Journal of Intelligent and Robotic Systems (2005)
DOI: 10.1007/s10846-005-9006-4

Comparison of Different Methods for Computing the Forward Kinematics of a Redundant Parallel Manipulator

H. SADJADIAN and H. D. TAGHIRAD
Advanced Robotics and Automated Systems (ARAS), Electrical Engineering Department, K.N. Toosi University of Technology, P.O. Box 16315-1355, Tehran, Iran; e-mail: Sadjadian@alborz.kntu.ac.ir, Taghirad@kntu.ac.ir

(Received: 29 January 2003; in final form: 16 June 2005)

Abstract. In this paper, three numerical methods are presented to solve the forward kinematics of a three DOF actuator-redundant hydraulic parallel manipulator. It is known, that on the contrary to series manipulators, the forward kinematic map of parallel manipulators involves highly coupled nonlinear equations, whose closed-form solution derivation is a real challenge. This issue is of great importance noting that the forward kinematics solution is a key element in closed loop position control of parallel manipulators. The proposed methods, namely the Neural Network Estimation, the Quasi-closed Solution, and the Taylor series approximation, are using mainly numerical computations, with different ideas to solve the problem in hand. The latter two methods are proposed for the first time in literature to solve the forward kinematics of a parallel manipulator. These methods are compared in detail and the advantages or the disadvantages of each method in computing the forward kinematic map of the given mechanism is discussed. It is shown that a 4th order Taylor series approximation to the problem provides a good compromise for practical applications compared to that of other methods considered in this paper.

Key words: closed-form, forward kinematics, neural networks, numerical solution, parallel manipulator, performance comparison, quasi-closed form, Taylor series.

1. Introduction

Over the last two decades, parallel manipulators have been among the most considerable research topics in the field of robotics. These robots are used in real-life applications such as force sensing robots, fine positioning devices, and medical applications [10, 11]. As in the case of conventional serial robots, kinematic analysis of parallel manipulators is also performed in two phases. In forward or direct kinematics the position and orientation of the mobile platform as the robot end effector is determined given the leg lengths. This is done with respect to a base reference frame. In inverse kinematics, the position and orientation of the mobile platform is used to determine actuator lengths. It is known that unlike serial manipulators, inverse position kinematics for parallel robots is usually sim-

ple and straightforward. In most cases joint variables (actuator displacements) may be computed independently using the given pose of the movable platform. The solution to this problem is in most cases uniquely determined.

However, the forward kinematics of parallel manipulators is generally very complicated. It usually involves a set of highly coupled nonlinear equations, that in general there is no closed form and usually no unique solution for it. In the literature, mostly six DOF parallel mechanisms based on the Stewart–Gough platform are analyzed. However, parallel manipulators with three DOF have been also implemented for applications where six DOF are not required, such as high-speed machine tools. Recently, three DOF parallel manipulators with more than three legs have been investigated [22]. Complete kinematic modeling and Jacobian analysis of such mechanisms have not received much attention so far and is still regarded as a challenging problem in parallel robotics research [23]. Different approaches are provided in literature to solve this problem either generally or in special cases. There are also numerous cases in which the solution to this problem is provided for a special or novel architecture [20, 21]. In general, different solutions to this problem can be placed in one of the following forms [3, 4]:

- Numerical approaches
- Analytical approaches
- Closed-form solution for special architectures

In this paper three different, mainly numerical approaches are being analyzed to solve the kinematics of a three DOF actuator redundant hydraulic parallel manipulator. This solution is an inherent part of closed-loop feedback design for position control requirements. The comparisons given in this paper, although is performed for a special mechanism, is an important research asset in the field of parallel robot analysis, since for the first time it provides an engineering judgment between three different approaches to solve the forward kinematic map in the literature, and is a useful guide for control designers to choose and integrate the right method in feedback control routines. It is notable that research results with similar scope have not received much attention in the literature, and only a few results are published with the same objectives but mainly for serial or biological manipulators [24, 25].

To accomplish this objective, the paper is organized as follows. Section 2 contains the mechanism description, and kinematic modeling of the manipulator is discussed in Section 3, where inverse and forward kinematics is studied in detail and the need for appropriate method to solve the forward kinematics is justified. In Section 4, three different methods to solve the forward kinematics problem are discussed in detail. First, two different but mostly common neural networks are used to estimate the forward kinematic map of the given mechanism. In the second method a quasi-closed form is provided for the same purpose which combines the numerical and analytical schemes. Finally, with a

new approach, conventional Taylor series expansion is applied for the problem in order to approximate the nonlinear map with required precision. In Section 5, these methods are simulated and compared in two parts. First each method is simulated with different parameters and structure, and the best compromise is nominated, then among the best alternatives, a comparison on the estimation error and complexity in implementation is performed and the advantages or disadvantages of each scheme are elaborated. It is concluded that a fourth order Taylor series approximation provides a good compromise for similar applications as in Hydraulic shoulder, with the assigned required accuracies and performance.

2. Mechanism Description

A three DOF actuator-redundant hydraulically-driven parallel manipulator is used as the basis of our study, which is called 'Hydraulic Shoulder' herein after. The mechanism is designed by Dr. V. Hayward [7, 8], borrowing design ideas from biological manipulators and specially the biological shoulder [6]. The interesting features of the mechanism and its similarity to human shoulder have made it a unique design, which can serve as a basis for a good experimental setup for parallel robot research. As it is shown in Figure 1 the mobile platform of the hydraulic shoulder is constrained to spherical motions. Four high performance hydraulic actuators are used to give three degrees of freedom in the mobile

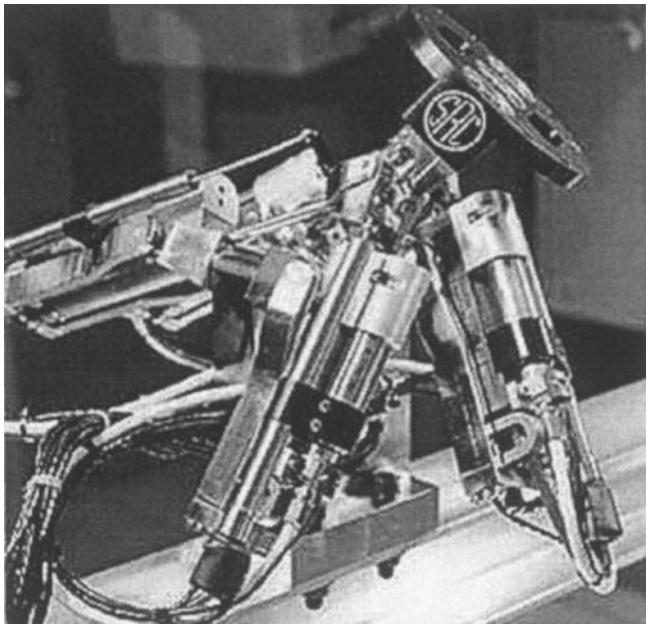

Figure 1. The hydraulic shoulder in movement.

platform. Each actuator includes a position sensor of LVDT type and an embedded Hall Effect force sensor. Simple elements like spherical and universal joints are used in the structure. A complete kinematic analysis of such a careful design will provide us with good results regarding the structure itself and its performance.

