Parameter-Constrained Adaptive Control[†]

W. D. Timmons,*[,‡] H. J. Chizeck,[§,||] F. Casas,[‡] V. Chankong,[§] and P. G. Katona[⊥]

Wm Timmons & Associates, 513 Sill Avenue, Cuyahoga Falls, OH 44221, Departments of Systems Engineering and Biomedical Engineering, Case Western Reserve University, Cleveland, Ohio 44106, and The Whitaker Foundation, 1700 North Moore Street, Suite 2200, Rosslyn, Virginia 22209

Under certain conditions, parameter constraints that impose *a priori* information about the open-loop system can dramatically improve the performance of explicit adaptive controllers. Under other conditions, the constraints can actually decrease performance. First, this paper presents a novel parameter-constrained identifier on the basis of an efficient, quadratic program solver applied semirecursively, making it ideal for real-time adaptive control. Second, several useful linear constraints for second-order ARMAX models are provided, along with a few examples of their development. Third, the algorithm and the constraints are applied to a benchmark model to explore several conditions, summarized as six guidelines, under which parameter constraints improve or worsen adaptive control. In this last part, it is shown that common orthogonal projection can produce poor results. It is also shown that *a priori* information is increasingly valuable as excitation decreases and that it is especially useful for adaptive control when combined with re-identification techniques. These results are then applied to the pharmacological control of a time-varying second-order ARMAX model of blood pressure.

1. Introduction

For systems that are time-varying or nonlinear, an adaptive controller based on a time series model (ARMAX, CARIMA, NARMAX, etc.) may be a useful alternative to a classical controller design, since little *a priori* information is required. Ironically, the information that is available is often discarded when an adaptive controller is used. As a result, the controller may need open-loop probing before initiating control; after initiation, it may be susceptible to gain and offset disturbances, temporary bursting, and other problems. If the *a priori* knowledge were used instead of being discarded, these problems might be avoided or reduced. This idea is not new; almost 20 years ago the imposition of exact information as equality constraints on one or more of the model parameters was shown to dramatically improve controller performance (Goodwin and Payne, 1977). Others have since confirmed and refined these findings (Bai and Sastry, 1986; Chia et al., 1991; Clary and Franklin, 1984; Fletcher, 1987; Li, 1989; Zheng, 1989; Dasgupta, 1986). Unfortunately, exactness is rather restrictive, and so this approach is not widely applicable. Instead, most knowledge falls into the class of inequality constraints. An algorithm that imposes both equality and inequality constraints would be more useful and have wider applicability. However, these algorithms are complicated and difficult to program. Probably for this reason, most studies on inequality-constrained identification for adaptive control are limited to either the bounding of a few critical parameters (Goodwin and Sin, 1984) or the use of simplified, suboptimal algorithms with loose parameter bounds. The latter includes the σ-modifiers (Ortega and Tang, 1989), the radial contractors (Praly et al., 1989), and the simple saturators (Goodwin and Sin, 1984). Those studies primarily address the related and important issue of global asymptotic stability, which is not the same as the *practical use* of *a priori* information as parameter constraints.

While simplified algorithms and reduced constraint sets often dramatically improve controller performance, it is shown below that they can sometimes *decrease* controller performance and even *exacerbate* temporary disturbances. The possibility of poor behavior raises considerable safety concerns for life-critical applications such as the control of vital signs in hospital patients, chemotherapy veno-infusion, and anesthesia regulation, where temporary instabilities and poor transient responses can be life-threatening. For these types of applications, it becomes important to understand how and when *a priori* information helps or hinders controller performance.

The addition of mixed equality and inequality constraints to an adaptive controller creates a nonlinear system that sometimes exhibits unexpected behaviors. Since an analytic analysis would be difficult, an empirical approach is used here to develop and demonstrate six guidelines for the safe and effective use of parameter-constrained adaptive control. First, however, equations are developed for a novel identifier that imposes mixed equality and inequality constraints both *optimally* and *semirecursively*, two key properties that make the identifier attractive for real-time adaptive control. Since the equations form a quadratic program (QP), a fast, compact, and novel QP algorithm is included in the Appendix. Following the section on identifier development, the construction of several constraints from commonly available *a priori* information is illustrated and then followed by the empirical analysis. The results of the analysis are then applied to the pharmacological control of a time-varying second-order ARMAX model of blood pressure.

[†] A preliminary version of this work was presented at the 12th IFAC World Congress, July 18−23, 1993, Sydney, Australia.

[‡] Wm Timmons & Associates.

[§] Department of Systems Engineering, Case Western Reserve University.

[||] Department of Biomedical Engineering, Case Western Reserve University.

[⊥] The Whitaker Foundation.

S0888-5885(96)00659-8 CCC: $14.00

2. Problem Formulation

Consider a linear or pseudolinear model in predictor form:

$$y(t) = \phi^{T}(t-1)\boldsymbol{\theta} + e(t) \tag{1}$$

where $e(t)$ is a zero mean white noise sequence. This form includes a broad class of time series models (ARMAX, NARMAX, CARIMA, etc.). For example, given the discrete time ARMAX model

$$y(t) = a_1 y(t-1) + ... + a_n y(t-n) + b_1 u(t-1) + ... b_m u(t-m) + e(t) \tag{2}$$

the vectors $\boldsymbol{\theta}$ and $\phi(t-1)$ may be defined as

$$\boldsymbol{\theta} \mathrel{\underline{\Delta}} [a_1, ..., a_n, b_1, ..., b_m]^{T}$$

$$\phi(t-1) \mathrel{\underline{\Delta}} [y(t-1), ..., y(t-n), u(t-1), ..., u(t-m)]^{T} \tag{3}$$

