International Journal of Dynamics and Control
https://doi.org/10.1007/s40435-018-0487-y

# Intelligent controller for hybrid force and position control of robot manipulators using RBF neural network

**Komal Rani**[1,2] **· Naveen Kumar**[1]

Received: 29 March 2018 / Revised: 2 October 2018 / Accepted: 19 October 2018



## Abstract

In this paper, an intelligent controller is developed for hybrid force and position control of robot manipulators in the presence of external disturbances and the model uncertainties. The proposed controller consists of a model based controller and neural network based model free controller with an adaptive bound part. A non linear function of model dynamics is identified by employing a radial basis function neural network. The role of adaptive bound part is to estimate the bounds on model disturbances, friction term and neural network reconstruction error. The Lyapunov function candidate is used to prove the stability of the proposed controller and to show that the errors are asymptotically convergent. Finally numerical simulation results are presented for two link robot manipulator to show excellent performance of the proposed controller in comparison to other control schemes such as model based computed torque control and neural network based model free controller.


**Keywords** Hybrid force and position control · Constrained robot manipulator · Disturbances · Neural network · Asymptotically stable

## 1 Introduction

There are many industrial applications such as grinding, fine polishing, contour following, deburring and assembly in which the robot end effector comes in the contact with the environment. During the execution of such types of tasks, the control objective is to regulate the contact forces exerted by manipulator end effector on the environment while simultaneously regulating the position of end effector. Such type of control problem i.e. the hybrid force and position control of robotic manipulators has been a subject of interest in the literature of control system. Raibert and Craig [1] developed the hybrid position/force controller to control manipulator end effector trajectory by using force sensor. Lozano and Brogliato [2] presented a hybrid position/force controller to control the reduntant mechanical robot manipulator. By extending the work of [1] Yoshikawa and Sudou [3] developed a dynamic hybrid control problem for constrained robot with an undefined constraint surface. Kwan [4] developed a robust adaptive sliding control method for force/motion control of constrained manipulator with motor dynamics. Kouya et al. [5] introduced an adaptive position/force controller by utilizing the strict feedback back-stepping method which depends on passivity. De Queiroz et al. [6] proposed an adaptive controller for rigid robot manipulator to control force and position during constrained motion without velocity measurement. Cheah et al. [7] proposed an adaptive force/motion controller which worked well without the exact knowledge of Jacobian matrix, dynamics and kinematics. Filaretov and Zuev [8] developed a control scheme without force sensors to control force and position of rigid robot manipulator. Kouya [9] designed a hybrid force/position controller for constrained manipulators with uncertainty. Roy and Whitcomb [10] designed an adaptive force control method for robot arm which comes into contact with surface of unknown linear compliances. In order to control hybrid position/force control of robot manipulators, Pliego-Jimenez and Arteaga-Perez [11] developed the hybrid position/force controller for robot manipulator in contact with uncertain constraint surface. In this scheme measured force end effector position are used to estimates the constraint surface. The calculation of complex regression matrices are required for these techniques, which is very tedious task. In recent times, neu-

✉ Naveen Kumar
navindma@gmail.com

1 Department of Mathematics, National Institute of Technology Kurukshetra, Kurukshetra, Haryana 136119, India

2 Department of Mathematics, Babu Anant Ram Janta College Kaul, Kaithal, Haryana 136021, India

Published online: 27 October 2018

ral networks have achieved popularity among community of control system due to their learning ability. The neural network based controllers are being used to learn uncertainties or unknown non linear function. There are many researches in the literature of control system that take the benefit of NN to compensate the uncetainties [12,13]. Fanaei and Farrokhi [14] designed the robust and adaptive neuro-fuzzy controller to compensate the friction force between the end effector and the surface. Liu et al. [15] designed the force controller for robot manipulator based on fuzzy prediction for reference trajectory in the impedence model. Karayiannidis et al. [16] proposed an adaptive controller for robotic manipulator to deal with the problem of force and position control in a compliant surface. Bechlioulis et al. [17] presented NN based adaptive force and position controller with the uncertainties in the dynamical system. Kumar et al. [18] proposed an adaptive control method to deal with the problem of hybrid force/position control of robotic manipulators. Feedforward neural network is used to compensate the uncertainties existing in the dynamical system. Singh and Sukavanam [19] extended the work of [18] and developed a NN based robust adaptive controller by including external disturbances and achieved the stability of system. Mahjoub et al. [20] developed an adaptive sliding mode control using radial basis function neural network for underactuated robotic manipulators. Li et al. [21] designed a hybrid position/force controller based on neural network for constrained reconfigurable manipulators. Lee and Wang [22] developed the robust adaptive force and position tracking controller using fuzzy neural network for robot manipulators under unknown environment. Ghajar et al. [23] introduced hybrid position and force controller for constrained manipulators with contact friction between the robot end effector and surface. De Oliveira et al. [24] presented a dual mode control method for adjusting the RBF weights. Rani and Kumar [25] designed a neural network based hybrid force and position controller for constrained robot manipulators. But their controller may face singularity issue. In the present paper, controller scheme is designed to address the singularity issue of [25] and to improve the performance of the controller.

The novelty of the paper is the design of an asymptotically stable controller for hybrid force and position control of robot manipulators by utilizing model based techniques and neural network based model free technique. Generally, all neural network based hybrid force and position controllers require no information regarding system dynamics. But in practical situation, we are always available with partial information. For effective controller, it would be better to use this information. In the present paper, utilizing partial information about the system dynamics, we have proposed a controller consisting of the merits of model based technique and RBF neural network based model free technique. The disturbances, unstructured uncertainties and network approximation errors are compensated with the adaptive bound part of the controller. The numerical simulations are performed to prove the better performance of the proposed controller, i.e. a hybrid controller consisting of model based controller together with RFNNN based model free controller, as compare to model based controller and FFNN based model free controller.

The paper is structured as: In Sect. 2, the details about the dynamical model and its decomposition is presented. The radial basis function neural network based controller is developed in Sect. 3. Section 4, deals with the stability analysis. Simulation studies is discussed in details in Sect. 5 and the concluding remarks are presented in Sect. 6.

