Proceedings of the 10th Mediterranean Conference on Control and Automation - MED2002 Lisbon, Portugal, July 9-12, 2002.

OPTIMAL ℓ_2 RIPPLE-FREE DEADBEAT CONTROLLERS FOR SYSTEMS WITH TIME DELAYS

H.A.Elaydi*, **R.A. Paz**†
* Electrical and Computer Engineering Department
Islamic University of Gaza
PO Box 108, Gaza, Palestine
e-mail: helaydi@ieee.org
Fax: +972-8-286-0800
† Klipsch School of Electrical and Computer Engineering
New Mexico State University
PO Box 30001, Dept. 3-O, Las Cruces, NM 88003
e-mail: rpaz@nmsu.edu

Abstract

A ripple-free deadbeat controller for a system with time-delays is provided. Previous approaches to this problem rely on the Diophantine equation. However, until recently, solutions to the Diophantine equation would exhibit extremely bad transient responses, because the inherent freedom in the equations was difficult to manage. This approach uses an affine parametrization of solutions of the Diophantine equation. Based on this parametrization, optimal ℓ_2 controllers are obtained.

1 Introduction

We consider the problem of tracking a unit step reference signal in a deadbeat fashion for continuous, linear, time-invariant, single-input, single-output systems. We give a design procedure for a causal controller under which the output of the closed-loop system exactly coincides with the reference input after a fixed (finite) time. The design here provides an optimal ℓ_2 controller subject to a constraint on the controller order.

1.1 Digital Deadbeat Systems

The study of deadbeat error control of discrete-time systems dates back to the early 1950's [17, 4]. An excellent perspective into the long and rich history of this problem is given in [10].

The general solution to the ripple-free deadbeat problem was given by Kučera [11] and Sirisena [19], in terms of the solutions of a Diophantine equation. These approaches provide all stabilizing solutions and hold for general tracking problems in discrete-time (though not necessarily in continuous-time). For the (hybrid) control of continuous-time systems, it was found that there must, in general, be a continuous-time internal model of the (continuous-time) reference signal [20, 8]. For the problem with a unit step reference signal, this is taken into account in the zero-order hold (ZOH), and so the conditions in [19] are both necessary and sufficient for this hybrid system problem.

More recently, results for ripple-free deadbeat controllers have included solutions to multi-input, multi-output systems in a state-space format [20], solutions with constraints on the control magnitude [12], on the overshoot (or undershoot) [16] or with a robustness consideration [22]. The ℓ_2 optimal response has been considered, but in a simplified case, without time delays, and with the solution given in terms of an iterative algorithm [9, 18]. A simplified ℓ_2 solution (along with solutions of other optimal control problems) was also given in [15] in terms of LMI (instead of closed-form), and did not consider time-delayed systems.

The approach considered here provides a straightforward method for obtaining an internally stabilizing, optimal, ripple-free deadbeat controller. Optimization here is in terms the system norm performance measure ℓ_2. The additional constraint of controller order, is also taken into account. Examples are provided to show the effectiveness of the result.

1.2 System Formulation

We consider a single-input, single-output (SISO), linear time-invariant (LTI), minimal, strictly proper, continuous-time plant having the transfer function

$$\frac{y(s)}{u(s)} = \left(e^{-s\tau_d}\right) P(s) = \left(e^{-s\tau_d}\right) \frac{N_{pc}(s)}{D_{pc}(s)} \tag{1}$$

where $\tau_d \geq 0$ is the value of the pure time delay of the system, and $N_{pc}(s)$, $D_{pc}(s)$ are polynomials. This delay may arise in a number of ways in the continuous-time system, for example as a transport delay, communication delay etc. It may also be considered to arise as a computational delay associated with the time required for the computer to calculate the control value based on the present data in a hybrid control setting.

We desire to control this system in a unity feedback, hybrid control setting. This configuration is illustrated in Figure 1. In order to design a controller for this system, we desire to apply ripple-free design techniques to the control of this system.

We first obtain a discrete model of the plant that is accurate at the sampling instants, $t = kT$, (where T is the sampling period). The time delay for the system can be written

$$\tau_d = (l + \mathcal{Q})T, l - \text{integer}, 0 \leq \mathcal{Q} < 1.$$

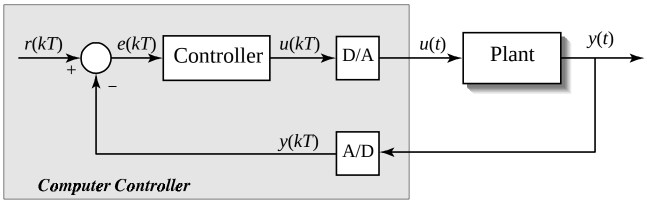

Figure 1: Closed-Loop Hybrid Control System

The transfer function may now be given in terms of the delay element q[1]

$$\frac{y(q)}{u(q)} = [P]_{Hm}(q) = \frac{N_p(q)}{D_p(q)} \tag{2}$$

where

$$[P]_{Hm}(q) = q^l (1-q)\, \mathcal{Z}\left\{ \frac{e^{-\mathcal{Q}Ts} P(s)}{s} \right\}\Bigg|_{q=z^{-1}} \tag{3}$$

where $[\bullet]_{Hm}$ is the so-called "modified zero-order hold transform [14]", and where $\mathcal{Z}\{\bullet\}$ is the standard $\mathcal{Z}$-transform. Here we have the m^{th} order numerator polynomial, $N_p(q)$, and the n^{th} order denominator polynomial $D_p(q)$. The continuous plant appears to have less poles than the discrete plant, i.e., $n = n_c + l + 1$, but this is illusory since the delay element can be considered to be infinite dimensional (having an infinite number of poles).

