

# モデル規範型ロバストデッドビート制御

－フィードフォワード制御によるリップル回避－

## Model Reference Robust Deadbeat Control

－ Avoiding of Ripples by Feedforward Control －

青 木 立<sup>1)</sup>

Tatsu AOKI<sup>1)</sup>

**Abstract:**One of methods for the design of a digital controller is that first the transfer function of a plant with zero-order hold is modeled as a discrete-time system, and next, the controller is directly designed in the  $z$ -plane. The deadbeat controller is studied intense in such methods. The plant can follow the step signal by the deadbeat control in one or more sampling periods. However, the deadbeat control is very sensitive to the change of the plant parameters and this variation may lead to oscillatory response or ripples. In this paper the deadbeat control that can avoid these ripples is proposed. The deadbeat control has been generally realized by the feedback control on the plant output. Since the poles of the control system change by the change of the plant parameters, the control system becomes unstable, and ripples are induced. Thus, it is considered that a ripple-free deadbeat control may be realized only by a feedforward control instead of a feedback control. Since a feedforward control cannot adapt to the change of plant parameters, it is necessary to add a feedback control to give robust performance. A model reference adaptive control is considered for this. The proposed control method is that an adaptive control law is deleted in a model reference adaptive control. As an illustration, second-order systems are considered. The effectiveness of the proposed method was verified on set-point and servo control.

Keywords: Deadbeat control, Feedforward control, Robust control, Ripple-free control, Model reference adaptive control

### 1. はじめに

ディジタル制御では、制御器は $z$ 変換などを用いた離散時間系で表現されるが、制御対象は連続時間系で表現される。この両時間系の間ではサンプラーと0次ホールドを介して信号が授受される。連続時間系である制御対象の出力をサンプラーによりサンプルすることにより離散時間系の信号に変換する。離散時間で表現された制御器ではこの信号から制御アルゴリズムに基づいた信号を出力する。この信号は0次ホールドにより連続時間系の信号に変換され、制御対象へ出力される。0次ホールドはサンプリング点間の信号について新たな信号がサンプルされるまでその値を保持する。このように制御システムでは両時間系が混在するため、ディジタル制御器の設計手法は大きく分けて二通りある。一つの設計法は、連続時間系において制御器を設計し、それを離散化する手法である。他の設計法は、連続時間系の制御対象をまず離散化し、離散化した制御対象に基づいて直接、離散時間系の制御器を設計する手法である。両者の設計方法にはそれぞれメリット、デメリットがある。前者のメリットは、従来の連続時間系での制御器の設計手法をそのまま適用できることである。しかし、デメリットとしてサンプリング及び0次ホールドを考慮せずに設計しているため、連続時間系で設計された制御器を離散化した場合、非常に短いサンプリング時間を設定しないと所望の制御性能が得られない。一方、離散時間系

において制御器を直接設計する手法のメリットはサンプリング及び0次ホールドを設計当初より考慮しているため、各サンプリング時点における制御対象の挙動は数式どおり制御できる。しかし、デメリットとして制御対象が連続時間系であることを考慮していないため、サンプリング点間における制御対象の挙動は全く制御できない。これはサンプリング点間では制御器の出力は0次ホールドにより一定であり、オープンループ制御になるためである。ところで、離散時間系では有限のサンプリング時間で制御対象を目標に一致させるデッドビート制御手法がある。この手法はサンプリング及び0次ホールドという概念がない連続時間系の制御器には実現できない手法である。デッドビート制御により制御対象は1、ないしは、数サンプリング周期で目標に到達する。しかし、制御対象のパラメータが変動するとデッドビート制御が実現できなくなるだけではなく、サンプリング点間において制御対象に振動、すなわち、リップルが発生する。この問題を解決するため国内外において種々の手法が研究されている [1][2]。

本研究の目的は、一般に広く利用できるように初等的な制御理論のみでリップルフリーかつロバストなデッドビート制御が可能な手法の開発である。2章ではリップルを回避するためフィードフォワード制御のみによる制御手法を提案する。3章では提案手法のロバスト性を確保するため、モデル規範型制御を併用した手法について述べ、シミュレーションにより提案手法の有効性を検証する。4章では結論を述べる。