2.1. The Unconstrained Estimates. For $\boldsymbol{\theta}$ unknown, the exponentially weighted recursive least squares (RLS) estimate is (Goodwin and Sin, 1984):

$$\boldsymbol{\theta}_f(t) = \boldsymbol{\theta}_f(t-1) + \frac{\mathbf{P}(t-2)\phi(t-1)\cdot[y(t) - \phi(t-1)^{T}\boldsymbol{\theta}_f(t-1)]}{\alpha(t-1) + \phi(t-1)^{T}\mathbf{P}(t-2)\phi(t-1)}$$

$$\mathbf{P}(t-1) = \frac{1}{\alpha(t-1)}\cdot\left[\mathbf{P}(t-2) - \frac{\mathbf{P}(t-2)\phi(t-1)\phi(t-1)^{T}\mathbf{P}(t-2)}{\alpha(t-1) + \phi(t-1)^{T}\mathbf{P}(t-2)\phi(t-1)}\right] \tag{4}$$

where $0 < \alpha(t) \le 1$ is a scalar forgetting factor, $\mathbf{P}(t-1)$ is a matrix proportional to the inverted data covariance matrix, and the subscript f signifies that $\boldsymbol{\theta}$ is the *free*, or *unconstrained*, solution. The RLS estimate minimizes the sum of the squared prediction errors:

$$\mathbf{J} = {}^{1}/_{2}[\mathbf{Y}(t) - \mathbf{X}(t-1)\hat{\boldsymbol{\theta}}]^{T}\mathbf{W}(t)[\mathbf{Y}(t) - \mathbf{X}(t-1)\hat{\boldsymbol{\theta}}] \tag{5}$$

where $\hat{\boldsymbol{\theta}}$ is the vector of unknown parameters to be estimated, $\mathbf{W}(t)$ is a diagonal matrix defined by the $\alpha(t)$, and

$$\mathbf{Y}(t) \mathrel{\underline{\Delta}} [y(1), y(2), ..., y(t)]^{T}$$

$$\mathbf{X}(t-1) \mathrel{\underline{\Delta}} [\phi(0)|\phi(1)| ... |\phi(t-1)]^{T} \tag{6}$$

The equivalent nonrecursive solution is

$$\boldsymbol{\theta}_f(t) = \mathbf{P}(t-1)\mathbf{X}^{T}(t-1)\mathbf{W}(t)\mathbf{Y}(t)$$

$$\mathbf{P}(t-1) = [\mathbf{X}^{T}(t-1)\mathbf{W}(t)\mathbf{X}(t-1)]^{-1} \tag{7}$$

2.2. The Optimal Projector. Now consider the minimization of (5) subject to the following linear equality and inequality constraints (a convex quadratic program):

$$\mathbf{M}\boldsymbol{\theta} = \mathbf{K}$$

$$\mathbf{L}\boldsymbol{\theta} \le \mathbf{C} \tag{8}$$

The Lagrangian for this system is (Fletcher, 1987):

$$L(\hat{\boldsymbol{\theta}},\boldsymbol{\mu},\boldsymbol{\lambda}) = {}^{1}/_{2}[\mathbf{Y}(t) - \mathbf{X}(t-1)\hat{\boldsymbol{\theta}}]^{T}\mathbf{W}(t) \times [\mathbf{Y}(t) - \mathbf{X}(t-1)\hat{\boldsymbol{\theta}}] - \boldsymbol{\mu}^{T}(\mathbf{K} - \mathbf{M}\hat{\boldsymbol{\theta}}) - \boldsymbol{\lambda}^{T}(\mathbf{C} - \mathbf{L}\hat{\boldsymbol{\theta}}) \tag{9}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$ are vectors of Lagrange multipliers associated with the equality and inequality constraints, respectively. For positive definite $\mathbf{X}^{T}\mathbf{W}\mathbf{X}$, or equivalently, positive definite $\mathbf{P}$, the first-order Kuhn−Tucker conditions are the necessary and sufficient conditions for a unique global minimum (Luenberger, 1984):

$$\mathbf{K} - \mathbf{M}\hat{\boldsymbol{\theta}} = 0 \tag{10}$$

$$\mathbf{C} - \mathbf{L}\hat{\boldsymbol{\theta}} \ge 0 \tag{11}$$

$$\boldsymbol{\lambda} \ge 0 \tag{12}$$

$$\boldsymbol{\lambda}^{T}(\mathbf{C} - \mathbf{L}\hat{\boldsymbol{\theta}}) = 0 \tag{13}$$

$$-\mathbf{X}^{T}\mathbf{W}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\theta}}) + \mathbf{M}^{T}\boldsymbol{\mu} + \mathbf{L}^{T}\boldsymbol{\lambda} = 0 \tag{14}$$

(dropping the time subscripts for convenience). Solving (14) for $\hat{\boldsymbol{\theta}}$ (which we denote as $\boldsymbol{\theta}_c$ since it is the *constrained* solution), we obtain

$$\boldsymbol{\theta}_c = \boldsymbol{\theta}_f - \mathbf{P}\mathbf{M}^{T}\boldsymbol{\mu} - \mathbf{P}\mathbf{L}^{T}\boldsymbol{\lambda} \tag{15}$$

which is in the form of the unconstrained estimate plus corrections due to the constraints. $\mathbf{P}$ and $\boldsymbol{\theta}_f$ can be computed from (4) or (7), depending on the speed and numerical requirements: in applications in which time is scarce (e.g., adaptive control), the recursive form is preferred. All that remains is to determine the Lagrange multipliers $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$, which must be performed in batch using quadratic programming (hence the semirecursive nature of the identifier). Positive semidefinite complementary linear programming (CLP) is a relatively fast, compact quadratic-programming technique similar to the simplex algorithm for linear programming (Dantzig and Cottle, 1967; Golub and Saunders, 1970). An enhanced version developed by Timmons (1992) specifically for real-time applications is described in the Appendix.