## 2 Dynamics of constrained robot manipulators

The general equation of motion for n link revolute constrained robot is expressed as [19]

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

where $B(q) \in R^{n\times n}$ symbolizes the inertia matrix. $C_m(q, \dot{q}) \in R^{n\times n}$ stands for the centripetal-coriolis matrix. $G(q) \in R^n$ represents the gravity effects. $F_f(\dot{q}) \in R^n$ stands for the friction effects. $D \in R^n$ is a bounded disturbance. $\tau \in R^n$ is the torque input vector and $\tau_e \in R^n$ is the interaction torque.

The link position, velocity and acceleration are denoted by $q$, $\dot{q}$ and $\ddot{q}$ respectively.

**Property 1** The matrix $B(q)$ is symmetric and positive definite and satisfies the following inequality:

$$a_1||\xi||^2 \leq \xi^T B(q)\xi \leq a_2||\xi||^2 \qquad \forall \xi \in R^n \tag{2}$$

where $a_1$ and $a_2$ are positive constant.

**Property 2** The matrix $(\dot{B}(q) - 2C_m(q, \dot{q}))$ is skew symmetric matrix i.e for any $z \in R^n$, we have

$$z^T(\dot{B}(q) - 2C_m(q, \dot{q}))z = 0 \tag{3}$$

Some useful assumptions for the development of the controller are taken as:

**Assumption 1** We assume that $(||F_f(\dot{q})|| \leq b + c||\dot{q}||)$, where $b$ and $c$ are positive constants.

**Assumption 2** $(||D|| \leq e)$, where $e$ is positive constant.

The relation between interaction torque $(\tau_e)$ and interaction force $(\bar{F} \in R^m)$ is defined as follows

$$\tau_e = O^T(q)\bar{F} \tag{4}$$

where Jacobian matrix $O(q)$ belongs to $R^{m\times n}$.

The kinematic equation of robot is defined by

$$x = f(q) \tag{5}$$

where position vector of the end-effector is $x \in R^m$ and the kinematic transformation of robot is $f(q)$.

By taking derivative of Eq. (5), we have

$$\dot{x} = O(q)\dot{q} \tag{6}$$

In the task space, force ($\bar{F} \in R^m$) and environment deformation i.e. $(x - x_c)$ are related by

$$\bar{F} = \bar{S}(x - x_c) \tag{7}$$

where ($\bar{S} \in R^{m\times m}$) is constant stiffness matrix and $x_c \in R^m$ is the point of contact.

Equation (7) can be rearranged as:

$$\bar{F} = \begin{bmatrix} F \\ F' \end{bmatrix} = \begin{bmatrix} S \\ S' \end{bmatrix}(x - x_c) \tag{8}$$

where $F \in R^l$, $F' \in R^{n-l}$, $S \in R^{l\times m}(l \leq m)$ and $l$ is the dimension of subspace where force can be controlled. In this way, we can control $(m - l)$ cartesian position, $l$ forces and $(n - m)$ redundant joints velocities.

### 2.1 Decomposition of dynamical model

In this subsection, we decompose the robot dynamics (1) into position $S^-x$, force $F$ and redundant joint subspace $O^-q$ using the identities:

$$I = O^+O + O^- \tag{9}$$
$$I = S^+S + S^- \tag{10}$$

where

$$O^+ = O^T[OO^T]^{-1} \tag{11}$$
$$S^+ = S^T[SS^T]^{-1} \tag{12}$$
$$O^- = I - O^T[OO^T]^{-1}O \tag{13}$$
$$S^- = I - S^T[SS^T]^{-1}S \tag{14}$$

where $O^+$ and $S^+$ are penrose pseudo inverses and $O^-$, $S^-$ are idempotent matrices.

Using Eqs. (9) and (5), $\dot{q}$ is decomposed as

$$\dot{q} = O^+O\dot{q} + O^-\dot{q} = O^+\dot{x} + O^-\dot{q} \tag{15}$$

Using Eqs. (10) and (8), $\dot{x}$ is decomposed as

$$\dot{x} = S^+S\dot{x} + S^-\dot{x} = S^+\dot{F} + S^-\dot{x} \tag{16}$$

Differentiating Eq. (6), we have

$$\ddot{x} = O\ddot{q} + \dot{O}\dot{q} \tag{17}$$

Taking Eqs. (9) and (17), $\ddot{q}$ can be decomposed as

$$\ddot{q} = O^+O\ddot{q} + O^-\ddot{q} = O^+(\ddot{x} - \dot{O}\dot{q}) + O^-\ddot{q} \tag{18}$$

Differentiating Eq. (16), we get

$$\ddot{x} = S^+S\ddot{x} + S^-\ddot{x} = S^+\ddot{F} + S^-\ddot{x} \tag{19}$$

Using Eqs. (15–19) in Eq. (1), we get

$$\begin{aligned} &B(O^+(S^+\ddot{F} + S^-\ddot{x} - \dot{O}\dot{q}) + O^-\ddot{q}) \\ &\quad + C_m(O^+(S^+\dot{F} + S^-\dot{x}) \\ &\quad + O^-\dot{q}) + G(q) + F_f(\dot{q}) + D = \tau - \tau_e \end{aligned} \tag{20}$$

Equations (20) and (1) are equivalent equations and clearly Eq. (20) is a decomposition of position, force and redundant joints velocity.

## 3 Control system development

In this section, we design the control scheme. Let us define the filtered tracking error $r$ as

$$r = O^+[S^+(\dot{F} - v_f) + S^-(\dot{x} - v_x)] - O^-\dot{\tilde{q}} \tag{21}$$

where

$$v_f = \dot{F}_s - \lambda\tilde{F}, v_x = \dot{x}_s - \lambda\tilde{x} \tag{22}$$
$$\tilde{F} = F - F_s, \tilde{x} = x - x_s, \tilde{q} = q - q_s \tag{23}$$

where $\lambda > 0$ and $F_s, x_s, q_s$ are desired values of $F, x, q$.