1) 東京都立産業技術高等専門学校 ものづくり工学科 電気電子工学コース

## 2. フィードフォワード制御によるデッドビート制御

### 2.1 制御器の導出

図 1 に 0 次ホールド (ZOH) を含めたフィードフォワード制御系を示す。制御対象を

$$P(s) = \frac{N_p(s)}{D_p(s)} \quad (1)$$

とし、0 次ホールドにより離散化した伝達関数を

$$P(z) = \frac{N_p(z)}{D_p(z)} \quad (2)$$

とする。ここで、制御器  $C(z)$  を

$$C(z) = \frac{D_p(z)}{N_p(z)} \frac{1}{z^n} \quad (3)$$

とした場合、制御系の伝達関数は

$$\begin{aligned} \frac{Y(z)}{R(z)} &= C(z)P(z) \\ &= \frac{D_p(z)}{N_p(z)} \frac{1}{z^n} \frac{N_p(z)}{D_p(z)} \\ &= \frac{1}{z^n} \end{aligned} \quad (4)$$

になり、デッドビート制御が実現できる。なお、 $n$  は式 (3) に示す制御器が実現可能なように分母の次数が分子の次数に一致するように設定する。ところで、プラントを 0 次ホールドにより離散化した結果、式 (5) に示すように負の零点が発生する場合がある。

$$P(z) = \frac{N_p^+(z)N_p^-(z)}{D_p(z)} \quad (5)$$

ここで、 $N_p^+(z)$ 、 $N_p^-(z)$  は、それぞれ、正の零点、負の零点のみを含む多項式とする。式 (6) に示す制御器

$$C(z) = \frac{D_p(z)}{N_p^+(z)N_p^-(z)} \frac{1}{z^n} \quad (6)$$

を使用した場合、

$$\begin{aligned} \frac{Y(z)}{R(z)} &= C(z)P(z) \\ &= \frac{D_p(z)}{N_p^+(z)N_p^-(z)} \frac{1}{z^n} \frac{N_p^+(z)N_p^-(z)}{D_p(z)} \\ &= \frac{1}{z^n} \end{aligned} \quad (7)$$

となり、デッドビート制御が実現できる。しかし、式 (6) に示す制御器には負の極があり、制御器の出力  $u(k)$  はサンプリング周期毎に正負にスイッチングする。この結果、連続時間系のプラントは各サンプリング時点ではデッドビート制御が実現しているがそれ以外の時間では振動する。このリップルを避けるため、負の零点を消去しない制御器

$$C(z) = \frac{D_p(z)}{N_p^+(z)} \frac{1}{\gamma z^n} \quad (8)$$

を考える。ここで、 $\gamma$  は  $N_p^-(z)$  の DC ゲインとする。この  $\gamma$  により制御系全体の DC ゲインが 1 になる。また、 $n$  は前出同様、制御器が実現可能なように設定する。

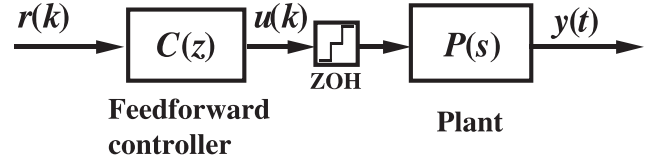


Fig. 1 Feedforward controller

Table 1 Prant parameters

Plant	$\zeta$	$\omega$ rad/s	Gain	$a$	$b$	$P_z$	Zeros
$P_m(s)$	0.3	$2\pi 10$	1.0	-	-	50	0.99501
$P_1(s)$	0.4	$2\pi 11$	0.9	-	-	40	0.99601
$P_2(s)$	0.3	$2\pi 8$	1.1	-	-	60	0.99402
$P_3(s)$	0.9	$2\pi 5$	1.5	-	-	60	0.99402
$P_4(s)$	0.3	$2\pi 10$	1.0	-	-	-	-0.99874
$P_5(s)$	-	-	1.0	100	100	-	-0.99667
$P_6(s)$	0.3	$2\pi 10$	1.0	100	100	-	-9.87180
							-0.99725
							-0.10074

