Intelligent Systems Based Solutions for the Kinematics Problem of the Industrial Robot Arms

Emre SARIYILDIZ
Department of System Design Engineering
Keio University
Yokohama, Japan
emre@sum.sd.keio.ac.jp

Kemal UCAK
Department of Control Engineering
Istanbul Technical University
Istanbul, Turkey
kemal.ucak@itu.edu.tr

Kouhei OHNISHI
Department of System Design Engineering
Keio University
Yokohama, Japan
ohnishi@sd.keio.ac.jp

Gulay OKE
Department of Control Engineering
Istanbul Technical University
Istanbul, Turkey
gulay.oke@itu.edu.tr

Hakan TEMELTAS
Department of Control Engineering
Istanbul Technical University
Istanbul, Turkey
temeltas@elk.itu.edu.tr

Abstract— **In this paper, three intelligent system methods namely Artificial Neural Network (ANN), Support Vector Machine (SVM) and Adaptive Neuro Fuzzy Inference Systems (ANFIS), are implemented to solve the inverse kinematics problem of the industrial robot arms. The main advantages of the intelligent system based solutions in the robot kinematics are that they can be easily implemented in analysis of complex mechanisms and their solutions do not suffer by the singularity that is one of the fundamental problems of inverse kinematics. The screw theory and quaternion algebra based kinematic model is used to improve the model efficiency by decreasing the computational complexity and load. The kinematics problem of the Staubli TX-60L industrial robot arm is analyzed by using the proposed intelligent system based solutions and simulation results are given.**

Keywords- ***Artificial Neural Networks, ANFIS, Redundancy, Robot Arm, Singularity, Support Vector Machine, Trajectory Tracking***

I. INTRODUCTION

The kinematics is the analysis of the motion without considering the forces and torques which causes the motion[1]. The kinematics based robot control is widely used in industrial applications due to their computational efficiency, safety, reliability etc. [2 and 3].

Several methods have been proposed to solve the kinematics problem of industrial robot arms [4 and 5]. Among them, geometric and analytic based solutions are widely used to solve the kinematics problem of industrial robot arms [5]. However, these solutions can only be implemented if the robot has specific mechanical structures, such as intersection of elbow joints, non-redundant robot manipulators which have less than six joints and so on. Therefore, the industrial robot arms, in general, have been designed by considering the mechanical structure constraints of these solutions. The differential kinematics is another method that has also been widely used in many industrial robot arm control systems [6 and 7]. The main advantage of the differential kinematics is that it can be easily implemented to solve the kinematics problem of any kind of mechanisms. Also, accurate and efficient kinematic based trajectory tracking applications can be easily implemented by using this method [8]. Jacobian is used as a velocity mapping operator, which transforms the joint velocities into the Cartesian linear and angular velocities of the end effector, in differential kinematics. Highly complex and nonlinear inverse kinematics problem of robot manipulators can be easily solved by just taking inverse of the Jacobian matrix operator. However, it has several disadvantages. The first one is that the differential kinematics based solutions are locally linearized approximation of the inverse kinematics problem [9 and 10]. Thus, we can only obtain the approximate solutions of the inverse kinematic problem by using this method. Although the solution results of this method are not real, the approximation results are generally quite sufficient. The second disadvantage of this method is that it has heavy computational loads and big computational time because of numerical iterative approach. To obtain the inverse kinematic solution, we need to calculate the inverse of the Jacobian matrix. Taking inverse of a matrix is generally a hard task, and the singularity is one of the main problems of this calculation. Several researches have been proposed to solve the singularity problem of inverse kinematics. There are, in general, four main techniques to cope with the singularity problem of robot manipulators. These are avoiding singular configuration method, robust inverse kinematics method, a normal form approach method and the extended Jacobian method [11 - 14]. However given techniques have some disadvantages which include computational load and errors. And the last disadvantage of the differential kinematic method is that it requires numerical integration, which suffers from numerical errors, to obtain the joint positions from the joint velocities [10].

