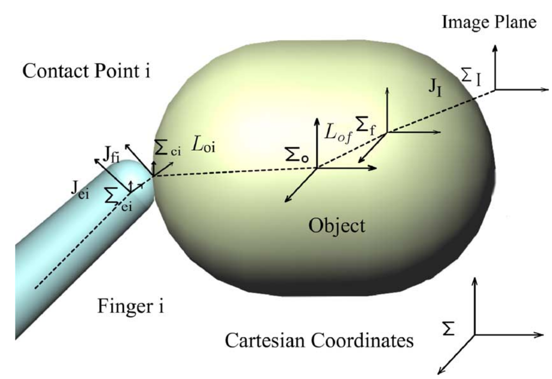

Fig. 3. Definitions of coordinates.

then the weights W are updated according to an algorithm based on the output errors [24]. In this paper, radial basis function (RBF) network is used and the NN weights are adjusted online using an update law [18], [22]. The structure of the NN is shown in Fig. 1.

From Fig. 1, the function approximation using an RBF network is

$$f(x) = W\theta(x) + E \tag{1}$$

where W is the matrix of NN weights and E is called NN functional approximation error [18], [22], [25]. It generally decreases when the number of neurons increases and $\theta(x)$ is the activation function. There are many kinds of activation functions that can be chosen for RBF networks. It has been shown that a linear superposition of Gaussian RBF results in an optimal mean square approximation to an unknown function, which is infinitely differentiable and whose values are specified by a finite set of points in $\Re$ [22]. Therefore, Gaussian RBF network is used in this paper. The Gaussian function is given as [24]

$$\theta(x) = \exp\left[\frac{-(x-\mu)^2}{\sigma^2}\right] \tag{2}$$

where μ is called center and σ is called distance. In this paper, the weight matrix is updated online and the update law will be derived from a Lyapunov-like method.

III. Dynamics Equations and Problem Formulation

We consider a set of k fingers holding an object as illustrated in Fig. 2. Let Σ denote the Cartesian coordinate, Σ_o be the object coordinate frame fixed at the mass center of the object and moving with the object, Σ_{ci} be the contact point frame located at the contact point of the ith finger, and Σ_{ei} be the finger coordinate frame located at the ith finger as shown in Fig. 3. The velocity vector v_o of the object in the object coordinate frame Σ_o is related to the velocity vector v_{ci} at the contact point of the ith finger in the contact point frame Σ_{ci} as [6], [26]

$$v_{ci} = L_{oi}v_o \tag{3}$$

where $L_{oi} \in \Re^{n_o \times n_o}$ denotes a nonsingular transformation matrix from the object coordinate frame Σ_o to the contact point frame Σ_{ci}. The velocity vector v_{ci} at the contact point in the contact point coordinate frame Σ_{ci} is related to the velocity vector v_{ei} of the i^{th} finger in the finger coordinate frame Σ_{ei} as

$$v_{ci} = J_{fi}v_{ei} \tag{4}$$

where $J_{fi} \in \Re^{n_o \times n_o}$ denotes a Jacobian matrix from the finger coordinate frame Σ_{ei} to the contact coordinate frame Σ_{ci}. Next, v_{ei} in the finger coordinate frame Σ_{ei} is expressed as

$$v_{ei} = J_{ei}(q_i)\dot{q}_i \tag{5}$$

where $q_i \in \Re^{ni}(n_i > n_o)$ is the joint coordinates of the ith finger and $J_{ei}(q_i) \in \Re^{n_o \times n_i}$ is the Jacobian matrix of Σ_{ei} in q_i. From (3)–(5), the velocities of the joint variables $q = [q_1^T, \ldots, q_k^T]^T$ and the velocity of the object v_o are constrained by the following:

$$L_o v_o = J_e(q)\dot{q} \tag{6}$$

where $L_o = [L_{o1}^T, \ldots, L_{ok}^T]^T$ and $J_e(q) = \mathrm{diag}\{J_{f1}J_{e1}(q_1), \ldots, J_{fk}J_{ek}(q_k)\}$. In addition, let $r \in \Re^{n_o}$ denote the position vector of the object coordinate frame Σ_o in the Cartesian coordinate Σ, then the velocity vector $\dot{r}$ is related to the velocity vector v_o of the object in object coordinate frame as [6], [26]

$$v_o = J_v(r)\dot{r} \tag{7}$$

where $J_v(r)$ is a nonsingular Jacobian mapping from the Cartesian coordinate Σ to the object coordinate frame Σ_o. From (6) and (7), the kinematic constraint between $\dot{r}$ and $\dot{q}$ is

$$J(r)\dot{r} = J_e(q)\dot{q} \tag{8}$$

where $J(r) = L_o J_v(r)$.

In this paper, cameras that are fixed in the workspace are used to observe a feature located on the surface of the object. We define a feature coordinate frame Σ_f fixed on the feature of the object and $r_f \in \Re^{n_o}$ is the position vector relative to the origin of the feature coordinate frame Σ_f in the Cartesian coordinate Σ. Then $\dot{r}$ has the following relation with $\dot{r}_f$:

$$\dot{r} = L_{of}\dot{r}_f \tag{9}$$

where $L_{of} \in \Re^{n_o \times n_o}$ denotes a transformation matrix from the feature coordinate frame to the object coordinate frame. From (8) and (9), we can get

$$J_f(r)\dot{r}_f = J_e(q)\dot{q} \tag{10}$$

where

$$J_f(r) = J(r)L_{of}. \tag{11}$$

The mapping from the feature space to image space requires a camera lens model in order to present the projection of features onto the charge-coupled camera (CCD) image plane. The pinhole camera model [27] is widely used and has proven adequate for most visual servoing tasks. Let $x \in \Re^p$ denote a vector of image feature parameters and $\dot{x}$ the corresponding vector of image feature parameter rates of change. The relationship between Cartesian space and image space is represented by [27]

$$\dot{x} = J_I(r_f)\dot{r}_f \tag{12}$$

where $J_I(r_f) \in \Re^{p \times n_o}$ is the image Jacobian matrix.

The dynamics equations of the ith finger are described in the joint coordinates q_i as [6], [26]

$$M_i(q_i)\ddot{q}_i + \left(\frac{1}{2}\dot{M}_i(q_i) + S_i(q_i, \dot{q}_i)\right)\dot{q}_i + g_i(q_i) + d_i = \tau_i - J_{ei}^T(q_i)J_{fi}^T f_{ei} \tag{13}$$

where $M_i(q_i) \in \Re^{ni \times ni}$ is the inertia matrix, which is symmetric and positive definite for all q_i, $g_i(q_i) \in \Re^{ni}$ is the gravitational force, $\tau_i \in \Re^{ni}$ is the control input, and $d_i \in \Re^{ni}$ is a vector of bounded disturbance. $f_{ei} \in \Re^{ni}$ is the force exerted on the object by the ith finger and $S_i(q_i, \dot{q}_i)$ is a skew-symmetric matrix

$$S_i(q_i, \dot{q}_i)\dot{q}_i = \frac{1}{2}\dot{M}_i(q_i)\dot{q}_i - \left\{\frac{\partial}{\partial q_i}\dot{q}_i^T M_i(q_i)\dot{q}_i\right\}^T. \tag{14}$$

The dynamic equations of the k fingers can be expressed in q as

$$M(q)\ddot{q} + \left(\frac{1}{2}\dot{M}(q) + S(q, \dot{q})\right)\dot{q} + g(q) + d = \tau - J_e^T(q)f_e \tag{15}$$

where $M(q) = \mathrm{diag}\{M_1(q_1), \ldots, M_k(q_k)\}$, $g(q) = [g_1^T(q), \ldots, g_k^T(q)]^T$, $S(q) = \mathrm{diag}\{S_1(q_1, \dot{q}_1), \ldots, S_k(q_k, \dot{q}_k)\}$, $\tau = [\tau_1^T, \ldots, \tau_k^T]^T$, $d = [d_1, \ldots, d_k]^T$, and $f_e = [f_{e1}^T, \ldots, f_{ek}^T]^T$. Note that $M(q)$ is also symmetric and positive definite and $S(q, \dot{q})$ is skew-symmetric.