From the structural point of view, the shoulder mechanism falls into an important class of robotic mechanisms called parallel robots. In these robots, the end effector is connected to the base through several closed kinematic chains. The motivation behind using these types of robot manipulators is to compensate for the shortcomings of the conventional serial manipulators such as low precision, low stiffness, high error accumulation and low load carrying capability. Parallel structures are usually lighter and simpler than their serial counterparts. However, they have their own disadvantages, which are mainly smaller workspace and many singular configurations. Recently, hybrid structures are designed which combine the advantages of both serial and parallel robots.

The hydraulic shoulder, being a parallel structure, has the general features of these structures; however, the design is such that the redundancy of the actuators prohibits any singular configuration in its workspace [7]. It can be thought of as a shoulder for a light weighed seven DOF robotic arm, which can carry loads several time its own weight. Simple elements, used in this design, add to its lightness and simplicity. The workspace of such a mechanism can be considered as part of a spherical surface. The orientation angles are limited to vary between $-\pi/6$ and $\pi/6$.

3. Kinematics

The hydraulic shoulder is kinematically over constrained. The inverse kinematics problem is easily solved, given the orientation of the mobile plate. This is also the case for general parallel robots. The inverse kinematics problem has a unique solution, in our case meaning that, the hydraulic shoulder cannot be optimized by choosing between inverse kinematics solutions. But, in contrast to serial structures, the forward kinematics is very complicated and there is no closed form solution in general. Figure 2 depicts a geometric model for the mechanism which will be used for its kinematics derivation.

The parameters used in kinematics can be defined as:

l_b Base actuator offset lengths, $l_b : \left\|\overrightarrow{CA_i}\right\|$

l_p Distance between the fixed and moving platform centers, $l_p : \left\|\overrightarrow{CC_1}\right\|$

l_k Distance from the moving platform center and actuator moving endpoints along Z_1, $l_k : \left\|\overrightarrow{C_1P_i}\right\|_{z_1}$

l_d Distance from the moving platform center and actuator moving endpoints along Y_1, $l_d : \left\|\overrightarrow{C_1P_i}\right\|_{y_1}$

α The angle between CA_4 and y_0
C Center of the reference frame
C_1 Center of the moving plate
ρ_i Link lengths $i = 1, 2, 3, 4$
P_i Moving endpoints of the actuators
A_i Fixed endpoints of the actuators

Two coordinate frames are defined as depicted in Figure 2. The base frame $X_0Y_0Z_0$ is centered at C (rotation center) with its Z_0-axis perpendicular to the plane defined by $A_1A_2A_3A_4$ and an X_0 axis parallel to the bisector of angle $\angle A_1CA_4$. The second frame, namely $X_1Y_1Z_1$ is centered at C_1 (center of the moving plate) with its Z_1 axis perpendicular to the line defined by the actuators moving end points (P_1P_2) and horizontal Y axis along C_1P_2.

3.1. INVERSE KINEMATICS

In modeling the inverse kinematics of the hydraulic shoulder we must determine link lengths (ρ_i) as the joint space variables given the task space variables, namely θ_x, θ_y and θ_z as the orientation angles of the moving platform. First we note that the fixed end points of the actuators (A_i) can be written in the base frame as:

$$\begin{aligned} A_1^0 &= (l_b\sin\alpha \quad -l_b\cos\alpha \quad 0) \\ A_2^0 &= (-l_b\sin\alpha \quad -l_b\cos\alpha \quad 0) \\ A_3^0 &= (-l_b\sin\alpha \quad l_b\cos\alpha \quad 0) \\ A_4^0 &= (l_b\sin\alpha \quad l_b\cos\alpha \quad 0) \end{aligned} \tag{1}$$

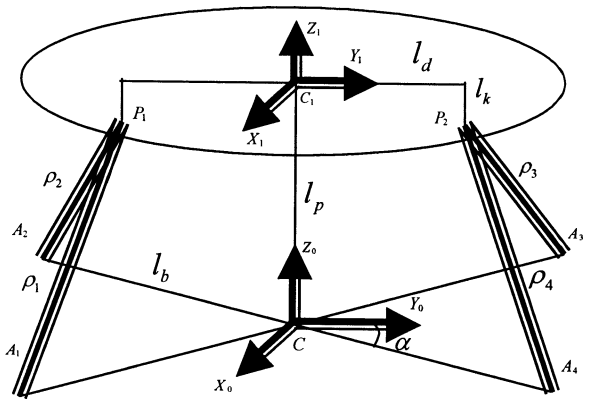

Figure 2. A geometric model for the hydraulic shoulder.

Also:

$$\begin{aligned} P_1^1 &= (0 \quad -l_d \quad -l_k) \\ P_2^1 &= (0 \quad l_d \quad -l_k) \end{aligned} \tag{2}$$

These must be transferred to the base frame using the rotation matrix R_1^0, where:

$$S_{3\times 3} = R_1^0 = R_z(\theta_z)R_y(\theta_y)R_x(\theta_x) \tag{3}$$

As a result the rotation matrix components are computed as following:

$$\begin{aligned} \mathrm{S}_{11} &= \cos(\theta_z)\cos(\theta_y) \\ \mathrm{S}_{21} &= \sin(\theta_z)\cos(\theta_y) \\ \mathrm{S}_{31} &= -\sin(\theta_y) \\ \mathrm{S}_{12} &= \cos(\theta_z)\sin(\theta_y)\sin(\theta_x) - \sin(\theta_z)\cos(\theta_x) \\ \mathrm{S}_{22} &= \sin(\theta_z)\sin(\theta_y)\sin(\theta_x) + \cos(\theta_z)\cos(\theta_x) \\ \mathrm{S}_{32} &= \cos(\theta_y)\sin(\theta_x) \\ \mathrm{S}_{13} &= \cos(\theta_z)\sin(\theta_y)\cos(\theta_x) + \sin(\theta_z)\sin(\theta_x) \\ \mathrm{S}_{23} &= \sin(\theta_z)\sin(\theta_y)\cos(\theta_x) - \cos(\theta_z)\sin(\theta_x) \\ \mathrm{S}_{33} &= \cos(\theta_y)\cos(\theta_x) \end{aligned} \tag{4}$$

