PRACTICAL APPLICATION OF A LINEAR TIME-VARYING POLE PLACEMENT TECHNIQUE TO TRAJECTORY TRACKING CONTROL OF A MANIPULATOR

Yasuhiko Mutoh and Masakatsu Kemmotsu
Department of Engineering and Applied Sciences
Sophia University
Chiyoda-ku, Tokyo, Japan
email: y MUTOU@sophia.ac.jp

ABSTRACT
The author proposed the simple design procedure of pole placement controller for linear time-varying multi-input systems. The feedback gain can be obtained directly from the plant parameters without transforming the system into any standard form. This design method will be applied to the problem of the trajectory tracking control of 2-link manipulator.

KEY WORDS
Tracking Control; Manipulator Control; Nonlinear System; Linear Time-Varying System; Pole Placement.

1 Introduction
The control of the robot manipulator is a nonlinear control problem. To stabilize nonlinear systems around some particular position, we approximate the system around this operating point by a linear time-invariant system, and then, linear time-invariant control design methods are applied. But, if we need to track some particular trajectory in wide range, gain scheduling strategies or nonlinear controllers are used, in general. The computed torque method is one of the most useful such controllers [1] [2]. On the other hand, there is another basic and classic control strategy. That is, by approximating the system around some desired trajectory by a linear time varying system, and then, various linear time varying controller can be applied. However, since, controller design method for linear time-varying system is not necessarily simple [5] [9] [10], it seems that this control strategy is not commonly used. The author et. al. have proposed a simple pole placement controller design method [7]. 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. It was also shown that this type of controller can be applied to the trajectory tracking control problem by using a numerical example [6].

In this paper, such time-varying pole placement control technique will be used to the trajectory tracking problem of the real 2-link robot manipulator to show the validity of time varying controller approach. The problem is to design the input torque to the joint so that the end portion tracks some desired trajectory. Since, the proposed method is based on the linear time-varying approximate model around some desired trajectory, it has robustness for the modeling error for the nonlinearity and has applicability of the observer-based controller for this type of manipulator control in the framework of the linear time-varying control theory.

In Section 2, the linear time-varying pole placement design procedure is summarized. Then, this method is applied to a real 2-link robot manipulator control problem, and an experimental result will be shown in Section 3.

2 Pole Placement of Multi-Input Systems
2.1 Tracking Control of Nonlinear Systems
In this section, we summarize the trajectory tracking controller design method of nonlinear systems using linear time-varying control technique. Consider the following nonlinear system.

\[ \dot{x}(t) = f(x(t), u(t)) \]  \hspace{1cm} (1)

Here, \( x(t) \in R^n \) and \( u(t) \in R^m \) are the state variable and the input signal. Let \( x^*(t) \) and \( u^*(t) \) be some desired trajectory and desired input signal, that is,

\[ \dot{x}^*(t) = f(x^*(t), u^*(t)), \quad x^*(0) = x_0^* \]  \hspace{1cm} (2)

where the initial state, \( x_0^* \) is on the desired trajectory. The problem is to design the controller to make \( x(t) \) track the desired trajectory, \( x^*(t) \).

Let \( \Delta x(t) \) and \( \Delta u(t) \) be defined by

\[ \Delta x(t) = x(t) - x^*(t) \]
\[ \Delta u(t) = u(t) - u^*(t) \]  \hspace{1cm} (3)

Then, (1) can be approximated by the following linear time-varying system around \( (x^*(t), u^*(t)) \).

\[ \Delta \dot{x}(t) = A(t) \Delta x(t) + B(t) \Delta u(t) \]  \hspace{1cm} (4)
Consider the following linear time-varying system.

\[
A(t) = \frac{\partial}{\partial x} f(x^*(t), u^*(t)) \\
B(t) = \frac{\partial}{\partial u} f(x^*(t), u^*(t)) \tag{5}
\]

This system is a linear time-varying multi input system. In this paper, by stabilizing \(\Delta x(t)\), the trajectory tracking controller is designed using linear time-varying pole placement technique.

