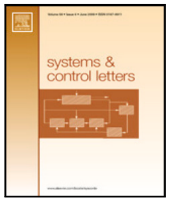
Systems & Control Letters 185 (2024) 105744

Contents lists available at ScienceDirect

Systems & Control Letters

journal homepage: www.elsevier.com/locate/sysconle



Performance-variable decomposition in retrospective cost adaptive control of linear time-varying systems

Sneha Sanjeevini *, Dennis S. Bernstein

Department of Aerospace Engineering, University of Michigan, Ann Arbor, USA

ARTICLE INFO

Keywords:
Discrete-time
Linear time-varying systems
Adaptive control
Input–output models

ABSTRACT

This paper develops a discrete-time, linear time-varying (DTLTV) framework for analyzing the retrospective performance variable used in retrospective cost adaptive control (RCAC). This is done by first developing expressions for transforming between DTLTV state–space models and DTLTV input–output models. These expressions are then used to derive an additive decomposition of the retrospective performance variable in terms of a predicted-performance term and a model-matching term that measures the closeness between the closed-loop dynamics and the target model. Numerical examples are given to illustrate the modeling information required by RCAC and provide insight into how RCAC achieves closed-loop performance and model matching.

1. Introduction

Unlike optimal and robust feedback controllers, the gains of an adaptive controller change over time based on the response of the actual plant to its initial conditions, exogenous inputs, and control inputs [1–3]. Hence, for an adaptive control algorithm, a linear, time-varying (LTV) framework is needed for analyzing closed-loop performance.

The present paper focuses on the analysis of retrospective cost adaptive control (RCAC) [4], which is a discrete-time, direct adaptive control algorithm for stabilization, command following, and disturbance rejection. RCAC is based on the concept of retrospectively optimized control, where past controller coefficients used to generate past control inputs are reoptimized in the sense that, if the reoptimized coefficients had been used over a previous window of operation, then the performance would have been better. RCAC has been used in various applications including flight control [5], noise control [6], and quadrotor control [7].

The modeling information required by RCAC is embedded in a filter that serves as the target model for a specific closed-loop transfer function. As shown in [4], the essential modeling information for single-input, single-output (SISO) systems includes the sign of the leading numerator coefficient, the relative degree, and all nonminimum-phase (NMP) zeros. Numerical examples in [4] and the current paper show that RCAC cancels unmodeled NMP zeros leading to unstable pole-zero cancellations in the closed-loop transfer function.

More recently, an indirect adaptive control extension of RCAC that incorporates online system identification to update the target model was developed in [8]. It is shown in [8] that the retrospective performance variable, which is the variable used to define the cost function, can be decomposed as the sum of a predicted-performance term and a model-matching term that measures the closeness between the closed-loop dynamics and the target model. The development in [8], which is based on input–output models, incorrectly accounts for the LTV controller dynamics that arise from the controller update, and considers only linear, time-invariant (LTI) plants and target models. The present paper revisits the retrospective-performance-variable decomposition by focusing on discrete-time LTV (DTLTV) models in state–space and input–output representations as well as transformations between them.

The underlying motivation for the present paper arises from the fact that RCAC is based on DTLTV input–output models; analysis of RCAC thus depends on interconnections of these models. Unlike LTI input–output models, simple examples show that naive multiplication of LTV input–output models does not yield a correct LTV input–output model of the cascaded dynamics. To overcome this impediment, transformations between LTV input–output models and LTV state–space models are required. The LTV input–output models are transformed to LTV state–space models, which are interconnected, and the resulting expressions are transformed back to LTV input–output models. By taking this approach, the present paper correctly accounts for the interconnection of DTLTV input–output models.

LTV state–space models are considered for continuous-time systems in [9–14], and for discrete-time systems in [9,15–21]. Analogously, LTV input–output models are considered for continuous-time systems

* Corresponding author.
E-mail address: snehasnj@umich.edu (S. Sanjeevini).

https://doi.org/10.1016/j.sysconle.2024.105744
Received 4 May 2023; Received in revised form 15 December 2023; Accepted 31 January 2024

in [9–11], and for discrete-time systems in [9,15–19]. The problem of transforming between DTLTV state–space and DTLTV input–output models is considered in [18]. The present paper derives simple and directly implementable algebraic expressions for transforming DTLTV state–space models to DTLTV input–output models and vice versa.

The main contribution of the present paper is a derivation of the retrospective-performance-variable decomposition that accounts for the LTV dynamics of the plant, controller, and target model as well as the effect of the initial conditions of the plant. Numerical examples are given to illustrate the use of retrospective-performance-variable decomposition in understanding the modeling information required by RCAC and in providing insight into how RCAC achieves closed-loop performance and model matching. A related development is given in [22] within the context of adaptive input estimation. The development in the present paper goes beyond [22] by providing complete proofs for transforming between DTLTV state–space models and DTLTV input–output models.

Section 2 presents definitions used in the paper. Section 3 summarizes the RCAC algorithm. The retrospective-performance-variable decomposition is derived in Section 4 and analyzed in Section 5. Section 6 presents illustrative numerical examples, and Section 7 concludes the paper. Expressions for the transformations between DTLTV state–space models and DTLTV input–output models are derived in Appendix A.

Notation. $\mathbb{R}^{p\times m}$ denotes the set of $p\times m$ matrices with real coefficients, z denotes a complex number, $\mathbb{R}(\mathrm{z})^{p\times m}$ denotes the set of all transfer functions, that is, the set of $p\times m$ matrices each of whose entries is a rational function of z with real coefficients, and $\mathbb{R}(\mathrm{z})^{p\times m}_{\mathrm{prop}}$ denotes the proper transfer functions in $\mathbb{R}(\mathrm{z})^{p\times m}$. $\mathbf{q}^{-1}$ denotes the time-domain, backward shift operator. $(x_k)_{k=0}^{\infty}$ denotes the sequence $(x_0, x_1, \ldots)$.

2. Discrete-time, linear time-varying models

This section presents definitions relating to discrete-time, linear time-varying (DTLTV) models that will be used in later sections.

Definition 2.1. Let $y_{\mathrm{io},-n}, y_{\mathrm{io},-n+1}, \ldots, y_{\mathrm{io},-1} \in \mathbb{R}^p$, and consider the DTLTV input–output model given by, for all $k \geq 0$,

$$y_{\mathrm{io},k} + D_{1,k}y_{\mathrm{io},k-1} + \cdots + D_{n,k}y_{\mathrm{io},k-n} = N_{0,k}u_k + \cdots + N_{n,k}u_{k-n}, \tag{1}$$

where $u_k \in \mathbb{R}^m$ is the input, $y_{\mathrm{io},k} \in \mathbb{R}^p$ is the output, $D_{1,k}, \ldots, D_{n,k} \in \mathbb{R}^{p\times p}$, $N_{0,k}, \ldots, N_{n,k} \in \mathbb{R}^{p\times m}$, and n is the order of (1). Define

$$D_k(\mathbf{q}^{-1}) \triangleq I_p + D_{1,k}\mathbf{q}^{-1} + \cdots + D_{n,k}\mathbf{q}^{-n}, \tag{2}$$

$$N_k(\mathbf{q}^{-1}) \triangleq N_{0,k} + N_{1,k}\mathbf{q}^{-1} + \cdots + N_{n,k}\mathbf{q}^{-n}. \tag{3}$$

Then, $G_k \triangleq D_k^{-1}N_k$ is the *time-domain transfer function of* (1) *at step* k. In terms of G_k, (1) is written as

$$y_{\mathrm{io},k} = G_k(\mathbf{q}^{-1})u_k, \tag{4}$$

and, in terms of N_k and D_k, (1) is written as

$$D_k(\mathbf{q}^{-1})y_{\mathrm{io},k} = N_k(\mathbf{q}^{-1})u_k. \tag{5}$$

Note that the input–output model (1) and its backward-shift representation (5) are time-domain models, which include the effect of initial conditions, as discussed in [23].

The following definition is based on the definition of the observability matrix for DTLTV systems given in [24].

Definition 2.2. Consider the DTLTV state–space model

$$x_{k+1} = A_kx_k + B_ku_k, \tag{6}$$

$$y_{\mathrm{ss},k} = C_kx_k + E_ku_k, \tag{7}$$

where, for all $k \geq 0$, $x_k \in \mathbb{R}^n$ is the state, $u_k \in \mathbb{R}^m$ is the input, and $y_{\mathrm{ss},k} \in \mathbb{R}^p$ is the output. Define the *observability matrix at step* k as

$$\mathcal{O}_k \triangleq \begin{bmatrix} C_k \\ C_{k+1}\Psi_{k,k} \\ C_{k+2}\Psi_{k+1,k} \\ \vdots \\ C_{k+n-1}\Psi_{k+n-2,k} \end{bmatrix}, \tag{8}$$

and the *controllability matrix at step* k as

$$C_k \triangleq \begin{bmatrix} B_{k-1} & \Psi_{k-1,k-1}B_{k-2} & \Psi_{k-1,k-2}B_{k-3} & \cdots & \Psi_{k-1,k-n+1}B_{k-n} \end{bmatrix}, \tag{9}$$

where

$$\Psi_{i,j} \triangleq \begin{cases} A_iA_{i-1}\cdots A_j, & i > j, \\ A_i, & i = j, \\ 0_{n\times n}, & i < j. \end{cases} \tag{10}$$

Define the sequences $\mathcal{A} \triangleq (A_1, A_2, \ldots)$, $\mathcal{B} \triangleq (B_1, B_2, \ldots)$, and $\mathcal{C} \triangleq (C_1, C_2, \ldots)$. If, for all $k \geq 0$, $\operatorname{rank}\mathcal{O}_k = n$, then $(\mathcal{A}, \mathcal{C})$ is *completely observable*. If, for all $k \geq n$, $\operatorname{rank}C_k = n$, then $(\mathcal{A}, \mathcal{B})$ is *completely controllable*. Finally, if $(\mathcal{A}, \mathcal{B})$ is completely controllable and $(\mathcal{A}, \mathcal{C})$ is completely observable, then $(\mathcal{A}, \mathcal{B}, \mathcal{C})$ is *minimal*.

Definition 2.3. (1) is a *DTLTV input–output model corresponding to the DTLTV state–space model* (6), (7) if, for all $x_0 \in \mathbb{R}^n$ and all $(u_k)_{k=0}^{\infty} \subset \mathbb{R}^m$, there exist $y_{\mathrm{io},0}, y_{\mathrm{io},1}, \ldots, y_{\mathrm{io},n-1} \in \mathbb{R}^p$ such that $(y_{\mathrm{io},k})_{k=n}^{\infty} = (y_{\mathrm{ss},k})_{k=n}^{\infty}$.

Note that, in Definition 2.3, the DTLTV input–output model (1) is valid only for $k \geq n$.

Definition 2.4. (6), (7) is a *DTLTV state–space realization of the DTLTV input–output model* (1) if, for all $y_{\mathrm{io},-n}, y_{\mathrm{io},-n+1}, \ldots, y_{\mathrm{io},-1} \in \mathbb{R}^p$ and all $(u_k)_{k=0}^{\infty} \subset \mathbb{R}^m$, there exists $x_0 \in \mathbb{R}^{pn}$ such that $(y_{\mathrm{io},k})_{k=0}^{\infty} = (y_{\mathrm{ss},k})_{k=0}^{\infty}$.

Note that, in Definition 2.4, the dimension of the state x_k of the DTLTV state–space model (6), (7) is pn.

Definition 2.5. The *time-domain transfer function corresponding to the DTLTV state–space model* (6), (7) *at step* k is the time-domain transfer function of the DTLTV input–output model (1) corresponding to (6), (7) at step k. Furthermore, the *Markov parameters of the time-domain transfer function corresponding to* (6), (7) *at step* k are defined as

$$H_{i,k} \triangleq \begin{cases} E_k, & i = 0, \\ C_kB_{k-1}, & k \geq i = 1, \\ C_k\Psi_{k-1,k-i+1}B_{k-i}, & k \geq i \geq 2, \\ 0_{p\times m}, & i > k. \end{cases} \tag{11}$$

The following definition is given in [8]. This definition concerns DTLTV input–output models whose input u_k is a function of a parameter vector θ and such that, at step k, all of the inputs $u_k, \ldots, u_{k-n}$ are evaluated at the same parameter vector θ_k; in other words, the value of θ_k is fixed at the current time step. Because of this dependence, the output y_k depends on θ_k.