Now differentiating Eq. (21) and using Eqs. (22) and (23), we can write Eq. (20) in terms of $r$ as

$$B\dot{r} + C_m r = \tau - \tau_e - h(y) - F_f - D \tag{24}$$

where $h(y) = B(O^+[S^+\dot{v_f} + S^-\dot{v_x} - \dot{O}\dot{q}] + O^-\ddot{q_d} - z)$ $+ C_m(O^+[S^+v_f + S^-v_x] + O^-\dot{q_d}) + G$ is a non linear function. Here $z = \frac{dO^+}{dt}[S^+(\dot{F} - v_f) + S^-(\dot{x} - v_x)] + \frac{dO^-}{dt}\dot{\tilde{q}}$. We write the function $h(y)$ as summation of two parts one corresponding to known dynamics and another corresponding to unknown dynamics as follows

$$h(y) = \hat{h}(y) + \tilde{h}(y) \tag{25}$$

where

$$\hat{h}(y) = \hat{B}(O^{+}[S^{+}\dot{v_f} + S^{-}\dot{v_x} - \dot{O}\dot{q}] + O^{-}\ddot{q_d} - z) + \hat{C}_m(O^{+}[S^{+}v_f + S^{-}v_x] + O^{-}\dot{q_d}) + \hat{G} \quad (26)$$

and

$$\tilde{h}(y) = \tilde{B}(O^{+}[S^{+}\dot{v_f} + S^{-}\dot{v_x} - \dot{O}\dot{q}] + O^{-}\ddot{q_d} - z) + \tilde{C}_m(O^{+}[S^{+}v_f + S^{-}v_x] + O^{-}\dot{q_d}) + \tilde{G} \quad (27)$$

and the vector $y$ is given by $y = [r^T, \bar{F}^T, q^T, \dot{q}^T]$.

### 3.1 Radial basis function neural network (RBFNN)

RBF neural network is utilized to identify a non linear function of model dynamics and it is defined as follows:

$$\tilde{h}(y) = W^T\zeta(y) + \varepsilon(y) \quad (28)$$

where $W \in R^{P\times n}$ symbolizes the matrix of weights of neural network, $\varepsilon(y) : R^{3n+1} \to R^n$ denotes NN reconstruction error with $||\varepsilon(y)|| \le \varepsilon_P$ for some $\varepsilon_P$ and $\zeta(y) : R^{3n+1} \to R^P$ is a smooth basis function defined as

$$\zeta_i(y) = exp\left(-\frac{||(y-c_i)||^2)}{\sigma_i^2}\right) \qquad i = 1, 2, \ldots P \quad (29)$$

where $c_i \in R^{3n+1}$ and $\sigma_i \in R$ are centers and width of nodes and $P$ is the no. of nodes of NN.

Using Eqs. (25) and (28) in Eq. (24), we get

$$B\dot{r} = -C_m r + \tau - \tau_e - \hat{h}(y) - W^T\zeta(y) - \varepsilon(y) - F_f - D \quad (30)$$

### 3.2 Adaptive bound

In this subsection, we use Assumptions 1, 2 and bound of $\varepsilon(y)$, such that:

$$||F_f(\dot{q}) + D + \varepsilon(y)|| \le b + c||\dot{q}|| + e + \varepsilon_P \quad (31)$$

Now we define $\beta = b + c||\dot{q}|| + e + \varepsilon_P$ as an adaptive bound and it can be rewritten as

$$\beta = \begin{bmatrix} 1 & ||\dot{q}|| & 1 & 1 \end{bmatrix}\begin{bmatrix} b & c & e & \varepsilon_P \end{bmatrix}^T = Q^T(||\dot{q}||)\phi \quad (32)$$

where $Q \in R^k$ is known vector of joint velocity and $\phi \in R^k$ is a parameter vector, where $k$ is positive fixed number.
To achieve the desired control objective, the controller is proposed as:

$$\tau = \tau_e + \hat{h}(y) - K_g r + \hat{W}^T\zeta(y) - \frac{\hat{\beta}^2 r}{\hat{\beta}||r|| + \delta} \quad (33)$$

where gain matrix $K_g \in R^{n\times n}$ is a positive matrix, $\hat{\beta} = Q^T\hat{\phi}$, $\hat{\phi}$ and $\hat{W}$ are estimated values of parameter vector and neural network weights respectively and $\dot{\delta} = -\gamma\delta$, $\delta(0)$ = designed constant $> 0$, $\gamma > 0$.

Using $\tau$ given by Eq. (33) in Eq. (30), we get

$$B\dot{r} = -(K_g + C_m)r - \tilde{W}^T\zeta(y) - (F_f + D + \varepsilon(y)) - \frac{\hat{\beta}^2 r}{\hat{\beta}||r|| + \delta} \quad (34)$$

where $\tilde{W} = W - \hat{W}$

## 4 Stability analysis

In this section, we analysis the system stability using following theorem.

**Theorem** *If the control input $\tau$ given by Eq. (33) for the robot dynamics (1) and adaptation laws for NN weights and adaptive bound are selected as given by Eqs. (35) and (36) respectively, then the controller is stable as well as the filtered tracking error $r$ and henceforth the position error $S^{-}x$, force error $\tilde{F}$ and the redundant velocity error $O^{-}\dot{\tilde{q}}$ converges to zero as $t \to \infty$*

$$\dot{\hat{W}} = -\Gamma_W\zeta(y)r^T \quad (35)$$

$$\dot{\hat{\phi}} = \Gamma_\phi Q||r|| \quad (36)$$

*where $\Gamma_W = {\Gamma_W}^{-1} \in R^{N\times N}$, $\Gamma_\phi = {\Gamma_\phi}^{-1} \in R^{k\times k}$*

***Proof*** Consider the following Lyapunov function candidate:

$$L = \frac{1}{2}r^T Br + \frac{1}{2}\mathrm{tr}(\tilde{W}^T\Gamma_W^{-1}\tilde{W}) + \frac{1}{2}\mathrm{tr}(\tilde{\phi}^T{\Gamma_\phi}^{-1}\tilde{\phi}) + \frac{\delta}{\gamma} \quad (37)$$

where $\tilde{W} = W - \hat{W}$ and where $\tilde{\phi} = \phi - \hat{\phi}$

Taking derivative of the Lyapunov function w.r.t, we have

$$\dot{L} = \frac{1}{2}r^\top\dot{B}r + r^\top B\dot{r} + \mathrm{tr}(\tilde{W}^\top\Gamma_W\dot{\tilde{W}}) + \mathrm{tr}(\tilde{\phi}^\top\Gamma_\phi\dot{\tilde{\phi}}) + \frac{\dot{\delta}}{\gamma} \quad (38)$$

