Intelligent Control and Automation, 2017, 8, 139-153
http://www.scirp.org/journal/ica
ISSN Online: 2153-0661
ISSN Print: 2153-0653

An Improvement on Data-Driven Pole Placement for State Feedback Control and Model Identification

Pyone Ei Ei Shwe[1], Shigeru Yamamoto[2]

[1]Division of Electrical Engineering and Computer Science, Graduate School of Natural Science and Technology, Kanazawa University, Kanazawa, Japan
[2]Faculty of Electrical and Computer Engineering, Institute of Science and Engineering, Kanazawa University, Kanazawa, Japan
Email: pyone@moccos.ec.t.kanazawa-u.ac.jp, shigeru@se.kanazawa-u.ac.jp

How to cite this paper: Shwe, P.E.E. and Yamamoto, S. (2017) An Improvement on Data-Driven Pole Placement for State Feedback Control and Model Identification. *Intelligent Control and Automation*, **8**, 139-153.
http://dx.doi.org/10.4236/ica.2017.83011

Received: May 25, 2017
Accepted: July 15, 2017
Published: July 18, 2017

Abstract

The recently proposed data-driven pole placement method is able to make use of measurement data to simultaneously identify a state space model and derive pole placement state feedback gain. It can achieve this precisely for systems that are linear time-invariant and for which noiseless measurement datasets are available. However, for nonlinear systems, and/or when the only noisy measurement datasets available contain noise, this approach is unable to yield satisfactory results. In this study, we investigated the effect on data-driven pole placement performance of introducing a prefilter to reduce the noise present in datasets. Using numerical simulations of a self-balancing robot, we demonstrated the important role that prefiltering can play in reducing the interference caused by noise.

Keywords

Data-Driven Control, State Feedback, Pole Placement, Nonlinear Systems

1. Introduction

In state feedback pole placement, the state feedback gain must be determined for a given system such that the closed-loop poles coincide with the desired locations. This is a well-known problem, and the pole placement methods have been extensively discussed in the literature [1] [2] [3] [4]. In standard pole placement methods, a state space model is assumed to be given by a system identification technique using data from past experiments. Whereas the traditional approach combines the identification of the state space model with the standard pole placement method; an alternative approach called "data-driven pole placement" has recently been proposed [5]. In this approach, the state space model and pole placement feedback gain are identified simultaneously from the set of state measurements and control input sequences. The method proposed in [5] is based on the data-driven control framework ([6] and references therein) such as unfalsified control [7], virtual reference feedback tuning (VRFT) [8] [9], or fictitious reference iterative tuning (FRIT) [10] [11] [12] [13]. In the data-driven control framework, where no explicit mathematical plant model is used, a feedback controller must be derived that satisfies the prescribed closed-loop performance and fits to known experimental data. In contrast with traditional model-based controller designs, techniques such as controller identification [14] or a combination of plant model and controller identification must be applied [15] [16].

Many studies of data-driven control have focused on output feedback control and data-driven state feedback control [11] [12] [13], in which the prescribed closed-loop performance is achieved by applying a closed-loop reference transfer function. Such methods can be applied to the data-driven pole placement problem by choosing a reference transfer function with the desired poles. However, the zeros of the reference transfer function cannot normally be specified, because the zeros of the plant are unknown. In contrast, the data-driven pole placement method presented in [5] requires only a state space representation of the closed-loop system to specify the prescribed closed-loop performance, as shown in Section 2. This avoids the zero assignment issue that arises in the transfer function approach used in [5].

This data-driven pole placement method can, therefore, be applied to linear and time-invariant systems with measurable states. The method is briefly reviewed in Section 2. However, the capacity of the data-driven pole placement method to handle noise remains an open issue, though in [5], the total least square (TLS) method [17] was claimed to be effective. Measurement noise is one of the issues which may surely face in practical applications. Therefore, to resolve this, we introduced a prefiltering technique that reduces the effect of measurement noise in Section 3. More specifically, a finite impulse response (FIR) filter is used to prefilter the data, as this makes them easier to manipulate. In Section 4, by using the numerical example of a self-balancing robot, we discuss the effect of applying this prefiltering technique, together with the least square (LS) and TLS methods, to a self-balancing robot model. We investigate the ability of the data-driven pole placement method to produce a linearized model using numerical simulations as in [18]. A nonlinear differential equation was used to represent the dynamics of a self-balancing robot there. Moreover, we evaluate the effects by two different exciting signals, the random and the chirp exciting signal, along with TLS and prefiltering. Finally, we compare all the results for the pole placement error and identification error when two exciting signals are applied.

Notation: Let A and B be $m \times n$ and $p \times q$ matrices, respectively. Then, the Kronecker product of A and B is a $mp \times nq$ matrix, defined as follow:

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}, \tag{1}$$

where $a_{ij}(i=1,\cdots,m,\ j=1,\cdots,n)$ is the ij^{th} element of A. The vectorization of then stacks the columns into a vector:

$$\text{vec}(A) = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}, \tag{2}$$

in which a_j is the j^{th} column of A. The Frobenius norm of matrix $A \in R^{m\times n}$ is defined as

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}. \tag{3}$$

2. Data-Driven Pole Placement

In this section, we briefly review the data-driven pole placement method formulated in [5].

Consider a discrete-time linear time-invariant system and static state feedback

$$x(k+1) = Ax(k) + Bu(k) \tag{4}$$

$$u(k) = Fx(k) + v(k) \tag{5}$$

where $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input vector, $F \in \mathbb{R}^{m\times n}$ is the feedback gain, and $v \in \mathbb{R}^m$ is the external input to the closed loop system.

The data-driven pole placement problem was formulated in [5] as follows:

Problem 1. *We assume that the order of the plant n is known, state n is measurable, pair* (A,B) *is controllable but the exact value is unknown and* B *is of full rank. Let* $\Lambda = \{p_1,\cdots,p_n\}$ *be a self-conjugate set of n complex numbers in the unit circle. Given the input and output measurement data sequence* $(x_0(k),u_0(k))$ *of* (4), *find a state feedback gain* F *from the observed data* $(x_0(k),u_0(k))$ *such that* $\{\lambda_i(A+BF)\} = \Lambda$.

