Proceedings of the 8th
World Congress on Intelligent Control and Automation
July 6-9 2010, Jinan, China

Reference Compensation Technique of Neural Force Tracking Impedance Control for Robot Manipulators*

Seul Jung
Department of Mechatronics Engineering
Chungnam National University,
Daejeon, Korea
jungs@cnu.ac.kr

T. C. Hsia
Electrical & Computer Engineering
University of California, Davis
Davis, CA 95616, USA
tcshsia@gmail.com

***Abstract*—In this paper, the neural impedance controller is formulated to regulate the contact force with the environment. When robot uncertainties are present, the performance of the impedance controller is degraded. To compensate for uncertainties in both robot dynamics and environment, neural network is introduced at the desired trajectory. The training signal is defined to satisfy the desired goal. This leads to the remarkable advantage of no requirement of modifying an internal force control structure. Neural network actually compensates for uncertainties at the input trajectory level in on-line fashion. The robust position and force tracking performance of a robot manipulator is confirmed by simulation studies.**

Keywords-component; neural network, robot manipulator

I. INTRODUCTION

Recently, the robot is required to interact more frequently with humans. The demanded tasks for the humanoid robot become more challenging to perform not only stable walking but also sophisticated manipulation with two arms although they concentrate more on the stable walking than two arm manipulation tasks. However, in the near future, it is for sure that the service robot should handle objects to satisfy the given task specifications [1]. This kind of interaction with the environment requires force control.

Force control has a long history in the robot research communities. From the pioneering works of the impedance control and the hybrid force control algorithms to intelligent force control algorithms, there has been active research conducted on this subject [2,3]. Modifications of the previous force control algorithms have been made to improve problems of proposed methods [4-8].

Recently, due to the trend of merging the concept of the intelligence into the system, intelligent force control methods have been considered to be an important topic. Intelligent tools known as neural network or fuzzy logic are incorporated into the robot controllers to improve the performances [9-18].

Fuzzy logic has been well known for interpretation of human expression to the system. Although fuzzy logic controllers can be easily implemented by the hardware, they suffer from the difficulty of finding right rules for the fuzzy controller. It requires a time consuming process to find optimal rules. As another intelligent tool, neural network has been well known for its capabilities such as nonlinear mapping, adaptation, and learning. Many successful results have been reported in the literature although it has the difficulty of hardware implementation.

In the force control subject, neural network algorithms have been used for industrial manipulators to deal with unknown objects as well as unknown robot dynamics [9-15]. A combined structure of the H_∞ control method and neural network has been proposed [16]. To remedy defects of fuzzy logic and neural networks, a combined structure of neural network and fuzzy logic has been proposed [17,18].

In our previous research [9], neural network is used to cancel out uncertainties. Experimental studies show that neural network works very well in on-line fashion. However, since the compensating signal is added to the control input level, the modification of an internal control structure has been required. Some robots having sealed controllers may not have a chance to be implemented with neural network.

In this paper, in the same framework of using neural network to compensate for uncertainties in robot force control systems[9,11], the different control structure is proposed. Borrowing the concept from the reference compensation technique used for position control of robot manipulators, neural network can be used outside the system by compensating signals at the desired trajectory. This provides the remarkable advantage of not requiring modification of the pre-installed controllers in many applications. Extensive simulation studies are conducted to show the feasibility of the proposed control method.

II. REVIEW OF IMPEDANCE CONTROL

The dynamic equation of an n degrees-of-freedom robot manipulator in joint space coordinates is given by

$$D(q)\ddot{q}+C(q,\dot{q})\dot{q}+G(q)=\tau-\tau_e \tag{1}$$

where the vectors q, $\dot{q}$, $\ddot{q}$ are the n x 1 joint angle, the n x 1 joint angular velocity, and the n x 1 joint angular acceleration,

*This research has been supported by the Ministry of Education, Science & Technology (MEST) through the specified basic research program (No. 2009-0059407).

respectively, $D(q)$ is the n x n symmetric positive definite inertia matrix, $C(q,\dot{q})$ is the n x 1 vector of Coriolis and centrifugal torques, $G(q)$ is the n x 1 gravitational torques, τ is the n x 1 vector of actuator joint torques, and τ_e is the n x 1 vector of external disturbance joint torques.

