INT. J. CONTROL, 1998, VOL. 71, NO. 6, 1087–1104

Optimal ripple-free deadbeat controllers

ROBERT PAZ† and HATEM ELAYDI†

A ripple-free deadbeat controller for a system exists if and only if there are no transmission zeros coinciding with the poles of the reference signal. Approaches to this problem often use the Diophantine equation solution. However, solutions provided by the Diophantine equation often exhibit extremely bad transient responses. This approach gives a new affine parametrization of solutions of the Diophantine equation. Based on this parametrization, LMI conditions are used to provide optimal or constrained controllers for design quantities such as overshoot, undershoot, control amplitude, 'slew rate' as well as for norm bounds such as ℓ_1, ℓ_2 and ℓ_∞.

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 provided here allows for constraints on control magnitude as well as on many time domain properties such as overshoot, undershoot, slew rate, and also on such system norm quantities as ℓ_1, ℓ_2, ℓ_∞ and $\mathcal{H}_\infty$ norms. The approach also provides an efficient computational design using linear matrix inequalities (LMIs), which may be solved using readily available software such as MATLAB‡ (see also Vandenberghe and Boyd 1994, El-Ghaoui *et al.* 1997).

1.1. *Digital deadbeat systems*

The study of deadbeat error control of discrete-time systems dates back to the early 1950s (Porter and Stoneman 1950, Bergen and Ragazzini 1954). An excellent perspective into the long and rich history of this problem is given in Hartley *et al.* (1996).

The general solution to the problem at hand was given by Kučera (1979) and Sirisena (1985), in terms of the solutions of a Diophantine equation. These solutions hold for general tracking problems in discrete-time, but 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 (Franklin and Emami-Naeini 1986, Urikura and Nagata 1987). For the problem with a unit step reference signal, this is taken into account in the

Received 15 July 1997. Revised 20 May 1998.

† Klipsch School of Electrical and Computer Engineering, New Mexico State University, PO Box 30001, Dept. 3-O, Las Cruces, NM 88003, USA. e-mail: rpaz@nmsu.edu, http://ctrlsys.nmsu.edu

‡ MATLAB, which is a registered trademark of the MathWorks Inc., has a computational toolbox called the LMI Toolbox that can solve problems cast as LMI problems.

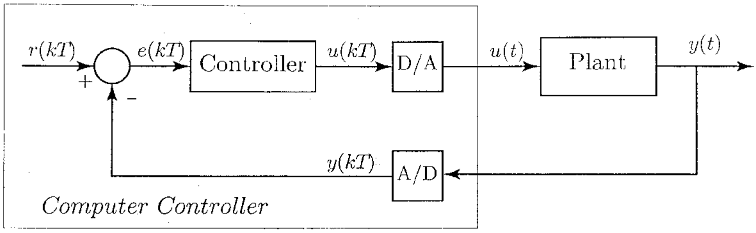

Figure 1. Closed-loop hybrid control system.

zero-order hold (ZOH), and so the conditions in Sirisena (1985) are both necessary and sufficient for the hybrid system.

1.2. *System formulation*

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

$$\frac{y(s)}{u(s)} = P(s) = \frac{N_{pc}(s)}{D_{pc}(s)} \tag{1.1}$$

We desire to control this system in a 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.

Obtaining a discrete model of the plant that is accurate at the sampling instants, kT (where T is the sampling period), we obtain the transfer function

$$\frac{y(q)}{u(q)} = P_H(q) = \frac{N_p(q)}{D_p(q)} \tag{1.2}$$

where

$$P_H(q) = z\left\{\left(\frac{1 - e^{-sT}}{s}\right)P(s)\right\}\Bigg|_{q=z^{-1}} \tag{1.3}$$

expressed in terms of the delay element q†. Here we have the mth order numerator polynomial, $N_p(q)$, and the nth order denominator polynomial $D_p(q)$. Since the continuous-time plant was assumed to be minimal, then the discrete-time plant is minimal for almost all sampling periods T. Also, 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. By causality, we also assume that $D_p(0) = 1$, and also $N_p(0) = 0$. In general, the reference signal can be considered to be the sampled version of a continuous-time signal or a discrete signal. The solution presented here solves essentially the discrete-time ripple-free deadbeat problem. The complete solution to the continuous-time ripple-free deadbeat problem requires the use of a continuous-time internal tracking model. In the case of a unit step input, this is solved by the use of a zero-order hold (D/A convertor).

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

Here we assume that this signal is a unit step function and has the representation†

$$r(q) = \frac{1}{1 - q} \tag{1.4}$$

We also define the tracking and manipulation errors

$$e_y(q) = r(q) - y(q), \quad e_u(q) = \frac{u_{ss}}{1 - q} - u(q) \tag{1.5}$$

where $u_{ss} = D_p(1)/N_p(1)$ is a finite, constant value.

Definition 1: The ripple-free deadbeat control problem is to find a causal controller

$$C(q) = \frac{u(q)}{e_y(q)} = \frac{N_c(q)}{D_c(q)} \tag{1.6}$$

where $N_c(q)$ and $D_c(q)$ are coprime and $D_c(0) = 1$, such that the closed-loop system is internally stable, and the errors e_y and e_u settle to zero in N discrete steps.

The above definition 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 of the overall system is 'nice' according to some measure.

1.3. *All solutions to the ripple-free deadbeat problem*

We first introduce the factorization

$$D_p(q) = D_{ps}(q)D_{pu}(q) \tag{1.7}$$

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 $q = 1$‡. The following result gives all solutions to the ripple-free deadbeat control problem§ (Sirisena 1985).

Lemma 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*

$$N_c(q) = D_{ps}(q)Q_n(q) \tag{1.8}$$

$$D_c(q) = (1 - q)Q_d(q) \tag{1.9}$$

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

$$\left[N_p(q)\right]Q_n(q) + \left[D_{pu}(q)(1 - q)\right]Q_d(q) = 1 \tag{1.10}$$

such that $Q_d(0) = 1$.

Equations (1.8)–(1.10) 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} the largest stable factor and by obtaining the smallest order Diophantine solution.

---

† In general, we may wish to consider a general, non-decaying signal.

‡ 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.

§ For general inputs, this result is necessary, but not sufficient (Franklin and Emami-Naeini 1986, Urikura and Nagata 1987).

The Diophantine equation has an infinite number of solutions, and each of them provide an internally stabilizing controller that solves the ripple-free deadbeat problem. These solutions solve the steady-state portion of the time response, but ensure nothing for the transient response. Using the design freedom of this equation has not been easy, in general. We next present a simple parametrization for the solution of the Diophantine equation that allows ready access to the design freedom.

