Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Stable neural PD controller for redundantly actuated parallel manipulators with uncertain kinematics

G. Loreto and R. Garrido

Departamento de Control Automático

CINVESTAV-IPN, Av.IPN 2508 México D.F., 07360, México

fax: (52) 55 57 47 70 89

{gloreto, garrido}@ctrl.cinvestav.mx.

Abstract— **This paper proposes a stable Proportional Derivative Controller applied to redundantly actuated parallel robots with uncertainty in the kinematic parameters. It is shown that all the closed loop signals are** *uniformly ultimately bounded*. **Gravitational terms are approximated using a Radial Basis Function Neural Network with joint information feeding their activation functions and with on-line real-time learning. A depart from current approaches is the fact that damping is added at the joint level using the robot active joints and the fact that it does not require the exact knowledge of the kinematic parameters. The learning rule for the neural network weights is obtained from a Lyapunov stability analysis. Simulation results are reported and demonstrate the effectiveness of the proposed controller.**

Keywords— **Radial basis function, regulation, parallel robots, actuator redundancy.**

I. Introduction

Control of parallel robots is a current and interesting topic in robot control research. A motivation is the fact that parallel robots have several advantages, i.e., higher stiffness, accuracy and speed, compared with their open link counterparts [6], [8], [9].

One of the key issues in parallel mechanisms is the possibility of singularities. When a parallel robot reaches a singular configuration, its stiffness and accuracy deteriorates and in some cases it is not longer possible to move the robot through the actuators attached to it [1]. A method for removing singularities consists in adding more branches and actuators to the original mechanism, i.e., the mechanism becomes redundantly actuated [1], [2]. Another effect of redundancy is that torques or forces required for a given control law are better distributed among all the actuators. Using the D'Alembert formulation, Cheng and coworkers [2] developed a novel approach for modeling redundantly actuaded parallel robots. The proposed methodology takes into account the kinematic constraints due to the parallel robots nature and solves the problems encountered in early developments on the subject.

Using the aforementioned modeling methodology, in [2] several control laws were proposed for redundantly actuated parallel robots. Proportional-Derivative (PD) controllers at joint and task levels were experimentally tested. A shortcoming of the joint-based PD controller is the fact that it does not exploit actuator redundancy. More advanced controllers at the task level, based on the computed torque technique and on feedforward compensation were also tested. The aim at using the advanced controllers was to improve the trajectory following capabilities of the simple task level PD controller. It is interesting to point out that in all the task space controllers mentioned above, damping is added at the task level and the knowledge of gravity forces is needed. According to [1], where the experiments for the above controllers were firstly reported, experimental results were conducted using a revolute joint planar robot actuated redundantly and moving in the horizontal plane. Other works related to control of redundant parallel robots are [4], [5]. However, the above schemes may exhibit poor performance if the kinematic parameters are not exactly known, therefore, the robot is required to interact with its environment and hence the overall parameters would change according to different tasks. In the case of the serial manipulators some works related to this problem are [3] and [13], to the best of the authors knowledge there is not previous work about this problem for redundantly actuated parallel robots

The aim of this work is to present a stable PD controller for set point control of redundantly actuated parallel manipulators using neural network compensation. A feature of the proposed controller is the fact that damping associated to the derivative action is done at the joint level. In this way, damping is made independent of the robot configuration and at the same time reducing the burden for computing the control law. The proposed scheme removes completely the requirement of the exact or partial knowledge of the gravity terms and it does not rely on exact knowledge of the kinematic parameters. Uniformly ultimately boundednes (UUB) of the closed loop signals is studied using Lyapunov Stability Theory. The paper is organized as follows. In Section 2 modelling of the robot and its main properties are presented. Stability analysis of the proposed

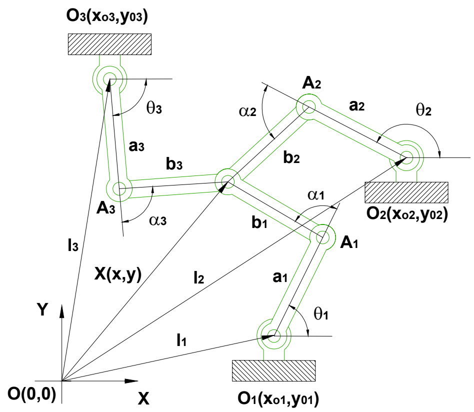

Fig. 1. Parallel robot in closed form.

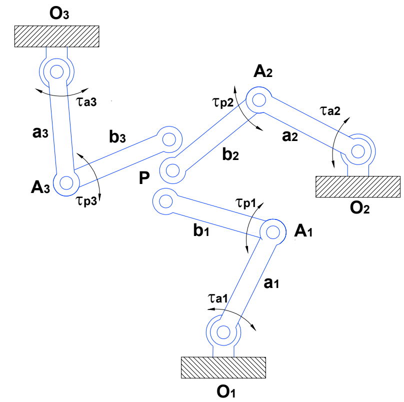

Fig. 2. Parallel robot in its open-chain form

control laws is performed in Section 3. Some remarks are given at the end of the paper.

