Robotics System Optimal Task Control (Neuro-Inverse Kinematics Approach)

Ebrahim A. Al-Gallaf

Department of Electrical and Electronics Engineering
College of Engineering, University of Bahrain.

P. O. Box 13184, Kingdom of Bahrain.

***Abstract* — A fast and efficient method for computing optimal grasping and manipulation forces is presented based on a Quadratic Optimisation formulation for a hand robotics system, where computation has been based on using the non-linear factual model of contacts. Furthermore, in order to achieve grasping while in motion, the Hand Inverse Jacobian has to be intensively computed, consequently, we investigate an efficient approach of employing an Artificial Neural Network for the multi-finger robot hand in which the object motion is defined in. The approach followed here is to let an ANN to learn the nonlinear Inverse Kinematics functional relating the hand joints positions and displacements to object displacement.**

***Index Terms* — Robotics Control, Manipulation, Neural Networks, Task Optimization.**

I. INTRODUCTION

Dexterous, multi-fingered grippers have been the subject of considerable research [1], with the kinematics and force control issues being investigated in [2]. Research has been carried out in the subject of dexterous manipulation and hand maneuvers, some including the dynamics of the hand. **A**rtificial **I**ntelligence (**AI**) and heuristics techniques have been introduced by many researchers in the area of robot control and motion planning. Neural networks have been heavily employed in robotics technology such as robot arm visual control as were introduced by [3-4], inverse kinematics problem of six degree of freedom robot arm as done by [5], the research introduced by [6] in which they employ a real-time learning neural robot controller for solving the inverse kinematics problem, and the research introduced by [6], in which they employ an artificial neural network for tracking and grasping a moving object observed by a six degrees of freedom robotics arm system.

A method has been proposed in [7] to reduce the computational burden by using a routine based on GIVENS rotations. Previous research contributions in task space measure can be found in [8], static and dynamic manipulability ellipsoids were used. Another employment of artificial neural networks has been the one presented by [9]. The conception of the control system was based on the combination of a neural network for adaptation of grasp parameters and a fuzzy logic approach for correction of parameters values given to a conventional controller. In addition, [10] has proposed a *fuzzy-neural* system control system that learns specific hand-object mapping, without considering the issue of interaction forces among the fingers. The research presented here is novel in a sense that, Jacobian hand inverse has been avoided to compute.

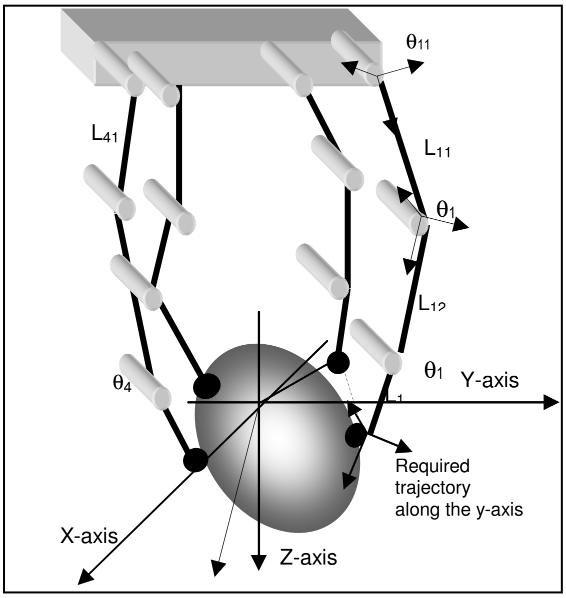

Fig. 1. The robotics system, an object to be maneuvered.

