Scholarly Journal of Mathematics and Computer Science Vol. 1(2), pp. 19-29, October 2012
Available online at http:// www.scholarly-journals.com/SJMCS
ISSN 2276-8947 ©2012 Scholarly-Journals

Full Length Research Paper

Multivariable Ripple-Free Deadbeat Control

Hatem ELAYDI[1] and Fadi ALBATSH[2]

[1] Electrical Engineering Department, Islamic University of Gaza, P.O.BOX 108, Gaza, Palestine
[2] Computer Technology and Industrial Professions Department, University College of Applied Sciences
P.O.BOX:1415, Gaza, Palestine

Accepted 31 March 2012

The design of multivariable ripple-free deadbeat controllers is a complex task. One approach which has shown promise for solving multivariable ripple-free deadbeat control (MRFDC) problems is the use of Diophantine equation parameters. The problem of solving robust multivariable ripple free deadbeat with time delays has not been solved. This paper proposes a hybrid two degree of freedom controller utilizing the parameterization of Diophantine equation to build a multivariable ripple-free deadbeat control (MRFDC). The important feature of the proposed approach is that it combines the concept of multivariable input with robust ripple- free deadbeat control. Simulation results show that the output signal tracks the input sinusoidal signal in short settling time.

Keywords: Single rate, Multivariable Ripple – Free Deadbeat Control (MRFDC), Diophantine equations.

INTRODUCTION

The study of deadbeat control started in 1950’s. Deadbeat control makes the output of the systems tracks the reference input signal in a finite period of time. Deadbeat control achieves exact settling after a finite number of discrete sampling instants; however, there may exist undesirable intersample ripple in the continuous response. To obtain a ripple – free deadbeat design, a continuous internal model with no plant zeros cancellation is a requirement (Franklm and Emami-naeini, 1986).

Deadbeat control plays a big role in tracking systems since the controller takes the system from any initial state to the origin in a minimum number of time steps.

The attractive feature for the deadbeat response is that the tracking error goes down to exactly zero in a finite number of control steps. However, ripples may appear in the plant output between sampling instants. The ripple source comes from cancellation of plant zeros by controller poles results in controller modes that may be excited by the reference signal (Yamada and Funahashi, 1998). On the other hand, a deadbeat control has three disadvantages in practical field: the robustness issue, the disturbance source in the continuous plant, and the efficiency of the tracking performance in the transient response (Sirisena, 1985).

Solutions to multivariable and ripple-free deadbeat control had been presented by several researchers. Elaydi and Paz, 1998, presented an Optimal Ripple-Free Deadbeat Controllers for Systems with Time Delays. A ripple free deadbeat controller for a system with time delays was proposed. Matrix parameterization of the Diophantine equation was the approach used to solve this problem. However, they didn't tackle the multivariable problem. Ito, 2001, improved performance of deadbeat servomechanism by means of multirate input control. A state-space approach was designed to solve a multirate input control problem. Multirate input mechanism achieved shorter settling time than single rate control using the same frequency of sampling. However, multirate control often exhibited intersample ripple and this problem was solved here. Furthermore, the proposed method guaranteed robustness against continuous-time model uncertainty and disturbance. However, they didn't deal with various reference input signals and the input signal was a step signal only.

Paz, 2006. proposed a ripple-free tracking approach with robustness. A hybrid two-degree-of freedom (2DOF) controller considering optimization problem and robustness was proposed. This controller was given in terms of the solution of two Diophantine equations.

Corresponding Author Email: helaydi@iugaza.edu.ps

However, he didn't tackle the multivariable problem. Salgado and Oyarzun, 2007, proposed two objective optimal multivariable ripple-free deadbeat control. A parameterization of all stabilizing ripple-free deadbeat controller was given. Also, optimized control signal and tracking performance were kept minimal using quadratic index. However, they didn't tackle the time delay problem and they didn't deal with various reference input signals. Moreover, the input signal was just a step signal and the paper never dealt with the robustness issue.

This paper presents new methodology for designing multivariable ripple – free deadbeat controller to solve the tracking of an arbitrary reference signal and deals with the robustness issue. Ripple – free deadbeat tracking is formulated based on the solution of the Diophantine equation. The ripple – free deadbeat tracking formulation is based on Paz results which delt with robustness (Paz, 2006) This paper also takes advantage of Salgado and Oyarzun approach which solved the multivariable problem (Salgado and Oyarzun, 2007). The multivariable ripple – free deadbeat control, based on the solution of the parametrization of the Diophantine equation, is proposed. The approach presented in this paper can handle systems with time delays, where the time delay is not an integer multiple of the sampling time. A discretizing form of the system with the time delay is obtained and used in the controller design. The input signal is not restricted to step but it can be any arbitrary reference signal.

This paper is organized as follows: section 2 discusses the Diophantine Equation parameterization, section 3 presents the solution of the ripple-free deadbeat control problem, section 4 introduces the multivariable ripple – free deadbeat control, section 5 shows the proposed approach including MATLAB simulation, and the last section concludes this paper.

Diophantine Equations Parameterization

The Diophantine equation plays an important role in the design and synthesis of controllers in the frequency domain. The Diophantine equation has an infinite number of solutions that all provide an internally stabilizing controller. However, the Diophantine equation in a polynomial form masks its design freedom. A parameterization of the Diophantine equation is obtained, allowing simple access to the degrees of freedom. Polynomial multiplication and division is given as matrix multiplication. The parameterization of the Diophantine equation is based on obtaining solution in a matrix form (Elaydi, 1998).

Methods for solving the Diophantine equation

Given the two polynomials, *A(q) and B(q)*, with

$$A(q) = a_o q^n + a_1 q^{n-1} + a_n, \text{ and } a_o \neq 0 \quad (1)$$

$$B(q) = b_o q^m + b_1 q^{m-1} + b_m, \text{ and } m < n \quad (2)$$

and given two more polynomials $Q_n(q)$, $Q_d(q)$ *and* $C(q)$ is defined as

$$C(q) = A(q)Q_n(q) + B(q)Q_d(q) \quad (3)$$

The polynomial equation (3) is called the Diophantine equation. This equation is linear in terms of the polynomials $Q_n(q)$ *and* $Q_d(q)$ for the known polynomials *A(q), B(q)* and *C(q)* where *A(q)* and *B(q)* are coprime. The coprimeness of *A(q) and B(q)* guarantees the existence of a solution to the above equation for any arbitrary *C(q)*. There are several methods for solving the Diophantine equation such as: Euclidean algorithm, Sylvester's resultant, Bezout's resultant, and MacDuffee's resultant (Elaydi, 1998, Kucera, 1980).