Definition 2.6. Let $D_{1,k}, \ldots, D_{n,k} \in \mathbb{R}^{p\times p}$, let $N_{0,k}, \ldots, N_{n,k} \in \mathbb{R}^{p\times m}$, let $Y_{-n}, \ldots, Y_{-1} \in \mathbb{R}^p$, let $(\theta_k)_{k=-n}^{\infty} \in \mathbb{R}^r$, and, for all $k \geq -n$, let $u_k : \mathbb{R}^r \to \mathbb{R}^m$. Then, the *fixed-input-argument (FIA) sequence* $(y_k(\theta_k))_{k=0}^{\infty}$ is given by the *FIA filter*

$$y_k(\theta_k) + D_{1,k}y_{k-1}(\theta_{k-1}) + \cdots + D_{n,k}y_{k-n}(\theta_{k-n}) = N_{0,k}u_k(\theta_k) + \cdots + N_{n,k}u_{k-n}(\theta_k), \tag{12}$$

where, for all $k \in [-n, -1]$, $y_k(\theta_k) \triangleq Y_k$.

At each step k, the arguments of $u_{k-n}, \ldots, u_k$ in (12) are fixed at the current value θ_k. In contrast, the left hand side defines the current

output $y_k(\theta_k)$, which depends on the past output values $y_{k-n}(\theta_{k-n}), \ldots, y_{k-1}(\theta_{k-1})$. In terms of $\mathbf{q}^{-1}$, (12) can be written as either

$$D_k(\mathbf{q}^{-1})y_k(\theta_k) = N_k(\mathbf{q}^{-1})u_k(\theta_{\overline{k}}) \tag{13}$$

or

$$y_k(\theta_k) = G_k(\mathbf{q}^{-1})u_k(\theta_{\overline{k}}), \tag{14}$$

where $G_k \triangleq D_k^{-1}N_k$ and where the notation $\overline{k}$ denotes the fact that, at each step k, all of the arguments of $u_k, \ldots, u_{k-n}$ in (12) are set to θ_k.

3. Retrospective cost adaptive control for linear time-varying systems

This section explains the retrospective cost adaptive control algorithm which can be used for stabilization, command following, and disturbance rejection.

Consider the DTLTV state–space model

$$x_{k+1} = A_k x_k + B_k u_k + B_{w,k} w_k, \tag{15}$$

$$y_k = C_k x_k + v_k, \tag{16}$$

where $k \geq 0$, $x_k \in \mathbb{R}^{l_x}$ is the state, $u_k \in \mathbb{R}^{l_u}$ is the control input, $w_k \in \mathbb{R}^{l_w}$ is the disturbance, $y_k \in \mathbb{R}^{l_y}$ is the measured output, and $v_k \in \mathbb{R}^{l_y}$ is the sensor noise. Define the command-following error

$$z_k \triangleq r_k - y_k, \tag{17}$$

where $r_k \in \mathbb{R}^{l_y}$ is the command signal. The objective of the adaptive control problem is to minimize the magnitude of z_k in the presence of w_k and v_k.

We define the strictly proper DTLTV controller using the input–output model

$$u_k = \sum_{i=1}^{n_c} P_{i,k}u_{k-i} + \sum_{i=1}^{n_c} Q_{i,k}z_{k-i}, \tag{18}$$

where $k \geq 0$, n_c is the order of the controller, and $Q_{1,k}, \ldots, Q_{n_c,k} \in \mathbb{R}^{l_u \times l_y}$ and $P_{1,k}, \ldots, P_{n_c,k} \in \mathbb{R}^{l_u \times l_u}$ are the numerator and denominator controller coefficient matrices, respectively. For convenience, a "cold" startup is assumed, where $Q_{1,0}, \ldots, Q_{n_c,0}$, $P_{1,0}, \ldots, P_{n_c,0}$, $u_{-n_c}, \ldots, u_{-1}$, and $z_{-n_c}, \ldots, z_{-1}$ are defined to be zero, and thus $u_0 = 0$.

Note that (18) can be written as

$$u_k = \phi_k \theta_{c,k}, \tag{19}$$

where

$$\phi_k \triangleq \begin{bmatrix} u_{k-1} \\ \vdots \\ u_{k-n_c} \\ z_{k-1} \\ \vdots \\ z_{k-n_c} \end{bmatrix}^{\mathsf{T}} \otimes I_{l_u} \in \mathbb{R}^{l_u \times l_{\theta_c}} \tag{20}$$

is the *controller regressor*,

$$\theta_{c,k} \triangleq \operatorname{vec}\begin{bmatrix} P_{1,k} & \cdots & P_{n_c,k} & Q_{1,k} & \cdots & Q_{n_c,k} \end{bmatrix} \in \mathbb{R}^{l_{\theta_c}} \tag{21}$$

is the *controller coefficient vector*, $l_{\theta_c} \triangleq n_c l_u (l_u + l_y)$, and "vec" is the column-stacking operator. In terms of $\mathbf{q}^{-1}$, the controller (18) can be expressed as

$$u_k = G_{c,k}(\mathbf{q}^{-1})z_k, \tag{22}$$

where

$$G_{c,k} \triangleq D_{c,k}^{-1}N_{c,k}, \tag{23}$$

$$D_{c,k}(\mathbf{q}^{-1}) \triangleq I_{l_u} - P_{1,k}\mathbf{q}^{-1} - \cdots - P_{n_c,k}\mathbf{q}^{-n_c}, \tag{24}$$

$$N_{c,k}(\mathbf{q}^{-1}) \triangleq Q_{1,k}\mathbf{q}^{-1} + \cdots + Q_{n_c,k}\mathbf{q}^{-n_c}. \tag{25}$$

To update (21), define the filtered signals

$$u_{f,k} \triangleq G_{f,k}(\mathbf{q}^{-1})u_k, \tag{26}$$

$$\phi_{f,k} \triangleq G_{f,k}(\mathbf{q}^{-1})\phi_k, \tag{27}$$

where $G_{f,k}$ is an $l_y \times l_u$ filter of order $n_f \geq 1$.

Next, define the *retrospective performance variable*

$$z_{rp,k}(\hat{\theta}) \triangleq z_k - u_{f,k} + \phi_{f,k}\hat{\theta}, \tag{28}$$

where $\hat{\theta} \in \mathbb{R}^{l_{\theta_c}}$ is the optimization variable. The rationale underlying the definition given in (28) is to replace u_k with $\phi_k\theta^*$, where θ^* is the retrospectively optimized controller coefficient vector obtained by optimization. The updated controller thus has the coefficients $\theta_{c,k+1} = \theta^*$. Note that $u_{f,k}$ depends on u_k and thus on the current controller coefficient vector $\theta_{c,k}$.

The retrospective performance variable $z_{rp,k}(\hat{\theta})$ is used to determine the updated controller coefficient vector $\theta_{c,k+1}$ by minimizing a function of $z_{rp,k}(\hat{\theta})$ namely the *retrospective cost function* defined as

$$J_k(\hat{\theta}) \triangleq \sum_{i=0}^{k} z_{rp,i}(\hat{\theta})^{\mathsf{T}} z_{rp,i}(\hat{\theta}) + (\hat{\theta} - \theta_{c,0})^{\mathsf{T}} P_{c,0}^{-1}(\hat{\theta} - \theta_{c,0}), \tag{29}$$

where $P_{c,0} \in \mathbb{R}^{l_{\theta_c} \times l_{\theta_c}}$ is positive definite. Then, for all $k \geq 0$, the unique global minimizer $\theta_{c,k+1}$ of J_k is given by the recursive least squares (RLS) solution [25]

$$P_{c,k+1} = P_{c,k} - P_{c,k}\phi_{f,k}^{\mathsf{T}}(I_{l_y} + \phi_{f,k}P_{c,k}\phi_{f,k}^{\mathsf{T}})^{-1}\phi_{f,k}P_{c,k}, \tag{30}$$

$$\theta_{c,k+1} = \theta_{c,k} + P_{c,k+1}\phi_{f,k}^{\mathsf{T}}z_{rp,k}(\theta_{c,k}). \tag{31}$$

Using the updated controller coefficient vector given by (31), the requested control at step $k+1$ is obtained by replacing k by $k+1$ in (19). Note that $P_{c,0}$ is a tuning parameter. As discussed in [4,8], for the case of linear, time-invariant single-input single-output plants, G_f is constructed based on the relative degree, leading numerator coefficient, and nonminimum-phase (NMP) zeros of the plant.

Note that, since $z_{rp,k}$ is used to determine the updated controller coefficient vector $\theta_{c,k+1}$ by minimizing J_k, which depends on $z_{rp,k}$, the optimal value of $z_{rp,k}$ is given by

$$z_{rp,k}(\theta_{c,k+1}) = z_k - u_{f,k} + \phi_{f,k}\theta_{c,k+1}, \tag{32}$$

which shows that the updated controller coefficient vector $\theta_{c,k+1}$ is "applied" retrospectively with the filtered controller regressor $\phi_{f,k}$. Furthermore, note that, in (32), $G_{f,k}$ is used to obtain $\phi_{f,k}$ from ϕ_k by means of (27) but does not include past changes in the controller coefficient vector, as indicated by the product $\phi_{f,k}\theta_{c,k+1}$. In effect, $\theta_{c,k+1}$ is fixed over $[k - n_f, k]$.

4. Retrospective-performance-variable decomposition

This section shows that the retrospective performance variable can be decomposed into the sum of a predicted-performance term and a model-matching term.

Define the *virtual external input perturbation*

$$\widetilde{u}_k(\hat{\theta}) \triangleq u_k - \phi_k\hat{\theta}. \tag{33}$$

Let $\widetilde{u}_{f,k}(\theta_{c,k+1})$ be given by the FIA filter

$$\widetilde{u}_{f,k}(\theta_{c,k+1}) \triangleq G_{f,k}(\mathbf{q}^{-1})\widetilde{u}_k(\theta_{c,\overline{k+1}}). \tag{34}$$

Note that $\widetilde{u}_{f,k}(\theta_{c,k+1})$ ignores the change in the argument $\theta_{c,k+1}$ of $\widetilde{u}_k$ over the interval $[k - n_f, k]$ in accordance with retrospective optimization. Using (34), it follows that (32) can be written as

$$\hat{z}_{rp,k}(\theta_{c,k+1}) \triangleq z_k - \widetilde{u}_{f,k}(\theta_{c,k+1}). \tag{35}$$