The equation of motion of the object can be written in Cartesian coordinate Σ as

$$M_o(r)\ddot{r} + \left(\frac{1}{2}\dot{M}_o(r) + S_o(r, \dot{r})\right)\dot{r} + g_o(r) = F \tag{16}$$

where $M_o(r)$ is a positive–definite inertial matrix, $S_o(r, \dot{r})$ is a skew-symmetric matrix, $g_o(r)$ is the gravitational force, and F is the total force exerted on the object by the fingers and is represented by

$$F = J^T(r)f_e. \tag{17}$$

Substituting (9) into (16), we get

$$M_o(r)\frac{d}{dt}(L_{of}\dot{r}_f) + \left(\frac{1}{2}\dot{M}_o(r) + S_o(r, \dot{r})\right)L_{of}\dot{r}_f + g_o(r) = F. \tag{18}$$

From (17) and (10), it follows that [26], [28], [29]

$$f_e = (J^T(r))^+ F + Z(r)f_{\mathrm{int}} \tag{19}$$
$$\dot{r}_f = J_f^+(r)J_e(q)\dot{q} = J_r(q)\dot{q} \tag{20}$$

where $(J^T(r))^+ F$ is a particular solution to (17) that represents force that cause motion of the object, $(J^T(r))^+$ is the generalized inverse of $J^T(r)$, $Z(r)f_{\mathrm{int}}$ represents the internal force that does not effect motion of the object, $Z(r)$ is a matrix of orthonormal that is obtained from a set of independent vectors of the null space of $J^T(r)$, and $f_{\mathrm{int}} \in \Re^{n_I}$ is the magnitude of the internal force [26], [28], [29], which can be obtained from force sensors.

Substituting (19) into (15) and using (18), we can rewrite the coordinated system as follows:

$$\begin{aligned} M(q)\ddot{q} &+ J_e^T(q)(J^+(r))^T M_o(r)L_{of}\ddot{r}_f + \left\{\frac{1}{2}\dot{M}(q) + S(q, \dot{q})\right\}\dot{q} \\ &+ J_e^T(q)(J^+(r))^T\left(\frac{1}{2}\dot{M}_o(r) + S_o(r, \dot{r})\right)L_{of}\dot{r}_f + g(q) \\ &+ J_e^T(q)(J^+(r))^T g_o(r) + d \\ = &- J_m^T(q)f_{\mathrm{int}} + \tau \end{aligned} \tag{21}$$

where

$$J_m(q) = Z^T(r)J_e(q). \tag{22}$$

From (22), (8), and (20), we can also conclude that

$$J_m(q)\dot{q} = Z^T(r)J_e(q)\dot{q} = Z^T(r)J(r)\dot{r} = 0 \tag{23}$$

which indicates the geometric constraint on the joint angle velocity vector.

From (12) and (20), the image feature velocity is related to the joint angles of the robot fingers as

$$\dot{x} = J_I(r_f)J_r(q)\dot{q} = J_x(q)\dot{q} \tag{24}$$

where $J_x(q) = J_I(r_f)J_r(q)$.

The above kinematic equation shows the relationship between task space (or image space) and joint space. The Jacobian matrix $J_r(q)$ describes the mapping from joint space to the

Cartesian space while the image Jacobian matrix $J_I(r_f)$ indicates the mapping from Cartesian space to image space. The image Jacobian was first introduced by Weiss *et al.* [30] as the feature sensitivity matrix. It is also referred to as the interaction matrix [31]. From (20), (11), and (8), note that $J_r(q) = J_f^+(r)J_e(q)$, $J_f(r) = J(r)L_{of}$, and $J(r) = L_oJ_v(r)$ and hence the modeling of Jacobian from joint space to the Cartesian space involves the modeling of various Jacobian and transformation matrices that are difficult to obtain exactly. For example, when the contact points of the fingers are uncertain, then the transformation matrices L_{oi} in (3) are uncertain. The current multifingered robot control approaches in the literature have assumed that the contact points between fingers and object are point contact so that the Jacobian and transformations matrices can be determined. However, it is well known that fingers with area contacts can improve grasping ability by providing more frictions at the contact points. In such cases, it is also difficult to determine L_{oi} because of the area contacts. The depressions at the finger tips also introduce uncertainty in the finger kinematics and hence the Jacobian matrices J_{fi} in (4) are uncertain. As seem from (24), the task space control problem can be formulated as visual servoing tasks [27] using the image Jacobian matrix. However, the use of cameras introduces additional uncertainty from Cartesian space to image space due to the camera calibration errors and hence result in uncertainty in image Jacobian matrix J_I. The uncertain location of the feature point also introduces uncertainty of L_{of} in (9).

IV. Adaptive NN Jacobian Control of Multifingered Robot Hands

As discussed in the previous section, the models of the Jacobian matrices of the multifingered robot hand are difficult to estimate exactly. To solve this problem, an NN Jacobian controller is proposed for multifingered robot hand in this section. NN is a powerful tool to deal with the uncertainties but existing adaptive NN controllers for robots can only deal with dynamic uncertainty. Our proposed controller does not require the exact model of the Jacobian matrices of the multifingered robot hand and hence it is able to deal with uncertainties in kinematics, dynamics, and Jacobian matrices during the object manipulation.

First, the gravity terms in (21) can be approximated by an NN as

$$g(q) + J_e^T(q)(J^+(r))^T g_o(r) = W_g\theta_g(q) + E_g \tag{25}$$

where $\theta_g(q)$ is a vector of activation functions, W_g is a constant matrix of ideal network weights, which are unknown, and E_g is an approximation error of the gravity terms.

Next, the Jacobian matrix $J_m(q)$ is approximated by an NN as

$$J_m^T(q) = (W_{f1}\theta_f(q), \ldots, W_{fn_I}\theta_f(q)) + E_f \tag{26}$$

where $\theta_f(q)$ is a vector of activation functions, W_{fi} are constant matrices of ideal network weights that are unknown, and E_f is an approximation error of the force term. The Jacobian matrix $J_x(q)$ can also be approximated by NNs as

$$J_x(q) = (W_{x1}\theta_x(q), \ldots, W_{xkni}\theta_x(q)) + E_x \tag{27}$$

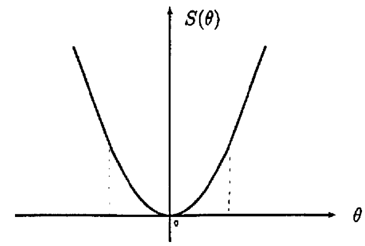

Fig. 4. Quasi-natural potential: $S(\theta)$.

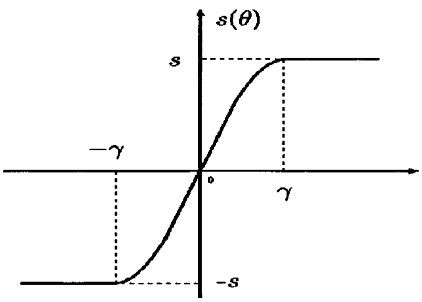

Fig. 5. Derivative of $S(\theta)$: $s(\theta)$.

where $\theta_x(q)$ is a vector of activation functions, W_{xi} are constant matrices of ideal network weights that are unknown, and E_x is the estimation error.

