

Linear Fractional Transformation Identification Using Retrospective Cost Optimization

Anthony M. D'Amato* Dennis S. Bernstein**

* Graduate Student, Department of Aerospace Engineering, The University of Michigan, Ann Arbor, MI 48109-2140. (e-mail: amdamato@umich.edu).

** Professor, Department of Aerospace Engineering, The University of Michigan, Ann Arbor, MI 48109-2140. (e-mail: dsbaero@umich.edu)

Abstract: In this paper we use retrospective cost optimization to identify linear fractional transformations (LFTs). This method uses an adaptive controller in feedback with a known system model. The goal is to identify the feedback portion of the LFT by adapting the controller with a retrospective cost. We demonstrate this method on numerical examples of increasing complexity, ranging from linear examples with unknown feedback terms to nonlinear examples. Finally, we examine methods for improving the retrospective cost optimization performance.

1. INTRODUCTION

Although system identification techniques are widely used to construct empirical models from available data, it is often the case that an initial model is available, either from analytical modeling or prior empirical modeling. The identification task is then to use available data to refine the available model, thereby improving its accuracy. This task is variously known as model correction, model refinement, or model updating [2, 3, 4, 5, 8, 10].

In the present paper we consider a model updating approach that is motivated by the similarity of the model reference adaptive control architecture to the model updating problem. This similarity was observed in [8], where the ARMARKOV adaptive control algorithm [6] was used to adaptively refine an initial model.

In contrast to standard system identification methods, model updating based on adaptive control algorithms provides a natural model update in terms of a subsystem interconnected to the primary system through feedback, that is, a linear fractional transformation. This architecture allows the adaptive algorithm to focus on updating only the interconnected subsystem, while accepting the primary system as correct.

The purpose of the present paper is to expand on the model updating results of [10] in several ways. First, while the results of [10] use a simple model (first-order delay) for the primary system, our goal is to consider more general examples in which the primary system has more significant dynamics. Next, we explore the modeling information needed from the primary subsystem as well as the persistency of the external signal needed to ensure convergence and accuracy of the identified subsystem.

* This work was supported by NSF award CNS 0539053 under the DDDAS program

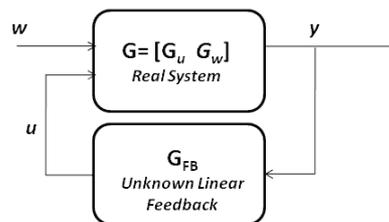


Fig. 1. Linear fractional transformation, a known system G with unknown linear feedback G_{FB} .

Finally, we extend this technique to nonlinear systems. Since the concept of impulse response parameters, which are needed by the adaptive algorithm, does not carry over to nonlinear systems, we consider a technique in which the role of the Markov parameters in the linear case is now played by parameters that are optimized on-line based on the fit accuracy. We demonstrate this technique on several illustrative nonlinear systems.

2. PROBLEM FORMULATION

We seek to identify the SISO feedback term G_{FB} shown in Figure 1 using a given initial model $G = [G_w G_u]$, which is assumed to be accurate. The objective is to identify \hat{G}_{FB} such that the resulting closed-loop model

$$\hat{G}_{cl} = \frac{G_w}{1 - G_u \hat{G}_{FB}} \quad (1)$$

matches the true closed loop system

$$G_{cl} = \frac{G_w}{1 - G_u G_{FB}}. \quad (2)$$

As shown in Figure 2, we use an adaptive feedback model structure in order to identify G_{FB} . To achieve model

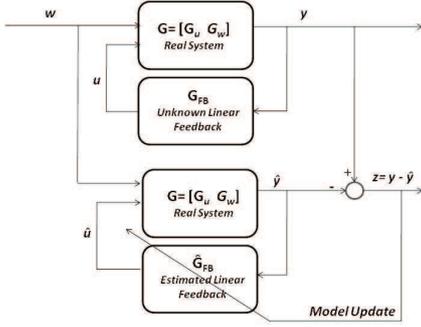


Fig. 2. Linear fractional transformation identification structure

matching, we minimize the performance variable z in the presence of the identification signal w . In particular, we use the retrospective correction filter (RCF) adaptive control algorithm given in [7]. The only signal available to the controller is the plant output y . This problem setup is a minor variation of the approach used in [8].