So we have:

$$p_1^0 = \begin{pmatrix} -l_{\mathrm{d}}s_{12} & -l_{\mathrm{k}}s_{13} \\ -l_{\mathrm{d}}s_{22} & -l_{\mathrm{k}}s_{23} \\ -l_{\mathrm{d}}s_{32} & -l_{\mathrm{k}}s_{33} \end{pmatrix}, \quad p_2^0 = \begin{pmatrix} l_{\mathrm{d}}s_{12} & -l_{\mathrm{k}}s_{13} \\ l_{\mathrm{d}}s_{22} & -l_{\mathrm{k}}s_{23} \\ l_{\mathrm{d}}s_{32} & -l_{\mathrm{k}}s_{33} \end{pmatrix} \tag{5}$$

The final step is to translate the resulting vectors P_i^0 by l_p along the Z axis. Having P_i^0 and A_j^0 in hand, the link lengths $\left\|\overrightarrow{P_iA_j}\right\|$ can be easily computed as:

$$\rho_j = \sqrt{(p_x - a_x)^2 + (p_y - a_y)^2 + (p_z - a_z)^2} \tag{6}$$

where:

$$P_i^0 = \begin{bmatrix} p_x & p_y & p_z \end{bmatrix}^T \quad i = 1, 2 \tag{7}$$

and:

$$A_j^0 = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}^T \quad j = 1, 2, 3, 4 \tag{8}$$

are defined in Equations (5) and (1), respectively.

Equation (6) completely models the inverse kinematics of the hydraulic shoulder. As it is obvious from the equations the link lengths will be uniquely computed given the orientation angles θ_x, θ_y, and θ_z. So the manipulator is not kinematically redundant, meaning that reaching a specific point in the task space can't be satisfied through different combinations of the link lengths.

3.2. FORWARD KINEMATICS

Equation (6) can also be used for the forward kinematics of the hydraulic shoulder but with the link lengths as the input and orientation angles θ_x, θ_y, θ_z as the unknowns. In fact, we have four nonlinear equations to solve for three unknowns. Obviously, solving such a system of nonlinear equations for a unique closed-form analytic solution to the forward kinematics problem is very complicated, although three equations of the four could be used. Several inconclusive attempts have been made in this direction; therefore, we propose using numerical methods or a combination of the numerical and analytic schemes to solve the forward kinematics problem as a basic element in modeling and control of the manipulator. These are studied in detail in the next section.

4. Forward Kinematics Solution

4.1. NEURAL NETWORKS ESTIMATION

One of the most interesting features of neural networks is undoubtedly their ability to approximate nonlinear maps or functions. Furthermore neural network schemes are independent of the system structure resulting in a robust approach with respect to environmental changes. There have been extensive applications of these networks to modeling and control of robot manipulators. Research results in this field have shown that neural networks are an ideal choice to compute the forward kinematics of parallel manipulators [5]. Various architectures of these networks have been used in literature to solve the forward kinematics problem even for the conventional manipulators. For example in [26] a simple multilayer feed forward network is used to solve the forward kinematics of Stewart platform. Geng and Haynes [5], have proposed another structure called CMAC (Cerebella Model Arithmetic Computer) which is a multiple neural network structure to solve the same problem. Nguyen, Pate1 and Khorasani [14] compared various neural network models for solving the problem which has been used for serial manipulators. Other structures such as kohonen self-organizing networks [19], linear estimators using neural networks [17] and bidirectional mapping neural networks [9] have been also used for similar applications. In all of the above, fault tolerance, adaptability, learning capability and non-algorithm nature of modeling are considered as the most important advantages of neural networks over other numerical or analytical approaches to

solve the forward kinematics problem. Furthermore, no programming is required, the time to obtain a solution is independent of the number of degrees of freedom and real-time operation is also possible.

The results of these and also other similar works on this topic suggest using neural networks in robotics applications such as in the forward kinematics problem. Two most popular and of course suitable neural network models for function approximation are the multilayer feed forward and Radial Basis networks which have been studied in detail to solve the forward kinematics problem of the hydraulic shoulder.

4.1.1. *Multilayer Feed Forward Network*

A simple feed forward network with back propagation learning was used in the first step. It is a well-known theorem which states that for any function **f** and any positive ε there exists a three layer back propagation network that can be used to approximate **f** with ε as the mean-squared error accuracy [26]. Therefore, such a network can be applied to approximate the forward kinematics map of the hydraulic shoulder with desired accuracy. The input layer has as many nodes as the number of inputs to the map namely four link lengths. Similarly the output layer will have three nodes which represent the orientation of the moving plate (θ_x, θ_y, θ_z).

The number of neurons in the hidden layer was used as a design parameter. Sigmoid and linear transfer functions were selected for all hidden and output layer nodes respectively. Supervised learning scheme was used in which the manipulator is treated as a black box and the network is taught to learn the map by observing the inputs and outputs. Such a learning scheme will result in offline training. The supervised training method requires a training data pair which is generated easily using the inverse kinematics map for the hydraulic shoulder. The target pattern for training, the three orientation angles, was randomly generated within the workspace of the robot and the input pattern, four actuator displacements, was found using the inverse kinematics model. The pair was then used to train the network and the weights were updated in a back propagation process.

Random initialization was used for the weights. Different configurations of the feed forward network were tested by varying the number of neurons in the hidden layer between 5 and 35 and the performance of these networks was compared.

Different performance indices could be used in this case, the best of which could be the sum of square output errors, though other indices such as mean square or mean absolute error may also be used. Networks with best performance as indicated would be selected, from which the network with fewer hidden layer nodes will be the best choice since the number of weights and also the training time of the network will increase with more neurons in the hidden layer. About 30 multilayer feed forward networks with one hidden layer were trained by varying the number of neurons in the hidden layer from 5 to 35. The

network with 34 neurons in the hidden layers was the best compromise. As another configuration, the same multilayer feed forward network was used with two hidden layers. The activation function of the second hidden layer was also sigmoid. About 20 multilayer feed forward networks with two hidden layers were trained by varying the number of neurons from 10 to 25 in the first hidden layer and from 5 to 15 in the second hidden layer. These networks had a better performance in general compared to the networks with a single hidden layer (see Table I).

All the networks were trained over 1000 training epochs with Bayesian regularization training. Each network was evaluated by comparing the predictions to the true outputs, resulting in a prediction error for each orientation angle. The autocorrelation coefficients were also computed for the prediction error in each angle [15]. All the trainings and simulations of the neural networks were done on a Pentium 4, 2 GHz. Using the whole stated criteria, five networks with best performance were selected from each configuration. Table I summarizes the performance of these networks. It can be seen that networks with two hidden

Table I. Performance of multilayer feed forward and radial basis networks.