When the parameters and structure of the Jacobian matrices are uncertain, the Jacobian matrices are estimated as

$$\begin{aligned}\hat{J}_m^T(q,\hat{W}_f) &= (\hat{W}_{f1}\theta_f(q), \ldots, \hat{W}_{fn_I}\theta_f(q))\\ \hat{J}_x(q,\hat{W}_x) &= (\hat{W}_{x1}\theta_x(q), \ldots, \hat{W}_{xkni}\theta_x(q))\end{aligned} \tag{28}$$

where $\hat{W}_f$ and $\hat{W}_x$ are matrices of estimated network weights to be updated by update laws, and $\hat{W}_{xi}\theta_x(q)$ and $\hat{W}_{fi}\theta_f(q)$ are the ith column vectors of $\hat{J}_x(q,\hat{W}_x)$ and $\hat{J}_m^T(q,\hat{W}_f)$, respectively.

Let us define a scalar potential function $S_j(\theta)$ and its derivative $s_j(\theta)$ as shown in Figs. 4 and 5. The functions $S_j(\theta)$ and $s_j(\theta)$ have the following properties [26].

1) $S_j(\theta) > 0$ for $\theta \neq 0$ and $S_j(0) = 0$.
2) $S_j(\theta)$ is twice continuously differentiable, and the derivative $s_j(\theta) = dS_j(\theta)/d\theta$ is strictly increasing in θ for $|\theta| < \gamma_j$ with some γ_j and saturated for $|\theta| \geq \gamma_j$, i.e., $s_j(\theta) = \pm s_j$ for $\theta \geq +\gamma_j$ and $\theta \leq -\gamma_j$, respectively, where s_j is a positive constant.
3) There exists a constant $\bar{c}_j > 0$, such that
$$S_j(\theta) \geq \bar{c}_j s_j^2(\theta) \tag{29}$$
for $\theta \neq 0$.

Some examples of saturation function are given in [26].

The adaptive NN Jacobian controller is proposed as

$$\begin{aligned}\tau = &-\hat{J}_x^T(q,\hat{W}_x)K_p s(\Delta x) - \hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x)\dot{q}\\ &+ \hat{J}_m(q,\hat{W}_f)^T f_{\mathrm{int}} + \hat{J}_m(q,\hat{W}_f)^T RK_m\Big((f_{\mathrm{int}} - f_d)\\ &+ \gamma\int_o^t (f_{\mathrm{int}} - f_d)\mathrm{d}\tau\Big) + \hat{W}_g\theta_g(q) - K_f f_{\mathrm{int}}\end{aligned} \tag{30}$$

where $s(\Delta x) = (s_1(\Delta x_1), \ldots, s_p(\Delta x_p))^T$, $\Delta x = x - x_d = (\Delta x_1, \ldots, \Delta x_p)^T$ is a positional deviation from a desired position of the image feature x_d in image space, K_p and K_v are diagonal feedback gain matrices for the position error and the velocity, respectively, γ is a positive constant and f_d is a desired internal force, K_m is a gain matrix and $y = \dot{q} + \alpha \hat{J}_x^+(q, \hat{W}_x)s(\Delta x)$, K_f is a gain matrix defined as

$$K_{fij} = \bar{k}_{ij}\mathrm{sgn}(y_i f_{\mathrm{int}j}) \tag{31}$$

where $\bar{k}_{ij}$ is a positive constant, and y_i and $f_{\mathrm{int}j}$ are the ith and jth elements of y and f_{int}. The rotation matrix R is designed so that $y_r^T R \Delta F = 0$, where $R = \mathrm{diag}\{R_1, \ldots, R_{k-1}\}$ and $R_i \in \Re^{n_o \times n_o}$, $y_r = \hat{J}_m(q, \hat{W}_f)y$, $\Delta F = K_m((f_{\mathrm{int}} - f_d) + \gamma \int_o^t (f_{\mathrm{int}} - f_d)\mathrm{d}\tau)$. The estimated network weights $\hat{W}_g$, $\hat{W}_f$, and $\hat{W}_x$ in (30) and (28) are updated, respectively, by the following adaptive laws:

$$\begin{aligned}\dot{\hat{W}}_{gj}^T &= -k_1\hat{W}_{gj}^T + \mathrm{proj}(\Omega_{gj})\\ \dot{\hat{W}}_{fij}^T &= -k_2\hat{W}_{fij}^T + \mathrm{proj}(\Omega_{fij})\\ \dot{\hat{W}}_{xij}^T &= -k_3\hat{W}_{xij}^T + \mathrm{proj}(\Omega_{xij})\end{aligned} \tag{32}$$

where

$$\begin{aligned}&\Omega_{gj} = -\Gamma_{gj}^{-1}\theta_g(q)y_j, \quad \Omega_{fij} = -\Gamma_{fij}^{-1}\theta_f(q)y_j f_{\mathrm{int}i},\\ &\Omega_{xij} = \Gamma_{xij}^{-1}\theta_x(q)(k_{pi} + \alpha k_{vi})s(\Delta x_i)\dot{q}_j\end{aligned} \tag{33}$$

$\hat{W}_{gj}$ is the ith row vector of $\hat{W}_g$, $\hat{W}_{fij}$ is the ith row vector of $\hat{W}_{fj}$, $\hat{W}_{xij}$ is the ith row vector of $\hat{W}_{xj}$, Γ_{gj} Γ_{fij} and Γ_{xij} are positive–definite gain matrices and k_1, k_2, k_3 are positive constants, and the function $\mathrm{proj}(\Omega_{gj})$ is a projection algorithm defined as [32]

$$\mathrm{proj}(\Omega_{gj}) = \begin{cases} 0, & \text{if } \hat{W}_{gj} = \underline{W}_{gj} \text{ and } \Omega_{gj} < 0\\ & \text{or } \hat{W}_{gj} = \bar{W}_{gj} \text{ and } \Omega_{gj} > 0\\ \Omega_{gj}, & \text{else}\end{cases} \tag{34}$$

where $\underline{W}_{gj}$ and $\bar{W}_{gj}$ are the lower and upper bounds of W_{gj}. The projection algorithms $\mathrm{proj}(\Omega_f)$ and $\mathrm{proj}(\Omega_x)$ can be similarly defined as above. The projection functions are defined to ensure that $\hat{J}_x^T(q, \hat{W}_x)$ and $\hat{J}_m(q, \hat{W}_f)$ are bounded during adaptation while the terms $k_1\hat{W}_{gj}^T$, $k_2\hat{W}_{fij}^T$, and $k_3\hat{W}_{xij}^T$ are introduced to guarantee the uniformly ultimate boundedness of the system. The projection algorithms are only needed to ensure the boundedness of the updated NN weights. In practice, the bounds of the NN weights can be set to be large so long as they are bounded.

Substituting (30), (25), and (26) into the dynamics (21), we obtain the following closed-loop equation:

$$\begin{aligned}&M(q)\ddot{q} + J_e^T(q)(J^+(r))^T M_o(r)L_{of}\ddot{r}_f + \left\{\frac{1}{2}\dot{M}(q) + S(q,\dot{q})\right\}\dot{q}\\ &\quad + J_e^T(q)(J^+(r))^T\left(\frac{1}{2}\dot{M}_o(r) + S_o(r,\dot{r})\right)L_{of}\dot{r}_f\\ &\quad + \hat{J}_x^T(q,\hat{W}_x)K_p s(\Delta x) + \hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x)\dot{q}\\ &\quad + \Delta W_g\theta_g(q) + \Delta W_f\theta_f(q)f_{\mathrm{int}}\\ &= \hat{J}_m^T(q,\hat{W}_f)RK_m\left((f_{\mathrm{int}} - f_d) + \gamma\int_o^t (f_{\mathrm{int}} - f_d)\mathrm{d}\tau\right)\\ &\quad - E_g - d - E_f f_{\mathrm{int}} - K_f f_{\mathrm{int}}\end{aligned} \tag{35}$$

where $\Delta W_g = W_g - \hat{W}_g$ and $\Delta W_f = W_f - \hat{W}_f$ are the weight estimation errors.