A CLP for this problem may be constructed as follows. Define the slack variables $\boldsymbol{\sigma}$ and $\boldsymbol{\nu}$ as the constraint errors $(\mathbf{K} - \mathbf{M}\hat{\boldsymbol{\theta}})$ and $(\mathbf{C} - \mathbf{L}\hat{\boldsymbol{\theta}})$. Using (15), eliminate $\boldsymbol{\theta}_c$ to obtain

$$\begin{bmatrix}\boldsymbol{\sigma}\\ \boldsymbol{\nu}\end{bmatrix} = \begin{bmatrix}\mathbf{MPM}^{T} & \mathbf{MPL}^{T}\\ \mathbf{LPM}^{T} & \mathbf{LPL}^{T}\end{bmatrix}\begin{bmatrix}\boldsymbol{\mu}\\ \boldsymbol{\lambda}\end{bmatrix} + \begin{bmatrix}\mathbf{K} - \mathbf{M}\boldsymbol{\theta}_f\\ \mathbf{C} - \mathbf{L}\boldsymbol{\theta}_f\end{bmatrix} \tag{16}$$

with the conditions

$$\boldsymbol{\sigma} = 0, \quad \boldsymbol{\nu} \ge 0, \quad \boldsymbol{\lambda} \ge 0$$

$$\boldsymbol{\sigma}^{T}\boldsymbol{\mu} + \boldsymbol{\nu}^{T}\boldsymbol{\lambda} = 0 \tag{17}$$

Equations 16 and 17 form the CLP. Note that if there are only equality constraints, the solution reduces to a closed form:

$$\boldsymbol{\mu} = (\mathbf{MPM}^{T})^{-1}(\mathbf{M}\boldsymbol{\theta}_f - \mathbf{K})$$

$$(\boldsymbol{\lambda} = \varnothing) \tag{18}$$

2.3. The Orthogonal Projector. The solution (15) could also have been obtained by minimizing the cost function

$$J_{proj} = {}^{1}/_{2}[\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_f]^{T}\mathbf{P}^{-1}[\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_f] \tag{19}$$

subject to the constraints in (8). In this formulation, the optimal (least squares) correction terms in (15) represent the $\mathbf{P}$-weighted projection of $\boldsymbol{\theta}_f$ onto the

Table 1. ARMAX Constraints from Commonly Available *a Priori* Knowledge[a]

	knowledge	constraint	comments
1	individual parameters	$\theta_{i_{\min}} \le \theta_i \le \theta_{i_{\max}}$	
2	second-order open-loop stability	$a_1 + a_2 \le 1$ $-a_1 + a_2 \le 1$ $-a_2 \le 1$	
3	steady-state gain: $G_{ss_{\min}} \le G_{ss} \le G_{ss_{\max}}$	$\Sigma b_i + G_{ss_{\min}} \Sigma a_i \ge G_{ss_{\min}}$ $\Sigma b_i + G_{ss_{\max}} \Sigma a_i \le G_{ss_{\max}}$	Use only with constrained open-loop stable systems. Do not use with entry 5.
4	one SSP: (U_1,Y_1)	$\bar{y}(t) \underline{\Delta}\, y(t) - Y_1$ $\bar{u}(t) \underline{\Delta}\, u(t) - U_1$	Use $\bar{y}$ and $\bar{u}$ to estimate a and b parameters. This approach is equivalent to, but more efficient than, an equality constraint.
5	two SSP's: (U_1,Y_1), (U_2,Y_2)	$Y_2 - Y_1 = (Y_2 - Y_1)\Sigma a_i + (U_2 - U_1)\Sigma b_i$	Imposes a steady-state gain. Use in conjunction with entry 4, but in place of entry 3.
6	one SSP range: $(U_1,Y_{\min})$ to $(U_1,Y_{\max})$	$(1 - \Sigma a_i)Y_{\min} \ge (\Sigma b_i)U_1 + D$ $(1 - \Sigma a_i)Y_{\max} \le (\Sigma b_i)U_1 + D$	See text for explanation of D.
7	one SSP range: $(U_1,Y_{\min})$ to $(U_1,Y_{\max})$	$\mathbf{A}(q^{-1})(y_{lp}(t) - Y_{\min}) \le \mathbf{B}(q^{-1})(u_{lp}(t) - U_1)$ $\mathbf{A}(q^{-1})(y_{lp}(t) - Y_{\max}) \ge \mathbf{B}(q^{-1})(u_{lp}(t) - U_1)$	$\mathbf{A}(q^{-1})$ and $\mathbf{B}(q^{-1})$ is standard ARMAX notation. Subscript lp indicates low-pass filtering; high-pass filtered data is used to estimate parameters. Noise may invalidate constraints. See text.
8	open-loop settling time (second-order systems)	$ra_1 + a_2 \le r^2$ $-ra_1 + a_2 \le r^2$ $-a_2 \le r^2$	$r = \exp(-4.6T/\tau_s)$, where T is the sampling interval and τ_s is the settling time.
9	min and max settling time (certain second-order systems)	$-r_{\max}^2\, a_1 + (r_{\min} - 2r_{\max})a_2 \le -r_{\max}^2\, r_{\min}$ $r_{\max}a_1 + a_2 \le r_{\max}^2$ $a_2 \le 0$	Only applies to sampled, continuous time systems with real poles. $r_{\min}$ and $r_{\max}$ are the radii associated with the minimum and maximum settling times.
10	open-loop minimum phase zeros (second-order systems)	$\mathrm{sgn}(b_1)(b_1r_z^2 + b_2r_z + b_3) \ge 0$ $\mathrm{sgn}(b_1)(b_1r_z^2 - b_2r_z + b_3) \ge 0$ $\mathrm{sgn}(b_1)(b_1r_z^2 - b_3) \ge 0$	For stable second-order systems, b_1 has the same sign as G_{ss}. r_z = maximum radius of zeros (≤ 1).
11	minimum phase noise polynomials (second-order systems)	$-r_nc_1 - c_2 \le r_n^2$ $r_nc_1 - c_2 \le r_n^2$ $c_2 \le r_n^2$	r_n = maximum radius of the noise polynomial roots (≤ 1).

[a] See Timmons (1992) for detailed derivations.

constraint surface. By replacing **P** with the identity matrix, an orthogonal projector is obtained. Assuming the constraints are true, parameter bias can never be increased with this algorithm, which is something that cannot be said for the optimal projector (Chia, 1991).

The orthogonal projector can often be greatly simplified. For example, when the constraints consist of individual parameter bounds only, the orthogonal projector reduces to a saturator:

$$\theta_{c_i} = \begin{cases} \theta_{i_{\min}} & \text{if } \theta_{f_i} < \theta_{i_{\min}} \\ \theta_{f_i} & \text{if } \theta_{i_{\min}} \le \theta_{f_i} \le \theta_{i_{\max}} \\ \theta_{i_{\max}} & \text{if } \theta_{f_i} > \theta_{i_{\max}} \end{cases} \qquad \forall i \tag{20}$$

and when the constraints are in the form of a sphere, it reduces to a radial contractor

$$\boldsymbol{\theta}_c = \begin{cases} \boldsymbol{\theta}_f & \text{if } \|\boldsymbol{\theta}_f - \boldsymbol{\theta}_0\| \le \rho \\ \boldsymbol{\theta}_0 + \rho\dfrac{\boldsymbol{\theta}_f - \boldsymbol{\theta}_0}{\|\boldsymbol{\theta}_f - \boldsymbol{\theta}_0\|} & \text{otherwise} \end{cases} \tag{21}$$

where $\boldsymbol{\theta}_0$ is the center of the sphere and ρ is its radius. The orthogonal projector thus has a seeming advantage over the optimal projector. This is discussed further below.