このときの制御系全体の伝達関数は

$$\begin{aligned} \frac{Y(z)}{R(z)} &= C(z)P(z) \\ &= \frac{D_p(z)}{N_p^+(z)} \frac{1}{z^n} \frac{N_p^+(z)N_p^-(z)}{D_p(z)} \\ &= \frac{N_p^-(z)}{\gamma z^n} \end{aligned} \quad (9)$$

となり、 $N_p^-(z)$  に依存するもののデッドビート制御が実現できる。

### 2.2 提案制御手法の有効性の検証

MATLAB/Simulink を用いて制御対象のパラメータ変動に関する提案手法のロバスト性を検証する。

#### 2.2.1 制御対象

以下の 2 次系について検証する。

$$P_m(s) = \frac{\left(\frac{s}{p_z} + 1\right) \omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2} \quad (10)$$

表 1 に各パラメータを示し、サンプリング周期  $T$  は 0.1ms に設定する。ここで、 $-P_z$  は連続時間系における零点、 $P_m(s)$  はノミナルプラントとし、 $P_1(s)$  から  $P_3(s)$  は  $P_m(s)$  からパラメータが変動したプラントとする。 $P_1(s)$  から  $P_3(s)$  は離散時間系において正の零点のみを有するプラントである。一方、 $P_4(s)$  から  $P_6(s)$  は離散時間系において負の零点のみを有するプラントである。

$$P_4(s) = \frac{\omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2} \quad (11)$$

$$P_5(s) = \frac{b}{s + a} \frac{1}{s} \quad (12)$$

$$P_6(s) = P_4(s)P_5(s) \quad (13)$$

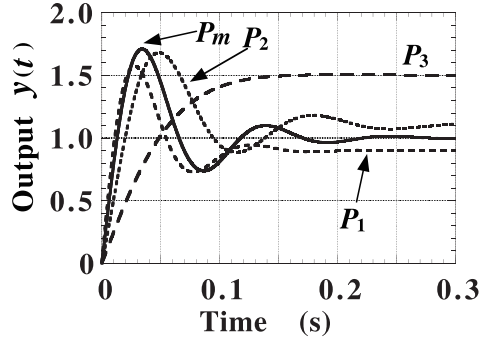


Fig. 2 Step responses of plants

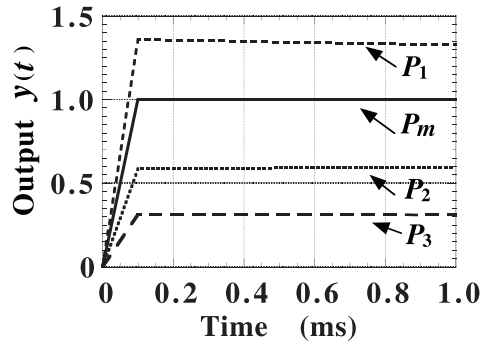
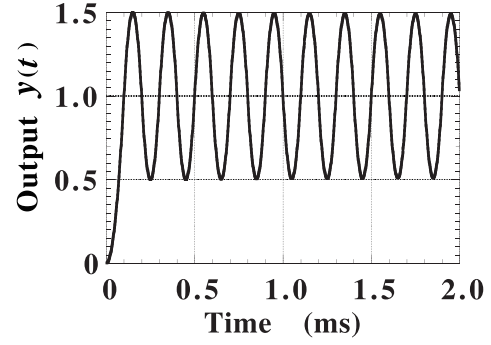
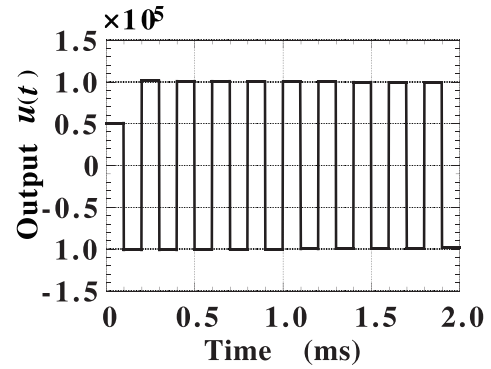


