# Trajectory Tracking Control of Robot Manipulators using Discrete Time-varying Pole Placement Technique

Yasuhiko Mutoh, Masakatsu Kemmotsu and Lisa Awatsu

*Department of Engineering and Applied Sciences, Sophia University, 7-1 Kiocho, Chiyoda-ku, Tokyo, Japan*

Keywords: Trajectory Tracking Control, Linear Time-varying System, Discrete System, Time-Varying Pole Placement Control.

Abstract: For the trajectory tracking control problem of nonlinear systems, the most basic and classic strategy may be applying the linear control technique to a linear time-varying approximate model around some desired trajectory. However, this method is not commonly used because the design of a linear time-varying controller is not simple. The authors proposed the simple design method of the pole placement controller for linear time-varying discrete systems. In this paper, to show the applicability of the proposed linear time-varying discrete pole placement technique to the trajectory tracking control problem of nonlinear systems, we apply this control method to actual 2-link robot manipulator and present the experimental results.



## 1 INTRODUCTION

For the trajectory tracking control problem of nonlinear systems, the most basic and classic strategy may be applying the linear control technique to a linear time-varying approximate model around some desired trajectory. This method can be applied to any type of nonlinear systems. However, since, controller design method for linear time-varying system is not necessarily simple (Nguyen(1987)) (Valsek(1995)) (Valsek(1999)), gain scheduling strategy, the nonlinear control strategy, or PID control is commonly used for such a control design problem.

The author et.al. proposed the simple pole placement controller design method for linear time-varying discrete systems (Mutoh(2011)) (Mutoh and Hara (2011)). Such controller is obtained by finding a new output signal so that the relative degree from the input to this new output is equal to the system degree.

In this paper, we apply this control method to the tracking control of an actual 2-link robot manipulator to show the applicability of the proposed linear time-varying discrete pole placement technique to the trajectory tracking control problem of practical nonlinear systems. In the following, some basic properties of linear time-varying discrete systems are stated in Section 2. Section 3 summarizes the design procedure of a pole placement controller for linear time-varying discrete systems. In Section 4, this control method is applied to the trajectory tracking control problems of practical 2-link robot manipulator and experimental results are presented to show the validity of this control system.

## 2 BASIC PROPERTIES OF LINEAR TIME-VARYING DISCRETE SYSTEMS

In this section, some basic properties of linear time varying multi variable discrete systems are presented. Consider the following system.

$$x(k+1) \quad = A(k)x(k) + B(k)u(k) \tag{1}$$

Here, $x \in R^n$ and $u \in R^m$ are the state variable and the input. $A(k) \in R^{n\times n}$ and $B(k) \in R^{n\times m}$ are time-varying coefficient matrices. The state transition matrix of the system (1) from $k = j$ to $k = i$, $\Phi(i, j)$, is defined as follows.

$$\Phi(i, j) = A(i-1)A(i-2)\cdots A(j) \quad i > j \tag{2}$$

**Definition 1.** *System* (1) *is called "completely reachable in n steps" if and only if, for any $x_1 \in R^n$ there exists a bounded input $u(l)$ $(l = k, \cdots, k+n-1)$ such that $x(k) = 0$ and $x(k+n) = x_1$ for all k.*

**Lemma 1.** *System* (1) *is completely reachable in n steps if and only if the rank of the reachability matrix defined below is n for all k.*

$$U_R(k) = \left[\ B_0(k), B_1(k), \cdots, B_{n-1}(k)\ \right] \tag{3}$$

Mutoh Y., Kemmotsu M. and Awatsu L..
Trajectory Tracking Control of Robot Manipulators using Discrete Time-varying Pole Placement Technique.
DOI: 10.5220/0005533603730379
In *Proceedings of the 12th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2015), pages 373-379
ISBN: 978-989-758-122-9

*where,*

$$
\begin{aligned}
B_0(k) &= B(k+n-1)\\
B_1(k) &= \Phi(k+n,k+n-1)B(k+n-2)\\
&\ \vdots \\
B_{n-1}(k) &= \Phi(k+n,k+1)B(k)
\end{aligned}
\tag{4}
$$