Since the continuous-time plant was assumed to be minimal, then the discrete-time plant is minimal for almost all sampling periods T, implying that the polynomials $N_p(q)$ and $D_p(q)$ are coprime for almost all sampling periods T. Here we assume T to be such that $N_p(q)$ and $D_p(q)$ are coprime. Without loss of generality, we also assume that $D_p(0) = 1$. The reference signal can be considered to be the sampled version of a continuous-time signal or a discrete-time signal. We assume that this signal is a unit step function and has the representation[2]

$$r(q) = \frac{1}{1-q}, \tag{4}$$

[1] The delay element may be defined in terms of the z transform variable $q = z^{-1}$.

[2] In general, we may wish to consider a general, non-decaying signal.

We also define the tracking and manipulation errors

$$e_y(q) = r(q) - y(q),\ e_u(q) = \frac{u_{ss}}{1-q} - u(q). \quad (5)$$

where $u_{ss} = \frac{D_p(1)}{N_p(1)}$ is a constant value.

Definition 1 *The Ripple Free Deadbeat Control Problem is to find a causal controller*

$$C(q) = \frac{u(q)}{e_y(q)} = \frac{N_c(q)}{D_c(q)} \quad (6)$$

where $N_c(q)$ and $D_c(q)$ are coprime and $D_c(0) = 1$, such that

1. *the closed-loop system is internally stable*
2. *the errors e_y and e_u settle to zero in a fixed number of discrete steps.*

Condition (2) implies that both $e_y(q)$ and $e_u(q)$ are polynomials of degree at most N. In addition to the above requirements, we will also desire that the transient response is "nice" according to some measure. This includes restrictions upon the output as well the control (manipulation) signal.

1.3 All Solutions to the Ripple-Free Deadbeat Problem

We first introduce the stable-unstable factorization

$$D_p(q) = D_{ps}(q)D_{pu}(q) \quad (7)$$

where D_{ps} is a stable factor having all its roots outside the closed unit disk. We also assume that D_{pu} has no root at $q = 1$[3]. The following result gives all solutions to the Ripple Free Deadbeat Control Problem[4] [19]

[3] This is not a fundamental problem with the design. It only makes the derivation simpler. Indeed, if there is a pole at $q = 1$, the order of the resulting controller may be reduced by 1.

[4] For general inputs, this result is necessary, but not sufficient [20, 8].

Lemma 1.1: *The Ripple Free Deadbeat Control problem has a solution if and only if $N_p(1) \neq 0$. Moreover, all solutions are of the form*

$$N_c(q) = D_{ps}(q)Q_n(q) \quad (8)$$
$$D_c(q) = (1-q)Q_d(q) \quad (9)$$

where $Q_n(q)$ and $Q_d(q)$ are polynomial solutions of the Diophantine equation

$$[N_p(q)]\, Q_n(q) + [D_{pu}(q)(1-q)]\, Q_d(q) = 1 \quad (10)$$

such that $Q_d(0) = 1$. ■

The condition that $N_p(1) \neq 0$ is that the numerator of the plant is coprime with $(1-q)$ and implies that the system has no transmission zero. Having a transmission zero would be like our system being a "notch" filter, and applying an input at the same frequency as the notch. Since this is not a reasonable thing to do, this condition is reasonable. We note that $N_p(1) \neq 0$ implies that $u_{ss} = \frac{D_p(1)}{N_p(1)}$ is a finite value.

Equations (8)-(10) characterize all solutions to the Ripple Free Deadbeat problem for a given plant with a step input. Minimum settling time may be achieved by making D_{ps} be the largest stable factor and by obtaining the smallest order Diophantine solution.

In practice, it is not advisable to cancel stable plant poles if they are poorly damped. This is achieved by including in D_{ps} only the "fast" poles, (i.e., those outside a disk of radius greater than 1.) However, the settling time increases by one sample period for each additional pole included in D_{pu}.

The Diophantine equation has an infinite number of solutions, and all of them provide an internally stabilizing controller that solves the Ripple Free Deadbeat problem. However, using the design freedom of this equation has not been easy, in general. One reason for this is that the standard solution of the Diophantine equation is often found in terms of the associated *resolvent* matrix [11, 1]. The matrix equation using the resolvent matrix, however, has not been particularly helpful in parametrizing the controllers or the design constraints; somewhat masking the true design freedom of the problem. We next consider a simpler parametrization for the solution of the Diophantine equation, first obtained in [15].

