2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Retrospective Cost Adaptive Control for Nonminimum-Phase Systems with Uncertain Nonminimum-Phase Zeros Using Convex Optimization

Alexey V. Morozov[1], Anthony M. D'Amato[1], Jesse B. Hoagg[2], and Dennis S. Bernstein[3]

Abstract— **Retrospective cost adaptive control (RCAC) can be applied to command following and disturbance rejection problems with plants that are possibly MIMO, unstable, and nonminimum phase. RCAC requires knowledge of a bound on the first nonzero Markov parameter as well as knowledge of the nonminimum-phase zeros of the plant, if any. The goal of the present paper is to increase the robustness of RCAC to uncertainty in the locations of the nonminimum-phase zeros. Specifically, a convex constraint is imposed on the poles of the controller in order to prevent the adaptive controller from attempting to cancel the nonminimum-phase zeros. Numerical results show that, when constrained convex optimization is used at each step, the transient response is improved and the adaptive controller has increased robustness to uncertainty in the locations of the nonminimum-phase zeros.**

I. Introduction

Nonminimum-phase zeros are a major impediment to achievable performance in feedback control. While all fields of science and technology that work with dynamical systems are familiar with poles, the field of control is unique in recognizing the role of zeros in systems with inputs and outputs. The multiple ways in which zeros impact the operation of control systems are discussed in [1]. In the classical case of root locus analysis, the attraction of poles to zeros limits the magnitude of the feedback gain. The same phenomenon occurs in LQG control for SISO and MIMO systems, where, in the high-authority limit, the controller obtained from the Riccati equations drives some of the closed-loop poles to the open-left-half-plane reflections of the open-loop nonminimum-phase (NMP) zeros. Nonminimum-phase zeros thus limit the achievable bandwidth and control authority. In addition, real nonminimum-phase zeros are responsible for initial undershoot and direction reversals in the step response. These issues are well understood for continuous-time systems; for sampled-data systems, analogous phenomena occur.

Nonminimum-phase zeros are especially challenging in adaptive control since an adaptive controller may attempt to cancel a nonminimum-phase zero. Such pole-zero cancellation is impossible, and thus, in fixed-gain control, the use of a nonminimum-phase zero to cancel an unstable pole is well known to be ineffective. In adaptive control, the reverse situation occurs, namely, an adaptive controller may attempt to cancel a nonminimum-phase zero that limits its performance; however, the nonminimum-phase zero cannot be removed. The detrimental affect of this attempt to cancel a nonminimum-phase zero is the fact that the controller may seek to apply unbounded control effort, thereby destabilizing the closed-loop system.

In the present paper we focus on retrospective cost adaptive control (RCAC) [2], [3]. RCAC uses a retrospective cost functional defined in terms of a surrogate performance variable, which is based on measured data over a past window of operation. In effect, the retrospective cost functional "looks backward" over the window of data in order to determine a controller modification that would have improved the past performance. This retrospective cost functional is optimized at each step in order to update the controller. The algorithm thus seeks the controller that achieves the best performance in terms of a prior window of operation as determined by the retrospective cost. The approach of "looking backward" rather than forward (as in the case, for example, of model predictive control) allows RCAC to control the system under minimal modeling information.

As shown in [2], [4], RCAC has the ability to adaptively control nonminimum-phase systems if the locations of the nonminimum-phase zeros are known. Modeling information that captures the locations of the nonminimum-phase zeros (either SISO or MIMO) is included in the matrix $\bar{B}_{zu}$, as described in the next section. The matrix $\bar{B}_{zu}$ can be defined in terms of the Markov parameters of the transfer function from the control input to the performance variable. The Markov parameters are coefficients of the Laurent expansion of the transfer function expressed in terms of powers of $1/z$. The Laurent expansion provides a convergent series for the transfer function outside of the spectral radius of the plant; consequently, this series automatically captures all nonminimum-phase zeros outside of the spectral radius. Alternatively, if the nonminimum-phase zeros are known, then their values can be used directly in $\bar{B}_{zu}$ in place of a finite number of Markov parameters. Consequently, identification of nonminimum-phase zeros is of interest in practice [5].

The above discussion leads to the following question: How does RCAC exploit knowledge of the nonminimum-phase zeros in order to avoid any attempt to cancel them? This question is addressed in [4], where it is shown that the nonminimum-phase zeros appear in the numerator of a filter that processes the data in the regressor used in the controller update. This filter removes spectral content corresponding to the locations of the nonminimum-phase zeros, thus avoiding the possibility of having the adaptive controller attempt to cancel a nonminimum-phase zero.