The solution of the Diophantine equation, using the resolving matrix is not compact for optimization purposes. The former parameterization does not lend itself well characterizing control constraint conditions that arise in practice. Since the Diophantine equation has two polynomial products, the vectorization is a convenient method to express the equation.

Matrix Representation of Polynomial Products

Suppose the polynomials *A(q) and B(q)* are given such that ((Elaydi, 1998, Fan and Chang, 1990)

$$A(q) = a_0 + a_1 q + a_2 q^2 + \ldots\ldots + a_m q^m \quad (4)$$

$$B(q) = b_0 + b_1 q + b_2 q^2 + \ldots\ldots + b_n q^n \quad (5)$$

The vectorized form of the polynomials is defined as

$$\vec{A} = \begin{bmatrix} a_0 \\ \vdots \\ a_m \end{bmatrix}, \vec{B} = \begin{bmatrix} b_0 \\ \vdots \\ b_n \end{bmatrix} \quad (6)$$

Indeed, any polynomial may be vectorized this way. This vectorization may be expressed as an operator. The expanded matrix form is also defined as

$$\bar{A}_p = \begin{bmatrix} \vec{A} & & 0 \\ & \ddots & \\ 0 & & \vec{A} \end{bmatrix} \in \mathbb{R}^{m+p+1 \times p}, \ \bar{B}_p = \begin{bmatrix} \vec{B} & & 0 \\ & \ddots & \\ 0 & & \vec{B} \end{bmatrix} \in \mathbb{R}^{n+p+1 \times p} \quad (7)$$

Lemma 1:

$\overrightarrow{AB}$

For vector cab be expressed as:

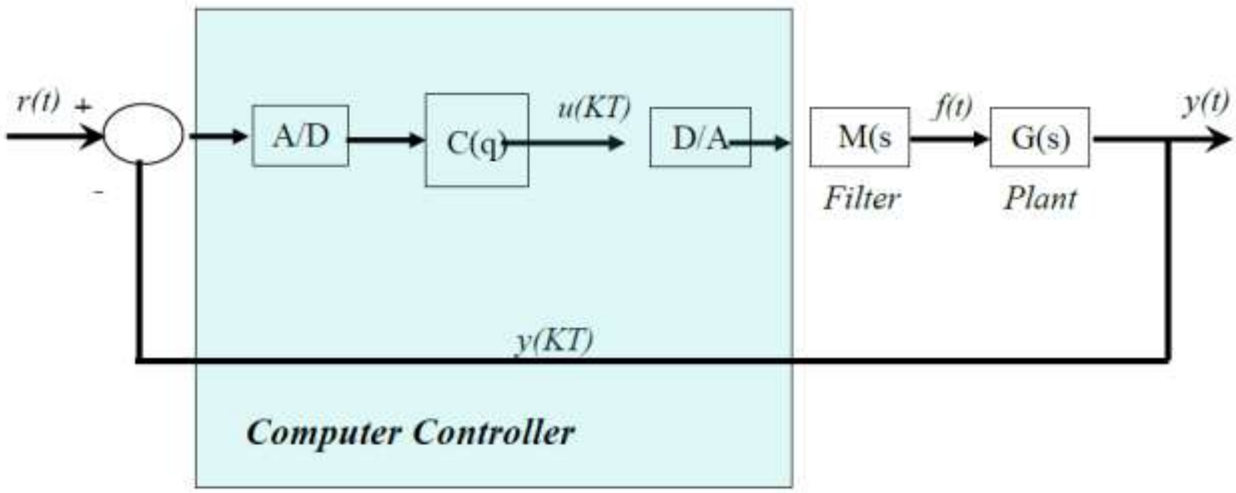

Figure 1: Hybrid 2DOF control configuration.

$$\left(\overrightarrow{AB}\right)=\bar{A}_{n+1}\vec{B}=\bar{B}_{m+1}\vec{A} \tag{8}$$

Proof of Lemma 1 can be found in (Elaydi, 1998). Lemma 1 illustrates the fact that the vectorization of 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 that *B(0)=1*, the polynomial division can be considered now

$$C(q)=\frac{A(q)}{B(q)}=c_0+c_1q+c_2q^2+..... \tag{9}$$

Noting that equation (9) may also be written as

$$A(q)=B(q)C(q) \tag{10}$$

Noting that the left hand side of equation (10) has at most *m+1* nonzero term; thus, the right hand side must also have the same number of term. Thus, restricting attention to a finite version of the sequence, the approach now considers the first *N* coefficients of *C*. The truncated version of this can be written as:

$$C_N(q)=c_0+c_1q+....+c_{N-1}q^{N-1} \tag{11}$$

Solving Ripple-Free deadbeat Control Problem

System demonstration

Consider the 2DOF hybrid system as shown in Figure 1 with the control system showing the continuous-time plant *G(s)* and the 2DOF discrete time controller *C(q)* (where $q=z^{-1}$) and z is the Z- transform variable along with a tracking filter *M(s)* (Paz, 2008).

The ripple-free deadbeat control problem definition

The ripple-free control problem has several goals (Sirisena, 1985). The closed-loop system is internally stable. The error of the system $e(t)=r(t)-y(t)=0$ for all $t \geq N_sT$, where N_s is the number of steps to settle. The control signal *u(KT)* settles down after a finite number of steps. It may be necessary to impose constraints on the transient response that arise from practical implementation issues. To find the control signal for the 2DOF system, we discretize the reference signal *R(s)*, the plant *P(s)* and the filter *M(s)* then rearrange the block diagram for the system as shown in Figure 2. The controller may be realized using the configuration shown in Figure 2. We note that $N_1(q)$ *and* $N_2(q)$ may be implemented as FIR filters. A polynomial in *q* corresponds to a rational function in z with all the poles at the origin, where ($q=z^{-1}$).