Fig. 3 Step responses by using feedforward controllers



(a) Step response



(b) Controller output

Fig. 4 The controller with the negative pole

図2にプラント単体のステップ応答を示す。図3に式(3)に示す制御器を使用したステップ応答を示す。ノミナルプラント  $P_m(z)$  の場合のみ一定値1になっており、パラメータ変動を抑制するフィードバック制御を併用する必要がある。次に、離散時間系における負の零点を持つプラント  $P_4(z)$  について考える。図4(a)及び(b)に式(3)に示す制御器を使用したステップ応答、制御器出力  $u(t)$  をそれぞれ示す。 $u(t)$  がサンプリング周期毎にスイッチングしているため、プラント出力  $y(t)$  にリップルが発生している、しかし、各サンプリング時点での値は目標値1になっている。次に、式(8)に示す制御器を使用した場合のステップ応答及び制御器出力  $u(t)$  を図5(a)及び(b)にそれぞれ示す。DCゲインが1になるように調整した  $N_p^-(z)$  のステップ応答になり、デッドビート制御が実現できている。さらに、プラントの極に0が含まれる  $P_5(s)$ ,  $P_6(s)$  について考える。表1に示すように  $P_5(s)$ ,  $P_6(s)$  は負の零点のみを有し、それぞれの伝達関数の分子は以下になる。

$$N_5^-(z) = (4.983z + 4.967) \times 10^{-7} \quad (14)$$

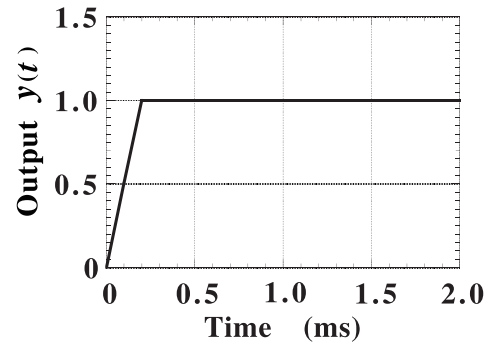
$$N_6^-(z) = (1.640z^3 + 17.00z^2 + 17.95z + 1.627) \times 10^{-12} \quad (15)$$

式(8)に示す制御器を用いると伝達関数は式(9)より

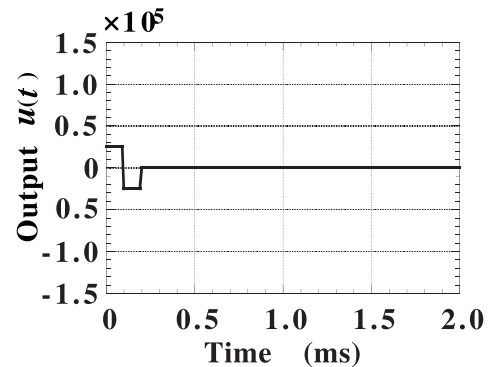
$$\frac{Y(z)}{R(z)} = \frac{N_5^-(z)}{\gamma_5 z^2} \quad (16)$$

$$\frac{Y(z)}{R(z)} = \frac{N_6^-(z)}{\gamma_6 z^4} \quad (17)$$

となる。従って、プラントの極に0が含まれる場合でもデッドビート制御が可能である。



(a) Step response



(b) Controller output

Fig. 5 The controller without the negative pole

### 3. モデル規範型制御を併用したデッドビート制御

#### 3.1 制御系の構成

図 6 に制御系の構成を示す。モデル規範型制御 (Model Reference Adaptive Control, MRAC) から適応機能を削除した制御系である [3]。\$P\_m(z)\$, \$P(s)\$ は、それぞれ、モデルと実際のプラントとし、\$d(t)\$ は外乱とする。フィードフォワード制御によりデッドビート制御を実現し、プラントパラメータによる出力誤差をゲイン \$K(z)\$ を介してフィードバックする。ここで、プラントの誤差を \$\Delta\_p\$、\$P(z) = P\_m(z)(1 + \Delta\_p)\$ とすると

$$\frac{Y(z)}{R(z)} = \frac{C(z)P_m(z)(1 + \Delta_p)(1 + K(z)C(z)P_m(z))}{1 + K(z)C(z)P_m(z)(1 + \Delta_p)} \quad (18)$$

となる。\$\Delta\_p = 0\$ のとき、式 (18) から \$C(z)P\_m(z)\$ になる。さらに、図 6 に示す \$e(k)\$ については以下になる。

$$\frac{E(z)}{R(z)} = -\frac{C(z)P_m(z)\Delta_p}{1 + K(z)C(z)P_m(z)(1 + \Delta_p)} \quad (19)$$