Denoting $h = C(q,\dot{q})\dot{q} + G(q)$ can simplify the equation (1) as

$$D(q)\ddot{q} + h = \tau - \tau_e . \tag{2}$$

The Jacobian relationship between the joint velocity and the Cartesian velocity allows to have the robot dynamic equation model in the Cartesian space. The Jacobian relationship for the acceleration is

$$\ddot{q}(t) = J^{-1}(\ddot{X} - \dot{J}\dot{q}) . \tag{3}$$

Substituting (3) into (2) yields the Cartesian dynamic equation as

$$D^*\ddot{X} + h^* = F - F_e , \tag{4}$$

where $D^* = J^{T^{-1}} D J^{-1}$, $h^* = J^{T^{-1}} h - D^* \dot{J} J^{-1} \dot{X}$, F_e is the external force, and F is the control input force.

The impedance force control method regulates the contact force by selecting impedance parameters correctly [8]. The force tracking impedance function is given by

$$F_e - F_d = M\ddot{E} + B\dot{E} + KE , \tag{5}$$

where $E = X_e - X$, X_e is the environment position, M, B, K are impedance parameters. Setting $K = 0$ in (5) and assuming $F_e = -K_e E$ yields the stable impedance function.

$$-F_d = M\ddot{E} + B\dot{E} + K_e E . \tag{6}$$

In this formulation, without knowing the exact environment stiffness K_e , the force tracking $F_e = F_d$ is guaranteed at the steady state. If the environment position X_e is not accurately available, then the impedance function with the uncertainty can be represented as

$$M\ddot{E}' + B\dot{E}' = F_e - F_d , \tag{7}$$

where $E' = E + \delta X_e$ and $\delta X_e = X_e' - X_e$. However, to achieve the force tracking $F_e = F_d$ by realizing the ideal impedance function, uncertainties in robot dynamics have to be compensated at the same time. Here we propose to use neural network to handle both uncertainties in robot dynamics and environment.

III. NEURAL NETWORK FORCE CONTROL

The force control law in the Cartesian space is given by [7,8].

$$F = \hat{D}^* V + \hat{h}^* + F_e \tag{8}$$

And the control input is

$$V(t) \approx \ddot{X}(t) \tag{9}$$

From the impedance law (5) and (6), $\ddot{X}$ can be obtained as

$$\begin{aligned} V_p &= \ddot{X}_d + M^{-1}(B\dot{\varepsilon} + K(\varepsilon + \phi_p)) \\ V_f &= \ddot{X}'_e + M^{-1}(F_d + B\dot{E} + K(E + \phi_f)) \quad \text{free space} \\ V_f &= \ddot{X}'_e + M^{-1}(F_d - F_e + B(\dot{E} + \phi_f)) \quad \text{contact space} \end{aligned} \tag{10}$$

where $\varepsilon = X_d - X$, X_d is the desired trajectory, ϕ_p, ϕ_f are outputs of neural network and $V = [V_p^T V_f^T]^T$ where V_p is for the position controlled direction and V_f is for the force controlled direction.

Substituting the control law (8) into the Cartesian robot dynamic equation (4) yields

$$\hat{D}^*(V(t) - \ddot{X}(t)) = \Delta D^* \ddot{X} + \Delta h^* , \tag{11}$$

where $\Delta D^* = D^* - \hat{D}^*$ and $\Delta h^* = h^* - \hat{h}^*$. Arranging (11) gives the error equation as

$$(V(t) - \ddot{X}(t)) = \hat{D}^{*^{-1}} (\Delta D^* \ddot{X} + \Delta h^*) . \tag{12}$$

Consider the force controllable direction separately by substituting V_f of contact in (10) into (12). Then we have

$$\ddot{X}'_e + M^{-1}(F_d - F_e + B(\dot{E}' + \phi_f)) - \ddot{X} = \hat{D}^{*^{-1}} \Delta F \tag{13}$$

where the uncertainty terms $\Delta F = \Delta D^* \ddot{X} + \Delta h^*$. Thus the equation (13) is simplified as

$$M\ddot{E}' + B\dot{E}' + F_d - F_e + B\phi_f = M\hat{D}^{*^{-1}} \Delta F . \tag{14}$$