II. Background

A. Modelling of redundantly actuated parallel robots.

According to [1], [2], modeling of redundantly actuated parallel robots is done using the equivalent open-chain mechanism. As an example, Figure 1 shows a planar parallel robot closed form and Figure 2 shows its equivalent open-chain form. Note that in the example the robot is decomposed into 3 branches corresponding to two-link planar robots. For more complicated robot mechanical architectures a similar decomposition applies. Each of the branches of the open chain form is modeled according to the well known Euler Lagrange formalism [7] as a standard serial link robot. Concatenating the dynamic models of each branch the following equation is obtained

$$\bar{M}\ddot{\mathbf{q}} + \bar{C}\dot{\mathbf{q}} + \bar{G} = \boldsymbol{\tau} \tag{1}$$

with

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_a \\ \cdots \\ \mathbf{q}_p \end{bmatrix} \quad \boldsymbol{\tau} = \begin{bmatrix} \boldsymbol{\tau}_a \\ \cdots \\ \boldsymbol{\tau}_p \end{bmatrix} \tag{2}$$

where $\mathbf{q}_a \in \Re^m$ stands for the angles of the motorized or active joints and $\mathbf{q}_p \in \Re^{n\text{-}m}$ for the angles of the passive or non motorized joints. In the same way, $\boldsymbol{\tau}_a \in \Re^m$ and $\boldsymbol{\tau}_p \in \Re^{n\text{-}m}$ correspond to the torques in the active and passive joints. Note that if friction in the passive joints is neglected, $\boldsymbol{\tau}_p$ is zero.

The forward kinematics is given by

$$\mathbf{X} = \mathbf{h}(\mathbf{q}) = \begin{bmatrix} h_a(\mathbf{q}) \\ h_p(\mathbf{q}) \end{bmatrix} \tag{3}$$

where $\mathbf{X} \in \Omega \subset \Re^i$ are the coordinates of the robot end effector. The subset Ω where the end effector evolves will be called the robot task space The following relationships represent the differential kinematics

$$\dot{\mathbf{q}} = W\dot{\mathbf{X}}; \qquad \dot{\mathbf{q}}_a = S\dot{\mathbf{X}} \tag{4}$$

where $W \in \Re^{n\times i}$ and $S \in \Re^{m\times i}$ are jacobian matrices defined by

$$W = \begin{bmatrix} \frac{\partial h_a^{-1}}{\partial \mathbf{X}} \\ \frac{\partial h_p^{-1}}{\partial \mathbf{X}} \end{bmatrix} \tag{5}$$

$$S = \begin{bmatrix} \frac{\partial h_a^{-1}}{\partial \mathbf{X}} \end{bmatrix} \tag{6}$$

Note that if S is full rank then W is also full rank [12]. Another key relationship between the toques τ and τ_a is given by the following equation

$$W^T\boldsymbol{\tau} = S^T\boldsymbol{\tau}_a \tag{7}$$

Using (1) and (7) it is possible to write down the robot dynamics (1) in terms of the robot end effector coordinates $\mathbf{X}$

$$M\ddot{\mathbf{X}} + C\dot{\mathbf{X}} + G = S^T\boldsymbol{\tau}_a \tag{8}$$

where

$$M = W^T\bar{M}W$$

$$C = W^T\bar{M}\dot{W} + W^T\bar{C}W$$

$$G = W^T\bar{G}$$

It is interesting to point out that (8) relates the active joint torques τ_a and the end effector coordinates $\mathbf{X}$. In reference [2], [10] it is shown that matrices M and C have the following properties as long as matrix W is full rank.

Property 1: The inertia matrix M is symmetric and positive-definite.

Property 2: Matrix $\dot{M} - 2C$ is skew-symmetric, i.e., $\dot{M} = C + C^T$

The following properties of the Jacobian matrices S and W will be exploited in the sequel and are given in the following proposition [12].

Proposition 1: *Consider matrices $S \in \Re^{m\times i}$ and $W \in \Re^{n\times i}$ in (7). If $\dot{q}_a = 0$ then $\dot{X} = 0$. Moreover, if matrix S is full rank then matrix W is also full rank.*

B. Radial Basis Function Neural Networks

Consider the radial basis function neural network (RBFNN) with N_2 hidden neurons and N_3 output neurons. Assume that $c_{ji},\ j = 1\ldots N_2$ are the RBF centers, then, the output $y_k,\ k = 1,\ldots,N_3$ is given by [11]

$$y_k = \sum_{j=1}^{N_2} v_{kj}\sigma_j(\mathbf{x}) + v_{ko} \tag{9}$$

where $\mathbf{x} \in \Re^{N_1}$ is the input vector, v_{kj} is the weight connecting the hidden neuron j and the output neuron k, v_{ko} is the threshold offset of output neuron k and $\sigma_j(\mathbf{x}) = \prod_{i=1}^{N_1}\sigma_j(x_i)$ is an activation function selected as a Gaussian function $\sigma_j(x_i) = \exp\left(-\frac{(x_i-c_{ji})^2}{2p_{ji}^2}\right)$ where p_{ji} is a width parameter. Introducing the following notation $\mathbf{y} = \begin{bmatrix} y_1 & \cdots & y_{N_3}\end{bmatrix}^T$, $\boldsymbol{\sigma}(\mathbf{x}) = \begin{bmatrix} 1 & \sigma_1(\mathbf{x}) & \sigma_2(\mathbf{x}) & \cdots & \sigma_{N_2}(\mathbf{x})\end{bmatrix}^T$ and weight matrix $\mathbf{V}^T = [v_{kj}]$ including the thresholds v_{ko} as the first column of it, we can express (9) as

$$\mathbf{y} = \mathbf{V}^T\boldsymbol{\sigma}(\mathbf{x}) \tag{10}$$

where $\mathbf{V}^T \in \Re^{N_3\times(N_2+1)}$ and $\boldsymbol{\sigma}(\cdot) \in \Re^{N_2+1}$. In this paper, the RBFNN is used to design an adaptive controller since its structure is simpler that of the multilayer perceptron (MLP), so the learning speed of RBFNN is generally faster than that of MLP [11].

