On the Transforming of Control Space by Manipulator Jacobian

Mohammad Mehdi Fateh and Hasan Farhangfard

Abstract: The transposed Jacobian is proposed to transform the control space from task space to joint space, in this paper. Instead of inverse Jacobian, the transposed Jacobian is preferred to avoid singularity problem, short real time calculations and its generality to apply for rectangular Jacobian. On-line Jacobian identification is proposed to cancel parametric errors produced by D-H parameters of manipulator. To identify Jacobian, the joint angles and the end-effector position are measured when tracking a desired trajectory in task space. Stability of control system is analyzed. The control system is simulated for position control of a two-link manipulator driven by permanent magnet dc motors. Simulation results are shown to compare the roles of inverse Jacobian and transposed Jacobian for transforming the control space.

Keywords: D-H parameters, manipulator Jacobian, task space, transforming of control space.

1. INTRODUCTION

Transforming of control space is necessary since a robot manipulator is controlled in task space while its actuators operate in joint space. The operations of inverse Jacobian and transposed Jacobian in vector spaces lead to apply them for transforming control space from task space to joint space. Manipulator Jacobian is derived from kinematic equations which transform the vector of generalized joint coordinates to the vector of generalized position end effector. The vector of generalized end effector velocities is transformed to the vector of generalized joint velocities by inverse Jacobian [1]. However, if Jacobian matrix is singular or rectangular, the inverse Jacobian is not obtained. The pseudo-inverse of Jacobian is an alternative solution for inverting a rectangular Jacobian [2]. The transposed Jacobian transforms the vector of generalized end effector forces to the vector of generalized joint forces.

In practice, industrial robots operate in task space while they are normally controlled in joint space. The actual joint positions are compared with the desired values to determine the errors. The control laws in position control are then applied to actuators for compensating the errors. In this manner, the actuators, the control laws, the desired trajectories and the outputs operate in joint space. This method is common to industrial manipulators. However, the main goal is to control the end effector in task space. Therefore, the desired trajectory is transformed from task space to joint space by inverse kinematic transformation, in advance. However, this transformation involves problems such as non-uniqueness, singularity, solving a set of nonlinear equations, and requiring exact kinematic parameters. The desired trajectory has planned such that the determinant of Jacobian matrix to be non zero all over the operating range as shown in Fig. 7. A control-observer scheme was proposed for controlling manipulator in task space [3]. This control approach is based on measuring joint variables and using forward kinematic solution instead of using inverse kinematic solution. The transposed Jacobian technique can be applied instead of inverse kinematic solution for generating a trajectory [4]. Moreover, a controller was designed based on Lyapunov stability using transposed Jacobian and forward kinematic equations for tracking a prescribed pass in task space [5]. Although in these approaches the problem of inverse kinematic has been solved, the forward kinematic model and manipulator Jacobian can not work perfectly in the case of parametric errors.

The joint space control method operates well on a high quality rigid manipulator with precise parameters in a specific area of workspace. On the other hand, the research activities have been extended in the control of robot manipulators on unknown environment with presence of parametric and unmodeled uncertainties. Therefore, a joint space control method may not work in task space perfectly without measuring the end effector position and direction. At the moment, measuring variables in task space is not as convenient as in joint space. The sensing technologies such as visual [6,7], lazar [8], resistive, inductive, capacitive,

Manuscript received September 28, 2006; revised April 11, 2007 and July 20, 2007; accepted October 25, 2007. Recommended by Editorial Board member Dong Hwan Kim under the direction of Editor Jae Bok Song. This is to acknowledge Shahrood University of Technology for their supports and encouragements.

Mohammad Mehdi Fateh and Hassan Farhangfard are with the Department of Electrical and Robotic Engineering, Shahrood University of Technology, Shahrood, Iran (e-mails: mmfateh@shahroodut.ac.ir, farhangfarde@yahoo.com).

and ultrasonic [9] technologies may be used for measuring variables in task space. The use of optical encoders is common and convenient for measuring joint variables. In this paper, a compound sensing strategy is proposed by measuring the required variables in both task space and joint space for control purposes.

Inverse Jacobian of manipulator has been frequently used in control laws for controlling manipulator in task space [6,10,11]. However, errors are produced in the control system using a nominal Jacobian matrix. The nominal values of geometrical parameters such as link lengths, twists, and off sets are used to form manipulator Jacobian. However, manufacturing tolerances may cause the actual parameters to deviate from the nominal values. An identification algorithm proposed in [12] can be used to determine the Jacobian parameters. The aforementioned algorithm estimates the actual D-H parameters. The manipulator Jacobian matrix becomes uncertain in the presence of parametric errors. A task-space adaptive Jacobian control method was proposed to overcome the uncertainties adaptively [13]. The visual information of task space was used as feedback signal.