II. ROBOTICS SYSTEM DYNAMICS FORMULATION

Dynamic behavior of a grasp is defined as the time response of the grasp for changes in its motion or force trajectories, Fig. (1). The main control objective is to steer a grasped object to track a defined path (*which we shall designate as Task-Space Path*). Defining the Cartesian based posture error of the grasped object $e \in \Re^{6\times 1}$: as $e \cong u_d^c - u_a^c$ which is the difference of desired position-orientation vector of the object u_d^c and the actual position-orientation of the object u_a^c. If

$$e \cong u_d^c - u_a^c \quad \text{and} \quad e \cong u_d^c - u_a^c \qquad (1)$$

object-hand contact system of motion can be described in terms of τ_h, the applied joint torques, and described in terms of τ_h, the applied joint torques :

$$\tau_h = M_h\left(J_h^{-1}G^{+1}\ddot{\Theta}_a - J_h^{-1}\dot{J}_h\dot{\Theta}_h\right) + N_h + C_h + J_h^T G^{+}\left(M_o\ddot{\Theta}_a + N_h\dot{u} + C_h\right) + J_h^T\left(\phi_{cd} - Z_i\int\left(\phi_{cd} - \eta\lambda\right)\right)$$
$$\tau_h = M_h J_h^{-1}\left(G^{+1}\ddot{\Theta}_a - \dot{J}_h\dot{\Theta}_h\right) + T_{ex}$$
$$\tau_h = \left(M_h J_h^{-1} X_h + T_{ex}\right) \qquad (2)$$

where

$$\ddot{\Theta}_a = \left(\left(\dot{v}_o + \dot{\omega}_o\right)^T + k_p e + k_d \dot{e} + k_i \int_0^{\tau} e\, d\tau\right)$$

and

$$T_{ex} = N_k + C_k + J_h^T\left(\phi_{cd} - Z_i\int_0^{x}\left(\phi_{cd} - \eta\lambda\right)\right)$$
$$X_h = \left(G^{+1}\Theta_a - J_h\Theta_k\right) \in \Re^{12\times 1}$$

τ_h is the computed torque at each joint in the hand, M_h, N_h, and C_h are the hand dynamics. $k_p \cong diag(k_{p1} \ \cdots \ k_{p6})$, $k_d \cong diag(k_{d1} \ \cdots \ k_{d6})$, and $k_i \cong diag(k_{i1} \ \cdots \ k_{i6})$ with k_{pj}, k_{dj} and $k_{ij} > 0$ for all j are the corresponding cartesian PID controller parameters. The control law defined by (2) is that it depends on the hand inverse kinematics function and the hand inverse Jacobian. Computation of hand kinematics is not an easy task, specially once talking about real-time hand control. Hence, this is where the potential of employing the Artificial Neural Networks in the hand control can be seen. In addition to this, the problem of inverting the hand Jacobian matrix J_h^T, is a function of the four fingers all collectively. The control specified by (2) realizes not only the desired object trajectory, but also a desired internal grasping force, given by $J_h^T\left(\phi_{cd} - Z_i\int\left(\phi_{cd} - \eta\lambda\right)\right)$ as will be optimized afterward.

III. Nonlinear Optimal Force Distribution

A force f_{oi} and moment m_{oi} from each finger yield forces and moments f_{bi} and m_{bi} acting on the object computed from the eight vectors f_{oi} and m_{oi} according to a geometric relation :

$$f_{bi} = f_{oi} \qquad (3)$$
$$m_{bi} = m_{oi} + f_{oi} \times r_{oi} \quad i = 1, 2, 3, 4 \qquad (4)$$

in which r_{oi} defines a vector from the i^{th} contact location to the object centre of the gravity frame, as depicted in Fig. (1). Under the assumption that there is no change from the centre of each fingertip to the centre of the object, the external forces f_e and moments m_e on the object can be calculated from f_{bi} and m_{bi} as :

$$F_b^T = \left[f_{bi} \quad m_{bi}\right] \qquad (5)$$
$$F_b^T = \left[I_6 \quad I_6 \quad I_6 \quad I_6\right] f_{tip} \qquad (6)$$
$$F_b^T = G f_{tip}^T \qquad (7)$$

where $G \in \Re^{6\times 24}$; $I_6 \in \Re^{6\times 6}$ is a matrix given by :