III. Proposed control law

The control problem is stated as follows. We assume that the target is static and located inside the robot workspace so the control problem is then to synthesize a control law such that the actual position of the end effector $\mathbf{X}$ reaches the desired position $\mathbf{X}_d \in \Re^i$ with certain precision defined by

$$\|\mathbf{X}_d - \mathbf{X}\| \le \Gamma; \qquad \Gamma > 0 \tag{11}$$

Before proposing a solution to the problem stated above, some comments regarding the determination of the torques $\boldsymbol{\tau}_a$ are in order. Assume that a control law $\mathbf{u}$ defined in terms of the end effector coordinates is applied to the robot manipulator dynamics (8), then, torques $\boldsymbol{\tau}_a$ must be computed from the following expression

$$S^T\boldsymbol{\tau}_a = \mathbf{u} \tag{12}$$

The approach employed in [2] is to use the Moore-Penrose pseudoinverse $(S^T)^+ = S(S^TS)^{-1}$ of S^T [12] which is equivalent to solve (12) in a least squares sense. Hence, $\boldsymbol{\tau}_a$ is computed as

$$\boldsymbol{\tau}_a = (S^T)^+\mathbf{u} \tag{13}$$

Solution (13) makes sense only if the pseudoinverse $(S^T)^+$ is well defined, i.e., if S is full rank. Now, let consider the following PD plus gravity compensation control law

$$\boldsymbol{\tau}_a = (S^T)^+\left[\mathbf{K}_p\tilde{\mathbf{X}} + G\right] - \mathbf{K}_d\dot{\mathbf{q}}_a \tag{14}$$

proposed in [12], where $\tilde{\mathbf{X}} = \mathbf{X}_d - \mathbf{X}$ is the position error and $K_p > 0$, $K_d > 0$ are the proportional and derivative diagonal matrix gains. Using the neural network universal approximation property, the gravity term in (8) can be obtained as

$$\Psi(q) = (S^T)^+G = \mathbf{V}^T\boldsymbol{\sigma}(\mathbf{q}) + \boldsymbol{\varepsilon} \tag{15}$$

where $\mathbf{V^T} \in \Re^{N_3\times(N_2+1)}$, $\boldsymbol{\sigma}(\mathbf{q}) \in \Re^{N_2+1}$, ε is the neural network approximation error. For some unknown constant ideal weights $\mathbf{V}$, the reconstruction error is bounded by $\|\boldsymbol{\varepsilon}\| < k_\varepsilon$.

Then, an estimated of the graviy term is

$$\widehat{\Psi}(q) = \widehat{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) \tag{16}$$

where $\widehat{\mathbf{V}} \in \Re^{N_3\times(N_2+1)}$ are estimates of de ideal weights $\mathbf{V}$. If the matrix S is uncertain and only an estimate $\widehat{S}$ is available such that

$$\left\|S - \widehat{S}\right\| \le k_\Delta \tag{17}$$

where k_Δ is a positive constant and using (16), then, the proposed control law is

$$\boldsymbol{\tau}_a = (\widehat{S}^T)^+\mathbf{K}_p\widetilde{\mathbf{X}} - \mathbf{K}_d\dot{\mathbf{q}}_a + \widehat{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) \tag{18}$$

In order to implement the control law (18) the following assumption are needed

Assumption 3: The ideal weights, the Jacobian matrices S and $\widehat{S}$ are bounded by positive values k_v, k_s and $k_{\hat{s}}$ so that

$$\|\mathbf{V}\|_F \le k_v; \qquad \|S\|_F \le k_s; \qquad \left\|\widehat{S}\right\|_F \le k_{\hat{s}}$$

Substituting (18) into (8) and noting (15) yields

$$M\ddot{\mathbf{X}} + C\dot{\mathbf{X}} = S^T\left[(\widehat{S}^T)^+\mathbf{K}_p\tilde{\mathbf{X}} - \mathbf{K}_d\dot{\mathbf{q}}_a - \widetilde{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) - \boldsymbol{\varepsilon}\right] \tag{19}$$

where $\widetilde{\mathbf{V}} = \mathbf{V} - \widehat{\mathbf{V}}$ is the weight estimation error.

Assume that the end effector position and velocity are available for measurement, then, the following theorem shows how to adjust the weights of neural network (16) to guarantee closed-loop stability in spite of an unknown Jacobian matrix S and gravity terms G.