Now using the Eq. (34) and the facts $\dot{\tilde{W}} = -\dot{\hat{W}}$, $\dot{\tilde{\phi}} = -\dot{\hat{\phi}}$ and $\dot{\delta} = -\delta\gamma$, we get the following expression

$$\dot{L} = \frac{1}{2}r^T\dot{B}r + r^T\Big(-K_g r - C_m r - \tilde{W}^T\zeta(y) - (F_f + D + \varepsilon(y)) - \frac{\hat{\beta}^2 r}{\hat{\beta}||r|| + \delta}\Big) - \mathrm{tr}(\tilde{W}^T\Gamma_W^{-1}\dot{\hat{W}}) - \mathrm{tr}(\tilde{\phi}^T{\Gamma_\phi}^{-1}\dot{\hat{\phi}}) - \delta \quad (39)$$

Using Eqs. (35) and (36) in Eq. (39), we have

$$\dot{L} = \frac{1}{2}r^T\dot{B}r - r^TK_gr - r^TC_mr - r^T\tilde{W}^T\zeta(y) - r^T(F_f + D + \varepsilon(y)) - \frac{\hat{\beta}^2r^Tr}{\hat{\beta}||r|| + \delta} + \mathrm{tr}(\tilde{W}^T\Gamma_W^{-1}\Gamma_W\zeta(y)r^T - \mathrm{tr}(\tilde{\phi}^T\Gamma_\phi^{-1}\Gamma_\phi Q||r||) - \delta \tag{40}$$

$$\dot{L} = \frac{1}{2}r^T(\dot{B} - 2C_m)r - r^TK_gr - r^T(F_f + D + \varepsilon(y)) - \frac{\hat{\beta}^2||r||^2}{\hat{\beta}||r|| + \delta} - \mathrm{tr}(\tilde{\phi}^TQ||r||) - \delta \tag{41}$$

Using Property 2, we have

$$\dot{L} = -r^TK_gr - r^T(F_f + D + \varepsilon(y)) - \frac{\hat{\beta}^2||r||^2}{\hat{\beta}||r|| + \delta} - \mathrm{tr}(\tilde{\phi}^TQ||r||) - \delta \tag{42}$$

Now using Eqs. (31) and (32) , we find that

$$-r^T(F_f + D + \varepsilon(y)) \le ||r||||F_f + D + \varepsilon(y)|| \le ||r||\beta = ||r||Q^T\phi = ||r||Q^T(\tilde{\phi} + \hat{\phi}) \tag{43}$$

Using inequality given by Eq. (43) in Eq. (42), we get

$$\dot{L} \le -r^TK_gr + ||r||Q^T(\tilde{\phi} + \hat{\phi}) - \frac{\hat{\beta}^2||r||^2}{\hat{\beta}||r|| + \delta} - \mathrm{tr}(\tilde{\phi}^TQ||r||) - \delta \tag{44}$$

Since $\hat{\beta} = Q^T\hat{\phi}$, we have the following expression from Eq. (44)

$$\dot{L} \le -r^TK_gr + Q^T\tilde{\phi}||r|| + Q^T\hat{\phi}||r|| - \frac{(Q^T\hat{\phi})^2||r||^2}{Q^T\hat{\phi}||r|| + \delta} - \mathrm{tr}(\tilde{\phi}^TQ||r||) - \delta \tag{45}$$

$$\dot{L} \le -r^TK_gr + Q^T\hat{\phi}||r|| - \frac{(Q^T\hat{\phi})^2||r||^2}{Q^T\hat{\phi}||r|| + \delta} - \delta \tag{46}$$

$$\dot{L} \le -r^TK_gr - \frac{\delta^2}{Q^T\phi||r|| + \delta} \tag{47}$$

$$\dot{L} \le -\alpha_{min}||r||^2 \tag{48}$$

where $\alpha_{min}$ is minimum singular value of matrix $K_g$. Since we have $L(r(t), \tilde{W}, \tilde{\phi}) > 0$ and Eq. (48) gives $\dot{L}(r(t), \tilde{W}, \tilde{\phi}) \le 0$, from this we have $r, \tilde{W}$ and $\tilde{\phi}$ and hence $\hat{W}, \hat{\phi}$ ) are bounded. Now we define a new function $\theta(t) = \alpha_{min}||r||^2 \le -\dot{L}$ and taking integration of this w.r.t time, we have

$$\int_0^t \theta(t)dt \le L(0) - L(t) = L(r(0), \tilde{W}, \tilde{\phi}) - L(r(t), \tilde{W}, \tilde{\phi}) \tag{49}$$

Since $L(r(0), \tilde{W}, \tilde{\phi})$ is bounded function and $L(r(t), \tilde{W}, \tilde{\phi})$ being bounded and decreasing function, we find that

$$\lim_{t\to\infty} \int_0^t \theta(t)dt < \infty \tag{50}$$

Further using Eqs. (21) and (7), we get

$$r = O^+[S^+(S\dot{x} - v_f) + S^-(\dot{x} - v_x)] + O^-\dot{\tilde{q}} \tag{51}$$

