2016 American Control Conference (ACC)
Boston Marriott Copley Place
July 6-8, 2016. Boston, MA, USA

Retrospective Cost Adaptive Control Using Concurrent Controller Optimization and Target-Model Identification

Frantisek M. Sobolic, Ankit Goel, and Dennis S. Bernstein

Abstract— **Retrospective cost adaptive control (RCAC) is a discrete-time adaptive control algorithm for stabilization, command following, and disturbance rejection. RCAC requires knowledge of the nonminimum-phase (NMP) zeros in the transfer function from the control input to the performance variable. This knowledge is embedded in the target model used to define the retrospective performance variable. Without this knowledge, RCAC has a tendency to cancel unmodeled NMP zeros. The contribution of the present paper is an extension of RCAC that alleviates the need to know the NMP zeros a priori. In particular, concurrent optimization is used to update the coefficients of the controller and target model, thus providing estimates of the unmodeled NMP zeros. Since the retrospective cost is a biquadratic function of these coefficients, an alternating convex search algorithm takes advantage of the closed-form minimizers of both quadratic cost functions. For comparison, the Matlab fminsearch routine is used to jointly optimize the controller and target model. These techniques are illustrated for SISO plants that are asymptotically stable, unstable, minimum phase, and nonminimum phase.**

I. Introduction

Although the overarching motivation for using feedback control versus open-loop control is the ability to overcome uncertainty, feedback control depends on a model of the plant in order to operate reliably and without risking instability. Assuming an exact and complete model, LQG can stabilize all MIMO plants with optimal H_2 performance regardless of plant order, open-loop pole and zero locations, and channel coupling. In practice, however, uncertainty may be unavoidable due to complex, unknown, or unpredictably changing dynamics. To overcome model uncertainty, robust control techniques can be used to guarantee stability and performance, albeit at the expense of performance. Adaptive control can be viewed as a form of robust control, wherein the control law adjusts itself to the plant during operation, thereby overcoming the performance sacrifice inherent in fixed-gain robust control.

A fundamental goal of feedback control is to optimize closed-loop performance in the presence of prior model uncertainty. In the case of adaptive control, closed-loop performance must account for transient performance as the controller adjusts itself to the actual plant. For example, universal adaptive control laws [1] can adapt to uncertainty in the sign of the leading coefficient of the plant transfer function, although the transient response may be impractically large.

Nonminimum-phase (NMP) zeros also present a challenge to adaptive control; for example, the control laws in [2]–[4] assume that the plant is minimum phase. Adaptive control of NMP plants is considered in [5]–[10]. In [8]–[10] knowledge of the NMP zeros is embedded in the target model G_{f}, which is used to filter the past data in order to retrospectively optimize the controller coefficients.

The goal of the present paper is to extend retrospective cost adaptive control (RCAC) as presented in [8]–[10] to alleviate the need for prior modeling of both the sign of the leading coefficient of the plant transfer function as well as its NMP zeros. The key element of this extension is concurrent optimization of the target-model and the controller coefficients. The target-model optimization is a technique for identifying the leading coefficient, NMP zeros, and relative degree of the plant. Consequently, concurrent optimization facilitates the application of RCAC with less prior modeling information than is assumed in [8]–[10]. In particular, the number and location of the NMP zeros need not be known aside from the parity of the number of positive (and thus real) NMP zeros.

Concurrent optimization of the target model and controller is a quadratic optimization problem in the target-model and controller coefficients separately. However, this optimization problem is not convex as a joint function of both sets of variables, and therefore nonconvex optimization methods are needed. In the present paper, we address this problem in two different ways. First we take advantage of the biquadratic structure of the cost function by applying an alternating convex search algorithm [11]. Related techniques for biconvex and bilinear optimization are given in [12]–[15]. For comparison, the Matlab fminsearch routine is used to jointly optimize the controller and target-model coefficients.

II. PROBLEM FORMULATION

Consider the SISO discrete-time system

$$x(k+1) = Ax(k) + Bu(k) + D_1 w(k), \tag{1}$$
$$y(k) = Cx(k) + D_2 w(k), \tag{2}$$
$$z(k) = E_1 x(k) + E_0 w(k), \tag{3}$$

where $x(k) \in \mathbb{R}^n$ is the state, $y(k) \in \mathbb{R}$ is the measurement, $u(k) \in \mathbb{R}$ is the control input, $w(k) \in \mathbb{R}$ is the exogenous input, and $z(k) \in \mathbb{R}$ is the measured performance variable. The goal is to develop an adaptive output feedback controller that minimizes z in the presence of the exogenous signal

F. Sobolic and A. Goel are graduate students with the Aerospace Engineering Department at the University of Michigan, 1320 Beal Ave., Ann Arbor, MI 48109, USA fsobolic@umich.edu, ankgoel@umich.edu

D. S. Bernstein is a Professor with the Aerospace Engineering Department, University of Michigan, dsbaero@umich.edu

978-1-4673-8681-4/$31.00 ©2016 AACC

w with limited modeling information about (1)–(3). The components of w can represent either command signals to be followed, external disturbances to be rejected, or both, depending on the choice of D_1 and E_0. Depending on the application, components of w may or may not be measured. These components are included in y by suitable choice of C and D_2. No assumptions are made concerning the state space realization since RCAC requires only input-output model information.