Theorem 4: Consider system (8) in closed-loop with control law (18) where the updating law for the weights of the RBFNN is given by

$$\begin{aligned}\dot{\widehat{\mathbf{V}}} &= -\mathbf{K}_v\boldsymbol{\sigma}(\mathbf{q})\left[\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})\right]^T\widehat{S}^T \\ &\quad -\kappa_1\mathbf{K}_v\left\|\tilde{\mathbf{X}}\right\|\left\|\dot{\mathbf{X}}\right\|\widehat{\mathbf{V}} - \kappa_2\mathbf{K}_v\left\|\dot{\mathbf{X}}\right\|\widehat{\mathbf{V}}\end{aligned} \tag{20}$$

where $\mathbf{K}_v$ is a positive defined matrix, κ_1 and κ_2 are positive constants and $\mathbf{f}(\tilde{\mathbf{X}}) = \frac{\tilde{\mathbf{X}}}{1+\|\tilde{\mathbf{X}}\|}$. If the positive constant μ is chosen such that

$$\min\left\{\sqrt{\frac{\lambda_m\{\mathbf{K}_p\}}{\lambda_M\{M\}}}, \frac{\lambda_m\{S^T\mathbf{K}_d\mathrm{S}\}}{2c}, \frac{2\lambda_m\{\mathbf{K}_p\}}{\lambda_M\left\{\widehat{S}^T\mathbf{K}_d\widehat{S}\right\}}\right\} > \mu \tag{21}$$

where $\lambda_m\{\cdot\}$ and $\lambda_M\{\cdot\}$ denotes the smallest and largest eigenvalue, respectively of any matrix, and c is a positive constant, then, $\tilde{\mathbf{X}}$, $\dot{\mathbf{X}}$ and $\widetilde{\mathbf{V}}$ are UUB.

Proof: Define a Lyapunov function candidate as

$$\begin{aligned} V &= \frac{1}{2}\dot{\mathbf{X}}^T\mathrm{M}\dot{\mathbf{X}} + \frac{1}{2}\tilde{\mathbf{X}}^T\mathbf{K}_p\tilde{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathrm{M}\dot{\mathbf{X}} \\ &+ \frac{1}{2}\mathrm{tr}\left(\widetilde{\mathbf{V}}^T\mathbf{K}_v^{-1}\widetilde{\mathbf{V}}\right) \end{aligned} \tag{22}$$

where $\overline{\mathrm{S}} = \widehat{\mathrm{S}}^T\left(\mathrm{S}^T\right)^+$. Equation (22) is positive definite since by hypothesis $\sqrt{\frac{\lambda_m\{\mathbf{K}_p\}}{\lambda_M\{\mathrm{M}\}}} > \mu$. Taking the time derivative of (22), using closed-loop system (19) and the robot Properties, it is not difficult to show that

$$\begin{aligned} \dot{V} &= \dot{\mathbf{X}}^T\left\{\mathrm{S}^T\left[\left(\widehat{\mathrm{S}}^T\right)^+\mathbf{K}_p\tilde{\mathbf{X}} - \mathbf{K}_d\dot{\mathbf{q}}_a \right.\right. \\ &\qquad \left.\left. - \widetilde{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) - \boldsymbol{\varepsilon}\right]\right\} \\ &- \tilde{\mathbf{X}}^T\mathbf{K}_p\dot{\mathbf{X}} - \mu\dot{\mathbf{f}}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathrm{M}\dot{\mathbf{X}} \\ &- \mu\mathbf{f}(\tilde{\mathbf{X}})^T\dot{\overline{\mathrm{S}}}\mathbf{M}\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathbf{C}^T\dot{\mathbf{X}} \\ &- \mu\mathbf{f}(\tilde{\mathbf{X}})^T\left\{\mathbf{K}_p\tilde{\mathbf{X}} + \widehat{\mathrm{S}}^T\left[-\mathbf{K}_d\dot{\mathbf{q}}_a \right.\right. \\ &\qquad \left.\left. - \widetilde{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) - \boldsymbol{\varepsilon}\right]\right\} \\ &+ \mathrm{tr}\left(\widetilde{\mathbf{V}}^T\mathbf{K}_\mathbf{v}^{-1}\dot{\widetilde{\mathbf{V}}}\right) \end{aligned} \tag{23}$$

where the equality $\dot{\tilde{\mathbf{X}}} = -\dot{\mathbf{X}}$ was employed. Defining $\Delta = \mathrm{S} - \widehat{\mathrm{S}}$ and using the above definition yields

$$\begin{aligned} \dot{V} &= \dot{\mathbf{X}}^T\left\{\mathrm{S}^T\left[-\Delta\mathbf{K}_p\tilde{\mathbf{X}} - \mathbf{K}_d\dot{\mathbf{q}}_a\right]\right\} \\ &- \dot{\mathbf{X}}^T\left\{\left[\widehat{\mathrm{S}}^T\widetilde{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) + \Delta\widetilde{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) + \mathrm{S}^T\boldsymbol{\varepsilon}\right]\right\} \\ &- \mu\dot{\mathbf{f}}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathbf{M}\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})^T\dot{\overline{\mathrm{S}}}\mathbf{M}\dot{\mathbf{X}} \\ &- \mu\mathbf{f}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathrm{C}^T\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})^T\left\{\mathbf{K}_p\tilde{\mathbf{X}} \right. \\ &\qquad \left. + \widehat{\mathrm{S}}^T\left[-\mathbf{K}_d\dot{\mathbf{q}}_a - \widetilde{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) - \boldsymbol{\varepsilon}\right]\right\} \\ &+ \mathrm{tr}\left(\widetilde{\mathbf{V}}^T\mathbf{K}_v^{-1}\dot{\widetilde{\mathbf{V}}}\right) \end{aligned} \tag{24}$$