In recent years, Artificial Intelligence (AI) based solution methods are proposed to solve the kinematics problem of robot manipulators [10,15,16,22,23]. There are two important advantages of AI based kinematics solution methods. The first one is that these methods do not suffer from the complex and highly nonlinear inverse kinematic equations, and they can be easily implemented to analyze any kind of mechanism. And the second one is that ANN based inverse kinematic solution methods do not suffer from singularity problem.

In ANN and ANFIS, the objective function is non-convex so the models can only be obtained locally. Support Vector Regression (SVR) is another intelligent system identification method ensuring global minimal solution. The non-convex objective function in primal form of SVR is converted to a convex one in dual formulation. Thus, the solution of SVR doesn't get stuck at local minima. In recent years, SVM theory has been used in many identification problems, instead of ANN approach. In this paper, ANN, ANFIS and SVR models are used to solve the kinematic problem of the 6-DOF robot arm. These methods are compared with respect to their validation performance and simulation results are given.

The rest of the paper is organized as follows. The kinematic model of the industrial robot arm is given in section II. In section III, a brief overview of the intelligent methods are given. Simulation results are presented in section IV. The paper ends with a brief conclusion in Section V.

II. KINEMATICS MODEL OF THE INDUSTRIAL ROBOT ARM

The training is implemented by using the forward kinematics model of the industrial robot arm. This model was derived by the authors using the screw theory and quaternion algebra. The main advantages of the screw theory and quaternion algebra based solution in robot kinematics is that it improves the performance of the intelligent systems by simplifying the kinematics model and decreasing the computational load. The kinematics model of the robot manipulators can be derived as follows [17 and 18]

Step 1: Determine joints' axis and moment vectors: Firstly, the axis vectors which describe the motion of the joints are attached. Then, the moment vectors of these axes are obtained for revolute joints. Hence, the Plücker coordinate notations of these axes are obtained.

Step2: Obtain transformation operators: Dual-quaternion transformation operators can be obtained as follows

$$\hat{q}_i = (\hat{q}_{Si}, \hat{\mathbf{q}}_{\mathbf{Vi}}),\ \hat{q}_i = q_i + \varepsilon q_i^o \tag{1}$$

For prismatic joints:

$$q_i = (1,0,0,0) \quad q_i^o = (0, q_1^o, q_2^o, q_3^o).$$

For revolute joints:

$$q_i = \cos\left(\frac{\theta_i}{2}\right) + \sin\left(\frac{\theta_i}{2}\right)\mathbf{d_i}, q_i^o = \frac{1}{2}(p_i - q_i \otimes p_i \otimes q_i^*) \otimes q_i, q_i^o = [0, \sin\left(\frac{\theta_i}{2}\right)\mathbf{m_i}]$$

where $i = 1, 2, \cdots n.$

Step3: Formulate rigid motion: The general rigid body transformation can be formulized for n degrees of freedom robot manipulator given by

$$\hat{q}_{1n} = \hat{q}_1 \Theta \hat{q}_2 \Theta .....\Theta \hat{q}_n \tag{2}$$

where $\hat{q}_{1n} = q_{1n} + \varepsilon q_{1n}^o$. The orientation and position of the end effector can be found as follows,

Let, $\hat{l}_n = l_n + \varepsilon l_n^o$ and $\hat{l}_{n-1} = l_{n-1} + \varepsilon l_{n-1}^o$ be the n^{th} and $n-1^{th}$ joints' Plücker coordinate representations, respectively. Let $\hat{l}_n' = l_n' + \varepsilon l_n^{o\prime} = \hat{q}_{1n} \Theta \hat{l}_n \Theta \hat{q}_{1n}^*$ and $\hat{l}_{n-1}' = l_{n-1}' + \varepsilon l_{n-1}^{o}{}' = \hat{q}_{1n-1} \Theta \hat{l}_{n-1} \Theta \hat{q}_{1n-1}^*$ be, respectively, the n^{th} and $n-1^{th}$ joints' Plücker coordinate representations after the transformation. The orientation of the end effector is $\hat{l}_n'$. The position of the end effector can be found using

