Time-Varying Deadbeat Controller Design

Manoranjan Majji‡
Texas A&M University, College Station, Texas 77843-3141
Jer-Nan Juang‡
National Cheng Kung University, Tainan 70101, Taiwan, Republic of China
and
John L. Junkins†
Texas A&M University, College Station, Texas 77843-3141
DOI: 10.2514/1.51563

A time-varying generalization of the classical deadbeat controller is derived directly from input-output data in the present paper. The natural extension of the time invariant deadbeat condition to the time-varying case is presented by using the key developments of the observer/Kalman filter time-varying system identification theory reported recently by the authors. In stark contrast to the time invariant deadbeat controller design, it is shown that the time-varying deadbeat controller parameters do not follow the recursive relationship that was developed in the past. In the special case of periodic systems, where repeated experimental data are available naturally, it seems that such a controller can be designed to set the output of the closed-loop system to rest after a few time steps. Numerical simulation example demonstrates the efficacy of the deadbeat controller design method proposed in this paper. A deadbeat controller, realized from the departure motion dynamics of a spacecraft, is then shown to enhance the performance of an attitude control system by minimizing the deviations from a reference trajectory.

I. Introduction

One of the central problems involved in classical control deals with the stabilization of the system response by means of the output feedback [1]. Significant advancements in modern control theory have produced several state-space methods to design the output feedback control mechanisms. However, to practicing engineers, the input-output map still poses many attractive avenues of investigation in control system design. In modern systems, this problem is often compounded by the parametric uncertainty ever-present in the system model [2]. In addressing such practical challenges involved in model-based estimation and control, system identification has emerged as an important discipline of modern system theory [3]. In problems where control and output regulation are the most important objective, it is sometimes found that the control system design can frequently be accomplished directly. In other words, the analyst may sometimes be able to bypass the four stages of controller design, namely system modeling, identification testing, controller design, and verification. Such methods are frequently known as model predictive control solutions [4, 5]. In several practical applications, such designs lead to cost-effective controller design solutions. In some problems of aerospace engineering such as the aircraft control and stabilization and vibration suppression of spacecraft, the response by the traditional controller design methods may not even be possible due to large model uncertainties, fast changes in disturbance profiles (e.g., during atmospheric reentry), and complex models to simulate and execute [6, 7]. In such situations, online system identification and adaptive control designs appear to be attractive solutions [2]. However, adaptive control strategies are notorious for producing high-order controllers and large transient characteristics that may be undesirable especially in nonlinear systems. Furthermore, the asymptotic stability conditions of the complex controllers realized usually imply a great deal of tuning on the part of the analyst before implementation of the controller.

Deadbeat controller design has been an area of active, fundamental research that provided the fertile ground for proliferation of the more popular model predictive control. In control literature, deadbeat control dates back to the beginnings of discrete time systems. One of the early works that suggests deadbeat design appears in a book by Bode [8], outlining the prominence of the deadbeat control design in system theory. Many other prominent papers have detailed deadbeat control strategies and the paper by Hartley et al. [9] provides an excellent, thorough overview of the historical developments. Most notable contributions to deadbeat control design were made by Minamide [10], Minamide et al. [11], Kucera [12], Kaczorek [13–17] and others (cf. Crossley and Porter [18], Porter and Bradshaw [19], Hostetter [20], Goodwin and Sin [21], and others) in several papers, as they addressed several fundamental issues. These contributions ultimately impacted the evolution of several areas of research such as parameter identification [22], adaptive [21, 22], and optimal control [21].

Deadbeat and predictive control has been applied quite successfully to control several nonlinear dynamical systems. Lu [23] developed a nonlinear model predictive control approach for continuous time systems and successfully demonstrated the methodology by designing a control system for a missile. Crassidis and Markley [24, 25], following this approach, designed predictive attitude control and estimation schemes. They successfully implemented these algorithms in the attitude control system design for the microwave anisotropy probe (MAP). Gawronski [26, 27] successfully applied model predictive control techniques to NASA deep space network antennas, realizing significantly improved tracking performance.

In application of the traditional model predictive control approaches to nonlinear problems, further challenges face the analyst. Typical nonlinear deadbeat/predictive control strategies [9, 16, 17] and model inversion techniques strongly depend on the model structure. In addition to the nonlinear predictive control

—

Presented at the AAS/AIAA Spaceflight Mechanics Meeting, San Diego, CA, 14–17 February 2010; received 13 July 2010; revision received 22 December 2010; accepted for publication 1 March 2011. Copyright © 2011 by Majji, M., Juang, J.N., and Junkins, J.. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923: include the code 0731-5090/12 and $10.00 in correspondence with the CCC.

‡Research Associate, Aerospace Engineering Department, 3141-TAMU; majji@tamu.edu. Member AIAA.
†Professor, Department of Engineering Science. Fellow AIAA.
§Distinguished Professor, Regents Professor, Royce E. Wisenbaker Chair in Engineering, Aerospace Engineering Department, 3141-TAMU. Fellow AIAA.
MAJJI, JUANG, AND JUNKINS

II. Basic Formulation

In this section, we review the developments of the time-varying system identification theory developed recently by the authors (cf. Majji et al. [33, 34]). Fundamental difference equation dictating the time evolution of the state is given by

\[ x_{k+1} = A_k x_k + B_k u_k \] (1)

with the state, output, and input dimensions \( x_k \in \mathbb{R}^n \), \( y_k \in \mathbb{R}^m \), \( u_k \in \mathbb{R}^r \) and the system matrices to be of compatible dimensions \( \forall k \in \mathbb{Z} \), an index set. The solution of the state evolution is given by

\[ x_k = \Phi(k, k_0) x_0 + \sum_{j=k_0}^{k-1} \Phi(k, j+1) B_j u_j \] (3)

where \( k_0 \) can denote any general time step before \( k \) (in particular let us assume it to denote the initial time such that \( k_0 = 0 \)).