Therefore, neural network compensating signal ϕ_f can compensate for those dynamic uncertainties ΔF as

$$M\ddot{E}' + B\dot{E}' + F_d - F_e = M\hat{D}^{*^{-1}} \Delta F - B\phi_f . \tag{15}$$

After the convergence of errors, neural network output becomes

$$\phi_f = B^{-1} M \hat{D}^{*^{-1}} \Delta F . \tag{16}$$

Then the desired impedance function can be realized as

$$M\ddot{E}' + B\dot{E}' + K_e E = -F_d\,, \quad (17)$$

where $F_e = -K_e E$.

In the same manner, we can obtain the following equation for the position controlled direction.

$$M\ddot{\varepsilon} + B\dot{\varepsilon} + K\varepsilon = M\hat{D}^{*^{-1}}\Delta F - K\phi_p\,, \quad (18)$$

where ϕ_p is a neural network output for position controlled direction. The proposed neural network force control structure is shown in Figure 1.

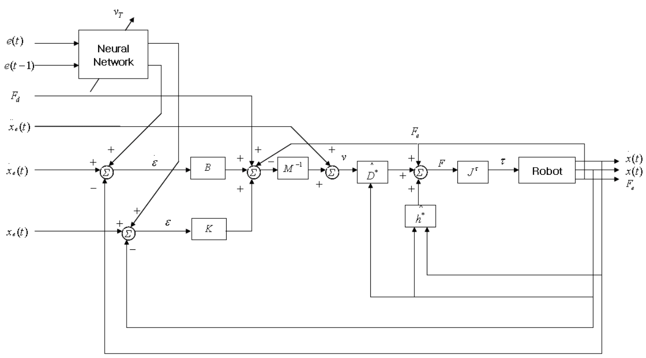

Fig. 1. Neural network force control structure

IV. Neural Network Learning Algorithm

Next is to design the training signal for the neural network. Our goal is to achieve $F_e = F_d$ in the force controlled direction. Learning algorithm is similar to those presented in[9,11]. Therefore, we can have the training signal from equation (7),

$$v_{force} = F_d - F_e\ . \quad (19)$$

Then, equation (15) by assuming that the desired trajectory is flat becomes

$$v_{force} = M\ddot{X} + B\dot{X} + M\hat{D}^{*^{-1}}\Delta F - B\phi_f\ . \quad (20)$$

If the environment position profile is not flat then good force tracking results cannot be achieved due to $\dot{X}, \ddot{X}$ terms. To overcome this problem we use neural network to cancel out those terms in (20).

In order to do that, the training signal for neural network should be designed properly so that the force error is minimized. Two separate training signals are designed: $v_{position}, v_{force}$, one for position control and the other for force control, respectively. The training signals for v_{force} are obtained from equations (10) as two cases: one in free space and the other in contact space.

$$v_{free} = \ddot{e} + \frac{1}{m}(b\dot{e} + ke + f_d)\,. \quad (21)$$

$$v_{contact} = f_d - f_e\,. \quad (22)$$

The objective function to be minimized is defined as

$$\Im = \frac{1}{2}v^T v\,, \quad (23)$$

where $v = [v_{position}^T v_{force}^T]^T$. Making use of the definition of v in equation (21) and (22) yields the gradient of as $\Im$

$$\frac{\partial \Im}{\partial w} = [\frac{\partial v}{\partial w}]^T v = -[\frac{\partial \phi}{\partial w}]^T v\ . \quad (24)$$

The neural network output is $\phi = [\phi_p^T \phi_f^T]^T$. The back-propagation update rule for the weights with a momentum term is

$$\Delta w(t) = \eta[\frac{\partial \phi}{\partial w}]^T v + \alpha \Delta w(t-1)\,, \quad (25)$$

where η is the update rate and α is the momentum coefficient.