2 A Parametrization of Solutions

The solution of the Diophantine equation, using the resolvent matrix is not convenient for optimization purposes. As a result, searching for optimal solutions is complicated. This parametrization does not lend itself well to characterizing control constraint conditions that arise in practice. We now consider a different parametrization of the solutions of the Diophantine equation that also lends itself well to the characterization of control constraints. To begin, we consider two basic properties of polynomials

2.1 A Matrix Representation of Polynomial Products

Suppose we have the polynomials

$$A(q) = a_0 + a_1 q + a_2 q^2 + \cdots + a_m q^m \quad (11)$$
$$B(q) = b_0 + b_1 q + b_2 q^2 + \cdots + b_n q^n \quad (12)$$

We define the *vectorized* form of the polynomials

$$\vec{A} = \begin{bmatrix} a_0 \\ \vdots \\ a_m \end{bmatrix}, \vec{B} = \begin{bmatrix} b_0 \\ \vdots \\ b_n \end{bmatrix} \quad (13)$$

Indeed, we may vectorize any polynomial this way. We may express this vectorization as an operator $(\vec{\bullet})$ We also define the *expanded matrix* form

$$\bar{A}_p = \begin{bmatrix} \underbrace{\begin{matrix} \vec{A} & & 0 \\ & \ddots & \\ 0 & & \vec{A} \end{matrix}}_{p-\text{columns}} \end{bmatrix} \in \mathbb{R}^{m+p+1 \times p}, \quad (14)$$

$\bar{B}_P$ is similar to $\bar{A}_P$. We have the following result, the proof of which is by simply multiplying the polynomials and gathering the coefficients of each power of q.

Lemma 2.1: *The following hold*

$$\left(\overrightarrow{AB}\right) = \bar{A}_{n+1}\vec{B} = \bar{B}_{m+1}\vec{A}. \blacksquare \quad (15)$$

Lemma 2.1 illustrates the fact that a polynomial product may be written in terms of a matrix product. In a comparable way, a polynomial division may also be written as a matrix equation.

Assuming now (without loss of generality) that $B(0) = 1$, we now consider the polynomial division

$$C(q) = \frac{A(q)}{B(q)} = c_0 + c_1 q + c_2 q^2 + \cdots \quad (16)$$

The right hand side of equation (16) is the Maclauren series expansion of the function and thus (in general) has an infinite number of nonzero terms. We note that equation (16) may also be written

$$A(q) = B(q)C(q) \quad (17)$$

We note that the left hand side of equation (17) has at most $m+1$ non-zero terms, and thus the right hand side must also. Thus, restricting our attention to a finite version of the sequence, we now consider the first N coefficients of C. We write the truncated version of this

$$C_N(q) = c_0 + c_1 q + \cdots + c_{N-1} q^{N-1} \quad (18)$$

and state the following result. The proof is in [15].

Lemma 2.2: *For the polynomial equation, $A(q) = B(q)C(q)$, the first N coefficients of C may be computed by*

$$\vec{C}_N = B_x^{-1} \begin{bmatrix} \vec{A} \\ 0_{N-m-1 \times 1} \end{bmatrix} \quad (19)$$

where B_x is obtained from the decomposition

$$\begin{bmatrix} B_x \\ B_y \end{bmatrix} = \bar{B}_N, B_x \in \mathbb{R}^{N \times N}, B_y \in \mathbb{R}^{n+1 \times N} \blacksquare \quad (20)$$

2.2 A Matrix Parametrization of Diophantine Solutions

Since the Diophantine equation has two polynomial products, this vectorization is a convenient way to express the equation. Considering that D_{pu} is of order r, we now define

$$\begin{aligned} D_y(q) &= (1-q)D_{pu}(q) && (21)\\ &= 1 + d_1 q + \cdots + d_{r+1}q^{r+1} && (22)\\ D_x(q) &= \frac{D_y(q)-1}{q} && (23)\\ &= d_1 + d_2 q + \cdots + d_{r+1}q^{r} && (24) \end{aligned}$$

We next assume that the solution to the Diophantine equation is of the form

$$\begin{aligned} Q_n(q) &= \gamma_0 + \gamma_1 q + \cdots + \gamma_p q^p && (25)\\ Q_d(q) &= 1 + \delta_1 q + \cdots + \delta_\nu q^\nu && (26) \end{aligned}$$

where p, the order of Q_n and ν, the order of Q_d are to be determined. The minimal settling time of the system is $N_{\min}$ steps. To determine this value, we examine the Diophantine equation

$$N_p(q)Q_n(q) + (1-q)D_{pu}(q)Q_d(q) - 1 = 0$$

and note that Q_n has $p{+}1$ and Q_d has ν unknowns giving a total of $\nu+p+1$ unknowns. There must be at least $\nu+p+1$ equations for a solution to exist. The term $(1-q)D_{pu}(q)Q_d(q)-1$ has $\nu{+}r{+}1$ coefficients and thus that many equations. Thus, we must have

$$r + \nu_{\min} + 1 = \nu_{\min} + p_{\min} + 1 \qquad (27)$$

or

$$p_{\min} = r. \qquad (28)$$

In order to find $\nu_{\min}$, we note that the degree of $N_p(q)Q_n(q)$ is $m+p$, and the degree of $(1-q)D_{pu}(q)Q_d(q)-1$ is $\nu+r+1$. Equating these with $p = p_{\min}$ gives

$$\nu_{\min} = p_{\min} + m - r - 1 = m - 1. \qquad (29)$$

Since the term $N_pQ_d{+}(1{-}q)D_{pu}Q_d$ is the denominator of the closed-loop system, and is of degree $\nu{+}p{+}1$, then $N = \nu{+}p{+}1$. The minimum value is