In a conventional approach, this problem is solved in two steps: A and B are identified from $(x_0(k),u_0(k))$, then F is derived using the standard pole placement algorithms. In contrast, the data-driven pole placement method solves the two steps simultaneously. To achieve this, the method uses the equivalency between the closed-loop system

$$x(k+1) = (A+BF)x(k) + Bv(k), \tag{6}$$

with the desired pole placement gain F and

$$x_d(k+1) = A_d x_d(k) + B_d v(k), \tag{7}$$

$$x_d(k) = Tx(k), \tag{8}$$

where (A_d, B_d) with $\lambda_i(A_d) = p_i$ is an appropriate controllable pair. This equivalency requires the nonsingular matrix T to exist. Then, we remove v

from (7) by using (5), to obtain

$$x_{\mathrm{d}}(k+1)=A_{\mathrm{d}}x_{\mathrm{d}}(k)+B_{\mathrm{d}}u(k)-B_{\mathrm{d}}Fx(k). \tag{9}$$

Then, using (8), we obtain

$$Tx(k+1)=A_{\mathrm{d}}Tx(k)+B_{\mathrm{d}}u(k)-B_{\mathrm{d}}Fx(k). \tag{10}$$

If $(x_0(k),u_0(k))(k=i,\cdots,i+N)$ satisfies (10),

$$TX_0P_1=A_{\mathrm{d}}TX_0P_2+B_{\mathrm{d}}U_0-B_{\mathrm{d}}FX_0, \tag{11}$$

where

$$X_0=\left[x_0(i)\ \ x_0(i+1)\ \cdots\ x_0(i+N)\right], \tag{12}$$

$$U_0=\left[u_0(i)\ \ u_0(i+1)\ \cdots\ u_0(i+N-1)\right], \tag{13}$$

$$P_1=\begin{bmatrix}0_{1\times N}\\ I_N\end{bmatrix},\ P_2=\begin{bmatrix}I_N\\ 0_{1\times N}\end{bmatrix}. \tag{14}$$

In [5], Equation (11) is cast into

$$S_1\begin{bmatrix}T\\ F\end{bmatrix}X_0P_1+S_2\begin{bmatrix}T\\ F\end{bmatrix}X_0P_2=B_{\mathrm{d}}U_0, \tag{15}$$

$$S_1=\begin{bmatrix}I_n & 0_{n\times m}\end{bmatrix},\ S_2=\begin{bmatrix}-A_{\mathrm{d}} & B_{\mathrm{d}}\end{bmatrix}, \tag{16}$$

and

$$F=\begin{bmatrix}f_1\\ \vdots\\ f_m\end{bmatrix}\in\mathbb{R}^{m\times n},\ T=\begin{bmatrix}t_1\\ \vdots\\ t_m\end{bmatrix}\in\mathbb{R}^{n\times n}. \tag{17}$$

Remark 1. *The system in* (7) *can be interpreted as a reference model within VRFT* (*e.g.*, [8] [9]) *and FRIT* (*e.g.*, [10] [11] [12] [13]). *The idea of eliminating* v *in* (9) *is also based on FRIT. In* [10] [11] [12], *a similar state feedback control problem has been discussed within the FRIT framework. To apply these FRIT techniques to the data-driven pole placement problem, the desired transfer function must be specified from* u *to* x, *rather than* x_{d}. *When precise values for* (A,B) *are not available, it becomes impossible to specify the zeros of the desired transfer function.*

Remark 2. *To obtain the datasets in* (12) *by applying state feedback in* (5) *to the system in* (4), *the initial feedback gain* F *should be based on* (A,B). *Hence, in Problem* 1, *the exact value of* (A,B) *is assumed to be unknown.*

When applying the property of Kronecker product $\operatorname{vec}(MND)=\left(N^T\otimes M\right)\operatorname{vec}D$ (see for example Th.2.13 in [19]) to the transpose of (15) to solve (15) for F and T, a further linear equation is derived, as follows:

$$\mathrm{X}\eta=\mathbf{U}, \tag{18}$$

where

$$\eta=\begin{bmatrix}t_1 & \cdots & t_n & f_1 & \cdots & f_m\end{bmatrix}^{\mathrm{T}}\in\mathbb{R}^{(n+m)n}, \tag{19}$$

$$\mathrm{X} = S_1 \otimes \left(X_0 P_1\right)^{\mathrm{T}} + S_2 \otimes \left(X_0 P_2\right)^{\mathrm{T}} \in \mathbb{R}^{nN \times (n+m)n}, \tag{20}$$

$$\mathbf{U} = \left(B_{\mathrm{d}} \otimes U_0^{\mathrm{T}}\right)\left(\operatorname{vec} I_m\right) \in \mathbb{R}^{nN}. \tag{21}$$

If T is nonsingular, the model coefficients can be obtained

$$A = T^{-1} A_{\mathrm{d}} T - T^{-1} B_{\mathrm{d}} F, \quad B = T^{-1} B_{\mathrm{d}}. \tag{22}$$

3. Prefiltering Noisy Measurement

When the measurement of x is contaminated by noise ε,

$$x_0(k) = x(k) + \varepsilon(k). \tag{23}$$

Then, (10) becomes

$$\begin{aligned} &T\left(x_0(k+1) - \varepsilon(k+1)\right) \\ &= A_{\mathrm{d}} T\left(x_0(k) - \varepsilon(k)\right) + B_{\mathrm{d}} u_0(k) - B_{\mathrm{d}} F\left(x_0(k) - \varepsilon(k)\right). \end{aligned} \tag{24}$$

Hence, if $\left(x_0(k), u_0(k)\right)(k = i, \cdots, i+N)$ satisfies the above equation,

$$\begin{aligned} &T\left(X_0 - E\right) P_1 \\ &= A_{\mathrm{d}} T\left(X_0 - E\right) P_2 + B_{\mathrm{d}} U_0 - B_{\mathrm{d}} F\left(X_0 - E\right) P_2, \end{aligned} \tag{25}$$

where

$$E = \left[\varepsilon(i) \quad \varepsilon(i+1) \quad \cdots \quad \varepsilon(i+N)\right]. \tag{26}$$