III. RCAC ALGORITHM

A. Controller Structure

The adaptive control algorithm is constructed as a strictly proper time-series dynamic compensator of order n_c, such that the control $u(k)$ is given as

$$u(k)=\sum_{i=1}^{n_c}P_i(k)u(k-i)+\sum_{i=1}^{n_c}Q_i(k)z(k-i), \tag{4}$$

where $P_i(k),Q_i(k)\in\mathbb{R}$ are the controller coefficients. In terms of the forward shift operator $\mathbf{q}$, the transfer function of the controller from z to u is given by

$$G_{\rm c}(\mathbf{q})=\left(\mathbf{q}^{n_{\rm c}}-\mathbf{q}^{n_{\rm c}-1}P_1(k)-\cdots-P_{n_{\rm c}}(k)\right)^{-1}\cdot\left(\mathbf{q}^{n_{\rm c}-1}Q_1(k)+\cdots+Q_{n_{\rm c}}(k)\right). \tag{5}$$

In the present paper we focus on SISO controllers, and hence $G_{\rm c}$ can be written as

$$G_{\rm c}(\mathbf{q})=\frac{Q_1(k)\mathbf{q}^{n_{\rm c}-1}+\cdots+Q_{n_{\rm c}}(k)}{\mathbf{q}^{n_{\rm c}}-P_1(k)\mathbf{q}^{n_{\rm c}-1}-\cdots-P_{n_{\rm c}}(k)}. \tag{6}$$

Note that (5) is an infinite impulse response (IIR) controller. The controller (4) can be expressed as

$$u(k)=\phi(k)\theta(k), \tag{7}$$

where the regressor matrix $\phi(k)$ and controller coefficient matrix $\theta(k)$ is defined as

$$\phi(k)^{\rm T}\triangleq\begin{bmatrix}u(k-1)\\ \vdots\\ u(k-n_{\rm c})\\ z(k-1)\\ \vdots\\ z(k-n_{\rm c})\end{bmatrix}\in\mathbb{R}^{l_\theta},$$

$$\theta(k)\triangleq\left[\ P_1(k)\cdots P_{n_{\rm c}}(k)\ Q_1(k)\cdots Q_{n_{\rm c}}(k)\ \right]^{\rm T}\in\mathbb{R}^{l_\theta},$$

where $l_\theta\triangleq 2n_{\rm c}$.

B. Retrospective Performance

The *retrospective control* is defined as

$$\hat{u}(k,\hat{\theta})\triangleq\phi(k)\hat{\theta}, \tag{8}$$

where $\hat{\theta}\in\mathbb{R}^{l_\theta}$ is determined by optimizing the cost function defined in Section III-C. The corresponding *retrospective performance variable* is defined as

$$\hat{z}(k,\hat{\theta})\triangleq z(k)+G_{\rm f}(\mathbf{q})\left[\hat{u}(k,\hat{\theta})-u(k)\right], \tag{9}$$

where the target model $G_{\rm f}(\mathbf{q})$ is a finite impulse response (FIR) filter written as

$$G_{\rm f}(\mathbf{q})=\hat{N}_{n_{\rm f}}\mathbf{q}^{n_{\rm f}-1}+\cdots+\hat{N}_1, \tag{10}$$

where the target-model order $n_{\rm f}\geq 1$, and $\hat{N}_i\in\mathbb{R}$ for all $1\leq i\leq n_{\rm f}$. Next, we define $Z(k)\in\mathbb{R}^{p_{\rm c}}$, $U(k)\in\mathbb{R}^{p}$, and $\Phi(k)\in\mathbb{R}^{p\times l_\theta}$ by

$$Z(k)\triangleq\begin{bmatrix}z(k)\\ \vdots\\ z(k-p_{\rm c}+1)\end{bmatrix},\ U(k)\triangleq\begin{bmatrix}u(k-1)\\ \vdots\\ u(k-p)\end{bmatrix},$$

$$\Phi(k)\triangleq\begin{bmatrix}\phi(k-1)\\ \vdots\\ \phi(k-p)\end{bmatrix},$$

where $p\triangleq p_{\rm c}+n_{\rm f}-1$, and $p_{\rm c}\geq 1$ is the least squares window size.

Furthermore, we define the *extended retrospective performance vector*

$$\hat{Z}(k,\hat{\theta},\hat{N})=Z(k)+\bar{N}(\Phi(k)\hat{\theta}-U(k)), \tag{11}$$

where $\hat{Z}(k,\hat{\theta},\hat{N})\in\mathbb{R}^{p_{\rm c}}$ and the matrix $\bar{N}\in\mathbb{R}^{p_{\rm c}\times p}$ is the block-Toeplitz representation of the FIR target model given by

$$\bar{N}=\begin{bmatrix}\hat{N}^T & 0 & 0\\ \vdots & \ddots & 0\\ 0 & 0 & \hat{N}^T\end{bmatrix}, \tag{12}$$

where

$$\hat{N}\triangleq\begin{bmatrix}\hat{N}_{n_{\rm f}}\\ \vdots\\ \hat{N}_1\end{bmatrix}. \tag{13}$$