Using the definition of the compound state transition matrix, the input–output relationship is given by

\[ y_k = C_k \Phi(k, 0) x_0 + \sum_{j=0}^{k-1} h_{k,j} u_j + D_k u_k \] (5)

where the generalized Markov parameters are given by

\[ h_{k,j} = \begin{cases} C_k \Phi(k,j+1) B_j, & \text{if } j < k - 1 \\ C_k B_{k-1}, & j = k - 1 \\ 0, & \text{if } j < k - 1 \end{cases} \] (7)

The input–output relation can be written in matrix form as

\[ y_k = C_{\mathcal{A}} A_{\mathcal{A},1} \cdots A_{\mathcal{A},0} x_0 + \begin{bmatrix} D_{\mathcal{A}} C_{\mathcal{A}} B_{\mathcal{A},-1} & \cdots & C_{\mathcal{A}} A_{\mathcal{A},-1} & \cdots & C_{\mathcal{A}} A_{\mathcal{A},0} \end{bmatrix} \begin{bmatrix} u_k \\ u_{k-1} \\ \vdots \\ u_0 \end{bmatrix} \] (8)

A. Time-Varying Observer/Kalman Filter System Identification Theory

Let us now quickly summarize the main results and formulation involved in the observer/Kalman filter identification theory. The key step is to introduce an addition and subtraction of an identically zero quantity in the difference equation model Eq. (1) shown as

\[ x_{k+1} = A_k x_k + B_k u_k + G_k y_k - G_k y_k \]

\[ = (A_k + G_k C_k) x_k + (B_k + G_k D_k) u_k - G_k y_k \]

\[ = \tilde{A}_k x_k + \begin{bmatrix} (B_k + G_k D_k) \end{bmatrix} u_k - \begin{bmatrix} G_k \end{bmatrix} y_k \]

\[ = \tilde{A}_k x_k + \tilde{B}_k y_k \] (9)

with no change in the measurement equations at the time step \( t_k \)

\[ y_k = C_k x_k + D_k u_k \] (10)

The outputs at the consecutive time steps, starting from the initial time step \( t_0 \) (denoted by \( k_0 = 0 \)) are therefore written as
\(y_0 = C_0 x_0 + D_0 u_0\)
\(y_1 = C_1 \hat{A}_0 x_0 + D_1 u_1 + C_1 \hat{B}_0 v_0 = C_1 \hat{A}_0 x_0 + D_0 u_0 + \hat{h}_{1,0} v_0\)
\(y_2 = C_2 \hat{A}_1 \hat{A}_0 x_0 + D_2 u_2 + C_2 \hat{B}_1 v_1 + C_2 \hat{A}_1 \hat{B}_0 v_0\)
\[= C_2 \hat{A}_1 \hat{A}_0 x_0 + D_2 u_2 + \hat{h}_{2,1} v_1 + h_{2,0} v_0\]
\vdots
\(y_k = C_k \hat{A}_{k-1} \cdots \hat{A}_0 x_0 + D_k u_k + \hat{h}_{k,k-1} v_{k-1}\)
\[+ \hat{h}_{k,k-2} v_{k-2} + \cdots + \hat{h}_{k,0} v_0\]  
(11)

with the definition of generalized observer Markov parameters
\[
\hat{h}_{k,i} = \begin{cases} 
C_i \hat{A}_{k-1} \cdots \hat{A}_{k-j+1} \hat{B}_j, & \forall k > i + 1 \\
C_i \hat{B}_{k-1}, & \forall k < i + 1 \\
0, & \text{otherwise}
\end{cases}
\]  
(12)

that yields the general relationship
\[
y_k = C_k \hat{A}_{k-1} \cdots \hat{A}_0 x_0 + D_k u_k + \sum_{j=1}^{k} \hat{h}_{k,k-j} v_{k-j}
\]  
(13)

The generalized observer Markov parameters have two block components similar to the linear time invariant case shown in the partitions to be
\[
\hat{h}_{k,k-j} = C_i \hat{A}_{k-1} \cdots \hat{A}_{k-j+1} \hat{B}_j
\]
\[= \left[C_i \hat{A}_{k-1} \cdots \hat{A}_{k-j+1} (B_{k-j} + G_{k-j} C_{k-j}) - C_i \hat{A}_{k-1} \cdots \hat{A}_{k-j+1} G_{k-j} \right]
\]
\[
= \left[h^{(1)}_{k,k-j} - h^{(2)}_{k,k-j} \right]
\]  
(14)

where the partitions \(h^{(1)}_{k,k-j}, h^{(2)}_{k,k-j}\) are used in the calculations of the time-varying system Markov parameters and observer gain sequence in our recent paper (cf. Majii et al. [34]). The closed-loop observer thus constructed is now forced to have an asymptotically stable origin.

### B. Time-Varying Deadbeat Observers

Typically the goal of an observer constructed in a mechanization suggested in the previous section is to enforce certain desirable (stabilizing) characteristics into the closed loop (e.g., deadbeatlike stabilization, etc.) of the observer.