Then, the resulting linear equation is given as

$$\left(\mathrm{X} + \Delta\mathrm{X}\right)\eta = \mathbf{U} + \Delta\mathbf{U}, \tag{27}$$

where the effect of noise $\Delta\mathrm{X}$ has the same structure as X in (20), then

$$\Delta\mathrm{X} = -S_1 \otimes \left(E P_1\right)^{\mathrm{T}} - S_2 \otimes \left(E P_2\right)^{\mathrm{T}}, \tag{28}$$

and $\Delta\mathbf{U}$ is the equation error. Following [5], we can solve $\eta \in R^{(n+m)n}$ to (27) as a TLS problem [17], by minimizing the Frobenius norm $\left\|\left[\Delta\mathrm{X} \quad \Delta\mathbf{U}\right]\right\|_F$. It is known that the TLS solution is given as

$$\eta = -\frac{1}{V_{22}} V_{12} \tag{29}$$

based on the singular value decomposition

$$\left[\mathrm{X} \quad \mathbf{U}\right] = \left[\mathbf{U}_1 \quad \mathbf{U}_2\right] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}^{\mathrm{T}}, \tag{30}$$

where these matrices are partitioned into blocks corresponding to X and $\mathbf{U}$.

Here, we assume that there exists $M > 0$ such that

$$\frac{1}{M} \sum_{j=1}^{M} \varepsilon(i+j) \approx 0 \tag{31}$$

for all i. This means that when $N > M$,

$$E P_1 \Phi \approx 0, \; E P_2 \Phi \approx 0 \tag{32}$$

for the matrix

$$\Phi = \frac{1}{M}\begin{bmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 1 & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & 1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 \end{bmatrix} \in \mathbb{R}^{N\times(N-M+1)}, \tag{33}$$

where each column has M elements of 1. Therefore,

$$T\tilde{X}_0 P_1 = A_d T\tilde{X}_0 P_2 + B_d \tilde{U}_0 - B_d F\tilde{X}_0 P_2 \tag{34}$$

where

$$\tilde{X}_0 = X_0\Phi,\ \tilde{U}_0 = U_0\Phi. \tag{35}$$

This multiplication by Φ represents the prefiltering of signals via an Mth order FIR filter.

When the systems (4) and (7) are driven by the exciting signal, we have

$$(X_0 - E)P_1 = A(X_0 - E)P_2 + BU_0, \tag{36}$$

$$U_0 = F(X_0 - E)P_2 + V, \tag{37}$$

$$X_d = T(X_0 - E)P_2, \tag{38}$$

$$X_d P_1 = A_d X_d P_2 + B_d V, \tag{39}$$

where

$$X_d = \begin{bmatrix} x_d(i) & x_d(i+1) & \cdots & x_d(i+N) \end{bmatrix}, \tag{40}$$

$$V = \begin{bmatrix} v(i) & v(i+1) & \cdots & v(i+N-1) \end{bmatrix}. \tag{41}$$

By applying Φ to these systems, we obtain

$$X_0 P_1 \Phi = AX_0 P_2 \Phi + BU_0 \Phi, \tag{42}$$

$$U_0 \Phi = FX_0 P_2 \Phi + V\Phi, \tag{43}$$

$$X_d P_2 \Phi = TX_0 P_2 \Phi, \tag{44}$$

$$X_d P_1 \Phi = A_d X_d P_2 \Phi + B_d V \Phi. \tag{45}$$

Here, if $V\Phi \approx 0$, (34) cannot be satisfied. Hence, for all i, $V\Phi \neq 0$, that is

$$\frac{1}{M}\sum_{j=1}^{M} v(i+j) \neq 0 \tag{46}$$

must be satisfied.

4. Numerical Example: Self-Balancing Robot

We next applied the data-driven pole placement method described above to the model of a self-balancing robot [21] [20] as shown in **Figure 1**. The robot is equipped with right and left wheels driven by direct current (DC) motors whose voltages v_r and v_1 can be controlled. Because the motion dynamics can be decomposed by the input u, the control input to the robot was represented as

$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} v_r + v_1 \\ v_r - v_1 \end{bmatrix}. \tag{47}$$

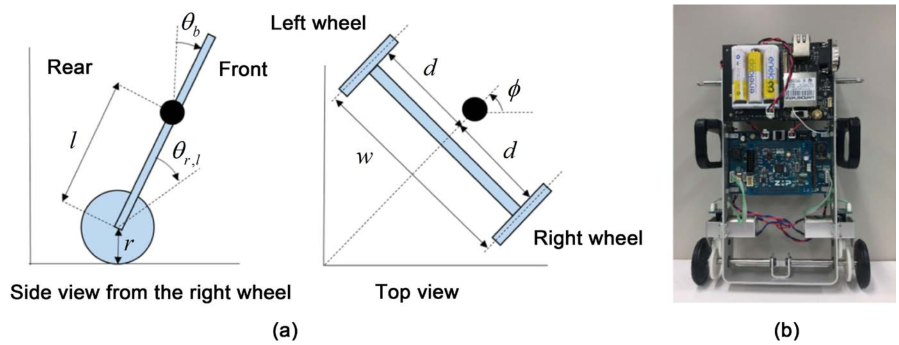

Figure 1. (a) Coordinates of the self-balancing robot; (b) photo.

We assume that the pitch angle θ_{b} and the pitch angular velocity $\dot{\theta}_{\mathrm{b}}$ of the body could be measured, as well as the angles θ_{r} and θ_{l} of the right and left wheels, and their angular velocities $\dot{\theta}_{\mathrm{r}}$ and $\dot{\theta}_{\mathrm{l}}$, respectively. We define the mean values of the right and left wheel angles θ_{r} and θ_{l}, and the yaw angle of the body as follows:

$$\theta_{\mathrm{w}} = \frac{1}{2}(\theta_{\mathrm{r}} + \theta_{\mathrm{l}}), \tag{48}$$

$$\phi = \frac{r}{w}(\theta_{\mathrm{r}} - \theta_{\mathrm{l}}), \tag{49}$$

where r is the radius of the wheel and $w = 2d$ is the distance between the two wheels.