2. A parametrization of solutions

The solution of the Diophantine equation, using the so-called *resolvant* matrix (Kučera 1979) is not convenient for optimization purposes. As a result, searching for optimal solutions is complicated. We now consider a different parametrization of the solutions of the Diophantine equation that allows ready access to the design freedom in 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

$$A(q) = a_0 + a_1 q + a_2 q^2 + \cdots + a_m q^m \tag{2.1}$$

$$B(q) = b_0 + b_1 q + b_2 q^2 + \cdots + b_n q^n \tag{2.2}$$

We define the *vectorized* form of the polynomials

$$\vec{A} = \begin{bmatrix} a_0 \\ \vdots \\ a_m \end{bmatrix} \in \mathbb{R}^{m+1}, \qquad \vec{B} = \begin{bmatrix} b_0 \\ \vdots \\ b_n \end{bmatrix} \in \mathbb{R}^{n+1} \tag{2.3}$$

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

$$\bar{A}_p = \begin{bmatrix} \vec{A} & & 0 \\ & \ddots & \\ 0 & & \vec{A} \end{bmatrix} \in \mathbb{R}^{m+p+1\times p}, \qquad \bar{B}_p = \begin{bmatrix} \vec{B} & & 0 \\ & \ddots & \\ 0 & & \vec{B} \end{bmatrix} \in \mathbb{R}^{n+p+1\times p} \tag{2.4}$$

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: *The following hold*

$$(\overrightarrow{AB}) = \bar{A}_{n+1}\vec{B} = \bar{B}_{m+1}\vec{A} \tag{2.5}$$

Lemma 2 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

$$C(q) = \frac{A(q)}{B(q)} = c_0 + c_1 q + c_2 q^2 + \cdots \tag{2.6}$$

The right-hand side of equation (2.6) is the Maclauren series expansion of the function and thus has an infinite number of terms. We note that equation (2.6) may also be written

$$A(q) = B(q)C(q) \tag{2.7}$$

We note that the left-hand side of equation (2.7) 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

$$C_N(q) = c_0 + c_1 q + \cdots + c_{N-1} q^{N-1} \tag{2.8}$$

and state the following result. The proof is in the Appendix.

Lemma 3: *For the polynomial equation, $A(q) = B(q)C(q)$, the first N coefficients of C may be computed by*

$$\vec{C}_N = B_x^{-1} \begin{bmatrix} \vec{A} \\ 0_{N-m-1\times 1} \end{bmatrix} \tag{2.9}$$

where B_x is obtained from the decomposition

$$\begin{bmatrix} B_x \\ B_y \end{bmatrix} = \bar{B}_N, \qquad B_x \in \mathbb{R}^{N\times N}, \qquad B_y \in \mathbb{R}^{n+1\times N} \tag{2.10}$$

2.2. *A matrix parametrization of solutions*

Since the Diophantine equation has two polynomial products, this vectorization is a convenient way to express the equation. Considering that the product $(1 - q)D_{pu}$ is of order r, we now define

$$D_y(q) = (1 - q)D_{pu}(q) = 1 + d_1 q + \cdots + d_r q^r \tag{2.11}$$

$$D_x(q) = \frac{D_y(q) - 1}{q} = d_1 + d_2 q + \cdots + d_r q^{r-1} \tag{2.12}$$

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

$$Q_n(q) = \alpha_0 + \alpha_1 q + \cdots + \alpha_p q^p \tag{2.13}$$

$$Q_d(q) = 1 + \beta_1 q + \cdots + \beta_\ell q^\ell \tag{2.14}$$

where p, the order of Q_n and ℓ, the order of Q_d are to be determined. We also define

$$W(q) = \frac{Q_d(q) - 1}{q} = \beta_1 + \beta_2 q \cdots + \beta_\ell q^{\ell-1} \tag{2.15}$$

We note that by expanding, we may write

$$D_y(q)Q_d(q) = 1 + q(D_x(q) + D_y(q)W(q)) \tag{2.16}$$

Since $N_p(0) = 0$, then we can write $N_p(q) = qN_{p1}(q)$, for some polynomial $N_{p1}(q)$. The Diophantine equation becomes

$$N_{p1}(q)Q_n(q) = -D_x(q) - D_y(q)W(q) \tag{2.17}$$

The system with the given controller structure will settle in a minimum of $N_{\min}$ steps. In the Diophantine equation, there are $p + \ell + 1$ unknown values. The term $1 - D_y Q_d$ has $r + \ell$ coefficients, while the product $N_p Q_n$ has $m + p$ coefficients. In order to get a solution to the Diophantine equation, the highest power of these polynomials must cancel each other, requiring $r + \ell = m + p = N$. In order to get an *exact* solution to the Diophantine equation, we need at least as many equations as unknowns, requiring that at $N_{\min}$, $p + \ell + 1 = r + \ell = m + p$. Thus

$$N_{\min} = m + r - 1 \tag{2.18}$$

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

$$p = N - m, \qquad \ell = N - r \tag{2.19}$$

We desire to vectorize the equation (2.17). We thus define the following matrices dependent on the known polynomials

$$\mathcal{A} = \overline{(N_p(q))}_{p+1} \in \mathbb{R}^{N\times p+1}, \qquad \mathcal{B}_0 = \begin{bmatrix} -\vec{D}_x \\ 0_{N-r\times 1} \end{bmatrix} \in \mathbb{R}^{N\times 1} \tag{2.20}$$

$$\mathcal{B}_1 = \overline{(-D_y(q))}_\ell \in \mathbb{R}^{N\times \ell} \tag{2.21}$$

and vectors that depend upon the unknown polynomials

$$\mathbb{N} = \vec{Q}_n = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_p \end{bmatrix}, \qquad \mathbb{D} = \vec{W} = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_\ell \end{bmatrix} \tag{2.22}$$

In addition we also define some matrices associated with the solution of the Diophantine equation

$$X_a = I_N - \mathcal{A}\mathcal{A}^+ = I_N - \mathcal{A}(\mathcal{A}^{\mathrm{T}}\mathcal{A})^{-1}\mathcal{A}^{\mathrm{T}} \tag{2.23}$$

$$X_0 = -(X_a\mathcal{B}_1)^+ X_a \mathcal{B}_0 \in \mathbb{R}^{s\times 1} \tag{2.24}$$

and where $X_1 \in \mathbb{R}^{\ell\times N-N_{\min}}$ as the maximal full rank complement

$$X_1 = (X_a\mathcal{B}_1)_\perp, \qquad \text{with} \quad (X_a\mathcal{B}_1)X_1 = 0 \tag{2.25}$$

Note that all these matrices are defined only in terms of the known 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 the Appendix.