Theorem 4.1 given below presents the *retrospective-performance-variable decomposition*, which shows that $z_{rp,k}$ is the sum of the closed-loop performance and a measure of the extent to which the updated

closed-loop transfer function from $\widetilde{u}_k(\theta_{\mathrm{c},k+1})$ to z_k matches $G_{\mathrm{f},k}$. Since $G_{\mathrm{f},k}$ provides a matching objective for the closed-loop transfer function from $\widetilde{u}_k(\theta_{\mathrm{c},k+1})$ to z_k, the filter $G_{\mathrm{f},k}$ is henceforth called the *target model*. The statement of Theorem 4.1 depends on the matrices

$$A_{\mathrm{c},k} \triangleq \begin{bmatrix} 0 & \cdots & 0 & P_{n_{\mathrm{c}},k+n_{\mathrm{c}}} \\ I & \cdots & 0 & P_{n_{\mathrm{c}}-1,k+n_{\mathrm{c}}-1} \\ \vdots & \cdots & \vdots & \vdots \\ 0 & \cdots & I & P_{1,k+1} \end{bmatrix} \in \mathbb{R}^{l_u n_{\mathrm{c}} \times l_u n_{\mathrm{c}}}, \tag{36}$$

$$B_{\mathrm{c},k} \triangleq \begin{bmatrix} Q_{n_{\mathrm{c}},k+n_{\mathrm{c}}} \\ Q_{n_{\mathrm{c}}-1,k+n_{\mathrm{c}}-1} \\ \vdots \\ Q_{1,k+1} \end{bmatrix} \in \mathbb{R}^{l_u n_{\mathrm{c}} \times l_y}, \tag{37}$$

$$\widetilde{B}_{\mathrm{c},k} \triangleq \begin{bmatrix} P_{n_{\mathrm{c}},k+n_{\mathrm{c}}} \\ P_{n_{\mathrm{c}}-1,k+n_{\mathrm{c}}-1} \\ \vdots \\ P_{1,k+1} \end{bmatrix} \in \mathbb{R}^{l_u n_{\mathrm{c}} \times l_u}, \quad C_{\mathrm{c}} \triangleq \begin{bmatrix} 0_{l_u \times l_u (n_{\mathrm{c}}-1)} & I_{l_u} \end{bmatrix}, \tag{38}$$

$$\hat{A}_k \triangleq \begin{bmatrix} A_k & B_k C_{\mathrm{c}} \\ -B_{\mathrm{c},k+1} C_k & A_{\mathrm{c},k+1} \end{bmatrix} \in \mathbb{R}^{(l_x + l_u n_{\mathrm{c}}) \times (l_x + l_u n_{\mathrm{c}})}, \tag{39}$$

$$\hat{B}_k \triangleq \begin{bmatrix} B_{w,k} & 0_{l_x \times l_y} & B_k \\ 0_{l_u n_{\mathrm{c}} \times l_w} & B_{\mathrm{c},k+1} & \widetilde{B}_{\mathrm{c},k+1} \end{bmatrix} \in \mathbb{R}^{(l_x + l_u n_{\mathrm{c}}) \times (l_w + l_y + l_u)}, \tag{40}$$

$$\hat{C}_k \triangleq \begin{bmatrix} -C_k & 0_{l_y \times l_u n_{\mathrm{c}}} \end{bmatrix} \in \mathbb{R}^{l_y \times (l_x + l_u n_{\mathrm{c}})}, \tag{41}$$

$$\hat{D} \triangleq \begin{bmatrix} 0_{l_y \times l_w} & I_{l_y} & 0_{l_y \times l_u} \end{bmatrix}. \tag{42}$$

Theorem 4.1. *Let $z_{\mathrm{rp},k}$ be defined by (28) and let $\theta_{\mathrm{c},k}$ denote the controller coefficient vector at step k. Then, for all $k \geq 0$,*

$$z_{\mathrm{rp},k}(\theta_{\mathrm{c},k+1}) = z_{\mathrm{pp},k}(\theta_{\mathrm{c},k+1}) + z_{\mathrm{mm},k}(\theta_{\mathrm{c},k+1}), \tag{43}$$

where the predicted-performance term $z_{\mathrm{pp},k}(\theta_{\mathrm{c},k+1})$ and the model-matching term $z_{\mathrm{mm},k}(\theta_{\mathrm{c},k+1})$ are defined by

$$z_{\mathrm{pp},k}(\theta_{\mathrm{c},k+1}) \triangleq G_{z\bar{u},k}(\mathbf{q}^{-1})\bar{u}_k, \tag{44}$$

$$z_{\mathrm{mm},k}(\theta_{\mathrm{c},k+1}) \triangleq G_{z\widetilde{u},k}(\mathbf{q}^{-1})\widetilde{u}_k(\theta_{\mathrm{c},k+1}) - G_{\mathrm{f},k}(\mathbf{q}^{-1})\widetilde{u}_k(\theta_{\mathrm{c},\overline{k+1}}), \tag{45}$$

and $\bar{u}_k \triangleq \begin{bmatrix} w_k^{\mathsf{T}} & (r_k - v_k)^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$. Furthermore, the time-domain transfer functions $G_{z\bar{u},k} \in \mathbb{R}(\mathbf{q}^{-1})_{\mathrm{prop}}^{l_y \times (l_w + l_y)}$ and $G_{z\widetilde{u},k} \in \mathbb{R}(\mathbf{q}^{-1})_{\mathrm{prop}}^{l_y \times l_u}$ are defined by

$$\begin{bmatrix} G_{z\bar{u},k} & G_{z\widetilde{u},k} \end{bmatrix} \triangleq G_{z\hat{u},k}, \tag{46}$$

where $G_{z\hat{u},k}$ is the time-domain transfer function corresponding to the state–space model

$$\hat{x}_{k+1} = \hat{A}_k \hat{x}_k + \hat{B}_k \hat{u}_k, \tag{47}$$

$$z_k = \hat{C}_k \hat{x}_k + \hat{D}\hat{u}_k, \tag{48}$$

at step k, where $\hat{u}_k \triangleq \begin{bmatrix} \bar{u}_k^{\mathsf{T}} & \widetilde{u}_k(\theta_{\mathrm{c},k+1})^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$, $\hat{x}_0 \triangleq \begin{bmatrix} x_0^{\mathsf{T}} & 0_{1 \times l_u n_{\mathrm{c}}} \end{bmatrix}^{\mathsf{T}}$, and $\hat{A}_k$, $\hat{B}_k$, $\hat{C}_k$, and $\hat{D}$ are defined by (39)–(42).

Proof. Note that (20) and (21) imply that

$$\phi_k \theta_{\mathrm{c},k+1} = \sum_{i=1}^{n_{\mathrm{c}}} P_{i,k+1} u_{k-i} + \sum_{i=1}^{n_{\mathrm{c}}} Q_{i,k+1} z_{k-i}. \tag{49}$$

Substituting $\hat{\theta} = \theta_{\mathrm{c},k+1}$ and (49) into (33) yields

$$u_k = \widetilde{u}_k(\theta_{\mathrm{c},k+1}) + \sum_{i=1}^{n_{\mathrm{c}}} P_{i,k+1} u_{k-i} + \sum_{i=1}^{n_{\mathrm{c}}} Q_{i,k+1} z_{k-i}. \tag{50}$$

Using (24) and (25), it follows from (50) that

$$u_k = \widetilde{u}_k(\theta_{\mathrm{c},k+1}) + u_k - D_{\mathrm{c},k+1}(\mathbf{q}^{-1})u_k + N_{\mathrm{c},k+1}(\mathbf{q}^{-1})z_k,$$

which, using (23), can be rewritten as

$$u_k = D_{\mathrm{c},k+1}^{-1}(\mathbf{q}^{-1})\widetilde{u}_k(\theta_{\mathrm{c},k+1}) + G_{\mathrm{c},k+1}(\mathbf{q}^{-1})z_k. \tag{51}$$

Note that (24), (25), and Proposition A.2 imply that a state–space realization of (51) is given by

$$x_{\mathrm{c},k+1} = A_{\mathrm{c},k+1} x_{\mathrm{c},k} + B_{\mathrm{c},k+1} z_k + \widetilde{B}_{\mathrm{c},k+1} \widetilde{u}_k(\theta_{\mathrm{c},k+1}), \tag{52}$$

$$u_k = C_{\mathrm{c}} x_{\mathrm{c},k} + \widetilde{u}_k(\theta_{\mathrm{c},k+1}), \tag{53}$$

where $A_{\mathrm{c},k}$, $B_{\mathrm{c},k}$, $\widetilde{B}_{\mathrm{c},k}$, and C_{c} are defined by (36)–(38), and $x_{\mathrm{c},0} \triangleq 0_{l_u n_{\mathrm{c}} \times 1}$.

Next, substituting (16) into (17) yields

$$\begin{aligned} z_k &= -C_k x_k - v_k + r_k \\ &= -C_k x_k + \begin{bmatrix} 0 & I_{l_y} \end{bmatrix} \bar{u}_k. \end{aligned} \tag{54}$$

Furthermore, substituting (53) into (15) yields

$$\begin{aligned} x_{k+1} &= A_k x_k + B_k C_{\mathrm{c}} x_{\mathrm{c},k} + B_k \widetilde{u}_k(\theta_{\mathrm{c},k+1}) + B_{w,k} w_k \\ &= A_k x_k + B_k C_{\mathrm{c}} x_{\mathrm{c},k} + \begin{bmatrix} B_{w,k} & 0 \end{bmatrix} \bar{u}_k + B_k \widetilde{u}_k(\theta_{\mathrm{c},k+1}), \end{aligned} \tag{55}$$

and substituting (54) into (52) yields

$$\begin{aligned} x_{\mathrm{c},k+1} &= -B_{\mathrm{c},k+1} C_k x_k + A_{\mathrm{c},k+1} x_{\mathrm{c},k} + \widetilde{B}_{\mathrm{c},k+1} \widetilde{u}_k(\theta_{\mathrm{c},k+1}) - B_{\mathrm{c},k+1} v_k + B_{\mathrm{c},k+1} r_k \\ &= -B_{\mathrm{c},k+1} C_k x_k + A_{\mathrm{c},k+1} x_{\mathrm{c},k} + \begin{bmatrix} 0 & B_{\mathrm{c},k+1} \end{bmatrix} \bar{u}_k + \widetilde{B}_{\mathrm{c},k+1} \widetilde{u}_k(\theta_{\mathrm{c},k+1}). \end{aligned} \tag{56}$$

Define $\hat{x}_k \triangleq \begin{bmatrix} x_k^{\mathsf{T}} & x_{\mathrm{c},k}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$. Then, (47) and (48) follow from (54), (55), and (56). Since $G_{z\hat{u},k}$ is the time-domain transfer function corresponding to the state–space model (47), (48) at step k, it follows from (46) that

$$z_k = G_{z\bar{u},k}(\mathbf{q}^{-1})\bar{u}_k + G_{z\widetilde{u},k}(\mathbf{q}^{-1})\widetilde{u}_k(\theta_{\mathrm{c},k+1}). \tag{57}$$

Finally, substituting (57) into (35) yields (43). □

Note that $G_{z\hat{u},k}$ is obtained from (47), (48) in accordance with Definition 2.1 and Proposition A.1. In order to apply Proposition A.1, (47), (48) must be transformed to a completely observable state–space model. The time-varying eigensystem realization algorithm explained in Section 4 of [26] provides a method for reducing a given DTLTV state–space model to a minimal DTLTV state–space model.

5. Analysis of the retrospective-performance-variable decomposition

In this section, we analyze the retrospective-performance-variable decomposition given by Theorem 4.1.

Using (43) in (29) yields

$$\begin{aligned} J_k(\theta_{\mathrm{c},k+1}) = \sum_{i=0}^{k} \Big(& z_{\mathrm{pp},i}(\theta_{\mathrm{c},k+1})^{\mathsf{T}} z_{\mathrm{pp},i}(\theta_{\mathrm{c},k+1}) + z_{\mathrm{mm},i}(\theta_{\mathrm{c},k+1})^{\mathsf{T}} z_{\mathrm{mm},i}(\theta_{\mathrm{c},k+1}) \\ & + 2 z_{\mathrm{pp},i}(\theta_{\mathrm{c},k+1})^{\mathsf{T}} z_{\mathrm{mm},i}(\theta_{\mathrm{c},k+1}) \Big) \\ & + (\theta_{\mathrm{c},k+1} - \theta_{\mathrm{c},0})^{\mathsf{T}} P_0^{-1} (\theta_{\mathrm{c},k+1} - \theta_{\mathrm{c},0}). \end{aligned} \tag{58}$$

Note that, in (58), the first two terms in the sum are nonnegative, whereas the third term can have an arbitrary sign. This suggests that RLS can minimize $J_k(\theta_{\mathrm{c},k+1})$ by making the third term negative while the nonnegative terms remain large. In the case where P_0 is large, using RLS to minimize (58) yields, for $k \geq k_0 \in \mathbb{R}$,

$$z_{\mathrm{rp},k}(\theta_{\mathrm{c},k+1}) \approx 0, \tag{59}$$

which, using (43), implies that

$$z_{\mathrm{pp},k}(\theta_{\mathrm{c},k+1}) \approx -z_{\mathrm{mm},k}(\theta_{\mathrm{c},k+1}). \tag{60}$$

Note that (60) implies that $z_{\mathrm{pp},k}$ and $z_{\mathrm{mm},k}$ may be large in magnitude with opposite signs.