*Let $b_i^l(k)$ be the i-th column of $B_l(k)$, then, the reachability matrix $U_R(k)$ can be written as*

$$
U_R(k) = \left[\ b_1^0(k)\cdots b_m^0(k)|\cdots|b_1^{n-1}(k)\cdots b_m^{n-1}(k)\ \right] \tag{5}
$$

Note that $b_i^r(k)$ also satisfies the same equation as (4), i.e.,

$$
\begin{aligned}
b_i^0(k) &= b_i(k+n-1)\\
b_i^1(k) &= \Phi(k+n,k+n-1)b_i(k+n-2)\\
&\ \vdots \\
b_i^{n-1} &= \Phi(k+n,k+1)b_i(k) \ \ (i=1,\cdots,m)
\end{aligned}
\tag{6}
$$

where $b_i(k)$ is the $i$-th column of $B(k)$. Suppose that the system (1) is completely reachable in $n$ steps. Then, the reachability indices, $\mu_i(i=1,\cdots,m)$, can be defined such that

$$
\sum_{i=1}^{m}\mu_i = n \tag{7}
$$

and the $n\times n$ truncated reachability matrix

$$
R(k) = \left[b_1^0(k),\cdots,b_1^{\mu_1-1}(k)|\cdots|b_m^0(k),\cdots,b_m^{\mu_m-1}\right] \tag{8}
$$

is non-singular. It is assumed that $\mu_1\geq\mu_2\geq\cdots\geq\mu_m$ without loss of generality.

Finally, the vector relative degree of a linear MIMO system is defined. Let the following $\eta(k)\in R^m$ be the output vector of the system (1).

$$
\eta(k) = H(k)x(k), \quad H(k)\in R^{m\times n} \tag{9}
$$

**Definition 2.** *System (1), (9) has the vector relative degree, $r_1$, $r_2$, $\cdots$, $r_m$ from $u$ to $\eta$, if and only if there exist some matrix $D(k)\in R^{m\times n}$ and some nonsingular matrix $\Lambda(k)\in R^{m\times m}$ that satisfy the following equation.*

$$
\begin{bmatrix} \alpha^1(z) & & \\ & \ddots & \\ & & \alpha^m(z) \end{bmatrix}\eta(k) = D(k)x(k)+\Lambda(k)u(k) \tag{10}
$$

*Here, $\alpha^k(z)$ is an arbitrary monic polynomial of degree $r_k$ and $z$ is a forward shift operator.*

# 3 DESIGN OF DISCRETE TIME-VARYING POLE PLACEMENT CONTROLLER

In this section, the design procedure of the pole placement controller for linear time-varying multi input discrete systems is summarized. Suppose that the system (1) is completely reachable with its reachability indices, $\mu_1,\cdots,\mu_m$. The problem is to design a state feedback for the system (1) so that the resulting time-varying closed-loop system becomes equivalent to some linear time-invariant system with arbitrarily stable poles. For this purpose, we first define a new output signal $y(k)\in R^m$ of the system (1) by

$$
y(k) = C(k)x(k) \tag{11}
$$

so that the total relative degree from $u(k)$ to $y(k)$ is equal to the system degree $n$. Here,

$$
y(k) = \begin{bmatrix} y_1(k)\\ y_2(k)\\ \vdots \\ y_m(k)\end{bmatrix}\in R^m,\ \ C(k) = \begin{bmatrix} c_1(k)\\ c_2(k)\\ \vdots \\ c_m(k)\end{bmatrix}\in R^{m\times n} \tag{12}
$$

where $y_i(k)\in R$ and $c_i(k)\in R^{1\times n}$. We have the following Theorem (Mutoh and Hara (2011)).