Theorem 1: *The Diophantine equation* (1.10) *has a solution if and only if there exist* $\mathbb{N}$ *and* $\mathbb{D}$ *such that*

$$\mathcal{A}\mathbb{N} = \mathcal{B}_0 + \mathcal{B}_1\mathbb{D} \tag{2.26}$$

If this is the case, then for any $\mathcal{R} \in \mathbb{R}^{N-N_{\min}}$, *the following give a solution to* (2.26).

$$\mathbb{D} = X_0 + X_1\mathcal{R} \tag{2.27}$$

$$\mathbb{N} = \mathcal{A}^+(\mathcal{B}_0 + \mathcal{B}_1\mathbb{D}) \tag{2.28}$$

From this theorem, we see that there are $N - N_{\min}$ free parameters in the Diophantine equation.

This parametrization is affine in the parameter $\mathcal{R}$. The use of LMI in the optimization process is made convenient by this parametrization.

3. LMI optimization conditions and constraints

Recent results in optimization theory have shown that if a problem can be cast as a linear matrix inequality (LMI), then efficient algorithms can be employed to solve them (e.g. interior point methods (Nesterov and Nemirovskii 1994)). As a result, many previously unsolvable problems (e.g. problems with no closed-form solution) can now have numerical solutions (Vandenberghe and Boyd 1994, El-Ghaoui *et al.* 1997).

The use of the Diophantine solution to compute a solution to the ripple-free deadbeat problem has yielded controllers that, with good steady-state response, 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 two kinds of constraints. The first kind we call 'standard' constraints. This refers to the standard kind of properties that arise in control design problems. Here, we consider the transient response quantities of overshoot, undershoot, maximum control amplitude, settling time and 'slew rate'. The second kind of property is the performance norm type of quantities. These are the ℓ_1, ℓ_2 and ℓ_∞ norms as well as the $\mathcal{H}_\infty$ norm. In addition other performance indices such as the STAE (sum of time-weighted absolute error) and STSE (sum of time-weighted square error) criteria are also considered. Because of the LMI framework, any of these quantities can be combined. Moreover, we may either optimize with respect to the quantity or simply require a hard constraint upon it. We consider each of the quantities in the discrete-time.

3.1. *'Standard' conditions*

In working with the quantities, we exploit the fact that the system settles in a fixed number (N) of steps. Thus, we are only concerned about the sequences

$$y(k), u(k), \dot{y}(k), e_y(k), \qquad 0 \le k \le N-1 \tag{3.29}$$

We thus work with the vectorized forms of these signals:

$$\overrightarrow{y_N} = \begin{bmatrix} y(0) \\ \vdots \\ y(N-1) \end{bmatrix}, \quad \overrightarrow{u_N} = \begin{bmatrix} u(0) \\ \vdots \\ u(N-1) \end{bmatrix},$$

$$\overrightarrow{\dot{y}_N} = \begin{bmatrix} \dot{y}(0) \\ \vdots \\ \dot{y}(N-1) \end{bmatrix}, \quad \overrightarrow{e_y} = \begin{bmatrix} y(0)-1 \\ \vdots \\ y(N-1)-1 \end{bmatrix} \tag{3.30}$$

Because of our affine parametrization of the controller, and the use of Lemmas 2 and 3 we may write these signals in matrix form. First, we note that

$$y(q) = \frac{N_p(q)Q_n(q)}{1 - q}, \qquad u(q) = \frac{D_p(q)Q_n(q)}{1 - q} \tag{3.31}$$

In order to compute the derivative of y, $\dot{y}$, we refer to the fact that we have the original continuous-time system. We thus compute

$$\frac{\dot{y}(q)}{u(q)} = z\left\{(1 - e^{-sT})P(s)\right\}\Big|_{q=z^{-1}} = \frac{N_d(q)}{D_p(q)} \tag{3.32}$$

Thus, we obtain

$$\dot{y}(q) = \frac{N_d(q)Q_n(q)}{1 - q} \tag{3.33}$$

By Lemmas 2 and 3 we may write

$$\overrightarrow{y_N} = E^{-1}\mathcal{A}\,\mathbb{N} \qquad \overrightarrow{u_N} = E^{-1}\mathcal{C}\,\mathbb{N} \qquad \overrightarrow{\dot{y}_N} = E^{-1}\mathcal{D}\,\mathbb{N} \tag{3.34}$$

where

$$\mathcal{C} = \left(\overline{D_p(q)}\right)_{p+1}, \quad \mathcal{D} = \left(\overline{N_d(q)}\right)_{p+1}, \quad E = I_N - \left[\begin{array}{c|c} \multicolumn{2}{c}{0_{1\times N}} \\ \hline I_{N-1} & 0_{N-1\times 1} \end{array}\right] \tag{3.35}$$

We are now in a position to provide the LMI conditions, for the standard problems, which we simply list. The proof of these is elementary. We also define diag$(\bullet)$ of a vectorized signal to be a diagonal matrix with the elements of the vector along the diagonal.

(1) *Settling time*
We consider the settling time to be $t_s = NT$. Thus, for $\psi_1 > 0$,

$$t_s < \psi_1 \quad \Leftrightarrow \quad T < \frac{\psi_1}{N} \tag{3.36}$$

The values T and N may be used as design parameters.

(2) *Overshoot*
The overshoot, M_1, corresponds to the peak value of y or $(\overrightarrow{e_y})$. Thus, for $\psi_2 > 0$,

$$M_1 < \psi_2 \quad \Leftrightarrow \quad \psi_2 I_N - \mathrm{diag}\,(\overrightarrow{e_y}) > 0 \tag{3.37}$$

(3) *Undershoot*
The understoot, M_2, corresponds to the minimum value of y. Thus, for $\psi_3 > 0$,

$$M_2 < \psi_3 \quad \Leftrightarrow \quad \psi_3 I_N - \mathrm{diag}\,(\overrightarrow{y_N}) > 0 \tag{3.38}$$

Note that if there is no overshoot or understoot, then (3.37) or (3.38) are automatically satisfied.

(4) *Maximum control amplitude*
The maximum control amplitude, M_u, corresponds to the maximum absolute value of the control signal. In this case, we have for $\psi_4 > 0$,

$$M_u < \psi_4 \quad \Leftrightarrow \quad \begin{cases} \psi_4 I_N - \mathrm{diag}\,(\overrightarrow{u_N}) > 0 \\ \psi_4 I_N + \mathrm{diag}\,(\overrightarrow{u_N}) > 0 \end{cases} \tag{3.39}$$

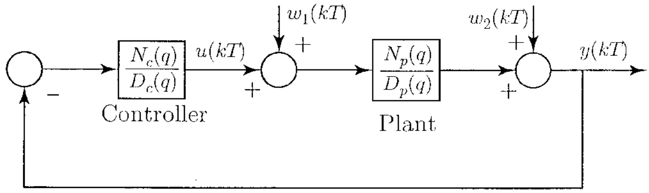

Figure 2. Closed loop system for robustness considerations.