$$I_6 = \left[I_a \quad I_b\right]$$

$$I_a = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \hline & S(r_{oi}) & \\ & & \\ & & \end{bmatrix} \in \Re^{6\times 3}$$

$$I_b = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \Re^{6\times 3}$$

$S(r_{oi})$ is *skew-symmetric* of r_{oi}, and f_{tip} is $\Re^{24\times 1}$ dimensional force vector :

$$f_{tip}^T = \left[f_{tip1} \quad f_{tip2} \quad f_{tip3} \quad f_{tip4}\right] \in \Re^{24\times 1}$$

i = 1, 2, 3, 4 (7)

Contact force vector for the entire hand is expressed as :

$$f_{tip}^T = \left[f_{tip1} \quad f_{tip2} \quad f_{tip3} \quad f_{tip4}\right] \in \Re^{12\times 1}$$
$$f_{tip}^T = \left[f_{x1} \ f_{y1} \ f_{z1} \ f_{x2} \ f_{y2} \ f_{z2} \ f_{x3} \ f_{y3} \ f_{z3} \ f_{x4} \ f_{y4} \ f_{z4}\right] \in \Re^{12\times 1} \qquad (8)$$

According to [8], the fingertip force vector associated with an object dynamic F_b is defined by :

$$f_{tip} = \left[GG^T\right]^{-1} G^T F_b + \eta\lambda \qquad (9)$$

Alternatively:

$$f_{tip} = f_{par} + f_{hom} \qquad (10)$$

here f_{par} is the *particular solution* presented by $\left[GG^T\right]^{-1} G^T F_b$, f_{hom} is the *null space* solution of the system given by $\eta\lambda$, which represents a force redundancy, and λ is a free vector that parameterizes the homogeneous part of the solution. External wrench or force vector F_b on an object grasped by the four fingers can be expressed in the following form :

$$F_b = c_{1(1)} w_{1(1)} + c_{2(1)} w_{2(1)} + \cdots\cdots c_{4(4)} w_{4(4)} \qquad (11)$$

in which $C_{1(1)}W_{1(1)} + C_{2(1)}W_{2(1)} + \cdots\cdots C_{4(4)}W_{4(4)}$ are four linearly independent contact wrenches with their respective intensities. From the contact kinematics point of view, the effect of the individual wrenches produces a redundancy in the force computation according to (7). In (8), it represents the solution of a force redundancy with possible adjustable force vector f_{hom} in such a way that the solution of f_{tip} satisfies the contact cone and actuator's torque constraints. Geometric constraints can be represented by the fact that the i^{th} normal force f^{c}_{zi} is acting in the same direction as the unit e_{ni} normal at the contact. Thus:

$$e_{ni} = \frac{Grad \quad S(r_i)}{\|Grad \quad S(r_i)\|} \quad \in \quad \Re^3 \tag{12}$$

for an object shape represented by $S(x,y,z)=0$. $S(x,y,z)$ is considered to be at least once differentiable at the points of contact. If the unknown contact force vectors onto the object are expressed at the contact coordinate frame and designated by :

$$f^{T}_{tipi} = \begin{bmatrix} f^{c}_{xi} & f^{c}_{yi} & f^{c}_{zi} \end{bmatrix} \tag{13}$$

then with the contact normal f^{c}_{z} along the z direction and directed outward with a coefficient of friction μ, the friction force cone constraints may be expressed as follows :

$$\begin{bmatrix} -1 & 0 & \frac{\mu}{\sqrt{2}} \\ 1 & 0 & \frac{\mu}{\sqrt{2}} \\ 0 & -1 & \frac{\mu}{\sqrt{2}} \\ 0 & 1 & \frac{\mu}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} f^{c}_{xi} \\ f^{c}_{yi} \\ f^{c}_{zi} \end{bmatrix} \le \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{14}$$