**Theorem 1.** *If the system (1) is completely reachable in $n$ steps, there exists a new output $y(k)$ such that the vector relative degree from $u(k)$ to $y(k)$ becomes $\mu_1,\cdots,\mu_m$, which implies that the total relative degree from $u(k)$ to $y(k)$ is $n$. And, such $C(k)$ can be calculated by the following equation.*

$$
C(k) = WR^{-1}(k-n) \tag{13}
$$

*where*

$$
\begin{aligned}
W &= diag(w_1,w_2,\cdots,w_m)\\
w_i &= \begin{bmatrix}0 & \cdots & 0 & 1\end{bmatrix}\in R^{1\times\mu_i}\\
&\qquad (i=1,\cdots,m)
\end{aligned}
\tag{14}
$$

From this, the pole placement state feedback is obtained in the following procedure.

Let $q^i(z)$ be the desired stable characteristic polynomial of $z$ as

$$
q^i(z) = z^{\,\mu_i}+\alpha^i_{\mu_i-1}z^{\,\mu_i-1}+\cdots+\alpha^i_1 z+\alpha^i_0. \quad (i=1,\cdots,m) \tag{15}
$$

Since, the vector relative degree from $u(k)$ to $y(k)$ is $\mu_1$, $\mu_2$, $\cdots$, $\mu_m$, we have the following equation.

$$
\begin{bmatrix} q^1(z) & & \\ & \ddots & \\ & & q^m(z) \end{bmatrix} y(k) = D(k)x(k)+\Lambda(k)u(k) \tag{16}
$$

Here,

$$D(k)=\begin{bmatrix} D_1(k)\\ D_2(k)\\ \vdots \\ D_m(k)\end{bmatrix},\ \Lambda(k)=\begin{bmatrix} \Lambda_1(k)\\ \Lambda_2(k)\\ \vdots \\ \Lambda_m(k)\end{bmatrix} \tag{17}$$

and $D_i(k)\in R^{1\times n}$ and $\Lambda_i(k)\in R^{1\times n}$ are defined by

$$D_i(k)=\left[\alpha_0^i,\alpha_1^i,\cdots,\alpha_{\mu_i-1}^i,1\right]\begin{bmatrix} c_i^0(k)\\ c_i^1(k)\\ \vdots \\ c_i^{\mu_i}(k)\end{bmatrix} \tag{18}$$

$$\Lambda_i(k)=[0,\cdots,0,1,\gamma_{i(i+1)},\cdots,\gamma_{im}].$$

In the above equation, $c_i^l(k)$ is defined by the following recursive equations, using $c_i(k)$,

$$\begin{aligned} c_i^0(k)&=c_i(k)\\ c_i^{(l+1)}(k)&=c_i^l(k+1)A(k) \\ &\quad (l=0,1,2,\cdots)\end{aligned} \tag{19}$$

and,

$$\gamma_{ij}=c_i^{\mu_i-1}(k+1)b_j(k) \tag{20}$$

for $i=1,\cdots,m$. Then, by applying the state feedback

$$u(k)=-\Lambda^{-1}(k)D(k)x(k) \tag{21}$$

to the system (1), the closed loop system becomes as follows.

$$\begin{bmatrix} q^1(z) & & \\ & \ddots & \\ & & q^m(z)\end{bmatrix} y(k)=0 \tag{22}$$

This system is time-invariant and has the following state representation.

$$w(k+1)=A^* w(k) \tag{23}$$

where $w(k)\in R^n$ is the new state variable. The matrices $A^*\in R^{n\times n}$ is written by

$$A^*=\begin{bmatrix} A_1^* & & 0\\ & \ddots & \\ 0 & & A_m^*\end{bmatrix} \tag{24}$$

and $A_i^*\in R^{\mu_i\times\mu_i}$ is defined as follows.

$$A_i^* = \begin{bmatrix} 0 & 1 & & 0\\ \vdots & & \ddots & \vdots\\ \vdots & & & 1\\ -\alpha_0^i & \dots & \dots & -\alpha_{\mu_i-1}^i\end{bmatrix} \quad (i=1,\dots,m) \tag{25}$$

From this, the characteristic polynomial of $A^*$ is written as follows using $q^i(z)$ defined by (15).

$$q(z)=\prod_{i=1}^{m} q^i(z) \tag{26}$$

(19) and (22) imply that $w(k)$ is written as follows.