(5) *Slew rate*
The slew rate is the maximum rate of change of the signal y, i.e. the maximum value of the derivative, $\dot{y}$. Thus, we have for $\psi_5 > 0$, $\psi_6 > 0$,

$$-\psi_5 < \dot{y}(q) < \psi_6 \quad \Leftrightarrow \quad \begin{cases} \psi_5 I_N + \text{diag}\,(\overrightarrow{\dot{y}_N}) > 0 \\ \psi_6 I_N - \text{diag}\,(\overrightarrow{\dot{y}_N}) > 0 \end{cases} \tag{3.40}$$

Here we have provided separate upper and lower bounds in the event that they are to be considered separately.

We now move on to consider some system norm bounds.

3.2. *Norm conditions*

Here, we are concerned with the norm of the closed-loop system as illustrated in figure 2, where we have added two possible sources of disturbance. The disturbance w_1 is a process noise, while w_2 is a sensor noise. The disturbance w_2 is more closely associated with the tracking performance problem, and thus we consider its effects more in detail. We also now neglect $r(q)$ as a step input, and set it equal to zero. We also assume that the disturbances lie in admissible signal spaces for the norm under consideration. We note, from figure 2, that we have the two transfer functions

$$\mathcal{T}_1(q) = \frac{y(q)}{w_1(q)} = \frac{(1-q)N_p(q)Q_d(q)}{D_{ps}(q)} \tag{3.41}$$

$$\mathcal{T}_2(q) = \frac{y(q)}{w_2(q)} = (1-q)D_{ps}(q)Q_d(q) \tag{3.42}$$

$$= \varepsilon_0 + \varepsilon_1 q + \cdots + \varepsilon_N q^N, \quad (\varepsilon_0 = 1) \tag{3.43}$$

Note that $\mathcal{T}_1$ is not a finite-settling†, while $\mathcal{T}_2$ is finite-settling. As such, we can vectorize it

$$\overrightarrow{\mathcal{T}_2(q)} = X_{e1} + X_{e2}\mathbb{D}, \qquad X_{e1} = \begin{bmatrix} 1 \\ \vec{D}_x \\ 0_{\ell-1\times 1} \end{bmatrix}, \qquad X_{e2} = \begin{bmatrix} 0_{1\times\ell} \\ (D_y)_\ell \end{bmatrix} \tag{3.44}$$

and also define

$$\boldsymbol{\varepsilon} = \text{diag}\,([\varepsilon_0, \ldots, \varepsilon_N]) \tag{3.45}$$

† If we initially chose $D_{pu}(q) = D_p(q)$, $D_{ps}(q) = 1$, then $\mathcal{T}_2$ would be finite settling.

and a set of scalars $p_0, \ldots, p_N$ that we put on the diagonal of the matrix

$$\mathcal{P} = \text{diag}\left(\left[p_0, \ldots, p_N\right]\right) \tag{3.46}$$

For all except the $\mathcal{H}_\infty$ norm, we restrict ourselves to working with $\mathcal{T}_2$.

(1) *ℓ_1 norm*
This corresponds to the sum of the absolute values of the errors (SAE†)

$$\left\|\mathcal{T}_2\right\|_{\ell_1} = \sum_{k=0}^{N-1} T\left|\varepsilon_i\right| \tag{3.47}$$

Thus, for $\psi_7 > 0$, $\left\|\mathcal{T}_2\right\|_{\ell_1} < \psi_7$ if and only if there exists diagonal $\mathcal{P} > 0$ such that

$$\mathcal{P} - \varepsilon > 0, \qquad \mathcal{P} + \varepsilon > 0, \qquad \psi_7 - T\left[1 \quad \cdots \quad 1\right]\mathcal{P}\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} > 0 \tag{3.48}$$

(2) *ℓ_2 norm*
This is the sum of the squares of the error (SSE)

$$\left\|\mathcal{T}_2\right\|_{\ell_2}^2 = \sum_{k=0}^{N-1} T\varepsilon_i^2 \tag{3.49}$$

Thus, for $\psi_8 > 0$,

$$\left\|\mathcal{T}_2\right\|_{\ell_2}^2 < \psi_8 \quad \Leftrightarrow \quad \begin{bmatrix} \psi_8 & \left(\overrightarrow{\mathcal{T}_2(q)}\right)^{\mathrm{T}} \\ \overrightarrow{\mathcal{T}_2(q)} & (1/T)I_{N+1} \end{bmatrix} > 0 \tag{3.50}$$

(3) *ℓ_∞ norm*
This is the peak value of the error. Thus, for $\psi_9 > 0$,

$$\left\|\mathcal{T}_2\right\|_{\ell_\infty} < \psi_9 \quad \Leftrightarrow \quad \begin{cases} \psi_9 I_{N+1} - \varepsilon > 0 \\ \psi_9 I_{N+1} + \varepsilon > 0 \end{cases} \tag{3.51}$$

(4) *$\mathcal{H}_\infty$ norm*
For the transfer function, $\mathcal{T}_1$, we may obtain the state model

$$\mathcal{T}_1(q) = \left[\begin{array}{c|c} F_1 & G_1 \\ \hline C_1 & 0 \end{array}\right] \tag{3.52}$$

where

$$F_1 = \begin{bmatrix} F_{11} & G_{11}C_{11} \\ 0 & F_{12} \end{bmatrix}, \qquad G_1 = \begin{bmatrix} 0_{N+n-r-1\times 1} \\ 1 \end{bmatrix} \tag{3.53}$$

$$C_1 = \left[\beta_{N+n-r} \quad \ldots \quad \beta_1\right] \tag{3.54}$$

† The sampling period, T, is included here so that this cost is like the integral of absolute error (IAE) cost in continuous-time. This also holds for the ℓ_2, STAE and STSE costs.