Next, we consider the initial conditions associated with $G_{z\bar{u},k}$ and $G_{z\widetilde{u},k}$. Since $G_{z\hat{u},k}$ is the time-domain transfer function corresponding to (47), (48) at step k, it follows from (46) that

$$z_k = G_{z\hat{u},k}\hat{u}_k = G_{z\bar{u},k}(\mathbf{q}^{-1})\bar{u}_k + G_{z\widetilde{u},k}(\mathbf{q}^{-1})\widetilde{u}_k(\theta_{\mathrm{c},k+1}). \tag{61}$$

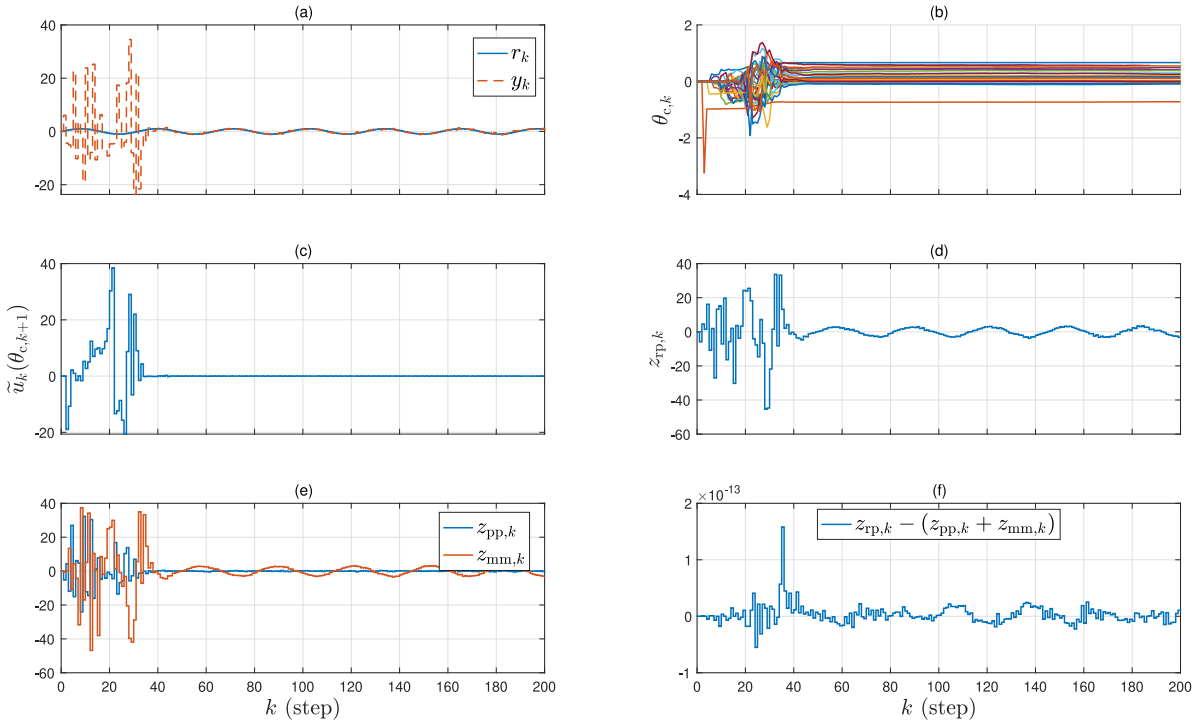


Fig. 1. *Example* 6.1 *with* G_f *given by* (65). (a) After an initial transient of 50 steps, the output y follows the command signal r. (b) The estimator coefficient θ_c converges after about 50 steps. (c) The virtual external input perturbation $\widetilde{u}$ converges to zero after about 50 steps. (d) For all $k \geq 50$, $z_{rp,k} \approx 0$. (e) For all $k \geq 50$, $z_{pp,k} \approx z_{mm,k} \approx 0$. (f) For all $k \geq 0$, $|z_{rp,k} - (z_{pp,k} + z_{mm,k})| \leq 2 \times 10^{-13}$, which confirms (43).

Furthermore, (46), (47), and (48) imply that $G_{z\bar{u},k}$ and $G_{z\widetilde{u},k}$ are the time-domain transfer functions corresponding to the state–space models $(\hat{A}_k, B_{\bar{u},k}, \hat{C}_k, D_{\bar{u}})$ and $(\hat{A}_k, B_{\widetilde{u},k}, \hat{C}_k, 0_{l_y \times l_u})$ at step k, respectively, where

$$B_{\bar{u},k} \triangleq \begin{bmatrix} B_{w,k} & 0_{l_x \times l_y} \\ 0_{l_u n_c \times l_w} & B_{c,k+1} \end{bmatrix}, \quad D_{\bar{u}} \triangleq \begin{bmatrix} 0_{l_y \times l_w} & I_{l_y} \end{bmatrix}, \quad B_{\widetilde{u},k} \triangleq \begin{bmatrix} B_k \\ \widetilde{B}_{c,k+1} \end{bmatrix}. \tag{62}$$

Note from Theorem 4.1 that $\hat{x}_0$ is the initial condition associated with the state space model $(\hat{A}_k, \hat{B}_k, \hat{C}_k, \hat{D})$. Now, let the initial condition corresponding to the state–space model $(\hat{A}_k, B_{\bar{u},k}, \hat{C}_k, D_{\bar{u}})$ be denoted by $x_{\bar{u},0}$, and let the initial condition corresponding to the state–space model $(\hat{A}_k, B_{\widetilde{u},k}, \hat{C}_k, 0_{l_y \times l_u})$ be denoted by $x_{\widetilde{u},0}$. Since the transfer functions $G_{z\bar{u},k}$ and $G_{z\widetilde{u},k}$ have the same state matrix $\hat{A}_k$ in their corresponding state–space models and since z_k is additively decomposed into two parts based on the partitioning of the input vector $\hat{u}_k$ into $\bar{u}_k$ and $\widetilde{u}_k(\theta_{c,k+1})$, it follows that $\hat{x}_0 = x_{\bar{u},0} + x_{\widetilde{u},0}$. This implies that, in order to implement (44) and (45), $\hat{x}_0$ must be additively decomposed into two values, namely, $x_{\bar{u},0}$ and $x_{\widetilde{u},0}$. Since $x_{\bar{u},0}$ and $x_{\widetilde{u},0}$ are not uniquely determined, it follows that the transient responses of z_{pp} and z_{mm}, which depend on the values of $x_{\bar{u},0}$ and $x_{\widetilde{u},0}$, are not uniquely determined.

6. Numerical examples

This section presents numerical examples to illustrate the retrospective-performance-variable decomposition. These examples demonstrate how the decomposition can be used for the analysis of RCAC.

Example 6.1. This example illustrates how the performance of RCAC is affected by the choice of G_f. In particular, this example shows that the retrospective-performance-variable decomposition provides insight into understanding why nonminimum phase (NMP) zeros of the plant must be included in the filter G_f.

Consider the state–space model (15), (16), where, for all $k \geq 0$,

$$A_k = A \triangleq \begin{bmatrix} 0 & 1 \\ -1.3 & -0.67 \end{bmatrix}, \quad B_k = B \triangleq \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C_k = C \triangleq \begin{bmatrix} -1.1 & 1 \end{bmatrix}, \tag{63}$$

$w \equiv 0$, v_k is Gaussian white noise with mean zero and variance 0.01, and $x_0 = \begin{bmatrix} -2 & -2 \end{bmatrix}^{\mathsf{T}}$. Note that the transfer function corresponding to the state–space model in (63) is given by

$$G_d(\mathbf{q}^{-1}) \triangleq \frac{-\mathbf{q}^{-1} - 1.1\mathbf{q}^{-2}}{1 + 0.67\mathbf{q}^{-1} + 1.3\mathbf{q}^{-2}}, \tag{64}$$

which is unstable and has a NMP zero at 1.1.

First, RCAC is applied to command following with $r_k = \sin(0.2k)$, $n_c = 15$, $P_{c,0} = 10I_{30}$, and, for all $k \geq 0$,

$$G_{f,k}(\mathbf{q}^{-1}) = -\mathbf{q}^{-1} - 1.1\mathbf{q}^{-2}. \tag{65}$$

Note that G_f given by (65) contains the NMP zero of G_d. The convergence to zero of the error between the output y and the command signal r, the convergence of the estimator coefficients θ_c, and the convergence of the virtual external input perturbation $\widetilde{u}$ are shown in (a), (b), and (c), respectively, of Fig. 1. Furthermore, (d) and (e) of Fig. 1 show that, after an initial transient, (59) and (60) are satisfied. Finally, (f) of Fig. 1 shows that the difference between z_{rp} and $z_{pp} + z_{mm}$ is negligible, which confirms (43). In order to observe the asymptotic behavior of the time-domain transfer functions G_c, $G_{z\bar{u}}$, and $G_{z\widetilde{u}}$ after θ_c converges, the frequency-response plots of $G_{c,200}$, $G_{zw,200}$, and $G_{zr,200}$ are shown in (a), (c), (e), and (f) of Fig. 2, where $\begin{bmatrix} G_{zw,200} & G_{z,r-v,200} \end{bmatrix} = G_{z\bar{u},200}$, and the extent to which the frequency response of $G_{z\widetilde{u},200}$ matches the frequency response of $G_{f,200}$ is shown in (b) and (d) of Fig. 2.

Next, the simulation is repeated with, for all $k \geq 0$,

$$G_{f,k}(\mathbf{q}^{-1}) = -\mathbf{q}^{-1} \tag{66}$$

and with the remaining parameters unchanged. Since G_f given by (66) does not possess the NMP zero of G_d, which is unmoved by feedback,

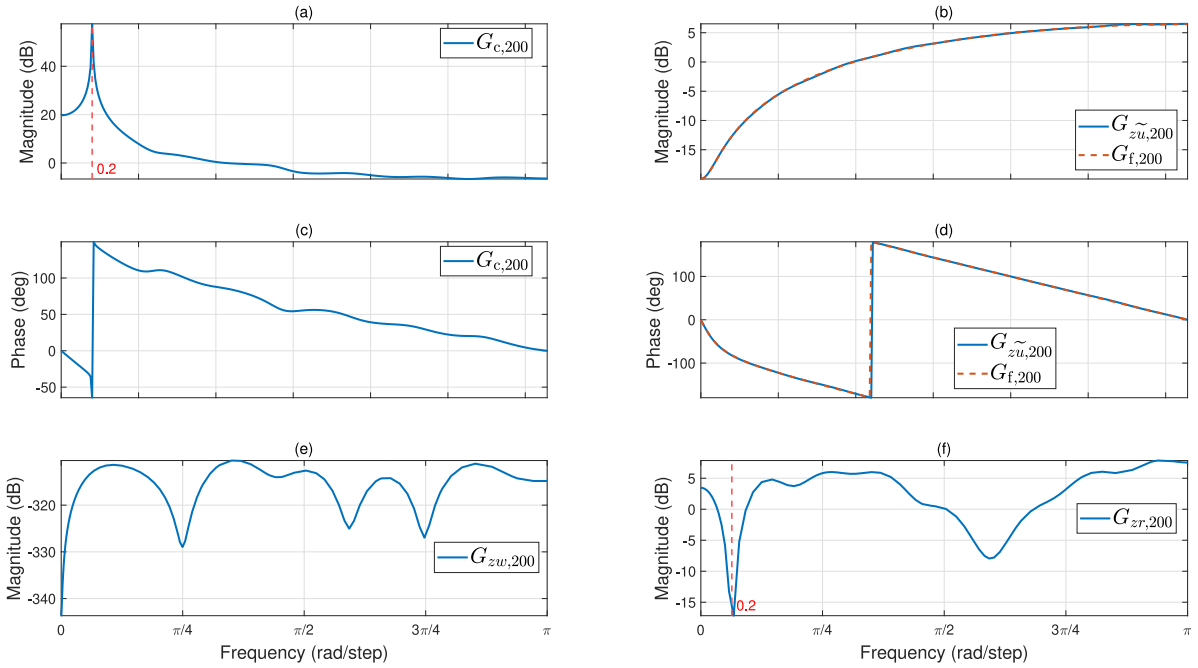


Fig. 2. *Example 6.1 with G_f given by (65).* (a), (c) The frequency-response plots of $G_{c,200}$. The magnitude of $G_{c,200}$ peaks at the frequency 0.2 rad/step, which is the frequency of the command signal r. (b), (d) Comparison of the frequency response of $G_{z\tilde{u},200}$ with the frequency response of G_f. The magnitude and phase plots match approximately. (e) The magnitude of $G_{zw,200}$ is approximately zero at all frequencies. (f) The magnitude of $G_{zr,200}$ at the frequency 0.2 rad/step is approximately zero. These observations show that, for large k, $z_{pp,k} \approx 0$ and $z_{mm,k} \approx 0$.