In finding a solution to (9) an optimisation is used. Due to Nahon and Angeles [36] whom formulated a *quadratic objective function.* Variables of optimisation are the internal forces' magnitudes, which are used to determine the amount of stress on an object to be manipulated according to the following linear equality and linear inequality constraints :

$$\begin{aligned} &\textit{Optimise} && \Phi(f_{tip}) \\ &\textit{Subject to} && G\, f_{tip} = F_b \\ & && \prod f_{tip} \le B \end{aligned} \tag{15}$$

where

$$B^T = \begin{bmatrix} P & P & P & P \vdots \tau_1 & \tau_2 & \tau_3 & \tau_4 \end{bmatrix} \quad \in \quad \Re^{28\times 1}$$

$$\textit{and} \quad P^T = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \in \Re^{4\times 1}$$

$$\tau^{T}_{i} = \begin{bmatrix} \tau_{1m} & \tau_{2m} & \tau_{3m} \end{bmatrix} \in \Re^{4\times 1}$$

The need for minimization of the grasp stress (or amount of squeeze) requires the minimization of the actuator torques τ_a. If the vector of all the actuator torques in the system is given by τ_a, then $\tfrac{1}{2}(\tau^{T}_{a}\tau_a)$ defines the actuator torque norm.

$$\begin{aligned} \Phi(f_{tip}) &= \tfrac{1}{2}(\tau^{T}_{a}\tau_a) \\ \tau_a &= \tau_u + J^{T}_{h} f_{tip} \\ \tau^{T}_{a} &= \tau^{T}_{u} + J_h f^{T}_{tip} \end{aligned} \tag{16}$$

Expanding (16) in terms of τ_a and τ^{T}_{a} gives:

$$\begin{aligned} \Phi(f_{tip}) &= \tfrac{1}{2}(\tau^{T}_{a}\tau_a) \\ &= \tfrac{1}{2}(\tau^{T}_{u} + J_h f^{T}_{tip})(\tau_u + J^{T}_{h} f_{tip}) \\ &= \tfrac{1}{2}(\tau^{T}_{u}\tau_u + \tau^{T}_{u} J^{T}_{h} f_{tip} + J_h f^{T}_{tip} J^{t}_{h} f_{tip} + J_h f^{T}_{tip}\tau_u) \\ &= \tau^{T}_{u} J^{T}_{h} f_{tip} + \tfrac{1}{2} f^{T}_{tip} J_h J^{T}_{h} f_{tip} \end{aligned} \tag{17}$$

This can be expressed in a general form as :

$$\Phi(f_{tip}) = C^T f_{tip} + \tfrac{1}{2} f^{T}_{tip} W f_{tip} \tag{18}$$

where C^T is given by

$$C^T = \begin{bmatrix} \tau^{T}_{1u} J^{T}_{1} f_{tip1} & \tau^{T}_{2u} J^{T}_{2} f_{tip2} & \tau^{T}_{3u} J^{T}_{3} f_{tip3} & \tau^{T}_{4u} J^{T}_{4} f_{tip4} \end{bmatrix}$$

and W is the multiplication of $J_h J^{T}_{h}$:

$$W = \begin{bmatrix} J_1 & 0 & 0 & 0 \\ 0 & J_2 & 0 & 0 \\ 0 & 0 & J_3 & 0 \\ 0 & 0 & 0 & J_4 \end{bmatrix} \times \begin{bmatrix} J^{T}_{1} & 0 & 0 & 0 \\ 0 & J^{T}_{2} & 0 & 0 \\ 0 & 0 & J^{T}_{3} & 0 \\ 0 & 0 & 0 & J^{T}_{4} \end{bmatrix} \tag{19}$$