3. Armax Constraints

A range on steady-state gain and settling time, an approximate pole or zero location, and the sign of one or more parameters are often known. Classifications such as "open-loop stable" or "well-damped" also convey useful information. Linear ARMAX constraints that impose this type of information are listed in Table 1. While some constraints are general, for clarity and simplicity, most are targeted toward the common, second-order ARMAX model. Many of the constraints rely on the Jury criteria for the sign of $1 - \Sigma a_i$ and hence require stability. For unstable systems, the constraints must be reformulated. These modifications, as well as extensions to other model forms, are mostly straightforward. In the following, constraint development is briefly illustrated and then followed by an elaboration on several of the more complicated constraints in Table 1.

3.1. Example of Constraint Development: Steady-State Gain. Since steady-state gain is only defined for stable systems, open-loop stability must also be imposed on the estimates. From eq (2), G_{ss} may be calculated as

$$G_{ss} = \frac{\sum_i b_i}{1 - \sum_i a_i} \tag{22}$$

For $G_{ss_{\min}} \le G_{ss} \le G_{ss_{\max}}$, we obtain two inequality constraints, which can be put into linear (in θ) form *only if the sign of* $1 - \Sigma a_i$ *is known*. By the Jury criteria, the sign must be positive, resulting in entry 3 in Table 1.

3.2. Steady-State Points. A steady-state point (SSP) is a known steady-state input/output pair associated with the model. Often the background level (the steady-state unforced output) is known. Sometimes another point, such as an equilibrium point for a chemical reaction, is also known. For many systems, these points drift. For nonlinear systems, these points may appear to drift due to linearization and hence may range farther than anticipated. In such systems, these points may better be regarded as free or slack variables

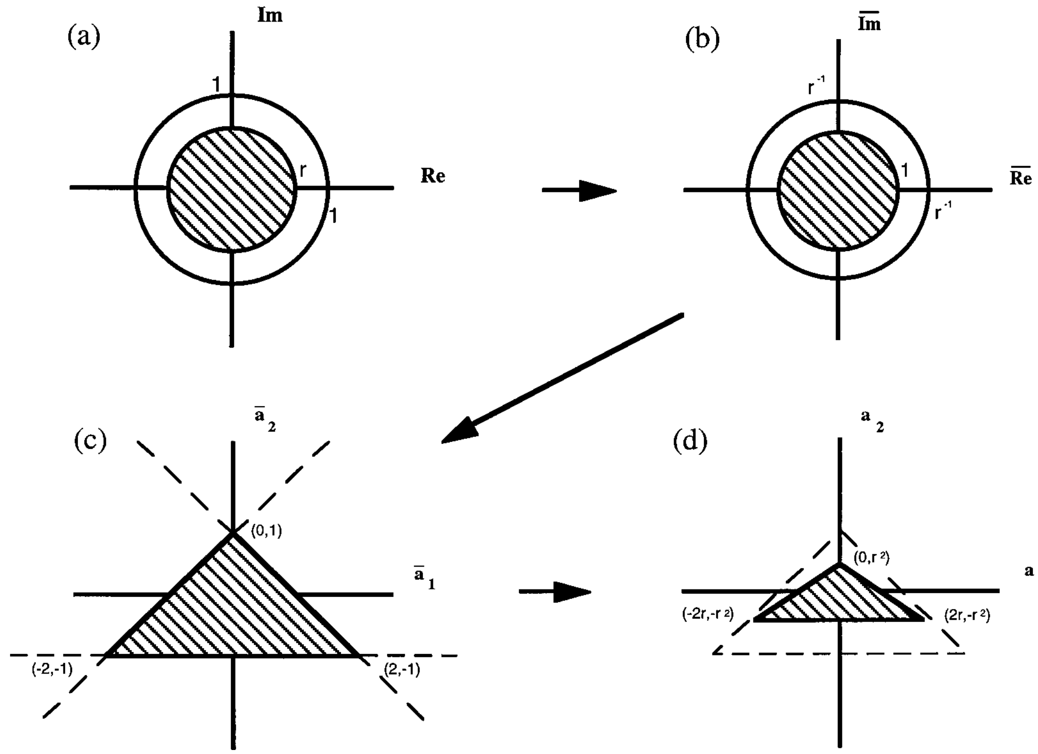

Figure 1. Settling time conditions for second-order systems: (a) poles restricted to circle of radius r (<1); (b) transformed system poles are now restricted to the unit disc; (c) stability conditions for the transformed system; (d) inverse-transformed stability conditions result in settling time conditions.

that impart integral action into the estimator and its coupled controller. Nevertheless, with care, a specified range may be advantageous. Several approaches are summarized in Table 1. They are briefly described below.

ARIMA- and CARIMA-type approaches estimate the parameters using incremental outputs and inputs $y_\Delta(t)$ and $u_\Delta(t)$, defined as

$$y_\Delta(t) \underline{\Delta}\, y(t) - y(t-d)$$

$$u_\Delta(t) \underline{\Delta}\, u(t) - u(t-d) \qquad (23)$$

where d is an arbitrary constant such as the system dead time. By using this method, offsets are implicitly removed. Because of this formulation, a known SSP or its range is not easily imposed on the identification (though two SSP's and their ranges can be imposed as constraints on steady-state gain using Table 1, entry 5). If instead, an offset term D were added to the right side of eq (2), linear inequality constraints for the range of D may be formed with knowledge of the sign of $1 - \Sigma a_i$. If the system is open-loop stable (and the model is constrained as such), then the sign will be positive, and the constraint in Table 1, entry 6 may be used. The floating identifier (FI), described in Timmons et al. (1991), is another, related approach. It high pass or band pass filters the outputs and inputs: offsets are removed for dynamics estimation and then added back for offset estimation. As before, if the model is constrained to be open-loop stable, then we arrive at a similar set of inequality constraints (entry 7, Table 1). However, the constraints now depend on recent input/output data so that noise may invalidate them; hence, one must use care when imposing them.