Two layered feed-forward neural network is used as a compensator as shown in Figure 2. The output of hidden units is filtered by a nonlinear function

$$f(x) = \frac{1 - \exp(-x)}{1 + \exp(-x)}\ . \quad (26)$$

while the output units are linear. We have chosen six hidden neurons for our experiments. The back propagation algorithm parameter η is optimized and $\alpha = 0.9$ is used. Weights are randomly selected with small values.

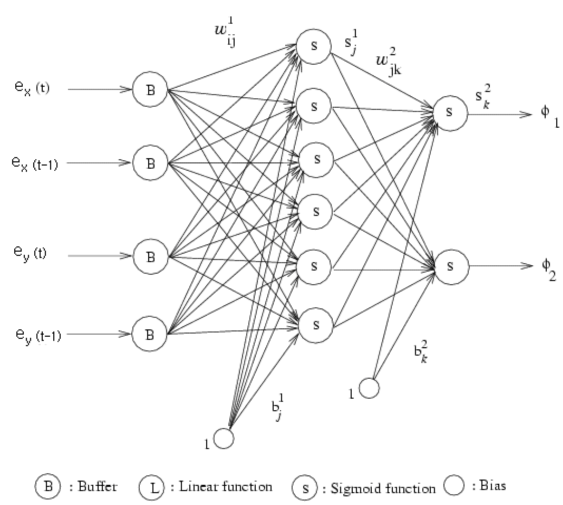

Fig. 2. Neural network structure

V. Simulation Results

A. Simulation Setsup

To validate the proposed force control algorithm, simulation studies of a two link robot manipulator are conducted. The link length is 0.4m and the weight is 5Kg for two links. Robot dynamic uncertainties include the case of which the robot dynamics model is assumed to be unknown and each joint has friction terms. The environment is assumed to have the unknown stiffness of 10,000N/m and the desired force is set to 20N. The normal force to the environment in the x axis is regulated. The sampling time is 0.005seconds. The robot is required to follow the environment in the *y* axis regulating a desired force in *x* axis as shown in Fig. 3.

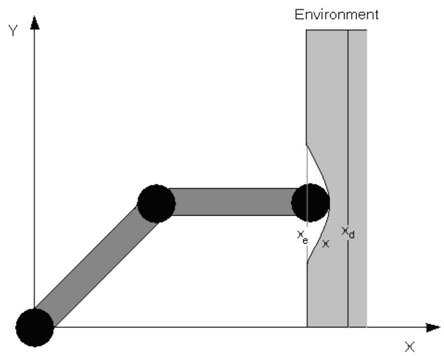

Fig. 3. Force tracking environment

B. *No Uncertainties in Robot Dynamics*

First, the robot is required to track on the environment that has the stiffness of 10,000N/m. The desired force is 20N and the force tracking result is shown in Fig. 4. Position controller gains are *kp=100* and *kd=20*, and force controller gains are *m=1* and *b=100*. From the initial position, the robot moves slowly and makes contact with the environment at about 0.13 seconds.

Without knowing the environment stiffness and the position information, the impedance force control algorithm is robust enough to perform force tracking control well. X axis position and *y* axis sinusoidal trajectory tracking performance are shown in Fig.5 and 6, respectively.