Control approaches were proposed in task space for robot manipulators using estimated Jacobian [14-17]. A PD control law based on approximate Jacobian was provided for set point control of robots with uncertain Jacobian matrix. Required conditions for the bound of the estimated Jacobian matrix and stability conditions with feedback gains were considered. Despite the fact that the Jacobian is not known, the asymptotic stability can be satisfied for the control system [17].

Identification methods have found many applications in control systems such as parameter identification, dynamic identification and system identification. In order to work in the joint control space, an identified dynamic model of manipulator can be used. Parameter identification of a manipulator is proposed to determine the dynamic model [18]. Three sets of special tests are proposed for identifying coefficients of dynamic equation derived from Lagrangian formulation [19]. The tests are static, constant angular velocity motion, and accelerated motion.

In Section 2, the manipulator Jacobian is formulated and in Section 3 is identified by the use of Jacobian formula and sensing required variables. In Section 4, the inverse Jacobian and transposed Jacobian for transforming control space from task space to joint space are analyzed and compared in the control system. Advantages and drawbacks of the control approaches are investigated. The control system stability is then analyzed in Section 5 to consider stabilization problem and tracking problem. Finally, the control system is simulated for position control of the two-link manipulator system and simulation results are then considered to improve the system performance.

2. MANIPULATOR JACOBIAN

Manipulator Jacobian or Jacobian matrix is an important subject for robot control and analysis. Jacobian matrix is used for planning smooth trajectory, determining the singular cases, and transforming the control space. Parametric errors will cause Jacobian error, and then Jacobian matrix causes velocity error in the Cartesian space. Jacobian matrix is derived using forward kinematic equations [20] as follows

$$T = \mathrm{Fkin}(\mathbf{q}), \tag{1}$$

where $\mathbf{q} \in R^n$ is the vector of generalized joint coordinates, Fkin is the forward kinematic function, T is a transformation matrix defined as $T_0^i = \begin{bmatrix} R_0^n & \mathbf{d}_0^n \\ \mathbf{0} & 1 \end{bmatrix}$, R_0^n is a rotation matrix to show the direction of the end-effector and $\mathbf{d}_0^n$ is the position of the end-effector, n is number of end effector frame, and the base is numbered by 0. Jacobian matrix is defined by

$$\begin{bmatrix} \mathbf{v}_0^n \\ \boldsymbol{\omega}_0^n \end{bmatrix} = \begin{bmatrix} J_v \\ J_w \end{bmatrix} \dot{\mathbf{q}}, \tag{2}$$

where $\mathbf{v}_0^n$ and $\boldsymbol{\omega}_0^n$ are the vector of linear and angular velocities of the end-effector, respectively, J_v and J_ω are $3 \times n$ matrixes, respectively. Manipulator Jacobian is introduced to be

$$J = \begin{bmatrix} J_v \\ J_w \end{bmatrix}. \tag{3}$$

For revolute joint i, the i-th column of Jacobian matrix is given by

$$\mathbf{J}_i = \begin{bmatrix} \mathbf{z_{i-1}} \times (\mathbf{o_n} - \mathbf{o_{i-1}}) \\ \mathbf{z_{i-1}} \end{bmatrix}, \tag{4}$$

where based on the D-H representation, $\mathbf{z_{i-1}}$ is the z axis of frame i-1, $\mathbf{o_n}$ is the origin of the frame n, and $\mathbf{o_{i-1}}$ is the origin of frame i-1. For prismatic joint i, the *i-th* column of Jacobian matrix is

$$\mathbf{J}_i = \begin{bmatrix} \mathbf{z_{i-1}} \\ \mathbf{0} \end{bmatrix}. \tag{5}$$

In this paper, we consider to track the end-effector position in the work space, so J_v of Jacobian matrix is simply denoted by J. The kinematic equation for a

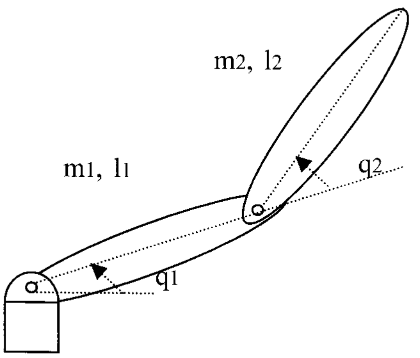


Fig. 1. The two-link manipulator.

two-link manipulator shown in Fig. 1, is formulated as

$$X_1 = L_1 \cos q_1 + L_2 \cos(q_1 + q_2),$$
$$X_2 = L_1 \sin q_1 + L_2 \sin(q_1 + q_2), \tag{6}$$

where $L = [L_1\ L_2]^T$ is length of links, $\mathbf{q} = [q_1\ q_2\ ]^T$ is joint variable vector and $\mathbf{X} = [X_1\ X_2]^T$ is position vector of the end point. Jacobian matrix is derived

$$J = \begin{bmatrix} -L_1 \sin q_1 - L_2 \sin(q_1 + q_2) & -L_2 \sin(q_1 + q_2) \\ L_1 \cos q_1 + L_2 \cos(q_1 + q_2) & L_2 \cos(q_1 + q_2) \end{bmatrix}. \tag{7}$$

