2007 International Symposium on Nonlinear Theory and its Applications
NOLTA'07, Vancouver, Canada, September 16-19, 2007

Control Dynamics for Redundant Manipulator in Three-Dimensional Space

Masanori Kano†, Hisashi Aomori† and Mamoru Tanaka†

†Sophia University
Department of Electrical and Electronics Engineering
7-1 Kioi-cho, Chiyoda-ku, Tokyo, 102-8554, Japan
Phone:+81-3-3238-3878, FAX:+81-3-3238-3321
E-mail: kanou-m@sophia.ac.jp

Abstract—In this paper, we propose a method of decreasing a computational cost that is needed to control a redundant manipulator in three-dimensional space. Especially, in case of an unknown nonlinear system of the manipulator, the computational cost becomes high for deriving an approximated Jacobian matrix. We use controllable dumping coefficient to decrease computational cost. Also, we propose the method of deriving the approximated Jacobian matrix for unknown nonlinear system.

1. Introduction

Recently, the development of a robot that can adjust to changes in the environment as a next-generation robot has received research attention. The manipulator of the robot should be redundant since flexibility and speed are especially demanded in such a robot in adjusting to a change in the environment. It is necessary to provide redundancy for the manipulator. A redundant manipulator is excellent in operationality and generality compared with the manipulator without redundancy since the redundant manipulator can evade a singular point and an obstacle using redundancy [1]. When a joint angle is calculated from the end-point position of the manipulator that has a redundant degree of freedom, we must solve an inverse problem of kinematics. The inverse problem involves deriving an input from an output. A generalized inverse matrix is used to solve it [2]. However, the cost for deriving the generalized inverse matrix is high. Therefore, we cannot use it in real-time control.

In this research, we propose a method of decreasing a computational cost that is needed to control a redundant manipulator in three-dimensional space. Additionally, we proposed the method of deriving the approximated Jacobian matrix for unknown nonlinear system. [3].

1.1. Inverse Kinematics Problem

We consider the control of the end-point position of a manipulator that has n degrees of freedom. Let θ_i be a joint angle, then the relation of θ_i and a vector $\mathbf{\Theta}$ is defined as

$$\mathbf{\Theta} = [\theta_1,\ \theta_2,\ \dots,\ \theta_n]^T \in R^n, \tag{1}$$

where $\mathbf{A}^T$ denotes the transposed matrix of $\mathbf{A}$. Also, a vector of the end-point position of the manipulator is defined as

$$\mathbf{r} = [r_1,\ r_2,\ \dots,\ r_m]^T \in R^m. \tag{2}$$

Then, the geometrical relation of $\mathbf{\Theta}$ and $\mathbf{r}$ is given by

$$\mathbf{r} = \mathbf{g}(\mathbf{\Theta}), \tag{3}$$

where $\mathbf{g}$ is a nonlinear function that depends on the form of the manipulator. Generally, when the joint angle vector $\mathbf{\Theta}$ is given, the end-point position $\mathbf{r}$, which varies with $\mathbf{\Theta}$, is uniquely decided. The computational cost is low. On the other hand, if only $\mathbf{r}$ is given, deriving $\mathbf{\Theta}$ is an inverse problem described by

$$\mathbf{\Theta} = \mathbf{g}^{-1}(\mathbf{r}). \tag{4}$$

It is difficult to solve (4). The control system is shown in Fig. 1, where $\mathbf{r}_d$ is the target point.

When a rotational joint type manipulator is controlled, a nonlinear function of the control system contains trigonometric funcitons [4]. The manipulator can be controlled by solving the inverse problem.