Consider a realization of the linear discrete-time system given by

$$x(k+1) = Ax(k) + Bw(k), \quad (3)$$

$$y(k) = Cx(k), \quad (4)$$

where $x(k) \in \mathbb{R}^n$, $y_k \in \mathbb{R}^{l_y}$, and $w(k) \in \mathbb{R}^{l_w}$. Furthermore, define

$$z(k) \triangleq y(k) - \hat{y}(k), \quad (5)$$

We thus seek an adaptive output feedback controller \hat{G}_{FB} such that the performance variable z is minimized in the presence of the identification signal w .

3. CONTROLLER CONSTRUCTION

In this section we give a brief overview of the RCF adaptive control algorithm for the control problem represented by [7, 8, 10]. This algorithm is derived from [6] and [7], and the full details of the algorithm are presented in [7].

This algorithm depends on several parameters that are selected *a priori*. Specifically, n_c is the estimated feedback 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 $\alpha(k) > 0$, referred to as the *learning rate* since it affects the convergence speed of the adaptive control algorithm. The methodology for choosing these parameters is as follows, n_c is overestimated, that is, chosen to be greater than the expected order of the unknown feedback. μ is chosen to be 1, generally, μ is used to account for nonminimum phase zeros in G_{cl} . The data window can be chosen as $p \geq 1$, increasing p will quicken convergence at the expense of higher computational complexity. Finally, α is generally chosen as 1, but maybe increased if the algorithm fails to converge.

We use an exactly proper time-series controller of order n_c such that the control $u(k)$ is given by

$$u(k) = \sum_{i=1}^{n_c} M_i(k)u(k-i) + \sum_{i=0}^{n_c} N_i(k)y(k-i), \quad (6)$$

where $M_i \in \mathbb{R}^{l_u \times l_u}$, $i = 1, \dots, n_c$, and $N_i \in \mathbb{R}^{l_u \times l_y}$, $i = 0, \dots, n_c$, are given by an adaptive update law. The control can be expressed as

$$u(k) = \theta(k)\phi(k), \quad (7)$$

where

$$\theta(k) \triangleq [N_0(k) \cdots N_{n_c}(k) M_1(k) \cdots M_{n_c}(k)]$$

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

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

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

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

where $p_c \triangleq \mu + p$.

From (7), it follows that the extended control vector $U(k)$ can be written as

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

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

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

where

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

and $\hat{\theta}(k) \in \mathbb{R}^{l_u \times [n_c l_u + (n_c+1)l_y]}$ is the *surrogate controller parameter block matrix*. The block-Toeplitz *surrogate control matrix* \bar{B}_{zu} is given by

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

where the *relative degree* d is the smallest positive integer i such that the i th 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} .

Furthermore, we define

$$D(k) = \sum_{i=1}^{n_c + \mu - 1} \phi^T(k - i + 1) \otimes L_i, \quad (15)$$

$$f(k) = Z(k) - \bar{B}_{zu} U(k). \quad (16)$$

$$(17)$$

We now consider the cost function

$$J(\hat{\theta}, k) \triangleq \hat{Z}^T(\hat{\theta}, k) R_1(k) \hat{Z}(\hat{\theta}, k) + \hat{U}^T(\hat{\theta}, k) R_2(k) \hat{U}(\hat{\theta}, k) \quad (18)$$

$$+ \text{tr} \left[R_3(k) (\hat{\theta} - \theta(k))^T R_4(k) (\hat{\theta} - \theta(k)) \right], \quad (19)$$

where $R_1(k) \triangleq I_{pl_z}$, $R_2(k) \triangleq 0_{pl_z}$, $R_3(k) \triangleq \alpha(k) I_{n_c(l_u + l_y)}$, $R_4(k) \triangleq I_{l_u}$ and $\alpha(k)$ the positive scalar is the learning rate.