and since the closed-loop transfer function $G_{z\tilde{u}}$ tries to match the filter G_f, the converged controller has an unstable pole at 1.1, as shown in Fig. 5. This leads to unstable pole-zero cancellation, which makes the signals y, $\tilde{u}$, z_{rp}, and z_{mm} diverge, as shown in Fig. 3. The frequency-response plots of $G_{c,200}$, $G_{zw,200}$, and $G_{zr,200}$ are shown in (a), (c), (e), and (f) of Fig. 4, and the extent to which the frequency response of $G_{z\tilde{u},200}$ matches that of $G_{f,200}$ is shown in (b) and (d) of Fig. 4. The two choices of G_f in this example and their corresponding effect on the performance confirms the RCAC modeling requirement that G_f contain the NMP zeros of G_d. ⋄

Example 6.2. This example illustrates how the retrospective-performance-variable decomposition is affected by the initial conditions associated with $G_{z\bar{u},k}$ and $G_{z\tilde{u},k}$. In particular, we show that the initial conditions associated with $G_{z\bar{u},k}$ and $G_{z\tilde{u},k}$ affect only the transient response of the predicted-performance term and model-matching term and not the steady-state response.

Consider the state–space model given by (15), (16), where, for all $k \geq 0$,

$$A_k = 0.5 - 0.2\sin(0.01k), \quad B_k = B \triangleq 1, \quad C_k = C \triangleq 1, \tag{67}$$

$w \equiv v \equiv 0$, and $x_0 = 4$. Let the command signal be $r_k = \sin(0.2k)$. Let $n_c = 7, n_f = 2, P_{c,0} = 10I_{14}$, and, for all $k \geq 0$, let $G_{f,k}(\mathbf{q}^{-1}) = -\mathbf{q}^{-2} - \mathbf{q}^{-1}$.

First, the decomposition is performed with $x_{\bar{u},0} = 3.6$ and $x_{\tilde{u},0} = 0.4$, where $x_{\bar{u},0}$ and $x_{\tilde{u},0}$ are initial conditions corresponding to the state–space models $(\hat{A}_k, B_{\bar{u},k}, \hat{C}_k, D_{\bar{u}})$ and $(\hat{A}_k, B_{\tilde{u},k}, \hat{C}_k, 0_{l_y \times l_u})$, respectively. The convergence to zero of the error between the output y and the command signal r, the convergence of the estimator coefficients θ_c, and the convergence of the virtual external input perturbation $\tilde{u}$ are shown in (a), (b), and (c), respectively, of Fig. 6. Plots (d) and (e) of Fig. 6 show that, after an initial transient, (59) and (60) are satisfied. Plot (f) of Fig. 6 shows that the difference between z_{rp} and $z_{pp} + z_{mm}$ is negligible, which confirms (43). The frequency-response plots of $G_{c,200}$, $G_{zw,200}$, and $G_{zr,200}$ are shown in (a), (c), (e), and (f) of Fig. 7, and the extent to which the frequency response of $G_{z\tilde{u},200}$ matches that of $G_{f,200}$ is shown in (b) and (d) of Fig. 7. Note that, since the plant is time-varying, the time-domain transfer functions G_c, $G_{z\bar{u}}$, and $G_{z\tilde{u}}$ do not converge after the estimator coefficient θ_c converges. Next, the decomposition is repeated with $x_{\bar{u},0} = 8$ and $x_{\tilde{u},0} = -4$. The change in initial conditions changes the transient responses of z_{pp} and z_{mm}, as shown in (e) of Fig. 8. However, the asymptotic behavior of z_{pp} and z_{mm} is the same as in the previous case. ⋄

Example 6.3. This example illustrates the retrospective-performance-variable decomposition for a multi-input multi-output system. It also shows, through retrospective-performance-variable decomposition, how RCAC adapts when there are sudden changes in the nature of the command signal.

Consider the state–space model (15), (16), where, for all $k \geq 0$,

$$A_k = A \triangleq \begin{bmatrix} 0.8777 & 0.0847 & 0.0777 & 0.0338 \\ -0.7630 & 0.1363 & 0.4241 & 0.2262 \\ 0.1436 & 0.0423 & 0.8429 & 0.1333 \\ 1.0741 & 0.2827 & -1.2433 & 0.4077 \end{bmatrix}, \tag{68}$$

$$B_k = B \triangleq \begin{bmatrix} 0.0112 & 0.0034 \\ 0.0847 & 0.0423 \\ 0.0034 & 0.0191 \\ 0.0423 & 0.1666 \end{bmatrix}, \tag{69}$$

$$B_{w,k} = B_w \triangleq \begin{bmatrix} 0.01 \\ 0.01 \\ 0.01 \\ 0.01 \end{bmatrix}, \quad C_k = C \triangleq \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{70}$$

w_k is standard Gaussian white noise, v_k is Gaussian white noise with mean zero and variance $10^{-4}I_2$, and $x_0 = \begin{bmatrix} 1 & -1 & 0 & 0.5 \end{bmatrix}^{\mathrm{T}}$. Let the command signal be

$$r_k = \begin{cases} \begin{bmatrix} 1 & 2 \end{bmatrix}^{\mathrm{T}}, & k \leq 300, \\ \begin{bmatrix} -1 & -3 \end{bmatrix}^{\mathrm{T}}, & k > 300. \end{cases} \tag{71}$$

Let $n_c = 2, n_f = 1, P_{c,0} = I_{16}$, and, for all $k \geq 0$, let $G_{f,k}(\mathbf{q}^{-1}) = -CB\mathbf{q}^{-1}$. The convergence to zero of the error between the output y and the command signal r, the convergence of the estimator coefficients θ_c, and

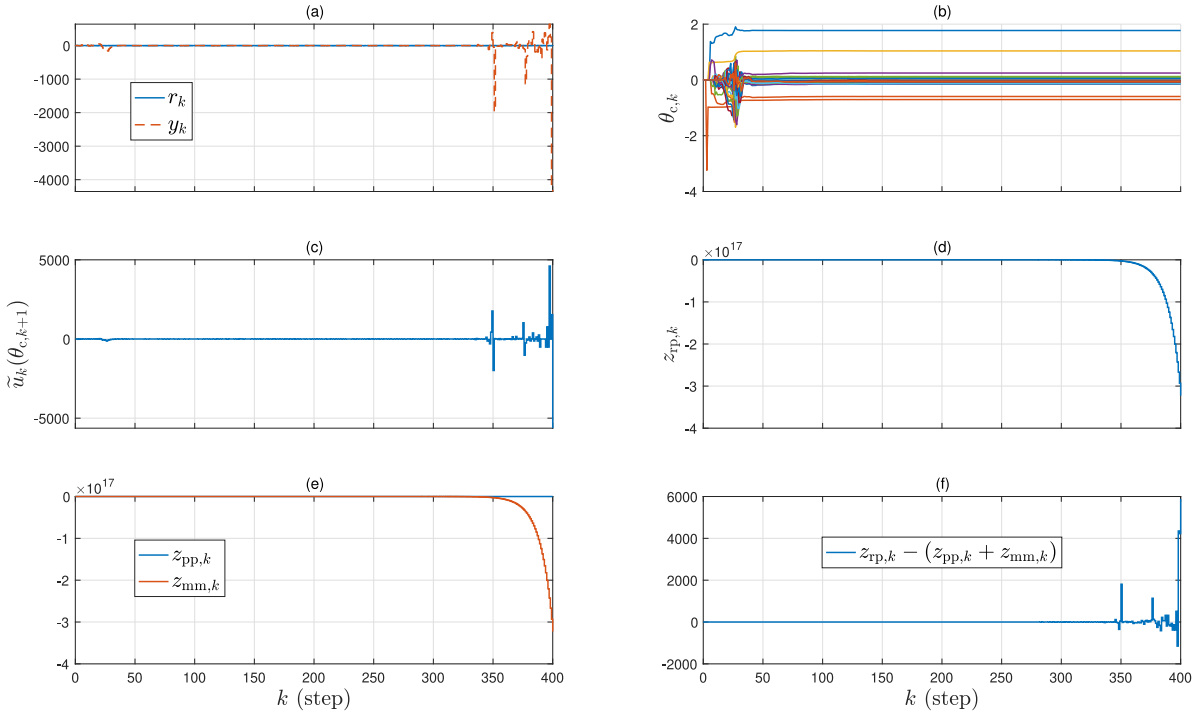


Fig. 3. *Example* 6.1 *with* G_f *given by* (66). The estimator coefficients converge, but y, $\widetilde{u}$, and z_{rp} diverge. Since z_{pp} is computed using an input–output model corresponding to a minimal state–space model, z_{pp} does not diverge. Although z_{mm} is also computed using an input–output model corresponding to a minimal state–space model, since $\widetilde{u}$ diverges, z_{mm} diverges.

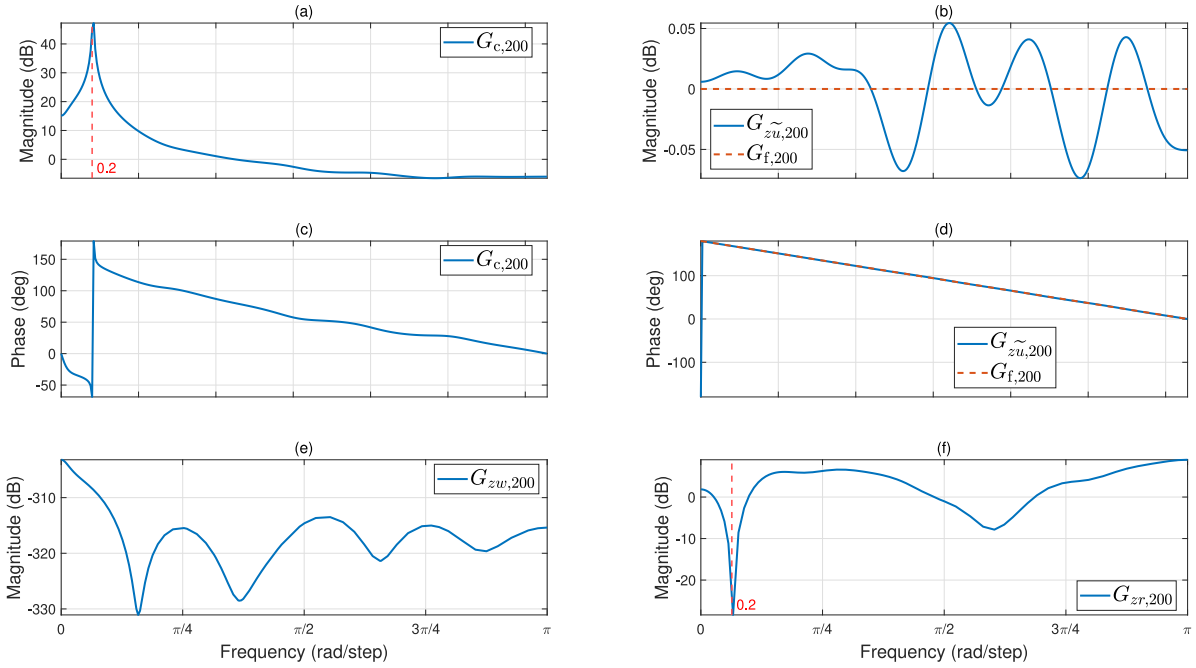


Fig. 4. *Example* 6.1 *with* G_f *given by* (66). (a), (c) The frequency-response plots of $G_{c,200}$. The magnitude of $G_{c,200}$ peaks at the frequency 0.2 rad/step, which is the frequency of the command signal r. (b), (d) Comparison of the frequency response of $G_{\widetilde{zu},200}$ with the frequency response of G_f. The magnitude and phase plots match approximately. (e) The magnitude of $G_{zw,200}$ is approximately zero at all frequencies. (f) The magnitude of $G_{zr,200}$ at the frequency 0.2 rad/step is approximately zero. However, these observations do not ensure that RCAC performs as expected. Due to a hidden instability, the output y diverges, as shown in Fig. 3.