and where

$$D_{ps}(q) = 1 + \alpha_1 q + \cdots + \alpha_{n-r} q^{n-r} \tag{3.55}$$

$$P_2(q) = (1 - q) N_p(q) Q_d(q) = \beta_1 q + \cdots + \beta_{N+n-r} q^{N+n-r} \tag{3.56}$$

and

$$F_{11} = \left[\begin{array}{c|c} 0_{N-1\times 1} & I_{N-1} \\ \hline \multicolumn{2}{c}{0_{1\times N}} \end{array}\right], \qquad G_{11} = \begin{bmatrix} 0_{N-1\times 1} \\ 1 \end{bmatrix} \tag{3.57}$$

$$C_{11} = \begin{bmatrix} 1 & 0_{1\times N-1} \end{bmatrix}, \qquad F_{12} = \left[\begin{array}{ccc} 0_{n-r-1\times 1} & \multicolumn{2}{|c}{I_{n-r-1}} \\ \hline -\alpha_{n-r} & \ldots & -\alpha_1 \end{array}\right] \tag{3.58}$$

We note that all the unknown parameters appear in an affine way in C_1. Thus, we have (Doyle *et al.* 1991), for $\psi_{10} > 0$, that $\|\mathcal{T}_1\|_{\mathcal{H}_\infty} < \sqrt{\psi_{10}}$, if and only if there exists $W_1 > 0$ such that

$$\begin{bmatrix} F_1^{\mathrm{T}} W_1 F_1 - W_1 & F_1^{\mathrm{T}} W_1 G_1 & C_1^{\mathrm{T}} \\ G_1^{\mathrm{T}} W_1 F_1 & G_1^{\mathrm{T}} W_1 G_1 - \psi_{10} I & 0 \\ C_1 & 0 & -1 \end{bmatrix} < 0 \tag{3.59}$$

For the transfer function, $\mathcal{T}_2$, we obtain the state model

$$\mathcal{T}_2(q) = \left[\begin{array}{c|c} F_2 & G_2 \\ \hline C_2 & D_2 \end{array}\right] \tag{3.60}$$

where

$$F_2 = F_{11}, \quad G_2 = G_{11}, \quad C_2 = \begin{bmatrix} \varepsilon_{\mathrm{N}} & \ldots & \varepsilon_1 \end{bmatrix}, \qquad D_2 = 1 \tag{3.61}$$

We note that all the unknown parameters appear in an affine way in C_2. Thus, we have (Doyle *et al.* 1991), for $\psi_{11} > 0$, that $\|\mathcal{T}_2\|_{\mathcal{H}_\infty} < \sqrt{\psi_{11}}$ if and only if there exists $W_2 > 0$ such that

$$\begin{bmatrix} F_2^{\mathrm{T}} W_2 F_2 - W_2 & F_2^{\mathrm{T}} W_2 G_2 + C_2^{\mathrm{T}} D_2 & C_2^{\mathrm{T}} \\ G_2^{\mathrm{T}} W_2 F_2 + D_2^{\mathrm{T}} C_2 & G_2^{\mathrm{T}} W_2 G_2 + D_2^{\mathrm{T}} D_2 - \psi_{11} I & 0 \\ C_2 & 0 & -1 \end{bmatrix} < 0 \tag{3.62}$$

3.3. *Other performance indices*

Other performance criteria are also considered in the optimization of system responses. The next two criteria are the sum of the time-weighted absolute error (STAE) and the sum of the time-weighted square error (STSE). These criteria tend to more strongly penalize later part of the response. In addtion, we also consider the transient control energy.

(1) *STAE*
This is the sum of the time-weighted absolute errors

$$(\mathcal{T}_2)_{\mathrm{STAE}} = \sum_{k=0}^{N-1} kT^2 |\varepsilon_i| \tag{3.63}$$

Thus, for $\psi_{12} > 0$, $(\mathcal{T}_2)_{\mathrm{STAE}} < \psi_{12}$ if and only if there exists diagonal $\mathcal{P} > 0$ with

$$\mathcal{P} - \mathcal{E} > 0, \qquad \mathcal{P} + \mathcal{E} > 0, \qquad \psi_{12} - T^2 \begin{bmatrix} 0 & \dots & N \end{bmatrix} \mathcal{P} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} > 0 \tag{3.64}$$

(2) *STSE*

This is the sum of the time-weighted square errors

$$(\mathcal{T}_2)_{\mathrm{STSE}} = \sum_{k=0}^{N-1} kT^2 \varepsilon_i^2 \tag{3.65}$$

Thus, for $\psi_{13} > 0$,

$$(\mathcal{T}_2)_{\mathrm{STSE}} < \psi \Leftrightarrow \begin{bmatrix} \frac{1}{T^2}\psi_{13} & \begin{bmatrix} \varepsilon_1 & \dots & \varepsilon_N \end{bmatrix} \\ \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_N \end{bmatrix} & \mathrm{diag}\left(\begin{bmatrix} 1 & \dots & \frac{1}{N} \end{bmatrix} \right) \end{bmatrix} > 0 \tag{3.66}$$

(3) *Transient control energy*

The control energy in the transient portion of the response is also an important quantity. We note that

$$\vec{u}_{N-1} = X_{u1} + X_{u2}\mathcal{R} \tag{3.67}$$

where

$$X_{u1} = E^{-1} c \mathcal{A}^+ (\mathcal{B}_0 + \mathcal{B}_1 X_0), \qquad X_{u2} = E^{-1} c \mathcal{A}^+ \mathcal{B}_1 X_1 \tag{3.68}$$

The transient control energy is given by

$$\|u_N\|_{\ell_2}^2 = \sum_{k=0}^{N-1} Tu(kT)^2 \tag{3.69}$$

Thus, for $\psi_{14} > 0$,

$$\|u_N\|_{\ell_2}^2 < \psi_{14} \quad \Leftrightarrow \quad \begin{bmatrix} \frac{1}{T}\psi_{14} & (X_{u1} + X_{u2}\mathcal{R})^{\mathrm{T}} \\ (X_{u1} + X_{u2}\mathcal{R}) & I_N \end{bmatrix} \tag{3.70}$$

4. Computational issues

In the numerical solution of LMIs, software packages often allow the use of both linear matrix inequalities (LMI) as well as linear matrix equalities (Vandenberghe and Boyd 1994, El-Ghaoui *et al.* 1997). In equations (2.27) and (2.28), the solution of the Diophantine equation is given in terms of two conditions including free parameters. These are equivalent to the single LME

$$\mathcal{A}\Gamma - (\mathcal{B}_0 + \mathcal{B}_1\Delta) = 0 \tag{4.1}$$

as found in Theorem 1.