To prove the stability of the adaptive NN Jacobian controller, we first define a Lyapunov-like function candidate as

$$\begin{aligned}V = &\frac{1}{2}\dot{q}^T M(q)\dot{q} + \frac{1}{2}\dot{r}_f^T L_{of}^T M_o(r)L_{of}\dot{r}_f\\ &+ \alpha s^T(\Delta x)(\hat{J}_x^+(q,\hat{W}_x))^T M(q)\dot{q}\\ &+ \alpha s^T(\Delta x)(\hat{J}_x^+(q,\hat{W}_x))^T J_e^T(q)(J^+(r))^T M_o(r)L_{of}\dot{r}_f\\ &+ \frac{1}{2}\sum_{j=1}^{n}\Delta W_{gj}\Gamma_{gj}\Delta W_{gj}^T + \frac{1}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\Delta W_{fij}\Gamma_{fij}\Delta W_{fij}^T\\ &+ \frac{1}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\Delta W_{xij}\Gamma_{xij}\Delta W_{xij}^T\\ &+ \sum_{j=1}^{p}(k_{pj} + \alpha k_{vj})S_j(\Delta x_j)\end{aligned} \tag{36}$$

where $\Delta W_x = W_x - \hat{W}_x$ are the weight estimation errors.

We will first show that the Lyapunov-like function candidate V in (36) is positive definite in $s(\Delta x)$, $\dot{q}$, ΔW_{gk}, ΔW_x, and ΔW_f and then proceeds to differentiate V with respect to time and prove the ultimate uniform boundedness of these errors. Note that V in (36) can be rewritten as

$$\begin{aligned}V = &\frac{1}{4}\dot{q}^T M(q)\dot{q} + \frac{1}{4}\dot{r}_f^T L_{of}^T M_o(r)L_{of}\dot{r}_f\\ &+ \sum_{j=1}^{p}(k_{pj} + \alpha k_{vj})S_j(\Delta x_j) + \frac{1}{4}(\dot{q} + 2\alpha\hat{J}_x^+(q,\hat{W}_x)s(\Delta x))^T\\ &\cdot M(q)(\dot{q} + 2\alpha\hat{J}_x^+(q,\hat{W}_x)s(\Delta x))\\ &+ \frac{1}{4}(L_{of}\dot{r}_f + 2\alpha J^+(r)J_e(q)\hat{J}_x^+(q,\hat{W}_x)s(\Delta x))^T\\ &\cdot M_o(r)(L_{of}\dot{r}_f + 2\alpha J^+(r)J_e(q)\hat{J}_x^+(q,\hat{W}_x)s(\Delta x))\\ &- \alpha^2 s^T(\Delta x)((\hat{J}_x^+(q,\hat{W}_x))^T M(q)\hat{J}_x^+(q,\hat{W}_x)\\ &+ (\hat{J}_x^+(q,\hat{W}_x))^T J_e^T(q)(J^+(r))^T\\ &\cdot M_o(r)J^+(r)J_e(q)\hat{J}_x^+(q,\hat{W}_x))s(\Delta x)\\ &+ \frac{1}{2}\sum_{j=1}^{n}\Delta W_{gj}\Gamma_{gj}\Delta W_{gj}^T + \frac{1}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\Delta W_{fij}\Gamma_{fij}\Delta W_{fij}^T\\ &+ \frac{1}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\Delta W_{xij}\Gamma_{xij}\Delta W_{xij}^T.\end{aligned} \tag{37}$$

From the above equation, we can conclude that

$$\begin{aligned}V \geq &\frac{1}{4}\dot{q}^T M(q)\dot{q} + \frac{1}{4}\dot{r}_f^T L_{of}^T M_o(r)L_{of}\dot{r}_f\\ &+ \sum_{j=1}^{p}(k_{pj}\bar{c}_j + \alpha(k_{vj}\bar{c}_j - \alpha\lambda_m))s_j^2(\Delta x_j)\\ &+ \frac{1}{2}\sum_{j=1}^{n}\Delta W_{gj}\Gamma_{gj}\Delta W_{gj}^T + \frac{1}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\Delta W_{fij}\Gamma_{fij}\Delta W_{fij}^T\\ &+ \frac{1}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\Delta W_{xij}\Gamma_{xij}\Delta W_{xij}^T\end{aligned} \tag{38}$$

where λ_m denotes the maximum bound of the matrix

$$\begin{aligned}&(\hat{J}_x^+(q,\hat{W}_x))^T M(q)\hat{J}_x^+(q,\hat{W}_x)\\ &+ (\hat{J}_x^+(q,\hat{W}_x))^T J_e^T(q)(J^+(r))^T M_o(r)J^+(r)J_e(q)\hat{J}_x^+(q,\hat{W}_x).\end{aligned}$$

As seen from (38), V is positive definite in $\Delta x, \dot{q}, \Delta W_{gk}, \Delta W_x$, and ΔW_f, when α and k_v are chosen so that

$$k_{vj}\bar{c}_j - \alpha\lambda_m > 0. \tag{39}$$

Next, differentiating the Lyapunov-like function V in (36) with respect to time, using (21) and (30), we have

$$\begin{aligned}\dot{V} = &-\dot{q}^T(\hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x))\dot{q} - \alpha s^T(\Delta x)K_p s(\Delta x)\\ &+s^T(\Delta x)(K_p+\alpha K_v)(J_x(q)-\hat{J}_x(q,\hat{W}_x))\dot{q} - y^T\Delta W_g\theta_g(q)\\ &-y^T\Delta W_f\theta_f(q)f_{\text{int}} - y^T(E_g+d+E_f f_{\text{int}}+K_f f_{\text{int}})-h\\ &-\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\Delta W_{xij}\Gamma_{xij}\dot{\hat{W}}_{xij}^T - \Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\Delta W_{fij}\Gamma_{fij}\dot{\hat{W}}_{fij}^T\\ &-\Sigma_{j=1}^{kni}\Delta W_{gj}\Gamma_{gj}\dot{\hat{W}}_{gj}^T\end{aligned} \tag{40}$$

where

$$\begin{aligned}h = \alpha\Big\{&s^T(\Delta x)(\hat{J}_x^+(q,\hat{W}_x))^T\left(-\frac{1}{2}\dot{M}(q)+S(q,\dot{q})\right)\dot{q}\\ &+s^T(\Delta x)(\hat{J}_x^+(q,\hat{W}_x))^T J_e^T(q)(J^+(r))^T\\ &\cdot\left(-\frac{1}{2}\dot{M}_o(r)+S_o(r,\dot{r})\right)L_{of}\dot{r}_f - \dot{s}^T(\Delta x)(\hat{J}_x^+(q,\hat{W}_x))^T\\ &\cdot M(q)\dot{q} - s^T(\Delta x)(\dot{\hat{J}}_x^+(q,\hat{W}_x))^T M(q)\dot{q}\\ &-\frac{d}{dt}[s^T(\Delta x)(\hat{J}_x^+(q,\hat{W}_x))^T J_e^T(q)(J^+(r))^T]M_o(r)L_{of}\dot{r}_f\Big\}.\end{aligned} \tag{41}$$

As seen from (41), since $s(\Delta x)$ is a saturation function of the position error and the estimated network weights are bounded by the projection algorithms defined in (34), there exists a constant c so that [26]

$$|h| \le \alpha c\|\dot{q}\|^2. \tag{42}$$

Using (27), (28), (32), and (33), $\dot{V}$ in (40) can be expressed as follows:

$$\begin{aligned}\dot{V} \le &-\dot{q}^T(\hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x))\dot{q} - \alpha s^T(\Delta x)K_p s(\Delta x)\\ &+s^T(\Delta x)(K_p+\alpha K_v)E_x\dot{q} - y^T(E_g+d\\ &+E_f f_{\text{int}}+K_f f_{\text{int}})-h+k_3\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\Delta W_{xij}\Gamma_{xij}\hat{W}_{xij}^T\\ &+k_2\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\Delta W_{fij}\Gamma_{fij}\hat{W}_{fij}^T + k_1\sum_{j=1}^{n}\Delta W_{gj}\Gamma_{gj}\hat{W}_{gj}^T\end{aligned} \tag{43}$$

where $k_{p\max} = \lambda_{\max}[K_p]$ and $k_{v\max} = \lambda_{\max}[K_v]$. Here, $\lambda_{\max}[K_p]$ and $\lambda_{\max}[K_v]$ denote the maximum eigenvalues of K_p and K_v, respectively. Since E_x is bounded, let b_{ex} be the upper bound of E_x, then

$$\begin{aligned}&s^T(\Delta x)(K_p+\alpha K_v)E_x\dot{q}\\ &\quad\le \frac{1}{2}b_{ex}(k_{p\max}+\alpha k_{v\max})(\|s(\Delta x)\|^2+\|\dot{q}\|^2).\end{aligned} \tag{44}$$