It is not difficult to show that

$$-\dot{\mathbf{X}}^T\mathrm{S}^T\mathbf{K}_d\dot{\mathbf{q}}_a \leq -\frac{1}{2}\dot{\mathbf{X}}^T\mathrm{S}^T\mathbf{K}_d\mathrm{S}\dot{\mathbf{X}} - \frac{1}{2}\lambda_m\left\{\mathrm{S}^T\mathbf{K}_d\mathrm{S}\right\}\left\|\dot{\mathbf{X}}\right\|^2$$

It now follows from the above inequalities that the time derivative of the Lyapunov function candidate (24) satisfies

$$\begin{aligned} \dot{V} &\leq -\frac{1}{2}\left[\mathrm{S}\dot{\mathbf{X}} - \mu\widehat{\mathrm{S}}\mathbf{f}(\tilde{\mathbf{X}})\right]^T\mathbf{K}_d\left[\mathrm{S}\dot{\mathbf{X}} - \mu\widehat{S}\mathbf{f}(\tilde{\mathbf{X}})\right] \\ &+ \frac{1}{2}\mu^2\mathbf{f}(\tilde{\mathbf{X}})^T\widehat{\mathrm{S}}^T\mathbf{K}_d\widehat{S}\mathbf{f}(\tilde{\mathbf{X}}) - \frac{1}{2}\lambda_m\left\{\mathrm{S}^T\mathbf{K}_d\mathrm{S}\right\}\left\|\dot{\mathbf{X}}\right\|^2 \\ &+ \dot{\mathbf{X}}^T\left\{-\mathrm{S}^T\Delta\mathbf{K}_p\tilde{\mathbf{X}} - \Delta\widetilde{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) - \mathrm{S}^T\boldsymbol{\varepsilon}\right\} \\ &- \mu\dot{\mathbf{f}}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathbf{M}\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})^T\dot{\overline{\mathrm{S}}}\mathrm{M}\dot{\mathbf{X}} \\ &- \mu\mathbf{f}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathrm{C}^T\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})^T\left\{\mathbf{K}_p\tilde{\mathbf{X}} - \widehat{\mathrm{S}}^T\boldsymbol{\varepsilon}\right\} \\ &+ \mathrm{tr}\left(\widetilde{\mathbf{V}}^T\left\{\mathbf{K}_v^{-1}\dot{\widetilde{\mathbf{V}}} - \boldsymbol{\sigma}(\mathbf{q})\left[\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})\right]^T\widehat{\mathrm{S}}^T\right\}\right) \end{aligned} \tag{25}$$

We now provide upper bounds on the following terms:

$$\frac{1}{2}\mu^2\mathbf{f}(\tilde{\mathbf{X}})^T\widehat{\mathrm{S}}^T\mathbf{K}_d\widehat{S}\mathbf{f}(\tilde{\mathbf{X}}) \leq \frac{\mu^2}{2\left(1+\|\tilde{\mathbf{X}}\|\right)}\lambda_M\left\{\widehat{\mathrm{S}}^T\mathbf{K}_d\widehat{S}\right\}\left\|\tilde{\mathbf{X}}\right\|^2$$

$$-\dot{\mathbf{X}}^T\mathrm{S}^T\Delta\mathbf{K}_p\tilde{\mathbf{X}} \leq k_s k_\Delta\lambda_M\{\mathbf{K}_p\}\left\|\tilde{\mathbf{X}}\right\|\left\|\dot{\mathbf{X}}\right\|$$

$$-\dot{\mathbf{X}}^T\Delta\widetilde{\mathbf{V}}^T\boldsymbol{\sigma}(\mathbf{q}) \leq k_\Delta\left\|\dot{\mathbf{X}}\right\|\left\|\widetilde{\mathbf{V}}\right\|$$

$$-\dot{\mathbf{X}}^T\mathrm{S}^T\boldsymbol{\varepsilon} \leq k_s k_\varepsilon\left\|\dot{\mathbf{X}}\right\|$$

$$-\mu\mathbf{f}(\tilde{\mathbf{X}})^T\mathbf{K}_p\tilde{\mathbf{X}} \leq -\frac{\mu}{1+\left\|\tilde{\mathbf{X}}\right\|}\lambda_m\{\mathbf{K}_p\}\left\|\tilde{\mathbf{X}}\right\|^2$$

$$\mu\mathbf{f}(\tilde{\mathbf{X}})^T\widehat{\mathrm{S}}^T\boldsymbol{\varepsilon} \leq \frac{\mu}{1+\left\|\tilde{\mathbf{X}}\right\|}k_{\widehat{s}}k_\varepsilon\left\|\tilde{\mathbf{X}}\right\|$$

where we have used the following inequalities:

$$\left\|\mathbf{f}(\tilde{\mathbf{X}})\right\| \leq 1; \qquad \left\|\dot{\mathbf{f}}(\tilde{\mathbf{X}})\right\| \leq 2\left\|\dot{\tilde{\mathbf{X}}}\right\| \tag{26}$$

and $k_{\widehat{s}}$ and k_s are the norm bounds for $\widehat{\mathrm{S}}$ and S, respectively. Since $\mathbf{f}(\tilde{\mathbf{X}})$ and $\overline{\mathrm{S}}$ are bounded, there exists a constant $c > 0$ such that

$$-\mu\dot{\mathbf{f}}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathbf{M}\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})^T\dot{\overline{\mathrm{S}}}\mathrm{M}\dot{\mathbf{X}} - \mu\mathbf{f}(\tilde{\mathbf{X}})^T\overline{\mathrm{S}}\mathrm{C}^T\dot{\mathbf{X}} \leq \mu c\left\|\dot{\mathbf{X}}\right\|^2$$