Network structure	Multilayer feed forward one hidden layer				
	No. of hidden layer neurons	Training time (s)	MSE	SSE	MAE
Network performance	$S = 27$	$7.3e^{3}$	$2.8e^{-5}$	0.644	0.0037
	$S = 29$	$8.2e^{3}$	$2.9e^{-5}$	0.66	0.0035
	$S = 30$	$8.6e^{3}$	$1.9e^{-5}$	0.428	0.0028
	$S = 34$	$1e^{4}$	$1.1e^{-5}$	0.242	0.0022
	$S = 35$	$1.1e^{4}$	$1.1e^{-5}$	0.26	0.0022
Network Structure	Multilayer feed forward two hidden layers				
Network performance	S1 = 10 S2 = 15	$9.5e^{3}$	$6.8e^{-6}$	0.154	0.0018
	S1 = 12 S2 = 15	$2.9e^{4}$	$2.8e^{-6}$	0.062	0.0011
	S1 = 17 S2 = 15	$6.1e^{4}$	$8.1e^{-7}$	0.018	$6e^{-4}$
	S1 = 17 S2 = 9	$1e^{4}$	$5.6e^{-6}$	0.12	0.0016
	S1 = 17 S2 = 12	$2.3e^{4}$	$1.9e^{-6}$	0.044	$9e^{-4}$
Network structure	Radial basis function				
Network performance	Training time (s)		MSE	SSE	MAE
RBF1	750		$1.3e^{-5}$	0.1	0.0019
RBF2	680		$9.9e^{-6}$	0.074	0.0017

layers have a better performance in general. The network with 12 neurons in the first and 15 neurons in the second hidden layer provides a good compromise. It should be also noted that the mean square of error is approximately equal to the square of the maximum error. So a mean square error of $1e^{-5}$ will correspond to about 0.18° of accuracy for the forward kinematics solution.

4.1.2. *Radial Basis Neural Network*

Radial basis function (RBF) neural network architecture was tested as another choice for computing the forward kinematics of the hydraulic shoulder. In general, RBF networks require more neurons but much less training time than feed forward back propagation networks. RBF networks consist of two layers: a hidden radial basis layer and an output linear layer. Input and output patterns were generated in a same procedure as in the multilayer feed forward network. Supervised learning method was used in a way to reduce the estimated error of the network. Other specifications such as weight initialization, network evaluation and performance indices were just the same as the multilayer feed forward network. About 10 different configurations with different spread parameters were trained and compared, from which two networks with best performance were selected. The performance of these networks is also shown in Table I, in which MSE stands for mean square error, SSE for sum of square of error, and MAE for mean absolute error. From the comparison of the selected structures in Table I, it can be seen that the multilayer feed forward network with two hidden layers have a better performance in general, regarding the training times, training errors and number of weights, compared to other network structures. However, such accuracy may not meet accurate robotic applications such as our redundant parallel manipulator.

4.2. QUASI-CLOSED SOLUTION METHOD

Most of the research regarding the closed form solution to the forward kinematics problem of parallel manipulators has assumed simplified or special conditions under which a closed form solution to the problem could be found [13]. Some researchers have focused on a special or a novel architecture [1, 12, 18]. Also, in [2] a closed form solution has been provided for parallel manipulators with planar base and mobile platform which is based on the use of three linear extra sensors to provide additional information.

In this section, we propose a quasi-closed form solution method to estimate the forward kinematics map of the hydraulic shoulder. First, recall from Equation (6) that the link lengths are described as:

$$\rho_i = \left\| \overrightarrow{p_j^0 A_i^0} \right\| \quad i = 1,2,3,4, \quad j = 1,2 \tag{9}$$

Substituting P_j^0 and A_i^0 from Equations (5) and (1), we can rewrite the kinematic equations as:

$$\begin{aligned}
\rho_1 &= \sqrt{A + Bs_{12} + Cs_{13} - Ds_{22} - Es_{23} - Fs_{32} - Gs_{33}} \\
\rho_2 &= \sqrt{A - Bs_{12} - Cs_{13} - Ds_{22} - Es_{23} - Fs_{32} - Gs_{33}} \\
\rho_3 &= \sqrt{A + Bs_{12} - Cs_{13} - Ds_{22} + Es_{23} + Fs_{32} - Gs_{33}} \\
\rho_4 &= \sqrt{A - Bs_{12} + Cs_{13} - Ds_{22} + Es_{23} + Fs_{32} - Gs_{33}}
\end{aligned} \tag{10}$$

where parameters A, B, C, D, E, F and G are used for simplicity of notations and depend only on the geometric features of the mechanism described in Figure 2 as follows:

$$A = l_b^2 + l_d^2 + l_k^2 + l_p^2,\ B = 2l_b l_d \sin\alpha,\ C = 2l_b l_k \sin\alpha,$$
$$D = -2l_b l_d \cos\alpha,\ E = -2l_b l_k \cos\alpha,\ F = -2l_d l_p,\ G = -2l_k l_p$$

These parameters are measured and calculated as follows:

$$A = 0.0268\,\text{m}, B = 0.0045\,\text{m}, C = 0.0083\,\text{m}, D = 0.0026\,\text{m}, E = 0.0048\,\text{m},$$
$$F = 0.0092\,\text{m}, G = 0.0169\,\text{m}.$$

The s_{ij}'s are the nine entries of the rotation matrix which represent the orientation of the moving platform, and ρ_1, ρ_2, ρ_3, ρ_4 are the link lengths. It is fairly easy to obtain the forward kinematics equations having the rotation matrix S in hand. Hence, the problem reduces to solving for the rotation matrix instead, with nine entries as the unknowns. Noting that these entries are not independent, there would be no need to compute all nine unknowns.

Furthermore, The elements in the first column of S, namely: s_{11}, s_{21}, and s_{31} are not present in the kinematic equations, which simplifies the problem as we can find the second and third columns of S and the first column will be simply computed as their cross product.

Hence, the problem is solving the kinematic Equations (10) for the rotation matrix with the following constraints:

$$s_{12}^2 + s_{22}^2 + s_{32}^2 = 1 \tag{11}$$

$$s_{13}^2 + s_{23}^2 + s_{33}^2 = 1 \tag{12}$$

$$\text{col}(2).\text{col}(3) = 0 \tag{13}$$

co1 (2) and co1 (3) are defined as the second and third columns of the rotation matrix S, respectively. We must note that S is an orthonormal matrix so the second and third columns must be orthogonal with unit lengths. As the cross

product of two orthonormal vectors would also be orthonormal, the other constraints on the rotation matrix entries would be trivial.

From Equations (10), we can solve for s_{12}, s_{13} as:

$$\begin{aligned} s_{12} &= \frac{\rho_1^2 + \rho_3^2 - \rho_2^2 - \rho_4^2}{4B} \\ s_{13} &= \frac{\rho_1^2 + \rho_4^2 - \rho_2^2 - \rho_3^2}{4C} \end{aligned} \tag{14}$$

Equation (14) is in the form of an analytic closed form solution. Unfortunately the high coupling of the forward kinematic equations makes the closed form computation of other entries of **S** complicated. Several inconclusive attempts were made to find an analytic solution for these entries; therefore we proposed a new approach combining the analytic and numerical methods to solve for the remaining entries of the rotation matrix in a quasi-closed form.