$$\begin{aligned}\mathbf{p}_n = &(V\left\{R\left\{\hat{q}_{1n}\Theta\hat{l}_n\Theta\hat{q}_{1n}^*\right\}\right\} \times V\left\{D\left\{\hat{q}_{1n}\Theta\hat{l}_n\Theta\hat{q}_{1n}^*\right\}\right\}) + \\ &(V\left\{R\left\{\hat{q}_{1n-1}\Theta\hat{l}_{n-1}\Theta\hat{q}_{1n-1}^*\right\}\right\} \times V\left\{D\left\{\hat{q}_{1n-1}\Theta\hat{l}_{n-1}\Theta\hat{q}_{1n-1}^*\right\}\right\}) \\ &\cdot V\left\{R\left\{\hat{q}_{1n}\Theta\hat{l}_n\Theta\hat{q}_{1n}^*\right\}\right\} * V\left\{R\left\{\hat{q}_{1n}\Theta\hat{l}_n\Theta\hat{q}_{1n}^*\right\}\right\}\end{aligned} \tag{3}$$

III. INTELLIGENT METHODS

Complex nonlinear dynamics can be identified utilizing the powerful generalization performance of artificial intelligent methods. In this section, the basic principles of the ANN, ANFIS and SVM are given .

A. Artificial Neural Network

In this section, the basics of NN structure and its learning algorithm are presented. The architecture of the MLP NN utilized to approximate nonlinear kinematic equation of manipulator in this study is illustrated in fig. 1. $w_{j,i}^h$'s are the weights from the i^{th} input to the j^{th} neuron and b_j^h is the bias of the j^{th} neuron in the hidden layer, $w_{1,j}^o$ is the weight from the j^{th} neuron to the output and b_1^o is the bias of the output in the output layer. Tan-sigmoid function has been chosen as the activation function in hidden layer. The nonlinear regression function obtained by means of the NN structure is given as follows:

$$\begin{aligned}\hat{y}_n &= \sum_{j=1}^{S} w_{1,j}^o h(d_{j,n}) + b_1^0 \\ d_{j,n} &= \sum_{i=1}^{k} w_{j,i}^h x_i + b_j^h\end{aligned} \tag{4}$$

where $h(.)$ denotes activation function. The adjustable parameters to be optimized are the weights and biases in layers. The objective function is selected as in (5):

$$F = \sum_{n=1}^{N} [\,y_n - \hat{y}_n\,]^2 \tag{5}$$

The adjustable parameters are optimized using Levenberg Marquard algorithm as learning algorithm as given in (6)

$$W^{new} = W^{old} - (J^T J + \mu I)^{-1} J^T \underline{e} \tag{6}$$

where $\underline{e}$ is training error vector and W is the parameter vector including all adjustable parameters in the NN.

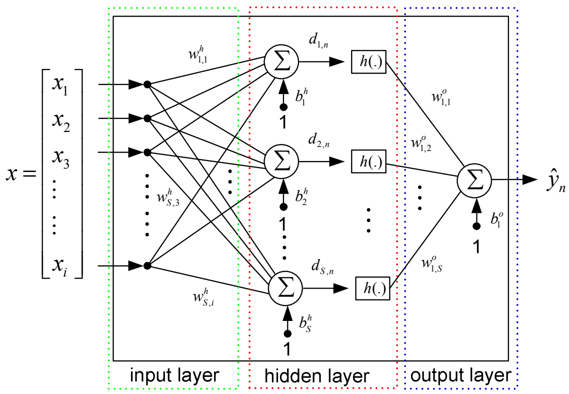

Fig. 1. NN Structure

B. Adaptive Neuro-Fuzzy Inference System

The structure of ANFIS which is functionally equivalent to a Sugeno fuzzy model is as in fig. 2[23, 24]. The structure consists of 5 layer. The layers and their function can be summarized as follows:

Layer 1 : The membership degree of the inputs are computed in this layer. The membership values directly depends on the membership function. Since the membership functions have adjustable parameters, this layer can be called adaptive layer and the parameters in this layer are referred as premise parameters.