Finally the $\tfrac{1}{2}(f^{T}_{tip} W f_{tip})$ term in (18) is re-expressed in terms of a weighting matrix depending on hand posture

$$\Phi(f_{tip}) = \tfrac{1}{2}\begin{bmatrix} f_{tip1} & f_{tip2} & f_{tip3} & f_{tip4} \end{bmatrix} \times \begin{bmatrix} J_1 J^{T}_{1} & 0 & 0 & 0 \\ 0 & J_2 J^{T}_{2} & 0 & 0 \\ 0 & 0 & J_3 J^{T}_{3} & 0 \\ 0 & 0 & 0 & J_4 J^{T}_{4} \end{bmatrix} \tag{20}$$

$$\begin{aligned} \Phi_1(f_{tip}) &= C^T f_{tip} \\ \Phi_2(f_{tip}) &= \tfrac{1}{2} f^{T}_{tip} W f_{tip} \\ \Phi(f_{tip}) &= \Phi_1(f_{tip} + \Phi_2(f_{tip})) \end{aligned} \tag{21}$$

IV OBJECT TO JOINT-SPACE MAPPING AND NONLINEAR FUNCTIONAL APPROXIMATION

Once a multi-finger hand controller receives sensory information for which the hand motion has to be made, it calculates a trajectory through the inverse kinematics (using hand Jacobian). An artificial neural network can be used for approximating a function from a set of available examples called learning samples or training patterns. The relation which we shall let the employed neural network to learn, Fig. (2), is defined in terms of some training patterns of object. Cartesian posture u_a^c, rate of change of object posture Δu_a^c, and rate of change of hand joints space $\Delta\Theta_{k-1}$ as given by (22) :

$$\Delta\Theta_k = f_{neural}\left(NN, \Delta u_a^c, u_a^c, \Delta\Theta_{k-1}\right)$$
$$\Delta\Theta_k = f_{neural}\left(\varphi_{1f}, \varphi_{2f}, \varphi_{3f}, \varphi_{4f}, w_{1ij}, w_{2ij}, w_{3ij}, w_{4ij}, \ldots, \Delta u_a^c, u_a^c, \Delta\Theta_{k-1}\right) \quad (22)$$

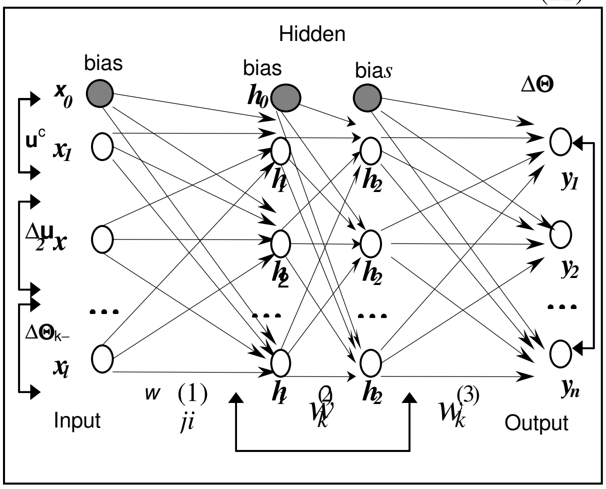

Fig. 2 . Supervised Neural Network Structure and Training

In (22), the computed change in hand joints space $\Delta\Theta_k$ is made a function of the neural network structural parameters in addition to the grasped object $\left(\varphi_{1f}, \varphi_{2f}, \varphi_{3f}, \varphi_{4f}, w_{1ij}, w_{2ij}, w_{3ij}, w_{4ij}, \ldots\right)$, motion parameters ($\Delta u_a^c$, u_a^c, $\Delta\Theta_{k-1}$). Hence the main purpose of the neural network structure is to approximate the nonlinear mapping between changes in hand joints to changes in the object position. Inputs and Output training data from the hand-object system have to be prepared and collected for training, as depicted in Fig. (2). Inputs to the neural network are: desired Cartesian object positions u_a^c, changes in such Cartesian positions Δu_a^c, one step change in position of the entire joints in radians $\Delta\Theta_{k-1}$. The outputs are the required changes in joint angles for the entire hand $\Delta\Theta_k$. The desired object posture values are obtained in advanced by moving the object to the required position.