ここでゲイン \$K(z)\$ を以下の PI 制御器とする。

$$K(z) = k_p + \frac{k_i}{z-1} \quad (20)$$

式 (20) を式 (19) に代入すると

$$\frac{E(z)}{R(z)} = -\frac{C(z)P_m(z)\Delta_p(z-1)}{z-1 + (k_p(z-1) + k_i)C(z)P_m(z)(1 + \Delta_p)} \quad (21)$$

となる。そこで、ステップ信号

$$R(z) = \frac{z}{z-1} \quad (22)$$

を入力したときの誤差 \$E(z)\$ が求まる。

$$E(z) = -\frac{C(z)P_m(z)\Delta_p z}{z-1 + (k_p(z-1) + k_i)C(z)P_m(z)(1 + \Delta_p)} \quad (23)$$

式 (21) が安定であれば \$z\$ 変換の最終値の定理により

$$\begin{aligned} e(\infty) &= \lim_{z \rightarrow 1} (z-1)E(z) \\ &= 0 \end{aligned} \quad (24)$$

となり、プラント誤差 \$\Delta\_p\$ がある場合でも誤差 \$e(k)\$ が 0 になることがわかる。次に、外乱 \$d(t)\$ の影響について考える。

$$\frac{E(z)}{D(z)} = -\frac{P_m(z)(1 + \Delta_p)}{1 + K(z)C(z)P_m(z)(1 + \Delta_p)} \quad (25)$$

ここで、式 (20) に示す PI 制御器を用いると式 (25) は

$$\frac{E(z)}{D(z)} = -\frac{P_m(z)(1 + \Delta_p)(z-1)}{z-1 + (k_p(z-1) + k_i)C(z)P_m(z)(1 + \Delta_p)} \quad (26)$$

となる。式 (26) よりステップ応答は

$$E(z) = -\frac{P_m(z)(1 + \Delta_p)z}{z-1 + (k_p(z-1) + k_i)C(z)P_m(z)(1 + \Delta_p)} \quad (27)$$

となる。式 (26) が安定であれば \$z\$ 変換の最終値の定理により

$$\begin{aligned} e(\infty) &= \lim_{z \rightarrow 1} (z-1)E(z) \\ &= 0 \end{aligned} \quad (28)$$

が得られ、外乱及びプラント誤差 \$\Delta\_p\$ がある場合でも誤差 \$e(k)\$ が 0 になることがわかる。

#### 3.2 モデル規範型制御器の特性検証

\$k\_p = 0.7\$, \$k\_i = 0.7\$ のとき、表 1 に示す \$P\_1\$ から \$P\_3\$ の制御結果を図 7 に示す。\$\Delta\_p\$ がある場合でも整定時間は多少長くなるもののデッドビート制御が実現できている。

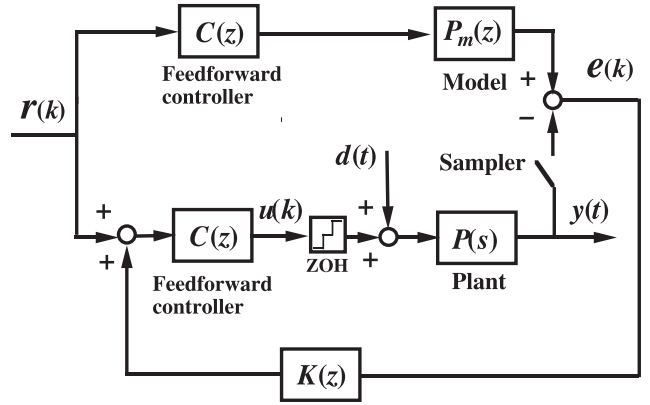
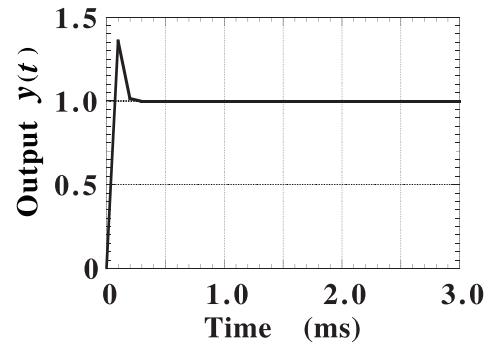
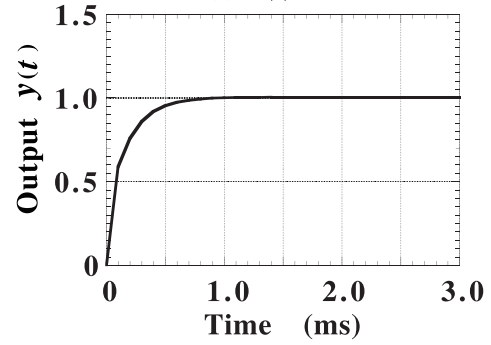