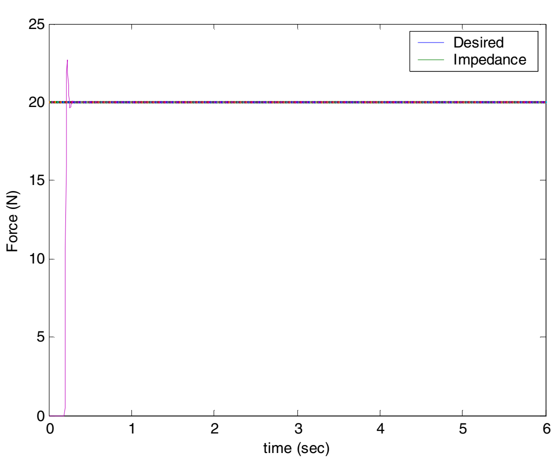

Fig. 4. Force tracking performances

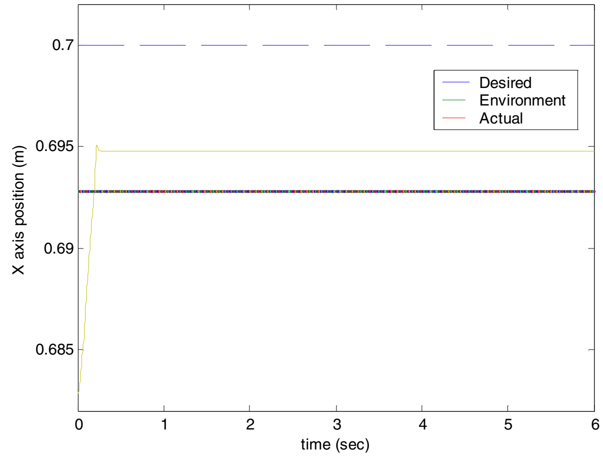

Fig. 5. X axis position tracking result

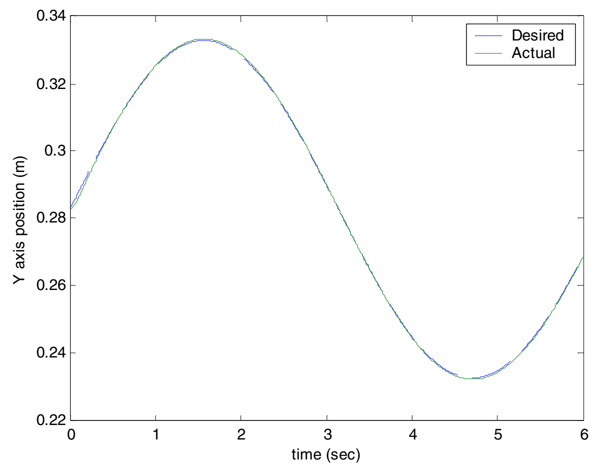

Fig. 6. Y axis position tracking result

C. *Uncertainites in Robot Dynamics*

However, when uncertainties in robot dynamics such as joint friction and model mismatches are present, the force tracking performance is much degraded as shown in Fig. 7. Position tracking result in y axis shows the notable tracking error as shown in Fig. 9. This error should be compensated to get the better force tracking performance.

After compensation for uncertainties in robot dynamics, the force tracking performance is improved as shown in Fig. 7. The force overshoot is settled down within 0.5 seconds. In Fig. 8, the corresponding position tracking results show that the environment position is estimated inside the environment. The position tracking in *y* axis is also improved by the neural network controller shown in Fig. 9. Initial tracking errors are minimized fast as the neural network learns. Figs. 10 and 11 show the neural network compensation signals for force control and position control respectively.