Now we can find the 2DOF control law as the following form

$$U(q)=\frac{1}{D_c(q)}V(q) \tag{12}$$

$$V(q)=N_1(q)R(q)-N_2(q)Y(q) \tag{13}$$

From (12) and (13), we get the transfer function of the control signal U(q) as follow:

$$U(q)=\frac{N_1(q)R(q)-N_2(q)Y(q)}{D_c(q)} \tag{14}$$

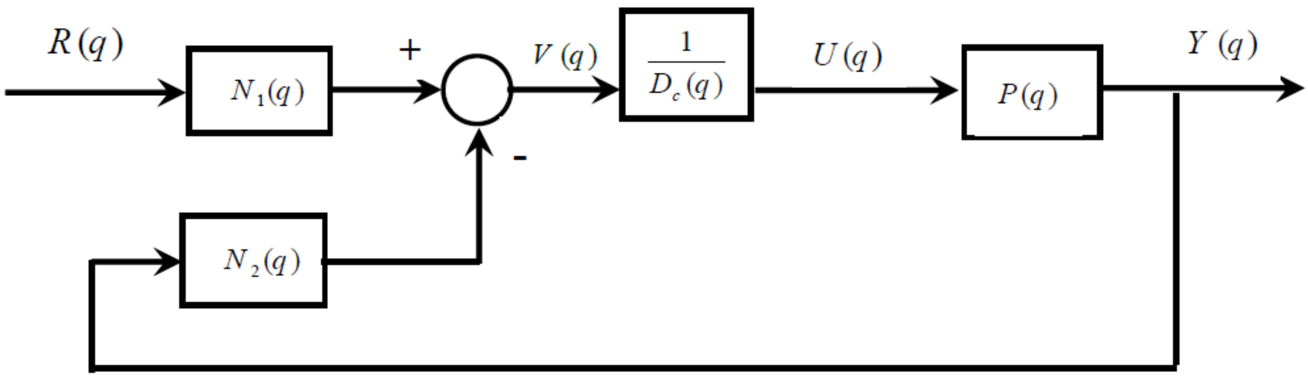

Figure 2: Implementation of the Deadbeat controller.

Ripple free deadbeat control problem (RFCP)Solution

The controller polynomials are obtained by solutions of the Diophantine equations (6)

$$N_p(q)N_1(q)+D_r(q)Q_1(q)=1 \quad (15)$$

$$N_p(q)N_2(q)+D_p(q)D_c(q)=1 \quad (16)$$

Where Q_1 is a polynomial.
The solution of the RFCP requires the solution of two Diophantine equations. Any N_1, N_2 *and* D_c that satisfy the above Diophantine equations give a solution to the RFCP. Since the Diophantine equation has an infinite number of solutions, we will seek specific solutions that provide desired response.
Our two Diophantine equations have solutions as follow

$$N_1(q)=N_{1\min}(q)-D_r(q)v_1(q) \quad (17)$$

$$Q_1(q)=Q_{1\min}(q)+N_p(q)v_1(q) \quad (18)$$

$$N_2(q)=N_{2\min}(q)-D_p(q)v_2(q) \quad (19)$$

$$D_c(q)=D_{c\min}(q)+N_p(q)v_2(q) \quad (20)$$

While v_1 *and* v_2 are arbitrary polynomials of degrees that define the degree of freedom in the design or free parameters. If we applied the controller in (18), we obtain the transfer function

$$\frac{Y(q)}{R(q)}=\frac{\left(N_1/D_c\right)\left(N_P/D_P\right)}{1+\left(N_2/D_C\right)\left(N_P/D_P\right)}=\frac{N_1N_P}{N_2N_P+D_PD_C}=N_1N_P \quad (21)$$

by (14) we also have the error signal

$$\begin{aligned}E(q)&=R(q)-Y(q)=(1-N_1N_P)R(q)\\&=(D_rQ_1)\frac{N_r}{D_r}=N_rQ_1\end{aligned} \quad (22)$$

Which is a polynomial implying that *e(KT)=0* for $K\geq N_s=deg(N_rQ_1)$
Using (16) when $N_p(q)N_2(q)+D_p(q)D_c(q)=1$, we have the control signal transfer function

$$\frac{U(q)}{R(q)}=\frac{\left(N_1/D_C\right)}{1+\left(N_2/D_C\right)\left(N_P/D_P\right)}=\frac{D_PN_1}{N_PN_2+D_PD_C}=D_PN_1 \quad (23)$$

Multivariable ripple-free deadbeat control

Salgado and Oyarzun, 2007, presented a new method to demonstrate a multivariable ripple-free deadbeat control. Their study was applied to discrete-time, stable, linear and time invariant plant model. A simple parameterization of all stabilizing ripple-free deadbeat controllers of a given order was considered. The free parameter was then optimized in the sense that a quadratic index is kept minimal. The optimality criterion had the advantage of accounting for both tracking performance and magnitude of the control effort. A control strategy leads to settle the tracking error sequence to zero in a minimum number of time steps based on pole zero cancellations between controller and plant model. The basic idea behind this approach is in order to avoid any intersample ripple after the settling time; the control sequence must also reach its steady state in, at most, the same number of samples. Dealing with a combined optimality criterion and ensuring a MIMO ripple-free deadbeat response were presented in this approach.

Assumptions and definitions

We consider the plant model *G(q)* to be a stable $p\times p$ discrete-time transfer matrix. We will assume that *G(q)* is represented in right coprime polynomial matrix fraction description as

$$G(q)=B(q)A(q)^{-1} \quad (24)$$

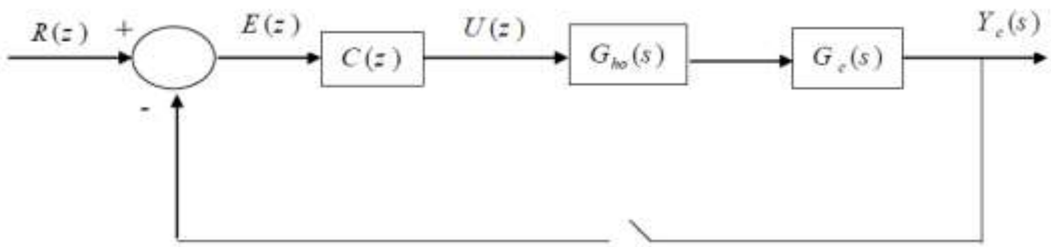

Figure 3: Multivariable Sampled data control loop

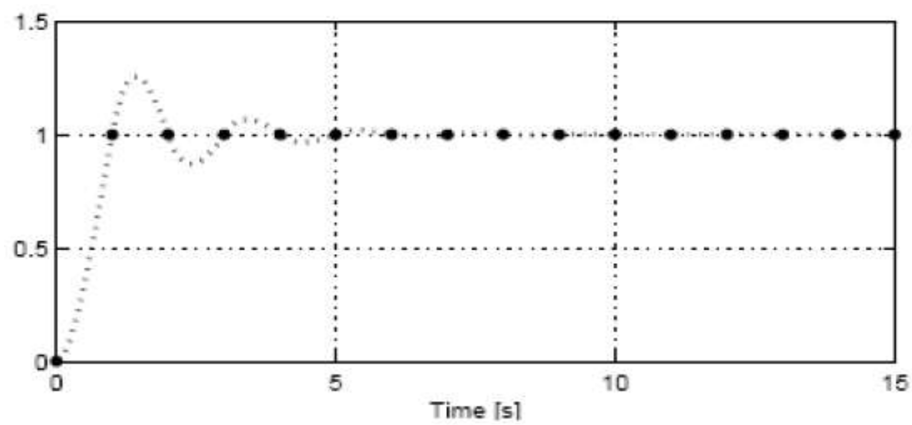

Figure 4: Continuous-time deadbeat response with ripple.