By defining $v(k)\triangleq\phi(k)\hat{\theta}-u(k)$, (11) can be written as

$$\hat{Z}(k,\hat{\theta},\hat{N})=Z(k)+V(k)^{\rm T}\hat{N}(k), \tag{14}$$

where

$$V(k)=\begin{bmatrix}v(k-1) & \cdots & v(k-p_{\rm c})\\ \vdots & \vdots & \vdots\\ v(k-n_{\rm f}) & \cdots & v(k-p)\end{bmatrix}\in\mathbb{R}^{n_{\rm f}\times p_{\rm c}}.$$

The target model $G_{\rm f}$ sets target locations for the closed-loop poles [10]. Since we define $G_{\rm f}$ as an FIR target model, the closed-loop pole locations are attracted to the origin. Also, in [10], the ideal $G_{\rm f}$ is chosen to match the NMP zeros of the plant. This feature ensures that the NMP zeros of the plant are not canceled by the poles of $G_{\rm c}$. In the present paper, we concurrently optimize the coefficients of the controller and the target model. Optimization of the target model identifies the plant NMP zeros, which reduces the need for prior modeling information.

C. Retrospective Cost

Consider the retrospective cost function

$$J(k,\hat{\theta},\hat{N}) \triangleq \hat{Z}(k,\hat{\theta},\hat{N})^T R_z \hat{Z}(k,\hat{\theta},\hat{N}) + (\hat{\theta}(k)-\theta(k-1))^T R_\delta (\hat{\theta}(k)-\theta(k-1)), \quad (15)$$

where the positive-definite matrices $R_z \in \mathbb{R}^{p_c \times p_c}$ and $R_\delta \in \mathbb{R}^{l_\theta \times l_\theta}$ are the performance and learning-rate weightings, respectively.

IV. Biquadratic Optimization

We use the Matlab fminsearch function for nonlinear function optimization, as well as the alternating convex search (ACS) algorithm [11]. In ACS, only the variables that are active are optimized while the remaining variables are kept fixed. Although ACS converges to a stationary point of the cost function, it is shown in [11] that global convergence is not guaranteed.

A. Alternating Convex Search Minimizers

1) Minimizing $\hat{\theta}$ for fixed $\hat{N}$: For fixed $\hat{N}$, substituting (11) into (15) yields

$$J(k,\hat{\theta},\hat{N}) = \hat{\theta}^T(k)A_\theta(k)\hat{\theta}(k) + 2\hat{\theta}^T(k)b_\theta(k) + c_\theta(k), \quad (16)$$

where

$$\begin{aligned}
A_\theta(k) &\triangleq \Phi_{\rm f}(k)^T R_z \Phi_{\rm f}(k) + R_\delta,\\
b_\theta(k) &\triangleq \Phi_{\rm f}(k)^T R_z (Z(k)-U_{\rm f}(k)) - R_\delta \theta(k-1),\\
c_\theta(k) &\triangleq (Z(k)-U_{\rm f}(k))^T R_z (Z(k)-U_{\rm f}(k))\\
&\quad + \theta(k-1)^T R_\delta \theta(k-1),\\
U_{\rm f}(k) &\triangleq \bar{N}U(k), \qquad \Phi_{\rm f}(k) \triangleq \bar{N}\Phi(k).
\end{aligned}$$

Since $A_\theta(k)$ is positive definite, (16) has the unique global minimizer $\hat{\theta} = \hat{\theta}(k)$, where

$$\hat{\theta}(k) \triangleq -A_\theta(k)^{-1} b_\theta(k). \quad (17)$$

2) Minimizing $\hat{N}$ for fixed $\hat{\theta}$: For fixed $\hat{\theta}$, substituting (14) into (15) yields

$$J(k,\hat{\theta},\hat{N}) = \hat{N}^{\rm T}(k)A_N(k)\hat{N}(k) + 2\hat{N}(k)^{\rm T} b_N(k) + c_N(k), \quad (18)$$

where

$$\begin{aligned}
A_N(k) &\triangleq V(k)R_z V(k)^{\rm T},\\
b_N(k) &\triangleq V(k)R_z Z(k),\\
c_N(k) &\triangleq Z(k)^{\rm T} R_z Z(k) + \Delta\theta(k)^{\rm T} R_\delta \Delta\theta(k),\\
\Delta\theta(k) &\triangleq \theta(k) - \theta(k-1). \quad (19)
\end{aligned}$$

If $A_N(k)$ is positive definite, then (18) has the unique global minimizer $\hat{N} = \hat{N}(k)$, where

$$\hat{N}(k) \triangleq -A_N(k)^{-1} b_N(k). \quad (20)$$

B. ACS

ACS consists of the following steps:

Step 1 Choose a nonzero starting point $\hat{N}(k) \in \mathbb{R}^{n_{\rm f}}$ for $k \geq p+1$.

Step 2 For fixed $\hat{N}(k)$, solve for $\hat{\theta}(k)$ using (17) and the associated cost using (16).

Step 3 For fixed $\hat{\theta}(k)$ found in Step 2, solve for $\hat{N}(k)$ using (20) and the associated cost using (18).