$$N_{\min} = \nu_{\min} + p_{\min} + 1 = m + r. \qquad (30)$$

From this, we find that for $N \geq N_{\min}$, the Diophantine polynomials must have the orders

$$p = N - m,\ \nu = N - r - 1 \qquad (31a)$$

For $N \geq N_{\min}$, we now define the following matrices

$$\begin{aligned} \Gamma &= \left(\overrightarrow{Q_n(q)}\right) \in \mathbb{R}^{p+1\times 1}, && (32)\\ \mathcal{A} &= \left(\overline{N_p(q)}\right)_{p+1} \in \mathbb{R}^{N\times p+1} && (33) \end{aligned}$$

$$\begin{aligned} \mathcal{B}_0 &= \begin{bmatrix} \vec{D}_x^T \\ 0_{N-r-1\times 1} \end{bmatrix} \in \mathbb{R}^{N\times 1}, && (34)\\ \mathcal{B}_1 &= \left(\overline{-D_y(q)}\right)_\nu \in \mathbb{R}^{N\times\nu}, && (35)\\ \Delta &= \begin{bmatrix} \delta_1 \\ \vdots \\ \delta_\nu \end{bmatrix} \in \mathbb{R}^{\nu\times 1} && (36) \end{aligned}$$

$$\begin{aligned} X_a &= I_N - \mathcal{A}\mathcal{A}^+ && (37)\\ X_0 &= -\left(X_a\mathcal{B}_1\right)^+ X_a\mathcal{B}_0 \in \mathbb{R}^{\nu\times 1} && (38) \end{aligned}$$

and where $X_1 \in \mathbb{R}^{\nu\times N-r-m}$ is the largest full rank compliment

$$X_1 = \left(X_a\mathcal{B}_1\right)_\perp,\ \text{with}\ \left(X_a\mathcal{B}_1\right)X_1 = 0. \qquad (39)$$

Note that all these matrices are defined only in terms of the system polynomials.

The following theorem presents the parametrization of solutions of the Diophantine equation in terms of the above defined matrices. The proof is given in [15]. Again, it is assuming that $N \geq N_{\min}$ is fixed.

Theorem 2.1: *The Diophantine equation (10) has a solution if and only if there exists a solution to the equation*

$$\mathcal{A}\Gamma = \mathcal{B}_0 + \mathcal{B}_1\Delta. \qquad (40)$$

In this case, for any $\mathcal{R} \in \mathbb{R}^{N-r-m\times 1}$

$$\begin{aligned} \Delta &= X_0 + X_1\mathcal{R} && (41)\\ \Gamma &= \mathcal{A}^+\left(\mathcal{B}_0 + \mathcal{B}_1\Delta\right) && (42) \end{aligned}$$

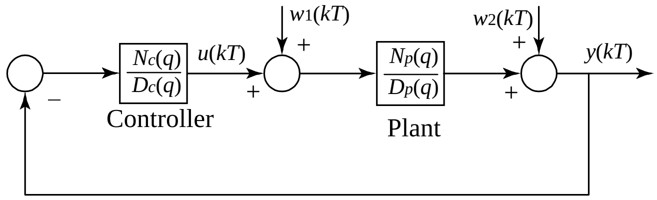

Figure 2: Closed Loop for Optimization Considerations

generate a solution to the Diophantine equation. ■

From this theorem, we see that there are $N-r-m$ free parameters in the Diophantine equation. When $N = N_{\min}$ we have a unique solution to the Diophantine equation and thus no free parameters.

This parametrization is affine in the parameter $\mathcal{R}$. The characterization of the ℓ_2 cost is made convenient by this parametrization.

3 Optimization of the ℓ_2 Norm

The use of the Diophantine solution to compute a solution to the Ripple Free Deadbeat problem has yielded controllers that, while maintaining the ripple free deadbeat property, have had terrible transient response. Thus, optimizing or constraining this response is of critical importance for this approach to be useful. In this section we consider the ℓ_2 norm as the optimization cost. This cost is considered in the discrete-time and gives rise to a least-squares optimization problem. In addition, we may also consider a related, weighted least-squares cost.

The norm under consideration is the ℓ_2 norm of the system errors. Previously, we defined these errors $e_y(q)$ and $e_u(q)$, which are illustrated as in Figure 2. The transfer functions are given by

$$\begin{aligned} \mathcal{T}_1(q) &= \frac{e_y(q)}{r(q)} = \frac{1}{1+\left(\frac{N_c}{D_c}\right)\left(\frac{N_p}{D_p}\right)} \\ &= \frac{(1-q)D_{pu}Q_d}{N_pQ_n+(1-q)D_{pu}Q_d} \quad (43) \\ &= (1-q)D_{pu}Q_d \quad (44) \end{aligned}$$

and

$$\begin{aligned} \mathcal{T}_2(q) &= \frac{e_u(q)}{r(q)} \\ &= \frac{D_p(1)}{N_p(1)} - \frac{D_pQ_n}{N_pQ_n+(1-q)D_{pu}Q_d} \quad (45) \\ &= \frac{D_p(1)}{N_p(1)} - D_pQ_n \quad (46) \end{aligned}$$

We note that since $N_pQ_n = 1-(1-q)D_{pu}Q_d$, that $N_p(1)Q_n(1) = 1-0 = 1$, and $Q_n(1) = \frac{1}{N_p(1)}$. Thus, $\mathcal{T}_2(1) = 0$. We note also that $\mathcal{T}_1(1) = 0$. Because of this, we see that $e_y(q)$ and $e_u(q)$ are signals that are non-zero for only a finite duration.