Next, we note that the following inequalities hold for the last three terms of (43):

$$\begin{aligned}k_1\Delta W_{gj}\Gamma_{gj}\hat{W}_{gj}^T &\le \frac{k_1 l_{gj}}{2}(\|W_{gj}\|^2 - \|\Delta W_{gj}\|^2)\\ k_2\Delta W_{fij}\Gamma_{fij}\hat{W}_{fij}^T &\le \frac{k_2 l_{fij}}{2}(\|W_{fij}\|^2 - \|\Delta W_{fij}\|^2)\\ k_3\Delta W_{xij}\Gamma_{xij}\hat{W}_{xij}^T &\le \frac{k_3 l_{xij}}{2}(\|W_{xij}\|^2 - \|\Delta W_{xij}\|^2).\end{aligned} \tag{45}$$

Substituting inequalities (42) and (45) into (43) and using (44) yields

$$\begin{aligned}\dot{V} \le &-\left(\lambda_1 - \alpha c - \frac{1}{2}b_{ex}(k_{p\max}+\alpha k_{v\max})\right)\|\dot{q}\|^2\\ &-\left(\alpha k_{p\min} - \frac{1}{2}b_{ex}(k_{p\max}+\alpha k_{v\max})\right)\|s(\Delta x)\|^2\\ &-y^T(E_g+d+E_f f_{\text{int}}+K_f f_{\text{int}}) - \frac{k_1 l_{gj}}{2}\Sigma_{i=1}^{n}\|\Delta W_{gj}\|^2\\ &-\frac{k_3 l_{xij}}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\|\Delta W_{xij}\|^2 - \frac{k_2 l_{fij}}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\|\Delta W_{fij}\|^2\\ &+\frac{k_1 l_{gj}}{2}\Sigma_{i=1}^{n}\|W_{gj}\|^2 + \frac{k_3 l_{xij}}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\|W_{xij}\|^2\\ &+\frac{k_2 l_{fij}}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\|W_{fij}\|^2\end{aligned} \tag{46}$$

where λ_1 denotes the lower bound of $\lambda_{\min}[\hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x)]$ and $k_{p\min} = \lambda_{\min}[K_p]$. The terms $\lambda_{\min}[\hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x)]$ and $\lambda_{\min}[K_p]$ denote the minimum eigenvalues of $\hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x)$ and K_p, respectively. When $\bar{k}_{ij}$ is chosen sufficiently large so that $\bar{k}_{ij} > |E_{fij}|$, we have

$$\begin{aligned}\dot{V} \le &-\left(\lambda_1 - \alpha c - \frac{1}{2}b_{ex}(k_{p\max}+\alpha k_{v\max})\right)\|\dot{q}\|^2\\ &-\left(\alpha k_{p\min} - \frac{1}{2}b_{ex}(k_{p\max}+\alpha k_{v\max})\right)\|s(\Delta x)\|^2\\ &-y^T(E_g+d) - \frac{k_1 l_{gj}}{2}\Sigma_{i=1}^{n}\|\Delta W_{gj}\|^2\\ &-\frac{k_3 l_{xij}}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\|\Delta W_{xij}\|^2 - \frac{k_2 l_{fij}}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\|\Delta W_{fij}\|^2\\ &+\frac{k_1 l_{gj}}{2}\Sigma_{i=1}^{n}\|W_{gj}\|^2 + \frac{k_3 l_{xij}}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\|W_{xij}\|^2\\ &+\frac{k_2 l_{fij}}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\|W_{fij}\|^2.\end{aligned} \tag{47}$$

Next, note that

$$\begin{aligned}\dot{q}^T(E_g+d) &\le \frac{1}{2}(\|\dot{q}\|^2 + \|E_g+d\|^2)\\ \alpha s^T(\Delta x)(\hat{J}_x^+(q,\hat{W}_x))^T(E_g+d) &\le \frac{\alpha b_{jx}}{2}(\|s(\Delta x)\|^2 + \|E_g+d\|^2)\end{aligned} \tag{48}$$

where b_{jx} is the upper bound of $(\hat{J}_x^+(q,\hat{W}_x))^T$. Since $y = \dot{q} + \alpha\hat{J}_x^+(q,\hat{W}_x)s(\Delta x)$, we have

$$\begin{aligned}\dot{V} \le &-\left(\lambda_1 - \alpha c - \frac{1}{2}b_{ex}(k_{p\max}+\alpha k_{v\max}) - \frac{1}{2}\right)\|\dot{q}\|^2\\ &-\left(\alpha k_{p\min} - \frac{1}{2}b_{ex}(k_{p\max}+\alpha k_{v\max}) - \frac{\alpha b_{jx}}{2}\right)\|s(\Delta x)\|^2\\ &-\frac{k_1 l_{gj}}{2}\Sigma_{i=1}^{n}\|\Delta W_{gj}\|^2 - \frac{k_3 l_{xij}}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\|\Delta W_{xij}\|^2\\ &-\frac{k_2 l_{fij}}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\|\Delta W_{fij}\|^2 + \mu\end{aligned} \tag{49}$$

where

$$\mu = \frac{1+\alpha b_{jx}}{2}\|E_g+d\|^2 + \frac{k_1 l_{gj}}{2}\Sigma_{i=1}^{n}\|W_{gj}\|^2 + \frac{k_3 l_{xij}}{2}\Sigma_{j=1}^{kni}\Sigma_{i=1}^{p}\|W_{xij}\|^2 + \frac{k_2 l_{fij}}{2}\Sigma_{i=1}^{n_I}\Sigma_{j=1}^{kni}\|W_{fij}\|^2. \quad (50)$$

From the upper bound of V, there exist K_v, K_p, and a positive constant $\bar{\gamma}_1$, such that

$$\left(\lambda_1 - \alpha c - \frac{1}{2}b_{ex}(k_{p\,\max} + \alpha k_{v\,\max}) - \frac{1}{2}\right)\|\dot{q}\|^2 \geq \bar{\gamma}_1(\dot{q}^T M(q)\dot{q} + \dot{r}_f^T L_{of}^T M_o(r) L_{of}\dot{r}_f)$$

$$\left(\alpha k_{p\,\min} - \frac{1}{2}b_{ex}(k_{p\,\max} + \alpha k_{v\,\max}) - \frac{\alpha b_{jx}}{2}\right)\|s(\Delta x)\|^2 \geq \sum_{j=1}^{p}\left(k_{pj}\bar{c}_j + \alpha(k_{vj}\bar{c}_j + \frac{1}{2}\alpha\lambda_m)\right)s_j^2(\Delta x_j) \quad (51)$$

where we note that b_{ex} is a small bound of the NN approximation error. Let $\bar{\gamma} = \min\{\bar{\gamma}_1, k_1, k_2, k_3\}$, then one has

$$\dot{V} \leq -\bar{\gamma}V + \mu. \quad (52)$$

We are now ready to state the following theorem.

Theorem: The closed-loop system described by (35) with the adaptive Jacobian matrices updated by (32) and (33), gives rise to the uniformly ultimately boundedness of $(\Delta x, \dot{q})$ and boundedness of $(f_{\rm int} - f_d)$ as $t \to \infty$ if the feedback gains K_p and K_v are chosen to satisfy conditions (39) and (51).