V. Task-Space definition and Hand System Simulation

An object motion and path were defined, where it was a motion along different axes in a sinusoidal manner. Hand training patterns have been generated via letting the hand to follow some pre-defined Cartesian trajectory, while holding a grasped object of known physical dimensions. The task is to create 3-D *sinusoidal* object motion with no change in orientation at a given rate. robot hand has been simulated dynamically by Matlab-software, where such hand dynamic motion simulation is presented in Fig. (3). The network does consist of 18 inputs, 12 outputs and 50 hidden units. These nets map the 18 inputs characterizing the object.

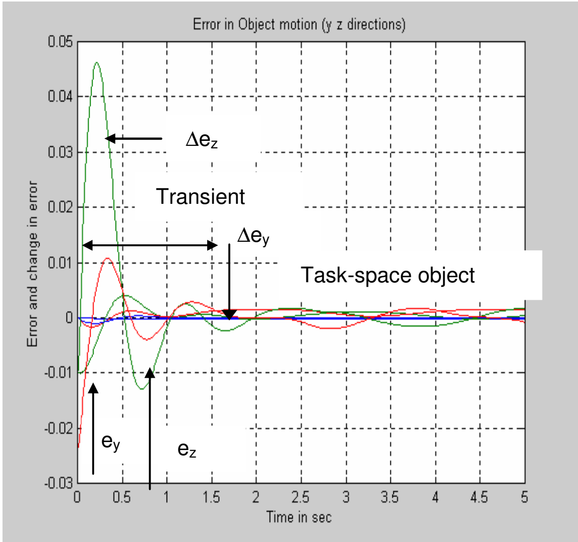

Fig. 3. Validating the neural hand controller.
Object Cartesian error and error change as ANN controller.

Cartesian position and hand joint positions into the 12 differential change in fingers positions, as can be seen in (2). In order to assess the proposed control algorithm, simulation of a constrained dynamics has been initially achieved using the kinematics and dynamic models of the CYBHAND [12]. At the beginning, the hand has been simulated with conventional inverse kinematics algorithms, where training patterns have been generated. Such training patterns have been based on object Cartesian motion and associated joints displacement. The hand has been run for large number of trials to produce as large as possible of training patterns.

The defined hand motion was described by moving an object center of gravity along the *y-axis* and the *z-axis* in a sinusoidal fashion. In this sense, the associated patterns Δu_a^c, u_a^c, $\Delta\Theta_{k-1}$ are tabulated in the proper format to be suitable for the neural network training. The quantity of the training pattern was reaching a size of (500) for a single variable (e.g. Δu_a^c). Hence to validate the neural network ability to model the hand inverse kinematics, the error between a typical neural output node (e.g. θ_{33}) with the actual one has been computed and analyzed. The learning network is *presented* with some object motion directions, where it compute the required hand joint-space used in hand controller. The execution process starts first with employing the trained **ANN** in the hand dynamic controller (*which mainly depends on the hand Jacobian inverse*). Once the object position and orientation have been defined, the **ANN**, hence computes the associated hand joint positions by presenting the network with some patterns which were not included during the training process. The **ANN** is employed in the hand controller for the calculation of joint displacement as required by the full controller already presented by (1). For instant, Fig. (3) shows the error associated with the object displacement.

Contrary to the method of internal forces optimisation, Fig. (4) shows the distributed **contact forces** based on the use of the nonlinear constraint optimisation of (15). With respect to the figures, the contact forces satisfy the *geometric*, *kinematics* and *frictional cone* constraints. Furthermore, it can be observed that the distributed contact forces are symmetrical around the middle of the path. The normal forces along the *z axis* direction are much greater than the two associated frictional forces by a factor determined by the type of material used at the fingertip-surface interaction. It is noticeable, how the employed optimisation technique has indeed computed the most suitable forces while satisfying all the stated constraints.