3.1 The ℓ_2 Norm

The ℓ_2 norm, for each of the errors are thus defined (for $T > 0$)[5]

$$\|e_y\|_2 = \sqrt{T\sum_{k=0}^{\infty} e_y(kT)^2} \quad (47)$$

$$\|e_u\|_2 = \sqrt{T\sum_{k=0}^{\infty} e_u(kT)^2} \quad (48)$$

For a fixed ϕ such that $0 \le \phi \le 1$, we consider the optimization cost

$$J = \tfrac{\phi}{T}\|e_y\|_2^2 + \tfrac{1-\phi}{T}\|e_u\|_2^2 \quad (49)$$

This cost allows for a trade-off between the tracking error energy and the control energy. We will see, as in standard LQ control problems, allowing $\phi = 1$ leads to a "cheap control." Here the control is constrained in its order, or equivalently, constrained in its settling time.

The minimizing controller, subject to the settling time condition $t_s = NT$, $N > N_{\min}$, is given in terms of the following matrices

$$\begin{aligned} \mathcal{C} &= \left(\overline{D_p(q)}\right)_{p+1}, \mathcal{D} = \left(\overline{D_{pu}(q)}\right)_{\nu} \quad (50) \\ \mathcal{E} &= \left[0_{N+n-m+1} \;\middle|\; E_1^{-1}\right], \quad (51) \end{aligned}$$

[5] The sampling period $T > 0$ is included to make this norm comparable to the continuous-time $\mathcal{L}_2$ norm $\|e_y(t)\|_2^2 = \int_0^\infty e_y^2(t)dt$.

$$E_1 = I_{N+n-m+1} \quad (52)$$

$$- \left[0_{N+n-m+1\times 1} \left| \begin{array}{c} I_{N+n-m} \\ 0_{1\times N+n-m} \end{array} \right. \right] \quad (53)$$

$$Y_0 = \left[\begin{array}{c} \left[\begin{array}{c} \vec{D}_{pu} \\ 0_{N-r-1} \end{array} \right] \\ \mathcal{ECA}^+\mathcal{B}_0 \end{array} \right], \quad (54)$$

$$Y_1 = \left[\begin{array}{c} \mathcal{D} \\ \mathcal{ECA}^+\mathcal{B}_1 \end{array} \right] \quad (55)$$

$$Z_0 = Y_0 + Y_1 X_0, \; Z_1 = Y_1 X_1 \quad (56)$$

$$Q_0(\phi) = \left[\begin{array}{cc} \phi I_N & 0 \\ 0 & (1-\phi) I_{N+n-m+1} \end{array} \right], \quad (57)$$

$$Q_1(\phi) = Z_1^T Q_0(\phi) Z_1 \quad (58)$$

$$\mathcal{R}_o = -Q_1^{-1}(\phi) Z_1^T Q_0(\phi) Z_0. \quad (59)$$

We now provide the optimal ℓ_2 norm solution, the proof of which is in the Appendix.

Theorem 3.1: *The cost $J(\mathcal{R})$ is minimized by $\mathcal{R}_o$. The optimal cost is given by*

$$J(\mathcal{R}_o) = Z_0^T \left(I - Q_0 Z_1 Q_1^{-1} Z_1^T Q_0 \right) Z_0. \blacksquare \quad (60)$$

The proof of this theorem relies upon the vectorization of the error e_y (which is zero after N steps) and of e_u (which is zero after $N+n-m+1$ steps). Thus, for

$$\vec{e}_y = \left[\begin{array}{c} e_y(0) \\ \vdots \\ e_y(N-1) \end{array} \right], \quad (61)$$

$$\vec{e}_u = \left[\begin{array}{c} e_u(0) \\ \vdots \\ e_u(N+n-m+1) \end{array} \right], \quad (62)$$

we write

$$J = \left[\; (\vec{e}_y)^T \quad (\vec{e}_u)^T \; \right] Q_0(\phi) \left[\begin{array}{c} \vec{e}_y \\ \vec{e}_u \end{array} \right]. \quad (63)$$

3.2 Weighted ℓ_2 Norm

We now consider a weighted ℓ_2 norm problem. Consider the weighting matrix

$$Q_a = \left[\begin{array}{cc} Q_{a1} & Q_{a2} \\ Q_{a2}^T & Q_{a3} \end{array} \right] \geq 0 \quad (64)$$

where the columns of Z_1 do not lie in the null-space of Q_a. This implies that

$$Q_b = Z_1^T Q_a Z_1 > 0. \quad (65)$$

We now define

$$\mathcal{R}_1 = -Q_b^{-1} Z_1^T Q_a Z_0. \quad (66)$$

and provide the optimal weighted ℓ_2 norm solution, associated with the cost

$$\tilde{J} = \left[\; (\vec{e}_y)^T \quad (\vec{e}_u)^T \; \right] Q_a \left[\begin{array}{c} \vec{e}_y \\ \vec{e}_u \end{array} \right], \quad (67)$$

the proof of which is in the Appendix.