Layer 2 : The firing strengths of the each rules are assessted in this layer. The layer is called fixed layer labeled Π . w_i's are the firing strengths given by $w_{k(i-1)+j} = \mu_{A_i}(x_1)\mu_{B_i}(x_2)$, $i = 1,2,...,k$, $j = 1,2,..,l$ where μ_{A_i}'s and μ_{B_i}'s membership functions.

Layer 3 : The firing strengths are normalized in this layer. $\overline{w}_i$'s denote the normalized firing strengths given

$$\overline{w}_i = \frac{w_i}{\sum_{j=1}^{R} w_j}$$

$R = k.l$ is the number of the rules.

Layer 4 : The parameters of this node are $\{p_i, q_i, r_i\}$ and referred to as consequent parameters. This layer includes adaptive parameters. f_i's are given by $f_i = p_i x_1 + q_i x_2 + r_i$.

Layer 5 : Outputs of the fourth layer are collected in this layer. The regression function obtained by means of the ANFIS structure is given as follows:

$$\hat{y}_n = \sum_{i=1}^{R} \overline{w}_i f_i \tag{7}$$

Layer 1 and layer 4 are called adaptive layers. Owing to this, a nonlinear function can be approximated optimizing the parameters in this layer.

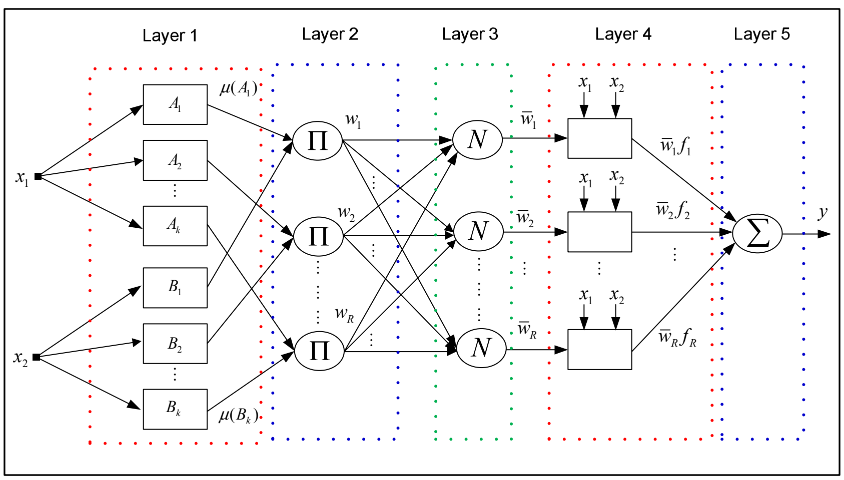

Fig. 2. ANFIS Structure

Owing to the fact that ANFIS has MISO structure, it requires the use of multiple ANFIS for MIMO identification problems . Thus, it has been employed different ANFIS for each output.

C. Support Vector Regression

The network structure of SVR is illustrated in fig. 3. Given a training data set:

$$(y_1, \underline{x}_1).....(y_k, \underline{x}k)\ ,\ \ x \in R^n, y \in R \quad k = 1,2,,,,N\ . \tag{8}$$

where N is the size of training data and n is the dimension of the input matrix, can be approximated by a linear function, with the following form,

$$f(\underline{x}) = \langle \underline{w}, \underline{x} \rangle + b \tag{9}$$

where $\langle .\ ,\ . \rangle$ denotes the inner product.

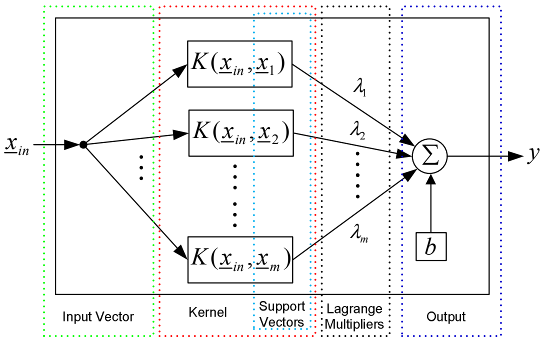

Fig. 3. SVR Network Structure

The training data, not separable by a linear plane in the input space, can be mapped to a high dimensional feature space using kernel where linear regression can be successfully performed. Linear regression techniques are then applied in the high-dimensional feature space The optimum solution for regression problem is obtained by the minimum of the following optimization problem:

$$\min_{\alpha,\alpha^*} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\left\langle x_i,x_j\right\rangle - \sum_{i=1}^{N}\alpha_i(y_i-\varepsilon)-\alpha_i^*(y_i+\varepsilon) \tag{10}$$

subject to

$$0\le\alpha_i\le C \quad i=1,2,3,,,,,,,,,,l$$

$$0\le\alpha_i^*\le C \quad i=1,2,3,,,,,,,,,,l$$

$$\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)=0$$

The objective function in (10) is convex and therefore a global solution of the problem is ensured. Using quadratic programming solver, the optimal Lagrange multipliers are obtained. The training data with non-zero Lagrange multipliers are called support vectors. The solution of the regression problem can be approximated by the support vectors and the corresponding Lagrange multipliers as follows:

$$f(\underline{x})=\sum_{i\in SV}\lambda_i K(\underline{x}_i,\underline{x})+b\,,\ \lambda_i=\alpha_i-\alpha_i^* \tag{11}$$

where

$$b=\frac{1}{N}\sum_{i=1}^{N}(y_i-\sum_{i\in SV}\lambda_i K(\underline{x}_i,\underline{x}))$$

and

$$K(\mathbf{x},\mathbf{y})=\exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right) \tag{12}$$

For detailed information we refer to [19-22].

IV. Simulation Results

The kinematics problem of the 6-DOF robot arm, illustrated in fig. 4, is worked out in this section. The kinematics structure of the robot manipulator is designed by considering the Stäubli RX 60L industrial robot manipulator.

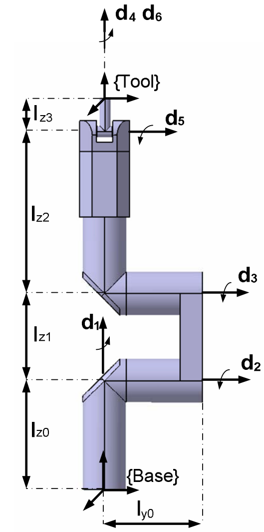

Fig. 4. 6-DOF Robot Arm

A random trajectory signal has been applied to the each joints of the robot manipulator in order to reveal all possible nonlinear kinematics dependencies of the robot manipulator. Then, 600 data are selected randomly for training process and 500 for validation. The inputs and outputs of the intelligent models are as in fig. 5.

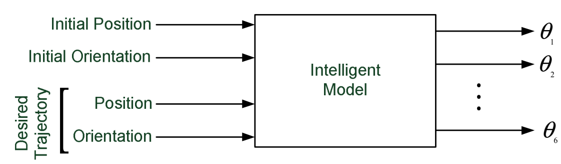

Fig. 5. ANN-SVR Models

The modeling problem is converted to finding the solution of the regression problem. The same training data set has been utilized in ANN, ANFIS and SVM. Six separate SVR MISO structures have been combined to model the MIMO model of the manipulator with SVM model, and six separate ANFIS MISO structures have been utilized to configure the MIMO model of the manipulator with ANFIS model. In ANN, 20 neuron has been utilized in hidden layer. The model parameters for SVM has been chosen $\varepsilon_1=0.01$, $\sigma_1=0.75$, $\varepsilon_2=0.01$, $\sigma_2=1.5$, $\varepsilon_3=0.007$, $\sigma_3=0.25$, $\varepsilon_4=0.01$, $\sigma_4=1.5$, $\varepsilon_5=0.01$, $\sigma_5=0.25$, $\varepsilon_6=0.01$, $\sigma_6=1.5$. The approximation results of the intelligent models are depicted in fig. 6-7 for all links. The results indicates that the dynamics of the manipulator can be modeled using ANN,ANFIS and SVM with the proper parameter set.