[1]Graduate Student, Department of Aerospace Engineering, University of Michigan, email: morozova@umich.edu, amdamato@umich.edu

[2]Assistant Professor, Department of Mechanical Engineering, University of Kentucky, email: jhoagg@engr.uky.edu

[3]Professor, Department of Aerospace Engineering, University of Michigan, email: dsbaero@umich.edu

978-1-4577-0079-8/11/$26.00 ©2011 AACC

Since RCAC requires knowledge of the nonminimum-phase zeros of the plant, it is of interest to determine how accurately this knowledge must be known. Numerical examples in [2], [6] suggest a negative result, namely, that there may exist plants for which the robustness to uncertainty in the locations of the nonminimum-phase zeros may be arbitrarily small. This lack of robustness is manifested by the increasingly larger transients that arise as the nonminimum-phase-zero locations become increasingly uncertain. This negative result is consistent with [7], [8], namely, that adaptive control must confront plants that are inherently difficult to control. RCAC shows that this difficulty is inherent in the modeling information relating specifically to the nonminimum-phase zeros (if any are present).

In the present paper our goal is to develop a technique that increases the robustness of RCAC to uncertainty in the locations of the nonminimum-phase zeros. To do this, we consider an extension of RCAC, where the minimization of the retrospective cost is performed subject to a constraint on the allowable locations of the controller poles. A convex constraint on eigenvalue locations is given in [9] and is used in [10] for model identification with guaranteed stability. However, this approach cannot be used with RCAC since RCAC updates the coefficients of the denominator of the controller transfer function, rather than the entries of an unstructured dynamics matrix. We thus use this polynomial to construct a companion matrix. Since a bound on the spectral radius of the companion matrix does not provide a convex constraint on the coefficients of the polynomial, we bound the spectral radius with a matrix norm, which defines a convex constraint. Although bounding the spectral radius with a matrix norm introduces conservatism, this conservatism has minimal effect since the magnitude of the bound on the matrix norm can be adjusted as a design parameter.

II. Problem Formulation

Consider the multi-input, multi-output discrete-time system

$$x(k+1) = Ax(k) + Bu(k) + D_1w(k), \quad (1)$$

$$y(k) = Cx(k) + Du(k) + D_2w(k), \quad (2)$$

$$z(k) = E_1x(k) + E_2u(k) + E_0w(k), \quad (3)$$

where $x(k) \in \mathbb{R}^n$, $y(k) \in \mathbb{R}^{l_y}$, $z(k) \in \mathbb{R}^{l_z}$, $u(k) \in \mathbb{R}^{l_u}$, $w(k) \in \mathbb{R}^{l_w}$, and $k \geq 0$. Our goal is to develop an adaptive controller that generates a control signal u that minimizes the performance z in the presence of the exogenous signal w. We assume that measurements of the output y and the performance z are available for feedback; however, we assume that a direct measurement of the exogenous signal w is not available.

Note that w can represent either a command signal to be followed, an external disturbance to be rejected, or both. For example, if $D_1 = 0$, $E_2 = 0$, and $E_0 \neq 0$, then the objective is to have the output E_1x follow the command signal $-E_0w$. On the other hand, if $D_1 \neq 0$, $E_2 = 0$, and $E_0 = 0$, then the objective is to reject the disturbance w from the performance measurement E_1x. The combined command following and disturbance rejection problem is addressed when D_1 and E_0 are block matrices. Lastly, if D_1 and E_0 are empty matrices, then the objective is output stabilization, that is, convergence of z to zero.

The performance variable z can include the feedthrough term E_2u. This term allows us to design an adaptive controller where the performance z to be minimized can include a weighting on control authority. For example, if $E_1 = [\ \hat{E}_1^{\rm T}\ \ 0\ ]^{\rm T}$, $E_2 = [\ 0\ \ \hat{E}_2^{\rm T}\ ]^{\rm T}$, and $E_0 = [\ \hat{E}_0^{\rm T}\ \ 0\ ]^{\rm T}$, then the performance z consists of the components $z_1 \triangleq \hat{E}_1x + \hat{E}_0w$ and $z_2 \triangleq \hat{E}_2u$. In this case, the goal is to minimize a weighted combination of z_1 and z_2, where z_1 is the weighted state performance and z_2 is the weighted control authority.

We represent (1) and (3) as the time-series model from u and w to z given by

$$z(k)=\sum_{i=1}^{n}-\alpha_i z(k-i)+\sum_{i=d}^{n}\beta_i u(k-i)+\sum_{i=0}^{n}\gamma_i w(k-i), \quad (4)$$

where $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$, $\beta_d, \ldots, \beta_n \in \mathbb{R}^{l_z \times l_u}$, $\gamma_0, \ldots, \gamma_n \in \mathbb{R}^{l_z \times l_w}$, and the relative degree d is the smallest non-negative integer i such that the ith Markov parameter, either $H_0 \triangleq E_2$ if $i = 0$ or $H_i \triangleq E_1A^{i-1}B$ if $i > 0$, is nonzero. Note that $\beta_d = H_d$.