Theorem 3.2: *The cost $\tilde{J}(\mathcal{R})$ is minimized by $\mathcal{R}_1$. The optimal cost is given by*

$$\tilde{J}(\mathcal{R}_o) = Z_0^T \left(I - Q_a Z_1 Q_b^{-1} Z_1^T Q_a \right) Z_0. \blacksquare \quad (68)$$

The cost (67) is a very general cost and can represent quantities such as a correlation function or "covariance" function.

4 Example: An Unstable System

Consider the unstable, SISO, time-delayed, LTI plant

$$\frac{y(s)}{u(s)} = \frac{50e^{-0.14s}}{(s-1)(s^2+s+1)} \quad (69)$$

We desire to control this system with a ripple free deadbeat controller in the following way

$$\text{minimize } \|e_y\|_2 \quad (70)$$

subject to

$$t_s \leq 1, \; N - fixed. \quad (71)$$

In addition to the parameters in the controller, we also have the sampling period as a variable. Thus, for the above criteria, we have the sampling period given by

$$T = \frac{t_s}{N}. \quad (72)$$

Note that we have $\phi = 1$, and thus $Q_o(\phi) \geq 0$.

MATLAB was used to perform the calculations. Doing so, we considered various number of steps

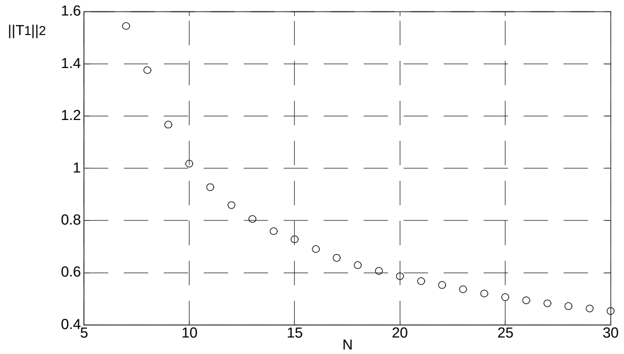

Figure 3: Optimal ℓ_2 Norm for the Deadbeat Controlled System

to settle, N, and solved the optimization. Figure 3 shows a plot of the minimum ℓ_2 norm versus N.

Here, we see that for $N > 20$, the norm does not decrease as rapidly as for $N < 20$. At $N = 7$, with $T = 1/7$ (where $\mathcal{Q} = 0.98, l = 0$) we have the plant

$$\frac{N_p(q)}{D_p(q)} = \frac{q(1.945\times10^{-7}+0.02678q+0.1047q^2+0.0255q^3)}{1-3.179q+3.326q^2-1.154q^3}$$

We found the optimal controller to be.

$$\frac{u(q)}{e_y(q)} = \frac{21.81-42.3q+23.87q^2-18.75q^3+28.23q^4-12.73q^5}{1+1.331q+1.187q^2-0.6495q^3-1.244q^4-1.343q^5-0.2815q^6}$$

At $N = 30$, with $T = 1/30$ (where $\mathcal{Q} = 0.3, l = 4$) we have the plant

$$\frac{N_p(q)}{D_p(q)} = \frac{10^{-3}q^5(0.1591+1.186q+0.5359q^2+0.002549q^3)}{1-3.035q+3.069q^2-1.034q^3}$$

We found a 29th order optimal controller (too big to display). The responses to each of these systems is shown in Figure 4. The $N = 7$ controller is seen to have substantially greater overshoot than the $N = 30$. The control magnitude for the $N = 30$ controller, is substantially larger. Both designs provide an internally stabilizing controller that is able to handle the time delay with ripple-free response after $t = 1$.

The resulting designs are fairly robust with respect to uncertainty in the time-delay. The following table shows the maximum and minimum time delay values such that stability is maintained. Here, we see that the $N = 30$ design is significantly more robust than the $N = 7$ design, even though.

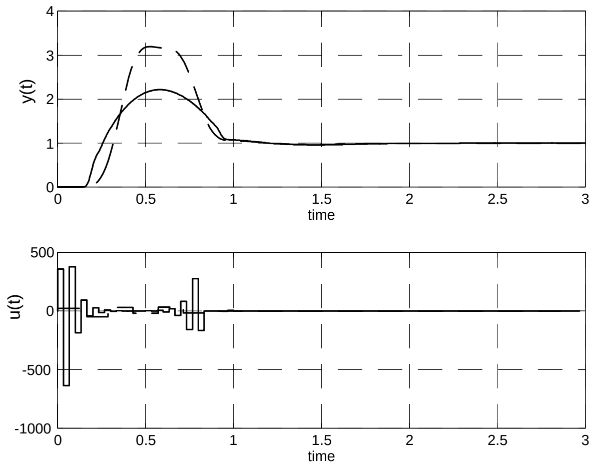

Figure 4: Time Responses for 6th and 29th Order Controllers

Table 4.1: Allowable Time-Delay for the Optimal Controllers

N	τ_d min	τ_d nominal	τ_d max
7	0.106	0.140	0.193
30	0.079	0.140	0.211

5 Conclusion

A solution to the optimal ripple-free deadbeat control problem has been obtained. A new parametrization of the controllers has been used, and conditions have been provided for optimizing the ℓ_2 norm of the closed-loop system. Future directions for research include considering the general tracking problem with multiple time delays.

References

[1] K. J. Åström and B. Wittenmark.1989, *Adaptive Control*, Reading MA: Addison-Wesley Publishing Company, (1989).