4.1. Equation of Motion

The equation of motion for the self-balancing robot can be derived as

$$\begin{cases} J_1(\theta_{\mathrm{b}}(t)) \begin{bmatrix} \ddot{\theta}_{\mathrm{w}}(t) \\ \ddot{\theta}_{\mathrm{b}}(t) \end{bmatrix} + D_1 \begin{bmatrix} \dot{\theta}_{\mathrm{w}}(t) \\ \dot{\theta}_{\mathrm{b}}(t) \end{bmatrix} - M_{\mathrm{b}} l \sin\theta_{\mathrm{b}}(t) \begin{bmatrix} r\dot{\theta}_{\mathrm{b}}^2(t) \\ g + l\cos\theta_{\mathrm{b}}(t)\dot{\phi}^2(t) \end{bmatrix} = H_1 u(t) \\ J_2(\theta_{\mathrm{b}}(t))\ddot{\phi}(t) + D_2\dot{\phi}(t) + \left(2M_{\mathrm{b}} l^2 \sin\theta_{\mathrm{b}}(t)\cos\theta_{\mathrm{b}}(t)\right)\dot{\theta}_{\mathrm{b}}(t)\dot{\phi}(t) = H_2 u(t), \end{cases} \tag{50}$$

where

$$J_1(\theta_{\mathrm{b}}) = \begin{bmatrix} 2\left(J_{\mathrm{w}} + g_{\mathrm{r}}^2 J_{\mathrm{m}}\right) + \left(2M_{\mathrm{w}} + M_{\mathrm{b}}\right) r^2 & -2g_{\mathrm{r}}^2 J_{\mathrm{m}} + M_{\mathrm{b}} lr\cos\theta_{\mathrm{b}} \\ -2g_{\mathrm{r}}^2 J_{\mathrm{m}} + M_{\mathrm{b}} lr\cos\theta_{\mathrm{b}} & J_{\mathrm{b}} + 2g_{\mathrm{r}}^2 J_{\mathrm{m}} + M_{\mathrm{b}} l^2 \end{bmatrix},$$

$$J_2(\theta_{\mathrm{b}}) = J_{\phi} + 2\frac{d^2}{r^2}\left(J_{\mathrm{w}} + g_{\mathrm{r}}^2 J_{\mathrm{m}}\right) + 2M_{\mathrm{w}} d^2 + M_{\mathrm{b}} l^2 \sin^2\theta_{\mathrm{b}},$$

$$D_1 = 2\begin{bmatrix} d_{\mathrm{b}} + d_{\mathrm{w}} & -d_{\mathrm{b}} \\ -d_{\mathrm{b}} - d_{\mathrm{w}} & d_{\mathrm{b}} \end{bmatrix}, \quad D_2 = 2\frac{d^2}{r^2}\left(d_{\mathrm{b}} + d_{\mathrm{w}}\right),$$

$$H_1 = b_{\mathrm{v}}\begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix}, \quad H_2 = b_{\mathrm{v}}\frac{d}{r}\begin{bmatrix} 0 & 1 \end{bmatrix},$$

$$d_{\mathrm{b}} := \frac{g_{\mathrm{r}}^2 K_{\mathrm{t}} K_{\mathrm{e}}}{R_{\mathrm{m}}} + g_{\mathrm{r}} d_{\mathrm{m}}, \quad b_{\mathrm{v}} := \frac{g_{\mathrm{r}} K_{\mathrm{t}}}{R_{\mathrm{m}}}.$$

The symbols are explained in Table 1. The parameters used in the simulations were taken from [20] [21].

Table 1. Parameters of the self-balancing robot [20] [21].

$g = 9.81$	acceleration due to gravity [m/s²]
$M_b = 0.5$	mass of body [kg]
$M_w = 0.07$	mass of wheel [kg]
$r = 0.025$	radius of wheel [m]
$J_w = 8.75 \times 10^{-5}$	moment of inertia of wheel [kg·m²]
$w = 2d = 0.12$	vehicle width [m]
$l = 0.1073$	distance from wheel center to center of gravity of robot body [m]
$J_b = 6.7 \times 10^{-3}$	moment of inertia of body (pitch) [kg·m²]
$J_\phi = 6 \times 10^{-4}$	moment of inertia of body (yaw) [kg·m²]
$J_m = 1.3 \times 10^{-4}$	moment of inertia of DC motor [kg·m²]
$R_m = 0.035$	resistance of DC motor [Ω]
$K_e = 0.02$	electromotive force constant of DC motor [V·s/rad]
$K_t = K_e$	torque constant of DC motor [N·m/A]
$g_r = 30$	gear ratio
$d_m = 0.0022$	coefficient of friction between wheel and DC motor
$d_w = 0$	coefficient of friction between wheel and floor

4.2. Linear Model and Feedback Gain

We linearized the equations of motion (50) around equilibrium states $\theta_w = 0$, $\theta_b = 0$, $\phi = 0$, $\dot{\theta}_w = 0$, $\dot{\theta}_b = 0$, $\dot{\phi} = 0$, and $u = 0$. Then, under the assumption that $\sin\theta_b(t) \approx \theta_b(t)$, $\cos\theta_b(t) \approx 1$, $\sin^2\theta_b(t) \approx 0$, $\theta_b^2(t) \approx 0$, $\phi^2(t) \approx 0$, and $\sin\theta_b(t)\cos\dot{\theta}_b(t)\dot{\phi}(t) \approx 0$, the linearized equations of motion can be derived as

$$\begin{cases} J_1\ddot{x}_a(t) + D_1\dot{x}_a(t) + K_1x_a(t) = H_1u(t) \\ J_2\ddot{x}_b(t) + D_2\dot{x}_b(t) = H_2u(t), \end{cases} \tag{51}$$

where

$$x_a(t) := \begin{bmatrix} \theta_w(t) \\ \theta_b(t) \end{bmatrix}, \; x_b(t) := \phi(t), \tag{52}$$

$$J_1 = J_1(0),\, J_2 = J_2(0),\, K_1 = M_b lg\begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix}. \tag{53}$$