The first step involved in achieving this goal of closed-loop asymptotic stability is to choose the number of time steps \(p_k\) (variable each time in general) sufficiently large so that the output of the plant (at \(t_k+p_k\)) strictly depends on only the \(p_k + 1\) previous augmented control inputs \(\{v_{k+1}, \ldots, v_{k+p_k}\}\) and \(u_{k+p_k}\), and independent of the state at every time step \(t_k\). Therefore, by writing
\[
y_{k+p_k} = C_k \hat{A}_{k-1} \cdots \hat{A}_0 x_0 + D_k u_k + \sum_{j=1}^{p_k+1} \hat{h}_{k+p_k,k-j} v_{k-j}
\]
\[+ \sum_{j=1}^{p_k+1} \hat{h}_{k+p_k,k-j+1} v_{k-j+1} \approx D_k u_k + \sum_{j=1}^{p_k+1} \hat{h}_{k+p_k,k-j+1} v_{k-j+1}
\]  
(15)

Note that in the above equation, we have set \(\hat{A} \cdots \hat{A}_0 = 0\). This leads to the construction of a generalized time-varying autoregressive with exogenous input (GTV-ARX for short) model at every time step. Also, note that the order \(p_k\) of the GTV-ARX model can also change with time (we coin the term “generalized” to describe this variability in the order). This variation and (time-varying observer/Kalman filter identification) complexity provides a large number of observer gains at the disposal of the analyst under the TOKID framework, as discussed in our recent paper [34]. In using this input–output relationship [Eq. (15)] instead of the exact relationship given in Eq. (8), we introduce damping into the closed loop. In this paper, we set this generally variable order to remain fixed and minimum (deadbeat) at each time step. That is to say \(p_k = p = p_{\min}\), where \(p_{\min}\) is the smallest positive integer such that \(mp_{\min} \geq n\), where \(m\) is the number of outputs, and \(n\) is the order of the system. This restriction forces a deadbeat-type observer at every time step and includes elements of linear time invariance (shift invariance) in the (closed loop) behavior of observer Markov parameters, providing ease in calculations by requiring minimum number of repeated experiments. As pointed out in our recent paper [34], the conventional deadbeat conditions cannot be ascertained in the case of the time-varying systems, due to the transition matrix product conditions [e.g., Eq. (15)] that are set to zero. This situation is in contrast with the time invariant systems where higher powers of the system matrix give sufficient conditions to place all the closed-loop system poles at the origin (deadbeat).

This condition can be formally written as

**Definition:**

A linear time-varying discrete time observer is said to be deadbeat if there exists a gain sequence \(G_t\) such that
\[
(A_{k+p+1} + G_{k+p-1} C_{k+p-1}) (A_{k+p-2} + G_{k+p-2} C_{k+p-2}) \cdots (A_k + G_k C_k) = 0_{m \times n}
\]  
(16)

for every \(k\), where \(p\) is the smallest integer such that the condition \(mp \geq n\) is satisfied.

More details and examples illustrating this observer are found in our companion paper. We outline a procedure that identifies the observer Markov parameters leading to the explicit calculation of the gain sequence in this paper.

Since this condition is satisfied at every time step in the GTV-ARX model, we have that
\[
y_k = D_k u_k + \hat{h}_{k,k-1} v_{k-1} + \hat{h}_{k,k-2} v_{k-2} + \cdots + \hat{h}_{k,0} v_0
\]  
(17)

By design it is then clear that the output at every time step \(t_k\) is a pure function of the control inputs \(\{u_1, u_2, \ldots, u_{p_k}\}\) and the outputs over the previous \(p\) time steps \(\{v_{k+1}, v_{k+2}, \ldots, v_{k+p}\}\). It is here that the analogous developments of the time invariant deadbeat controller design can be used to propose a generalized control scheme.

### III. Time-Varying Deadbeat Controller Design

We begin the developments of the key ideas of time-varying deadbeat controller design by outlining the fundamental aspects of model predictive control and its generalization to time-varying systems. To our knowledge, limited literature exists in this area, and our results appear to be unique in the sense that identified GTV-ARX model parameters directly determine the input–output model. Therefore the presentation is made as much self contained as possible in this section.

#### A. Time-Varying Multistep Output Prediction

Starting from Eq. (17), we can rewrite the time-varying finite difference model governing the output at every time step to be given by
\[
y_k = \beta_{k,0} u_k + \sum_{j=1}^{p} (\beta_{k,j} u_{k-j} + \alpha_{k,j} v_{k-j})
\]  
(18)

where the direct transmission matrix is \(\beta_{k,0} = D_k \in \mathbb{R}^{m \times n}\) and the other coefficient matrices are \(\beta_{k,j} = h^{(1)}_{k,k-j} \in \mathbb{R}^{m \times n}\) and \(\alpha_{k,j} = -h^{(2)}_{k,k-j} \in \mathbb{R}^{m \times n}\) (\(j = 1, \ldots, p\)) following the definitions in Eqs. (14) and (17) of the previous discussion on observer/Kalman filter system identification theory (cf. Majii et al. [34] for more details). Writing the similar output relation for the next time step, we have that
\[
y_{k+1} = \beta_{k+1,0} u_{k+1} + \sum_{j=1}^{p} (\beta_{k+1,j} u_{k+1-j} + \alpha_{k+1,j} v_{k+1-j})
\]  
(19)
This equation can be simplified as