3.3. Open-Loop Settling Time. Settling time, the time required for a step response to settle to within a band of its final value (*e.g.*, ±1%), is commonly known. For second-order discrete time systems, a known maximum settling time τ_s approximately translates to all poles lying within a circle centered at the origin in the Z-plane with radius $r = \exp(-4.6T/\tau_s)$, where T is the sampling interval (Franklin and Powell, 1980, pp 100−103).

Consider two poles, p_1 and p_2, with magnitude less than r as in Figure 1a. Let $\bar{p}_1 = p_1/r$ and $\bar{p}_2 = p_2/r$. In this new coordinate system, the original circle of radius r corresponds to the unit circle (Figure 1b). We can directly apply the stability constraints of Table 1 (entry 2) to the transformed system (Figure 1c):

$$\bar{a}_1 + \bar{a}_2 \le 1$$

$$-\bar{a}_1 + \bar{a}_2 \le 1$$

$$-\bar{a}_2 \le 1 \qquad (24)$$

For second-order model structures similar to eq (2),

$$a_1 = p_1 + p_2$$

$$a_2 = -p_1 p_2 \qquad (25)$$

and

$$\bar{a}_1 = \bar{p}_1 + \bar{p}_2 = (p_1 + p_2)/r = a_1/r$$

$$\bar{a}_2 = -\bar{p}_1\bar{p}_2 = -p_1p_2/r^2 = a_2/r^2 \qquad (26)$$

As shown in Figure 1d, we obtain the constraints listed in entry 8 of Table 1.

3.4. Example of Constraints Built on Other Constraints: Open-Loop Zeros. The zeroes of minimum phase systems often lie within the unit circle in the Z-plane. For a second-order ARMAX-type model,

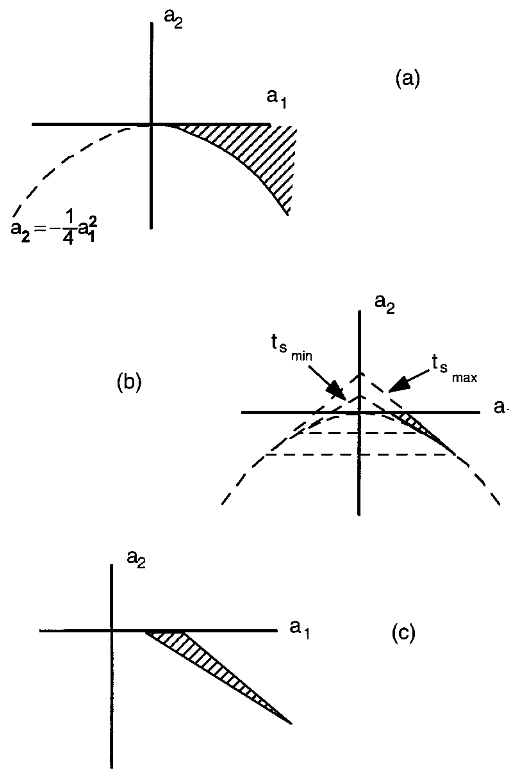

Figure 2. Example of combined conditions for a second-order system: (a) conditions for positive real poles; (b) conditions for minimum and maximum settling time superimposed on (a); (c) a closer look at the linear convex hull of (b).

the $B(q^{-1})$ polynomial (assuming a d-step delay) is given by

$$\mathbf{B}(q^{-1}) = b_1 q^{-d} + b_2 q^{-d-1} + b_3 q^{-d-2}$$
$$= b_1 q^{-d}\left(1 + \frac{b_2}{b_1}q^{-1} + \frac{b_3}{b_1}q^{-2}\right) \tag{27}$$

In this form, we can utilize the constraints for settling time (entry 8, Table 1) with a_1 and a_2 replaced by $-b_2/b_1$ and $-b_3/b_1$. The radius $r < 1$ now specifies the maximum radius of the zeros. For stable systems, we can convert these constraints to linear in θ form (entry 10, Table 1) by recognizing that b_1 has the same sign as G_{ss} (Timmons, 1992). A similar approach can be used for second-order minimum phase noise polynomials (entry 11, Table 1).

3.5. Example of Constraint Combinations. When constraints are combined, previously unusable (nonconvex) conditions may become usable (convex). For example, for second-order systems, minimum open-loop settling time cannot be used because it forms a set of nonconvex constraints. However, this potentially useful information can be used under certain common conditions, such as when the open-loop discrete time poles are positive and real (this condition results when continuous time systems with real poles are sampled (Kuo, 1963)). In this instance, the parameters are restricted to the fourth quadrant, above a concave parabola (Figure 2a). Now, constraints for *both* minimum *and* maximum settling times can be superimposed (Figure 2b), producing a greatly reduced feasible space whose linear convex hull (Figure 2c) is listed in Table 1 (entry 9).

4. Application Guidelines

In this section, the algorithms and several constraints developed in the previous two sections are applied to a first-order, time-varying benchmark model. The noise level, the presence or absence of constraints, the type of optimization, and the inclusion of a variable forgetting factor are used to illustrate some of the conditions for which parameter constraints improve or worsen adaptive control. These conditions are summarized by the six guidelines in Table 4.

4.1. Experiment Designs. 4.1.1. The Plant. The plant is a discrete time, linear, nonminimum phase ARMAX model. Except for an extra delay, it is identical to the benchmark model used in Clarke (1984), Timmons et al. (1991), and Voss et al. (1987). It was chosen because it is complex enough to produce several types of control anomalies, yet simple enough to allow for a clear, insightful analysis. It has a pole at 0.7, a nonminimum phase zero at -2, a delay of 2 steps, and a steady-state gain of 10:

$$y(t) = 0.7y(t-1) + u(t-2) + 2u(t-3) + e(t) \tag{28}$$

To challenge the controller, at simulation step $t = 105$, the steady-state gain is doubled (i.e., the second and third coefficients are doubled). For each of the demonstrations below, the plant is simulated three times with different levels of zero mean white gaussian noise: *no noise* ($e(t) \equiv 0$; SNR $= \infty$ dB), *small noise* ($\sigma_e^2 = 1$; SNR $= 31.1$ dB), and *large noise* ($\sigma_e^2 = 5$; SNR $= 24.1$ dB) (where the approximate SNR is given by 20 log(setpoint/σ_w) and $w(t) = 0.7w(t-1) + e(t)$).