$$w(k):=\begin{bmatrix} y_1(k)\\ \vdots\\ y_1(k+\mu_1-1)\\ \hline \vdots\\ \hline y_m(k)\\ \vdots\\ y_m(k+\mu_m-1)\end{bmatrix}=\begin{bmatrix} c_1^0(k)\\ \vdots\\ c_1^{\mu_1-1}(k)\\ \hline \vdots\\ \hline c_m^0(k)\\ \vdots\\ c_m^{\mu_m-1}(k)\end{bmatrix} x(k) = P(k)x(k) \tag{27}$$

On the other hand, from (1) and (21), the time-varying state equation of the closed loop system becomes

$$x(k+1)=(A(k)-B(k)\Lambda^{-1}D(k))x(k). \tag{28}$$

Thus, the system (28) is equivalent to the system (23), with the transformation matrix $P(k)$. It is then obvious that the following equation holds.

$$P(k+1)(A(k)-B(k)\Lambda^{-1}D(k))P^{-1}(k)=A^* \tag{29}$$

This implies that the state feedback (21) makes the closed loop system equivalent to the system (23) that has an arbitrarily stable characteristic polynomial, $q(z)$.

Note that the transformation matrix $P(k)$ and $P^{-1}(k)$ must be bounded functions, in other words, $P(k)$ must be a Lyapunov transformation, to ensure the stability of the closed-loop system.

The procedures to obtain the state feedback gain is summarized below.

**Pole Placement Design Procedure**

**STEP 1.** Calculate the reachability matrix $U_R(k-n)$ and the reachability indices $\mu_i$.

**STEP 2.** Calculate $C(k)=WR^{-1}(k-n)$ for the new output signal, $y(k)$, using the truncated reachability matrix $R(k)$.

**STEP 3.** Determine the desired stable closed-loop characteristic polynomials as follows for $i=1,\cdots,m$.

$$q^i(z)=z^{\mu_i}+\alpha_{\mu_i-1}^i z^{\mu_i-1}+\cdots+\alpha_1^i z+\alpha_0^i$$

**STEP 4.** Using (17) $\sim$ (20), calculate $D(k)$ and $\Lambda(k)$. Then, the state feedback for the pole placement is

$$u(k)=-\Lambda^{-1}(k)D(k)x(k)$$

# 4 TRAJECTORY TRACKING CONTROL OF 2-LINK MANIPULATORS

In this section, discrete time-varying pole placement technique is applied to the trajectory tracking control of a two-link robot manipulator.

## 4.1 The Model of the Manipulator

Fig. 1 and Fig. 2 show the picture and the model of the 2-link robot manipulator for the experiment. All links rotate in the horizontal plane.

Figure 1: Two-Link Manipulator(SR-402DDII.)

Its motion equation is described as follows.

$$M(\theta(t))\ddot{\theta}(t) + C(\theta(t),\dot{\theta}(t))\dot{\theta}(t) + D(\dot{\theta}(t)) = \tau(t) \tag{30}$$

where,

$$\begin{aligned}
\theta(t) &= \begin{bmatrix} \theta_1(t) \\ \theta_2(t) \end{bmatrix} \\
M(\theta(t)) &= \begin{bmatrix} J_1 + J_2 + 2m_2r_2l_1\cos\theta_2(t), & J_2 + m_2r_2l_1\cos\theta_2(t) \\ J_2 + m_2r_2l_1\cos\theta_2(t), & J_2 \end{bmatrix} \\
C(\theta(t),\dot{\theta}(t)) &= \begin{bmatrix} -2m_2r_2l_1\dot{\theta}_2(t)\sin\theta_2(t), & -m_2r_2l_1\dot{\theta}(t)_2\sin\theta_2(t) \\ m_2r_2l_1\dot{\theta}_1(t)\sin\theta_2(t), & 0 \end{bmatrix} \\
D(\dot{\theta}(t)) &= \begin{bmatrix} 2\mathrm{sgn}(\dot{\theta}_1(t)) \\ 0.25\mathrm{sgn}(\dot{\theta}_2(t)) \end{bmatrix} \\
J_i &= J_{l_i} + m_ir_i^2 \quad (i = 1,2).
\end{aligned}$$

Here, $\theta_i(t)$ and $\tau_i(t)$ are joint angle and input torque of $i$-th joint, $l_i$ and $r_i$ are length of the $i$-th link and the distance between the $i$-th joint and the center of gravity of the $i$-th link, and $J_{l_i}$ is the moment of inertia of the $i$-th link about its center of gravity $(i = 1,2)$. $D(\dot{\theta}(t))$ is a friction term which is estimated from the experimental data.