\[
\begin{align*}
y_{k+1} &= \beta_{k+1,0}u_{k+1} + \beta_{k+1,1}u_k + \alpha_{k+1,1}y_k + \sum_{j=2}^{p}(\beta_{k+1,j}u_{k+1-j}) \\
&
+ \alpha_{k+1,j}y_{k+1-j}) = \beta_{k+1,0}u_{k+1} + \beta_{k+1,1}u_k + \alpha_{k+1,1}y_k \\
&+ \sum_{j=1}^{p-1}(\beta_{k+1,j+1}u_{k+1-j} + \alpha_{k+1,j+1}y_{k+1-j}) \\
&+ \beta_{k+1,1}u_k + \alpha_{k+1,1}(\beta_{k,0}u_k + \sum_{j=1}^{p}(\beta_{k,j}u_{k+1-j} + \alpha_{k,j}y_{k+1-j})) \\
&+ \sum_{j=1}^{p-1}(\beta_{k+1,j+1}u_{k-j} + \alpha_{k+1,j+1}y_{k-j})
\end{align*}
\]

(20)

For convenience in the further developments of this paper, it is very helpful to consider a redefinition of the GTV-ARX series as

\[
y_{k+1} = \tilde{\beta}_{k+1,0}u_{k+1} + \tilde{\beta}_{k+1,1}u_k + \sum_{j=1}^{p}(\tilde{\beta}_{k+1,j}u_{k+1-j} + \tilde{\alpha}_{k+1,j}y_{k+1-j})
\]

(21)

This explicit redefinition of variables has been used to emphasize the analyst about the dependence of the \((k+1)\)th output on the \(p+2\) inputs \(\{u_{k+1}, u_k, y_{k+1}, \ldots, y_{k-p}\}\) and the \(p\) outputs \(\{y_{k-1}, y_{k-2}, \ldots, y_{k-p}\}\). Equating this redefinition of the output series to Eq. (20) and comparing coefficients of appropriate terms gives us recursive expressions for \(\tilde{\beta}_{k+1,0}, \tilde{\beta}_{k+1,1}, \tilde{\alpha}_{k+1,j}\), as

\[
\begin{align*}
\tilde{\beta}_{k+1,0} &= \beta_{k+1,0} \\
\tilde{\beta}_{k+1,1} &= \beta_{k+1,1} + \alpha_{k+1,1}\tilde{\beta}_{k,0} \\
\tilde{\alpha}_{k+1,j} &= \alpha_{k+1,j+1}\tilde{\beta}_{k,j} \quad \forall j = 1, 2, \ldots, p - 1 \\
\tilde{\beta}_{k+1,p} &= \alpha_{k+1,p}\tilde{\beta}_{k,p} \\
\tilde{\alpha}_{k+1,p} &= \alpha_{k+1,1}\tilde{\alpha}_{k,p}
\end{align*}
\]

(22)

where the initialization

\[
\begin{align*}
\tilde{\beta}_{k,0} &= \beta_{k,0} \\
\tilde{\beta}_{k,j} &= \beta_{k,j} \\
\tilde{\alpha}_{k,j} &= \alpha_{k,j} \\
\forall j &= 1, \ldots, p
\end{align*}
\]

(23)

has been assumed to start the recursive relationships.

Using similar manipulations, it can be shown that the output after \(v(<p)\) time steps can be written as

\[
y_{k+v} = \sum_{j=0}^{v}\tilde{\beta}_{k+v,j}u_{k+v-j} + \sum_{j=1}^{p}(\tilde{\beta}_{k+v,j}u_{k+1-j} + \tilde{\alpha}_{k+v,j}y_{k+1-j})
\]

(24)

with the expressions for the coefficients appropriately given as

\[
\begin{align*}
\tilde{\beta}_{k+v,0} &= \beta_{k+v,0} \\
\tilde{\beta}_{k+v,j} &= \beta_{k+v,j} + \sum_{i=j}^{v}\alpha_{k+v,i}\tilde{\beta}_{k+i-v,j-i} \quad \forall j \leq v < p
\end{align*}
\]

(25)

\[
\begin{align*}
\tilde{\alpha}_{k+v,j} &= \alpha_{k+v,j+1}\tilde{\beta}_{k,j} \quad \forall j = 1, 2, \ldots, p - 1 \\
\tilde{\alpha}_{k+v,p} &= \alpha_{k+v,1}\tilde{\alpha}_{k,p}
\end{align*}
\]

(26)

On the other hand, for all time steps \(v \geq p\) the formulae for recursive evaluation of the coefficients use different formulae and are given by

\[
\begin{align*}
\tilde{\beta}_{k+v,0} &= \beta_{k+v,0} \\
\tilde{\beta}_{k+v,j} &= \beta_{k+v,j} + \sum_{i=1}^{p}\alpha_{k+v,i}\tilde{\beta}_{k+i-v,j-i} \quad \forall j \leq p \\
\tilde{\beta}_{k+v,j} &= \beta_{k+v,j} + \sum_{i=1}^{p}\alpha_{k+v,i}\tilde{\beta}_{k+i-v,j-i} \\
\forall j &= 1, \ldots, p
\end{align*}
\]

(27)

\[
\begin{align*}
\tilde{\alpha}_{k+v,j} &= \alpha_{k+v,j+1}\tilde{\beta}_{k,j} \quad \forall j = 1, 2, \ldots, p - 1 \\
\tilde{\alpha}_{k+v,p} &= \alpha_{k+v,1}\tilde{\alpha}_{k,p}
\end{align*}
\]

(28)

It is of consequence to note that these expressions generalize the time invariant deadbeat controller design of previous researchers (cf. [30–32]). However, in stark contrast with the time invariant problem, it is important to note that the simplification of the above formulae to the very compact two-term recursive relations is generally not possible in time-varying systems. This is because of the time-varying Markov parameters involved in the calculations.