4.1.2. Simulation Protocol. Except for a few instances when large noise is present, the following protocol ensures that no constraints are invoked up to the time of the challenge so that all controller configurations enter the challenge under identical conditions. In the protocol, a small step input (0.1 units for no noise, 0.3 units for small noise, and 1.0 unit for large noise) is administered for open-loop identification. The loop is closed after five steps ($t = 5$) with a setpoint of 50. Undisturbed control then continues for 100 steps. At this point ($t = 105$), the challenge begins (unknown to the controller), and control is continued for another 100 steps.

4.1.3. Control Law. For all simulations, control is implemented by a receding horizon predictive controller as in Voss et al. (1987), with a fixed prediction horizon of four steps. For the control calculation at each step, this algorithm assumes all future inputs will be equal to the current input. While automatic tuning of the control advance could be used to improve performance in several instances, this feature was disabled so that performance changes would be due solely to the identifier and its constraints.

4.1.4. Design Factors. Five designs were crossed with the three noise levels: (1) loose vs tight constraints; (2) complete vs incomplete constraint sets; (3) true vs untrue constraints; (4) optimal vs orthogonal projection; (5) momentarily increased data discounting vs fixed rate data discounting. The constraint set definitions are summarized in Table 2. Further details of each design are described below.

4.2. Results and Discussion. The plant outputs for each experiment design are plotted in Figure 3, and the mean squared control errors following the plant change ($t = 105-205$) are recorded in Table 3. Each guideline is presented in turn. In each, details of the experimental conditions are described first, followed by simulation results, a discussion, and the concluding rule or guideline from Table 4.

Table 2. Constraints Used in Simulations

	G_{ss}		z_0		a_1		b_1		b_2	
constraint set	min	max	min	max	min	max	min	max	min	max
actual parameter values	10	20	−2.0	−2.0	0.700	0.700	1.0	2.0	2.0	4.0
loose					−10	10	−10	10	−10	10
tight[a]	5	30	−2.5	−1.5	0.596	0.762	0.5	5.0	0.5	5.0
incomplete			−2.5	−1.5			0.0			
untrue[a]	5	10	−2.5	−1.5	0.596	0.762	0.5	5.0	0.5	5.0

[a] Minimum and maximum a_1 were chosen to achieve minimum and maximum settling times of 8.91 and 16.91 steps (±3 steps from the plant's settling time of 12.91 steps).

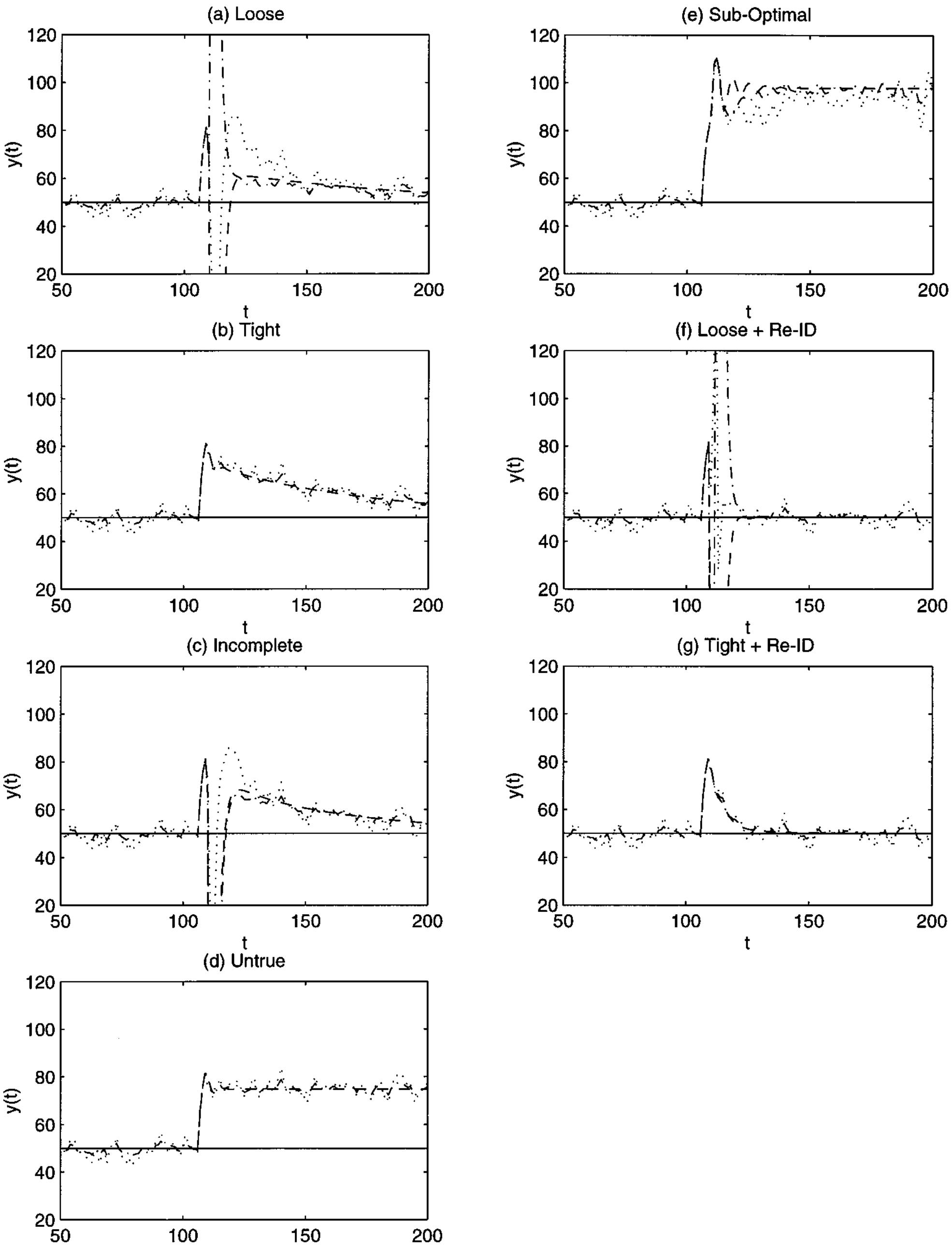

Figure 3. Closed-loop simulations for no noise (dash), small noise (dash-dot), and large noise (dot): (a) loose constraint set; (b) tight constraint set; (c) incomplete constraint set; (d) tight constraint set with a too small maximum steady-state gain; (e) tight constraint set with orthogonal projection; (f) loose constraint set with re-identification; (g) tight constraint set with re-identification.