Using Eq. (10), we have

$$r = O^+[\dot{x} - S^+v_f - S^-v_x] + O^-\dot{\tilde{q}} \tag{52}$$

$$r = O^+\dot{x} - O^+(S^+v_f + S^-v_x) + O^-\dot{q} - O^-\dot{q}_d \tag{53}$$

Using Eq. (15), we have

$$r = \dot{q} - O^+(S^+v_f + S^-v_x) - O^-\dot{q}_d \tag{54}$$

Since we have considered that the joints are revolute, therefore $O^+, O^-, S^+, S^-, F, x$ are bounded. Therefore, $v_f$ and $v_x$ given by Eq. (22) are bounded. Thus $q_d$ and $\dot{q}$ are bounded. If we take the derivative of $r$ in Eq. (54) and using Property 1, one can find that $\dot{r}$ and hence $\dot{\theta}(t)$ is bounded. It shows that $\dot{\theta}(t)$ is uniform continuous and from Barbalat's Lemma [26] $\lim_{t\to\infty}\theta(t) = 0 \Rightarrow r \to 0$ as $t \to \infty$. This proves that the system is asymptotically stable. Now, multiplying Eq. (21) by $SO$ , we have

$$SOr = \dot{F} - v_f \to 0 \tag{55}$$

Now multiplying Eq. (21) by $O$, we get

$$Or = S^-(\dot{x} - v_x) \to 0 \tag{56}$$

Using Eqs. (21), (54) and (55), we get

$$O^-\dot{\tilde{q}} \to 0 \tag{57}$$

Thus, taking into consideration Eqs. (55), (56), (22) and (23), we have $\dot{\tilde{F}} + \lambda\tilde{F} \to 0$ and $S^-\dot{\tilde{x}} + \lambda S^-\tilde{x} \to 0$. Hence the position error $S^-\tilde{x}$, force error $\tilde{F}$ as well as redundant joints velocity $O^-\dot{\tilde{q}}$ tends to zero. □

## 5 Simulation studies

In this section, the simulation results are provided for a two link rigid robot manipulator whose mathematical model is expressed as

$$\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} C_{m11} & C_{m12} \\ C_{m21} & C_{m22} \end{bmatrix}\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} + \begin{bmatrix} F_{f1} \\ F_{f2} \end{bmatrix} + \begin{bmatrix} D_1 \\ D_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} + O^T \bar{F} \tag{58}$$

where

$$\begin{aligned}
B_{11} &= l_2^2 m_2 + l_1^2(m_1 + m_2) + 2l_1 l_2 m_2 \cos(q_2)\\
B_{12} &= l_2^2 m_2 + l_1 l_2 m_2 \cos(q_2)\\
B_{21} &= l_2^2 m_2 + l_1 l_2 m_2 \cos(q_2)\\
B_{22} &= l_2^2 m_2\\
C_{m11} &= -l_1 l_2 m_2 \sin(q_2)\dot{q}_2\\
C_{m12} &= -l_1 l_2 m_2 \sin(q_2)(\dot{q}_1 + \dot{q}_2)\\
C_{m21} &= l_1 l_2 m_2 \sin(q_2)\dot{q}_1\\
C_{m22} &= 0\\
G_1 &= l_1 g_0 (m_1 + m_2)\cos(q_1) + l_2 m_2 g_0 \cos(q_1 + q_2)\\
G_2 &= l_2 m_2 g_0 \cos(q_1 + q_2)\\
F_f(q) &= 5.3\dot{q}_1 + 2.4\dot{q}_2\\
D_1 &= 80\cos(2t) \text{ and } D_2 = 80\sin(2t)
\end{aligned}$$

The parameters given as $\hat{m}_1 = 15.61kg$, $\hat{m}_2 = 11.36kg$, $\hat{l}_1 = 0.432m$, $\hat{l}_2 = 0.432m$ are masses and lengths of link 1 and 2 respectively with $g_0 = 9.8m/s^2$ .

We take the contact force $F = \kappa(x - x_{1c})$ in the $x_1$ direction where $\kappa = 10^4$ and $x_{1c} = 0.61m$.

The desired trajectory is taken as $x_{2s} = 0.5[1 - exp(-t)]$ in $x_2$ direction.

The desired force is taken as $F_s = 20[1 - 0.5\sin(t)]$.

Also, $S^+ = \frac{1}{\kappa}$, $S^- = \begin{bmatrix} 0\ 0\ ;\ 0\ 1 \end{bmatrix}$, $O^+ = O^{-1}$ and $O^- = 0$. The controller parameters are $\lambda = 100$, $K_g = 200I_2$.

RBFNN is constructed using 10 nodes, the central positions $c_i$ takes values from $[-2, 2]^7$ and the spread factors are taken as $\sigma_i = 0.2$. The matrices in the weight updates law are $\Gamma_W = 10I_{10}$ and $\Gamma_\phi = 5I_4$.

The simulation is exposed for 10 seconds. To show the effectiveness of the proposed control scheme, the simulation studies have been performed with three different controllers i.e. computed torque type model based controller [12], FFNN based model free controller [18] and the proposed controller , i.e. a hybrid controller consisting of model based controller together with RFNNN based model free controller, in a comparative manner. The Figs. 1 and 2 are produced to show the tracking of desired and actual force in Fig. 1 as well force tracking in Fig. 2 respectively, with all the three different controllers. The end effector trajectory tracking is shown in Fig. 3 with all the three controllers. From these figures, it is clear that the CT controller is unable to deal with uncertainties and disturbances, FFNN based model free controller achieves the desired tracking but better performance is achieved with the proposed controller as is clear from the comparison of the Figs. 1, 1 and 3. From these figures, one can conclude that the performance is enhanced with the proposed controller. The joint trajectories and the joint velocities with