Such output expressions can be stacked into a giant matrix equation that can be written as

\[
Y_{k+\nu} = \Gamma_{k+\nu}U_{k+\nu} + B_{k+\nu}U_{k-1-k-\nu} + A_{k+\nu}Y_{k-1-k-\nu}
\]

(29)

where

\[
\Gamma_{k+\nu} := \begin{bmatrix}
\tilde{\beta}_{k,0} & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\
\tilde{\beta}_{k+1,0} & \tilde{\beta}_{k+1,1} & 0 & \cdots & 0 & 0 & \cdots & 0 \\
\tilde{\beta}_{k+2,1} & \tilde{\beta}_{k+2,2} & \tilde{\beta}_{k+2,0} & \cdots & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\
\tilde{\beta}_{k+q,q} & \tilde{\beta}_{k+q,q+1} & \tilde{\beta}_{k+q,q+2} & \cdots & \tilde{\beta}_{k+q,0} & 0 & \cdots & 0 \\
\tilde{\beta}_{k+q+1,q+1} & \tilde{\beta}_{k+q+1,q+2} & \tilde{\beta}_{k+q+1,q+3} & \cdots & \tilde{\beta}_{k+q+1,1,0} & \tilde{\beta}_{k+q+1,1} & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \ddots & \cdots & \vdots \\
\tilde{\beta}_{k+s,s} & \tilde{\beta}_{k+s,s-1} & \tilde{\beta}_{k+s,s-2} & \cdots & \tilde{\beta}_{k+s,r-q} & \tilde{\beta}_{k+s,r-q-1} & \cdots & \tilde{\beta}_{k+s,0}
\end{bmatrix}
\]

(30)
Having outlined the key results of time-varying prediction models, we now proceed to detail the developments of the deadbeat controller design problem.

B. Time-Varying Deadbeat Controller Design

Given the detailed developments of the output sequences developed in the previous section, we are motivated to ask the following question: What should be the control future control inputs \( u_k, u_{k+1}, ..., u_{k+q} \) such that the future output sequence \( y_{k+q+1}, y_{k+q+2}, ..., y_N \) is set to zero? This is the objective of the deadbeat controller. It should be noted that the control action starts at time step \( k \) and ends at time step \( k + q \). This is to say that the control inputs for all subsequent time steps are set to zero

\[
u_{k+q+1} = u_{k+q+2} = \ldots = u_{\infty} = 0
\]

To achieve this controller design, let us consider the outputs from time step \( k + g + 1 \) through \( k + g + s \) for a sufficiently large number of steps \( s - g > p \). Collecting these outputs of interest, we have that

\[
Y_{k+q+1:k+s} = \Gamma_{k+q+1:k+s} U_{k+q} + B_{k+q+1:k+s} U_{k-1:k-p} + a_{k+q+1:k+s} Y_{k-1:k-p}
\]

where

\[
\begin{align*}
Y_{k+q+1:k+s} &= \begin{bmatrix} y_{k+q+1} \\ y_{k+q+2} \\ \vdots \\ y_{k+s} \end{bmatrix}, & Y_{k-1:k-p} &= \begin{bmatrix} y_{k-1} \\ y_{k-2} \\ \vdots \\ y_{k-p} \end{bmatrix} \\
U_{k+q} &= \begin{bmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+q} \end{bmatrix}, & U_{k-1:k-p} &= \begin{bmatrix} u_{k-1} \\ u_{k-2} \\ \vdots \\ u_{k-p} \end{bmatrix}
\end{align*}
\]

and

\[
\Gamma_{k+q+1:k+s} = \begin{bmatrix} \hat{\beta}_{k+q+1,1} & \hat{\beta}_{k+q+1,2} & \ldots & \hat{\beta}_{k+q+1,p} \\ \hat{\beta}_{k+q+2,1} & \hat{\beta}_{k+q+2,2} & \ldots & \hat{\beta}_{k+q+2,p} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\beta}_{k+s,1} & \hat{\beta}_{k+s,2} & \ldots & \hat{\beta}_{k+s,p} \end{bmatrix}
\]

By deadbeat controller design, we wish to impose the necessary condition that

\[
Y_{k+q+1:k+s} = 0
\]

This condition yields our control sequence to be calculated by solving the matrix equation

\[
U_{k+q} = -[\Gamma_{k+q+1:k+s}^\dagger] (B_{k+q+1:k+s} U_{k-1:k-p} + a_{k+q+1:k+s} Y_{k-1:k-p})
\]

where the dagger (\(^\dagger\)) indicates the Moore–Penrose pseudoinverse of a matrix, indicating a minimum norm solution if the number of time steps is large.

Notice that the solution proposed here automatically produces optimal solutions that minimize the norm of the control input if the entire time history is chosen to achieve the deadbeat condition. Therefore, from a practical standpoint, the analyst has a large degree of design freedom to investigate the number of time steps that are required to bring the outputs to rest. Consequently the control law can be modified to suit actuator limitations. We summarize the computational steps involved in the algorithm in the following section.