III. Review of RCAC

In this section we give a brief overview of the RCAC. Full details are given in [11]. RCAC depends on several parameters that are selected *a priori*. Specifically, $n_{\rm c}$ is the controller order, p is the data window size, and μ is the number of Markov parameters. The adaptive update law is based on a quadratic cost function, which involves a time-varying weighting parameter $\zeta(k) > 0$, referred to as the *learning rate* since it affects the convergence speed of the adaptive control algorithm.

We use a strictly proper time-series controller of order $n_{\rm c}$ such that the control $u(k)$ is given by

$$u(k) = \sum_{i=1}^{n_{\rm c}} M_i(k)u(k-i) + \sum_{i=1}^{n_{\rm c}} N_i(k)y(k-i), \quad (5)$$

where $M_i \in \mathbb{R}^{l_u \times l_u}$, $i = 1, \ldots, n_{\rm c}$, and $N_i \in \mathbb{R}^{l_u \times l_y}$, $i = 1, \ldots, n_{\rm c}$, are given by an adaptive update law. The control can be expressed as

$$u(k) = \theta(k)\phi(k), \quad (6)$$

where

$$\theta(k) \triangleq [\ N_1(k)\ \cdots\ N_{n_{\rm c}}(k)\ \ M_1(k)\ \cdots\ M_{n_{\rm c}}(k)\ ] \quad (7)$$

is the *controller parameter block matrix*, and the *regressor vector* $\phi(k)$ is given by

$$\phi(k) \triangleq \begin{bmatrix} y(k-1) \\ \vdots \\ y(k-n_{\rm c}) \\ u(k-1) \\ \vdots \\ u(k-n_{\rm c}) \end{bmatrix} \in \mathbb{R}^{n_{\rm c}(l_u+l_y)}. \quad (8)$$

For positive integers p and μ, we define the *extended performance vector* $Z(k)$, and the *extended control vector* $U(k)$ by

$$Z(k) \triangleq \begin{bmatrix} z(k-1) \\ \vdots \\ z(k-p_c) \end{bmatrix}, \quad U(k) \triangleq \begin{bmatrix} u(k-1) \\ \vdots \\ u(k-p_c) \end{bmatrix}, \quad (9)$$

where $p_c \triangleq n + \mu + p - 1$.

From (6), it follows that the extended control vector $U(k)$ can be written as

$$U(k) \triangleq \sum_{i=1}^{p_c} L_i \theta(k-i)\phi(k-i), \quad (10)$$

where

$$L_i \triangleq \begin{bmatrix} 0_{(i-1)l_u \times l_u} \\ I_{l_u} \\ 0_{(p_c-i)l_u \times l_u} \end{bmatrix} \in \mathbb{R}^{p_c l_u \times l_u}. \quad (11)$$

We define the *surrogate performance vector* $\hat{Z}(\hat{\theta}(k), k)$ by

$$\hat{Z}(\hat{\theta}(k), k) \triangleq Z(k) - \bar{B}_{zu}\left(U(k) - \hat{U}(k)\right), \quad (12)$$

where

$$\hat{U}(k) \triangleq \sum_{i=1}^{p_c} L_i \hat{\theta}(k)\phi(k-i), \quad (13)$$

and $\hat{\theta}(k) \in \mathbb{R}^{l_u \times [n_c(l_u+l_y)]}$ is the *surrogate controller parameter block matrix*. The block-Toeplitz *surrogate control matrix* $\bar{B}_{zu}$ is given by

$$\bar{B}_{zu} \triangleq \begin{bmatrix} 0_{l_z \times l_u} & \cdots & 0_{l_z \times l_u} & H_d & \cdots & \cdots & H_\mu & 0_{l_z \times l_u} & \cdots & 0_{l_z \times l_u} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0_{l_z \times l_u} & \cdots & 0_{l_z \times l_u} & 0_{l_z \times l_u} & \cdots & \cdots & 0_{l_z \times l_u} & H_d & \cdots & H_\mu \end{bmatrix}, \quad (14)$$

where the *relative degree* d is the smallest positive integer i such that the ith Markov parameter $H_i \triangleq C_0 A_0^{i-1} B_0$ is nonzero. The leading zeros in the first row of $\bar{B}_{zu}$ account for the nonzero relative degree d. The algorithm places no constraints on either the value of d or the rank of H_d or $\bar{B}_{zu}$.