It is shown that the Jacobian matrix is a function of joint variables and it comprises kinematic parameters. The end point velocity $\dot{\mathbf{X}}$ is given by

$$\dot{\mathbf{X}} = J\dot{\mathbf{q}}. \tag{8}$$

3. JACOBIAN IDENTIFICATION

The nominal values of geometrical parameters such as link lengths, twists, and off sets are used to form manipulator Jacobian. However, manufacturing tolerances may cause the actual parameters to deviate from the nominal values. Therefore, we should identify the actual values of D-H parameters to determine manipulator Jacobian. The first step to identify the Jacobian is recognition of the kinematic parameters and joint variables by considering (7). To calculate the parameters, (6) is rewritten as

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} \cos q_1 & \cos(q_1 + q_2) \\ \sin q_1 & \sin(q_1 + q_2) \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \end{bmatrix}, \tag{9}$$

Thus parameters by

$$\begin{bmatrix} L_1 \\ L_2 \end{bmatrix} = \begin{bmatrix} \cos q_1 & \cos(q_1 + q_2) \\ \sin q_1 & \sin(q_1 + q_2) \end{bmatrix}^{-1} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}. \tag{10}$$

In this study the manipulator Jacobian includes only the link length of D-H parameters named L_1 and L_2. To identify the parameters, it is required to measure the joint angles and the end point position shown in (10). The joint angles are measured by optical encoders and the position of the end effector is measured by a camera such that task space is defined as image plane and camera is fixed perpendicular to the image plane [7]. Calculations of parameters by (10) are failed in the case of singularity problem. Equation (10) is singular in where

$$\det \begin{bmatrix} \cos q_1 & \cos(q_1 + q_2) \\ \sin q_1 & \sin(q_1 + q_2) \end{bmatrix} = 0. \tag{11}$$

This leads to

$$\cos q_1 \sin(q_1 + q_2) - \sin q_1 \cos(q_1 + q_2) = 0. \tag{12}$$

Singularity problem will appear in where the determinant of Jacobian matrix is zero, as well. The desired trajectory is planed such as the determinant of Jacobian matrix is not zero, at all. From (7), the determinant of Jacobian matrix is given by

$$\det(J) = L_1 L_2 (\cos q_1 \sin(q_1 + q_2) - \sin q_1 \cos(q_1 + q_2)). \tag{13}$$

As a result of comparing (12) and (13), we can conclude that considering $\det(J) = 0$ is a sufficient condition to check the singularity problem in this control approach.

4. TRANSFORMING OF CONTROL SPACE

Equation (8) leads to

$$\delta\mathbf{X} = J\delta\mathbf{q}, \tag{14}$$

$$\delta\mathbf{q} = J^{-1}\delta\mathbf{X}, \tag{15}$$

where $\delta\mathbf{q}$ is a difference angle vector in joint space, and $\delta\mathbf{X}$ is a difference position vector in task space. Consequently, the Jacobian matrix transforms the data from joint space to task space. It is assumed that $\boldsymbol{\delta}\mathbf{q} = \mathbf{q_d} - \mathbf{q}$ and $\delta\mathbf{X} = \mathbf{X_d} - \mathbf{X}$ where $\mathbf{q_d}$ is the desired joint vector and $\mathbf{X_d}$ is the desired position vector. A PID control law is proposed for compensating the error in task space as

$$K_D\dot{\mathbf{e}} + K_P\mathbf{e} + K_I \int \mathbf{e}dt = \mathbf{v}, \tag{16}$$

where $\mathbf{e} = \mathbf{X_d} - \mathbf{X}$ is the vector of errors in task space, $\mathbf{v}$ is the control law in task space, K_D, K_p, and K_I are the derivative, the proportional and integrative coefficient diagonal matrixes, respectively. Giving appropriate units to coefficient diagonal matrixes K_D, K_P, and K_I, results in $\mathbf{v}$ as a position vector. The transformation given by (8) obtains

$$\mathbf{v} = J\mathbf{u}, \quad (17)$$

where $\mathbf{u}$ is the control law in joint space as a position vector. Substituting (17) into (16) results

$$J^{-1}(K_D\dot{\mathbf{e}} + K_P\mathbf{e} + K_I \int \mathbf{e}dt) = \mathbf{u}. \quad (18)$$