Step 4 Determine whether Step 2 or Step 3 produces the lower cost. If a stopping criteria is satisfied, set $\hat{N}(k+1) = \hat{N}(k)$ and $\hat{\theta}(k+1) = \hat{\theta}(k)$ for the next ACS nonzero starting point and increment k. Otherwise, set $\hat{N}(k)$ and $\hat{\theta}(k)$ to the corresponding step that produced the lower cost and go back to Step 2.

For the examples in Section V, the stopping criterion consists of 800 evaluations and a function tolerance of 1×10^{-4}.

C. fminsearch

The method used with the Matlab fminsearch algorithm consists of the following steps:

Step 1 Choose a starting point $(\hat{N}(k), \hat{\theta}(k)) \in \mathbb{R}^{n_{\rm f}+l_\theta}$ for $k = p+1$.

Step 2 At the current time step, minimize the cost function (15) until a stopping criteria is satisfied.

Step 3 Set $\hat{N}(k+1) = \hat{N}(k)$ and $\hat{\theta}(k+1) = \hat{\theta}(k)$ for the next starting point, increment k, and go to Step 2.

For the examples in Section V, the stopping criterion consists of 5000 function evaluations and iterations along with a function and variable tolerance of 1×10^{-4}.

V. Numerical Examples

In this section, we illustrate the concurrent optimization technique for the adaptive command-following problem shown in Fig. 1. The exogenous signal w is the command r, and $z = y - r$. This problem is a special case of (1)-(3) with $C = E_1$, $D_2 = 0$, and $E_0 = -1$. Hence $G(\mathbf{q}) = C(\mathbf{q}I - A)^{-1}B$.

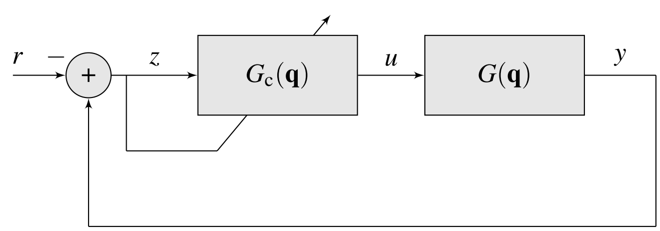

Fig. 1: The adaptive command-following problem.

In each of the examples in Section V all of the target-model coefficients are initialized at 0 except for $\hat{N}_1$. In particular, $\hat{N}_1$ is initialized based on the parity of the positive (and thus real) NMP zeros, that is, for all $0 \leq k \leq p+1$,

$$\hat{N}_1(k) = (-1)^h, \quad (21)$$

where h is the number of positive NMP zeros in G. For all of the examples below, for all $k < p+1$, the controller coefficients for both algorithms are $0_{l_\theta \times 1}$, and at $k = p+1$, the controller coefficients are initialized as $1_{l_\theta \times 1}$ and $0_{l_\theta \times 1}$ for ACS and fminsearch, respectively.

Example V.1: *Asymptotically stable, minimum-phase plant.* Consider the asymptotically stable, minimum-phase plant

$$G(\mathbf{q}) = \frac{2(\mathbf{q}-0.2)}{(\mathbf{q}-0.3)(\mathbf{q}-0.6)}, \tag{22}$$

and let r be a unit-height step command. We choose $n_\mathrm{f} = 2$ and note that, since G has no positive NMP zeros, $h = 0$ and thus, by (21), the target-model coefficients in $\hat{N}(k)$ are initialized as $[0 \ \ 1]^\mathrm{T}$ for all $0 \le k \le p+1$. Let the controller order $n_\mathrm{c} = 2$, the least squares window size $p_\mathrm{c} = 20$, $R_\delta = 1 \times 10^{-3} I_{l_\theta}$, $R_z = I_{p_\mathrm{c}}$, and $x(0) = [1 \ \ 1]^\mathrm{T}$. Fig. 2 shows the results of the concurrent optimization. Both algorithms give comparable results in the converged target-model and controller gain coefficients. Note that the target-model coefficient $\hat{N}_2$ obtained by both algorithms converges to the value 2 of the leading numerator coefficient of the plant, while the controller converges to an integrator internal model in order to follow the step command. ■

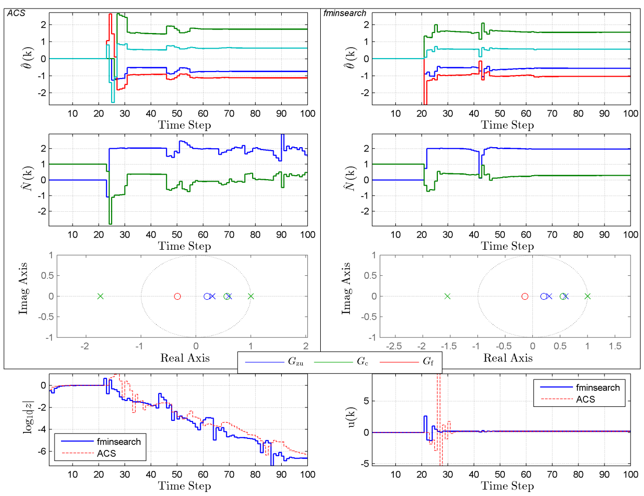

Fig. 2: **Example V.1:** *Asymptotically stable, minimum-phase system*. Concurrent optimization is applied to step-command following for the asymptotically stable minimum-phase plant (22). The three upper left figures show the result of using ACS, while the three upper right plots show the result of using fminsearch. Note that the target-model coefficient $\hat{N}_2$ converges to 2, the leading numerator coefficient of the plant, and the controller converges to an integrator internal model in order to follow the step command