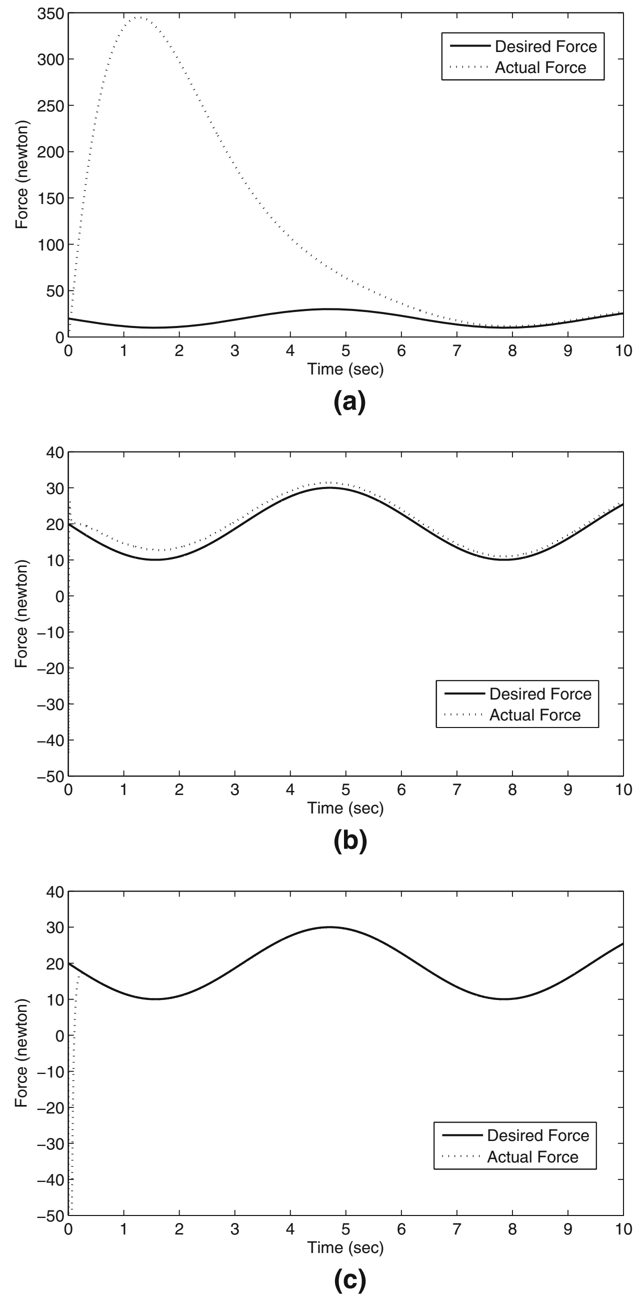


**Fig. 1** Desired and actual force tracking with **a** model based CT controller, **b** FFNN based model free controller, **c** proposed controller

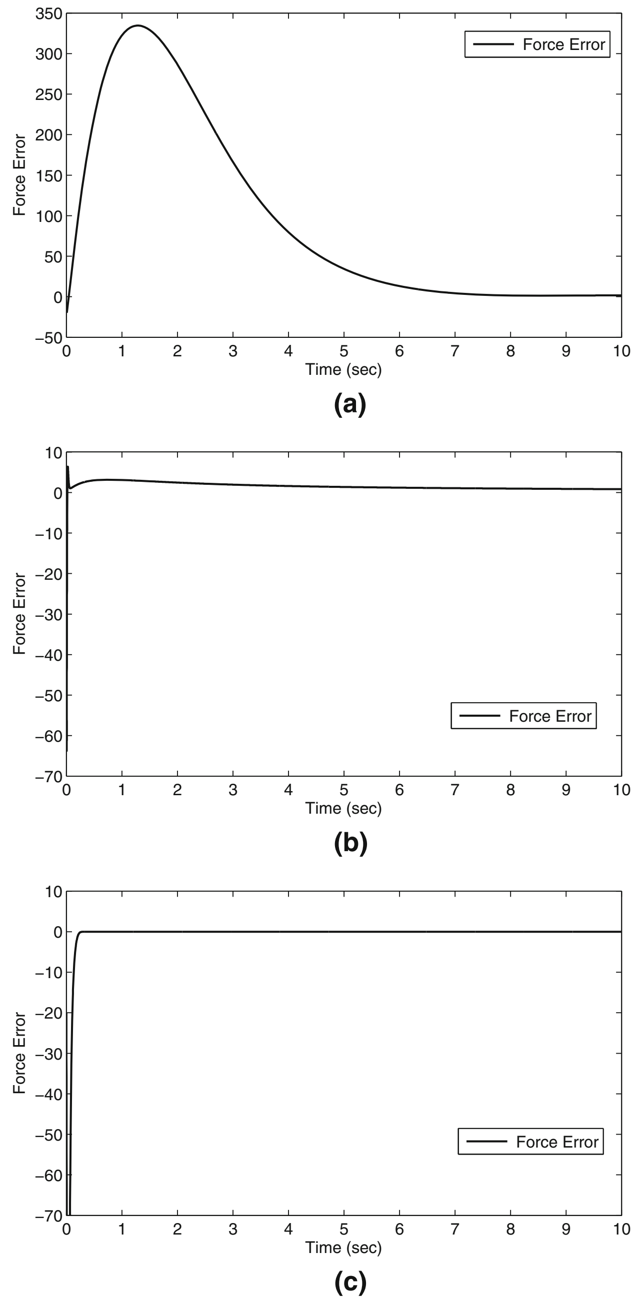


**Fig. 2** Force tracking error with **a** model based CT controller, **b** FFNN based model free controller, **c** proposed controller