2.2 Linear Time-Varying Pole Placement Design Procedure

Consider the following linear time-varying system.

\[
\dot{x}(t) = A(t) x(t) + B(t) u(t) \tag{6}
\]

where, \(x(t) \in \mathbb{R}^n\) and \(u(t) \in \mathbb{R}^m\) are the state variable and the input signal vectors. \(A(t) \in \mathbb{R}^{n \times n}\) and \(B(t) \in \mathbb{R}^{n \times m}\) are the time varying coefficient matrices, which are smooth functions of \(t\). Using vectors \(b_i(t) \in \mathbb{R}^n\), the matrix \(B(t)\) can be written as follows.

\[
B(t) = \begin{bmatrix} b_1(t) & b_2(t) & \cdots & b_m(t) \end{bmatrix} \tag{7}
\]

Then, the pole placement controller design procedure for linear time varying systems is as follows.

**STEP 1** Using the following recursive equations,

\[
b_k^0(t) = b_k(t) \\
b_k^{i+1}(t) = A(t)b_k^i(t) - \dot{b}_k^i(t) \\
\quad \quad k = 1, 2, \ldots, m \quad i = 0, 1, 2, \ldots \tag{8}
\]

we can define the controllability matrix by

\[
U_C(t) = \begin{bmatrix} b_1^0(t) & \cdots & b_m^0(t) \\ \vdots & \ddots & \vdots \\ b_1^{n-1}(t) & \cdots & b_m^{n-1}(t) \end{bmatrix} \tag{9}
\]

Then, check the controllability of the system (1). If the system is controllable, find the controllability indices \(\mu_1, \ldots, \mu_m\). Then, define the nonsingular matrix, \(R(t)\) by

\[
R(t) = \begin{bmatrix} b_1^0(t) & \cdots & b_1^{\mu_1-1}(t) \\ \vdots & \ddots & \vdots \\ b_m^0(t) & \cdots & b_m^{\mu_m-1}(t) \end{bmatrix} \tag{10}
\]

which is called the truncated controllability matrix.

**STEP 2** Calculate the new output matrix, \(C(t) \in \mathbb{R}^{m \times n}\), by

\[
C(t) = \begin{bmatrix} c_1(t) \\ \vdots \\ c_m(t) \end{bmatrix} = WR^{-1}(t). \tag{11}
\]

where

\[
W = \text{diag}(w_1, w_2, \ldots, w_m) \tag{12}
\]

and

\[
w_i = [0, \ldots, 0, 1] \in \mathbb{R}^{1 \times \mu_i} \\
\quad \quad (i = 1, \ldots, m) \tag{13}
\]

Here, \(c_i(t) \in \mathbb{R}^{1 \times m}\). Then, calculate \(c_i^j(t) \in \mathbb{R}^{1 \times m}\) by the following recursive equations.

\[
c_i^0(t) = c_i(t) \\
c_i^{j+1}(t) = c_i^j(t)A(t) + \dot{c}_i^j(t) \\
\quad \quad (k = 1, \ldots, m \quad i = 1, 2, \ldots) \tag{14}
\]

**STEP 3** Determine \(m\) desired stable characteristic polynomials for the closed system by

\[
\alpha^i(p) = p^{\mu_i} + \alpha_{\mu_i-1} p^{\mu_i-1} + \cdots + \alpha_0 \tag{15}
\]

where \(p\) is the differential operator.

If we define a new output signal \(y(t) \in \mathbb{R}^m\) by

\[
y(t) = C(t)x(t)
\]

the total relative degree from \(u(t)\) to this new output, \(y(t)\), becomes the system degree, \(n\), and the vector relative degree becomes \(\mu_1, \ldots, \mu_m\). And, hence, we have the following equation.