Example V.2: *Unstable, minimum-phase plant.* Consider the unstable, minimum-phase plant

$$G(\mathbf{q}) = \frac{\mathbf{q}-0.2}{(\mathbf{q}-0.6)(\mathbf{q}-1.3)}, \tag{23}$$

and let r be a unit-height step command. We choose $n_\mathrm{f} = 4$ and note that, since G has no positive NMP zeros, $h = 0$ and thus, by (21), the target-model coefficients in $\hat{N}(k)$ are initialized as $[0 \ 0 \ 0 \ 1]^\mathrm{T}$ for all $0 \le k \le p+1$. Let the controller order $n_\mathrm{c} = 4$, and the least squares window size $p_\mathrm{c} = 20$, $R_\delta = 1 \times 10^{-1} I_{l_\theta}$, $R_z = I_{p_\mathrm{c}}$, and $x(0) = [0 \ 0]^\mathrm{T}$. Fig. 3 shows the results of the concurrent optimization. Note that both algorithms converge to a controller with an integrator internal model in order to follow the step command and that the target-model coefficient $\hat{N}_4$ converges to 1, the leading numerator coefficient of the plant. ■

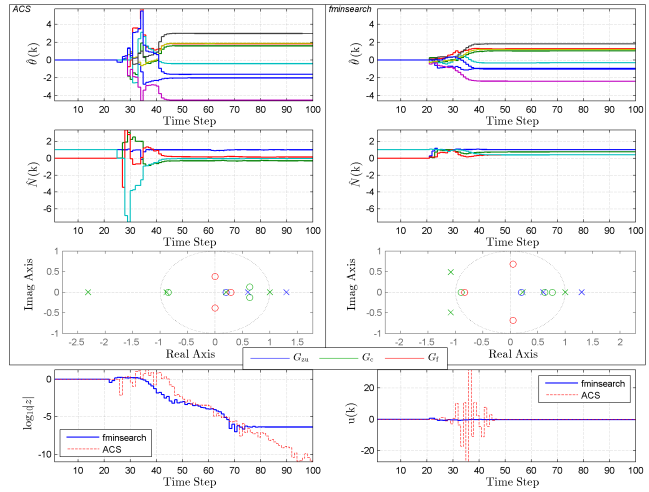

Fig. 3: Example V.2: Unstable, minimum-phase plant. Concurrent optimization is applied to step-command following for the unstable minimum-phase plant (23). Note that the controller stabilizes the unstable plant and develops an integrator internal model in order to follow the step command.

Example V.3: *Asymptotically stable, NMP plant.* Consider the asymptotically stable, NMP plant

$$G(\mathbf{q}) = \frac{\mathbf{q}-1.1}{(\mathbf{q}-0.3)(\mathbf{q}-0.6)}, \tag{24}$$

and let r be a unit-height step command. We choose $n_\mathrm{f} = 4$ and note that, since G has one positive NMP zero, $h = 1$ and thus, by (21), the target-model coefficients in $\hat{N}(k)$ are initialized as $[0 \ 0 \ 0 \ \ -1]^\mathrm{T}$ for all $0 \le k \le p+1$. Let the controller order $n_\mathrm{c} = 4$, and the least squares window size $p_\mathrm{c} = 30$, $R_\delta = 1 \times 10^{-3} I_{l_\theta}$, $R_z = I_{p_\mathrm{c}}$, and $x(0) = [0 \ 0]^\mathrm{T}$. Fig. 4 shows the results of the concurrent optimization. Note that the location of the NMP zero location is unknown. Fig. 5 shows the poles and zeros of (24), the controller, and the target model. Both algorithms converge to a target model G_f that has a zero at the location of the plant zero. Also,

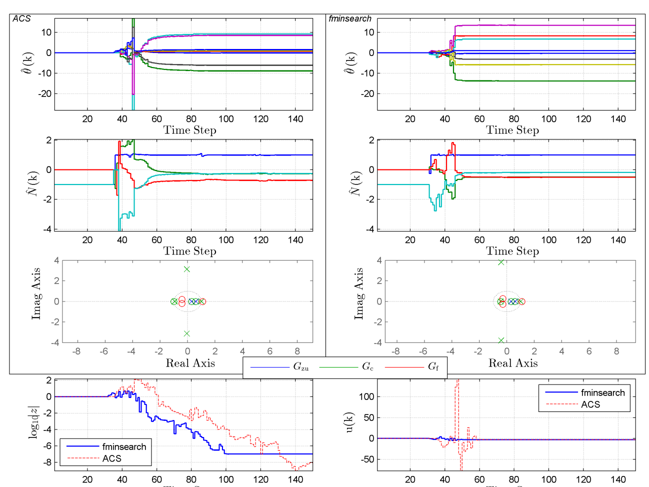

Fig. 4: Example V.3: Asymptotically stable, NMP plant. Concurrent optimization is applied to step-command following for the asymptotically stable, NMP plant (24), where the location of the NMP zero is unknown. Note that the controller develops an integrator internal model in order to follow the step command, and the target model captures the NMP zero location. Figure 5 shows a zoomed in view of the plant, controller, and target model pole/zero locations after convergence.