We now consider the cost function

$$J(\hat{\theta}, k) \triangleq \hat{Z}^{\mathrm{T}}(\hat{\theta}, k)\hat{Z}(\hat{\theta}, k) + \zeta(k)\mathrm{tr}\left[(\hat{\theta}-\theta)^{\mathrm{T}}(\hat{\theta}-\theta)\right], \quad (15)$$

where the positive scalar $\zeta(k)$ is the learning rate. Substituting (12) into (15), the cost function can be written as the quadratic form

$$J(\hat{\theta}, k) = \left(\mathrm{vec}\ \hat{\theta}\right)^{\mathrm{T}} A(k)\mathrm{vec}\ \hat{\theta} + b^{\mathrm{T}}\mathrm{vec}\ \hat{\theta} + c(k), \quad (16)$$

where

$$\begin{aligned} D(k) &\triangleq \sum_{i=1}^{p_c} \phi^{\mathrm{T}}(k-i) \otimes (\bar{B}_{zu} L_i), \\ f(k) &\triangleq Z(k) - \bar{B}_{zu} U(k), \\ A(k) &\triangleq D^{\mathrm{T}}(k)D(k) + \zeta(k) I_{n_c l_u(l_u+l_y)}, \\ b(k) &\triangleq 2D^{\mathrm{T}}(k)f(k) - 2\zeta(k)\mathrm{vec}\ \theta(k), \\ c(k) &\triangleq f(k)^{\mathrm{T}} f(k) + \zeta(k)\mathrm{tr}\left[\theta^{\mathrm{T}}(k)\theta(k)\right]. \end{aligned} \quad (17)$$

Since $A(k)$ is positive definite, $J(\hat{\theta}, k)$ has the strict global minimizer

$$\hat{\theta}(k) = -\frac{1}{2}\mathrm{vec}^{-1}(A(k)^{-1}b(k)). \quad (18)$$

The controller gain update law is $\theta(k+1) = \hat{\theta}(k)$.

IV. Constrained Convex Optimization

In this section we extend RCAC by using constrained convex optimization instead of (18) to update $\theta(k) \in \mathbb{R}^{1 \times 2n_c}$. For simplicity, we consider only SISO systems. The denominator coefficients of the controller can be constructed from the last n_c entries of $\theta(k)$ as

$$\mathrm{den}(\theta(k)) \triangleq [1 \ \ -M_1 \ \ -M_2 \ \ \cdots \ \ -M_{n_c}]. \quad (19)$$

The roots of the monic polynomial whose coefficients are given by $\mathrm{den}(\theta(k))$ are the pole locations of the adaptive controller at step k.

In order to prevent the poles of the adaptive controller from approaching the nonminimum-phase-zero locations, we constrain the poles to lie inside a disk centered at the origin of the complex plane. Accordingly, we modify the problem of minimizing (16) by imposing an additional constraint on the companion-form matrix

$$K \triangleq \begin{bmatrix} M_1 & M_2 & \ldots & M_{n_c-1} & M_{n_c} \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix}. \quad (20)$$

We constrain the poles to a disk by bounding the spectral radius of K by a matrix norm, which is a convex function and thus defines a convex region in terms of the denominator coefficients $\mathrm{den}(\theta(k))$. Various matrix norms can be used to bound the spectral radius. For example, every equi-induced norm provides an upper bound, see Corollary 9.4.5 of [12]. One such norm is the maximum singular value of K. Accordingly, the constraint we use is given by

$$\sigma_{\max}(K) \leq \gamma, \gamma > 1. \quad (21)$$

To investigate the conservatism of this bound when applied to matrices of form (20), we generate 10^5 10^{th}-order monic polynomials whose last ten coefficients are taken from a standard normal distribution. For each polynomial we compute the spectral radius $\rho(K)$ and the maximum singular value $\sigma_{\max}(K)$. Each dot in Figure 1 corresponds to a polynomial. Figure 1 suggests that there is little conservatism associated with constraint (21). For the rest of the paper we consider only the maximum singular value to bound the spectral radius. The use of alternative bounds is left for future work.

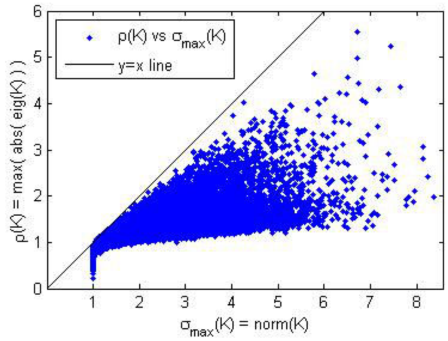

Fig. 1. Spectral radius $\rho(K)$ is plotted versus maximum singular value $\sigma_{\max}(K)$ for 10^5 10^{th} order random monic polynomials. The plot shows that the constraint (21) has little conservatism.