where *A(q) and B(q)* are right coprime polynomial matrices of dimension *p×p*. Coprime polynomial factorizations of transfer matrices can be implemented using many methods such as in (Wang and Davison, 1973, Patel, 1981). We assume that *G(q)* has no zeros on *q=1, i.e. B(1)* is non-singular and we further assume that *B(1)=I*. The non-singularity of *B(1)* is a standard condition necessary for being able to track constant reference signals. Given a proper transfer matrix *M(q)*, we define the right degree interactor (RDI) of *M(q)* as a polynomial matrix *E(q)* such that the product *M(q)E(q)* is biproper, i.e. (Silva and Salgado, 2005)

$$\lim_{q \to 0} M(q)E(q) = D$$

where $0 < \det\{D\} < \infty$

System demonstration

Consider a continuous-time plant with transfer function *Gc(s)* which is digitally controlled through a zero order sample and hold device with transfer function *Gho(s)*, by a linear discrete-time feedback controller with transfer function, *C(z)*. We can present sampled data control loop system in the block diagram as shown in Figure 3. Now, we will find the transfer function of the sampled data open loop system as follow:

$$G(q) = B(q)A(q)^{-1} \quad (26)$$

where *B(q) and A(q)* are right coprime polynomial matrices. Given that the reference vector signal is assumed to be step function, i.e. *r(k)= vμ(k), v* $\in \mathbb{R}^p$ then having a ripple-free deadbeat control loop means that $y_c(t)$ satisfies

$$y_c(t) = \psi \quad , \forall t > NT \quad (27)$$

Where $N \in \mathbb{N}$ is called the deadbeat horizon of the control system. A controller will be designed to achieve a ripple-free deadbeat response which provides perfect steady state tracking at D.C. and makes the output of the plant to settle in a finite number of samples, while avoiding any intersample ripple beyond the deadbeat horizon. Ripple in a deadbeat response arises when the controller cancels the minimum phase zeros of *G(q)*. Those cancelled zeros appear as closed loop poles and generate the intersample response of the continuous-time output. The response of the system can be shown in Figure 4, where it is noticeable that, although the sampled output settles in one sample, the continuous-time output exhibits considerable ripple. The discrete-time model usually contains sampling zeros located in the negative real axis; therefore its cancellation leads to oscillatory modes in the deadbeat response. To avoid the intersample ripple a sufficient condition must be applied to the control sequence which settles in N_s samples. This condition is equivalent to

$$u(k)=u_{ss}, \quad \forall k > N \tag{28}$$

where the constant vector u_{ss} is the steady state value of the control sequence

MIMO ripple-free deadbeat controllers

The ripple-free condition equation (28) implies that the control sensitivity must have the form

$$S_u(q)=K(q)q^N \tag{29}$$

where the deadbeat horizon $N \in \square$ and *K(q)* is a polynomial matrix with degree such that $S_u(q)$ is proper. On the other hand, the tracking error signal must also settle in a finite horizon, so that the complementary sensitivity function

$$T(q)=G(q)S_u(q)=B(q)A(q)^{-1}K(q)q^N \tag{30}$$

must have all its poles at $q^{-1} = 0$. This means that *K(q)* must be factored as *K(q)= A(q)V(q)* with *V(q)* being a polynomial matrix. Thus

$$S_u(q)=A(q)V(q)q^N \tag{31}$$

For convenience, we set *N=n+ 1,* where *n* is the degree of *A(q)* and Let also *V(q)* be written as *V(q) = E(q)W(q)*, with *E(q)* being a RDI of *A(q)* q^N, hence making *A(q)E(q)* q^N biproper. These definitions lead to

$$S_u(q)=A(q)q^n E(q)W(q)q^{\ell} \tag{32}$$

from where we have that the properness of $S_u(q)$ depends on the properness of $W(q)q^{\ell}$. It is worth noting that the degree of *A(q)* is always equal to that of *A(q)E(q)*. The form of *T(q),* $S_u(q)$ and the multivariable deadbeat controller *C(q)* can then be obtained simply as

$$T(q)=B(q)E(q)W(q)q^N \tag{33}$$

$$S_u(q)=A(q)E(q)W(q)q^N \tag{34}$$

$$C(q)=q^N S_u(q)\left(q^N I - T(q)\right)^{-1} \tag{35}$$

$$C(q)=q^N S_u(q)\left(q^N I - T(q)\right)^{-1} \tag{36}$$

$$N = n + \ell$$

Since we also need perfect steady state tracking of constant references, we must force *T(1)=I*, which, using (33) and the fact that *B(1)=I*, implies that $W(1)=E(1)^{-1}$. The controller given in (36) is then a general form of a MIMO deadbeat controller for stable plants and constant reference signals. We need to build the RDI *E(q)* which will be used in (36). Using this formulation we have the advantage to provide a unitary *E(q)* that also satisfies *E(1)=I*, which certainly simplifies the condition imposed on *W(q)* to *W(1)=I*. In the sequel, we will always assume that *E(q)* is a unitary RDI (Silva and Salgado, 2005). Moreover, from (34) it is clear that since, then the minimum deadbeat horizon is $N_{min} = n$, that is, the degree of *A(q)*. This is the multivariate version of the fact that for SISO systems, the minimum deadbeat horizon is given by the plant order. Next lemma gives a characterization of all polynomial matrices *W(q)* that yield the minimum horizon ripple-free deadbeat controller.

Lemma 2: Consider a stable transfer matrix *G(q)* and the MIMO deadbeat controller of (34). Then the minimum horizon deadbeat controller is given by:

$$C_{\min}(q)=\frac{A(q)E(q)}{I-B(q)E(q)} \tag{37}$$

$$W(q)=q^{-\ell}I, \forall \ell \in \square_0 \qquad (36).$$

and it is achieved by choosing in Proof of Lemma 2 can be found in (Salgado and Oyarzun, 2007).