In the case of the ℓ_2 norm, if we desire a global optimal solution for fixed N, we can get a closed-form solution. Indeed, since we have $\overrightarrow{\mathcal{T}_2(q)} = X_{e1} + X_{e2}\mathbb{D}$, then we may write

$$\|\mathcal{T}_2\|_{\ell_2}^2 = T(\overrightarrow{\mathcal{T}_2(q)})^{\mathrm{T}}(\overrightarrow{\mathcal{T}_2(q)}) \tag{4.2}$$

$$= T(X_{e3} + X_{e4}\mathcal{R})^{\mathrm{T}}(X_{e3} + X_{e4}\mathcal{R}) \tag{4.3}$$

which is minimized when $\mathcal{R} = -X_{e4}^{+}X_{e3}$, where

$$X_{e3} = X_{e1} + X_{e2}X_0, \qquad X_{e4} = X_{e2}X_1 \tag{4.4}$$

Similarly, for the transient control energy, the global optimal solution for fixed N is given by $\mathcal{R} = -X_{\sqcap \in}^{+}X_{\sqcap \infty}$.

Any or all of the above LMIs can be included in the design process. Some of the conditions may be used in the optimization (minimizing with respect to ψ_i, with ψ_i not fixed), while some may be used to provide a hard constraint (i.e. fixing ψ_i). A design procedure would be as follows:

(1) Determine which properties are most important to incorporate.
(2) If certain hard constraints must be satisfied, check to ensure that a feasible solution exists for those constraints. Modify the 'hard' constraints if necessary to obtain a feasible solution.
(3) Determine, one-at-a-time, what the minimum possible values are for each of the unconstrained ψ_i to be considered, keeping the hard constraints intact.
(4) Vary the weighting on the unconstrained ψ_i until a suitable controller is obtained. If no controller is suitable, go back to (1) to add or change constraints.

In practice, after an initial design has been done, it may be noticed that the hybrid system has large inter-sample ripple before the settling time (after which there is no ripple). For example, the discrete-time constraint on the overshoot may be satisfied, but in the hybrid response, inter-sample ripple causes the hybrid overshoot to violate the constraint. In this case, adding a condition on the slew rate can minimize the ripple.

5. An example

Consider the unstable, SISO, LTI plant with a non-minimum phase zero

$$P(s) = \frac{4.5(-2s + 5)}{(s - 0.1)(s^2 + 6s + 10)} \tag{5.1}$$

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

$$\text{minimize } \|\mathcal{T}_2\|_{\ell_2} \tag{5.2}$$

subject to

$$t_s \leq 5, \quad M_u \leq 1, \quad M_1 \leq 0.91, \quad M_2 \leq 0.3 \tag{5.3}$$

In addition to the parameters in the controller, we also have the order of the controller and sampling period as variables. Thus, for the above cost criteria, we have

$$\psi_1 = 5, \quad \psi_2 = 0.91, \quad \psi_3 = 0.3, \quad \psi_4 = 1 \tag{5.4}$$

and we wish to minimize ψ_8. Satisfying the settling time criteria is accomplished by making $T = 5/N$, where N is related to the order of the controller and the number of degrees of freedom in it. For a given T, we have the discrete-time model

$$P_H(q) = (1 - q)z\left\{\frac{P(s)}{s}\right\}_{q=z^{-1}} = \frac{q(A_0 + A_1 q + A_2 q^2)}{(1 - q\mathrm{e}^{T/10})(1 - 2\mathrm{e}^{-3T}\cos(T)q + \mathrm{e}^{-6T}q^2)} \tag{5.5}$$

for some coefficients A_0, A_1, and A_2 that are functions of the sampling period T. Thus, $m = 3$. In this case, we have $D_{pu} = 1 - q\mathrm{e}^{T/10}$, and $D_{ps} = 1 - 2\mathrm{e}^{-3T}\cos(T)q + \mathrm{e}^{-6T}q^2$. Since we have a unit step reference signal, then $D_y = (1 - q)(1 - q\mathrm{e}^{T/10})$, giving $r = 2$. Thus, the minimum N becomes $N_{\min} = m + r - 1 = 4$. For $N = 4$ (and $T = 5/N$), we have the plant and the (unique) controller

$$P_H(q) = \frac{q(0.6827 + 2.1745q + 0.0959q^2)}{(1 - 1.1331q)(1 - 0.0148q + 5.53 \times 10^{-4}q^2)} \tag{5.6}$$

$$C(q) = \frac{(1.06 - 0.7214q)(1 - 0.0148q + 5.53 \times 10^{-4}q^2)}{(1 - q)(1 + 1.4095q + 0.061q^2)} \tag{5.7}$$

A plot of the time response for this system is shown in figure 3. In this case, we are unable to meet the specs (5.3). Because there are no degrees of freedom, the response cannot be improved for this value of N.

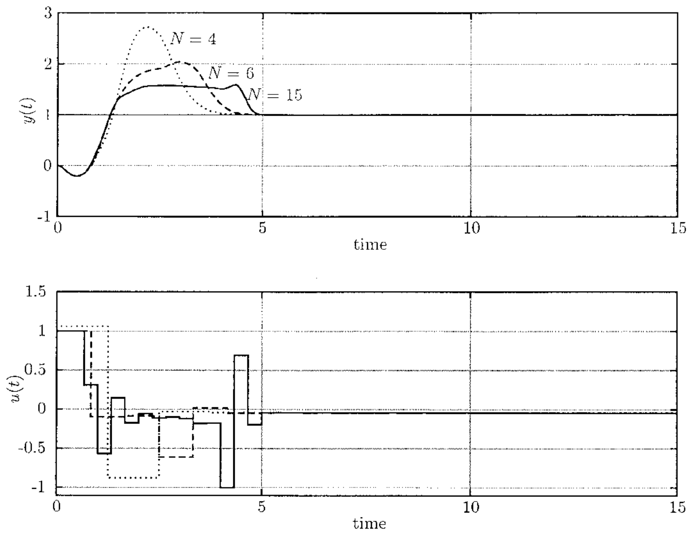

Figure 3. Step response for $N = 4$, $N = 6$, and $N = 10$.

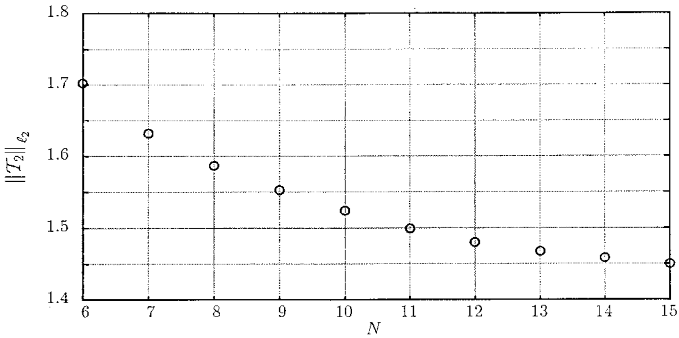

Figure 4. System ℓ_2 norm as a function of N.