V. Numerical Examples

We now demonstrate the performance of convex-constrained-RCAC (CC-RCAC) with constrained convex optimization on command following problems. The package CVX [13], [14] is used to minimize (16) subject to the constraint (21). For all examples in this section, the control objective is to have the plant output $y(k)$ follow a sinusoid with amplitude 1 and frequency $\frac{\pi}{10}\frac{\text{rad}}{\text{samp}} = 18\frac{\text{deg}}{\text{samp}}$, except where noted otherwise. The adaptive controller (6) is implemented in feedback with $n_c = 10$, $\mu = 1$, $p = 1$, $\zeta \equiv 1$, and $\theta(0) = 0$. Also, we assume that the relative degree d and the first nonzero Markov parameter H_d are known.

The closed-loop is simulated for 1000 steps. Initial conditions are generated at the beginning of each simulation from a Gaussian distribution with mean 0 and variance 0.3. The transient and the steady-state performances are the two performance metrics used to compare RCAC with CC-RCAC. By transient performance we mean the maximum of the absolute value of the performance variable $z(k)$, and by steady-state performance we mean the maximum of the absolute value of the performance variable over the last 100 steps of the simulation.

A. Third-Order Plant with a Known NMP Zero

We apply RCAC to a third-order plant with transfer function given by

$$G_3(z) \triangleq \frac{z - 1.4}{z^3 - 1.7z^2 + 1.2z - 0.35}. \tag{22}$$

RCAC requires knowledge of relative degree, the first nonzero Markov parameter, and the NMP zero. For this example the exact values of these parameters are assumed to be known, and therefore $d = 2$, $H_d = 1$, and $\bar{B}_{zu} = [\ 0 \quad 0 \quad 1 \quad -1.4\ ]$. For RCAC Figure 2 shows the performance $z(k)$, control input $u(k)$, and controller coefficients $\theta(k)$. After the controller is turned on at $k = 100$, the performance variable $z(k)$ approaches zero in about 200 steps, and the controller coefficients converge in about 400 steps. Figure 3 shows the evolution of the controller poles, which settle in about 400 steps.

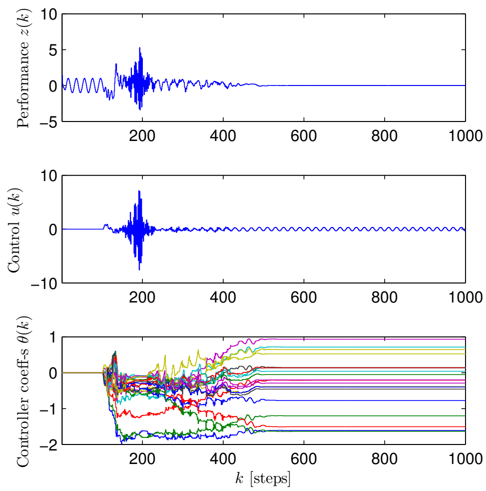

Fig. 2. RCAC performance, control, and controller coefficients are shown as a function of time for the transfer function (22).

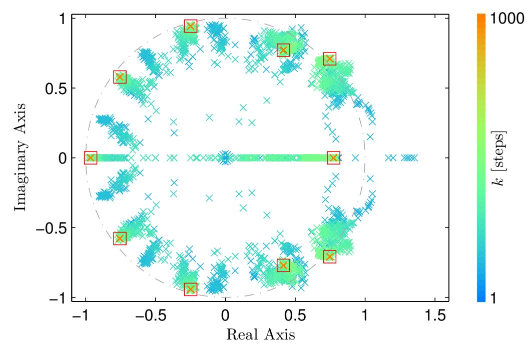

Fig. 3. Evolution of the RCAC controller poles is shown as a function of time in terms of color. Later pole locations are colored with warmer colors and the final locations are marked with a red square. After the controller is turned on, the poles settle in about 400 steps.

Under the same assumptions we apply CC-RCAC with $\gamma = 1.1$. For CC-RCAC Figure 4 shows the performance $z(k)$, control input $u(k)$, and controller coefficients $\theta(k)$. After the controller is turned on, the performance variable $z(k)$ approaches zero and the controller coefficients converge in about 200 steps. Figure 5 shows the evolution of the controller poles, which settle in about 200 steps. For this example Figures 2-5 show that, compared to RCAC, CC-RCAC provides improved performance in terms of decreasing the peak error by about 70 percent and decreasing the settling time by about 50 percent.