Proposed methodology

In this section, a multivariable ripple free deadbeat control is introduced. A combination between the robust ripple-free deadbeat approach proposed by (Paz, 2006) with the multivariable ripple-free deadbeat approach proposed by (Salgado and Oyarzun, 2007) is proposed for simulation of single rate ripple-free deadbeat control. Aftere that a MATLAB code and a block diagram in Simulink to simulate the multivariable ripple-free deadbeat control were developed. The code for multivariable consists of several functions; one of them is the computing of the Diophantine equation parameters.
To illustrate the proposed controller design procedure in this paper, consider the continuous time multivariable plant model [7] of 2nd order such as:

$$G_c(s)=\begin{bmatrix} \frac{24}{s^2+6s+5} & \frac{108}{s^2+11s+30} \\ \frac{-162}{s^2+11s+30} & \frac{30}{s^2+7s+6} \end{bmatrix} \tag{38}$$

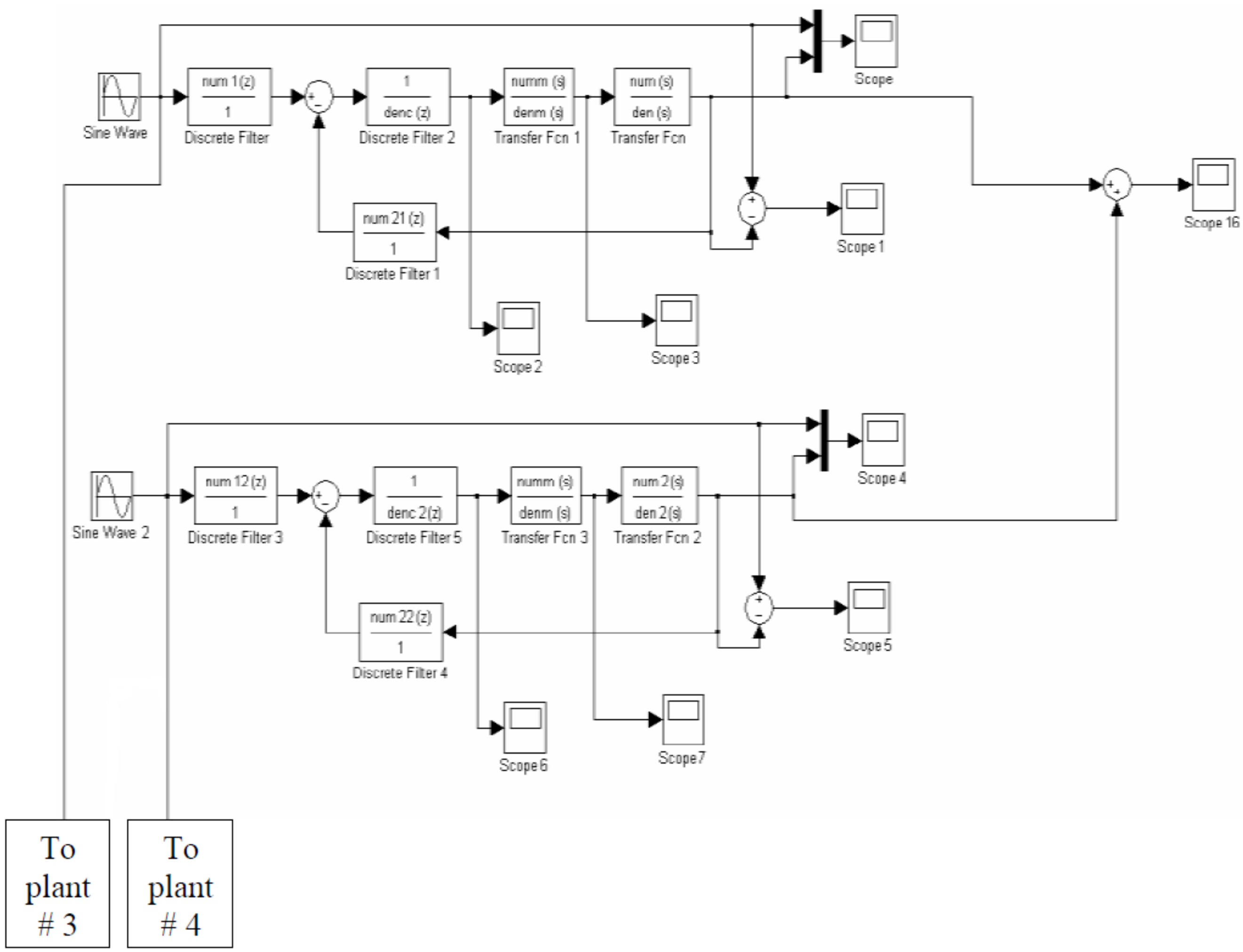

Figure 5: Overall system using Simulink.

The multivariable system has two inputs and two outputs with a transfer function matrix of 2x2. The state equation of an LTI system in state-variable form is (Soliman and Srinath, 1998)

$$\dot{x}(t) = A\,x(t) + B\,u(t) \tag{39}$$

$$y(t) = C\,x(t) + D\,u(t) \tag{40}$$

An overall system is presented using MATLAB Simulink as shown in Figure 5. The goal of the controller is to track the two input sinsodial signals. In Figure 5, we can see how we represent the two inputs by applying the concept in (39 and 40). Also, we divided the four transfer function of the plant by applying the concept in (39 and 40). Assume the transfer function of the plant is in the form of