In the above, the values of the physical parameters are shown in Table 1.

Table 1: Parameter of Manipulator.

| variable ($i = 1,2$) | unit | link1 $i = 1$ | link2 $i = 2$ |
|---|---|---|---|
| $m_i$ | [kg] | 3.43 | 1.55 |
| $l_i$ | [m] | 0.2 | 0.2 |
| $r_i$ | [m] | 0.1 | 0.1 |
| $J_{l_i}$ | [$kgm^2$] | 0.208 | 0.03 |

## 4.2 Experimental Results

In this section, we show the experimental result of the trajectory tracking control of the 2-link robot manipulator using the time-varying discrete pole placement controller.

To design the discrete controller, we discretize the manipulator system (30) by Euler method as follows. Here, $T_s$ is the sampling time.

$$\begin{aligned}
x(k+1) &= x(k) + \begin{bmatrix} 0 & T_s I_2 \\ 0 & T_s\,\Gamma(x(k)) \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ T_s\,\Phi(x(k)) \end{bmatrix} u(k) \\
&= f(x(k),u(k))
\end{aligned} \tag{31}$$

where

$$\begin{aligned}
x(k) &= \begin{bmatrix} \theta(k) \\ \dot{\theta}(k) \end{bmatrix} \in R^4 \\
u(k) &= \begin{bmatrix} \tau_1(k) \\ \tau_2(k) \end{bmatrix} \in R^2 \\
I_2 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
\Gamma(x(k)) &= -M(\theta(k))^{-1}C(\theta(k),\dot{\theta}(k)) \in R^{2\times 2} \\
\Phi(x(k)) &= M(\theta(k))^{-1} \in R^{2\times 2}
\end{aligned} \tag{32}$$

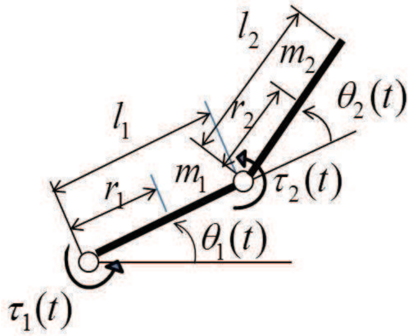


Figure 2: Two-Link Manipulator Model.

The sampling time $T_s$ is 10 [msec] for the experiment.

Let the desired trajectory of the end portion of this manipulator be the circle in the horizontal $X$-$Y$ work space as presented by the the following equation,

$$X(t) = 0.08\cos\frac{\pi}{5}t + 0.3 \tag{33}$$

$$Y(t) = 0.08\sin\frac{\pi}{5}t + 0.05 \tag{34}$$

which is described in Fig.3.

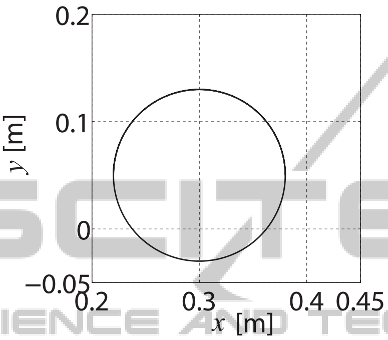


Figure 3: Desired Trajectory of the End Portion.