Actually, the inverse Jacobian in (18) transforms the control law from task space to joint space. The control inputs of motors are then calculated based on (18) and motors dynamics. The coefficient matrixes K_D, K_P, and K_I are then regulated to improve the performance of control system. In this control strategy, the end point position and the joint angles are measured. The Jacobian matrix is then identified and the inverse Jacobian is calculated. Fig. 2 shows the control scheme using inverse Jacobian transformation. The main problem of the above control strategy is singularity. When the determinant of Jacobian is zero, Jacobian matrix becomes singular. The system is not under the control in the case of singularity and solution is failed. Although we can choose a non-singular trajectory, tracking error may take the actual values in the area close to singularity. In addition, the inverse Jacobian is not defined for rectangular Jacobian. We propose transposed Jacobian to transform the control space. Transposed Jacobian can transform a force vector in task space to a torque vector in joint space as follows [20]

$$\boldsymbol{\tau} = J^T\mathbf{f}, \quad (19)$$

where $\boldsymbol{\tau}$ is the vector of generalized torques applied on the joints, $\boldsymbol{f}$ is the vector of generalized forces applied on the end effector and J^T is the transposed Jacobian. We prefer J^T for avoiding singularity, short real time calculations and to be applied on rectangular Jacobian. Using the transformation given by (19) yields

$$\mathbf{u} = J^T\mathbf{v}, \quad (20)$$

where $\mathbf{u}$ is the control law in joint space as a torque vector. In order to satisfy units, we can give appropriate units to coefficient matrixes in (16) to provide $\mathbf{v}$ as a force vector. Consequently, the coefficient matrixes in the control law have not the same units as before. This obtains

$$\hat{K}_D\dot{\mathbf{e}} + \hat{K}_P\mathbf{e} + \hat{K}_I \int \mathbf{e}dt = \mathbf{v}, \quad (21)$$

where $\hat{K}_D$, $\hat{K}_P$, and $\hat{K}_I$ are the derivative, the proportional and integrative coefficient diagonal matrixes, respectively. Substituting (21) into (20) leads to the following control law

$$J^T(K_D\dot{\mathbf{e}} + K_P\mathbf{e} + K_I \int \mathbf{e}dt) = \mathbf{u}. \quad (22)$$

The control inputs of motors are then calculated based on (22) and motors dynamics. The coefficient matrixes are then regulated to improve the performance of control system. Fig. 3 shows the control system using Jacobian transformation (21).

The PID controller is used just as one kind of position controllers to show the role of Jacobian in the control system. However, methods based on feedback linearization methods can be selected to cancel the nonlinearities in the dynamic model of the robot which are known as gravitational, centrifugal and Coriolis torques. This requires parameter identification to find the dynamic model. As usual, a gravity model of manipulator is provided by producers which can be used to cancel the gravitational torque. By the use of gravity model, a control law is then written

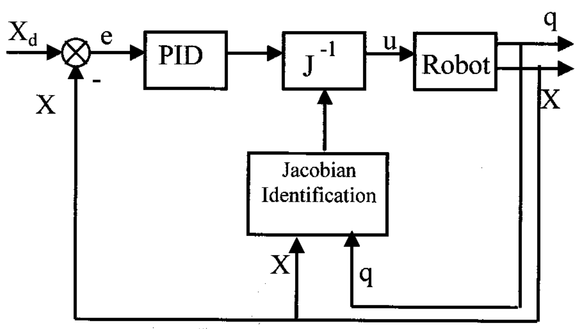


Fig. 2. The control system using inverse Jacobian.

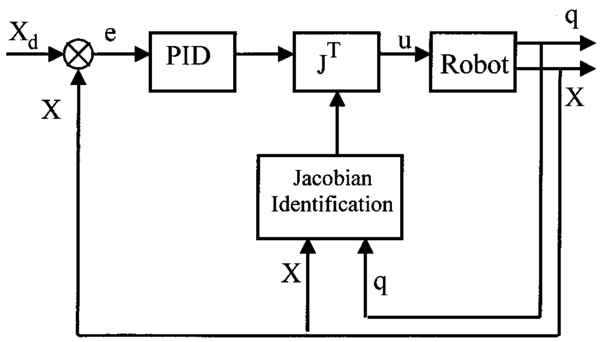


Fig. 3. The control system using transposed Jacobian.

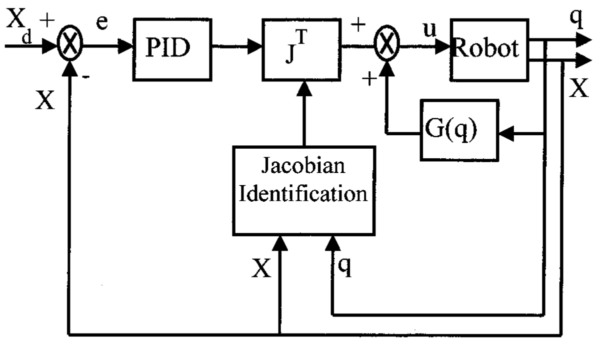


Fig. 4. The control system.

$$J^T(\hat{K}_D\dot{\mathbf{e}} + \hat{K}_P\mathbf{e} + \hat{K}_I\int\mathbf{e}dt) + \mathbf{g(q)} = \mathbf{u},$$

where $\mathbf{g(q)}$ is the gravitational torque. The proposed control system is shown in Fig. 4.