\[
\begin{bmatrix} \alpha^1(s) \\ \vdots \\ \alpha^m(s) \end{bmatrix} y(t) = D(t)x(t) + \Lambda(t)u(t)
\]

where

\[
D(t) = \begin{bmatrix} D_1(t) \\ D_2(t) \\ \vdots \\ D_m(t) \end{bmatrix}, \quad \Lambda(t) = \begin{bmatrix} \Lambda_1(t) \\ \Lambda_2(t) \\ \vdots \\ \Lambda_m(t) \end{bmatrix} \tag{16}
\]

and,

\[
D_i(t) = [\alpha_0^i, \alpha_1^i, \ldots, \alpha_{\mu_i-1}^i, 1] \\
\Lambda_i(t) = [0, \ldots, 0, 1, \gamma_{\mu_i}^i(t), \ldots, \gamma_{im}^i(t)] \tag{17}
\]

where

\[
\gamma_{ij}(t) = c_i^{(\mu_i-1)}(t)b_j(t) \tag{18}
\]

Then, we have the last STEP.

**STEP 4** Finally, the pole placement state feedback is

\[
u(t) = -\Lambda^{-1}(t)D(t)x(t) \tag{19}
\]
From the above steps, the new output \( y \) satisfies the following equation as a closed loop system.

\[
\begin{bmatrix}
\alpha^1(s) \\
\vdots \\
\alpha^m(s)
\end{bmatrix}
y(t) = 0
\tag{20}
\]

This has the following state realization with \( w(t) \in \mathbb{R}^n \) as a new state variable.

\[
\dot{w}(t) = A^* w(t) = \begin{bmatrix}
A_1^* & 0 \\
0 & \ddots \\
0 & \ddots & A_m^*
\end{bmatrix} w(t)
\tag{21}
\]

\[A_i^* = \begin{bmatrix}
0 & 1 & 0 \\
\vdots & \ddots & \vdots \\
-\alpha_i & \ldots & -\alpha_{i-1}
\end{bmatrix} \in \mathbb{R}^{\mu_i \times \mu_i}, (i = 1, \ldots, m)
\tag{22}\]

The characteristic polynomial of \( A_i^* \) is \( \alpha^i(p) \).

On the other hand, using (6) and (19), the closed loop state equation becomes

\[
\dot{x}(t) = (A(t) - B(t)A^{-1}(t)D(t))x(t)
\tag{23}
\]

It is readily shown that \( x(t) \) and \( \omega(t) \) satisfy the following relation by the straightforward calculation [7].

\[
w(t) = T(t)x(t)
\tag{24}
\]

\[
T(t) = \begin{bmatrix}
\epsilon_i(t) \\
\vdots \\
\epsilon_{i-1}(t) \\
\vdots \\
\epsilon_m(t)
\end{bmatrix}
\tag{25}
\]

It is well known that the exponential stability is preserved between two equivalent linear time-varying systems if the transformation matrix is Lyapunov transformation [3]. If \( T(t) \) is nonsingular, and both of \( T(t) \) and \( T^{-1}(t) \) are continuous and bounded for all \( t \), \( T(t) \) is called Lyapunov transformation matrix [4].

**Theorem 1** In the above pole placement control, the closed loop system is exponentially stable if the transformation matrix \( T(t) \) in (25) is Lyapunov transformation.

### 3 Tracking Control of a Two-Link Robot Manipulator

In this section, we apply the linear time-varying control technique [6] to the real manipulator equipment, to show its availability to real systems. Fig.1 is a picture of 2-link manipulator.

The model of the manipulator is depicted in Fig.2. All links rotate in the horizontal plane. Its motion equation is described as follows.

\[
M(\theta(t))\ddot{\theta}(t) + C(\theta(t), \dot{\theta}(t))\dot{\theta}(t) + D(\dot{\theta}(t)) = \tau(t)
\tag{26}
\]

where,

\[
\theta(t) = \begin{bmatrix}
\theta_1(t) \\
\theta_2(t)
\end{bmatrix}
\]

\[
M(\theta(t)) = \begin{bmatrix}
J_1 + J_2 + 2m_2r_2l_1\cos\theta_2(t), & J_2 + m_2r_1l_1\cos\theta_2(t), \\
J_2 + m_2r_1l_1\cos\theta_2(t), & J_2
\end{bmatrix}
\]