We used the LMITOOL (El-Ghaoui *et al.* 1997) and SP software (Vanderberghe and Boyd 1994) to solve this problem. This allows one to incorporate any or all of the specs by specifying the accompanying LMI. Doing so, we varied the number of steps to settle, N, and solved the optimization. It can be shown that the specs can only be met for $N \geq 6$ (the SP software checks for feasibility of the control problem). Figure 3 shows the step responses when $N = 6$ and $N = 15$, as well as $N = 4$. Figure 4 shows a plot of the minimum ℓ_2 norm versus N. This is not the globally optimal ℓ_2 norm, but the minimum norm subject to the other constraints. We see that minimizing the norm served also to reduce the overshoot. All designs met the $t_s \leq 5$ settling time imposed by the choice of N and T.

6. Conclusion

A solution to the optimal ripple-free deadbeat control problem has been obtained. A new parametrization of the controllers had been obtained, and LMI conditions have been provided for overshoot, undershoot, settling time, slew rate, ℓ_1, ℓ_2, ℓ_∞, norms, STAE and STSE costs, as well has $\mathcal{H}_\infty$ norms with respect to two different outputs. Future directions for research include considering the general tracking problem, including systems with time-delays, and are forthcoming.

Appendix

A.1. *Proof of Lemma* 3

We first define

$$D(q) = B(q)C(q) = d_0 + d_1 q + \cdots + d_N q^N + \cdots \tag{A.1}$$

$$D_N(q) = d_0 + d_1 q + \cdots + d_N q^N \tag{A.2}$$

and note that for $0 \leq k \leq n$

$$d_k = b_0 c_k + b_1 c_{k-1} + \cdots + b_{k-1} c_1 + b_k c_0 = \begin{bmatrix} b_k & \ldots & b_0 \end{bmatrix} \begin{bmatrix} c_0 \\ \vdots \\ c_k \end{bmatrix} \tag{A.3}$$

and for $n < k$

$$d_k = b_0 c_k + b_1 c_{k-1} + \cdots + b_{n-1} c_{k-n+1} + b_n c_{k-n} = \begin{bmatrix} b_n & \dots & b_0 \end{bmatrix} \begin{bmatrix} c_{k-n} \\ \vdots \\ c_k \end{bmatrix} \tag{A.4}$$

In order to satisfy $A(q) = D(q)$, we must have

$$d_k = \begin{cases} a_k, & 0 \le k \le m \\ 0, & m < k \end{cases} \tag{A.5}$$

Thus, we must show that $\vec{C}_N = B_x^{-1} \vec{D}_N$.

We now consider two cases: when $N > m$, and when $N \le m$. First, when $N > m$, and for $k = 1$ to N we stack the coefficients d_k and obtain

$$\begin{bmatrix} d_0 \\ \vdots \\ d_m \\ d_{m+1} \\ \vdots \\ d_{N-1} \end{bmatrix} = \begin{bmatrix} a_0 \\ \vdots \\ a_m \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} b_0 & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & b_0 & \ddots & & & \vdots \\ b_n & \vdots & \ddots & 0 & & 0 \\ 0 & b_n & \cdots & b_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots & \ddots & 0 \\ 0 & \cdots & 0 & b_n & \cdots & b_0 \end{bmatrix} \begin{bmatrix} c_0 \\ \vdots \\ c_n \\ c_{n+1} \\ \vdots \\ c_{N-1} \end{bmatrix} \tag{A.6}$$

or in more compact form $\vec{D}_N = B_x \vec{C}_N$. By the assumption that $b_0 = 1$, we see that B_x is invertible, and thus that we have the unique solution

$$\vec{C}_N = B_x^{-1} \vec{D}_N \tag{A.7}$$

Note that to put this into the expanded matrix form, we note that this is the top N rows of the matrix $\underline{B}_N$ which has $N + n + 1$ rows.

When $0 < N \le m$, $\vec{D}_N$ contains a truncated version of $\vec{A}$. However, the matrix B_x has the same basic form as in (A.6), and it is also invertible Thus, equation (A.7) again holds, and we see that B_x is again the top N rows of the matrix $\underline{B}_N$. □

The following result is needed in the proof of Theorem 1 (Ben-Israel 1974).

Lemma 4: *Given $A \in \mathbb{R}^{n_1 \times m_1}$, $n_1 \ge m_1$, which is full rank* ($\operatorname{rank}(A) = m_1$) *and the matrices $x \in \mathbb{R}^{m_1 \times p_1}$ and $y \in \mathbb{R}^{n_1 \times p_1}$. The linear equation $Ax = y$ has a solution for x if and only if the consistency condition*

$$(I_{n_1} - AA^+) y = 0 \tag{A.8}$$

holds. In this case all solutions of the linear equation are given by

$$x = A^+ y + A_\perp b \tag{A.9}$$

for arbitrary $b \in \mathbb{R}^{n_1 - m_1 \times p_1}$, and $A_\perp \in \mathbb{R}^{m_1 \times n_1 - m_1}$ a full rank complement of A such that $AA_\perp = 0$.

A.2. *Proof of Theorem* 1

Note that both sides of the Diophantine equation (1.10) are monic. This is a consequence of making $Q_d(0) = 1$. It is well known that this equation, for fixed N can be written in terms of the *resolvant matrix*† (Barnett 1983, Astrom and Wittenmark 1989)

$$\Psi = \begin{bmatrix} \mathcal{A} & -\mathcal{B}_1 \end{bmatrix} \tag{A.10}$$

where $\mathcal{A}$ and $\mathcal{B}$ have been previously defined. The Diophantine equation may be compactly written as

$$\Psi \Phi = \mathcal{B}_0 \tag{A.11}$$

where

$$\Phi = \begin{bmatrix} \mathbb{N} \\ \mathbb{D} \end{bmatrix} \tag{A.12}$$

is the matrix of unknown parameters. Note that $\mathcal{B}_0$ is previously defined in the theorem. The Diophantine equation has a solution if and only if the linear equation (A.11) has a solution. The resolvant matrix will always be full rank if and only if the polynomials $D_y(q)$ and $N_p(q)$ are coprime. Equation (A.11) can be rewritten

$$\mathcal{A}\mathbb{N} = \mathcal{B}_0 + \mathcal{B}_1 \mathbb{D} \tag{A.13}$$

Thus, the Diophantine equation has a solution if and only if (A.13) has a solution.