5. STABILITY ANALYSIS

We can design a perfect tracking control system if equilibrium point is asymptotically stable. The equilibrium point of closed loop control system shown in Fig. 3 is given by

$$\mathbf{e} = \mathbf{0}, \text{ and } \dot{\mathbf{e}} = \mathbf{0}, \tag{23}$$

where $\mathbf{x} = [e \quad \dot{e}]^T$ is the state vector. Once system locates in the equilibrium point, it remains there for all future time. By some calculations from (21), we have

$$\hat{K}_D\ddot{\mathbf{e}} + \hat{K}_P\dot{\mathbf{e}} + \hat{K}_I\mathbf{e} = J^{-T}\dot{\mathbf{u}} + \dot{J}^{-T}\mathbf{u}. \tag{24}$$

A Lyapunov function candidate is proposed as

$$V(\mathbf{x}) = \frac{1}{2}\dot{\mathbf{e}}^{\mathbf{T}}\hat{K}_D\dot{\mathbf{e}} + \frac{1}{2}\mathbf{e}^{\mathbf{T}}\hat{K}_I\mathbf{e}, \tag{25}$$

where V($\mathbf{x}$) is a scalar function with continuous first order derivatives and is positive definite. $\dot{V}(\mathbf{x})$ becomes

$$\dot{V}(\mathbf{x}) = \dot{\mathbf{e}}^{\mathbf{T}}\hat{K}_D\ddot{\mathbf{e}} + \dot{\mathbf{e}}^{\mathbf{T}}\hat{K}_I\mathbf{e}. \tag{26}$$

From (24) and (26), we have

$$\begin{aligned}\dot{V}(\mathbf{x}) &= \dot{\mathbf{e}}^{\mathbf{T}}(-\hat{K}_P\dot{\mathbf{e}} - \hat{K}_I\mathbf{e} + J^{-T}\dot{\mathbf{u}}) + \dot{\mathbf{e}}^{\mathbf{T}}\hat{K}_I\mathbf{e} \\ &= -\dot{\mathbf{e}}^{\mathbf{T}}\hat{K}_P\dot{\mathbf{e}} + J^{-T}\dot{\mathbf{u}} + \dot{J}^{-T}\mathbf{u}.\end{aligned} \tag{27}$$

The equilibrium point $\mathbf{0}$ is stable if $\dot{V}(\mathbf{x})$ is negative semi-definite. It means that

$$J^{-T}\dot{\mathbf{u}} + \dot{J}^{-T}\mathbf{u} \le \dot{\mathbf{e}}^{\mathbf{T}}\hat{K}_P\dot{\mathbf{e}}. \tag{28}$$

The equilibrium point $\mathbf{0}$ is asymptotically stable, if

$$J^{-T}\dot{\mathbf{u}} + \dot{J}^{-T}\mathbf{u} < \dot{\mathbf{e}}^{\mathbf{T}}\hat{K}_P\dot{\mathbf{e}}. \tag{29}$$

Dynamic equations of manipulator provide that $\mathbf{u}$ and J^{-T} are functions as $\mathbf{u} = f(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ and $J^{-T}(\mathbf{q})$, respectively. So, it is required that the desired trajectory must be bounded and smooth such that $\dot{\mathbf{u}}$ and $\dot{J}^{-T}$ satisfy (29). Matrix $\hat{K}_P$ should be large enough, as well. We can choose the diagonal coefficient matrixes $\hat{\mathbf{K}}_\mathbf{P}$, $\hat{\mathbf{K}}_\mathbf{I}$ and $\hat{\mathbf{K}}_\mathbf{I}$ such that local stability is provided. At equilibrium point, the error $\mathbf{e}$ and its derivative $\dot{\mathbf{e}}$ are zero. As a result of substituting equilibrium point into (21), $\mathbf{u}$ is required to be zero or a constant vector in equilibrium point. Equation (28) obtains

$$\dot{\mathbf{J}}^{-\mathbf{T}}\mathbf{u} \le 0. \tag{30}$$

For stabilization problem, vector $\mathbf{q}$ is constant at equilibrium point. Thus $\dot{J}^{-T} = 0$ and (30) will be satisfied. We can conclude that position error at stabilization design will be zero. In tracking design, vector $\mathbf{q}$ is not constant. Therefore at the equilibrium point it is required that

$$J^{-T}\dot{\mathbf{u}} + \dot{J}^{-T}\mathbf{u} \le 0, \tag{31}$$

$$\frac{d}{dt}(J^{-T}\mathbf{u}) \le 0. \tag{32}$$

There is no guaranty to satisfy (31). So, tracking error will not be zero. Therefore, a control method based on feedback linearization is proposed to reduce the nonlinearity as shown already in Fig. 4. This results in an approximate linear system and the desired tracking error.