It now follows from the above inequalities and using the inequality

$$\begin{aligned} \mathrm{tr}\{\widetilde{\mathbf{V}}^T(\mathbf{V} - \widetilde{\mathbf{V}})\} &\leq \left\|\widetilde{\mathbf{V}}\right\|_F\left(k_v - \left\|\widetilde{\mathbf{V}}\right\|_F\right) \\ &\leq -\left[\left(\left\|\widetilde{\mathbf{V}}\right\|_F - \frac{k_v}{2}\right)^2 - \frac{k_v^2}{4}\right] \end{aligned}$$

and the updating law (20), that the time derivative of the Lyapunov function candidate (25) satisfies

$$\begin{aligned}\dot{V} &= -\frac{1}{2}\left[\mathrm{S}\dot{\mathbf{X}}-\boldsymbol{\mu}\widehat{\mathrm{S}}\mathbf{f}(\tilde{\mathbf{X}})\right]^T\mathbf{K}_d\left[\mathrm{S}\dot{\mathbf{X}}-\boldsymbol{\mu}\widehat{S}\mathbf{f}(\tilde{\mathbf{X}})\right]\\ &\quad -\frac{\mu}{1+\left\|\tilde{\mathbf{X}}\right\|}\left[\gamma_p\left\|\tilde{\mathbf{X}}\right\|-k_{\widehat{\mathrm{s}}}k_\varepsilon\right]\left\|\tilde{\mathbf{X}}\right\|\\ &\quad -\left[\gamma_d\left\|\dot{\mathbf{X}}\right\|-k_s k_\varepsilon\right]\left\|\dot{\mathbf{X}}\right\| \\ &\quad -\Bigg[\kappa_1\left[\left(\left\|\widetilde{\mathbf{V}}\right\|_F-\frac{k_v}{2}\right)^2-\frac{k_v^2}{4}\right]\\ &\qquad\qquad -k_s k_\Delta\lambda_M\left\{\mathbf{K}_p\right\}\Bigg]\left\|\tilde{\mathbf{X}}\right\|\left\|\dot{\mathbf{X}}\right\|\\ &\quad -\left[\kappa_2\left[\left(\left\|\widetilde{\mathbf{V}}\right\|_F-\frac{k_v}{2}\right)^2-\frac{k_v^2}{4}\right]\right]\left\|\dot{\mathbf{X}}\right\|\end{aligned} \tag{27}$$

where

$$\begin{aligned}\gamma_d &= \tfrac{1}{2}\lambda_m\left\{\mathrm{S}^T\mathbf{K}_d\mathrm{S}\right\}-\mu c\\ \gamma_p &= \lambda_m\left\{\mathbf{K}_p\right\}-\tfrac{\mu}{2}\lambda_M\left\{\widehat{\mathrm{S}}^T\mathbf{K}_d\widehat{S}\right\}\end{aligned} \tag{28}$$

Since μ satisfies (21), then, γ_d and γ_p are positive constants. Finally, the time derivative of Lyapunov function (22) is guaranteed to be negative as long as

$$\begin{aligned}\tilde{\mathbf{X}} \notin \Omega_1 &= \left\{\tilde{\mathbf{X}}\left|\left\|\tilde{\mathbf{X}}\right\| \leq \frac{k_{\widehat{\mathrm{s}}}k_\varepsilon}{\gamma_p}\right.\right\}\\ \dot{\mathbf{X}} \notin \Omega_2 &= \left\{\dot{\mathbf{X}}\left|\left\|\dot{\mathbf{X}}\right\| \leq \frac{k_{\mathrm{s}}k_\varepsilon}{\gamma_d}\right.\right\}\\ \widetilde{\mathbf{V}} \notin \Omega_3 &= \left\{\widetilde{\mathbf{V}}\left|\left\|\widetilde{\mathbf{V}}\right\| \leq \beta\right.\right\}\end{aligned} \tag{29}$$

where

$$\beta = \max\left\{\frac{k_v}{2}+\sqrt{\frac{k_s k_\Delta\lambda_M\left\{\mathbf{K}_p\right\}}{\kappa_1}+\frac{k_v^2}{4}}, k_v+\frac{k_\Delta}{\kappa_2}\right\}$$

Ω_1, Ω_2 and Ω_3 are the convergence regions and practical bounds for the signals $\tilde{\mathbf{X}}$, $\dot{\mathbf{X}}$ and $\widetilde{\mathbf{V}}$ in the sense that excursions beyond these sets will be very small. According to the standard Lyapunov theory extension [11], the above demostrates that $\tilde{\mathbf{X}}$, $\dot{\mathbf{X}}$ and $\tilde{\mathbf{V}}$ are UUB. ■

Remark 5: As shown in (28) increasing $\mathbf{K}_p$ and $\mathbf{K}_d$ will help to reduce the size of Ω_1 and Ω_2, then, lower bounds for $\left\|\tilde{\mathbf{X}}\right\|$ and $\left\|\dot{\mathbf{X}}\right\|$ can be made smaller increasing the control law gains. In addition, the proposed controller is dependent on $\widehat{\mathrm{S}}$ but not on S, then, the exact knowledge of the forward kinematics (3) to obtain the Jacobian matrix S is relaxed and only an estimated is needed to obtain $\widehat{\mathrm{S}}$ which satisfies condition (17).