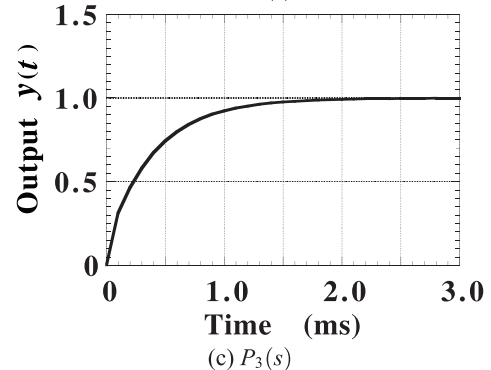
Fig. 6 Model reference controller



(a) \$P\_1(s)\$



(b) \$P\_2(s)\$



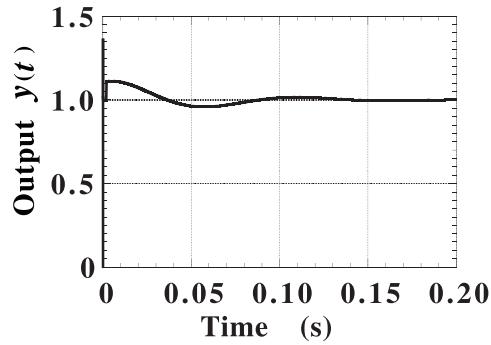
(c) \$P\_3(s)\$

Fig. 7 Step responses by using model reference controller

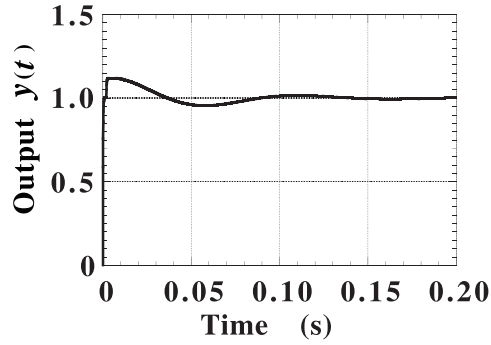
なお、図 7(a) のオーバーシュートを低減するため、あるいは、制御器の出力 \$u(t)\$ の大きさを低減するためには \$R(z)\$ を

$$R(z) = \frac{z^{r-1} + z^{r-2} + \dots + 1}{r} \frac{z}{z-1} \quad r = 1, 2, 3, \dots \quad (29)$$