By defining the state vector

$$x_1(t) = \begin{bmatrix} x_a(t) \\ \dot{x}_a(t) \end{bmatrix} = \begin{bmatrix} \theta_w(t) \\ \theta_b(t) \\ \dot{\theta}_w(t) \\ \dot{\theta}_b(t) \end{bmatrix}, \quad x_2(t) = \begin{bmatrix} x_b(t) \\ \dot{x}_b(t) \end{bmatrix} = \begin{bmatrix} \phi(t) \\ \dot{\phi}(t) \end{bmatrix}, \tag{54}$$

the linear state space model can be derived as

$$\begin{cases} \dot{x}_1(t) = A_{c1}x_1(t) + B_{c1}u_1(t), \\ \dot{x}_2(t) = A_{c2}x_2(t) + B_{c2}u_2(t), \end{cases} \tag{55}$$

where

$$A_{c1}=\begin{bmatrix}0_{2\times2} & I_2\\ -J^{-1}K_1 & -J^{-1}D_1\end{bmatrix},\quad B_{c1}=\begin{bmatrix}0_{2\times1}\\ b_v J_1^{-1}\begin{bmatrix}1\\-1\end{bmatrix}\end{bmatrix},$$

$$A_{c2}=\begin{bmatrix}0 & 1\\ 0 & -J_2^{-1}D_2\end{bmatrix},\quad B_{c2}=\begin{bmatrix}0\\ b_v\dfrac{d}{r}J_2^{-1}\end{bmatrix}.$$

Then, the feedback can be independently designed as

$$u_1=F_1x_1+v_1,\ u_2=F_2x_2+v_2. \tag{56}$$

Note that this can be more succinctly represented as

$$\dot{x}(t)=A_cx(t)+B_cu(t),\ u(t)=Fx(t),\ x(t)=\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}, \tag{57}$$

$$A_c=\begin{bmatrix}A_{c1} & 0_{4\times2}\\ 0_{2\times4} & A_{c2}\end{bmatrix},\ B_c=\begin{bmatrix}B_{c1} & 0_{4\times1}\\ 0_{2\times1} & B_{c2}\end{bmatrix},\ F=\begin{bmatrix}F_1 & 0_{1\times2}\\ 0_{1\times4} & F_2\end{bmatrix}. \tag{58}$$

When the parameters in **Table 1** are used and the sampling period is $h=0.1\,\text{s}$, the discrete-time model after discretizing (55) is

$$\begin{cases}x_1(k+1)=A_1x_1(k)+B_1u_1(k),\\ x_2(k+1)=A_2x_2(k)+B_2u_2(k),\end{cases} \tag{59}$$

where

$$A_1=\begin{bmatrix}1 & 0.1719 & 0.0226 & 0.0830\\ 0 & 1.1722 & 0.0113 & 0.0944\\ 0 & 3.5363 & 0.1388 & 1.0332\\ 0 & 3.5299 & 0.1386 & 1.0336\end{bmatrix},\ B_1=\begin{bmatrix}0.0641\\ -0.0094\\ 0.7131\\ -0.1148\end{bmatrix},$$

$$A_2=\begin{bmatrix}1 & 0.0113\\ 0 & 0.0001\end{bmatrix},\ B_2=\begin{bmatrix}0.0306\\ 0.3450\end{bmatrix}. \tag{60}$$

Here, we assume that the exact values of (60) are not available, but that uncertain values are available:

$$A_1=\begin{bmatrix}1 & 0.1897 & 0.0218 & 0.0844\\ 0 & 1.1900 & 0.0115 & 0.0947\\ 0 & 3.9115 & 0.1408 & 1.0489\\ 0 & 3.9151 & 0.1407 & 1.0492\end{bmatrix},\ B_1=\begin{bmatrix}0.0648\\ -0.0095\\ 0.7115\\ -0.1165\end{bmatrix},$$

$$A_2=\begin{bmatrix}1 & 0.0103\\ 0 & 0.0001\end{bmatrix},\ B_2=\begin{bmatrix}0.0310\\ 0.3450\end{bmatrix}. \tag{61}$$

The coefficients can be derived from J_1, J_2, with an assumed uncertainty of 10%. By applying linear quadratic optimal control theory to (61), the desired closed-loop pole locations can be chosen as

$$\lambda(A_1+B_1F_1)\in\Lambda_1=\{6.0355\times10^{-5}, 0.5253, 0.5745, 0.7630\}, \tag{62}$$

$$\lambda(A_2+B_2F_2)\in\Lambda_2=\{6.0426\times10^{-5}, 0.7835\}, \tag{63}$$

and the initial feedback gains needed to obtain datasets for the data-driven pole placement as

$$F_1 = [1.5216\ 124.181\ 2.3915\ 18.3089],\ F_2 = [-6.2764 - 0.0646]. \quad (64)$$

4.3. Comparison of Methods

Next, simulations were conducted and comparisons were made from the obtained results when using different methods and exciting signals.

Measurement noise was prepared with the Gaussian distribution $N(0,\sigma^2)$, where $\sigma^2 = 1.0\times10^{-3}$, 1.0×10^{-4} and 1.0×10^{-4} in θ_{w}, $\dot{\theta}_{\mathrm{b}}$, and ϕ, respectively. This is shown in Figure 2(a). We used the random exciting signal v shown in Figure 3(a) and the linear chirp signal $v(k)$ shown in Figure 3(b) with the uniform distribution $v_1(k) \sim U(-0.5, 0.5)$ and $v_2(k) \sim U(-0.1, 0.1)$. We set the order of the prefilter Φ (33) as $M = 6$. After prefiltering, the measurement noise in θ_{w}, $\dot{\theta}_{\mathrm{b}}$, and ϕ was reduced, as shown in Figure 2(b). The prefiltered exciting signals were shown in Figure 3(b) and Figure 3(d). It can be seen that the exciting signals v were not eliminated by prefilter Φ, but that the high-frequency elements were reduced.

A closed-loop response in the presence of measurement noise by state feedback (56), with initial gain (64), is shown in Figure 4. The response to the random exciting signal and the chirp exciting signal are shown in Figure 4(a) and Figure 4(b), respectively. Of particular note is that the responses of θ_{b}, $\dot{\theta}_{\mathrm{w}}$, and $\dot{\theta}_{\mathrm{b}}$ in Figure 4(b) show the high-pass filter-like gain characteristics of the transfer function from v to x.

For comparison, the dataset for the data-driven pole placement was chosen as $\{(x_0(k), u_0(k))\}_{k=50,\cdots,450}$ where $i = 50$ and $N = 400$.