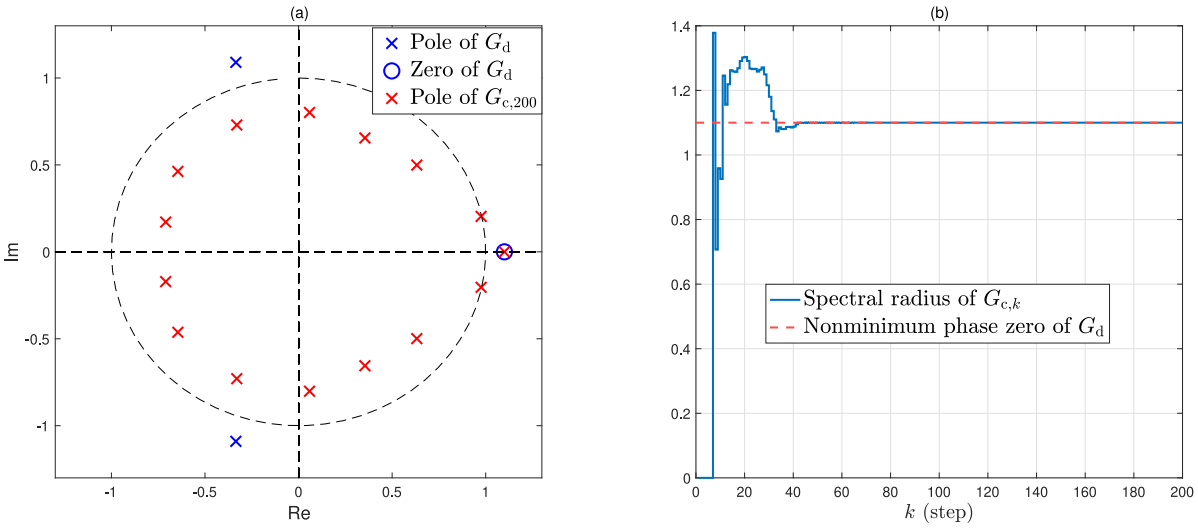


Fig. 5. *Example 6.1 with G_f given by (66).* (a) Pole-zero plots of G_d and $G_{c,200}$. $G_{c,200}$ has a pole at the NMP zero of G_d. This leads to an unstable pole-zero cancellation in the closed-loop system. (b) shows that the spectral radius of $G_{c,k}$ converges to the NMP zero of G_d.

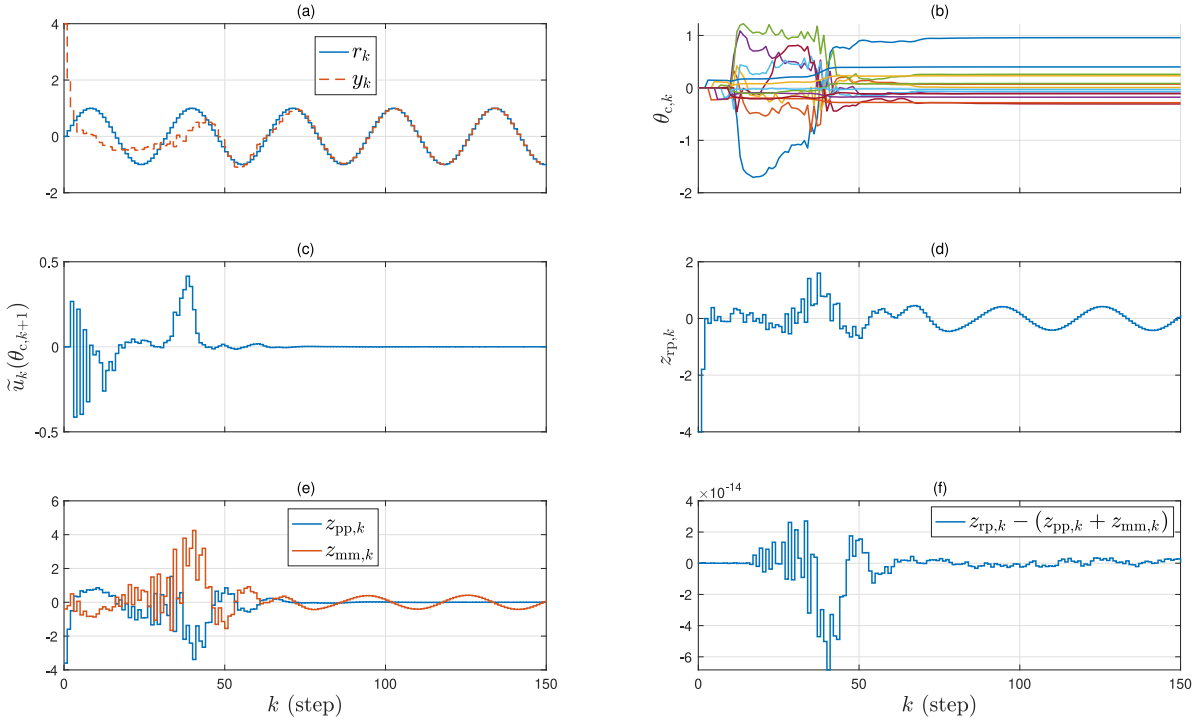


Fig. 6. *Example 6.2 with $x_{\widetilde{u},0} = 3.6$ and $x_{\widetilde{u},0} = 0.4$.* (a) After an initial transient of 75 steps, the output y follows the command signal r. (b) The estimator coefficient θ_c converges after about 75 steps. (c) The virtual external input perturbation $\widetilde{u}$ converges to zero after about 75 steps. (d) For all $k \geq 75$, $z_{rp,k} \approx 0$. (e) For all $k \geq 75$, $z_{pp,k} \approx z_{mm,k} \approx 0$. (f) For all $k \geq 0$, $|z_{rp,k} - (z_{pp,k} + z_{mm,k})| \leq 10^{-13}$, which confirms (43).

the convergence of the virtual external input perturbation $\widetilde{u}$ are shown in (a), (b), and (c), respectively, of Fig. 9. Plots (d) and (e) of Fig. 9 show that, after an initial transient, (59) and (60) are satisfied. Plot (f) of Fig. 9 shows that the difference between z_{rp} and $z_{pp} + z_{mm}$ is negligible, which confirms (43). Note that, as shown in Fig. 9, when the nature of the input changes, RCAC re-adapts and subsequently converges. ⋄

7. Conclusions

This paper developed transformations between discrete-time linear time-varying (DTLTV) state–space models and DTLTV input–output models. These transformations were used to derive and demonstrate the retrospective-performance-variable decomposition in retrospective cost adaptive control (RCAC). This decomposition shows how RCAC

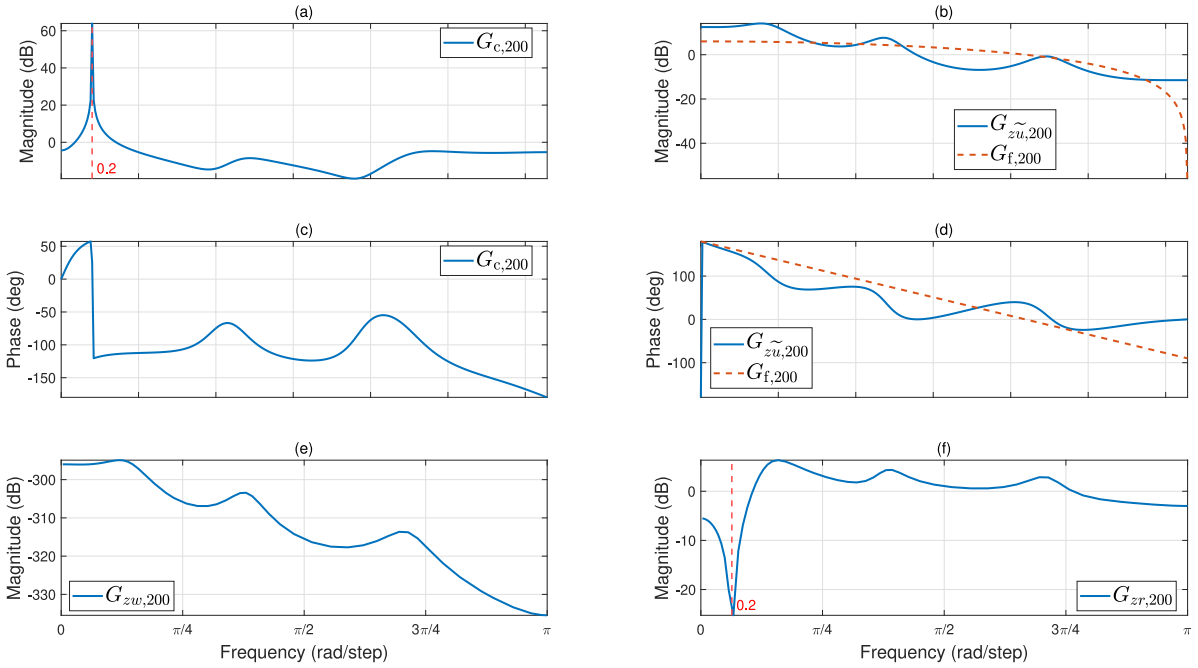


Fig. 7. *Example 6.2 with* $x_{\tilde{u},0} = 3.6$ *and* $x_{\tilde{u},0} = 0.4$. (a), (c) The frequency-response plots of $G_{c,200}$. The magnitude of $G_{c,200}$ peaks at the frequency 0.2 rad/step, which is the frequency of the command signal r. (b), (d) Comparison of the frequency response of $G_{z\tilde{u},200}$ with the frequency response of G_f. The magnitude and phase plots match approximately. (e) The magnitude of $G_{zw,200}$ is approximately zero at all frequencies. (f) The magnitude of $G_{zr,200}$ at the frequency 0.2 rad/step is approximately zero.