B. Third-Order Plant with an Uncertain NMP Zero

We now compare the performance of RCAC with CC-RCAC on the plant (22), but with a NMP zero whose location is varied from 1.1 to 4.1. In addition, we also test the robustness of both algorithms to uncertainty in the estimate of the NMP zero. Therefore, the estimate of the NMP zero is varied from 1.1 to 4.1 regardless of the location of the actual NMP zero. The true values for d and H_d are provided to the algorithm.

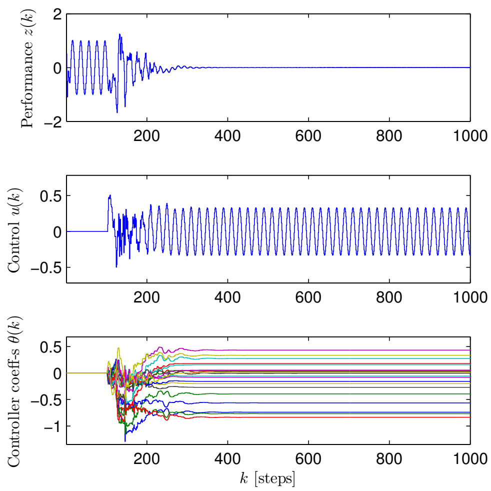

Fig. 4. CC-RCAC performance, control, and controller coefficients are shown as a function of time for the transfer function (22).

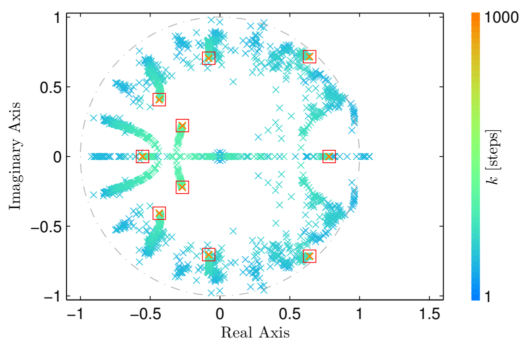

Fig. 5. Evolution of the CC-RCAC controller poles is shown as a function of time in terms of color. After the controller is turned on, the poles settle in about 200 steps.

Figure 6 shows the transient and steady-state performances for various locations of the nonminimum-phase zero and its estimates using RCAC. Note that in some cases (particularly small values of the estimate of the NMP zero locations and large actual NMP zero locations) the closed-loop becomes unstable as signified by white in these plots. Additionally, the color map for these plots is saturated so that dark red corresponds to transient performance greater than 100. Also note that, the diagonal in both plots corresponds to the nominal case, i.e., the zero estimate location is at the actual zero location. Figure 6 indicates that RCAC is more robust to overestimating the NMP zero location than underestimating it and that when the NMP zero is further out on the real axis, RCAC has greater stability margins than when the NMP zero is closer to the unit circle.

Next, Figure 7 shows the CC-RCAC transient and steady-state performance and indicates that, like RCAC, CC-RCAC is more robust to overestimating the NMP zero location than underestimating it. Also, CC-RCAC has wider stability margins when compared to RCAC since the white area in Figure 7 is reduced by about 70 percent as compared to Figure 6.

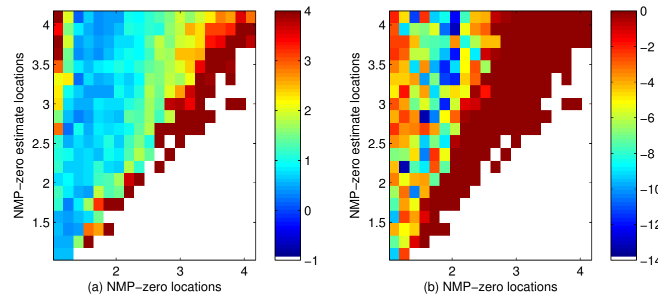

Fig. 6. For the third-order plant (22) with a nonminimum-phase zero whose position is given by the horizontal axis, RCAC uses the estimate of the nonminimum-phase zero whose location is given by the vertical axis. The color in (a) shows the logarithm of the transient performance, whereas (b) shows the logarithm of the steady-state performance.

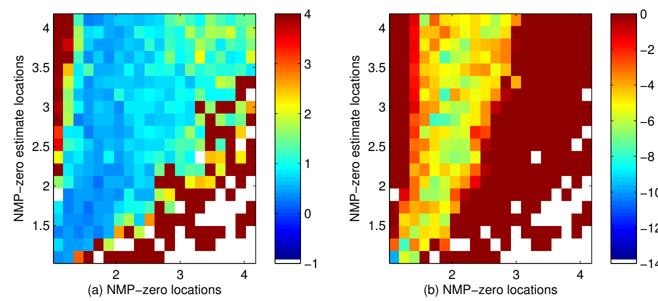

Fig. 7. For the third-order plant with a nonminimum-phase zero whose position is given by the horizontal axis, CC-RCAC with $\gamma = 1.1$ uses the estimate of the nonminimum-phase zero whose location is given by the vertical axis.