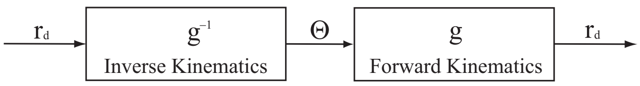

Figure 1: Control system

2. Solving the Inverse Problem

2.1. Newton Method

We use the Newton method to solve the inverse problem. The target position of the end-point position of the manipulator is defined as $\mathbf{r_d} = [r_{1,d},\ r_{2,d},\ \dots,\ r_{m,d}]^T \in R^m$. A nonlinear function is defined as

$$\mathbf{f}(\mathbf{\Theta}) = \mathbf{r_d} - \mathbf{g}(\mathbf{\Theta}) = 0. \tag{5}$$

The Newton method is used for solving the nonlinear function given by (5). Then, (5) is transformed into

$$\mathbf{f}(\mathbf{\Theta}^{k+1}) = \mathbf{f}(\mathbf{\Theta}^k) + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{\Theta}}\right|_{\mathbf{\Theta}=\mathbf{\Theta}_k} (\mathbf{\Theta}^{k+1} - \mathbf{\Theta}^k) = 0. \tag{6}$$

Using a Jacobian matrix $\mathbf{J}$, (6) is rewritten as

$$\mathbf{f}(\mathbf{\Theta}^k) + \mathbf{J}(\mathbf{\Theta}^k)(\mathbf{\Theta}^{k+1} - \mathbf{\Theta}^k) = 0, \tag{7}$$

$$\mathbf{J} = [\frac{\partial \mathbf{f}}{\partial \mathbf{\Theta}}] \in R^{m\times n}. \tag{8}$$

We obtain the following iterative equation by solving for $\mathbf{\Theta}^{k+1}$ in (7).

$$\mathbf{\Theta}^{k+1} = \mathbf{\Theta}^k - h\mathbf{J}^{-1}(\mathbf{\Theta}^k)\mathbf{f}(\mathbf{\Theta}^k), \tag{9}$$

where h is a dumping coefficient that determines the update frequency of $\mathbf{\Theta}$. $\mathbf{\Theta}$ is updated until (9) is satisfied with $\mathbf{r_d} = \mathbf{g}(\mathbf{\Theta}^{k+1})$. Then, the end-point position of the manipulator reaches the target position. This method can obtain the solution by repeating a simple numerical computation if the number of dimensions of the joint space is equal to that of the workspace. However, the number of dimensions of the joint space is larger than that of the workspace in a conventional manipulator. Therefore, if $m < n$ is satisfied, the Jacobian matrix becomes a rectangular matrix. To solve the inverse problem with a rectangular Jacobian matrix, a generalized inverse matrix should be used.

2.2. Generalized Inverse Matrix

A Jacobian matrix is defined as

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_1} & \frac{\partial f_1}{\partial \theta_2} & \cdots & \frac{\partial f_1}{\partial \theta_n} \\ \frac{\partial f_2}{\partial \theta_1} & \frac{\partial f_2}{\partial \theta_2} & \cdots & \frac{\partial f_2}{\partial \theta_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_m}{\partial \theta_1} & \frac{\partial f_m}{\partial \theta_2} & \cdots & \frac{\partial f_m}{\partial \theta_n} \end{bmatrix}, \tag{10}$$

where n is the input dimension and m is the output dimension. In the case of the same dimensions of I/O($m = n$), the Jacobian matrix becomes a square matrix. When the matrix is regular, its inverse matrix exists. However, the Jacobian matrix of a conventional manipulator is not square($n > m$). Hence, its inverse matrix does not exist in a conventional manipulator. It is impossible to solve such a nonlinear equation as (5) using (9). In this case, a generalized inverse matrix is used for solving the problem. Here, we consider an arbitrary matrix $\mathbf{A}$. Its generalized inverse matrix $\mathbf{A}^+$ is satisfied by following equation;

$$\begin{aligned} \mathbf{A}\mathbf{A}^+\mathbf{A} &= \mathbf{A}, \\ \mathbf{A}^+\mathbf{A}\mathbf{A}^+ &= \mathbf{A}^+, \\ (\mathbf{A}\mathbf{A}^+)^T &= \mathbf{A}\mathbf{A}^+, \\ (\mathbf{A}^+\mathbf{A})^T &= \mathbf{A}^+\mathbf{A}. \end{aligned} \tag{11}$$

The most common type of the generalized inverse matrix is the least-square type $\mathbf{J}^+ = (\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T$. However, it increases computational cost because of $\mathbf{J}^T\mathbf{J} \in R^{n\times n}$. The dimension increases. Therefore, we use the following type to decrease the dimension.