Proof: Refer to the Appendix.

Remark: In this paper, three NNs are used in the proposed adaptive NN controller. This is due to the presence of Jacobian uncertainties in kinematics and force, in addition to the dynamic uncertainty. As compared to the adaptive NN control system of robots with only dynamic uncertainty in the literature, it is therefore harder to tune the NN controllers. However, this is natural as there are more uncertainties presence in the system. All the NN weights are updated online by the update laws (32) and (33), but it is well known in NN control that the main difficulty in implementation is to initialize the weights to obtain a good transient performance. Since the system models are unknown and nonlinear, there is no systematic way of initializing the NN weights or tuning the controller gains. However, it is interesting to observe that infants also are not able to manipulate an object with unknown length and mass (i.e., kinematics and dynamics uncertainties) initially, but they are able to do it steadily after many trials and failures. Future work would be devoted to develop an adaptive learning algorithm for adjusting the NN weights by repeating the operations.

V. Simulation Results

In this section, simulation results are presented to illustrate the performance of the proposed NN controller for multifingered robots. We consider two planar fingers with three degrees of freedom grasping an object moving in a horizontal plane, as shown in Fig. 6. A fixed camera is placed at a distance away from the fingers. The position vector r of the object in Cartesian space is defined as

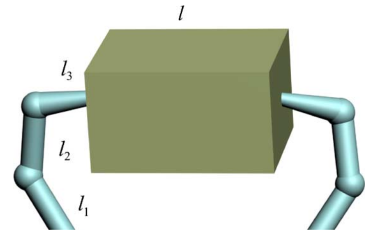

Fig. 6. Two three-link fingers grasping an object.

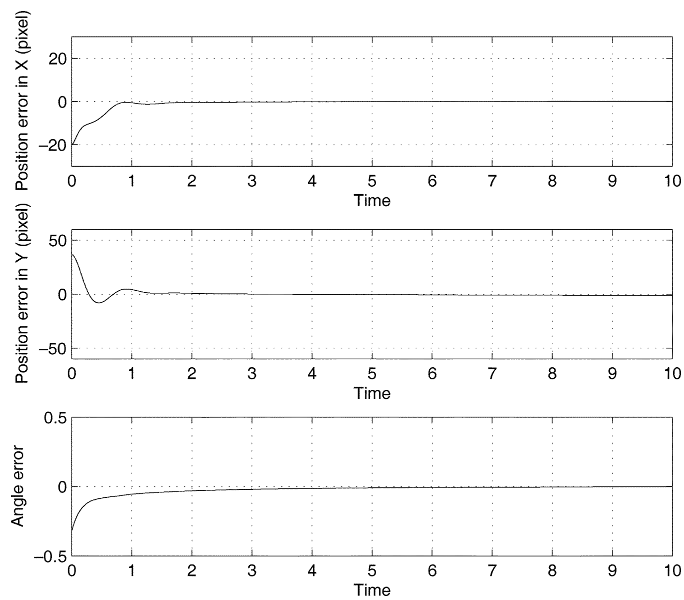

Fig. 7. Position errors with adaptive Jacobians.

$$r = [r_x, r_y, \theta]^T \quad (53)$$

where r_x and r_y are position variables on the x- and y-axis, respectively, and θ is the orientation angle of the object with respect to x-axis, $\theta = q_1 + q_2 + q_3$. Similarly, the task-space vector is defined as

$$x = [x_1, x_2, \theta]^T \quad (54)$$

where x_1 and x_2 are position variables in pixels on the two axes of image plane.

In the simulation, the exact masses of the three links were all set to 0.1 kg, the exact lengths l_1, l_2, and l_3 of the links were all set to 0.05 m, the focal length f_1 was chosen as 50 mm, the perpendicular distance between the robot and the camera z was chosen as 0.55 m, and the scaling factors that denoted the number of pixels per unit length were set as $\beta_1 = \beta_2 = 10\,000$ pixels/m.

The fingers were required to manipulate the object from the initial position [100, 87] pixels to the desired position [120, 50] pixels. The initial and desired orientation angles were set as 0 and $\pi/10$. The desired force was set as $[9.5, 3.1, 0]^T$. The initial values of the adaptive Jacobian matrices were set as constant matrices. The control gains were set as $\alpha = 0.5$, $K_m = 0.3I$, $\gamma = 8$, $\bar{k}_{ij} = 0.001$, and $k_2 = k_3 = 0.001$. The feedback gain

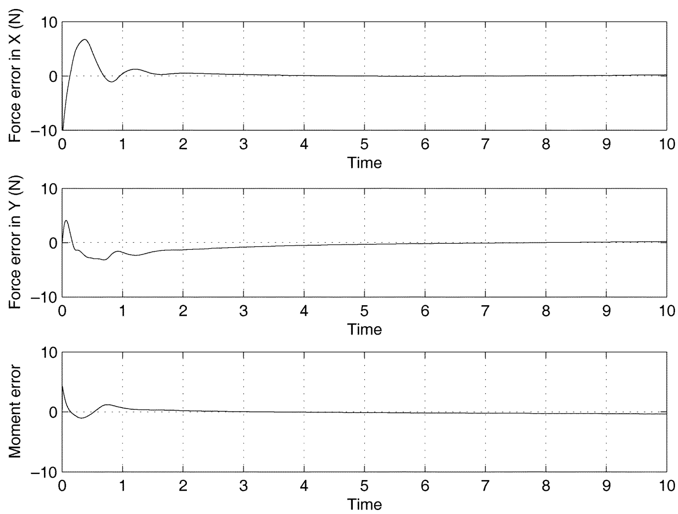

Fig. 8. Force errors with adaptive Jacobians.

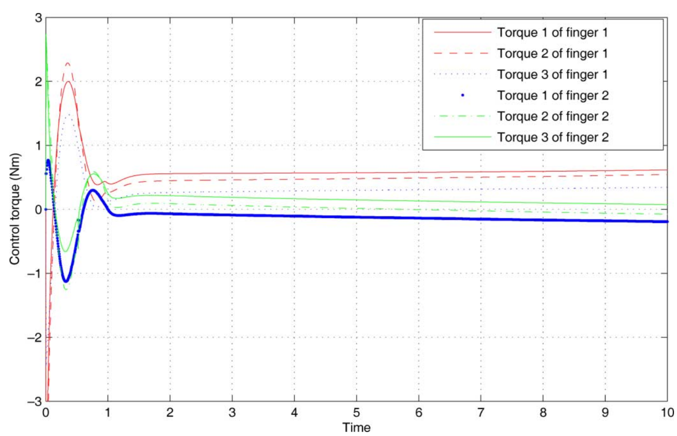

Fig. 9. Control inputs.

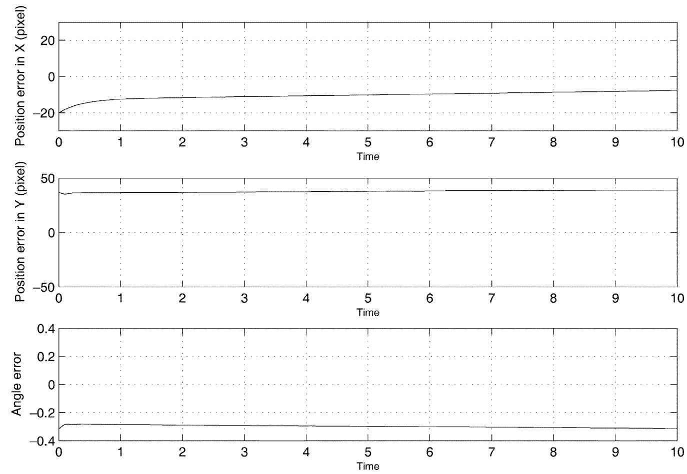

Fig. 10. Position errors without adaptive Jacobians.