From the desired trajectory of the end portion, the desired trajectory of the joint angles, $\theta^*(t)$, and their speed, $\dot{\theta}^*(t)$, can be calculated using the inverse kinematics. Which gives the desired state variable $x^*(t)$ as follows.

$$x^*(t) = \begin{bmatrix} \theta^*(t) \\ \dot{\theta}^*(t) \end{bmatrix} \tag{35}$$

The desired input signal $u^*(t)$ is obtained from $\theta^*(t)$ and $\dot{\theta}^*(t)$ using (30). MAXIMA is used to calculate the explicit function representations for $x^*(t)$ and $u^*(t)$, which are omitted here because of the space limitation. Instead of this, the graphs of $x^*(t)$ and $u^*(t)$ are shown in Fig.4 and 5.

By discretizing these signals using the sampling time $T_s$, the discrete desired state trajectory $x^*(k)$ and the discrete desired input $u^*(k)$ are obtained. Note that we use the same variable for continuous space and discrete space, i.e., $x(t)$ is continuous variable,

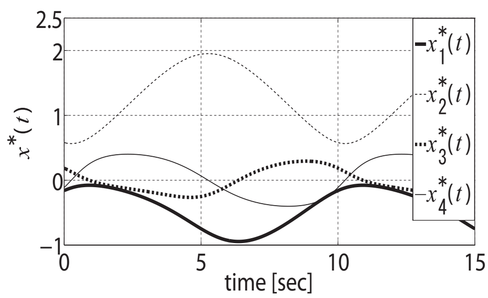


Figure 4: Desired Trajectory $x^*(t)$.

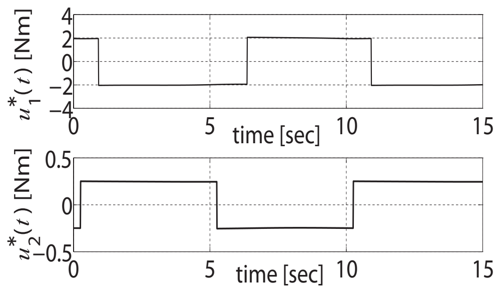


Figure 5: Desired Input $u^*(t)$.

$x(k)$ is a discrete variable of $k$-th step and $x(kT_s)$ is a sampling variable in the $t$-axis.

To obtain the linear time-varying approximate model around the desired trajectory, $x^*(k)$ and $u^*(k)$, define $\Delta x(k)$ and $\Delta u(k)$ by

$$\begin{cases} \Delta x(k) = x(k) - x^*(k) \\ \Delta u(k) = u(k) - u^*(k) \end{cases} \tag{36}$$

Then we have the following approximate model from (31).

$$\begin{aligned} x(k+1) &= \frac{\partial}{\partial x} f(x^*(k), u^*(k))\Delta x(k) \\ &\quad + \frac{\partial}{\partial u} f(x^*(k), u^*(k))\Delta u(k) \\ &= A(k)\Delta x(k) + B(k)\Delta u(k) \end{aligned} \tag{37}$$

where,

$$A(k) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & a_{32}(k) & a_{33}(k) & a_{34}(k) \\ 0 & a_{42}(k) & a_{43}(k) & a_{44}(k) \end{bmatrix} \tag{38}$$

$$B(k) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \beta_{31}(k) & \beta_{32}(k) \\ \beta_{41}(k) & \beta_{42}(k) \end{bmatrix} \tag{39}$$

Here, the explicit function representation of A(k) and B(k) are obtained by using MAXIMA, which are described in Appendix A, for reference.

Fig.6 shows the closed loop response of the manipulator end portion in the horizontal work space. The initial position of the end portion is $(0.4, 0)$ in the coordinate of the horizontal work space. This initial condition corresponds to the initial condition of state variable vector, $x_1(0) = x_2(0) = x_3(0) = x_4(0) = 0$. The response of the manipulator state variable $x(kT_s)$ (joint angles and their speed), the state error $\Delta x(kT_s) = x(kT_s) - x^*(kT_s)$, are shown in Fig.7

and Fig.8 respectively. The control input $u(kT_s) = u^*(kT_s) + \Delta u(kT_s)$ is shown in Fig.9. The desired stable poles of the closed loop system and the observer are chosen as $(-5, -90, -5, -90)$.