$$\mathbf{J}^+ = \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1} \tag{12}$$

Figure 2: Blockdiagram of unknown nonlinear system

The dimension of $\mathbf{J}\mathbf{J}^T$ is $m \times m$ in (12). The computational cost becomes lower. Then, using the generalized inverse matrix, (9) is rewritten as

$$\mathbf{\Theta}^{k+1} = \mathbf{\Theta}^k - h\mathbf{J}^+(\mathbf{\Theta}^k)\mathbf{f}(\mathbf{\Theta}^k). \tag{13}$$

Regardless of I/O dimension, the solution of the inverse problem can be derived using (13).

3. Proposed Method

3.1. Solution of the Inverse Problem for Unknown System

In case of an unknown nonlinear system, a Jacobian matrix is need to approximate since a system function cannot be used. We propose a method of approximating the Jacobian matrix to solve an inverse problem for unknown system as shown in Fig. 2. In this system, only $\mathbf{r}$ can be observed as an output. The Jacobian matrix is derived by giving a minute displacement to only one of the control signals and observing the output. The i-th column of the Jacobian matrix is shown by

$$\mathbf{J}_{ith-col} = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_i} & \frac{\partial f_2}{\partial \theta_i} & \cdots & \frac{\partial f_m}{\partial \theta_i} \end{bmatrix}^T. \tag{14}$$

A minute displacement input and a displacement output are given by

$$\Delta\Theta^i = \begin{bmatrix} 0 & \cdots & \Delta\theta_i & 0 & \cdots & 0 \end{bmatrix}, \tag{15}$$

$$\Delta\mathbf{r}^i = \begin{bmatrix} \Delta r_1^i & \Delta r_2^i & \cdots & \Delta r_m^i \end{bmatrix}. \tag{16}$$

Then, $\mathbf{J}_{ith-col}$ is approximated as follows

$$\mathbf{J}_{ith-col} = \begin{bmatrix} -\frac{\Delta r_1^i}{\Delta\theta_i} & -\frac{\Delta r_2^i}{\Delta\theta_i} & \ldots & -\frac{\Delta r_m^i}{\Delta\theta_i} \end{bmatrix}^T. \tag{17}$$

The Jacobian matrix is obtained by carrying this operation on all $\mathbf{\Theta}$s.

$$\mathbf{J} = \begin{bmatrix} -\frac{\Delta r_1^1}{\Delta\theta_1} & \cdots & -\frac{\Delta r_1^i}{\Delta\theta_i} & \cdots & -\frac{\Delta r_1^n}{\Delta\theta_n} \\ -\frac{\Delta r_2^1}{\Delta\theta_1} & \cdots & -\frac{\Delta r_2^i}{\Delta\theta_i} & \cdots & -\frac{\Delta r_2^n}{\Delta\theta_n} \\ \vdots & & \vdots & & \vdots \\ -\frac{\Delta r_m^1}{\Delta\theta_1} & \cdots & -\frac{\Delta r_m^i}{\Delta\theta_i} & \cdots & -\frac{\Delta r_m^n}{\Delta\theta_n} \end{bmatrix} \tag{18}$$

However, even if the solution is obtained by these equations, the computational cost is high.

3.2. Dumping Coefficient

The calculation cost is decreased using the dumping coefficient. We use a dumping coefficient h to decrease the calculation cost. It decides the update frequency of $\mathbf{\Theta}$. We propose a method to make the dumping coefficient h controllable. In this research, we compare constant h with controllable h. Constant h updates $\mathbf{\Theta}$ each time. The controllable h does not need to update $\mathbf{\Theta}$ each time. Using controllable h, the process of the operation is as follows.

STEP1:
The Jacobian matrix at the starting point is calculated. The h_{min} which is the minimum controllable h is selected as small as possible.
STEP2:
The controllable h is gradually enlarged while keeping the value of the Jacobian matrix constant so that the end-point position of the manipulator moves to the target straight.
STEP3:
If an orbit of the manipulator leaves the target orbit, the Jacobian matrix is renewed. In this case, the controllable h is returned to former value.
STEP4:
STEP2 and STEP3 is repeated until the end-point position of the manipulator reaches the target position. We define an evaluation function and repeat these operations until the evaluation value is smaller than a decided constant value by changing the controllable h.