To evaluate the obtained pole placement gain $\tilde{F}$, we introduced an accuracy measurement that takes the largest absolute difference in value between each eigenvalue of $A_i + B_i\tilde{F}_i$ and the corresponding $p_j \in \Lambda_i$,

$$\delta\lambda(A_{\mathrm{d}i}) := \max\left\{\left|\lambda_j\left(A_i + B_i\tilde{F}_i\right) - p_j\right| p_j \in \Lambda_i\right\}. \quad (65)$$

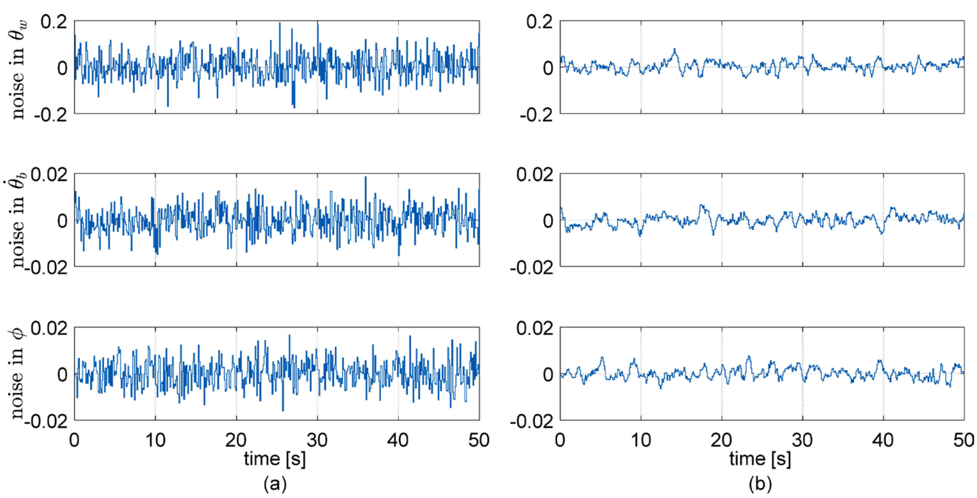

Figure 2. (a) Measurement noise; (b) Prefiltered measurement noise.

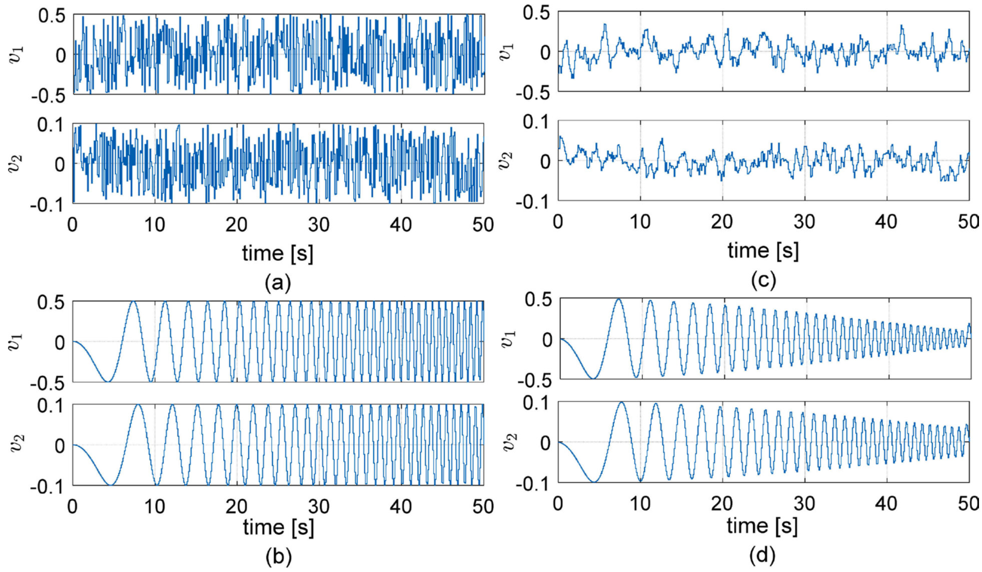

Figure 3. Exciting signal v (a) random, (b) chirp, (c) prefiltered random, (d) prefiltered chirp.

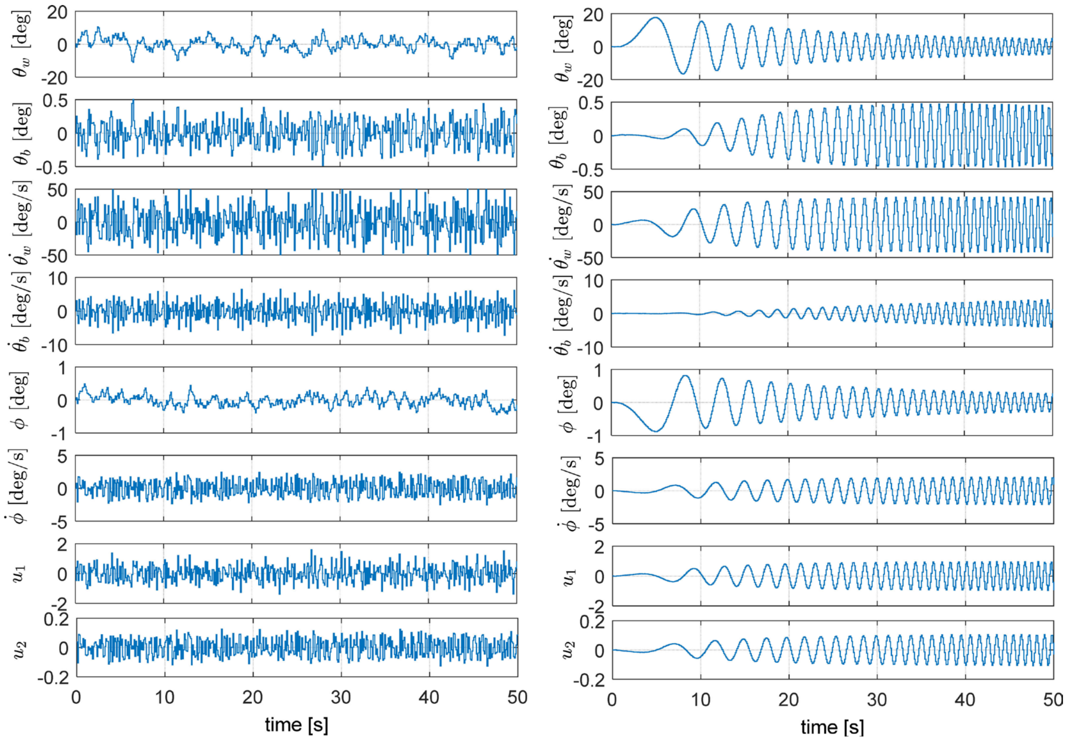

Figure 4. Closed-loop response by an initial state feedback via (a) random exciting signal v, (b) chirp exciting signal v.