We can relate the four remaining unknowns with the following equations:

$$\begin{aligned} s_{22} &= \frac{4A - \rho_1^2 - \rho_2^2 - \rho_3^2 - \rho_4^2 - 4Gs_{33}}{4D} \\ s_{23} &= \frac{4Fs_{32} + \rho_1^2 + \rho_2^2 - \rho_3^2 - \rho_4^2}{-4E} \end{aligned} \tag{15}$$

Or much simpler:

$$\begin{aligned} s_{22} &= \alpha + \beta s_{33} \\ s_{23} &= \gamma s_{32} + \eta \end{aligned} \tag{16}$$

where:

$$\begin{aligned} \alpha &= \frac{4A - \rho_1^2 - \rho_2^2 - \rho_3^2 - \rho_4^2}{4D}, \beta = -\frac{G}{D} \\ \eta &= \frac{\rho_1^2 + \rho_2^2 - \rho_3^2 - \rho_4^2}{-4E}, \gamma = -\frac{F}{E} \end{aligned} \tag{17}$$

Using constraint Equations (11)–(12) we have:

$$\begin{aligned} s_{12}^2 + (\alpha + \beta s_{33})^2 + s_{32}^2 &= 1 \\ s_{13}^2 + (\gamma s_{32} + \eta)^2 + s_{33}^2 &= 1 \end{aligned} \tag{18}$$

The remaining problem will be to determine s_{32}, s_{33} using the above system of nonlinear equations, given s_{12}, s_{13}. Having s_{32}, s_{33} in hand, we can easily find the remaining entries s_{22}, s_{23} to obtain the second and third columns of the rotation matrix **S**, from which the first column would be determined by a cross product.

The two entries s_{32}, s_{33} were obtained numerically from Equation (18) solving a constrained optimization problem. In the next step the rotation matrix **S** was

determined from which the orientation angles were easily obtained. Hence the forward kinematics problem was solved using the rotation matrix of the end effector as a means of simplifying the equations.

It should be noted that the degree of accuracy of the numerical optimization scheme to find an orthonormal rotation matrix is considered very important. The accuracy of the proposed method will definitely depend on the initial conditions used in the optimization process. This can be remedied in our application, by using the results of previous step approximation as the initial condition for the next step. We should also note that the proposed method uses a combination of analytic and numerical computation schemes, hence called a quasi-closed form solution method. The detailed simulation results of this method are given in Section 5, where the effectiveness and accuracy of approximation is compared to the other methods.

4.3. TAYLOR SERIES EXPANSION

As shown in Section 3, the forward kinematic model of the hydraulic shoulder involves four nonlinear equations with link lengths ($\rho_1, \ldots, \rho_4$) as the input and orientation angles of the end effector (θ_x, θ_y, θ_z) as the outputs, in other words:

$$\boldsymbol{\theta} = \mathbf{f}(\rho_1, \rho_2, \rho_3, \rho_4) \tag{19}$$

In which, $\mathbf{f}$ represents the forward kinematics map, that is subject of solution. A basic numerical approach to solve this problem is to approximate the nonlinear function $\mathbf{f}$ with a Taylor series expansion of arbitrary order:

$$\boldsymbol{\theta} = \mathbf{f}(\mathbf{0}) + \sum_{i=1}^{4} \frac{\partial \mathbf{f}}{\partial \rho_i} \rho_i + \sum_{i=1}^{4} \sum_{j=1}^{4} \frac{\partial^2 \mathbf{f}}{\partial \rho_i \partial \rho_j} \rho_i \rho_j + \circ \circ \circ \tag{20}$$

The number of the coefficients in the expansion is determined by the required degree of accuracy. Solving the forward kinematic problem, will, hence, be equal to computing these coefficients. In order to accomplish this task, different trajectories were considered for the end effector and the corresponding actuator displacements were determined using the inverse kinematics. The data pair was then used to compute the coefficients of the Taylor expansion using least square estimation. The trajectories in the task space must consider the whole workspace of the manipulator so that the estimated function for the forward kinematics could be used equally in different points of the workspace. Therefore, about 25 different trajectories are selected and being embedded into 70,000 sample points, used to estimate the coefficients in a least square approach. Different orders of expansion up to fourth ($O(n^5)$) were considered separately and the coefficients in each case were computed. The estimation error between the desired $\boldsymbol{\theta}$ and its estimate $\widehat{\boldsymbol{\theta}}$, namely: $\mathrm{e} = \boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}$ was used as a performance index of each scheme.

In general, the number of the coefficients in the expansion will increase with the order of approximation resulting in a better degree of accuracy but usually with a slight increase in the computation time. The estimation results for different orders of expansion are compared in Table II, in which PE stands for Prediction Error, SSE for sum of square of error, MSE for mean square error, and MAE for mean absolute error. This result enables the designer to choose the appropriate order, according to the performance requirement. The accuracy sought by the fourth order approximation seems to be a good compromise for robotic applications. The coefficients for such an approximation could be found in [16].

5. Simulation Results

The comparison study presented in this paper has two major parts. In first part as elaborated in Section 4, each method is examined with different parameters and structures, and the best compromise of parameters and structures is concluded in the above tables. In this part of comparisons, not only some measures of prediction error can be used as a quantitative measure, to select the best compromise, but also the complexity of each method and the amount of offline computations are compared in detail. In this section and as the second part of comparisons, a simulation study for selected representatives of all methods concluded in preceding is performed, in which a sample trajectory of the robot is considered in the reachable workspace of the manipulator. By this means and through these two part of comparison study, a fair judgment can be drawn not only between the various structures proposed in each method, but also between the selected representatives of each method. It should be noted that in the first part of com-

Table II. Measures of prediction errors for different orders of approximation, Taylor series expansion.

Approximation Order		2nd	3rd	4th
No. of coefficients		15	35	64
Max PE	θ_x	0.058	0.0116	0.0033
	θ_y	0.292	0.0475	0.014
	θ_z	0.025	0.0114	0.0011
SSE	θ_x	20.8	0.67	0.012
	θ_y	41.8	1.77	0.028
	θ_z	3.9	0.18	0.0025
MSE	θ_x	$3.1e^{-4}$	$1e^{-5}$	$1.8e^{-7}$
	θ_y	$6.3e^{-4}$	$2.7e^{-5}$	$4.2e^{-7}$
	θ_z	$5.9e^{-5}$	$2.8e^{-6}$	$3.7e^{-8}$
MAE	θ_x	0.011	0.0024	$2.9e^{-4}$
	θ_y	0.016	0.0038	$4.5e^{-4}$
	θ_z	0.0056	0.0012	$1.5e^{-4}$

parison, an additional test data has been used in the training process for the neural networks, to test the generalizability of the networks as given in Table I. However for the second part of comparison only one set of sample trajectories is used within the reachable workspace of the manipulator, whose specifications are explained in the following section.