Remark 6: the control law (14) may be though as a velocity inner loop at the joint level and a proportional plus gravity compensation outer loop at the task level, therefore, computational burden may be reduced even more since the velocity loop may be performed by the power amplifier driving the actuators. The above scheme is possible because today digital power amplifiers employed for driving brushed or brushless servomotors can be configured in velocity mode. Hence, the main processor is freed from executing the inner loop and it is charged only of computing the outer loop.

TABLE I
MANIPULATORS PARAMETERS

	notation	value	unit
Length link 1	a_i	0.1	m
Length link 2	b_i	0.1	m
Center of mass link 1	a_{ci}	0.05	m
Center of mass link 2	b_{ci}	0.05	m
Mass link 1	$m1_i$	0.5	kg
Mass link 2	$m2_i$	0.5	kg
Inertia link i	I_i	$4\mathrm{x}10^{-4}$	kg m^2
Gravity acceleration	g	9.8	$\mathrm{m/sec}^2$

Remark 7: Avoiding exact knowledge of the robot kinematics implies that the end-effector position, velocity and orientation must be measured directly without resorting on the direct kinematics, for example, using an image acquisition and processing unit.

IV. Simulation Results

A parallel manipulator moving in the vertical plane, shown in Fig. 1, is used for simulation. Complete information regarding the elements of dynamic system (8) can be found in [2]. The meaning of the symbols and numerical values used for simulation of dynamic (8) are listed in Table I.

If $q = \begin{bmatrix}\theta_1 & \theta_2 & \theta_3 & \varphi_1 & \varphi_2 & \varphi_3\end{bmatrix}^T$ in (1), then, The estimated Jacobian matrix $\widehat{W}$ and $\widehat{S}$ used are

$$\widehat{W} = \begin{bmatrix}\widehat{b}_{1x}r_1 & \widehat{b}_{1y}r_1\\ \widehat{b}_{2x}r_2 & \widehat{b}_{2y}r_2\\ \widehat{b}_{3x}r_3 & \widehat{b}_{3y}r_3\\ -\widehat{d}_{1x}r_1 & -\widehat{d}_{1y}r_1\\ -\widehat{d}_{2x}r_2 & -\widehat{d}_{2y}r_2\\ -\widehat{d}_{3x}r_3 & \widehat{d}_{3y}r_3\end{bmatrix} \qquad \widehat{S} = \begin{bmatrix}\widehat{b}_{1x}r_1 & \widehat{b}_{1y}r_1\\ \widehat{b}_{2x}r_2 & \widehat{b}_{2y}r_2\\ \widehat{b}_{3x}r_3 & \widehat{b}_{3y}r_3\end{bmatrix}$$

where $r_i = 1/e_i$, $e_i = \widehat{d}_{ix}\widehat{b}_{iy} - \widehat{d}_{iy}\widehat{b}_{ix}$ and

$$\begin{aligned}\widehat{b}_{ix} &= \widehat{a}_i\cos(\theta_i+\varphi_i); \qquad \widehat{b}_{ix} = \widehat{a}_i\cos(\theta_i+\varphi_i)\\ \widehat{d}_{ix} &= \widehat{a}_i\cos(\theta_i)+\widehat{b}_i\cos(\theta_i+\varphi_i)\\ \widehat{d}_{iy} &= \widehat{a}_i\sin(\theta_i)+\widehat{b}_i\sin(\theta_i+\varphi_i)\end{aligned}$$

The next values were used for simulation. The initial conditions of the robot position is $X(t_o) = [0.147, 0.218]^T$ *meters*, the target has been placed at $X_d = [0.1,\ 0.24]^T$ *meters*. The radial basis function neural networks were formed with 8 neurons, feeding only with joint information, their centers and their widths are assumed to be fixed. We choose the centers c_{ji} evenly spaced between $[10, 10, 10, 10, 10, 10]^T$ to $[-10, -10, -10, -10, -10, -10]^T$, their widths were set to $p_{ji} = 5$ and all the initial weights values were set to zero. The symmetric positive definite proportional, derivative and $\mathbf{K}_v$ matrices were chosen as

$$\mathbf{K}_p = diag\{15\}, \quad \mathbf{K}_d = diag\{0.5\}, \\ \mathbf{K}_v = diag\{0.2\}$$

and the constants $\kappa_1 = 0.1$, $\kappa_2 = 0.3$ and $\mu = 0.001$. All the gains in the control law are chosen to achieve the best transient control performance in simulation considering the requirement of stability, limitation of control effort and possible operating conditions. Figure 3 shows the end-effector position errors $\tilde{\mathbf{X}}$ of the end effector and figure 4 the path of the end-effector.