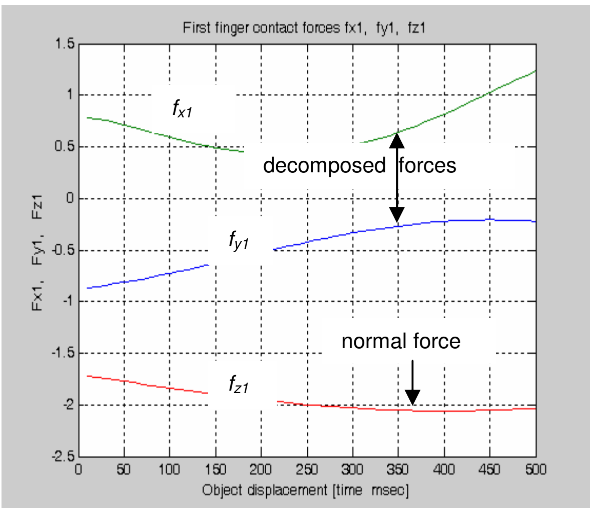

Fig. 4. Optimized and Computed hand grasping contact forces

VI. CONCLUSION

In conclusion, this article has presented a novel approach for computing both optimal set of fingertip forces distribution and an updating mechanism of the interrelated kinematics relation in a four fingers multi-finger robot hand system. The problem has been quadraticly formulated and structured in such a way as to compute fingertip forces at the contact frame. Secondly, for achieving a manipulation task, the issue of the inverse kinematics for multi-fingered robot hand has been computed where the object motion is defined in a Cartesian based system, hence the differential system Jacobian plays an important role. The nonlinear relation between a Cartesian object posture and associated hand joint-space settings, and control signals mapping were learned via four layers artificial neural networks trained for most possible object displacement.

REFERENCES

[1] S. C. Jacobsen, E. K. Iversen, R. T. Johnson and K. B. Biggers, “Design of the Utah/M.I.T. Dexterous Hand, “ *Proc. IEEE Conf. on Robotics and Automation, San* Francisco, CA (1986), pp. 1520-1532, 1986.

[2] J. Hollerhach and S. Narasimhan, “Finger force computation without the grip Jacobian. “ *In Prec. Of IEEE Conf. on Robotics and Automation*, San Francisco, pp. 871-875, 1986.

[3] N. Xi, J. Tran, & K. Bejczy, “ Intelligent Planning and Control for Multi-robot Coordination : An Event-Based Approach”, *IEEE Transactions on Robotics and Automation*, 12(3), pp. 439-445, 1996.

[4] H. Hashimoto, T. Kubota, M. Kuduo & F. Harashima, “Self-organizing Visual Servo System Based on Neural Networks,” *IEEE Control Systems Magazine*, pp. 31-36, 1992.

[5] A. Guez & Z. Ahmed, Solution to the Inverse Kinematics Problem in Robotics by Neural Networks. *Proceedings of the International Conference on Neural Networks*, USA, 1988.

[6] G. Schram, F. Linden, B. Krose & F. Groen, Visual Controller, *International Journal of Robotics and Autonomous Systems*, (18), pp. 293-299, 1996.

[7] A. Maciejewski & C. Klein, “The Singular Value Decomposition: Computation and Applications to Robotics,” *International Journal of Robotics Research*, vol. 8, no. 6, pp. 63-79, 1989.

[8] S. Lee, “Dual Redundant Arm Configuration Optimization with Task-Oriented Dual Manipulability, ” *IEEE Transactions in Robotics and Automation*, vol. 5, (1), 1989, pp. 78-97.

[9] G. Wohlke, “NEURO-FUZZY Based System Architecture for the Intelligent Control of Multi-Finger Robot hands”, *Proceedings of the IEEE International Conference on Fuzzy Systems*, ORLANDO, pp. 26-29, 1994.

[10] E. Al-Gallaf, “Neuro-Fuzzy Inverse Jacobian Mapping F or Multi-Finger Robot Hand Control, *International Journal of Intelligent & Robotic Systems*, vol. 18, no.1, 2003.