Substituting (12) and (13) into (19), the cost function can be written as the quadratic form

$$J(\hat{\theta}, k) = c(k) + b^T \text{vec } \hat{\theta} + (\text{vec } \hat{\theta})^T A(k) \text{vec } \hat{\theta}, \quad (20)$$

where

$$A(k) = D^T(k) D(k) + \alpha(k) I, \quad (21)$$

$$b(k) = 2D^T(k) f(k) - 2\alpha(k) \text{vec } \theta(k), \quad (22)$$

$$c(k) = f(k)^T R_1(k) f(k) + \text{tr} [R_3(k) \theta^T(k) R_4(k) \theta(k)]. \quad (23)$$

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

$$\hat{\theta}(k) = \frac{1}{2} \text{vec}^{-1} (A(k)^{-1} b(k)). \quad (24)$$

The controller gain update law is

$$\theta(k+1) = \hat{\theta}(k). \quad (25)$$

The key feature of the adaptive control algorithm (7) is surrogate performance variable $Z(k)$ based on the difference between the actual past control inputs $U(k)$ and the recomputed past control inputs based on the current control law $\hat{U}(k)$.

4. LINEAR NUMERICAL EXAMPLES

In this section we consider numerical examples where the model G is a linear system. Consider the second-order continuous-time system

$$G(s) = \frac{\omega_n^2}{s^2 + 2\omega_n \zeta s + \omega_n^2}, \quad (26)$$

where $\omega_n = 100$ and $\zeta = 0.008$. For numerical simulations we discretize $G(s)$ with zero order hold and sampling time $T_s = 0.1$.

For the following examples the estimated feedback order n_c is overestimated as the true order of the unknown feedback

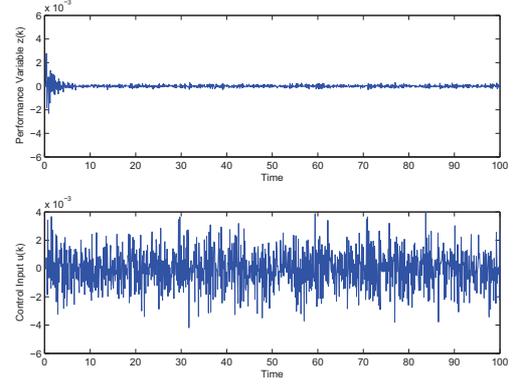


Fig. 3. Performance and feedback variables for Example 1, where z is the difference between the true and identified system outputs. The signal u is the output of the controller \hat{G}_{FB} .

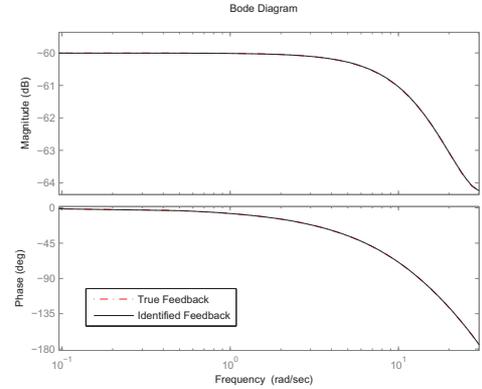


Fig. 4. Frequency response plot comparison, identified and true.

plus 3. We choose $\mu = 1$ and the learning rate $\alpha = 1$. The data window size is $p = 15$.

4.1 Example 1 - No Zeros in G_{FB}

Let G_{FB} be given by

$$G_{\text{FB}}(s) = \frac{1}{(s+20)(s+50)}, \quad (27)$$

where G_{FB} is discretized using a zero-order hold. The identification signal w is gaussian white noise. The performance of the RCF algorithm is shown in Figure 3, which shows that z converges in about 10 seconds. As a second performance metric, Figure 4 shows the frequency response of G_{FB} and the converged controller \hat{G}_{FB} .

4.2 Example 2 - Zeros in the Feedback Term

We now consider the feedback term

$$G_{\text{FB}}(s) = \frac{(s+30)(s+60)}{(s+20)(s+50)(s+10)}. \quad (28)$$

G_{FB} is discretized using a zero order hold with sampling time $T_s = 0.1$. The input signal is white noise. The performance variable shown in Figure 5 approaches zero in about 60 seconds, which corresponds to 600 data points. The frequency response plots shown in Figure 6 indicate

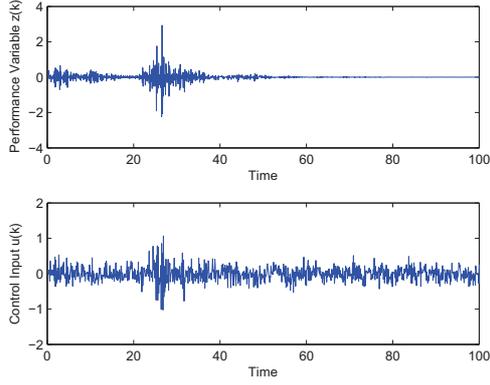


Fig. 5. Performance and feedback variables for Example 2, where z is the difference between the true and identified system outputs. The signal u is the output of the controller \hat{G}_{FB} .