4. Simulation Result

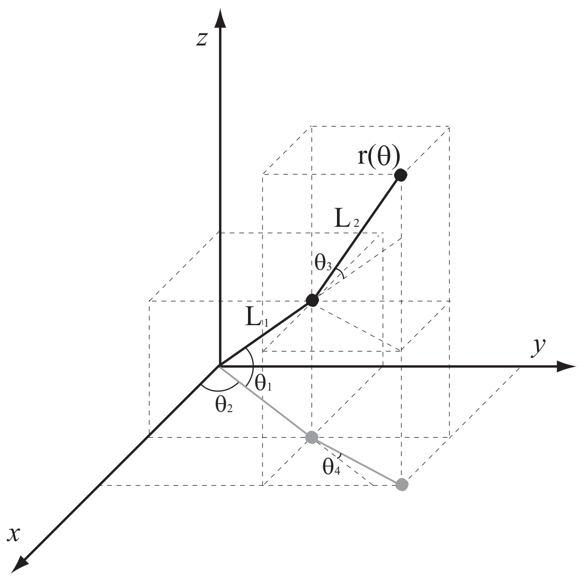

Figure 3: Manipulator of two joints

We simulate a manipulator of two joints shown in Fig. 3.

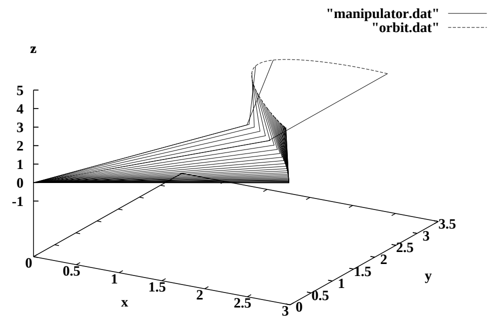

Figure 4: Orbit of manipulator (unknown system, controllable h)

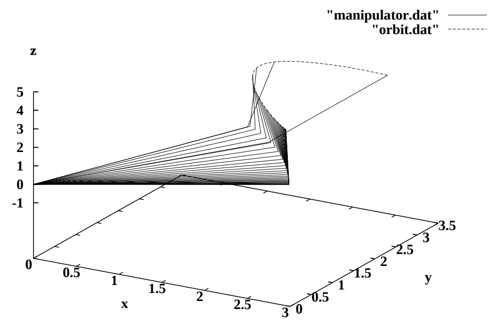

Figure 5: Orbit of manipulator (known system, constant h)

The nonlinear functions of this manipulator are given by

$$\mathbf{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} L_1 \cos\theta_1 \cos\theta_2 + L_2 \cos(\theta_1+\theta_3)\cos(\theta_2+\theta_4) \\ L_1 \cos\theta_1 \sin\theta_2 + L_2 \cos(\theta_1+\theta_3)\sin(\theta_2+\theta_4) \\ L_1 \sin\theta_1 + L_2 \sin(\theta_1+\theta_2) \end{bmatrix}, \tag{19}$$

where L_j is the length of j-th arm.
Length of arm:
$L_1 = 3.0$, $L_2 = 3.0$
Initial joint angle:
$\mathbf{\Theta}_0 = [\theta_1, \theta_2, \theta_3, \theta_4]^T = [30°, 45°, 30°, 15°]^T$
Initial end-point position of the maipulator:
$\mathbf{r} = [x, y, z]^T = [2.59\ ,\ 3.14\ ,\ 4.10\ ]^T$
Target end-point position of the manipulator:
$\mathbf{r}_d = [x_d, y_d, z_d]^T = [2.00\ ,\ 2.00\ ,\ 2.00\ ]^T$
Dumping coefficient:
Constant $h = 0.005$
$h_{min} = 0.001$

The end-point position of the arm reaches the target point in all the figures(Fig. 4-Fig. 7). Table. 3 shows conditions