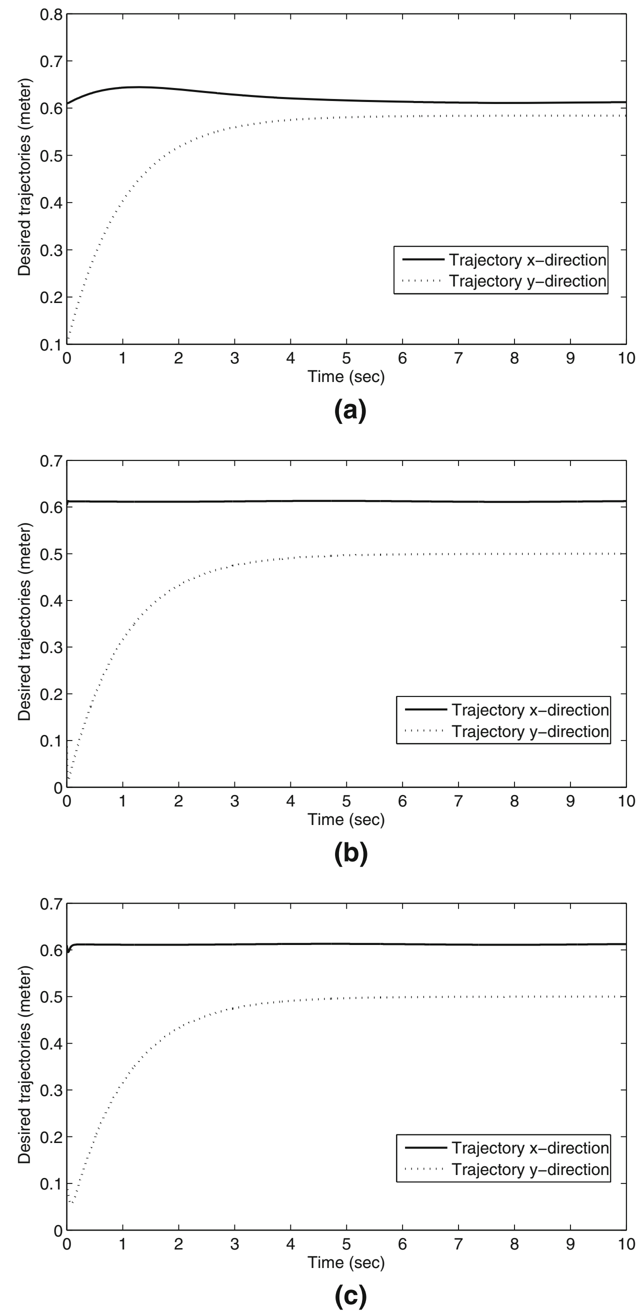


**Fig. 3** End effector trajectory tracking with **a** model based CT controller, **b** FFNN based model free controller, **c** proposed controller

all the three controllers are shown by Figs. 4 and 5 respectively.

## 6 Conclusion

In this paper, a new intelligent control scheme is proposed for hybrid force and position control of robot manipulators. The control scheme utilized the merits of both the model based computed torque type control and neural network based model free control techniques. The controller achieves not only the tracking of desired internal forces but also the tracking of end-effector trajectory effectively. The controller learns the uncertainties in online manner with the help of RBF neural network. The unknown bounds on uncertainties and the neural network approximation error are compensated effectively with the adaptive bound part of the controller. The Lyapunov stability theory is used to show that the proposed

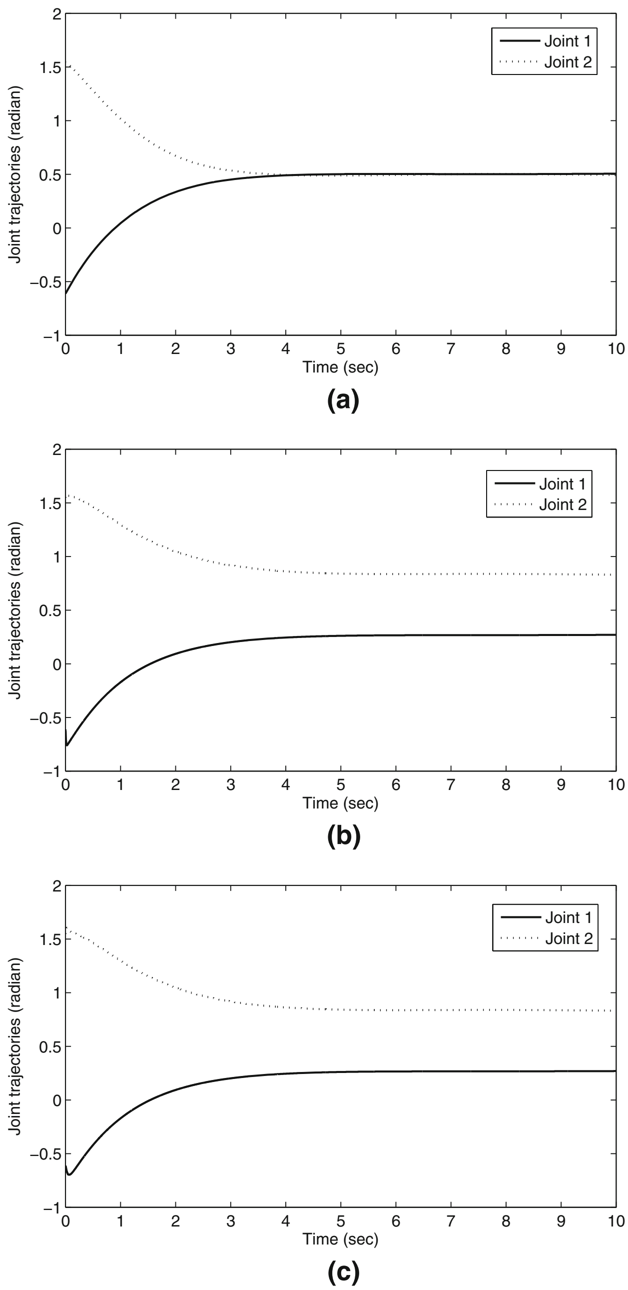


**Fig. 4** Joint trajectories with **a** model based CT controller, **b** FFNN based model free controller, **c** proposed controller