6. SIMULATION

The two-link manipulator used in this simulation has specifications of m_1 is 2kg, m_2 is 1kg, l_1 is 1m and l_2 is 5m driven by permanent magnet DC motors. The components of desired trajectory in task space, X_1 and X_2 are shown in Figs. 5 and 6, respectively. The desired trajectory has planned such that the determinant of Jacobian matrix to be non zero all over the operating range as shown in Fig. 7. The control approach presented in Fig. 2 is simulated by choosing,

$$K_P = \begin{bmatrix}1000 & 0\\ 0 & 1000\end{bmatrix},\ K_I = \begin{bmatrix}4000 & 0\\ 0 & 4000\end{bmatrix},\ K_D = \begin{bmatrix}100 & 0\\ 0 & 100\end{bmatrix}.$$

Tracking error increases to about 0.7 m when starting but it decreases after 1sec in task space as shown in Fig. 8. The control system can reduce the error. However, tracking error is too high when starting. We try again with other values for coefficient matrixes by selecting,

$$K_P = \begin{bmatrix}2000 & 0\\ 0 & 2000\end{bmatrix},\ K_I = \begin{bmatrix}10000 & 0\\ 0 & 10000\end{bmatrix},\ K_D = \begin{bmatrix}100 & 0\\ 0 & 100\end{bmatrix}.$$

Tracking error increases to about 0.01m when starting and it goes down very soon as shown in Fig. 9. Tracking error is reduced if comparing Fig. 9. with Fig. 8. As a result, we can conclude that the system performance can be improved by selecting suitable coefficients for the PID controller. Now, the role of transposed Jacobian is considered in control approach presented in Fig. 3. The control system is simulated by choosing the same coefficients as before

$$K_P = \begin{bmatrix} 2000 & 0 \\ 0 & 2000 \end{bmatrix}, K_I = \begin{bmatrix} 10000 & 0 \\ 0 & 10000 \end{bmatrix}, K_D = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}.$$

Tracking error in task space is presented in Fig. 10. It increases to about 0.002m in the first oscillation but oscillations approaches zero very soon. Since the determinant of Jacobian as shown in Fig. 7 has the lowest value at starting, it is close to singularity. This causes much more error in calculating inverse Jacobian. Moreover, because of time constant of motor response, the errors are relatively high in both approaches when starting. Tracking error obtained by transposed Jacobian approach is less than inverse Jacobian approach in the most of operating range, particularly in the initial part as compared in Fig. 10. The tracking error in inverse Jacobian approach is high close to the singular area. This shows an

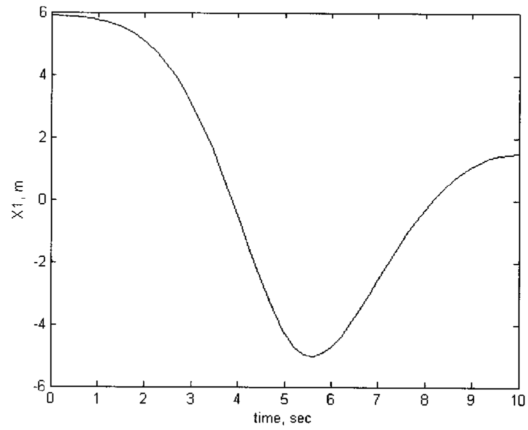


Fig. 5. The desired position of X_1.

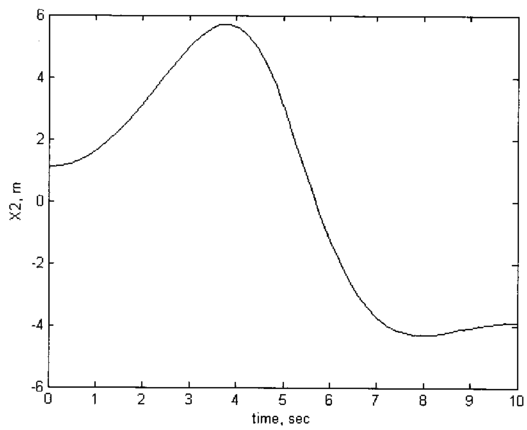


Fig. 6. The desired position of X_2.