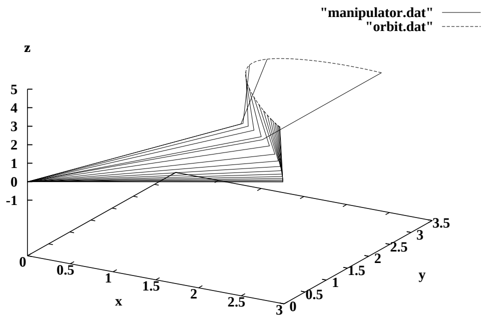

Figure 6: Orbit of manipulator (unknown system, constant *h*)

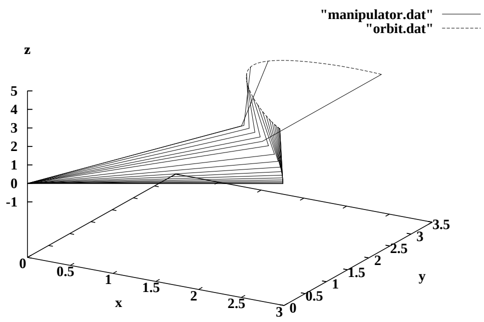

Figure 7: Orbit of manipulator (known system, controllable *h*)

Table 1: Comparison of update frequency (constant *h*)[iterations]

System	Joint angle $\boldsymbol{\Theta}$	Jacobian matrix
Known system	823	823
Unknown system	815	815

Table 2: Comparison of update frequency (controllable *h*)[iterations]

System	Joint angle $\boldsymbol{\Theta}$	Jacobian matrix
Known system	479	227
Unknown system	482	233

Table 3: Figures

Fig. #	System	Dumping coefficient *h*
4	Known	Constant
5	Unknown	Constant
6	Known	controllable
7	Unknown	controllable

of simulations. Whether the nonliear function is known or unknown, the orbits became about the same by using the approximated Jacobian matrix derived from (14)-(18). The result shows our proposed method is useful. The comparison of the update frequency is shown in Table. 1 and 2. In Table. 1, the update frequency of the Jacobian matrix is renewed each time. However, in Table. 2, the update frequency of the Jacobian matrix is 1/2 times as many as that of joint angle $\boldsymbol{\Theta}$ since controllable *h* does not need to update each time. Using controllable *h*, the update frequency of the Jacobian matrix is decreased. All the update frequencies of Table. 2 are much smaller than those of Table. 1. The computational cost become lower. The result shows controllable dumping coefficient *h* is effective.

5. Conclution

In this paper, we proposed a method of decreasing a computational cost that was needed to control a redundant manipulator in three-dimensional space. Comparing a controllable *h* with a constant *h*, the update frequency of joint angle $\boldsymbol{\Theta}$ became half and that of Jacobian matrix became quarter. Although the computational cost became lower, orbits of the manipulator was about the same. This result shows our proposed method is effective.

Additionally, we proposed the method of deriving the approximated Jacobian matrix for unknown nonlinear system. Whether the nonliear system was known or unknown, the orbits became about the same by using the the approximated Jacobian matrix. This result shows our proposed method that approximates Jacobian matrix is useful.

References

[1] Masao Inoue, “Control of Redundant Manipulator” *Papers of The Society of Instrument and Control Engineers*, vol. 26, no. 3, pp. 339-343, 1990

[2] Kenichi Tanaka, Yuko Shimizu and Kazuo Tsuchiya, “Solutionn of Inverse Kinematics for Redundant Manipulator by Neural Network,” *Papers of The Society of Instrument and Control Engineers*, vol.27 , no. 7, pp. 815-821, 1991.

[3] Koichi Koganezawa, “Fast Calculation Method of Inverse Kinematics for Manipulator which has degree of redundant freedom,” *Papers of The Society of Instrument and Control Engineers*, vol. 31, no. 3, pp. 341-346, 1995.

[4] Makoto Matsuyama, Naoaki Harada, Yuichi Tanji and Mamoru Tanaka, “Motion Dynamics Control by Circuit Analysis Method” *Report of The Institute of Electronics, Information and Communication Engineers*, NLP98-69, pp.39-44, Oct. 1998.