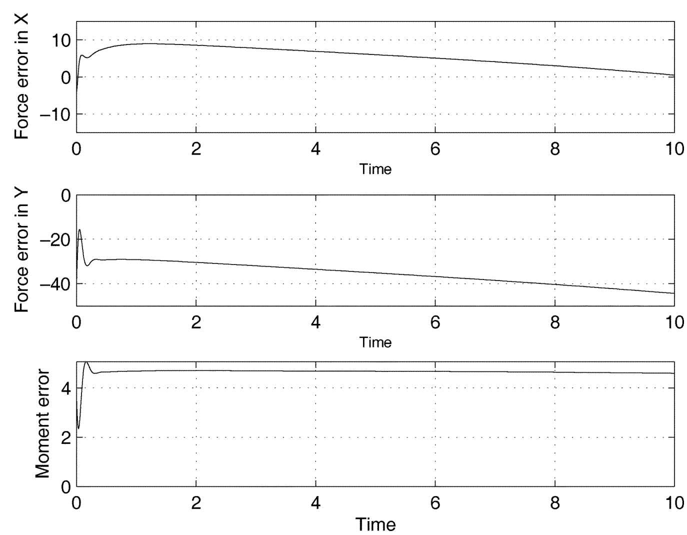

Fig. 11. Force errors without adaptive Jacobians.

matrices were set as $K_p = \mathrm{diag}\{5 \times 10^{-4}, 5 \times 10^{-4}, 5\}$ and $K_v = \mathrm{diag}\{2 \times 10^{-4}, 2 \times 10^{-4}, 20\}$.

In this simulation, Gaussian RBF NNs were used and the centers were chosen so that they were evenly distributed to span the input space of the network. The distance was fixed at $\pi/15$ and the number of neurons was set as 18. The gains for the networks were chosen as $\Gamma_f = 20I$ and $\Gamma_x = 50I$. The simulation results showing the object position errors and the finger force errors are plotted in Figs. 7 and 8, respectively. Fig. 9 shows the control inputs during the objection manipulation.

Existing controllers in the literature can only deal with dynamic uncertainty and hence these controllers may fail in the presence of kinematic uncertainty. To illustrate this point, another simulation was performed using the controller in [26] without using NN Jacobians. In this controller, fixed Jacobians with estimated kinematic parameters $\hat{l_1} = 0.052$ m, $\hat{l_2} = 0.048$ m, and $\hat{l_3} = 0.052$ m was used. The results are shown in Figs. 10 and 11. As seen from the results, this control system without using NN Jacobians failed in dealing with the uncertainties.

VI. Conclusion

In this paper, an adaptive NN Jacobian controller has been proposed for multifingered robot hands with unknown kinematics, dynamics, and Jacobian matrices. A new Lyapunov-like function has also been presented for the stability analysis of the multifingered robot control system. Sufficient conditions for choosing the feedback gains to guarantee the stability have been presented. It is shown that uniform ultimate boundedness of the position error can be achieved in presence of the aforementioned uncertainties. Simulation results have been presented to illustrate the performance of the proposed control law.

Appendix

In this Appendix, the proof of the theorem is presented. From inequality (52), we have

$$V \leq \frac{\mu}{\bar{\gamma}} + \left\{ V(0) - \frac{\mu}{\bar{\gamma}} \right\} e^{-\bar{\gamma} t} \tag{55}$$

and hence

$$V \leq \frac{\mu}{\bar{\gamma}} \tag{56}$$

as $t \rightarrow \infty$. From (38), note that V is positive definite in $s(\Delta x)$, $\dot{q}$, ΔW_{gk}, ΔW_x, and ΔW_f. Hence, from inequality (56), it can be concluded that Δx, $\dot{q}$, $\dot{r}_f$, ΔW_{gj}, ΔW_{fij}, and ΔW_{xij} are uniformly ultimately bounded. It is clear that the bounds of the object position error and the NN weights estimation errors can, therefore, be derived from (38) and (56). Substituting (9)–(11) into (35), we have

$$\begin{aligned}(M(q)+\bar{M}_o)\ddot{q} &+ \left(\frac{1}{2}\dot{M}(q)+S(q,\dot{q})+\frac{1}{2}\dot{\bar{M}}_o+\bar{S}_o\right)\dot{q}\\ &+\hat{J}_x^T(q,\hat{W}_x)K_p s(\Delta x)+\hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x)\dot{q}\\ &+\Delta W_g\theta_g(q)+W_f\theta_f(q)f_{\mathrm{int}}+E_g+d+E_f f_{\mathrm{int}}+K_f f_{\mathrm{int}}\\ =\hat{J}_m^T(q,\hat{W}_f)&RK_m\left(\Delta f+\gamma\int_o^t \Delta f(\tau)\mathrm{d}\tau\right)\end{aligned} \tag{57}$$

where $\bar{M}_o = J_e^T(q)(J^+(r))^T M_o(r)J^+(r)J_e(q)$ and $\bar{S}_o = J_e^T(q)(J^+(r))^T S_o(r,\dot{r})J^+(r)J_e(q)$ and $\Delta f = f_{\mathrm{int}} - f_d$. Next, let $M(q)+\bar{M}_o = \bar{M}$ and multiplying both sides of (57) by $J_m(q)\bar{M}^{-1}$ and using $J_m(q)\ddot{q} = -\dot{J}_m(q)\dot{q}$, the force error is subject to

$$\Delta(t) = \Delta f + \gamma(K_m - \bar{E})^{-1}K_m\int_o^t \Delta f(\tau)\mathrm{d}\tau \tag{58}$$

where

$$\begin{aligned}\Delta(t) &= (K_m-\bar{E})^{-1}\kappa r(t),\\ \bar{E} &= J_m(q)\bar{M}^{-1}(\Delta W_f\theta_f(q)+E_f+K_f)\\ \kappa &= R^T(J_m(q)\bar{M}^{-1}\hat{J}_m^T(q,\hat{W}_f))^{-1}\\ r(t) &= -\dot{J}_m(q)\dot{q}+J_m(q)\bar{M}^{-1}\\ &\quad\cdot\left\{\left(\frac{1}{2}\dot{M}(q)+S(q,\dot{q})+\frac{1}{2}\dot{\bar{M}}_o+\bar{S}_o\right)\dot{q}\right.\\ &\quad+\hat{J}_x^T(q,\hat{W}_x)K_p s(\Delta x)+\hat{J}_x^T(q,\hat{W}_x)K_v\hat{J}_x(q,\hat{W}_x)\dot{q}\\ &\quad\left.+\Delta W_g\theta_g(q)+\Delta W_f\theta_f(q)f_d+E_g+d+E_f f_d+K_f f_d\right\}.\end{aligned} \tag{59}$$

Since the object position error, velocity, and weight estimation errors are uniformly ultimately bounded, $\Delta(t)$ is also bounded. In order to prove the boundedness of the force error, let $z = \int_o^t \Delta f(\tau)\mathrm{d}\tau$, then from (58), we have

$$\|z\| \leq e^{-\lambda_\gamma t}\|z(0)\| + \int_o^t e^{-\lambda_\gamma \tau}\|\Delta(\tau)\|d\tau \tag{60}$$

where λ_γ denotes the lower bound of $\gamma(K_m - \bar{E})^{-1}K_m$. Therefore, $t \rightarrow \infty$, and we obtain

$$\|z\| \leq \frac{\mu_1}{\lambda_\gamma} \tag{61}$$

where μ_1 is the upper bound of $\Delta(t)$. Finally, from (61) and (58), we can obtain the norm bound for the force error as

$$\|\Delta f\| \leq \mu_2 \tag{62}$$

where $\mu_2 = \mu_1 + b_\gamma\mu_1/\lambda_\gamma$ and b_γ denotes the upper bound of $\gamma(K_m - \bar{E})^{-1}K_m$. $\triangle\triangle\triangle$

REFERENCES

[1] A. Bicchi, "Hands for dexterous manipulation and robust grasping: A difficult road toward simplicity," *IEEE Trans. Robot. Autom.*, vol. 16, no. 6, pp. 652–662, Dec. 2000.