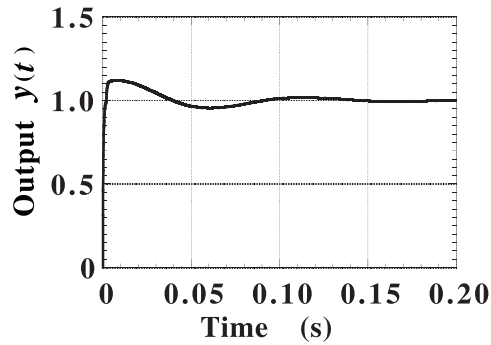
とすればよい。



(a)  $P_1(s)$



(b)  $P_2(s)$



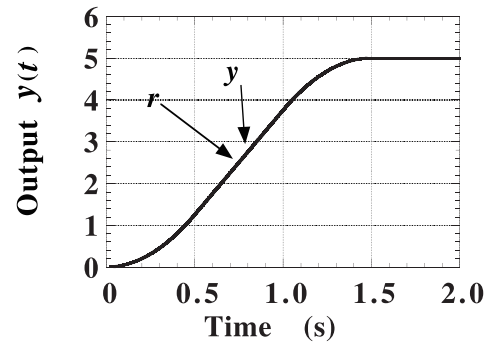
(c)  $P_3(s)$

Fig. 8 Step responses under disturbance

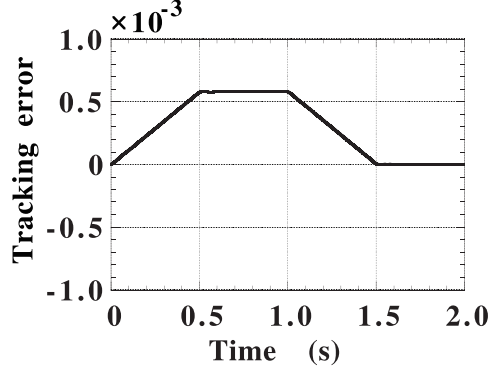
次に、図 8 に大きさ 5 のステップ状の外乱を 0.2s に加えた応答を示す。プラント誤差  $\Delta_p$  がある場合でも外乱の影響を除去できることがわかる。さらに、提案手法のサーボ性能を確認するため、台形速度パターンを積分して得られる S 字曲線を目標にした追従特性を調べる。図 9(a) に指令値  $r(t)$  と  $P_1(s)$ ,  $P_2(s)$ ,  $P_3(s)$  の出力  $y(t)$  を示す。これらの応答は目標に精度よく追従していることがわかる。図 9(b), (c), (d) にこのときのトラッキングエラーを示す。トラッキングエラーは  $C(z)P_m(z) = z^{-1}$  から入力に対する差分になるので、S 字曲線を微分した元の台形の形状になる。プラント誤差  $\Delta_p$  がある場合でも目標値の変化に追従していることがわかる。

#### 4. 結 論

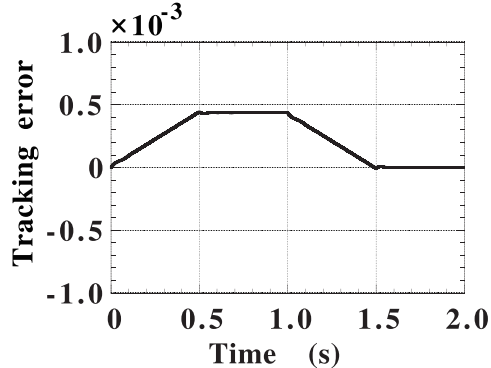
フィードフォワード制御によりデッドビート制御を実現し、モデル規範型フィードバック制御を併用することによりロバスト性を確保する簡便な制御手法を提案した。シミュレーションの結果、本手法によりロバスト性を有し、リップルフリーなデッドビート制御が可能であることを確認した。



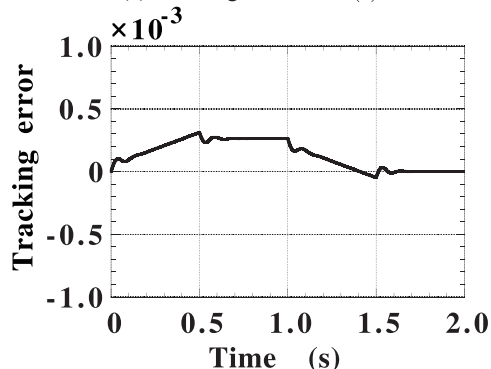
(a) Reference and plant  $P_1(s)$ ,  $P_2(s)$ ,  $P_3(s)$  outputs



(b) Tracking error on  $P_1(s)$



(c) Tracking error on  $P_2(s)$



(d) Tracking error on  $P_3(s)$

Fig. 9 Servo control by using proposed controller

#### 5. 参考文献

- [1] M. Salgadoa, D. Oyarzun, E. Silva : $H^2$  optimal ripple-free deadbeat controller design, Automatica, 43, pp. 1961–1967, 2007
- [2] Y. Funahashi, M. Yamada: Ripple-free deadbeat tracking problem, JSME Int. J. Series C, 37, 1, pp. 78–85, 1994
- [3] S. Sastry, M. Bodson: Adaptive Control – Stability, Convergence, and Robustness–, Prentice-Hall, pp. 99–156, 1989