5.1. SAMPLE TRAJECTORY GENERATION

We consider a smooth motion specified in terms of a desired pose of the moving platform of the hydraulic shoulder. The sample trajectory is easily defined given the initial and final points and the time to reach the final point. Figure 3 shows the sample trajectory for each orientation angle in the task space of the hydraulic shoulder. Note that the link lengths profile corresponding to the desired task space trajectory could be easily obtained through the inverse kinematics map of the manipulator.

5.2. NEURAL NETWORK ESTIMATION

Figures 4(a)–(c), show the simulation results using the trained neural networks of different structures. Best representatives from each structure, selected from

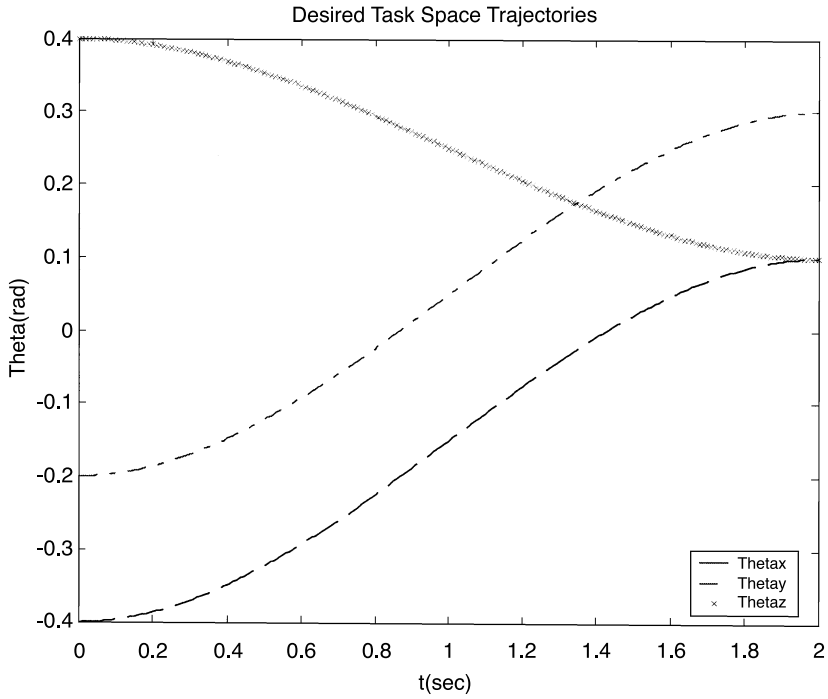

Figure 3. Sample trajectory for orientation angles.

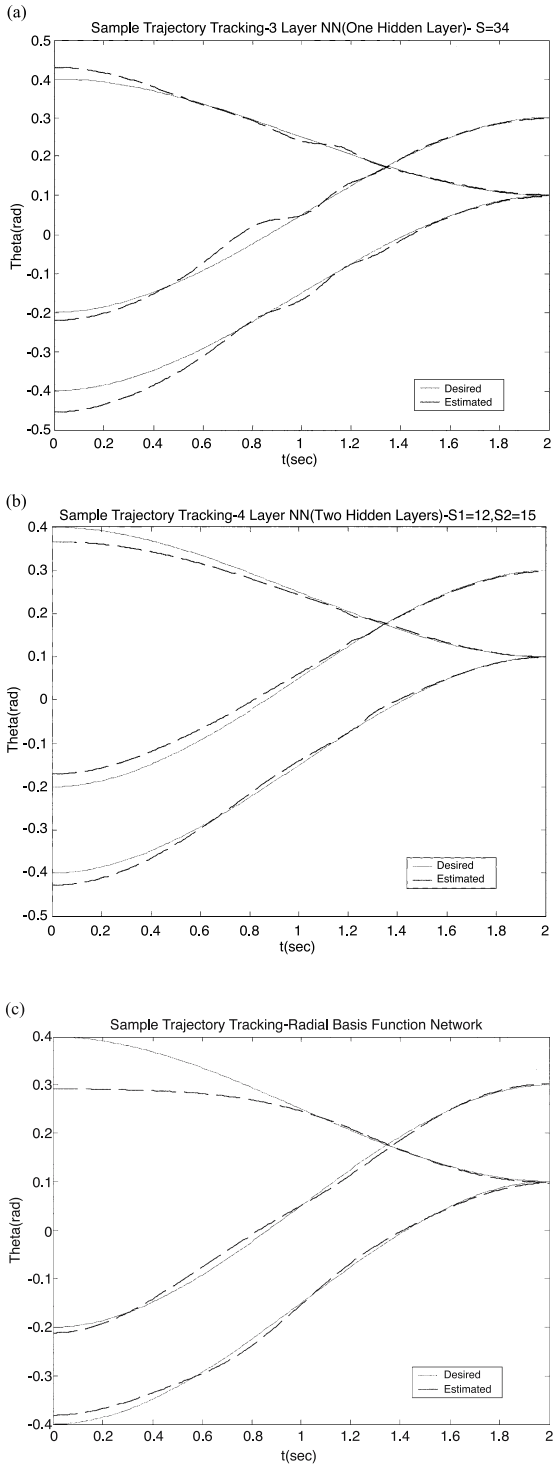

Figure 4. Tracking performance for different structures of neural networks.

Table I namely: Multilayer feed forward with 34 neurons in a single hidden layer, Multilayer feed forward with 12 and 15 neurons in two respective hidden layers, and also radial basis function network, were tested with the same sample trajectory. The Multilayer feed forward network with two hidden layers is seen to be the best compromise, compared to the other network structures, regarding the large number of neurons and weights in the RBF network.

5.3. QUASI-CLOSED METHOD SIMULATION

Figure 5 shows the simulation results for the Quasi-closed method applied to follow the sample trajectory along each orientation angle. The tracking performance is seen to be good but behind required for an accurate robotic application. However the method shows to be effective and comparable to the other methods proposed to solve the forward kinematics problem of the hydraulic shoulder in medium accurate applications.