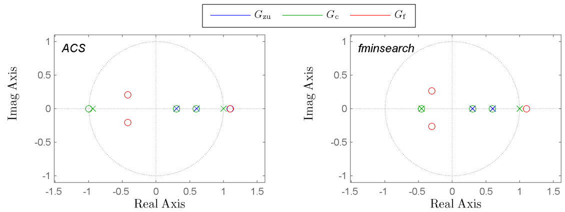

Fig. 5: Example V.3: Asymptotically stable, NMP plant. This figure shows a zoomed in pole/zero map of the plant, converged controller, and converged target model for ACS [left] and fminsearch [right]. Note that the target model captures the NMP zero location of the plant and that the controller converges to an integrator internal model in order to follow the step command.

both algorithms converge to a controller with an integrator internal model in order to follow the step command. ■

Example V.4: *Asymptotically stable, NMP plant with a negative NMP zero.* Consider the asymptotically stable, NMP plant with a negative NMP zero

$$G(\mathbf{q}) = \frac{\mathbf{q}+1.1}{(\mathbf{q}-0.3)(\mathbf{q}-0.6)}, \tag{25}$$

and let r be a unit-height step command. We choose $n_{\mathrm{f}} = 4$ and note that, since G has no positive NMP zeros, $h = 0$ and thus, by (21), the target-model coefficients in $\hat{N}(k)$ are initialized as $[0\ \ 0\ \ 0\ \ 1]^{\mathrm{T}}$ for all $0 \leq k \leq p+1$. Let the controller order $n_{\mathrm{c}} = 4$, and the least squares window size $p_{\mathrm{c}} = 50$, $R_\delta = 1\times 10^{-1} I_{l_\theta}$, $R_z = I_{p_{\mathrm{c}}}$, and $x(0) = [1\ 1]^{\mathrm{T}}$. Fig. 6 shows the results of the concurrent optimization. Both

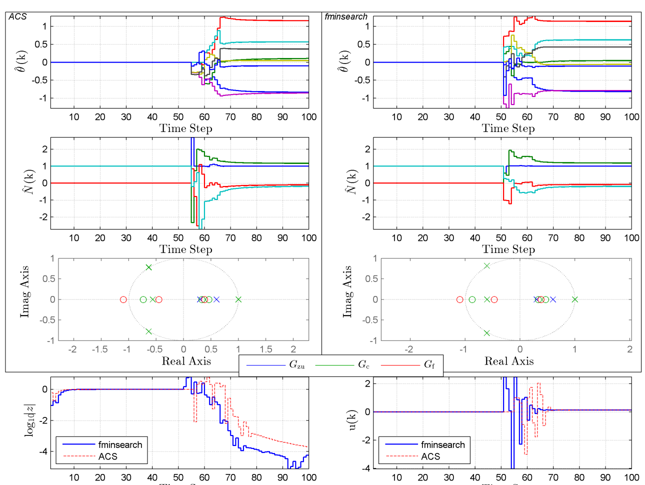

Fig. 6: Example V.4: Asymptotically stable plant with a negative NMP zero. Concurrent optimization is applied to a step-command following problem for the asymptotically stable plant with a negative NMP zero (25). The location of the NMP zero is unknown. The controller converges to an integrator internal model to follow the step command, and the target model captures the location of the negative NMP zero.

algorithms give comparable results in the converged target-model and controller gain coefficients. Note that the target-model coefficient $\hat{N}_4$ for both algorithms converges to the value 1 of the leading numerator coefficient of the plant, while the controller converges to an integrator internal model in order to follow the step command. Note that the location of the NMP zero is unknown to both algorithms and the zero of the converged target model converges to the NMP zero of (25). ■

Example V.5: *Asymptotically stable, NMP plant with relative degree 2.* Consider the asymptotically stable, NMP plant

$$G(\mathbf{q}) = \frac{\mathbf{q}-1.2}{(\mathbf{q}-0.1)(\mathbf{q}-0.3)(\mathbf{q}-0.6)}, \tag{26}$$

and let r be a unit-height step command. We choose $n_{\mathrm{f}} = 5$ and note that, since G has one positive NMP zero, $h = 1$ and thus, by (21), the target-model coefficients in $\hat{N}(k)$ are initialized as $[0\ 0\ 0\ 0\ \ -1]^{\mathrm{T}}$ for all $0 \leq k \leq p+1$. Let the controller order $n_{\mathrm{c}} = 6$, and the least squares window size $p_{\mathrm{c}} = 40$, $R_\delta = 1\times 10^{-2} I_{l_\theta}$, $R_z = I_{p_{\mathrm{c}}}$, and $x(0) = [1\ 1\ 1]^{\mathrm{T}}$. Fig. 7 shows the results of the concurrent optimization. Note that