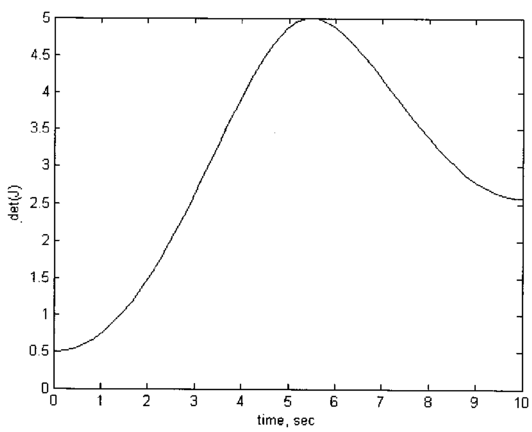


Fig. 7. Determinant of manipulator Jacobian.

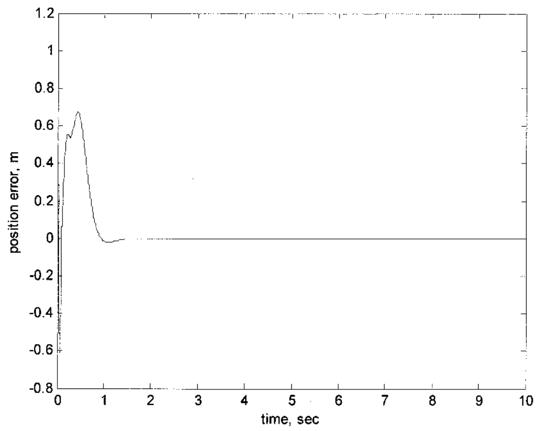


Fig. 8. Tracking error.

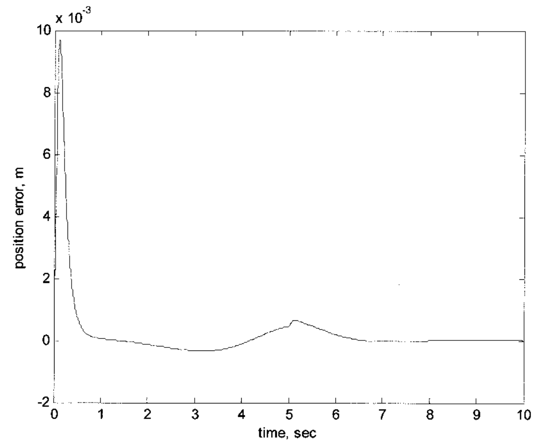


Fig. 9. Tracking error using inverse Jacobian.

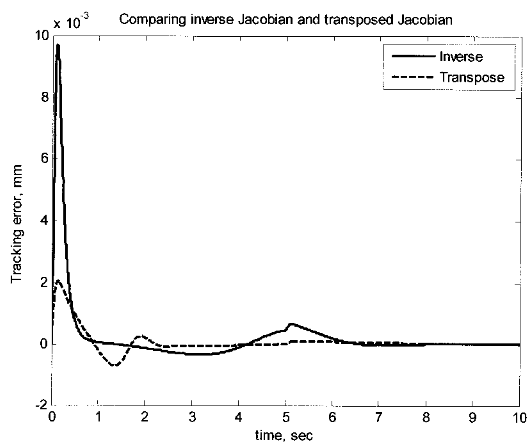


Fig. 10. A comparison on tracking error.

advantage of transpose Jacobian which works well in the singular area. In addition, the time of calculating the transpose Jacobian is much less than calculating the inverse Jacobian.

7. CONCLUSIONS

In this paper, the roles of inverse Jacobian and transposed Jacobian of manipulator have been investigated and analyzed to transform vectors from task space to joint space.

We prefer the transposed Jacobian to avoid singularity, short real time calculations and its generality as it can be applied to rectangular Jacobian. This transformation is very useful and efficient to control the robot manipulators in task space. The manipulator Jacobian includes the nominal values of geometrical parameters such as link lengths, twists, and offsets. However, manufacturing tolerances may cause the actual parameters to deviate from the nominal values. Identification of manipulator Jacobian was proposed to cancel parametric errors in nominal Jacobian. Jacobian matrix was identified by sensing the end effector position using visual technology and measuring joint angles by optical encoders.

Stability of control system using transposed Jacobian has been analyzed. It can be concluded that steady state position error at stabilization design will be zero. However, tracking error will not be zero.

The control systems of a two-link manipulator driven by permanent magnet dc motors were simulated. Simulation results were shown to compare the roles of transposed Jacobian and inverse Jacobian. Tracking errors are small enough all over the operating range using Transpose Jacobian. The feedback linearization method for compensating the gravitational torque vector was used for reducing tracking errors.

REFERENCES

[1] M. W. Spong, *Motion Control of Robot Manipulators, in Handbook of Control*, CRC Press, pp. 1339-1350, 1996.

[2] K. L. Doty, C. Melchiorri, and C. Bonivento, "A theory of generalized inverse applied to Robotics," *International Journal of Robotics Research*, vol. 12, no. 1, pp. 1-19, 1995.