\[
C(\theta(t), \dot{\theta}(t)) = \begin{bmatrix}
-2m_2r_2l_1\dot{\theta}_2(t)\sin\theta_2(t), & m_2r_2l_1\dot{\theta}_2(t)\sin\theta_2(t), \\
-m_2r_2l_1\dot{\theta}_2(t)\sin\theta_2(t), & 0
\end{bmatrix}
\]

\[
D(\dot{\theta}(t)) = \begin{bmatrix}
2\text{sgn}(\dot{\theta}_1(t)) & 0.25\text{sgn}(\dot{\theta}_2(t)) \\
J_i & 0.25\text{sgn}^2(\dot{\theta}_2(t))
\end{bmatrix}
\]

\[
J_i = J_i + m_i r_i^2 \quad (i = 1, 2).
\]

Here, \( \theta_i(t) \) and \( \tau_i(t) \) are a joint angle and an input torque of \( i \)-th joint, \( l_i \) and \( r_i \) are a length of the \( i \)-th link and the
where the \( i \)-th joint and the center of gravity of the \( i \)-th link, and \( J_{li} \) is the moment of inertia of \( i \)-th link about its center of gravity. \( D(\dot{\theta}(t)) \) is a friction term.

Eq.(26) can be written as a following state equation.

\[
\dot{x}(t) = f(x(t), u(t)) = \begin{bmatrix} 0 & I \\ 0 & \Gamma(x(t)) \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ \Phi(x(t)) \end{bmatrix} u(t) \tag{27}
\]

where,

\[
x(t) = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \end{bmatrix} \in \mathbb{R}^4
\]

\[
u(t) = \begin{bmatrix} \tau_1(t) \\ \tau_2(t) \end{bmatrix} \in \mathbb{R}^2 
\frac{\dot{\theta}}{2}
\]

\[
I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\]

\[
\Gamma(x(t)) = -M(\dot{\theta}(t))^{-1}C(\theta(t), \dot{\theta}(t)) \in \mathbb{R}^{2 \times 2}
\]

\[
\Phi(x(t)) = M(\dot{\theta}(t))^{-1} \in \mathbb{R}^{2 \times 2}
\]

The values of the physical parameters of our 2-link manipulator are shown in Table 1.

It should be noted that, if the number of links increases, the calculation process for the desired input signal and linear time-varying approximate model becomes complicated. From this point of view, we use MAXIMA which is the free software for symbolic calculation.

<table>
<thead>
<tr>
<th>Table 1. Parameter of Manipulator</th>
</tr>
</thead>
<tbody>
<tr>
<td>unit</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>Mass</td>
</tr>
<tr>
<td>Length</td>
</tr>
<tr>
<td>Center of Gravity</td>
</tr>
<tr>
<td>Inertia</td>
</tr>
</tbody>
</table>

In general, a problem of calculating the desired input signal \( u^*(t) \) for given desired trajectory \( x^*(t) \) is difficult. This is because \( x^*(t) \) and \( u^*(t) \) must satisfy the system differential equation (27). However, in the manipulator case, it is straightforward to calculate \( u^*(t) = \tau^*(t) \) from the desired joint angle \( \theta^*(t) \), \( \dot{\theta}^*(t) \) and \( \ddot{\theta}^*(t) \) using (27). Since many of mechanical systems can be represented by a state equation like (27), it is also straightforward to find \( u^*(t) \) from \( x^*(t) \) for those systems.

Let the desired trajectory of the end portion of this manipulator be the circle in the horizontal x-y work space described in Fig.3. From this 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, i.e., the desired state trajectory is defined by

\[
x^*(t) = \begin{bmatrix} \theta^*(t) \\ \dot{\theta}^*(t) \end{bmatrix} \tag{29}
\]

Then, the desired input signal \( u^*(t) \) for \( x^*(t) \) can be obtained from Eq.(26).

MAXIMA is used to calculate the explicit function representations for \( x^*(t) \) and \( u^*(t) \), which are omitted because of the space limitation. Instead of this, the graphs of \( x^*(t) \) and \( u^*(t) \) are shown in Fig.4 and 5.