To evaluate the obtained model $(\tilde{A}, \tilde{B})$, the following identification errors were used:

$$\Delta A_i := \left\| \tilde{A}_i - A_i \right\|, \Delta B_i = \left\| \tilde{B}_i - B_i \right\|, \tag{66}$$

$$\delta\lambda(A_i) := \max\left\{ \left| \lambda_j(A_i) - \lambda_j(\tilde{A}_i) \right| \right\}. \tag{67}$$

The eigenvalues λ_j were sorted by magnitude using the MATLAB command "sort". This further sorts elements of equal magnitude by the phase angle on the interval $(-\pi,\pi]$. The impulse response $G(z)=(zI-\tilde{A})^{-1}\tilde{B}$ was used to evaluate the model obtained, as follows:

$$\Delta G_i := \sqrt{\sum_{k=0}^{10}\left\|x_{\tilde{A}_i,\tilde{B}_i}(k)-x_{A_i,B_i}(k)\right\|^2}, \tag{68}$$

where $x_{\tilde{A}_i,\tilde{B}_i}$ and x_{A_i,B_i} are the impulse responses of $G_i(z):=(zI-\tilde{A}_i)^{-1}\tilde{B}_i$ and $G(z):=(zI-A)^{-1}B$, respectively.

From the perspective of system control, smaller is better, particularly in the case of $\delta\lambda(A_{di})$, $\delta\lambda(A_i)$, and ΔG_i. The following key results were contrastively found in Table 2:

1) The initial model and feedback gain were affected by uncertainty: The model errors and pole placement errors are shown in Table 2 (initial).

2) The results when using the LS method to solve linear Equation (27) for noiseless data are shown in Table 2(a). All errors were reasonably small, confirming that the data-driven method performs well when the measurement data $(x_0(k),u_0(k))$ are noiseless.

3) The results when using the LS method to solve linear Equation (27) for noisy data are shown in Table 2(b). All errors became larger when noise was added, suggesting that LS analysis is inadequate when the measurement data are contaminated by noise.

4) The results when using the TLS method to solve linear Equation (27) are shown in Table 2(c). The errors were significantly smaller than those reported in [5], using the LS method.

5) The results when applying prefiltering (PF) and using the TLS method to

Table 2. Comparison of errors.

	(initial)	(a)	(b)	(c)	(d)	(e)
noise	-	noiseless	noisy	noisy	noisy	noisy
method	-	LS	LS	TLS	TLS + PF	TLS + PF
exciting sig.	-	Random	Random	Random	Random	Chirp
$\delta\lambda(A_{d1})$	0.2426	0.0007	0.4597	0.1367	**0.0466**	1.2530
ΔA_1	0.5317	0.0016	36.295	1.6678	1.8763	17.246
ΔB_1	0.0025	0.0000	0.3400	0.0481	0.0415	0.2932
$\delta\lambda(A_1)$	0.0511	0.0000	0.3920	0.0194	**0.0177**	0.4695
ΔG_1	42.333	0.0082	629.67	44.718	**29.324**	106.04
$\delta\lambda(A_{d2})$	0.0029	0.0000	0.0092	0.0024	**0.0007**	0.0017
ΔA_2	0.0001	0.0000	0.0288	0.0064	0.0005	0.0007
ΔB_2	0.0004	0.0000	0.0031	0.0002	0.0002	0.0002
$\delta\lambda(A_2)$	0.0001	0.0000	0.0090	0.0012	0.0004	0.0001
ΔG_2	0.0036	0.0002	0.0525	0.0073	0.0019	0.0019

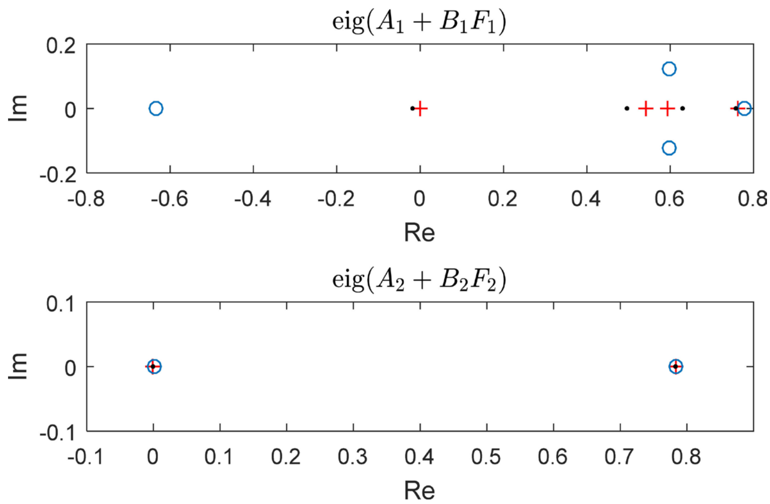

Figure 5. Comparison of pole locations ("+" indicates the desired poles, "." those obtained by the random exciting signal and "o" those obtained by the chirp exciting signal.).

solve linear Equation (27) are shown in **Table 2(d)**. The prefilter further reduced the errors, in particular, the pole placement error $\delta\lambda(A_{d1})$ and the impulse response error ΔG_1.

6) The results when applying PF and using the TLS method to solve the linear Equation (27), but with v as the chirp signal, are shown in **Table 2(e)**. No significant improvement in error rates was found with respect to A_2 when using the chirp exciting signal. However, the errors with respect to A_1 became significantly worse than when a random exciting signal was used. This was assumed to be because A_1 has an unstable eigenvalue of 1.7838. We conclude that a random exciting signal is more appropriate than a chirp exciting signal when using data-driven methods.

Finally, we compare the pole locations obtained as shown in **Figure 5**. As can be seen, a better performance was achieved when using the random exciting signal.