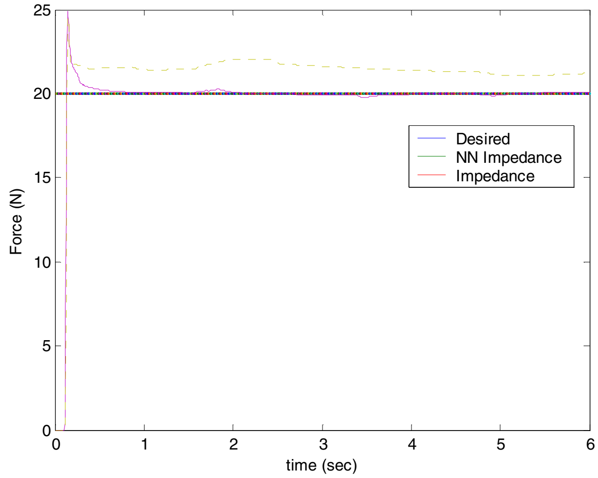

Fig. 7. Force tracking performances under robot dynamic uncertainties

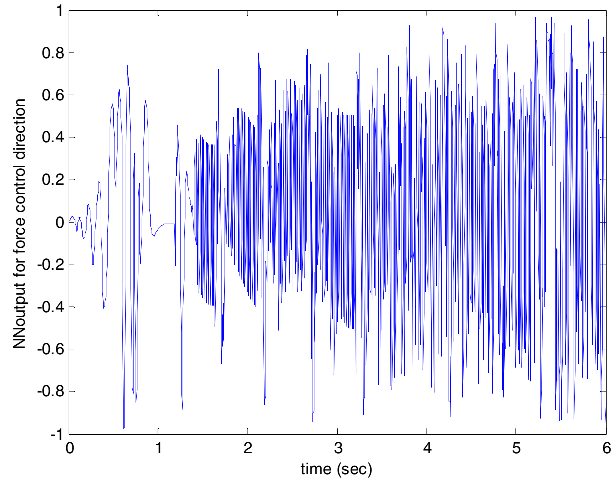

Fig. 10.Neural network compensation signal for force control direction

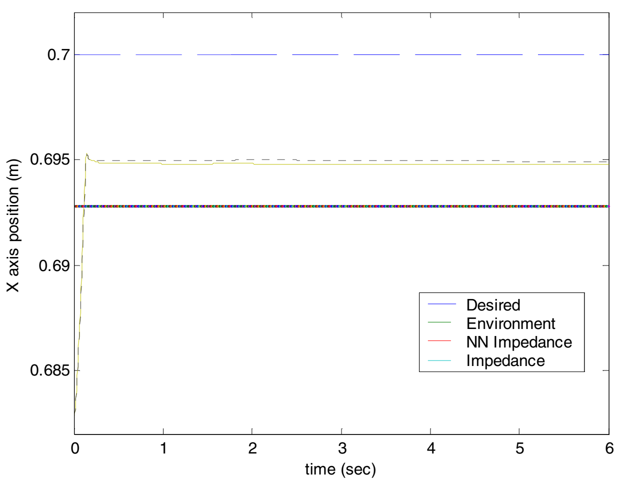

Fig. 8. X axis position tracking results under robot dynamics uncertainties

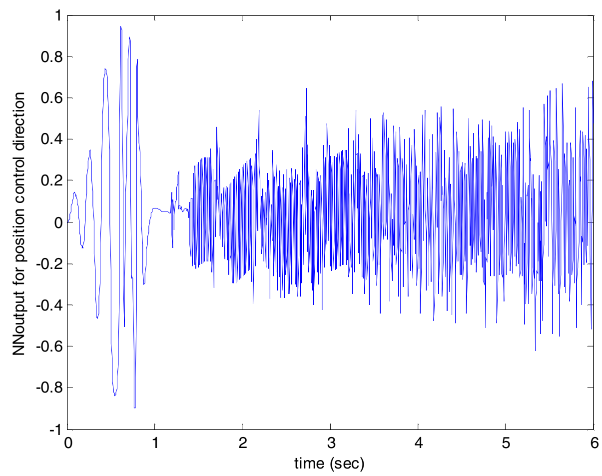

Fig. 11.Neural network compensation signal for position control direction

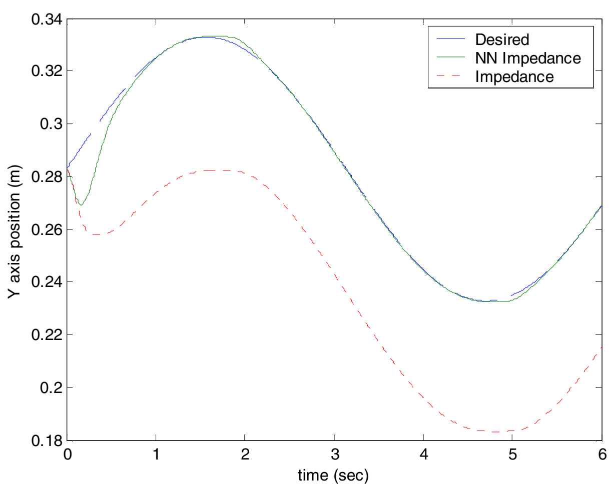

Fig. 9. Y axis position tracking results under robot dynamics uncertainties

VI. Conclusion

When the robot performs force control, performance is very much affected by the robustness of the position controller. Robot dynamics uncertainties are one of important problems to be solved in advance. Since neural network is a nonlinear controller, uncertainties can be compensated in not only the position controller but also the force controller. Performances are confirmed by extensive simulation studies.