5.4. TAYLOR SERIES EXPANSION

Figures 6(a)–(c), show the simulation results for the Taylor series method applied to follow the sample trajectory along each orientation angle. The tracking

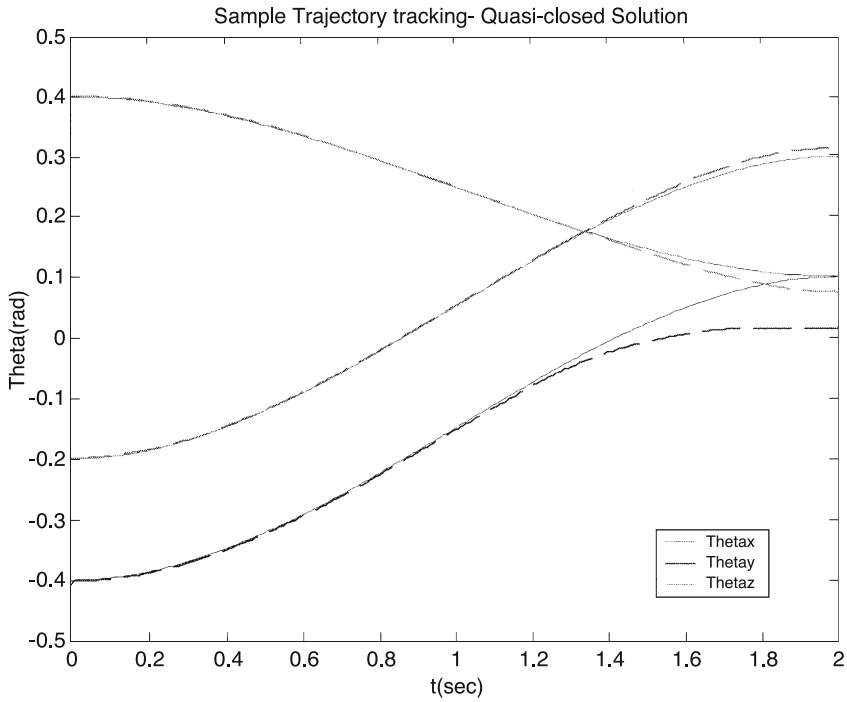

Figure 5. Sample trajectory tracking for quasi-closed solution method.

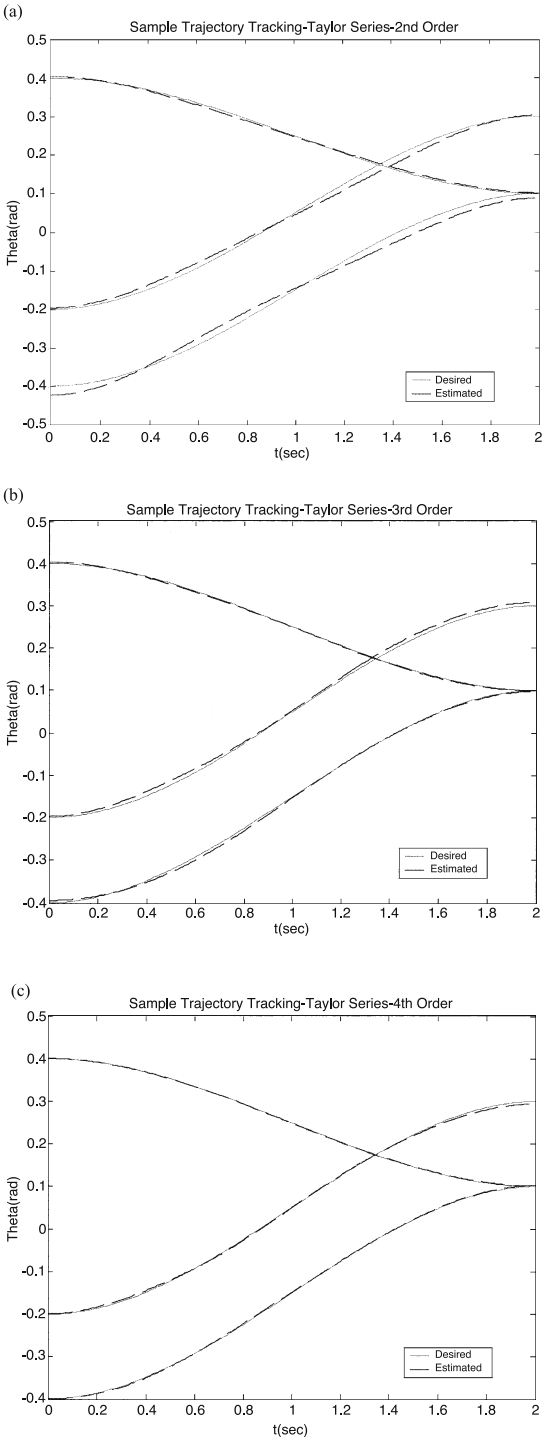

Figure 6. Tracking performance of Taylor series method with different orders of approximation.

performance is seen to improve as the order of approximation increases up to four. The results clearly show that the fourth order Taylor series approximation outperforms the other proposed methods tracking a typical robotic trajectory.

5.5. SIMULATION COMPARISON

Different measures were used for the tracking errors along the sample trajectory to test and compare the tracking performance of the methods. Table III summarizes the results, in which $E_{\max}$ stands for the maximum tracking error, SSE stands for sum of square of error, MSE stands for mean square error, and MAE stands for mean absolute error. As observed from Table III, the maximum approximation error reached by the suitable Neural network structures are limited to 0.03 radians (1.7°) and 0.086 (5°) for the quasi-closed method, which are way beyond required in an accurate robotic application.

Multilayer feed forward networks with two hidden layers show a slightly better performance compared to those with one hidden layer. The performance of RBF networks is also good; the training time for these networks is much less than their feed forward counterpart, but the weak point of such networks could be the big size leading to large number of neurons and weights. The main drawback of neural networks in this application would be the long training times and the big size of the networks resulting in much more number of weights compared to the number of coefficients used in Taylor series expansion. The accuracy sought

Table III. Measures of tracking errors.

Performance index		$E_{\max}$	SSE	MSE	MAE
Solution Method					
Fourth order Taylor expansion	θ_x	0.0028	0.0005	$2.5e^{-6}$	0.0013
	θ_y	0.0056	0.0018	$9.1e^{-6}$	0.0025
	θ_z	0.0017	0.00015	$7.5e^{-7}$	0.0006
Three layer feed-forward neural net ($s = 34$)	θ_x	0.0548	0.124	0.00061	0.017
	θ_y	0.0453	0.056	0.00028	0.011
	θ_z	0.0295	0.025	0.00013	0.0074
Four layer feed-forward neural net $s_1 = 12$, $s_2 = 15$	θ_x	0.028	0.032	0.00016	0.0091
	θ_y	0.03	0.069	0.00034	0.014
	θ_z	0.032	0.054	0.00027	0.012
RBF neural network	θ_x	0.018	0.019	$9.9e^{-5}$	0.008
	θ_y	0.017	0.016	$8.3e^{-5}$	0.0074
	θ_z	0.1	0.53	0.0026	0.033
Quasi-closed solution method	θ_x	0.086	0.225	0.0011	0.0193
	θ_y	0.014	0.0072	$3.6e^{-5}$	0.0041
	θ_z	0.025	0.0171	$8.5e^{-5}$	0.0053

by the different networks is also comparable only to a Taylor approximation of third order.

The quasi-closed form solution method uses a combination of analytic and numerical computation schemes. Hence the degree of accuracy of the method mainly depends on the numerical optimization scheme to find an orthonormal rotation matrix which is considered very important. This method provides good results comparable to neural network structures and the third order Taylor series approximation, yet behind demanded in an application such as feedback position control of the hydraulic shoulder. Another main drawback of this method is its dependency on the initial conditions used in the optimization process.