[3] M. A. Arteaga, A. C. Sanchez, and V. P. Vega, "Cartesian control of robots without dynamic model and observer design," *Automatica*, vol. 42, pp. 473-480, 2006.

[4] G. Antonelli, S. Chiaverini, and G. Fusco, "Real-time end-effector path following for robot manipulators subject to velocity, acceleration, and jerk joint limits," *Proc. of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 452-457, 2001.

[5] M. Galicki, "Path-constrained control of a redundant manipulator in a task space," *Robotics and Autonomous Systems*, vol. 54, pp. 234-243, 2006.

[6] D. Xiao, M. Song, B. K. Ghosh, N. Xi, T. J. Tarn, and Z. Yu, "Real-time integration of sensing, planning and control in robotic work-cells," *Control Engineering Practice*, vol. 12, pp. 653-663, 2004.

[7] G. Hutchinson, G. D. Hager, and P. Corke, "A tutorial on visual servo control," *IEEE Trans. on Robotics and Automation*, vol. 12, no. 5, pp. 651-670, 1996.

[8] G. Alici and B. Shirinzadeh, "A systematic technique to estimate positioning errors for robot accuracy improvement using laser interferometry based sensing," *Mechanism and Machine Theory*, vol. 40, pp. 879-906, 2005.

[9] M. Jarviluoma and T. Heikkila, "An ultrasonic multi-sensor system for object loading applications," *Mechatronics*, vol. 5, no. 4, pp. 433-440, 1995.

[10] T. J. Tarn, M. Song, N. Xi, and B. K. Ghosh, "Multi sensor fusion scheme for calibration-free stereo vision in a manufacturing work-cell," *Proc. of IEEE/SICE/RSJ Multi Sensor Fusion and Integration for Intelligent Systems*, Washington DC, USA, pp. 416-423, 1996.

[11] D. Xiao, B. K. Ghosh, N. Xi, and T. J. Tarn, "Intelligent robotic manipulation with hybrid position/force control in an uncalibrated workspace," *Proc. of IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1671-1676, Belgium, 1998.

[12] C. R. Mirman and K. C. Gupta, "Identification of position-independent robot parameter errors using special Jacobian matrices," *International Journal of Robotics Research*, vol. 12, no. 3, pp. 288-298, 1993.

[13] C. Liu, C. C. Cheah, and J. J. E. Slotine, "Adaptive Jacobian tracking control of rigid-link electrically driven robots based on visual task-space information," *Automatica*, vol. 42, pp. 1491-1501, 2006.

[14] H. Y. Cheah, S. Han, S. Kawamura, and S. Arimoto, "Grasping and position control for multi-fingered robot hands with uncertain jacobin matrices," *Proc. of the American Control Conference Arlington*, pp. 2403-2408, June 2001.

[15] H. Yazarel and C. C. Cheah, "Adaptive visual servoing of robots with uncertain gravity regressor and Jacobian matrices," *IEEE Trans. on Automatic Control*, vol. 46, no. 8, pp. 652-657, 2001.

[16] C. C. Cheah, S. Kawamura, S. Arimoto, and K. Lee, "H_∞ tuning for task-space feedback control of robot with uncertain Jacobin matrix," *IEEE*

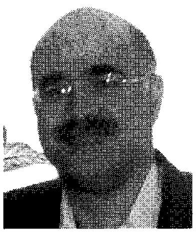
Trans. on Automatic Control, vol. 46, no. 8, pp. 1313-1318, 2001.

[17] C. C. Cheah, J. J. E. Slotine, and C. Liu, “Approximate Jacobian adaptive control for robot manipulators,” *Proc. of IEEE International Conference on Robotics & Automation*, New Orleans, LA, pp. 3075-3080, April 2004.

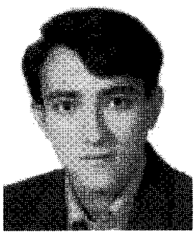
[18] Z. Alexei and H. Sandor, “Parameter identification of a robot arm using genetic algorithms,” *Periodica Polytechnic Ser. El. Eng.*, vol. 45, no. 3-4, pp. 195-209, 2001.

[19] H. Mayeda, K. Osuka, and A. Kangawa, “A new identification method for serial manipulator arms,” *Proc. of the IFAC World Congress*, pp. 74-79, 1984.

[20] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*, John Wiley & Sons, New York, 1989.



Mohammad Mehdi Fateh received the Ph.D. degree in Robotic Engineering from Southampton University, U.K. in 2001. He is with the Department of Electrical and Robotic Engineering at Shahrood University of Technology in Iran. His research interests include robot control and modelling.



Hasan Farhangfard received the M.Sc. degree in Control Engineering from Shahrood University of Technology, Iran in 2006. His research interest is robot control.