V. Concluding Remarks

This work presented theoretical and simulation results for control of redundantly actuated parallel robots with neural network compensation. Radial basis function neural network was used for compensating the gravitational torques without needing off-line phase training. As a depart from other approaches, the proposed controller avoids the requirement of the Jacobian matrix S, moreover, it does not need the exact knowlegde of the forward kinematics and it does not need any knowledge on the gravity terms structure. An advantage of the proposed approach is that damping is independent of the robot configuration. Stability was assed through Lyapunov Stability Theory. Simulation results validate the proposed approach.

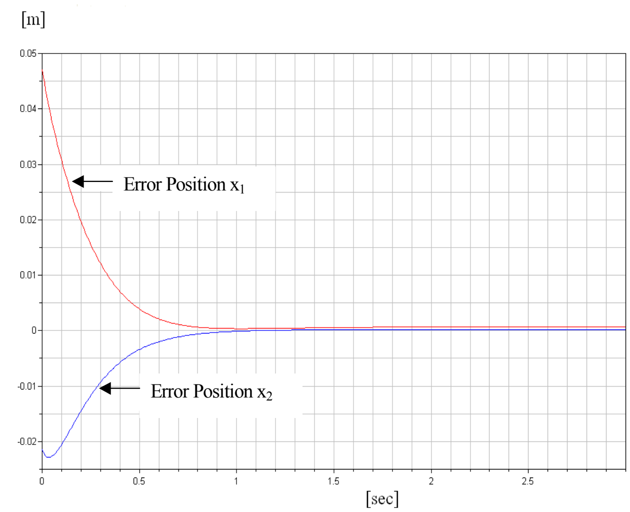

Fig. 3. End-effector position error

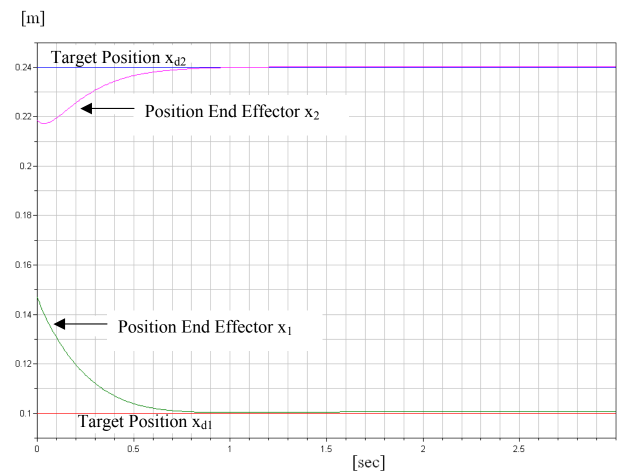

Fig. 4. Path of the End-effector

References

[1] H. Cheng. Dynamics and control of parallel manipulators with actuation redundancy. M. Sc. Thesis. Department of Electrical and Electronic Engineering. The Hong Kong University of Science and Technology. 2001.

[2] H. Cheng, Y. K. Yiu, Z. Li. Dynamics and Control of redundantly actuated parallel manipulators. IEEE/ASME Transactions on Mechatronics. Vol.8, No. 4, pp 483-491. 2003.

[3] Cheah, C.C., M. Hirano, S. Kawamura & S. Arimoto, Approximate Jacobian Control for Robots with Uncertain Kinematics and Dynamics, *IEEE Trans. on Robotics and Automation,* Vol. 19, No. 4, pp.692-702, Aug. 2003.

[4] S. Kock, W. Schumacher. A mixed elastic and rigid-body dynamic model of an actuation redundant parallel robot with high-reduction gears. Proc. of the Int. Conf. on Robotics and Automation. San Francisco, U.S.A., 2000.

[5] S. Kock, W. Schumacher. Control of a fast parallel robot with a redundant chain and gearboxes: Experimental results. Proc. of the Int. Conf. on Robotics and Automation. San Francisco, U.S.A., 2000.

[6] J.P. Merlet. *Parallel Robots.* Klwuer Academic Publishers. 2000.

[7] Spong M. & Vidyasagar M., *"Robot Dynamics and Control"*, Wiley, New York 1989.

[8] S. Tadokoro. Control of parallel mechanisms. Advanced Robotics. Vol. 8, No. 6, pp 559-571, 1994.

[9] L.W. Tsai. Robot Analysis. Jhon Wiley and Sons Inc. 1999.

[10] Y.K. Yiu, H. Cheng, Z.H. Xiong, G.F. Liu, Z.X. Li. On the dynamics of parallel manipulators. Proc. of the Int. Conf. on Robotics and Automation. Seoul, Korea. 2001.

[11] Lewis F. L., Jagannathan S. and Yesildirek A., *Neural Network Control of Robot Manipulators and Nonlinear Systems,* Taylor & Francis, Philadelphia 1999.

[12] R. Garrido and D. Torres-Cruz, On PD control of parallel robots with redundant actuation, Int. conf. on electrical and electronics engineering and X conf. on electrical enginering, pp 356-360, México 2004.

[13] Yazarel, H. & C. C. Cheah, Task-Space Adaptive Control of Robotic Manipulators with Uncertainties in Gravity Regressor Matrix and Kinematics, *IEEE Trans. on Automatic Control,* Vol 47, No. 9, pp. 1580-1585, Sep. 2002.