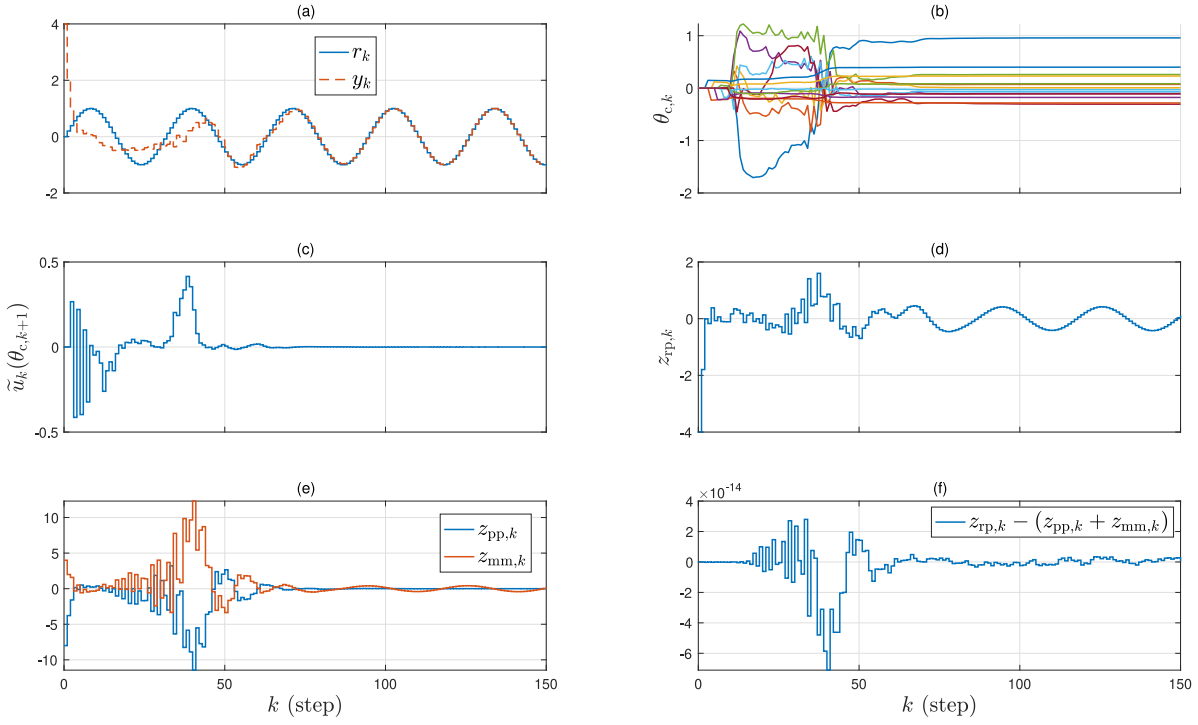


Fig. 8. *Example 6.2 with* $x_{\tilde{u},0} = 8$ *and* $x_{\tilde{u},0} = -4$. In (e), the transient responses of z_{pp} and z_{mm} are different from the case where $x_{\tilde{u},0} = 3.6$ and $x_{\tilde{u},0} = 0.4$ (Fig. 6). However, the asymptotic behavior of z_{pp} and z_{mm} are the same as in the previous case. All other plots are the same as in Fig. 6.

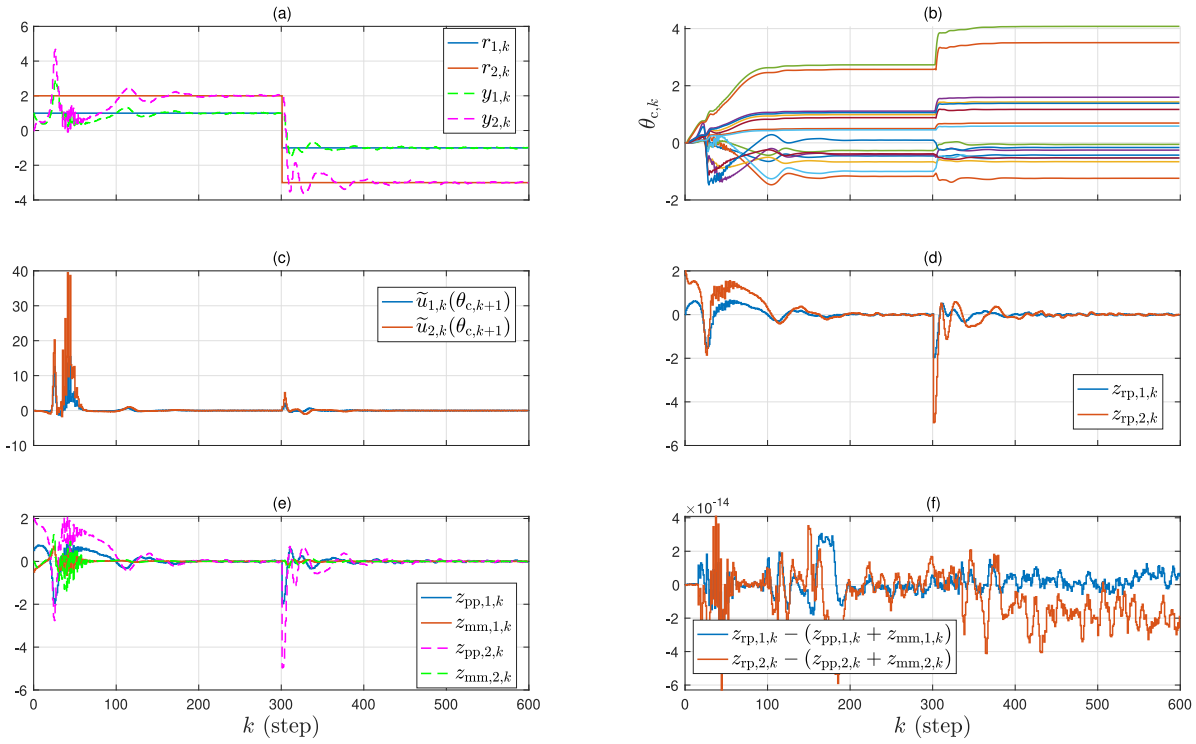


Fig. 9. *Example* 6.3. (a) The components y_1 and y_2 of the output y follow the components r_1 and r_2 of the command signal r. (b) The estimator coefficient θ_c converges. (c) The components $\widetilde{u}_1$ and $\widetilde{u}_2$ of the virtual external input perturbation $\widetilde{u}$ converge to zero. (d) For all $k \geq 400$, $z_{rp,1,k} \approx 0$ and $z_{rp,2,k} \approx 0$, where $z_{rp,k} = [z_{rp,1,k} \quad z_{rp,2,k}]^{\mathrm{T}}$. (e) For all $k \geq 400$, $z_{pp,1,k} \approx z_{mm,1,k} \approx 0$ and $z_{pp,1,k} \approx z_{mm,1,k} \approx 0$, where $z_{pp,k} = [z_{pp,1,k} \quad z_{pp,2,k}]^{\mathrm{T}}$ and $z_{mm,k} = [z_{mm,1,k} \quad z_{mm,2,k}]^{\mathrm{T}}$. (f) For all $k \geq 0$, $|z_{rp,k} - (z_{pp,k} + z_{mm,k})| \leq 10^{-13}$, which confirms (43).

achieves closed-loop performance and matches the closed-loop dynamics to the target model. These results and insights are a key step toward analyzing the convergence and asymptotic stability of RCAC algorithm.

CRediT authorship contribution statement

Sneha Sanjeevini: Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Writing – original draft. **Dennis S. Bernstein:** Conceptualization, Funding acquisition, Supervision, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This research was supported by National Science Foundation, USA under Grant CMMI 2031333 and the Air Force Office of Scientific Research, USA under grant FA9550-20-1-0028. The authors thank the reviewers for numerous helpful comments and suggestions.

Appendix A. Transformations between DTLTV state-space models and DTLTV input-output models

This section constructs DTLTV state–space realizations of DTLTV input–output models as well as DTLTV input–output models corresponding to DTLTV state–space models. These constructions were used in the decomposition of the retrospective performance variable in Section 4. The following result provides an explicit expression for a DTLTV input–output model corresponding to a DTLTV state–space model.

Proposition A.1. *Consider the DTLTV state–space model* (6), (7), *and assume that* $(\mathcal{A}, C)$ *is completely observable. Then, a DTLTV input–output model corresponding to* (6), (7) *is given by* (1), *where, for all* $k \in [0, n-1]$, $y_{\mathrm{io},k} \triangleq y_{\mathrm{ss},k}$, *and, for all* $k \geq n$,

$$N_{i,k} \triangleq \begin{cases} H_{0,k}, & i = 0, \\ H_{i,k} + \displaystyle\sum_{j=1}^{i} D_{j,k} H_{i-j,k-j}, & 1 \leq i \leq n, \end{cases} \tag{A.1}$$

$$\begin{bmatrix} D_{n,k} & \cdots & D_{1,k} \end{bmatrix} \triangleq -C_k \Psi_{k-1,k-n} \mathcal{O}^{\mathrm{L}}_{k-n}, \tag{A.2}$$

and $\mathcal{O}^{\mathrm{L}}_k$ *is a left inverse of* $\mathcal{O}_k$.

Proof. Post-multiplying (A.2) by $\mathcal{O}_{k-n}$ yields, for all $k \geq n$,

$$\begin{aligned} 0 &= \begin{bmatrix} D_{n,k} & \cdots & D_{1,k} \end{bmatrix} \mathcal{O}_{k-n} + C_k \Psi_{k-1,k-n} \\ &= C_k \Psi_{k-1,k-n} + D_{1,k} C_{k-1} \Psi_{k-2,k-n} + \cdots \\ &\quad + D_{n-1,k} C_{k-n+1} \Psi_{k-n,k-n} + D_{n,k} C_{k-n}. \end{aligned} \tag{A.3}$$

Next, it follows from (6), (7) that, for all $k \geq n$,

$$\begin{aligned} y_{\mathrm{ss},k} &= C_k \Psi_{k-1,k-n} x_{k-n} + C_k \sum_{i=0}^{n-2} \Psi_{k-1,k-n+i+1} B_{k-n+i} u_{k-n+i} \\ &\quad + C_k B_{k-1} u_{k-1} + E_k u_k. \end{aligned}$$

Hence, for all $k \geq n$,

$$\begin{aligned} y_{\mathrm{ss},k} &+ D_{1,k}y_{\mathrm{ss},k-1} + \cdots + D_{n-1,k}y_{\mathrm{ss},k-n+1} + D_{n,k}y_{\mathrm{ss},k-n} \\ &= C_k\Psi_{k-1,k-n}x_{k-n} + C_k\sum_{i=0}^{n-2}\Psi_{k-1,k-n+i+1}B_{k-n+i}u_{k-n+i} \\ &\quad + C_kB_{k-1}u_{k-1} + E_ku_k \\ &\quad + D_{1,k}\Big(C_{k-1}\Psi_{k-2,k-n}x_{k-n} \\ &\quad + C_{k-1}\sum_{i=0}^{n-3}\Psi_{k-2,k-n+i+1}B_{k-n+i}u_{k-n+i} + C_{k-1}B_{k-2}u_{k-2} \\ &\quad + E_{k-1}u_{k-1}\Big) + \cdots + D_{n-1,k}(C_{k-n+1}\Psi_{k-n,k-n}x_{k-n} \\ &\quad + C_{k-n+1}B_{k-n}u_{k-n} + E_{k-n+1}u_{k-n+1}) \\ &\quad + D_{n,k}(C_{k-n}x_{k-n} + E_{k-n}u_{k-n}). \end{aligned} \tag{A.4}$$

Regrouping the terms in (A.4) and using (11) yields, for all $k \geq n$,

$$\begin{aligned} y_{\mathrm{ss},k} &+ D_{1,k}y_{\mathrm{ss},k-1} + \cdots + D_{n-1,k}y_{\mathrm{ss},k-n+1} + D_{n,k}y_{\mathrm{ss},k-n} \\ &= (C_k\Psi_{k-1,k-n} + D_{1,k}C_{k-1}\Psi_{k-2,k-n} + \cdots \\ &\quad + D_{n-1,k}C_{k-n+1}\Psi_{k-n,k-n} + D_{n,k}C_{k-n})x_{k-n} \\ &\quad + \left(H_{n,k} + \sum_{j=1}^{n}D_{j,k}H_{n-j,k-j}\right)u_{k-n} \\ &\quad + \left(H_{n-1,k} + \sum_{j=1}^{n-1}D_{j,k}H_{n-1-j,k-j}\right)u_{k-n+1} \\ &\quad + \cdots + \left(H_{1,k} + D_{1,k}H_{0,k-1}\right)u_{k-1} + H_{0,k}u_k. \end{aligned} \tag{A.5}$$

Then, substituting (A.3) and (A.1) into (A.5) yields, for all $k \geq n$,

$$y_{\mathrm{ss},k} + D_{1,k}y_{\mathrm{ss},k-1} + \cdots + D_{n,k}y_{\mathrm{ss},k-n} = N_{0,k}u_k + \cdots + N_{n,k}u_{k-n}. \tag{A.6}$$

Since, for all $k \in [0, n-1]$, $y_{\mathrm{io},k} = y_{\mathrm{ss},k}$, it follows from (1) and (A.6) that, for all $k \geq n$, $y_{\mathrm{io},k} = y_{\mathrm{ss},k}$. □

The following result provides a DTLTV state–space realization of a DTLTV input–output model.