C. Computational Steps to Time-Varying Deadbeat Controller

The steps to be followed in the time-varying deadbeat controller developed in this paper are as follows:

1) Compute the time-varying system and observer Markov parameters using the observer/Kalman filter time-varying system identification theory developed in [34].
2) Select time steps \( k, q, s \) of interest.
3) Determine \( \hat{\beta}_{k,j}, \hat{\beta}_{k,j}^\dagger \) using the definitions in Eq. (18).
4) Compute \( \hat{\beta}_{k,j}, \hat{\beta}_{k,j}^\dagger, \hat{\beta}_{k,j} \) using Eqs. (22–28).
5) Assemble and form composite matrices \( \Gamma_{k+q+1:k+s}, B_{k+q+1:k+s}, A_{k+q+1:k+s}, U_{k-1:k-p} \) and \( Y_{k-1:k-p} \) using Eqs. (36–39).
6) Determine the time-varying deadbeat control inputs \( U_{k+q} \) as the solution from Eq. (41).

We now proceed to provide a numerical simulation to demonstrate the effectiveness of this controller.

IV. Numerical Example

To demonstrate the effectiveness of the controller design strategy presented in the paper, we consider a time-varying linear system with characteristics that are oscillatory in nature. This simple, idealized example is provided to help the reader follow the steps of realizing a time-varying deadbeat controller.

Furthermore, it is of consequence to note that the current example is not stable even in the bounded-input–bounded-output sense (BIBO) (cf. Vidyasagar [35]). Although we do not use the small gain arguments in the present paper, the divergent open-loop response to two-step control input conclusively shows this nature of the system. The plant model was intentionally chosen to display such characteristics to demonstrate the efficacy of the control solution being designed in the present paper.
Consider the time-varying system described by the following matrices

\[ A_k = \exp[A_k \Delta t], \quad B_k = \begin{bmatrix} 1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix}, \quad C_k = \begin{bmatrix} 1 & 0 & 1 & 0.2 \\ 1 & -1 & 0 & -0.5 \end{bmatrix}, \quad D_k = 0.1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \]

where the matrix \( A_c \) is given by

\[ A_c = \begin{bmatrix} 0_{2 \times 2} & I_{2 \times 2} \\ -K_{t} & 0_{2 \times 2} \end{bmatrix} \]

with \( K_{t} = \begin{bmatrix} 4 + 3\tau_k & -1 \\ -1 & 7 + 3\tau_k \end{bmatrix} \) and \( \tau_k, \tau_k' \) are defined as \( \tau_k = \sin(10\tau_k), \quad \tau_k' := \cos(10\tau_k) \). The TOKID algorithm, as described in the main body of the paper, was applied to this example to calculate the system Markov parameters and the observer gain Markov parameters from the simulated repeated experimental data.

Since the number of outputs and inputs for this problem is known to be \( m = r = 2 \), an examination of the necessary minimum order for the time-varying deadbeat condition is found to be \( p = 2 \).

To simulate the presence of nonzero initial conditions and past control inputs, we chose arbitrary values for the first two output and input values. In the present example, two output vectors

\[ y_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \times 10^{-2}; \quad y_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \times 10^{-2} \]

were chosen, and two input vectors

\[ u_1 = \begin{bmatrix} 1 \\ -2 \end{bmatrix} \times 10^{-3}; \quad u_2 = \begin{bmatrix} 3 \\ 5 \end{bmatrix} \times 10^{-3} \]

were applied. To demonstrate the effects of longer durations of control input application, various step sizes \( s = 2, 3, \ldots, 8 \) were considered, and the response and the associated control inputs realized by the deadbeat controller are plotted.

Figure 1 shows the system response for each of the controller designs (number of steps the controller is in the “on” position) and compares the output with an open-loop simulation where the...
controller has been turned “off” after the application of two inputs indicated in Eq. (45).

In Fig. 1, eight different step sizes for controller design were considered. We first note the diverging trend in the open-loop response where the controller is switched off after the first two time steps. This rather unusual behavior is characteristic trademark of the time-varying systems where classical notions of stability for linear time-invariant systems do not hold anymore. Furthermore, although stabilized, the time-varying system of the present example, even though excited by a relatively small input elicits large, diverging response characteristics, particularly challenging to control. The same profiles are plotted in Fig. 2 in logarithmic scale.

The various control actions are illustrated clearly in this figure. It is indeed obvious that the control inputs maintain divergence until their moment arises and set the output to zero at the correct time instant. The small driftlike variation in the response is due to numerical artifacts in the computations of the control gains. Since the outputs and inputs are also zero in subsequent time steps, the system is stabilized uniformly in the absence of disturbances. If one anticipates that the noise inherently present in most physical systems should excite such dynamics again, one would apply this technique repeatedly to maintain the state at origin uniformly as often as desired. Therefore, the technique produces a rather useful set of controller options.

Control profiles used in each of the stabilization strategies are plotted in Fig. 3.

Quite clearly, it is also pointed out that the magnitudes associated with each control law are naturally in the decreasing order of magnitude. This is a physically appealing aspect of the current controller design, where longer available “time-to-go” naturally yields a lower power consumption for the controller. This control magnitude plots along with the labeling of individual control laws is provided in Fig. 4.