$$\begin{bmatrix} T_1 & T_2 \\ T_3 & T_4 \end{bmatrix} \tag{41}$$

Where

$$T_1 = \frac{24}{s^2 + 6s + 5} \tag{42}$$

$$T_2 = \frac{108}{s^2 + 11s + 30} \tag{43}$$

$$T_3 = \frac{-162}{s^2 + 11s + 30} \tag{44}$$

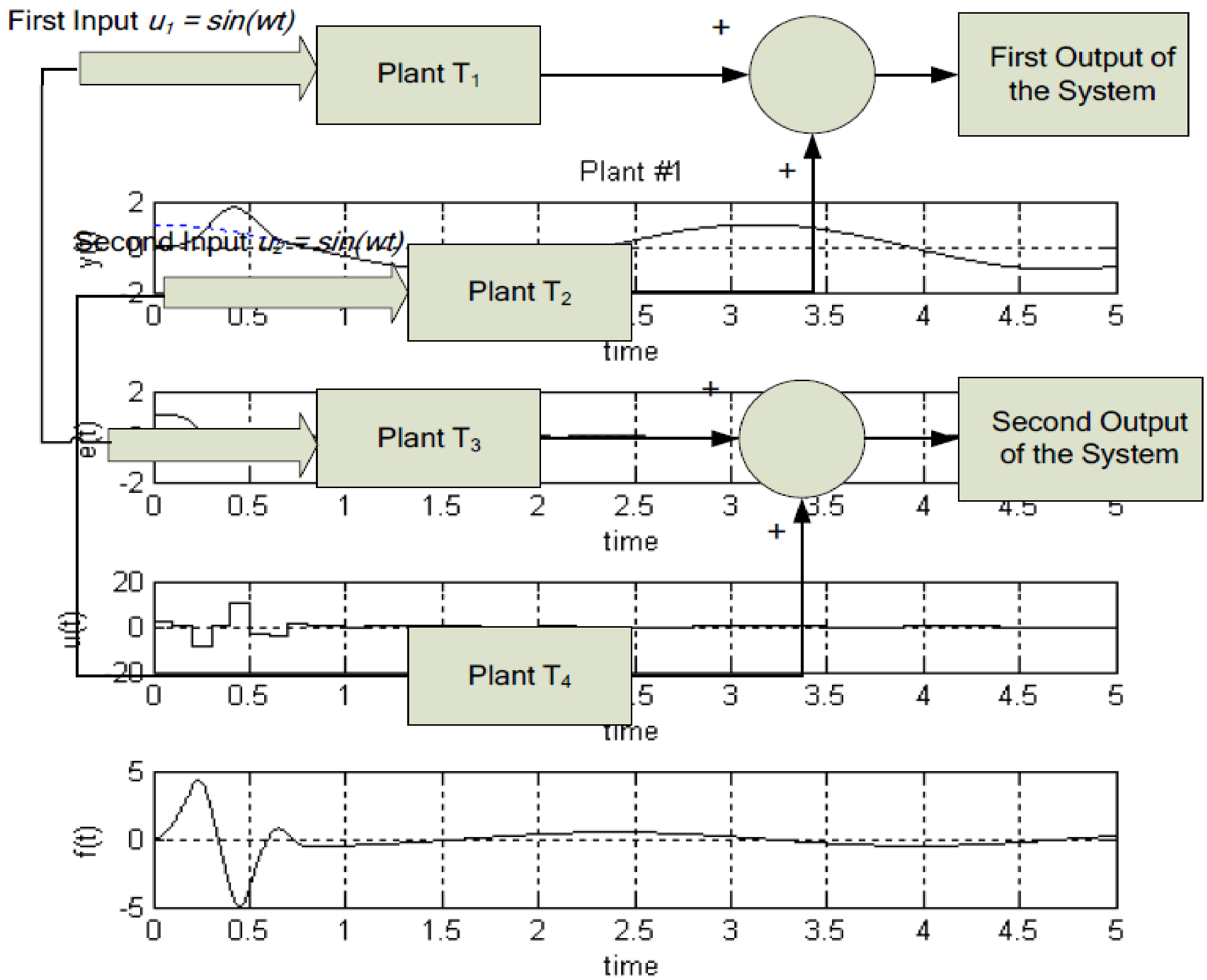

Figure 6: Multivariable system with two inputs and two outputs.

$$T_4 = \frac{30}{s^2 + 7s + 6} \quad (45)$$

The following steps must be followed to obtain the controller. First step is finding the discrete-time reference signal. The second step is computing the discrete-time model for every plant. We choose a sampling time of 0.1 second inline with Salgado and Oyarzun, 2007. For multivariable plant in (38), the discrete-time model for all plants is computed as follow:

$$G(q) = \begin{bmatrix} \frac{0.0988q + 0.081q^2}{1 - 1.512q + 0.549q^2} & \frac{0.3775q + 0.262q^2}{1 - 1.56q + 0.333q^2} \\ \frac{-0.5663q - 0.392q^2}{1 - 1.156q + 0.333q^2} & \frac{0.119q + 0.095q^2}{1 - 1.156q + 0.333q^2} \end{bmatrix} \quad (46)$$

A MATLAB code is used to compute the Diophantine equation parameters. $N_1(q)$, $N_2(q)$ and $D_c(q)$ are used for finding the controller as shown in (14). Then, the ripple-free deadbeat control for the first plant in the system is obtained. The other ripple- free deadbeat control for the rest of other plant components is obtained in similar manners. Figure 6. Shows the multivariable system with two inputs and two outputs giving a plant consisting of 4 transfer function matrix of size 2×2. The input consisting of $u_1 = sin(wt)$ and $u_2 = sin(wt)$. The output of the system consists of two parts: the first part combines the output of the plant T_1 and the output of the plant T_2 while the second part combines the output of the plant T_3 and the output of the plant T_4.

From Figure (7) to (10), we can see that the output signals track the input sinusoidal signals in minimum settling time (about 0.7 second). Also, we can see that