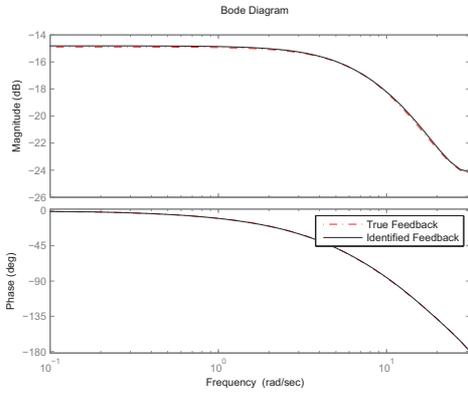


Fig. 6. Frequency response plot comparison, identified and true.

the closeness of the frequency response of \hat{G}_{FB} and G_{FB} .

5. NONLINEAR EXTENSIONS

As a nonlinear extension, (29) and (30) are redefined as a realization of the nonlinear discrete-time system given by

$$x(k+1) = f(x(k), x_0) + h(w(k)), \quad (29)$$

$$y(k) = Cx(k), \quad (30)$$

where $x(k) \in \mathbb{R}^n$, $y_k \in \mathbb{R}^{l_y}$, $w(k) \in \mathbb{R}^{l_w}$. Consider the forced van der Pol oscillator

$$\ddot{x} + \zeta(x^2 - 1)\dot{x} + x = w, \quad (31)$$

where w is the input signal. We discretize (31) and obtain the continuous-time state space equations

$$\dot{x}_1 = x_2, \quad (32)$$

$$\dot{x}_2 = -\zeta x_1^2 x_2 + \zeta x_2 - x_1 + w, \quad (33)$$

$$y = Cx. \quad (34)$$

Introducing a linear feedback

$$G_{\text{FB}} = C_{\text{FB}}(sI - A_{\text{FB}})^{-1}B_{\text{FB}}, \quad (35)$$

we integrate (35) into (32)–(34) as

$$\dot{x}_1 = x_2, \quad (36)$$

$$\dot{x}_2 = -\zeta x_1^2 x_2 + \zeta x_2 - x_1 + u + y_{\text{FB}}, \quad (37)$$

$$\dot{x}_{\text{FB}} = A_{\text{FB}}x_{\text{FB}} + B_{\text{FB}}Cx, \quad (38)$$

$$y_{\text{FB}} = C_{\text{FB}}x_{\text{FB}}, \quad (39)$$

$$y = Cx. \quad (40)$$

Finally, the discrete system is

$$x_1(k+1) = x_2(k)T_s + x_1(k), \quad (41)$$

$$x_2(k+1) = (-\zeta x_1(k)^2 x_2(k) + \zeta x_2(k) - x_1(k) + u(k) + y_{\text{FB}}(k))T_s + x_2(k), \quad (42)$$

$$x_{\text{FB}}(k+1) = \tilde{A}_{\text{FB}}x_{\text{FB}}(k) + \tilde{B}_{\text{FB}}Cx(k), \quad (43)$$

$$y_{\text{FB},k} = C_{\text{FB}}x_{\text{FB}}(k), \quad (44)$$

$$y_k = Cx(k), \quad (45)$$

where

$$\tilde{A}_{\text{FB}} = e^{A_{\text{FB}}T_s}, \quad (47)$$

$$\tilde{B}_{\text{FB}} = A_{\text{FB}}^{-1}(\tilde{A}_{\text{FB}} - I)B_{\text{FB}}, \quad (48)$$

where T_s is the sampling time. By choosing $\zeta = 0.8$ for the following examples, the trajectory of the van der Pol oscillator enters a limit cycle. In the linear case, \bar{B}_{zu} is constructed with the Markov parameters from the known system (14). In the nonlinear case we extract estimates of the Markov parameters by linearizing the van der Pol equations (32)–(34), around the unstable equilibrium at the origin,