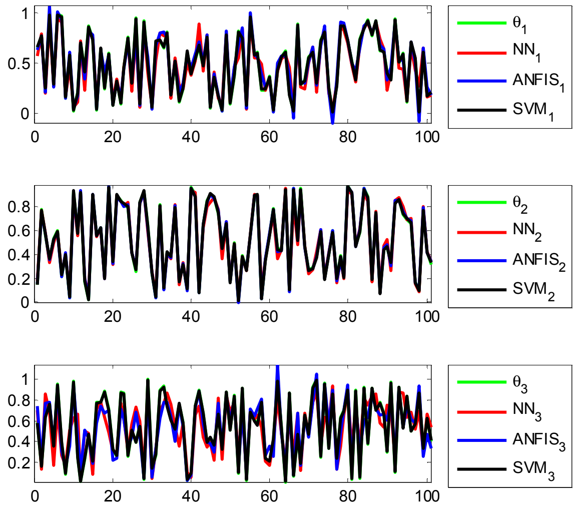

Fig. 6. Model Outputs for θ_1 , θ_2 and θ_3

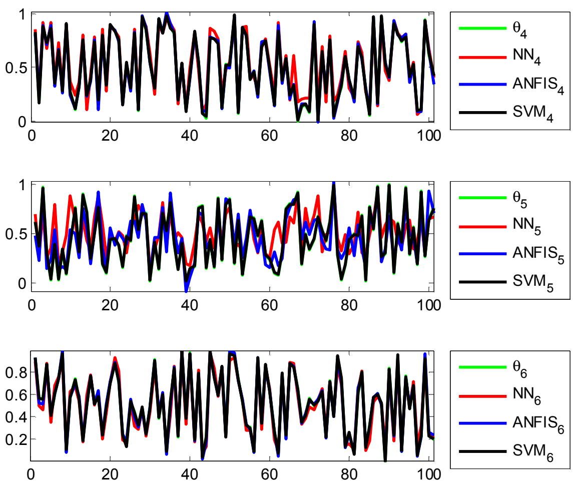

Fig. 7. Model Outputs for θ_4 , θ_5 and θ_6

TABLE I. VALIDATION ERRORS

e_v	Validation Errors		
	ANN	***ANFIS***	***SVM***
e_{v1}	0.0614	0.0482	0.0094
e_{v2}	0.0248	0.0118	0.0083
e_{v3}	0.1355	0.0915	0.0070
e_{v4}	0.0494	0.0170	0.0088
e_{v5}	0.1837	0.1221	0.0099
e_{v6}	0.0403	0.0160	0.0093

The averaged validation errors for graphs in figure 6-7 are given in table 1. The objective function in ANN and ANFIS are non-convex, so these result in local models. As can be seen from figure 6-7 and table 1, SVM model has better results owing to ensuring global solution of the optimization problem.

V. CONCLUSION

In this paper, the kinematic problem of a 6-DOF arm is solved by using ANN,ANFIS and SVM. The intelligent methods do not suffer from singularity problems and this methods can be easily applied for the kinematic modeling of robot manipulator. Modeling performance depends on the network structure and the parameters to be tuned. ANN and ANFIS have non-convex objective functions, so regression models can be obtained locally. SVM based solutions give better results due to the fact that it doesn't get stuck at local minima and ensure global solution of the identification problem.

In the future works, dynamic modeling of robot manipulator can be studied by using intelligent methods.

REFERENCES

[1] B. Siciliano, O. Khatib, "Handbook of Robotics", *Springer,*2008

[2] D. Bertram, J. Kuffner, R. Dillmann, T. Asfour, "An Integrated Approach to Inverse Kinematics and Path Planning for Redundant Manipulators", *IEEE, International Conference on Robotics and Automation (ICRA)*, May., 2006, pp. 1874-1879,

[3] M.Murray, Z.Li and S.S. Sastry, A mathematical introduction to robotic manipulation, Boca Raton FL:CRC Press,1994

[4] M.W. Spong, S. Hutchinson, M. Vidyasagar, Robot Modeling and Control, *Wiley*, 2005

[5] J.J. Craig, Introduction to robotics: Mechanics and Control, 3rd ed.*, Prentice Hall*, 2006

[6] F. Caccavale, P. Chiacchio, S. Chiaverini, B. Siciliano, "Experiments of Kinematic Control on a Redundant Robot Manipulator", *Laboratory Robotics and Automation*, vol. 8, pp. 25–36, 1996

[7] P. Chiacchio, S. Chiaverini, L. Sciavicco, and B. Siciliano, "Closed-Loop Inverse Kinematics Schemes for Constrained Redundant Manipulators with Task Space Augmentation and Task Priority Strategy", *Int. Journal of Robotics Research*, vol. 10, pp. 410-425, 1991.