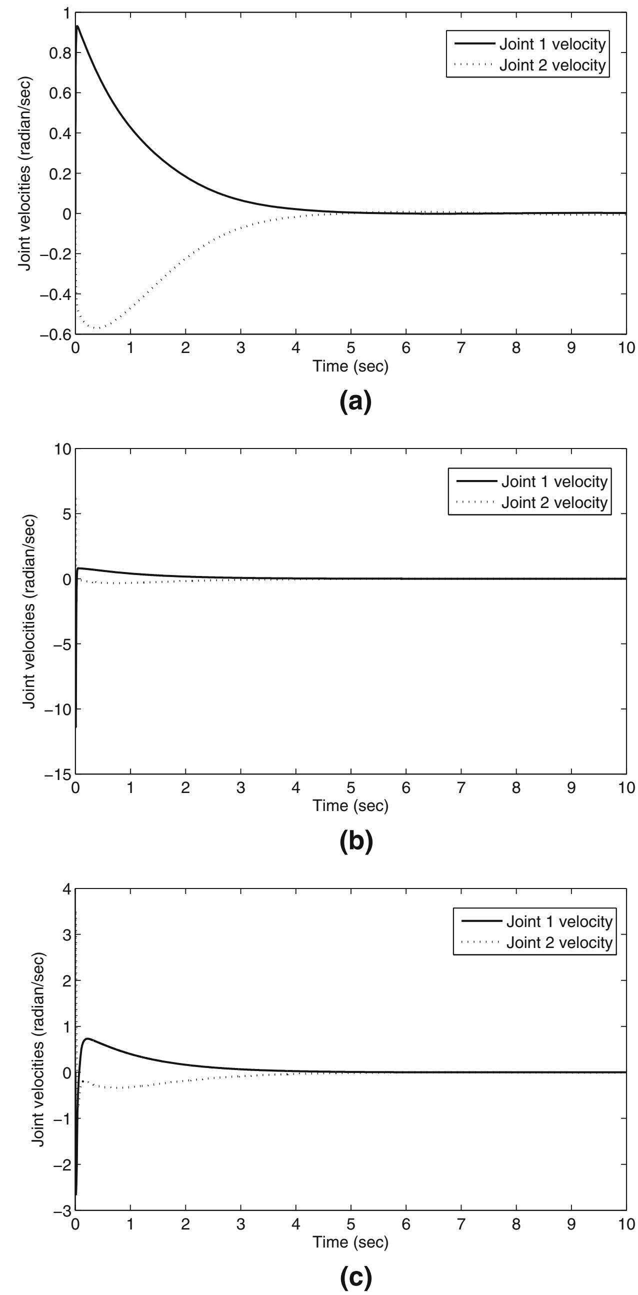


**Fig. 5** Joint velocities with **a** model based CT controller, **b** FFNN based model free controller, **c** proposed controller

controller is stable and the errors converge asymptotically. Finally, through the numerical simulation, the effectiveness of proposed controller is proved in a comparative manner with model based and model free controllers. From overall analysis and simulation studies, it is concluded that the proposed controller is able to deal with uncertainties and disturbances in effective manner and the tracking performance is enhanced with the proposed controller as compare to the model based and model free controllers.

## References

1. Raibert MH, Craig JJ (1981) Hybrid position/force control of manipulators. ASME J Dyn Syst Meas Control 103:126–133
2. Lozano R, Brogliato B (1992) Adaptive hybrid position/force control of redundant manipulators. IEEE Trans Autom Control 37(10):1501–1505
3. Yoshikawa T, Sudou A (1993) Dynamic hybrid position/force control of 5 robot manipulators: online-estimation of unknown constraint. IEEE Trans Robot Autom 9(2):220–226
4. Kwan CM (1996) Robust adaptive force/motion control of constrained robots. IEE Proc Control Theory Appl 143(1):103–109

5. Kouya DN, Saad M, Lamarche L (2002) Backstepping adaptive hybrid force/position control for robotic manipulators. In: Proceedings of the American control conference, pp 4595–4600
6. De Queiroz MS, Jun H, Dawson DM (1997) Adaptive position/force control of robot manipulators without velocity measurements: theory and experimentation. IEEE Trans Syst Man Cybern Part B Cybern 27(5):796–809
7. Cheah CC, Zhao Y, Slotine JJE (2006) Adaptive Jacobian motion and force control for constrained robots with uncertainties. In: Proceedings of the international conference on robotics and automation, pp 2226–2231
8. Filaretov VF, Zuev AV (2008) Adaptive force/position control of robot manipulators. In: Proceedings of the 2008 IEEE/ASME international conference on advanced intelligent mechatronics, pp 96–101
9. Kouya DN (2008) Adaptive hybrid force-position control of a robotic manipulator. In: 2nd European computing confrence, pp 323–331
10. Roy J, Whitcomb LL (2002) Adaptive force control of position/velocity controlled robots: theory and experiment. IEEE Trans Robot Autom 18(2):121–137
11. Pliego-Jimenez J, Arteaga-Perez MA (2015) Adaptive position/force control for robot manipulators in contact with a rigid surface with uncertain parameters. Eur J Control 22:1–12
12. Lewis FL, Jagannathan S, Yesildirek A (1999) Neural network control of robot manipulators and nonlinear systems. Taylor and Francis, Brighton
13. Lewis FL, Yesildirek A, Liu K (1996) Multilayer neural net robot controller with quaranted tracking performance. IEEE Trans Neural Netw 7(2):388–399
14. Fanaei A, Farrokhi M (2006) Robust adaptive neuro-fuzzy controller for hybrid position/force control of robot manipulators in contact with unknown environment. J Intell Fuzzy Syst 17(2):125–144
15. Liu H, Wang L, Wang F (2007) Fuzzy force control of constrained robot manipulators based on impedance model in an unknown environment. Front Mech Eng China 2(2):168–174
16. Karayiannidis Y, Rovithakis K, Doulgeri Z (2007) Force/position tracking for a robotic manipulator in compliant contact with a surface using neuro-adaptive control. Automatica 43(7):1281–1288
17. Bechlioulis CP, Doulgeri Z, Rovithakis GA (2010) Neuroadaptive force/position control with prescribed performance and guaranteed contact maintenance. IEEE Trans Neural Network 21(12):1857–1868
18. Kumar N, Panwar V, Sukavanam N, Sharma SP, Borm JH (2011) Neural network based hybrid force/position control for robot manipulators. Int J Precis Eng Manuf 12(3):419–426
19. Singh HP, Sukavanam N (2013) Stability analysis of robust adaptive hybrid position/force controller for robot manipulators using neural network with uncertainties. Neural Comput Appl 22(7–8):1745–1755
20. Mahjoub S, Mnif F, Derbel N, Hamerlain M (2014) Radial-basis-functions neural network sliding mode control for underactuated mechanical systems. Int J Dyn Control 2(4):533–541
21. Li Y, Wang G, Dong B, Zhao B (2015) Hybrid positionforce control for constrained reconfigurable manipulators based on adaptive neural network. Adv Mech Eng 7(9):1–10
22. Lee CH, Wang WC (2016) Robust adaptive position and force controller design of robot manipulator using fuzzy neural networks. Nonlinear Dyn 85(1):343–354
23. Ghajar MH, Keshmiri M, Bahrami J (2018) Neural-network-based robust hybrid force/position controller for a constrained robot manipulator with uncertainties. Trans Inst Meas Control 40(5):1625–1636
24. De Oliveira OC, De Medeiros Martins A, De Araujo AD (2017) A dual-mode control with a RBF network. J Control Autom Electr Syst 28(2):180–188
25. Rani K, Kumar N (2018) Design of intelligent hybrid force and position control of robot manipulators. In: Proceedings of the 6th international conference on smart computing and communications, pp. 42–49
26. Slotine JJE, Li W (1991) Applied nonlinear control. Prentice-Hall, Englewood Cliffs