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -1 & \zeta \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \quad (49)$$

$$y = [0 \ 1] x,$$

where

$$A = \begin{bmatrix} 0 & 1 \\ -1 & \zeta \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = [0 \ 1]. \quad (50)$$

The first nonzero discrete-time Markov parameter, where $T_s = 0.1$ and $\zeta = 0.8$, is

$$H = CA^{-1}(e^{AT_s} - I)B = 0.01. \quad (51)$$

The resulting surrogate control matrix assuming $\mu = 1$ and $p = 1$ is

$$\bar{B}_{zu} = [0 \ 0.01 \ 0]. \quad (52)$$

5.1 Example 3 - No Zeros in Feedback Term

Assume G is given by (31) and let

$$G_{\text{FB}}(s) = \frac{1}{s+20}. \quad (53)$$

G_{FB} is discretized with a sampling time of 0.1 and zero initial conditions are assumed. The input identification signal is white noise. From Figure 9 the performance variable tends to zero, indicating that the feedback is identified. Note that significantly more data is required to identify the feedback terms for this nonlinear system than for the linear examples.

5.2 Example 4 - Zeros in Feedback Term

Let G be given by (31) and let

$$G_{\text{FB}}(s) = \frac{s+10}{(s+20)(s+50)}. \quad (54)$$

The surrogate control matrix \bar{B}_{zu} is chosen as in (52). From Figure 9 the performance variable tends to zero,

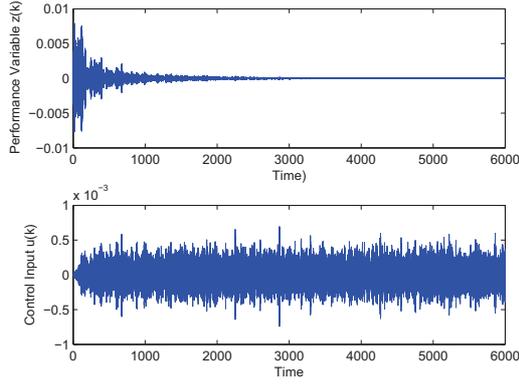


Fig. 7. Performance and feedback variables for Example 3, where z is the difference between the true and identified system outputs. The signal u is the output of the estimated subsystem \hat{G}_{FB} .

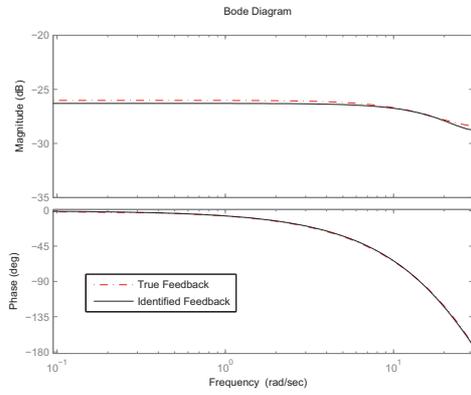


Fig. 8. Frequency response plot comparison, identified and true.

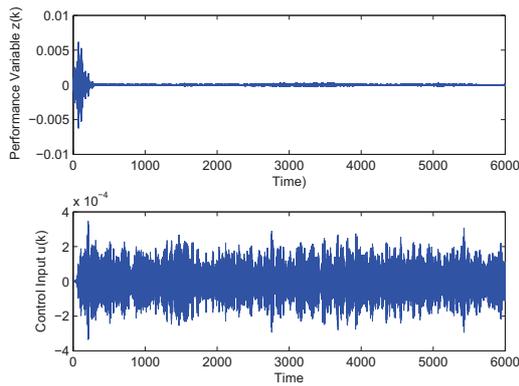


Fig. 9. Performance and feedback variables for Example 4, where z is the difference between the true and identified system outputs. The signal u is the output of the estimated subsystem \hat{G}_{FB} .

indicating that the feedback is identified. The frequency response plots in Figure 10 confirm that the feedback is well approximated. As in Example 3, a large amount of data is required to identify the feedback.