[2] S. Barnett. *Polynomials and Linear Control Systems*, New York: Marcel-Dekker Inc, (1983).

[3] A. Ben-Israel. *Generalized Inverses: Theory and Applications*, New York: John Wiley & Sons, (1974).

[4] A. R. Bergen and J. R. Ragazzini. “Sampled-Data Processing Techniques for Feedback Control Systems” *Trans. AIEE, Pt. II*, **volume 73**, pp. 236-247, (1954).

[5] J. Doyle, A. Packard, and K. Zhou. “Review of LFTs, LMIs, and μ”. *Proceedings of the 30th conference on Decision and Control*, Brighton, England, pp. 1227-1232, (1991).

[6] H. Elaydi. *Complex Controller Design Using Linear Matrix Inequalities.* Ph.D. thesis, New Mexico State University, (1998).

[7] L. El Ghaoui, F. Delebecque, and R. Nikoukhah. *LMITOOL: A User-Friendly Interface for LMI Optimization, User's Guide*, available via anonymous ftp from `ftp.ensta.fr` under the directory `/pub/elghaoui/limtool, (1997).`

[8] G. F. Franklin and A. Emami-Naeini. “Design of Ripple-Free Multivariable Robust Servomechanisms”. *IEEE Trans. Aut. Control*, **volume AC-31**, pp. 661-664, (1986).

[9] Y. Funahashi and H. Katoh. “Input/Output Deadbeat Control”, *International Journal of Control*, **volume 56**, pp. 515-530, (1992).

[10] T. T. Hartley, R. J. Veillette, and G. Cook. “Techniques in Deadbeat and One-Step-Ahead Control”, *Control and Dynamic Systems – Advances in Theory and Applications*, **volume 79**, Digital Control Systems Implementation and Computational Techniques. C.T. Leondes, ed., Academic Press, pp. 117-159, (1996)

[11] V. Kučera. *Discrete Linear Control - The Polynomial Equation Approach*, New York: John Wiley & Sons, (1979).

[12] E. B. Lee, T. Kaczorek, and S.H. Żak. “Minimal-time Dead-Beat Control with Control Action Constraints (SISO case)”, in *Proc. 21st IEEE Conf. Decis. Control*, Orlando, FL, pp. 969-972, (1982).

[13] Y. Nesterov and A. Nemirovskii. *Interior Point Polynomial Algorithms in Convex Programming*, **volume 13** of Studies in Applied Math, SIAM, Philadelphia, PA, (1994).

[14] R. A. Paz. *Computer Controlled Systems*, to be published, New York: John Wiley & Sons, (2001).

[15] R. Paz and H. Elaydi, “Optimal Ripple-Free Deadbeat Controllers”, *Int. Jour. Control,* **volume 71, no. 6,** pp. 1087-1104, (1998).

[16] Y. Peng and R. Hanus. “An FIR Prefilter Leading to Bounded Overshoot and Undershoot with Application to Deadbeat Control”, *Int. Jour. Control*, **volume 58**, pp. 459-470, (1993).

[17] A. Porter and F. Stoneman. “A New Approach to the Design of Pulse-Monitored Servo Systems”, *Proc. IEE, Pt. II*, **volume 97**, pp. 597, (1950)

[18] O. A. Sebakhy and J. A. Assiri. “Optimal Response in SISO Ripple-Free Deadbeat Systems”, *Optim. Control App. & Meth.,* **volume 11**, pp. 103-109, (1990).

[19] H. R. Sirisena. “Ripple-Free Deadbeat Control of SISO Discrete Systems”, *IEEE Trans. Aut. Control,* **volume AC-30**, pp. 168-170, (1985).

[20] S. Urikura and A. Nagata. “Ripple-Free Deadbeat Control for Sampled Data Systems” *IEEE Trans. Aut. Control,* **volume AC-32**, pp. 474-482, (1987).

[21] L. Vandenberghe and S. Boyd. *SP: Software for Semidefinite Programming, User's Guide*, Available via anonymous ftp from `isl.stanford.edu`, under the directory, `/pub/boyd/semidef_prog, (1994).`

[22] Y. Zhao and H. Kimura. “Dead-Beat Control With Robustness”, *International Journal of Control*, **volume 43**, pp. 1427-1440, (1986).

A Appendix

A.1 Proof of Theorem 3.1

The error e_y is nonzero after N steps, and the non-zero terms may be expressed

$$\begin{aligned} \vec{e}_y &= \left(\overline{D_{pu}}\right)_{\nu+1} \vec{Q}_d && \text{(A.1)} \\ &= \begin{bmatrix} \vec{D}_{pu} \\ 0_{N-r-1} \end{bmatrix} + \left(\overline{D_{pu}}\right)_{\nu} \Delta && \text{(A.2)} \end{aligned}$$

and thus

$$\frac{1}{T} \|e_y\|_2^2 = (\vec{e}_y)^T \vec{e}_y \qquad \text{(A.3)}$$

We also note that $D_p(q)Q_n(q) = \begin{bmatrix} 1 & q & \cdots & q^{N+n-m+1} \end{bmatrix} (D_p)_{p+1} \Gamma$ and thus

$$\begin{aligned} \frac{D_p(1)}{N_p(1)} &= D_p(1)Q_n(1) && \text{(A.4)} \\ &= \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} (D_p)_{p+1} \Gamma && \text{(A.5)} \end{aligned}$$