Proposition A.2. *A completely observable DTLTV state–space realization of the DTLTV input–output model in (1) is given by (6), (7), where, for all $k \geq 0$,*

$$A_k \triangleq \begin{bmatrix} 0 & \cdots & 0 & -D_{n,k+n} \\ I & \cdots & 0 & -D_{n-1,k+n-1} \\ \vdots & \cdots & \vdots & \vdots \\ 0 & \cdots & I & -D_{1,k+1} \end{bmatrix} \in \mathbb{R}^{pn\times pn}, \tag{A.7}$$

$$B_k \triangleq \begin{bmatrix} N_{n,k+n} - D_{n,k+n}N_{0,k} \\ N_{n-1,k+n-1} - D_{n-1,k+n-1}N_{0,k} \\ \vdots \\ N_{1,k+1} - D_{1,k+1}N_{0,k} \end{bmatrix} \in \mathbb{R}^{pn\times m}, \tag{A.8}$$

$$C_k \triangleq \begin{bmatrix} 0_{p\times p(n-1)} & I_p \end{bmatrix} \in \mathbb{R}^{p\times pn}, \quad E_k \triangleq N_{0,k} \in \mathbb{R}^{p\times m}. \tag{A.9}$$

Furthermore, for all $k \geq 0$, $x_k \triangleq \begin{bmatrix} x_{1,k}^{\mathrm{T}} & x_{2,k}^{\mathrm{T}} & \cdots & x_{n,k}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$, where, for all $i \in [0, n-1]$,

$$x_{n-i,k} \triangleq \sum_{j=i+1}^{n}N_{j,k+i}u_{k-j+i} - \sum_{j=i+1}^{n}D_{j,k+i}y_{\mathrm{io},k-j+i}. \tag{A.10}$$

Proof. Rearranging terms in (1) yields

$$y_{\mathrm{io},k} = \sum_{j=1}^{n}N_{j,k}u_{k-j} - \sum_{j=1}^{n}D_{j,k}y_{\mathrm{io},k-j} + N_{0,k}u_k. \tag{A.11}$$

Setting $i = 0$ in (A.10) yields

$$x_{n,k} = \sum_{j=1}^{n}N_{j,k}u_{k-j} - \sum_{j=1}^{n}D_{j,k}y_{\mathrm{io},k-j},$$

which, along with (A.11), implies that

$$x_{n,k} = y_{\mathrm{io},k} - N_{0,k}u_k. \tag{A.12}$$

Next, setting $i = n-1$ in (A.10) yields

$$x_{1,k} = N_{n,k+n-1}u_{k-1} - D_{n,k+n-1}y_{\mathrm{io},k-1}. \tag{A.13}$$

Hence, it follows from (A.12) and (A.13) that

$$\begin{aligned} x_{1,k+1} &= N_{n,k+n}u_k - D_{n,k+n}y_{\mathrm{io},k} \\ &= -D_{n,k+n}(y_{\mathrm{io},k} - N_{0,k}u_k) + (N_{n,k+n} - D_{n,k+n}N_{0,k})u_k \\ &= -D_{n,k+n}x_{n,k} + (N_{n,k+n} - D_{n,k+n}N_{0,k})u_k. \end{aligned} \tag{A.14}$$

Furthermore, for all $i \in [0, n-2]$, (A.10) and (A.12) imply that

$$\begin{aligned} x_{n-i,k+1} &= \sum_{j=i+1}^{n}N_{j,k+i+1}u_{k-j+i+1} - \sum_{j=i+1}^{n}D_{j,k+i+1}y_{\mathrm{io},k-j+i+1} \\ &= \sum_{j=i+2}^{n}N_{j,k+i+1}u_{k-j+i+1} - \sum_{j=i+2}^{n}D_{j,k+i+1}y_{\mathrm{io},k-j+i+1} \\ &\quad + N_{i+1,k+i+1}u_k - D_{i+1,k+i+1}y_{\mathrm{io},k} \\ &= x_{n-i-1,k} + N_{i+1,k+i+1}u_k - D_{i+1,k+i+1}y_{\mathrm{io},k} \\ &= x_{n-i-1,k} - D_{i+1,k+i+1}(y_{\mathrm{io},k} - N_{0,k}u_k) \\ &\quad + (N_{i+1,k+i+1} - D_{i+1,k+i+1}N_{0,k})u_k \\ &= x_{n-i-1,k} - D_{i+1,k+i+1}x_{n,k} + (N_{i+1,k+i+1} - D_{i+1,k+i+1}N_{0,k})u_k. \end{aligned} \tag{A.15}$$

Hence, it follows from (A.7), (A.8), (A.14), and (A.15) that

$$\begin{aligned} A_kx_k + B_ku_k &= \begin{bmatrix} -D_{n,k+n}x_{n,k} \\ x_{1,k} - D_{n-1,k+n-1}x_{n,k} \\ \vdots \\ x_{n-1,k} - D_{1,k+1}x_{n,k} \end{bmatrix} + \begin{bmatrix} N_{n,k+n} - D_{n,k+n}N_{0,k} \\ N_{n-1,k+n-1} - D_{n-1,k+n-1}N_{0,k} \\ \vdots \\ N_{1,k+1} - D_{1,k+1}N_{0,k} \end{bmatrix}u_k \\ &= \begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \\ \vdots \\ x_{n,k+1} \end{bmatrix} = x_{k+1}. \end{aligned}$$

Finally, (A.9) and (A.12) imply that

$$C_kx_k + E_ku_k = x_{n,k} + N_{0,k}u_k = y_{\mathrm{io},k}. \tag{A.16}$$

Comparing (7) and (A.16) implies that, for all $k \geq 0$, $y_{\mathrm{ss},k} = y_{\mathrm{io},k}$. Since, for all $k \geq 0$, $\operatorname{rank}\mathcal{O}_k = pn$, it follows that $(\mathcal{A}, C)$ is completely observable. □

References

[1] P. Ioannou, J. Sun, Robust Adaptive Control, Dover, 2013, http://dx.doi.org/10.1007/978-1-4471-5102-9_118-1.

[2] G. Tao, Adaptive Control Design and Analysis, Wiley, 2003, http://dx.doi.org/10.1002/0471459100.

[3] P. Ioannou, B. Fidan, Adaptive Control Tutorial, SIAM, 2006, http://dx.doi.org/10.1137/1.9780898718652.

[4] Y. Rahman, A. Xie, D.S. Bernstein, Retrospective cost adaptive control: Pole placement, frequency response, and connections with LQG control, IEEE Control Syst. Mag. 37 (2017) 28–69, http://dx.doi.org/10.1109/MCS.2017.2718825.

[5] A. Goel, J.A. Paredes, H. Dadhaniya, S.A.U. Islam, A.M. Salim, S. Ravela, D. Bernstein, Experimental implementation of an adaptive digital autopilot, in: Proc. American Control Conference, 2021, pp. 3737–3742, http://dx.doi.org/10.23919/ACC50511.2021.9483005.

[6] J.A. Paredes, D.S. Bernstein, Experimental implementation of retrospective cost adaptive control for suppressing thermoacoustic oscillations in a Rijke tube, IEEE Trans. Control Syst. Technol. 31 (6) (2023) 2484–2498, http://dx.doi.org/10.1109/TCST.2023.3262223.

[7] B.Y. Lai, S.A.U. Islam, S. Nivison, D.S. Bernstein, Data-driven retrospective cost adaptive control of a quadrotor UAV, in: AIAA SCITECH 2023 Forum, 2023, p. 0694, http://dx.doi.org/10.2514/6.2023-0694.

[8] S.A.U. Islam, T.W. Nguyen, I.V. Kolmanovsky, D.S. Bernstein, Data-driven retrospective cost adaptive control for flight control applications, J. Guid. Control Dyn. 44 (10) (2021) 1732–1758, http://dx.doi.org/10.2514/1.G005778.

[9] J.A. Ball, I. Gohberg, M.A. Kaashoek, A frequency response function for linear, time-varying systems, Math. Control Signals Systems 8 (4) (1995) 334–351, http://dx.doi.org/10.1007/BF01209689.

[10] E. Emre, H.-M. Tai, J.H. Seo, Transfer matrices, realization, and control of continuous-time linear time-varying systems via polynomial fractional representations, Linear Algebra Appl. 141 (1990) 79–104, http://dx.doi.org/10.1016/0024-3795(90)90311-Y.

[11] K.S. Tsakalis, P.A. Ioannou, Linear Time-Varying Systems: Control and Adaptation, Prentice-Hall, Inc., 1993.

[12] A. Isidori, A. Ruberti, State-space representation and realization of time-varying linear input-output functions, J. Franklin Inst. B 301 (6) (1976) 573–592, http://dx.doi.org/10.1016/0016-0032(76)90079-X.

[13] R. Ravi, K.M. Nagpal, P.P. Khargonekar, H^∞ Control of linear time-varying systems: A state-space approach, SIAM J. Control Optim. 29 (6) (1991) 1394–1413, http://dx.doi.org/10.1137/0329071.

[14] X. Li, K. Zhou, A time domain approach to robust fault detection of linear time-varying systems, Automatica 45 (1) (2009) 94–102, http://dx.doi.org/10.1016/j.automatica.2008.07.017.

[15] E.W. Kamen, P.P. Khargonekar, K. Poolla, A transfer-function approach to linear time-varying discrete-time systems, SIAM J. Control Optim. 23 (4) (1985) 550–565, http://dx.doi.org/10.1137/0323035.

[16] B.K. Ghosh, P.R. Bouthellier, Simultaneous coefficient assignment of discrete-time multi-input multi-output linear time-varying system: A new approach for compensator design, SIAM J. Control Optim. 31 (6) (1993) 1438–1461, http://dx.doi.org/10.1137/0331067.

[17] P. Dewilde, A.-J. Van der Veen, Time-Varying Systems and Computations, Springer Science & Business Media, 1998, http://dx.doi.org/10.1007/978-1-4757-2817-0.

[18] R. Guidorzi, R. Diversi, Minimal representations of MIMO time-varying systems and realization of cyclostationary models, Automatica 39 (11) (2003) 1903–1914, http://dx.doi.org/10.1016/S0005-1098(03)00195-X.

[19] A. Halanay, V. Ionescu, Time-Varying Discrete Linear Systems: Input-Output Operators. Riccati Equations. Disturbance Attenuation, vol. 68, Birkhäuser, 2012, http://dx.doi.org/10.1007/978-3-0348-8499-0.

[20] A. Feintuch, Realization theory for time-varying, discrete-time linear systems, Adv. Appl. Math. 9 (2) (1988) 211–225, http://dx.doi.org/10.1016/0196-8858(88)90014-0.

[21] B. Nortmann, T. Mylvaganam, Data-driven control of linear time-varying systems, in: Conference on Decision and Control, 2020, pp. 3939–3944, http://dx.doi.org/10.1109/CDC42340.2020.9303845.

[22] S. Sanjeevini, D.S. Bernstein, Decomposition of the retrospective performance variable in adaptive input estimation, in: Proc. American Control Conference, 2022, pp. 242–247, http://dx.doi.org/10.23919/ACC53348.2022.9867833.

[23] K.F. Aljanaideh, D.S. Bernstein, Initial conditions in time-and frequency-domain system identification: Implications of the shift operator versus the z and discrete Fourier transforms, IEEE Control Syst. Mag. 38 (2) (2018) 80–93, http://dx.doi.org/10.1109/MCS.2017.2786419.

[24] M. Witczak, V. Puig, D. Rotondo, P. Witczak, A necessary and sufficient condition for total observability of discrete-time linear time-varying systems, IFAC-PapersOnLine 50 (1) (2017) 729–734, http://dx.doi.org/10.1016/j.ifacol.2017.08.232.

[25] S.A.U. Islam, D.S. Bernstein, Recursive least squares for real-time implementation, IEEE Control Syst. Mag. 39 (3) (2019) 82–85, http://dx.doi.org/10.1109/MCS.2019.2900788.

[26] M. Majji, J.-N. Juang, J.L. Junkins, Time-varying eigensystem realization algorithm, J. Guid. Control Dyn. 33 (1) (2010) 13–28, http://dx.doi.org/10.2514/1.45722.