V. Application to Spacecraft Attitude Control

Spacecraft attitude control is one of the most important and difficult tasks for operational engineers. Several nonlinear and optimal control strategies exist for stabilization and tracking of spacecraft attitude (cf. Vadali and Junkins [36], Tsiotras [37], Wie and Barba [38], and Crassidis et al. [24] and the references therein). Usually both open-loop and closed-loop design strategies are adopted and a variety of actuators (reaction wheels, control moment gyros, thrusters, etc.) enable the required maneuver execution and
precision pointing for communication and other mission-specific science requirements. Following the developments of Crassidis et al. [24], we will now set up a nominal large angle reorientation maneuver and develop a perturbative feedback control scheme in discrete time using the deadbeat controller design concepts discussed in the paper.

A. Spacecraft Model and Reference Controller Design

Consider a spin stabilized spacecraft similar to the microwave anisotropy probe (MAP) outlined by Crassidis et al. [24] (see reference for more information on the details of the spacecraft, including schematics and reference frames). Reference trajectory of the spacecraft attitude is defined by the following 3-1-3 Euler angle rotation relative to the rotating, sun-centered frame of reference given by the desired states as

$$\dot{\phi} = 1 \text{ rev/hr} \quad \dot{\theta} = 22.5 \text{ deg} \quad \dot{\psi} = 0.464 \text{ rpm} \quad (46)$$

and the spacecraft is assumed to be equipped with a star tracker and a rate integrating gyro for measurement of attitude and body angular velocity, respectively, enabling state feedback.

Spacecraft attitude kinematics and dynamics are given by the differential equations

$$\dot{q} = \frac{1}{2} \Omega(q) q = \frac{1}{2} \Xi(q) \omega \quad J \dot{\omega} = -[\omega \times] J \omega + \tau \quad (47)$$

where \( q \in \mathbb{R}^4 \) is the attitude quaternion that can be decomposed as

$$q = \begin{bmatrix} q_{13} \\ q_4 \end{bmatrix} \quad (48)$$

with \( q_{13} \in \mathbb{R}^3 \) and scalar \( q_4 \in \mathbb{R} \). \( J \) is the inertia matrix of the rigid body, \( \omega \in \mathbb{R}^3 \) denotes the angular velocity of the body, and the vector \( \tau \in \mathbb{R}^3 \) denotes the external torque (control) acting on the spacecraft. The matrices \( \Omega(\omega) \) and \( \Xi(q) \) are defined as

$$\Omega(\omega) = \begin{bmatrix} -[\omega \times] & \omega \\ \vdots & \vdots & \vdots \\ -\omega^T & 0 \end{bmatrix} \quad (49)$$

$$\Xi(q) = \begin{bmatrix} q_{13} J_{3,3} + [q_{13} \times] \\ \vdots \\ -q_{13}^T \end{bmatrix} \quad (50)$$

together with the definition of the cross product matrix

$$[a \times] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \quad (51)$$

$$\forall a = [a_1 \ a_2 \ a_3]^T$$

Following the classical solution of Wie and Barba [38], we stabilize the plant using a quaternion feedback control solution, given by

$$\tau = k_p \Xi^T(\bar{q}) q - K_q(\omega - \bar{\omega}) \quad (52)$$

where \( \bar{q} \), \( \bar{\omega} \) are the reference trajectories to be tracked by the controller. It has been shown by Wie and Barba [38] that in this controller, for appropriate choices of the control gains, the tracking error dynamics is asymptotically stable. However, upon changes in the initial conditions of the system, transient response characteristics may sometimes be unacceptable and the analyst may want to incorporate additional force profiles to achieve improved transient performance. In this paper, we set out to calculate the additional control torque inputs using the time-varying deadbeat control technique presented in the previous sections of this article.

We start by considering a reference trajectory given by the Euler angle rates specified by Eq. (46). These trajectories in terms of the quaternion are given by

$$\bar{q}_1 = \sin \left( \frac{\theta}{2} \right) \cos \left( \phi - \frac{\psi}{2} \right) \quad \bar{q}_2 = \sin \left( \frac{\theta}{2} \right) \sin \left( \frac{\phi - \psi}{2} \right) \quad (53)$$

$$\bar{q}_3 = \cos \left( \frac{\theta}{2} \right) \sin \left( \frac{\phi + \psi}{2} \right) \quad \bar{q}_4 = \cos \left( \frac{\theta}{2} \right) \cos \left( \frac{\phi + \psi}{2} \right)$$

while the body angular velocity reference trajectories are obtained from the Euler angle rates by the transformation

$$\bar{\omega} = \begin{bmatrix} \sin \theta \sin \psi & \cos \psi & 0 \\ \sin \theta \sin \psi & -\sin \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (54)$$

Fig. 5 Nominal and perturbed stabilizing trajectories of the spacecraft attitude.
Considering a spacecraft with principal inertias, \( J_{11} = 15.2, J_{22} = 50.2, J_{33} = 150 \) and initial conditions
\[
q(t_0) = [0.685 \quad 0.695 \quad 0.153 \quad 0.153]^T \\
\omega(t_0) = [0.53 \quad 0.53 \quad 0.53]^T \text{ deg/sec}
\]

The stabilizing control laws obtained by using the gains \( k_p = 1, K_d = 2 \) are shown in Figs. 5–7 (solid blue lines). Also plotted in the same figures are the trajectories obtained by perturbing the angular velocity initial conditions by
\[
\delta \omega(t_0) = \begin{bmatrix} 2 & -1 & 3 \end{bmatrix} \times 10^{-2} \text{ rad/sec}
\]

In a practical situation, this perturbation simulates gyro drift, where a direct use of angular rate information for tracking sets the spacecraft in a different state than the desired one.