We thus write

$$\begin{aligned} \mathcal{T}_2(q) &= \frac{D_p(1)}{N_p(1)} - D_p(q)Q_n(q) \\ &= (1-q) \begin{bmatrix} 1 & q & \cdots & q^{N+n-m} \end{bmatrix} \times \\ & \mathcal{ECA}^+ (\mathcal{B}_0 + \mathcal{B}_1 \Delta) && \text{(A.6)} \end{aligned}$$

Since $1 - q^\ell = (1-q)(1 + q + \cdots + q^{\ell-1})$

Thus we have

$$e_u(q) = \mathcal{T}_2(q) r(q)$$

and

$$\vec{e}_u(q) = \mathcal{ECA}^+ (\mathcal{B}_0 + \mathcal{B}_1 \Delta) \qquad \text{(A.7)}$$

with

$$\frac{1}{T} \|e_u\|_2^2 = \sum_{k=0}^{N+n-m} e_u(kT)^2 = (\vec{e}_u)^T \vec{e}_u$$

Stacking $\vec{e}_y$ and $\vec{e}_u$, we get

$$\begin{aligned} \vec{e} &= \begin{bmatrix} \vec{e}_y \\ \vec{e}_u \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \vec{D}_{pu} \\ 0_{N-r-1} \end{bmatrix} + \mathcal{D}\Delta \\ \mathcal{ECA}^+ (\mathcal{B}_0 + \mathcal{B}_1 \Delta) \end{bmatrix} \\ &= Y_0 + Y_1 \Delta && \text{(A.8)} \\ &= Y_0 + Y_1 (X_0 + X_1 \mathcal{R}) && \text{(A.9)} \\ &= Z_0 + Z_1 \mathcal{R} && \text{(A.10)} \end{aligned}$$

for given $\mathcal{R} \in \mathbb{R}^{N-r-m \times 1}$. We now write the cost as

$$J = J(\mathcal{R}) = \vec{e}^T Q_0(\phi) \vec{e} \qquad \text{(A.11)}$$

We note that $\mathcal{D}X_1$ and $\mathcal{ECA}^+\mathcal{B}_1 X_1$ are both full column rank. Thus,

$$Q_1(\phi) = Z_1^T Q_0(\phi) Z_1 > 0 \qquad \text{(A.12)}$$

for all $0 < \phi < 1$, even though $Q_0(\phi) \geq 0$ (and not positive definite) when $\phi = 1$ or $\phi = 0$.

Let $\mathcal{R} \in \mathbb{R}^{N-r-m \times 1}$ be an arbitrary, and ϕ be fixed, and consider the "error" $\varepsilon = \vec{e} - \vec{e}_0$, where $\vec{e}_0 = \vec{e}(\mathcal{R}_0)$. We note that the optimal value is orthogonal to the error. This is seen in that

$$\varepsilon^T Q_0 \vec{e} = (\vec{e} - \vec{e}_0)^T Q_0 \vec{e}_0 = 0 \qquad \text{(A.13)}$$

Thus, we consider the cost

$$\begin{aligned} J(\mathcal{R}) &= \vec{e}_0^T Q_0 \vec{e}_0 && \text{(A.14)} \\ & + (\vec{e} - \vec{e}_0)^T Q_0 (\vec{e} - \vec{e}_0) && \text{(A.15)} \end{aligned}$$

Thus, for all admissible $\mathcal{R}$, we have $J(\mathcal{R}) \geq J(\mathcal{R}_0)$, since $\mathcal{R}_0$ is an admissible value. The minimum value of the cost is given by

$$J(\mathcal{R}_0) = Z_0^T (I - Q_0 Z_1 Q_1^{-1} Z_1^T Q_0) Z_0 \blacksquare \qquad \text{(A.16)}$$

A.2 Proof of Theorem 3.2

Once again, we note that the optimal value is orthogonal to the error. Let $\mathcal{R} \in \mathbb{R}^{N-r-m \times 1}$ be an arbitrary, and ϕ be fixed, and consider the "error" $\varepsilon = \vec{e} - \vec{e}_1$, where $\vec{e}_1 = \vec{e}(\mathcal{R}_1)$. Thus, it is seen that

$$\begin{aligned} \varepsilon^T Q_a \vec{e} &= (\mathcal{R} - \mathcal{R}_1)^T (Z_1^T Q_a Z_0 - Z_1^T Q_a Z_0) \\ &= 0 \end{aligned}$$

Thus, we consider the cost

$$J(\mathcal{R}) = (\vec{e}_1)^T Q_a \vec{e}_1 + (\vec{e} - \vec{e}_1)^T Q_a (\vec{e} - \vec{e}_1)$$

Thus, for all admissible $\mathcal{R}$, we have $J(\mathcal{R}) \geq J(\mathcal{R}_1)$. Since $\mathcal{R}_1$ is an admissible value, then it is the optimal value. The minimum value of the cost is given by

$$\begin{aligned} J(\mathcal{R}_1) &= (\vec{e}_1)^T Q_a \vec{e}_1 \\ &= Z_0^T (I - Q_a Z_1 Q_b^{-1} Z_1^T Q_a) Z_0 \blacksquare \end{aligned}$$