The manipulator system (27) is approximated by the following linear time-varying system around \( (x^*(t), u^*(t)) \).

\[
\Delta \dot{x}(t) = A(t) \Delta x(t) + B(t) \Delta u(t) \tag{30}
\]

where,
\[ A(t) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ a_{32} & a_{33} & a_{34} \\ a_{42} & a_{43} & a_{44} \end{bmatrix} \]  

\[ B(t) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \beta_{31} & \beta_{32} \\ \beta_{41} & \beta_{42} \end{bmatrix} \]  

Here, the explicit functions of \( A(t) \) and \( B(t) \) are obtained by using MAXIMA which are shown below for reference.

\[
\alpha_{32} = \{ (893730 \cos^3 x_2 + 7475340 \cos x_2) x_4^2 \\
+ (1787460 \cos^3 x_2 + 14950680 \cos x_2) x_3 x_4 \\
+ (893730 \cos^3 x_2 + 18219599 \cos^2 x_2 + 7475340 \cos x_2 \\
- 9571560) x_3^2 \} / (961 \cos^2 x_2 - 9960)^2
\]

\[
\alpha_{33} = -62 \sin x_2 (30 x_4 + 31 (\cos x_2) x_3 + 30 x_3) \\
/ 961 \cos^2 x_2 - 9960
\]

\[
\alpha_{34} = -1860 \sin x_2 (x_4 + x_3) \\
/ 961 \cos^2 x_2 - 9960
\]

\[
\alpha_{42} = 31 \{ 28830 (\cos^3 x_2) x_4^2 + 587729 (\cos^2 x_2) x_4^2 \\
+ 241140 \cos x_2 x_4 x_3^2 - 308700 x_4^2 + 57660 (\cos^3 x_2) x_3 x_4 \\
+ 1175458 (\cos^2 x_2) x_3 x_4 + 482280 (\cos x_2) x_3 x_4 \\
+ 347882 (\cos^3 x_2) x_3^2 + 1175458 (\cos x_2 x_4)^2 x_3 \\
+ 2909756 (\cos x_2) x_3^3 - 617520 x_3^2 \} \\
/ (961 \cos^2 x_2 - 9960)^2
\]

\[
\alpha_{43} = 62 \sin x_2 (31 (\cos x_2) x_4 + 30 x_4 + 62 \cos x_2 x_3 + 362 x_3) \\
/ 961 \cos^2 x_2 - 9960
\]

\[
\alpha_{44} = 62 (31 \cos x_2 x_4 + 30 \sin x_2 (x_4 + x_3) \\
/ 961 \cos^2 x_2 - 9960
\]

\[
\beta_{31} = -\frac{961 \cos^2 x_2 - 8100}{30000}
\]

\[
\beta_{32} = \frac{31000 \cos x_2 + 30000}{30000}
\]

\[
\beta_{41} = \frac{31000 \cos x_2 + 30000}{961 \cos^2 x_2 - 8100}
\]

\[
\beta_{42} = \frac{-62000 \cos x_2 + 300000}{961 \cos^2 x_2 - 8100}
\]

Using the above equations, the pole placement controller is designed with the desired stable closed loop eigenvalues as \((-6, -150, -6, -100)\).

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 work space which corresponds to the initial state variable, \(x_1(0) = x_2(0) = x_3(0) = x_4(0) = 0\). The control input \(u(t) = u^*(t) + \Delta u(t)\) is shown in Fig.7. Fig.8 presents the response of the state variable vector (joint angles and their speed). From this, the linear time-varying pole placement technique works well for the trajectory tracking control of 2-link robot manipulator.

4 Conclusion

This paper concerned with the problem of trajectory tracking control of nonlinear systems. Nonlinear system can be approximated using a linear time-varying system along some desired trajectory. The author already proposed the simple design procedure of the pole placement controller for linear time-varying system. The paper showed that this design method has a good availability to the trajectory tracking control of real 2-link manipulator.

References