One advantage of the proposed control scheme is that there is no change required in the pre-designed controller. Uncertainties can be compensated at the input trajectory level. This leads to many advantages in real applications such as tele-operation control, haptic devices, and other nonlinear control systems. Future research will seek advantages by applying the proposed algorithms to applications.

REFERENCES

[1] Dong Min Kim, Ho Jin Choi, Jeong Seob Kim, Woon Kyu Lee, Duck Hui Song and Seul Jung," Tracking Control of a Moving Object for Robokers with Stereo Visual Feedback", IEEE ICIT, 2007

[2] N. Hogan, "Impedance Control : An Approach to Manipulator, Part i, ii, iii", *ASME Journal of Dynamics Systems, Measurements, and Control*, Vol. 3, pp. 1-24, 1985

[3] M. Raibert and J. J. Craig, "Hybrid Position/Force Control of Manipulators", *ASME Journal. of Dynamic Systems, Measurements, and Control*, Vol. 102, pp. 126-133, 1981

[4] R. Anderson and M. W. Spong, "Hybrid Impedance Control of Robotic Manipulators", *IEEE Conference on Robotics and Automations*, pp. 1073-1080, 1987.

[5] G. J. Liu and A. A. Goldenberg, "Robust Hybrid Impedance Control of Robot Manipulators", *Proc. IEEE Conference on Robotics and Automations*, pp.287-292, 1991.

[6] H. Seraji, "Adaptive Admittance Control : An Approach to Explicit Force Control in Compliant Motion", *Proc. IEEE Conference on Robotics and Automations*, pp. 2705-2712, 1994

[7] S. Jung and T. C. Hsia, "Adaptive Force Tracking Impedance Force Control of Robot for Cutting Process", *IEEE Conference on Robotics and Automations*, pp1800-1805,1999

[8] S. Jung, T. C. Hsia and R. G. Bonitz, "Force Tracking Impedance Control of Robot Manipulators Under Unknown Environment", *IEEE Trans. on Control Systems Technology,* Vol.12, No.2, pp. 474-483, 2004

[9] S. Jung, S. B. Yim, and T. C. Hsia, "Experimental Studies of Neural Impedance Force Control of Robot Manipulators", *Proc. IEEE Conference on Robotics and Automations*, pp. 3453-3458, 2001

[10] M. Dapper, R. Maab, V. Zahn, R. Eckmiller, "Neural Force Control Applied to Industrial Manipulators in Interaction with Moving Rigid Objects", *IEEE Conference on Robotics and Automations*, 2048-2053 1998.

[11] S. Jung and T. C. Hsia, "Robust Neural Force Control Scheme under Uncertainties in Robot Dynamics and Unknown Environment", *IEEE Trans. on Industrial Electronics*, vol.47, no.2, pp. 403-412, April, 2000

[12] S. Hu, M. Ang, and H. Krishnan, "NN controller fo the constrained robot under unknown constraint", IEEE Conf. of IECON, pp.2123-2128, 2000

[13] V. Mallapragada, D. Erol, and N. Sarkar, "A new method of force control for unknown environments", IEEE IROS, pp. 4509-4514, 2006

[14] T. Zhao and C. C. Cheah, "Position and force control of robot manipulators using neural networks", IEEE Conf. on Robotics, Automation, and Mechatronics, pp. 300-305, 2004

[15] F. Passold and M. Stemmer, "Force control of a SCARA robot using neural networks", Fourth International Workshop on Robot motion, and Control, pp. 247-252, 2004

[16] M. Hwang and X. Hu, "A robust position/force learning controller of manipulators via H_∞ control and neural networks", IEEE Trans. on Systems, Man, and Cybernetics, 30, 2, pp. 310-321, 2000

[17] K. Kiguchi and T. Fukuda, "Intelligent position/force controller for industrial robot manipulators-application of fuzzy neural networks", *IEEE Trans. on Industrial Electronics*, **44**, **6**, pp. 753- 761, 1997

[18] K. Kiguchi, K. Watanabe, K. Izumi, and T. Fukuda, "Application of multiple fuzzy-neuro force controllers in an unknown environment using genetic algorithms", *IEEE Conf. on Robotics and Automation*, pp. 2106-2111, 2000