[2] S. Arimoto, "Intelligent control of multi-fingered hand," *Annu. Rev. Control*, vol. 28, no. 1, pp. 75–85, 2004.

[3] T. Yoshikawa and K. Nagai, "Manipulating and grasping forces in manipulation by multifingered robot hands," *IEEE Trans. Robot. Autom.*, vol. 7, no. 1, pp. 67–77, Feb. 1991.

[4] X. Z. Zheng, R. Nakashima, and T. Yoshikawa, "On dynamic control of finger sliding and object motion in manipulation with multifingered hands," *IEEE Trans. Robot. Autom.*, vol. 16, no. 5, pp. 469–481, Oct. 2000.

[5] Z. Li, P. Hsu, and S. S. Sastry, "Grasping and coordinated manipulation by a multifingered robot hand," *Int. J. Robot. Res.*, vol. 18, no. 4, pp. 33–50, 1989.

[6] B. Yao and M. Tomizuka, "Adaptive coordinated control of multiple manipulators handling a constrained object," in *Proc. IEEE Conf. Robot. Autom.*, 1993, pp. 624–629.

[7] T. Naniwa, S. Arimoto, and V. P. Vega, "A model-based adaptive control scheme for coordinated control of multiple manipulators," in *Proc. IEEE/RSJ/GI Int. Conf. Intell. Robots Syst.*, 1994, pp. 695–702.

[8] D. Sun and J. Mills, "Manipulating rigid payloads with multiple robots using compliant gripper," *IEEE/ASME Trans. Mechatronics*, vol. 7, no. 1, pp. 23–34, Mar. 2002.

[9] S. Arimoto, P. T. A. Nguyen, H. Han, and Z. Doulgeri, "Dynamics and control of a set of dual fingers with soft tips," *Robotica*, vol. 18, no. 1, pp. 71–80, 2000.

[10] R. Ozawa, S. Arimoto, S. Nakamura, and J.-H. Bae, "Control of an object with parallel surfaces by a pair of finger robots without object sensing," *IEEE Trans. Robot.*, vol. 21, no. 5, pp. 965–976, Oct. 2005.

[11] C. C. Cheah, H. Y. Han, S. Kawamura, and S. Arimoto, "Grasping and position control for multi-fingered robot hands with uncertain Jacobian matrices," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1998, pp. 2403–2408.

[12] F. L. Lewis, J. Huang, T. Parisini, D. V. Prokhorov, and D. C. Wunsch, "Guest editorial special issue on neural networks for feedback control systems," *IEEE Trans. Neural Netw.*, vol. 18, no. 4, pp. 969–972, Jul. 2007.

[13] R. M. Sanner and J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Netw.*, vol. 3, no. 6, pp. 837–863, Nov. 1992.

[14] Y. H. Kim and F. L. Lewis, "Neural network output feedback control of robot manipulator," *IEEE Trans. Robot. Autom.*, vol. 15, no. 2, pp. 301–309, Apr. 1999.

[15] H. D. Patino, R. Carelli, and B. R. Kuchen, "Neural networks for advanced control of robot manipulators," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 343–354, Mar. 2002.

[16] R. Carelli, E. F. Camacho, and D. Patino, "A neural network based feedforward adaptive controller for robots," *IEEE Trans. Syst. Man Cybern.*, vol. 25, no. 9, pp. 1281–1288, Sep. 1995.

[17] W. T. Miller, "Real-time application of neural networks for sensor-based control of robots with vision," *IEEE Trans. Syst. Man Cybern.*, vol. 19, no. 4, pp. 825–831, Jul./Aug. 1989.

[18] F. L. Lewis, "Neural network control of robot manipulators," *Intell. Syst. Their Appl.*, vol. 11, no. 3, pp. 64–75, 1996.

[19] F. L. Lewis, K. Liu, and A. Yesildirek, "Neural net robot controller with guaranteed tracking performance," *IEEE Trans. Neural Netw.*, vol. 6, no. 3, pp. 703–715, May 1995.

[20] F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer neural-net robot controller with guaranteed tracking performance," *IEEE Trans. Neural Netw.*, vol. 7, no. 2, pp. 388–399, Mar. 1996.

[21] S. J. Yoo, J. B. Park, and Y. H. Choi, "Adaptive output feedback control of flexible-joint robots using neural networks: Dynamic surface design approach," *IEEE Trans. Neural Netw.*, vol. 19, no. 10, pp. 1712–1726, Oct. 2008.

[22] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators*. Singapore: World Scientific, 1998.

[23] S. Jagannathan and G. Galan, "Adaptive critic neural network-based object grasping control using a three-finger gripper," *IEEE Trans. Neural Netw.*, vol. 15, no. 2, pp. 395–407, Mar. 2004.

[24] S. Haykin, *Neural Networks- A Comprehensive Foundation*. New York: Prentice-Hall, 1999.

[25] T. Hayakawa, W. M. Haddad, and N. Hovakimyan, "Neural network adaptive control for a class of nonlinear uncertain dynamical systems with asymptotic stability guarantees," *IEEE Trans. Neural Netw.*, vol. 19, no. 1, pp. 80–89, Jan. 2008.
[26] S. Arimoto, *Control Theory of Nonlinear Mechanical Systems- A Passivity-Based and Circuit-Theoretic Approach*. Oxford, U.K.: Clarendon, 1996.
[27] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Trans. Robot. Autom.*, vol. 12, no. 5, pp. 651–670, Oct. 1996.
[28] Y. Nakamura, *Advanced Robotics*. Reading, MA: Addison-Wesley, 1985.
[29] R. M. Murray, S. S. Sastry, and Z. Li, *A Mathematical Introduction to Robotic Manipulation*. Reading, MA: Addison-Wesley, 1994.
[30] L. E. Weiss, A. C. Sanderson, and C. P. Neuman, "Dynamic sensor-based control of robots with visual feedback," *IEEE J. Robot. Autom.*, vol. 3, no. 5, pp. 404–417, Oct. 1987.
[31] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Robot. Autom.*, vol. 8, no. 3, pp. 313–326, Jun. 1992.
[32] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*. New York: Prentice-Hall, 1989.

Yu Zhao was born in China. He received the B.Eng. degree in automation from Beijing University of Aeronautics & Astronautics, Beijing, China, in 2002 and the Ph.D. degree in electrical engineering from Nanyang Technological University, Nanyang, Singapore, in 2007.

Currently, he is an I&C System Engineer at OGS Industries, Singapore. His research interests are in adaptive vision-force control of robot manipulators and multifingered robot hands using neural networks.

Chien Chern Cheah was born in Singapore. He received the B.Eng. degree in electrical engineering from National University of Singapore, Singapore, in 1990 and the M.Eng. and Ph.D. degrees in electrical engineering from Nanyang Technological University, Singapore, in 1993 and 1996, respectively.

From 1990 to 1991, he was a Design Engineer at Chartered Electronics Industries, Singapore. He was a Research Fellow at the Department of Robotics, Ritsumeikan University, Japan, from 1996 to 1998. He joined the School of Electrical and Electronic Engineering, Nanyang Technological University, as an Assistant Professor in 1998. Since 2003, he has been an Associate Professor at Nanyang Technological University. In November 2002, he received the overseas attachment fellowship from the Agency for Science, Technology and Research (A*STAR), Singapore to visit the Nonlinear Systems Laboratory, Massachusetts Institute of Technology, Cambridge.

Dr. Cheah was the Program Chair of the 2006 International Conference on Control, Automation, Robotics and Vision. He has been serving as an associate editor of the IEEE Robotics and Automation Society Conference Editorial Board since 2007.