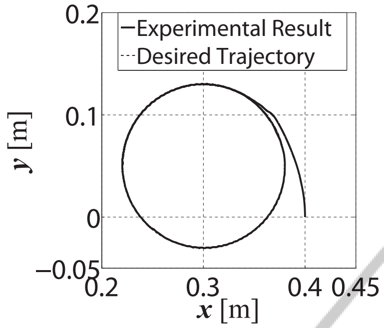


Figure 6: Response of End Portion.

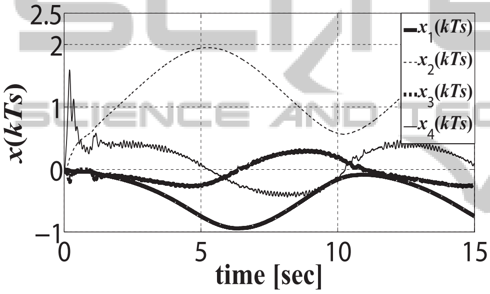


Figure 7: Response of State Variable $x(k)$.

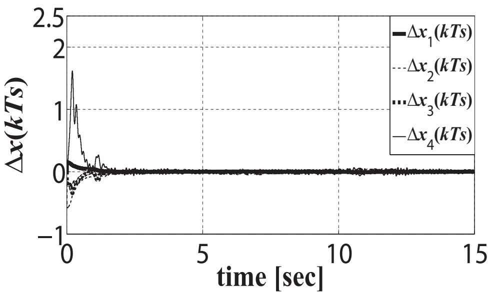


Figure 8: State Error $\Delta x(k) = x(k) - x^*(k)$.

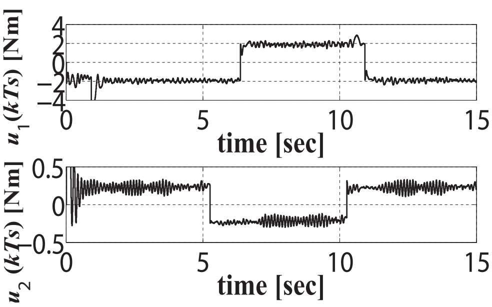


Figure 9: Input Torque $u(k)$.

# 5 CONCLUSIONS

In this paper, the trajectory tracking control of nonlinear systems was considered. For this purpose, the pole placement controller designed by the simple procedure was applied to the linear time-varying discrete approximate model of the system around some desired trajectory. This controller was applied to tracking control of the actual 2-link robot manipulator to show the applicability of this type of controller. The experimental results showed that this controller has very good performance.

# REFERENCES

Chi-Tsong Chen (1999) C *Linear System Theory and Design (Third edition).* Oxford University Press

Charles C. Nguyen (1987) C *Arbitrary eigenvalue assignments for linear time-varying multivariable control systems.* International Journal of Control, 45-3, 1051–1057

W. J. Rugh (1993) *Linear System Theory 2nd Edition* prentice hall

Y. Mutoh (2011) C *A New Design Procedure of the Pole-Placement and the State Observer for Linear Time-Varying Discrete Systems.* Informatics in Control, Automation and Robotics, p.321-334, Springer

Michael Valášek (1995) C *Efficient Eigenvalue Assignment for General Linear MIMO systems.* Automatica, 31-11, 1605–1617

Michael Valášek, Nejat Olgaç (1999) C *Pole placement for linear time-varying non-lexicographically fixed MIMO systems.* Automatica, 35-1, 101–108

Y. Mutoh and N. Kimura (2011) C *Observer-Based Pole Placement for Non-Lexicographically-fixed Linear Time-Varying Systems.* 50th IEEE CDC and ECC

Y. Mutoh and T. Hara (2011) C *A New Method for Pole Placement of Linear Time-Varying Discrete Multivariable Systems.* 30th CCC

# APPENDIX

The followings are the explicit form of functioins of the elements of $A(t)$ and $B(t)$ in equation (37) calculated by MAXIMA. We used this result with $T = T_s$.