C. Effect of Varying γ

We now examine the performance of CC-RCAC for various values of γ, more precisely, from 1.1 to 2.1. The third-order plant (22) with the NMP zero location varied from 1.1 to 4.1, is used. The exact location of the NMP zero is provided to the algorithm. Figure 8 shows the transient and steady-state performances for various values of γ and known NMP zero locations. The strip above each plot shows the RCAC performance for various NMP zero locations. Figure 8 suggests that the best performance for each location of the NMP zero is achieved at the lowest boundary of these plots, namely, $\gamma = 1.1$. $\gamma = 1.1$ was found to work satisfactorily in all cases tested. Additionally, plants with NMP zeros closer to the unit circle have smaller transients and smaller steady-state errors. Lastly, it can be seen that CC-RCAC produces improved transient and steady-state performances compared to RCAC.

D. Fourth-Order Plants with Uncertain NMP Zeros

In this section we compare RCAC and CC-RCAC performance on 20 fourth-order plants with uncertain NMP zeros. The plants used for this example have poles generated from uniform random distribution, first nonzero Markov parameter equaling 1, and the nonminimum-phase zero location of 2. The command following problem for these plants is simulated in feedback with RCAC, while the estimate of the NMP zero is varied from 1.4 to 2.6. Figure 9 shows the resulting transient and steady-state performance. Next, the

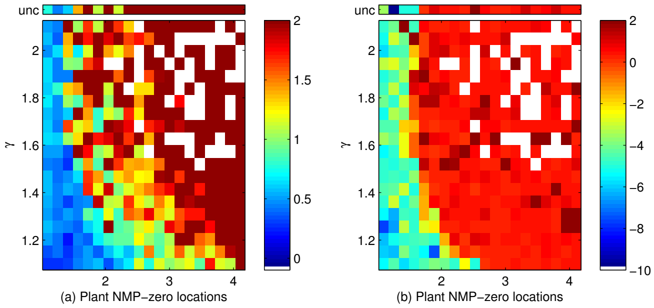

Fig. 8. For the $G_3(z)$ with a nonminimum-phase zero whose position is given by the horizontal axis, CC-RCAC uses the exact location of the nonminimum-phase zero and a value of γ that varies from 1.1 to 4.1. For 20 values of γ, the best performance is achieved with $\gamma = 1.1$.

same set of plants is simulated in feedback with CC-RCAC with $\gamma = 1.1$. Figure 10 shows the resulting transient and steady-state performance. Figures 9 and 10 suggest that both, RCAC and CC-RCAC have problems adapting to plants 7, 9, 11 and 19, since the steady-state performance for both versions of the algorithm is relatively high, as compared to the rest of the plants. Additionally, both versions of the algorithm are more robust to overestimating the location of the nonminimum-phase zero than to underestimating it. However, these Figures also suggest that, on average, CC-RCAC produces smaller transient responses, as compared to RCAC. Lastly, CC-RCAC has greater stability margin, as shown by the fewer white regions in Figure 10 than in Figure 9.

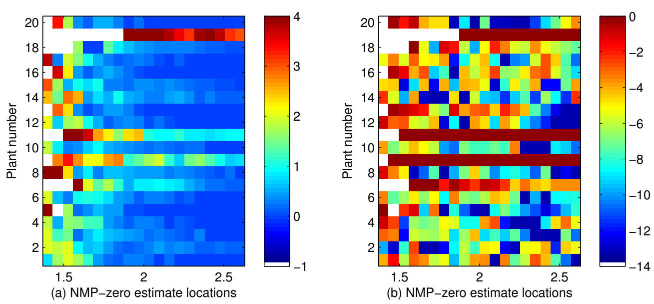

Fig. 9. For 20 random fourth-order plants with a nonminimum-phase zero at 2, RCAC uses an estimate of the nonminimum-phase zero whose location is given by the horizontal axis.

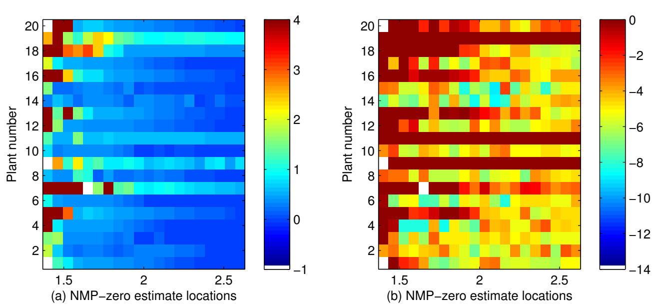

Fig. 10. For 20 random fourth-order plants, CC-RCAC with $\gamma = 1.1$ uses an estimate of the nonminimum-phase zero whose location is given by the horizontal axis.