By definition, it is easy to see that $\mathcal{A}$ is a full rank matrix. Equation (A.13) may be uniquely solved for $\mathbb{N}$ in terms of $\mathbb{D}$ whenever $m = 1$. When $m > 1$, the solution of (A.13) must satisfy the consistency equation

$$(I_N - \mathcal{A}\mathcal{A}^+)(\mathcal{B}_0 + \mathcal{B}_1 \mathbb{D}) = 0 \tag{A.14}$$

In examining this condition, we note that the resolvant matrix being full rank implies that the matrices $\mathcal{A}$ and $\mathcal{B}_1$ have complimentary left null spaces. We now compute the singular value decomposition

$$\mathcal{A} = U \Sigma V^{\mathrm{T}} \tag{A.15}$$

where $U^{\mathrm{T}} U = U U^{\mathrm{T}} = I_N$, $V^{\mathrm{T}} V = V V^{\mathrm{T}} = I_{p+1}$, and

$$U = \begin{bmatrix} U_1 & U_2 \end{bmatrix}, \qquad \Sigma = \begin{bmatrix} \Sigma_1 \\ 0_{m-1 \times p+1} \end{bmatrix}, \qquad \Sigma_1 > 0 \tag{A.16}$$

From this, we may write the pseudo-inverse

$$\mathcal{A}^+ = V \Sigma^+ U^{\mathrm{T}}, \qquad \Sigma^+ = \begin{bmatrix} \Sigma_1^{-1} & 0 \end{bmatrix} \tag{A.17}$$

Based on this, we define

$$\Sigma_0 = \begin{bmatrix} 0_{p+1} & 0 \\ 0 & I_{m-1} \end{bmatrix} \tag{A.18}$$

and compute

$$\begin{aligned} X_a = I_N - \mathcal{A}\mathcal{A}^+ &= I_N - U \Sigma V^{\mathrm{T}} V \Sigma^+ U^{\mathrm{T}} \\ &= U(I_N - \Sigma\Sigma^+) U^{\mathrm{T}} = U \Sigma_0 U^{\mathrm{T}} \end{aligned} \tag{A.19}$$

† This matrix is sometimes called the *Sylvester matrix*.

From this, we may observe that U_2^{T} spans the left null space of $\mathcal{A}$, i.e. $U_2^{\mathrm{T}}\mathcal{A} = 0$. Since the left null space of $\mathcal{A}$ is complimentary to that of $\mathcal{B}_1$ and since $\mathcal{B}_1$ is full rank, then $U_2^{\mathrm{T}}\mathcal{B}_1$ is full rank (i.e. rank $(U_2^{\mathrm{T}}\mathcal{B}_1) = m - 1$). Thus (A.14) holds if and only if $U\Sigma_0 U^{\mathrm{T}}(\mathcal{B}_0 + \mathcal{B}_1\mathbb{D}) = 0$, holds which is equivalent to $\Sigma_0 U^{\mathrm{T}}(\mathcal{B}_0 + \mathcal{B}_1\mathbb{D}) = 0$. This can be seen to be equivalent to $U_2^{\mathrm{T}}(\mathcal{B}_0 + \mathcal{B}_1\mathbb{D}) = 0$, which can be written

$$U_2^{\mathrm{T}}\mathcal{B}_1\mathbb{D} = -U_2^{\mathrm{T}}\mathcal{B}_0 \tag{A.20}$$

Since $U_2^{\mathrm{T}}\mathcal{B}_1$ is full rank, and since $m - 1 \le \ell$, then $(U_2^{\mathrm{T}}\mathcal{B}_1)(U_2^{\mathrm{T}}\mathcal{B}_1)^+ = I_{m-1}$. The condition (A.20) holds if and only if it is consistent, which is true since

$$0 = (I_{m-1} - (U_2^{\mathrm{T}}\mathcal{B}_1)(U_2^{\mathrm{T}}\mathcal{B}_1)^+)U_2^{\mathrm{T}}\mathcal{B}_0 = (I_{m-1} - I_{m-1})U_2^{\mathrm{T}}\mathcal{B}_0 = 0 \tag{A.21}$$

We now parametrize equation (A.20) by the parameter $\mathcal{R} \in \mathbb{R}^{N-N_{\min}\times 1}$ such that

$$\mathbb{D} = X_0 + X_1\mathcal{R} \tag{A.22}$$

where X_0 is defined in (2.24) and X_1 is a full-rank matrix defined in (2.25). By Lemma 4, this parameterizes all solutions to the linear equation (A.20). In this case, since (A.13) is consistent, then we obtain all solutions of this linear equation as in (2.28). □

References

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

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

Ben-Israel, A., 1974, *Generalized Inverses: Theory and Applications* (New York: Wiley).

Bergen, A., and Ragazzini, J., 1954, Sampled-data processing techniques for feedback control systems. *Transactions of the AIEE Pt. II*, **73**, 236–247.

Doyle, J., Packard, A., and Zhou, K., 1991, Review of lfts, lmis, and μ. *Proceedings of the 30th Conference on Decision and Control*, IEEE, Brighton, UK, pp. 1227–1232.

El-Ghaoui, L., Delebecque, F., and Nikoukhah, R., 1997, 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.

Franklin, G., and Emami-Naeini, A., 1986, Design of ripple-free multivariable robust servomechanisms. *IEEE Transactions on Automatic Control*, **AC-31**, 661–664.

Hartley, T., Veillette, R., and Cook, G., 1996, Techniques in deadbeat and one-step-ahead control. In C. T. Leondes (Ed) *Control and Dynamic Systems—Advances in Theory and Applications*, Vol. 79 of *Digital Control Systems Implementation and Computational Techniques* (New York: Academic Press), chapter 3.

Kučera, V., 1979, *Discrete Linear Control—The Polynomial Equation Approach* (New York: Wiley).

Nesterov, Y., and Nemirovskii, A., 1994, *Interior Point Polynomial Algorithms in Convex Programming*, Vol. 13 of *Studies in Applied Mathematics* (Philadelphia, PA: SIAM).

Porter, A., and Stoneman, F., 1950, A new approach to the design of pulse-monitored servo systems. *Proceedings of the IEE, Pt. II*, **97**, 597.

Sirisena, H., 1985, Ripple-free deadbeat control of siso discrete systems. *IEEE Transactions on Automatic Control*, **AC-30**, 168–170.

Urikura, S., and Nagata, A., 1987, Ripple-free deadbeat control for sampled data systems. *IEEE Transactions on Automatic Control*, **AC-32**, 474–482.

Vandenberghe, L., and Boyd, S., 1994, **SP**: Software for semidefinite programming, user's guide. Available via anonymous ftp from isl.standford.edu, under the directory, /pub/boyd/semidef_prog.