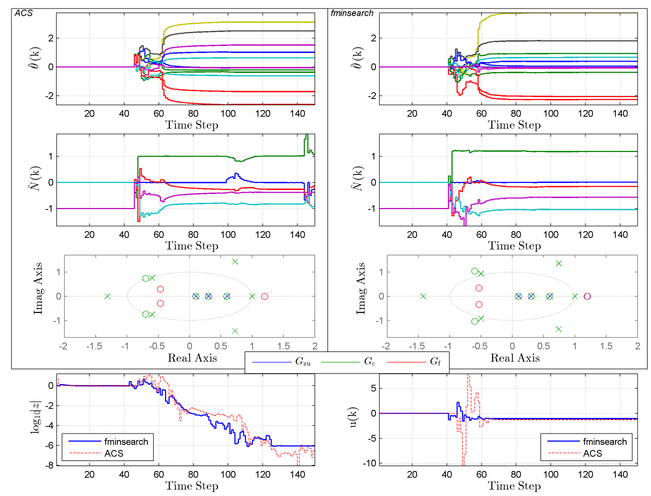

Fig. 7: Example V.5: Asymptotically stable NMP plant with relative degree 2. Concurrent optimization is applied to a step-command following problem for the asymptotically stable, NMP plant with relative degree 2 (26). Note that the location of the NMP zero is unknown. The controller converges to an integrator internal model in order to follow the step command, and the converged target model captures the location of the NMP zero.

the location of the NMP zero is unknown to both algorithms and that the zero of the target model converges to the NMP zero of (26). Also note that, since the relative degree of (26) is 2, the target-model coefficient $\hat{N}_5$ for both algorithms converges to 0, while the target model coefficient $\hat{N}_4$ converges to the value 1 of the leading numerator coefficient of the plant. Both algorithms produce a spurious target model zero of high magnitude, which is not included in Figure 7. The controller also converges to an integrator internal model in order to follow the step command. ■

Example V.6: *Asymptotically stable, minimum-phase plant.* Consider the asymptotically stable, minimum-phase plant

$$G(\mathbf{q}) = \frac{\mathbf{q}-0.3}{(\mathbf{q}-0.2)(\mathbf{q}-0.6)}, \tag{27}$$

and let r be a unit-amplitude harmonic command with frequency $\omega = \frac{2\pi}{25}$ rad/sample. We choose $n_{\mathrm{f}} = 3$ and note that, since G has no positive NMP zeros, $h = 0$ and thus, by (21), the target-model coefficients in $\hat{N}(k)$ are initialized as $[0\ 0\ 1]^{\mathrm{T}}$ for all $0 \leq k \leq p+1$. Let the controller order $n_{\mathrm{c}} = 4$, and the least squares window size $p_{\mathrm{c}} = 20$, $R_\delta = 1\times 10^{-3} I_{l_\theta}$, $R_z = I_{p_{\mathrm{c}}}$, and $x(0) = [0\ 0]^{\mathrm{T}}$. Fig. 8 shows the results of the

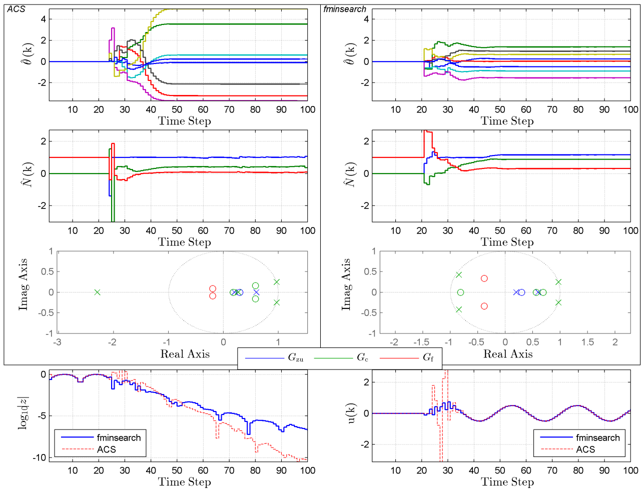

Fig. 8: Example V.6: Asymptotically stable, minimum-phase plant with harmonic-command following. Concurrent optimization is applied to a harmonic-command following problem for the asymptotically stable, minimum-phase plant (27). Note that both algorithms converge to an internal model controller with poles at the command frequency on the unit circle and the target-model coefficient $\hat{N}_3$ converges to 1, the leading numerator coefficient of the plant.

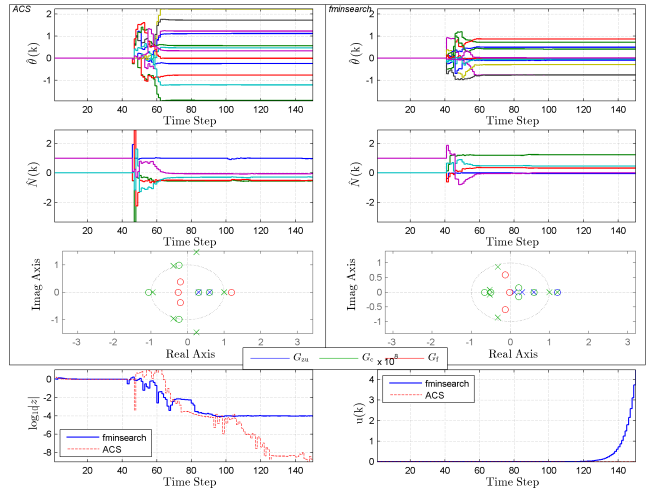

Fig. 9: Example V.7: Asymptotically stable NMP plant with relative degree 2. Concurrent optimization is applied to a step-command following problem for the asymptotically stable, NMP plant (26) with relative degree 2. The location of the NMP zero is unknown. For both algorithms, the controller converges to an integrator internal model in order to follow the step command. For ACS, the asymptotic target model captures the location of the NMP zero, whereas fminsearch does not capture the NMP zero, leading to cancellation of an unstable controller pole and a NMP plant zero.