VI. Conclusion

Retrospective cost adaptive control (RCAC) is applicable to command following and disturbance rejection problems under minimal modeling assumptions, namely, knowledge of the relative degree, first nonzero Markov parameter, and nonminimum-phase zeros. In RCAC, the controller is updated by optimizing a surrogate performance variable that is used to define a retrospective cost. The retrospective cost uses knowledge of the nonminimum-phase zeros to prevent unstable pole-zero cancelation. The goal of this paper is to increase the robustness of RCAC to uncertainty in the nonminimum-phase zero locations. To do this, we extend RCAC to include a convex constraint on the locations of the controller poles. Convex optimization is then used to optimize the retrospective cost subject to this constraint. The resulting convex-constrained retrospective cost adaptive controller (CC-RCAC) was found to have improved transient and steady-state performance as well as improved robustness to uncertainty in the locations of the nonminimum-phase zeros.

References

[1] Hoagg, J. B. and Bernstein, D. S., "Nonminimum-Phase Zeros: Much to Do about Nothing," *IEEE Contr. Sys. Mag.*, Vol. 27, No. June, 2007, pp. 45–57.

[2] Santillo, M. A. and Bernstein, D. S., "Adaptive Control Based on Retrospective Cost Optimization," *AIAA J. Guid. Contr. Dyn.*, Vol. 33, 2010, pp. 289–304.

[3] Hoagg, J. B. and Bernstein, D. S., "Cumulative Retrospective Cost Adaptive Control with RLS-Based Optimization," *Proc. Amer. Contr. Conf.*, Baltimore, MD, June 2010, pp. 4016–4021.

[4] Hoagg, J. B. and Bernstein, D. S., "Retrospective Cost Adaptive Control for Nonminimum-Phase Discrete-Time Systems Part 1: The Ideal Controller and Error System; Part 2: The Adaptive Controller and Stability Analysis," *Proc. Conf. Dec. Contr.*, Atlanta, GA, December 2010, pp. 893–904.

[5] Fledderjohn, M. S., Holzel, M. S., Morozov, A. V., Hoagg, J. B., and Bernstein, D. S., "On the Accuracy of Least Squares Algorithms for Estimating Zeros," *Proc. Amer. Contr. Conf.*, Baltimore, MD, June 2010, pp. 3729–3734.

[6] Morozov, A. V., Hoagg, J. B., and Bernstein, D. S., "A Computational Study of the Performance and Robustness Properties of Retrospective Cost Adaptive Control," *AIAA Guid. Nav. Contr. Conf.*, Toronto, August 2010, AIAA-2010-8011-911.

[7] Anderson, B. D. O., "Challenges of Adaptive Control–Past, Permanent, and Future," *Annual Reviews of Control*, Vol. 32, 2008, pp. 123–135.

[8] Anderson, B. D. O., "Failures of Adaptive Control Theory and Their Resolution," *Comm. Information Systems*, Vol. 5, 2005, pp. 1–20.

[9] Maciejowski, J. M., "Guaranteed stability with subspace methods," *Sys. Contr. Lett.*, Vol. 26, 1995, pp. 153–156.

[10] Lacy, S. L. and Bernstein, D. S., "Subspace Identification with Guaranteed Stability Using Constrained Optimization," *IEEE Trans. Autom. Contr.*, Vol. 48, 2003, pp. 1259–1263.

[11] Santillo, M. A. and Bernstein, D. S., "A Retrospective Correction Filter for Discrete-time Adaptive Control of Nonminimum Phase Systems," *Proc. Conf. Dec. Contr.*, Cancun, Mexico, December 2008, pp. 690–695.

[12] Bernstein, D. S., *Matrix Mathematics*, Princeton University Press, 2nd ed., 2009.

[13] Grant, M. and Boyd, S., "CVX: Matlab Software for Disciplined Convex Programming, version 1.21," `http://cvxr.com/cvx`, Aug. 2010.

[14] Grant, M. and Boyd, S., "Graph implementations for nonsmooth convex programs," *Recent Advances in Learning and Control*, edited by V. Blondel, S. Boyd, and H. Kimura, Lecture Notes in Control and Information Sciences, Springer-Verlag Limited, 2008, pp. 95–110, `http://stanford.edu/~boyd/graph_dcp.html`.