$$
\begin{aligned}
a_{32} &= \frac{31\left(28830\cos(x_2(k))^3 x_4(k)^2 + 185340\cos(x_2(k))\,x_4(k)^2\right)T}{(31\cos(x_2(k))-90)^2\,(31\cos(x_2(k))+90)^2} \\
&+ \frac{31\left(57660\cos(x_2(k))^3 x_3(k)\,x_4(k) + 370680\cos(x_2(k))\,x_3(k)\,x_4(k)\right)T}{(31\cos(x_2(k))-90)^2\,(31\cos(x_2(k))+90)^2} \\
&+ \frac{31\left(28830\cos(x_2(k))^3 x_3(k)^2 + 472409\cos(x_2(k))^2 x_3(k)^2\right)}{(31\cos(x_2(k))-90)^2\,(31\cos(x_2(k))+90)^2} \\
&+ \frac{31\left(185340\cos(x_2(k))\,x_3(k)^2 - 251100\,x_3(k)^2\right)T}{(31\cos(x_2(k))-90)^2\,(31\cos(x_2(k))+90)^2} \\
a_{33} &= 1 - \frac{62\sin(x_2(k))\,(30x_4(k) + 31\cos(x_2(k))\,x_3(k) + 30x_3(k))\,T}{(31\cos(x_2(k))-90)\,(31\cos(x_2(k))+90)} \\
a_{34} &= -\frac{1860\sin(x_2(k))\,(x_4(k)+x_3(k))\,T}{(31\cos(x_2(k))-90)\,(31\cos(x_2(k))+90)} \\
a_{42} &= -\frac{31(28830\cos(x_2(k))^3 x_4(k)^2 + 472409\cos(x_2(k))^2 x_4(k)^2)T}{(31\cos(x_2(k))-90)^2(31\cos(x_2(k))+90)^2} \\
&- \frac{31(185340*\cos(x_2(k))x_4(k)^2 - 251100x_4(k)^2)T}{(31\cos(x_2(k))-90)^2(31\cos(x_2(k))+90)^2} \\
&- \frac{31(57660\cos(x_2(k))^3 * x_3(k)x_4(k) + 944818\cos(x_2(k))^2 x_3(k)x_4(k))T}{(31\cos(x_2(k))-90)^2(31\cos(x_2(k))+90)^2} \\
&- \frac{31(370680\cos(x_2(k))x_3(k)x_4(k) - 502200x_3(k)x_4(k))T}{(31\cos(x_2(k))-90)^2(31\cos(x_2(k))+90)^2} \\
&- \frac{31(288300\cos(x_2(k))^3 x_3(k)^2 + 944818\cos(x_2(k))^2 x_3(k)^2)T}{(31\cos(x_2(k))-90)^2(31\cos(x_2(k))+90)^2} \\
&+ \frac{31(1853400\cos(x_2(k))^2 x_3(k) - 502200x_3(k)^2)T}{(31\cos(x_2(k))-90)^2(31*\cos(x_2(k))+90)^2} \\
a_{43} &= \frac{62\sin(x_2(k))\,(31\cos(x_2(k))\,x_4(k) + 30x_4(k) + 62\cos(x_2(k))\,x_3(k))\,T}{(31\cos(x_2(k))-90)\,(31\cos(x_2(k))+90)} \\
&+ \frac{31\,(300x_3(k))\,T}{(31\cos(x_2(k))-90)\,(31\cos(x_2(k))+90)} \\
a_{44} &= \frac{62\,(31\cos(x_2(k))+30)\,\sin(x_2(k))\,(x_4(k)+x_3(k))\,T}{(31\cos(x_2(k))-90)\,(31\cos(x_2(k))+90)} + 1 \\
b_{31} &= -\frac{30000\,T}{961\cos(x_2(k))^2 - 8100}
\end{aligned}
$$

$$
\begin{aligned}
b_{32} &= \frac{(31000\cos(x_2(k)) + 30000)\,T}{961\cos(x_2(k))^2 - 8100} \\
b_{41} &= \frac{(31000\cos(x_2(k)) + 30000)\,T}{961\cos(x_2(k))^2 - 8100} \\
b_{42} &= -\frac{(62000\cos(x_2(k)) + 300000)\,T}{961\cos(x_2(k))^2 - 8100}.
\end{aligned}
$$