The maximum error of approximation in fourth order Taylor series is at least 10 times better than that of other methods and typically about 0.005 radians (0.14°). It was observed that the number of the coefficients in the expansion will increase with the order of approximation resulting in a better degree of accuracy but usually with a slight increase in the computation time. This result enables the designer to choose the appropriate order, according to the performance requirement. The accuracy sought by the fourth order approximation seems to be a good compromise for robotic applications such as our redundant parallel manipulator.

6. Conclusions

In this paper, three different approaches were presented to solve the forward kinematics problem in a three DOF actuator redundant hydraulic parallel manipulator. First, neural networks of different structures were introduced to solve the forward kinematics problem. Multilayer feed forward and Radial Basis networks were considered separately. Alternatively, a quasi-closed solution method was developed which used the rotation matrix of the end effector to determine the corresponding orientation angles needed to solve the forward kinematics map. Finally, the Taylor series expansion was used in a least square estimation problem to solve for the unknown coefficients of the map.

From the comparison study made in two major parts, it can be concluded that the fourth order Taylor series approximation provides the best compromise with acceptable prediction errors for robotic applications, relatively less offline computation effort, and easy online digital implementation, compared to the different structures of neural networks or the quasi-closed solution method proposed. Further attempts to increase the order of approximation seems not to be worth considering the computation time, number of coefficients and required accuracy in our application.

References

1. Baron, L. and Angeles, J.: The kinematic decoupling of parallel manipulators using joint-sensor data, *IEEE Trans. Robot Autom.* **16**(6) (Dec 2000), 644–651.

2. Bonev, I. A. et al.: A closed-form solution to the direct kinematics of nearly general parallel manipulators with optimally located three linear extra sensors, *IEEE Trans. Robot Autom.* **17**(2) (April 2001), 148–156.
3. Dasgupta, B. and Mruthyunjaya, T. S.: The Stewart platform manipulator: A review, Elsevier Science, *Mech. Mach. Theory* **35**(1) (2000), 15–40.
4. Didrit, O., Petitot, M., and Walter, E.: Guaranteed solution of direct kinematic problems for genera1 configurations of parallel manipulators, *IEEE Trans. Robot Autom.* **14**(2) (April 1998), 259–266.
5. Geng, Z. and Haynes, L.: Neural network solution for the forward kinematics problem of a Stewart platform. in: *Proc. Of the 1991 IEEE Int'l Conf. on Robotics and Automation*, Sacramento, CA, Vol. 3, April 1991, pp. 2650–2655.
6. Hayward, V.: Borrowing some design ideas from biological manipulators to design an artificial one. in: *Robots and Biological System, NATO Series*, Springer, Berlin Heidelberg New York, 1993, pp. 135–148.
7. Hayward, V.: Design of a hydraulic robot shoulder based on a combinatorial mechanism, Experimental Robotics III: *The 3rd. Int'l Symposium, Japan, October 1994. Lecture Notes in Control & Information Sciences*, Springer, Berlin Heidelberg New York, 297–310.
8. Hayward, V. and Kurtz, R.: Modeling of a parallel wrist mechanism with actuator redundancy, Int'l J Laboratory Robotics and Automation, VCH Publishers **4**(2) (1992), 69–76.
9. Lee, S. and Kil, R. M.: Robot kinematic control based on bidirectional mapping neural network, in: *Proc. IJCNN*. San Diego, CA, Vol. 3, 1990, pp. 327–335.
10. Merlet, J. P.: Parallel robots: Open problems, in: *9th Int'l. Symp. of Robotics Research*. Snowbird, 9–12 October 1999.
11. Merlet, J. P.: Still a long way to go on the road for parallel mechanisms, ASME 2002 DETC Conference. Montreal, Canada, 2002.
12. Merlet, J. P.: Direct kinematics of planar parallel manipulators. *Int'l Conf On Robotics and Automation*, Minnesota, April 1996, 3744–3749.
13. Merlet, J. P.: Closed-form resolution of the direct kinematics of parallel manipulators using extra sensors data. in: *IEEE Int'l Conf Robotics and Automation* 1993, 200–204.
14. Nguyen, L., Patel, R.V., and Khorasani, K.: Neural Network Architectures for the forward kinematics problem in robotics. in: *Proc. of the Joint IEEE International Conference on Neural Networks*, San Diego, 1990, pp. 393–399.
15. Norgaard, M.: Neural network based system identification toolbox, Tech. Report 00-E-891, Department of Automation, Technical University of Denmark, 2000.
16. Sadjadian, H. and Taghirad, H. D.: Numerical methods for computing the forward kinematics of a redundant parallel manipulator. in: *IEEE Conference on Mechatronics and Robotics*, Aachen, Germany, Sept. 2004, pp. 557–562.
17. Sang, L. H. and Han, M. C.: The Estimation for forward kinematic solution of Stewart platform using the neural network, in: *Proc. Of the 1999 IEEERSJ Int'l Conf on Intelligent Robots & Systems* 1999, pp. 501–506.
18. Song, S. K. and Kwon, D. S.: Efficient formulation approach for the forward kinematics of the 3–6 Stewart-Gough platform, in: *Int'l. Conf. on Intelligent Robots and Systems*, Hawaii, USA, Oct. 2001, pp. 1688–1693.
19. Wang, D. and Zilouchian, A.: Solutions of kinematics of robot manipulators using a kohonen self organizing neural network, in: *Proc. Of the 1997 IEEE Int'l Symp. on intelligent control*. Turkey, July 1997, pp. 251–255.
20. Siciliano, B.: The tricept robot: Inverse kinematics, manipulability analysis and closed-loop direct kinematics algorithm, *Robotica* **17** (June 1999), 437–445.

21. Fattah, A. and Kasaei, G.: Kinematics and dynamics of a parallel manipulator with a new architecture, *Robotica* **18** (Sept. 2000), 535–543.
22. Joshi, S. and Tsai, L. W.: The kinematics of a class of 3-DOF, 4-Legged parallel manipulators, *Trans of the ASME* **125** (March 2003), 52–60.
23. Joshi, S. and Tsai, L. W.: A comparison study of two 3-DOF parallel manipulators: One with three and the other with four supporting legs, *IEEE Trans. Robot Autom.* **19**(2) (April 2003), 200–209.
24. Ouerfelli, M., Kumar, V. and Harwin, W. S.: Methods for kinematic modeling of biological and robotic systems, Technical note, Elsevier, *Med. Eng. Phys.* **22** (2000), 509–520.
25. Aspragathos, N. A. and Dimitros, J. K.: A comparative study of three methods for robot kinematics, *IEEE Trans. Syst. Man Cybern., Part B, Cybern.* **28**(2) (April 1998), 135–145.
26. Yee, C. S. and Lim, K. B.: Forward kinematics solution of Stewart platform using neural networks, Elsevier Science, *Neurocomputing* **16** (1997), 333–349.