4.2.1. Loose vs Tight Constraints. Two sets of constraints, "loose" and "tight", were used to demonstrate this point. Each set enforced minimum and maximum parameter values. In addition, since the tight set enforced open-loop stability (the bounds force the pole to lie within the unit circle), steady-state gain and a zero range were also included in this set (see Table 2).

In these simulations, the loose constraints were never invoked; the simulations were identical to unconstrained control (Figure 3a). The tight constraints, on the other hand, did become active (Figure 3b). For the no noise simulations, the mean squared control error (MSE) for the tight constraint set was more than 26 times lower than it was for the loose set. As the noise level increased, the unconstrained (loose) controller still

Table 3. Simulation Results: Mean Squared Control Error ($t = 105-205$)

conditions	no noise	small noise	large noise
loose	4746.92	3491.04	386.70
tight	179.24	191.18	210.68
incomplete	2519.12	3552.27	312.26
untrue	611.49	624.02	643.59
suboptimal tight	2226.18	2080.77	1736.14
loose + re-ident	3086.46	7116.13	210.39
tight + re-ident	44.76	47.81	56.13

exhibited instability at the plant change, though its overall MSE improved. The constrained (tight) controller only slightly worsened (Table 3). Eventually, for a large enough excitation, the *a priori* information would have provided little additional benefit.

Thus, the rule here is *constraints should be tight enough to be invoked*. That is, if we want a performance improvement using constraints, the bounds must be tight enough that they will become active at some time. A second rule can also be stated: *a priori information becomes increasingly important as excitation decreases*. This last rule is very important considering that most regulators try to achieve zero excitation.

4.2.2. Complete vs Incomplete Constraint Set. In this example, several of the constraints in the previous example were eliminated. When using minimum variance adaptive control, it is common to bound the sign of b_1 or fix it within a range of values so as to enforce stability (Åström and Wittenmark, 1973; Lozano-Leal and Collado, 1989). For extended and receding horizon control laws, however, b_1 is no longer the only important parameter (Egardt, 1980; Elliott, 1982; Lozano-Leal and Collado, 1989). Here, only the sign of b_1 and a range for the zero are enforced (see Table 2).

In this simulation, the reduced constraints did not eliminate the instability when the plant changed (Figure 3c), although for the no noise case they did decrease the input and output excursions compared to the unconstrained case. For the small noise case, the MSE with these constraints was worse than it was for no constraints (see Table 3).

In general, we have found that if only one part of a model is constrained, then the modeling error may be exaggerated in another. If the controller is sensitive to errors in the unconstrained part of the plant, then controller performance may suffer, sometimes more than if no constraints had been used. This raises the questions, should some types of information be ignored, or is there some way to complete the information set so that this problem does not occur? While these questions remain the topic of future research, constraints that enforce the global asymptotic stability conditions may suffice. For example, we have found that since receding and extended horizon control laws are sensitive to the location of plant zeros, constraints on the zeros often overcome this problem. Furthermore, the results from Eskinat et al. (1993) suggest that modeling the system around its crossover frequency might help generate the critical missing information. Also, as in Kwong et al. (1995), focusing on the control-relevant information should help address this issue. For now, however, the completeness issue remains a potentially serious drawback for life-critical applications. While it may be difficult to implement, the rule here is *the constraint set should be complete*.

4.2.3. True vs Untrue Constraints. In this example, the set of tight constraints in the first example was modified to impose too small of an upper limit on the steady-state gain. Although temporary instability was eliminated at the plant change, a constant steady-state error remained (Figure 3d). The sum of the squared errors would have continued to accumulate over time; eventually the MSE would have been larger for this case than for any of the previous examples. Thus, the rule here becomes *untrue constraints degrade performance*. While this rule may seem trivial and obvious, it is important because it is easily and unsuspectingly violated, especially when linear approximations are used for nonlinear plants.

4.2.4. Optimal vs Orthogonal Projection. In all previous simulations, the optimal (least squares) estimates were used. In this example (Figure 3e), the simulation experiment in Figure 3b (the tight set of constraints) is repeated with the orthogonal projector in place of the optimal projector. Here, just like the optimal projector (Figure 3b), the instability was eliminated. Unlike the optimal projector, however, *a steady-state error remained*. As in the example with the incorrect constraint (Figure 3d), the sum of the squared errors would have continued to increase linearly with time. Moreover, this steady-state error (Figure 3e) was larger at all noise levels than it was with the untrue constraint (Figure 3d). Application of the orthogonal projector to the other experiment designs of this study provided no additional information and so are not included here. However, in our experience with other models, the orthogonal projector occasionally impaired transient stability too.

The question might be asked, "Is performance worth trading for algorithm simplicity?" There are three responses. First, in this experiment, since the constraints do not form a set of simple bounds or a simple hypersphere, the orthogonal projector does not reduce to an attractive, simple algorithm, but remains as complex as the optimal projector. Second, simplicity could be regained if the constraints were relaxed to a convex hull made up of simple bounds or a hypersphere (potentially ignoring some useful *a priori* information), but then guideline 1 (Table 4) would be violated, potentially causing further degradation. Third, even if a simple algorithm could be used, the performance (as shown in this experiment) would still be unacceptable in life-critical applications. Thus, we conclude that the orthogonal projector is *not* a worthwhile substitute for the optimal projector, even when the constraint space is simple, and especially when a high price may be attached to poor performance. Interestingly, there is some evidence that the orthogonal projector may actually be a better choice when the **P**-matrix is poorly conditioned, which would be consistent with the results in Chia et al. (1991). This is the topic of future research. With this caveat in mind, the rule here is *use optimal projection*.