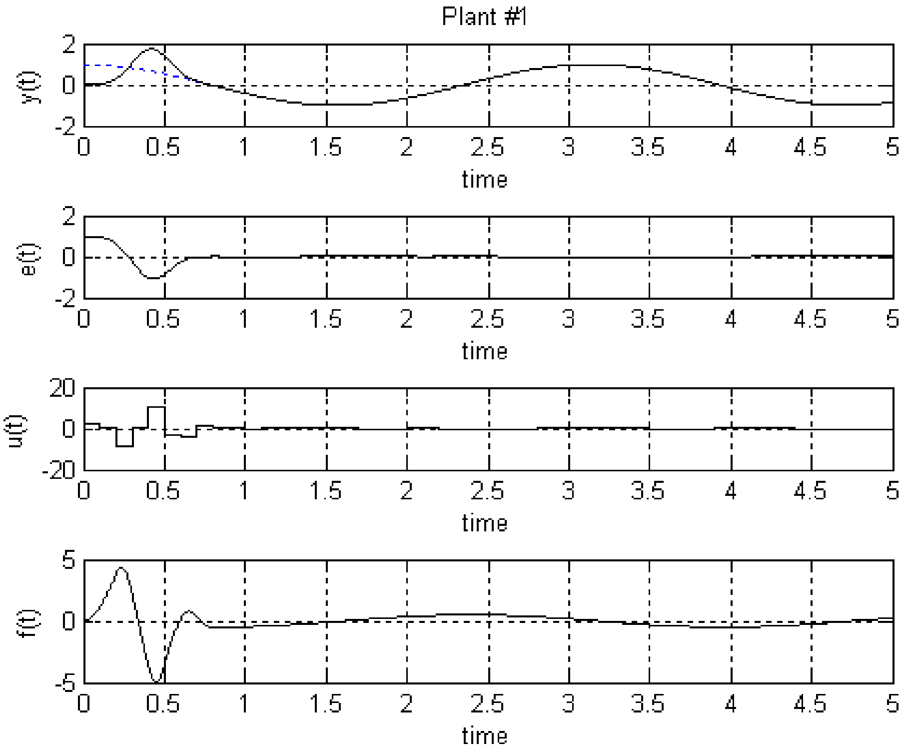

Figure 7: Time response for plant #1

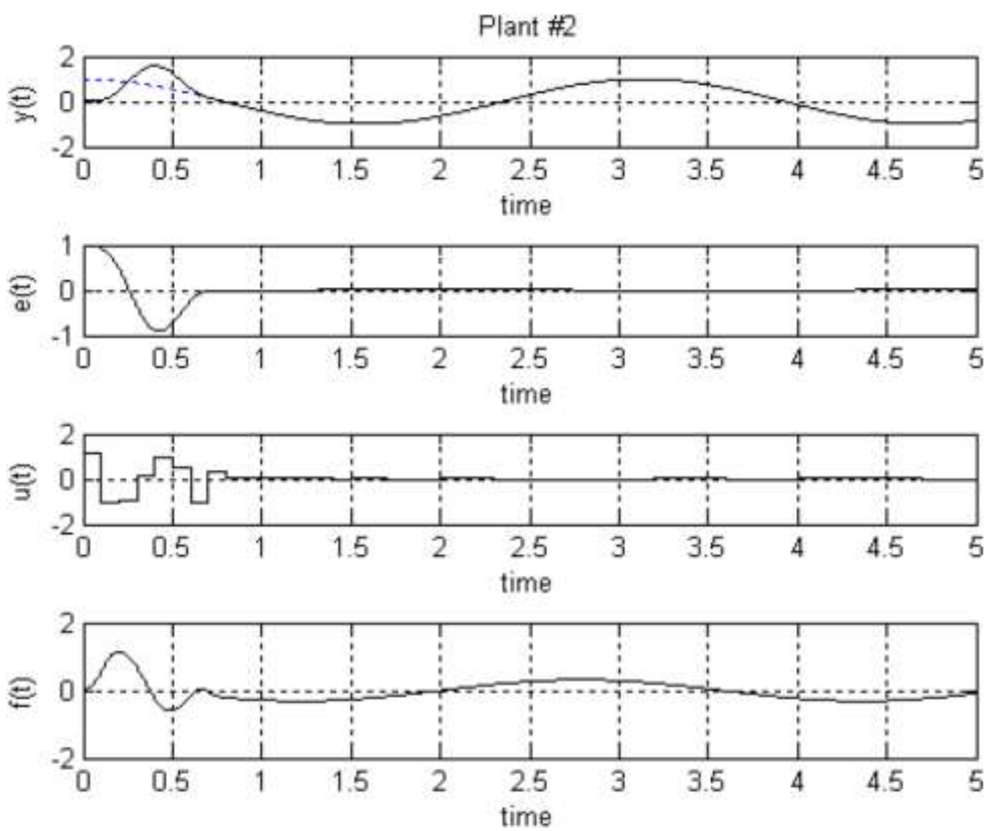

Figure 8: Time response for plant #2

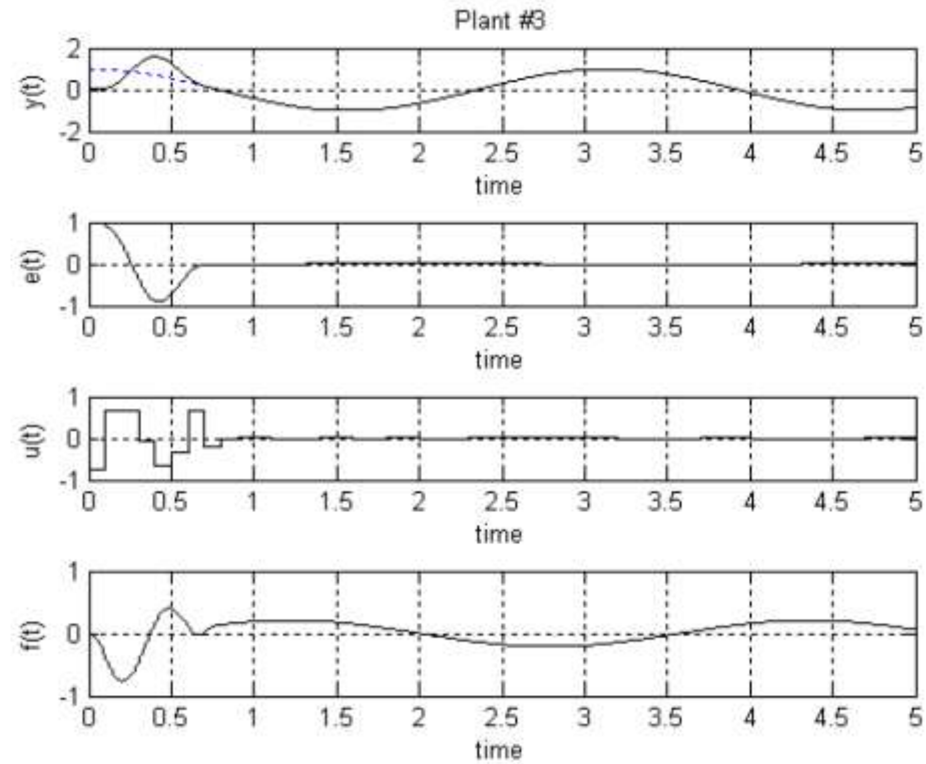

Figure 9: Time response for plant #3

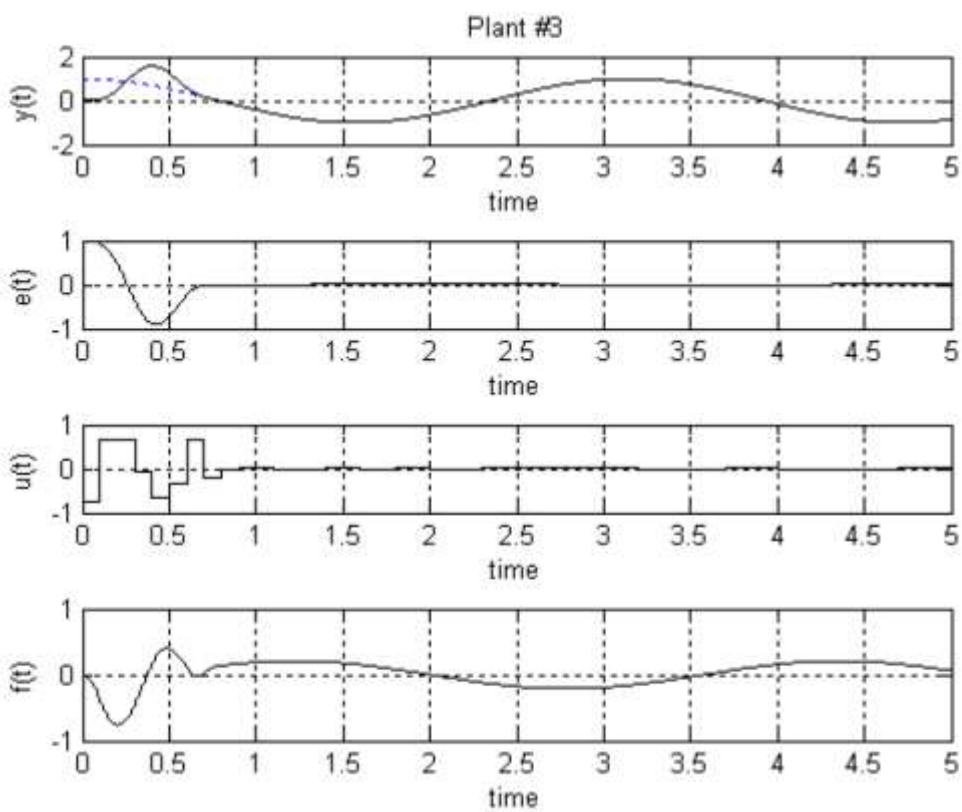

Figure 10: Time response for plant #4

Table 1: System preformance

	Plant 1	Plant 2	Plant 3	Plant 4
Overshoot	74 %	58 %	57 %	71 %
Settling time	0.709 s	0.702 s	0.69 s	0.71 s
Rise time	0.132 s	0.134 s	0.135 s	0.136 s
Steady state error	0	0	0	0

the error signal, *e(t),* and the control signals, *u(t),* for every plant settles down in finite number of steps . Figure 7-10 show the time response for the plants, T_1- T_4. We note that the output signal tracks the input sinusoidal signal in short settling time. It's clear that the response is a periodic and the time domain performance for the output signal is illustrated below: Table 1 summarizes the system performance in terms of the overshoot, settling time, rise time , and steady state error.

Thus, the results show that the output was able to track the sinusoidal inputs in a finite number of steps while driving the tracking error to zero. The overshoot is at an acceptable level and the settling and rising times are fine. These results show improvement to the results obtained by both Paz, 2008, Salgado and Oyarzun, 2007.

CONCLUSION

A new approach shows promise for solving robust multirate ripple-free deadbeat control (MRFDC) problems using Diophantine equation parameterization. This approach is based on the combining both the concept of multivariabl and robust single rate. This paper proposed a hybrid two degree of freedom controller for the fixed-order constrained optimization problem addressing performance and robustness specifications utilizing the parameters of Diophantine equation to build a robust multirate ripple-free deadbeat control.

Simulation results showed that the output signal tracked the input sinusoidal signal in short settling time either in single rate or multivariable setting. Also the ripple problem which caused by intersample was solved. The time domain specification for the output signal, control signal, error signal and the output of the filter signal were computed and satisfied the specifications.