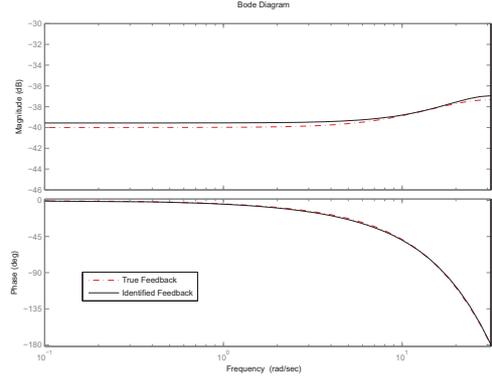


Fig. 10. Frequency response plot comparison, identified and true.

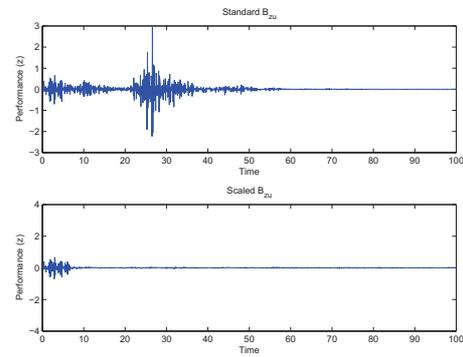


Fig. 11. Performance comparison of the system from Example 2. The top plot is the standard RCF algorithm from Section 3, where the lower plot is the RCF with the time-varying surrogate control matrix \bar{B}_{zu} from (55).

6. PERFORMANCE OPTIMIZATION

To improve identification performance by minimizing the amount of required data, we examine the effect of scaling the surrogate control matrix \bar{B}_{zu} . We introduce a scaling coefficient $\beta(k)$, which is the ratio of $\sigma_y(k)$ and $\sigma_u(k)$, where $\sigma_y(k)$ is the standard deviation of the signal y on the interval $[0, k]$, and $\sigma_u(k)$ is the standard deviation of the estimated feedback signal on $[0, k]$. The time-step-dependent surrogate control matrix becomes

$$B_{zu}(k) = \beta(k)\bar{B}_{zu}(0), \quad (55)$$

where $\bar{B}_{zu}(0)$ is the surrogate control matrix at time step zero. $\bar{B}_{zu}(0)$ is created based on (14) in the linear problem, whereas for the nonlinear problem, $\bar{B}_{zu}(0)$ is estimated. The scaling coefficient is

$$\beta(k) = \frac{\sigma_y(k)}{\sigma_u(k)}. \quad (56)$$

To demonstrate this method we reconsider Examples 2 and 4. First consider G as (26) with feedback given by (28). From Figure 11, we see that the scaling method reduces the transients in the performance variable. Furthermore, by scaling \bar{B}_{zu} we require less data than the same example without using a scaling coefficient. The time history of $\beta(k)$ for the linear example is shown in Figure 12. The coefficient converges almost immediately and remains

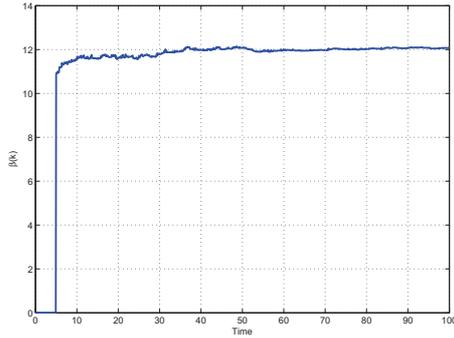


Fig. 12. Time history of $\beta(k)$. For this linear example, $\beta(k)$ reaches steady state after about 5 seconds.

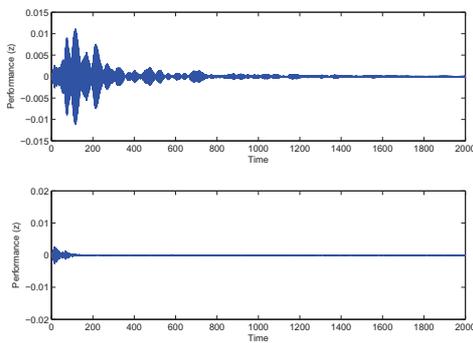


Fig. 13. Performance comparison of the system from Example 2. The upper plot is the standard RCF algorithm from Section 3, where the lower plot is the RCF with the time-varying surrogate control matrix \bar{B}_{zu} from (55).