[8] C.C. Cheah, M. Hirano, S. Kawamura, S. Arimoto, "Approximate Jacobian Control for Robots with Uncertain Kinematics and Dynamics", *IEEE Transactions on Robotics and Automation*, vol.19, no. 4, Aug., 2003

[9] P. H. Chang, "A Closed-Form Solution for Inverse Kinematics of Robot Manipulators with Redundancy", *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 5, Oct. ,1987

[10] A.T. Hasan, N. Ismail, A.M.S Hamouda, I. Aris, M.H. Marhaban, H.M.A.A. Al-Assadi, "Artificial Neural Network-Based Kinematics Jacobian Solution for Serial Manipulator Passing through Singular Configurations", *Advances in Engineering Software,* vol.41, pp. 359–367, 2010.

[11] Hu, Z., FU, Z., Fang, H., "Study of singularity robust inverse of Jacobian matrix for manipulator", International Conference on Machine Learning and Cybernetics, Beijing, pp. 406–410, 2002.

[12] Y. Nakamura, H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control", Journal of Dynamic Systems, Measurement and Control, vol.108, pp.163–71, 1986.

[13] C.W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods", IEEE Transactions on Systems, Man and Cybernetics,Vol.16, pp. 93–101, 1986.

[14] A. Balestrino, G. De Maria, L. Sciavicco, "Robust control of robotic manipulators", International proceedings of the 9th IFAC world congress, Budapest, vol.6, pp.80–85, 1984.

[15] B. Karilk, S. Aydin, "An improved approach to the solution of inverse kinematics problems for robot manipulators", J Eng Appl Artif Int 2000;13: 159–164.

[16] T. Ogawa, H. Matsuura, H. Kanada. "A Solution of inverse kinematics of robot arm using network inversion", In: Proceedings of

the 2005 international conference on computational intelligence for modelling, control and automation.
[17] E. Sariyildiz, H. Temeltas, “A New Formulation Method for Solving Kinematic Problems of Multi-Arm Robot Systems Using Quaternion Algebra in the Screw Theory Framework” , Tr. J. Elec. Eng. & Comp. Sci., 2012, Vol. 20, No.4, 607-628
[18] E. Sariyildiz, E. Cakiray, H. Temeltas, “A Comparative Study of Three Inverse Kinematic Methods of Serial Industrial Robot Manipulators in the Screw Theory Framework”, Int. J. Adv. Robotic Sys., 2011, Vol. 8, No. 5, 9-24
[19] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and other Kernel based Learning Methods, Cambridge University Press, 2000.
[20] J.A.K. Suykens, ”Nonlinear modeling and support vector machines” IEEE Instrumentation and Measurement Technology Conference Budapest, Hungary, May 2001, pp. 287-294.
[21] A.J. Smola, B. Scholkopf, “A tutorial on support vector regression” Statistics and Computing, vol. 14, pp. 199-222, August 2004.
[22] E. Sariyildiz, K. Ucak, G. Oke and H. Temeltas, ”A Trajectory Tracking Application of Redundant Planar Robot Arm via Support Vector Machines”, International Conference on Adaptive and Intelligent Systems, LNCS 2011, vol. 6943/2011, pp. 192-202, September 2011.
[23] S. Alavander, M.J. Nigam, "Neuro-Fuzzy based Approach for Inverse Kinematics Solution of Industrial Robot Manipulators" International Journal of Communications & Control, vol. III, no. 3, pp. 224-234, 2008.
[24] J-S.R. Jang, C.-T. Sun, E. Mizutani, Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence, Prentice Hall ,1997