REFERENCES

Elaydi, H (1998). Complex Controller Design Using Linear Matrix Inequalities, Ph.D Thesis, New Mexico State University, USA,.

Elaydi, H, Paz, R (1998). “Optimal Ripple-Free Deadbeat Controllers for Systems with Time Delays”, Philadelphia, Pennsylvania, Proceedings of the American Control Conference, 71(6):1087-1104,

Fan, C, Chang, F (1990). “A Novel Approach for Solving Diophantine Equations”, IEEE Transactions on circuits and systems, 31(11).

Franklm, G, Emami-naeini, A (1986). “Design of Ripple-Free Multivariable Robust Servomechanisms”, IEEE Transactions on automatic control, 31(7).

Ito, H (2001). “Improving performance of deadbeat servomechanism by means of multirate input control”, Department of Control Engineering and Science, Kyushu Institute of Technology, Japan, Access on 1/11/2008, http://search2.libi.kyutech.ac.jp /TR/pdf/csse pdf. p.14.

Kucera, V (1980). “A Deadbeat Servo Problem”, Int. J. Contr., 36: 107-113.

Patel, R (1981). “Computation of Matrix Fraction Descriptions of Linear Time- invariant Systems”, IEEE Transactions on automatic control, 26(1).

Paz, R (2006). “Ripple-free tracking with robustness”, Int. J. contr., 79 (6).

Paz, R (2008). Computer Controlled Systems, Lecturer notes, New Mexico state university.

Salgado, M, Oyarzun, D (2007). “Two objective optimal multivariable ripple- free deadbeat control”, Department of Electronic Engineering, University Tecnica Federico Santa Maria Valparaiso, Chile, Access on 1/4/2009, http://www.hamilton.ie/d.o/docs/saloya07ijc.pdf

Silva, E, Salgado, M (2005). “Performance bounds for feedback control of non- minimum phase MIMO systems with arbitrary delay structure”, IEEE Proceedings-Control Theory and Applications, 152(2): 211-219.

Sirisena, H (1985). “Ripple-Free Deadbeat Control of SISO Discrete Systems”, IEEE Transactions on automatic control, 30(2).

Soliman S, Srinath M (1998). Continuous and Discrete Signals and Systems, New Jersey, Prentice-Hall Inc,.

Wang, S, Davison, E (1973). “A Minimization Algorithm for the Design of Linear Multivariable Systems”, IEEE Transactions on automatic control, 18(3).

Yamada, M, Funahashi, Y (1998). “Multivariable Deadbeat Tracking Controllers”, Proceedings of the 37th IEEE Conference on Decision and Control, Tampa, Florida USA .