concurrent optimization. Note that the controller converges to a harmonic internal model in order to follow the same frequency of the command r and that the target-model coefficient $\hat{N}_3$ converges to the leading numerator coefficient 1 of the plant. ■

Example V.7: *Asymptotically stable, NMP plant with relative degree 2.* Consider the asymptotically stable, minimum-phase plant given in (26), and let r be a unit-height step command. We choose $n_{\rm f} = 5$ and note that, since G has one NMP zero, $h = 1$ and thus, by (21), the target-model coefficients in $\hat{N}(k)$ would be initialized as $[0\ 0\ 0\ 0\ -1]^{\rm T}$. In this example, we violate (21) by choosing the opposite sign of $\hat{N}_1(k)$ and let the target-model coefficients in $\hat{N}(k)$ be initialized as $[0\ 0\ 0\ 0\ 1]^{\rm T}$ for $k \le p+1$. Fig. 9 shows the results of the concurrent optimization. Note that the location of the NMP zero is unknown to both algorithms and that, for ACS, the asymptotic target model captures the location of the NMP zero. In contrast, fminsearch does not capture the NMP zero, which leads to cancellation between an unstable controller pole and a NMP plant zero. ■

VI. Conclusions

The cost function associated with retrospective cost adaptive control (RCAC) was used for concurrent optimization of the controller coefficients and identification of the target model. This function has a biquadratic structure. A modified version of the alternating convex search algorithm and the Matlab fminsearch algorithm demonstrated the ability to concurrently optimize these coefficients for an adaptive command-following problem. The novel element of the present paper is adaptive control without prior knowledge of the plant transfer function, including NMP zero locations. Future work will focus on improving the computational efficiency and accuracy of the concurrent optimization along with guarantees of global convergence.

References

[1] A. Ilchmann, *Non-Identifier-Based High-gain Adaptive Control*. Springer-Verlag New York, Inc., 1993.

[2] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Courier Corporation, 2012.

[3] K. J. Åström and B. Wittenmark, *Adaptive Control*. Courier Corporation, 2013.

[4] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, *Nonlinear and Adaptive Control Design*. John Wiley & Sons, Inc., 1995.

[5] K. J. Åström, "Direct methods for nonminimum phase systems," in *Proc. Conf. Dec. Contr., Albuquerque, NM*, no. 19, pp. 611–615, 1980.

[6] G. C. Goodwin and K. Sin, "Adaptive control of nonminimum phase systems," *IEEE Trans. Autom. Contr.*, vol. 26.

[7] Y. Rahman, K. F. Aljanaideh, E. D. Sumer, and D. S. Bernstein, "Adaptive control of aircraft lateral motion with an unknown transition to nonmimimum-phase dynamics," in *Amer. Contr. Conf., 2014*, pp. 2359–2364, IEEE, 2014.

[8] M. A. Santillo and D. S. Bernstein, "Adaptive control based on retrospective cost optimization," *AIAA J. Guid. Contr. Dyn.*, vol. 33, no. 2, pp. 289–304, 2010.

[9] J. B. Hoagg and D. S. Bernstein, "Retrospective cost model reference adaptive control for nonminimum-phase systems," *AIAA J. Guid. Contr. Dyn.*, vol. 35, no. 6, pp. 1767–1786, 2012.

[10] Y. Rahman, A. Xie, J. B. Hoagg, and D. S. Bernstein, "A tutorial and overview of retrospective-cost-adaptive-control," in *Proc. Amer. Contr. Conf., Boston, MA*, July 2016.

[11] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex sets and optimization with biconvex functions: a survey and extensions," *Mathematical Methods of Operations Research*, vol. 66, no. 3, pp. 373–407, 2007.

[12] C. Ling, J. Nie, L. Qi, and Y. Ye, "Biquadratic optimization over unit spheres and semidefinite programming relaxations," *SIAM J. Opt.*, vol. 20, no. 3, pp. 1286–1310, 2009.

[13] J. G. VanAntwerp and R. D. Braatz, "A tutorial on linear and bilinear matrix inequalities," *J. Proc. Contr.*, vol. 10, no. 4, pp. 363–385, 2000.

[14] H. D. Tuan and P. Apkarian, "Low nonconvexity-rank bilinear matrix inequalities: algorithms and applications in robust controller and structure designs," *IEEE Trans, Autom Contr*, vol. 45, no. 11, pp. 2111–2117, 2000.

[15] K.-C. Goh, M. Safonov, and J. Ly, "Robust synthesis via bilinear matrix inequalities," *Int. J. Robust Nonlinear Contr.*, vol. 6, no. 9-10, pp. 1079–1095, 1996.