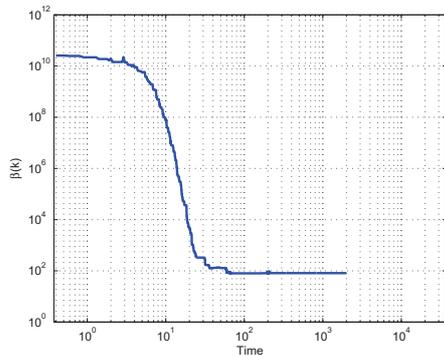


Fig. 14. Time history of $\beta(k)$. For this nonlinear example, $\beta(k)$ fluctuates through a larger range of scaling factors than the linear case, before convergence

constant throughout the run. Consider again G as (31), with feedback (54). From Figure 13, the magnitude of the transient is reduced, the performance also tends to zero at a higher rate than when \bar{B}_{zu} is a fixed estimated value. The time history of $\beta(k)$ for the nonlinear example is shown in Figure 14. This time history differs from the linear case in Figure 12, we observe that there is a longer transient period before reaching steady state. Scaling the surrogate control matrix based on the ratio of signal standard devia-

tions appears to give significant performance gains without sacrificing computational efficiency.

7. CONCLUSION

We have demonstrated the use of a retrospective correction filter to identify linear fractional transformations (LFTs). The method was demonstrated using a known linear initial model with feedback of varying complexity. We also demonstrated LFT identification with a nonlinear initial model with linear feedback of increasing complexity. Finally, a method for artificially increasing the adaptive control authority with the goal of decreasing the time and amount of data required to identify feedback was investigated and demonstrated on linear and nonlinear examples.

8. ACKNOWLEDGMENTS

The authors would like to acknowledge M.A. Santillo for his work developing the retrospective correction filter algorithm.

REFERENCES

- [1] C. Minas and D. Inman. Matching finite element models to modal data. *J. Vibration Acoust.*, vol. 112, pp. 84 - 92, 1990.
- [2] J.B. Carvalho, B.N. Datta, W. Lin, and C. Wang. Symmetry preserving eigenvalue embedding in finite-element model updating of vibrating structures. *J. Sound Vibration*, 290 (2006) 839 - 864.
- [3] M.I. Friswell, J.E. Mottershead. *Finite Element Model Updating in Structural Dynamics*. Kluwer, Dordrecht, 1995.
- [4] S. Mijanovic, G. E. Stewart, G. A. Dumont, and M. S. Davies. A controller perturbation technique for transferring closed-loop stability between systems, *Automatica*, vol. 39, pp. 1783 - 1791, 2003.
- [5] S. O. R. Moheimani. Model correction for sampled-data models of structures. *J. Guidance, Contr., and Dynamics*, vol. 24(3), pp. 634 - 637, 2001.
- [6] R. Venugopal and D. S. Bernstein. Adaptive Disturbance Rejection Using ARMARKOV System Representations. *IEEE Trans. Contr. Sys. Tech.*, Vol. 8, pp. 257 - 269, 2000.
- [7] M. A. Santillo and D. S. Bernstein. A retrospective correction filter for discrete-time adaptive control of nonminimum phase systems. in *Proc. Conf. Dec. Contr.*, pp. 690 - 695, Cancun, Mexico, December 2008.
- [8] H. Palanhandalam-Madapusi, E. L. Renk, and D. S. Bernstein. Data-Based Model Refinement for Linear and Hammerstein Systems Using Subspace Identification and Adaptive Disturbance Rejection. *Proc. Conf. Contr. Appl.*, pp. 1630 - 1635, Toronto, Canada, August 2005.
- [10] M. A. Santillo, A. M. D'Amato, and D. S. Bernstein. System Identification Using a Retrospective Correction Filter for Adaptive Feedback Model Updating. *Proc. ACC 2009*, St. Louis, MO.
- [10] A. M. D'Amato, B. J. Arrit, et al, "Structural Health Determination and Model Refinement for a Deployable Composite Boom" *Proc. AIAA SDM*, Palm Springs, CA, 2009.