Although the difference in attitude trajectories (as shown in Fig. 5, plotted in terms of the quaternion) may appear small, it is not the case. This is clarified by plotting the errors between the perturbed attitude motions and the reference (nominal: no perturbations in the initial angular velocity value) in the traditional Euler 3-2-1 angle sequence in Fig. 7. Quite clearly, the perturbed trajectory experiences large departure from the original (approved by the analyst by design and choice of the feedback gains) path. Let us now assume that the analyst wishes to minimize this departure and look at the methods to calculate appropriate corrections. Time-varying deadbeat control is now shown to provide answers in such a situation.

### B. Application of Deadbeat Control

For the application of “small” perturbative torque signals, the feedback controller with an additional perturbation force term is applied to the system. This can be written as
\[
\tau^{DB} = k_p \Sigma^T (\tilde{q}) q - K_d (\omega - \bar{\omega}) + \delta u(t)
\]

We now apply “zero-order-hold” type control torques of small magnitude such that these lead to small deviations in the outputs of the attitude and body angular rate information. While the choice of the sample size, etc., remain the tuning parameters of the algorithm, this procedure and associated multiple experiments, were shown to produce discrete time-varying models of reliable accuracy by use of
new methods of system identification in our recent research (cf. dissertation by Majji [39]).

Since the torques applied for the identification of the Markov parameters of the GTV-ARX model are “small,” one expects small attitude motions about the reference trajectory. Accordingly, the vector part of the quaternion and the angular velocity vector are considered as the outputs of interest for the identification of the GTV-ARX model coefficients governing the departure motion dynamics. This is to say that

$$\delta y(t) = \begin{bmatrix} \delta q_{13} \\ \delta \omega \end{bmatrix}$$  \hspace{1cm} (58)

where $\delta \omega = \omega_p - \omega_r$, the algebraic difference between the reference angular rate $\omega_r$ developed in the previous subsection and the perturbed angular rate $\omega_p$ with appropriately modified initial conditions. The vector part of the error quaternion $\delta q_{13}$ is, however, the vector part of the quaternion error $\delta q = q_p \otimes q_r^{-1}$. The quaternion multiplication and the inverse attitude quaternion used for its computation are developed in greater detail in Crassidis et al. [24]. It is well known that the vector part of the quaternion is sufficient for producing an accurate linearized model with a large domain of convergence. We make use of this fact and achieve a reduction in the output state dimension (only six states).

Having established the inputs and outputs for generation of the perturbation GTV-ARX model, we perform multiple experiments by limiting the perturbation control magnitude to within $||\delta u|| \leq 0.5 \text{ Nm}$. Using the steps outlined in the Sec. III.C, we then compute the control inputs required and apply to the perturbed plant.

Figure 8 shows the history of the attitude errors incurred in case of pure perturbations when compared with the application of the deadbeat control (analogous to Fig. 7). Therefore, it is clearly demonstrated that, barring the initial transience, application of the deadbeat control steps during the initial few time steps makes the response converge to the reference trajectory very quickly. Figure 9 shows that similar to the attitude tracking, we also achieve angular velocity tracking by application of the realized deadbeat control torques.

An interesting question that immediately arises is regarding the control input magnitudes and feasibility of implementation of such deadbeat control superimpositions. To answer this question, we plot the control torques required for the three different scenarios discussed in the attitude control problem in Fig. 10. The first control torque profile (shown in the solid line) is associated with the control

![Fig. 8 Attitude error of perturbed plant and deadbeat controlled plant.](image)

![Fig. 9 Angular velocity deviation profile comparison (with and without deadbeat control).](image)
torque requirements of the reference feedback solution. We also plot the control torque profiles for the feedback control solutions for the perturbed plant model (shown in the dash–dot line) and the control torque profile that includes the deadbeat control solution superimposed on the feedback (shown in the dotted line). Clearly, for this example, the control pulses required or the deadbeat controller are not necessarily larger in magnitude than that of the pure feedback solution. It is also interesting to note that after the first few time steps, the control torque profile follows that of the reference design since the state trajectories follow reference trajectories closely.

Thus, we demonstrate the efficacy of the deadbeat control design strategy developed in the paper by application to the nonlinear problem of controlling the attitude dynamics of a spacecraft. While this is necessarily only a starting point for applications of time-varying and nonlinear system identification, this example indicates a high degree of optimism about the utility of the algorithms under investigation.

VI. Conclusions
A novel approach to control of time-varying systems is presented in this paper. For the first time, a deadbeat controller is designed for a time-varying system indirectly from input–output experimental data. Recent developments of the observer/Kalman filter system identification theory are used to provide a controller design that stabilizes the time-varying plant in a few time steps, depending on the order of the plant model and the underlying state-space dimension of the problem involved. Generalized time-varying autoregressive with exogenous input model developed in the context of time-varying observer/Kalman filter system identification by the authors is used in the process for extraction of the coefficients. Subsequent conditions of the output stabilization are shown to naturally lead to a controller design that renders the output of the plant vanish in a few time steps. Example simulations detailing the procedure on a time-varying problem show a high degree of optimism on the application of the proposed methodology for guidance and control problems that are typically time-varying in nature.

Acknowledgments
The authors wish to acknowledge Texas A&M University and National Cheng Kung University for enabling this collaborative research. John Valasek, Johnny Hurtado, and Todd Griffith are gratefully acknowledged for discussions regarding the application of components of this research to several identification and control problems.

References
No. 4, pp. 393–405. doi:10.1080/00207728108963754