5. Conclusion

In this study, we evaluated the different approaches reducing the effect of measurement noise in data-driven pole placement methods for deriving a state space model and pole placement state feedback. Using numerical simulations of a self-balancing robot, which is a nonlinear system, we demonstrated the important role that prefiltering can play in reducing the interference caused by noise. Again using numerical simulation, we compared the use of two exciting signals: a random signal and a chirp signal. The use of a random exciting signal was found to be more effective with our proposed method. Further developments are needed in the methods used to cope with noise. A method such as that used in [9] may be appropriate for use in practical applications where noise is present, and adaptive control based on real-time updating [22] is a future promising approach.

Acknowledgements

This work was partially supported by JSPS KAKENHI Grant Number 16H04385.

References

[1] Wonham, W.M. (1967) On Pole Assignment in Multi-input Controllable Linear Systems. *IEEE Transaction on Automatic Control*, **12**, 660-665. https://doi.org/10.1109/TAC.1967.1098739

[2] Ackermann, J.E. (1977) On the Synthesis of Linear Control Systems with Specified Characteristics. *Automatica*, **13**, 89-94.

[3] Kimura, H. (1975) Pole Assignment by Gain Output Feedback. *IEEE Transaction Automatic Control*, **20**, 509-516. https://doi.org/10.1109/TAC.1975.1101028

[4] Hikita, H., Koyama, S. and Miura, R. (1975) The Redundancy of Feedback Gain Matrix and the Derivation of Low Feedback Gain Matrix in Pole Assignment. *The Society of Instrument and Control Engineers*, **11**, 556-560. (In Japanese)

[5] Yamamoto, S., Okano, Y. and Kaneko, O. (2016) A Data-driven Pole Placement Method Simultaneously Identifying a State Space Model. *Transaction of the Institute of Systems, Control and Information Engineers*, **29**, 275-284. (In Japanese)

[6] Hou, Z.S. and Wang, Z. (2013) From Model-Based Control to Data-Driven Control: *Survey. Classification and Perspective*, *Information Sciences*, **235**, 3-35.

[7] Safonov, M.G. and Tsao, T.C. (1997) The Unfalsified Control Concept and Learning. *IEEE Transactions on Automatic Control*, **42**, 843-847. https://doi.org/10.1109/9.587340

[8] Campi, M.C., Lecchini, A. and Savaresi, S.M. (2002) Virtual Reference Feedback Tuning: A Direct Method for the Design of Feedback Controllers. *Automatica*, **38**, 1337-1346.

[9] Sala, A. and Esparza, A. (2005) Extensions to "Virtual Reference Feedback Tuning: A Direct Method for the Design of Feedback Controllers". *Automatica*, **41**, 1473-1476.

[10] Souma, S., Kaneko, O. and Fujii, T. (2004) A New Method of a Controller Parameter Tuning Based on Input-output Data-Fictitious Reference Iterative Tuning. *Proceedings of the 2nd IFAC Workshop on Adaptation and Learning in Control and Signal Processing*, **37**, 789-794.

[11] Matsui, Y., Akamatsu, S., Kimura, T., Nakano, K. and Sakurama, K. (2011) Fictitious Reference Iterative Tuning for State Feedback Control of Inverted Pendulum with Inertia Rotor. *SICE Annual Conference*, Tokyo, 13-18 September 2011, 1087-1092.

[12] Matsui, Y., Akamatsu, S., Kimura, T., Nakano, K. and Sakurama, K. (2014) An Application of Fictitious Reference Iterative Tuning to State Feedback, *Electronics and Communications in Japan*, **97**, 1-11. https://doi.org/10.1002/ecj.11506

[13] Kaneko, O. (2015) The Canonical Controller Approach to Data-Driven Update of State Feedback Gain. *Proceedings of the 10th Asian Control Conference 2015 (ASCC 2015)*, Kota Kinabalu, 31 May-3 June 2015, 2980-2985. https://doi.org/10.1109/ASCC.2015.7244744

[14] Van Heusden, K., Karimi, A. and Soderstrom, T. (2011) Extensions to "On Identification Methods for Direct Data-Driven Controller Tuning". *International Journal of Adaptive Control and Signal Processing*, **25**, 448-465. https://doi.org/10.1002/acs.1213

[15] Kaneko, O., Miyachi, M. and Fujii, T. (2008) Simultaneous Updating of a Model and a Controller Based on the Data-Driven Fictitious Controller. *47th IEEE Conference on Decision and Control*, Cancun, 9-11 December 2008, 1358-1363.

[16] Kaneko, O., Miyachi, M. and Fujii, T. (2011) Simultaneous Updating of Model and

Controller Based on Fictitious Reference Iterative Tuning. *SICE Journal of Control, Measurement, and System Integration*, **4**, 63-70. https://doi.org/10.9746/jcmsi.4.63

[17] Markovsky, I. and Huffel, S.V. (2007) Overview of Total Least-Squares Methods. *Signal Processing*, **87**, 2283-2302.

[18] Shwe, P.E.E. and Yamamoto, S. (2016) Data-Driven Method to Simultaneously Obtain a Linearized State Space Model and Pole Placement Gain. *Proceedings of the 3rd Multi Symposium on Control Systems*, Nagoya, 7-10 March 2016, 3B3-2.

[19] Brewer, J. (1978) Kronecker Products and Matrix Calculus in System Theory. *IEEE Transactions on Circuits and Systems*, **25**, 772-781. https://doi.org/10.1109/TCS.1978.1084534

[20] ZMP Inc. Stabilization and Control of Stable Running of the Wheeled Inverted Pendulum, Development of Educational Wheeled Inverted Robot e-nuvo WHEEL ver.1.0. http://www.zmp.co.jp/e-nuvo/

[21] Nomura, T., Kitsuka, Y., Suemitsu, H. and Matsuo, T. (2009) Adaptive Back Stepping Control for a Two-Wheeled Autonomous Robot. *Proceedings of ICCAS-SICE*, Fukuoka, 18-21 August 2009, 4687-4692.

[22] Shwe, P.E.E. and Yamamoto, S. (2016) Real-Time Simultaneously Updating a Linearized State-Space Model and Pole Placement Gain. *Proceedings of SICE Annual Conference*, Tsukuba, 20-23 September 2016, 196-201.