4.2.5. Momentarily Increased Data Discounting vs Fixed Rate Data Discounting. After a plant change, data from the new plant must compete with data from the old during system identification. Therefore, it is common to discard old data when a change has been detected (Goodwin and Sin, 1984). The data

Table 4. Application Guidelines

1. Constraints should be tight enought to be invoked
2. *A priori* information decreases excitation needs
3. The constraint set should be complete
4. Untrue constraints degrade performance
5. Use optimal projection
6. Re-identify after suspected changes

can be discarded by either resetting the **P**-matrix (blanking past data) or increasing the speed of adaptation by lowering the estimator's exponential forgetting factor (rapidly discounting past data). In this next example, the past data is discounted by reducing the forgetting factor to 0.5 for five steps, then increasing it to 0.75 for eight steps, and finally returning it to its original value of 0.98. This approach is applied to the experiment designs in Figures 3a (loose) and 3b (tight). Results are plotted in Figures 3f and 3g, respectively.

Re-identification after the plant change when using the loose set of constraints (Figure 3f) did not eliminate the plant change instability. In fact, when small noise was present, the input and output excursions were nearly twice as large as when re-identification was not used. However, the control error did go to zero immediately after the instability, unlike Figure 3a.

When using the tight set of constraints, control remained stable and the control error went to zero rapidly (Figure 3g). For the no noise case, the MSE was reduced by more than a factor of 100 compared to the original loose constraint example in Figure 3a. Furthermore, with the tight constraint set, the MSE increased only slightly as the noise level increased (see Table 3). Note that this trend still supports guideline 2. More importantly, note that with appropriate parameter constraints, the adaptation gain can seemingly be increased without sacrificing stability. This suggests the final rule, *re-identify after suspected changes*.

5. Control of Second-Order Blood Pressure Armax Model

In this section the guidelines are applied to the control of a simulated mean arterial blood pressure (MAP) model using a vasodilator (sodium nitroprusside, SNP).

5.1. The ARMAX Model. In modeling the MAP/SNP system, the $y(t)$ in (2) represents the measured MAP at time t, while $u(t)$ corresponds to the SNP infusion rate at time t. To form a time-varying ARMAX model, a *before* and an *after* model were defined. The *before* model was obtained by probing a pentobarbital-anesthetized dog with SNP; the *after* model was obtained from the same dog following a disturbance caused by an unknown event which resulted in a 3-fold increase in the steady-state gain and a change in the base line blood pressure (Timmons, 1992). The disturbance was implemented by ramping the exogenous input parameters (the b's) between the *before* (−0.072, 0.036) and *after* (−0.216, 0.108) models, while maintaining the autoregressive terms (the a's) fixed (−0.7, 0.06). To obtain a resting MAP, base line levels of 150 mmHg for the *before* model and 111 mmHg for the *after* model were added. In addition, a zero mean white Gaussian noise with a variance of 0.25 $mmHg^2$ was added to the process as in (28).

5.2. Controller Setup and Simulation. For system identification, a recursive least squares algorithm with a variable forgetting factor was used. The identified model assumed a 20 s sampling interval, two autoregressive terms (a's), and three exogenous input terms (b's) with an input/output delay of one sample. For the constrained identification, minimum and maximum limits were imposed on the steady-state gain (−1.0, −0.04), the settling time (0.4, 0.7), and the magnitude of the zeros (0.7) (see Table 1, entries 3, 9, and 10). A receding horizon predictive controller with a prediction horizon of six samples (Voss, 1988) was used to generate the SNP infusion rates.

Two simulations were performed to compare adaptive control with and without constraints. After 5 min of probing, control was started with a target MAP of 100 mmHg, representing a drop of 50 mmHg from the base line. After an additional 15 min of control, the transition from the *before* model to the *after* model began unknown to the controller ($t = 20$ min). The transition was completed in 10 samples (3.3 min), and control was allowed to continue for another 16 min. On the basis of the definition of SNR given earlier (with $w(t) = 0.7w(t-1) - 0.06w(t-2) + e(t)$), going into the disturbance it was 37.5 dB, while coming out it was 24.4 dB.

5.3. Results and Discussion. Both simulations are plotted in Figure 4. Results show that large oscillations occur during the model transition for the unconstrained controller while only small oscillations occur for the constrained case. The mean squared error (MSE) calculated for times greater than 10 min is dramatically reduced when using the constrained controller (MSE = 36.67) as opposed to the unconstrained controller (MSE = 496.36). The maximum absolute error when using the unconstrained controller is 100 mmHg, clearly an unacceptable value for blood pressure regulation in a clinical setting. In contrast, the constrained controller was able to maintain MAP at acceptable levels throughout with a maximum absolute error of 33.5 mmHg. Furthermore the total amount of drug infused was less when the constrained strategy was used (28.5 mg when unconstrained vs 25.9 mg when constrained), and infusion rates were smoother. Several other applications with similar results may be found in Timmons (1992).

6. General Discussion

While specific cases were discussed above, this section discusses several general issues relating to the use and extension of parameter-constrained adaptive control. Our findings are also contrasted with the results of other studies.

Interestingly, there appears to be a certain degree of skepticism regarding the use of *a priori* information. Some investigators consider its use antithetic to the spirit of adaptive control, since they claim that the goal of adaptive control is to use as little *a priori* information as possible (see for example Giri et al., 1992). While this may be a worthwhile goal, as a practical matter, some information is almost always available. Since, as shown in Figure 3, *a priori* information *can* improve controller performance, often dramatically, it would seem prudent to use as much information as possible.

Others investigators conclude that projectors are not necessary because momentarily unstable parameter settings are self-correcting (see for example Kosut, 1987). While this statement may well be true, again as a practical matter, momentary instability may be undesirable. One problem with allowing momentary instability is that the adaptation gain becomes small, leaving the adaptive controller unable to compensate for later changes. Another problem is that instability may be dangerous or life-threatening in some situations (*e.g.*, medical systems for human patients, *cf.* Figure 4). Finally, if rapid learning is desired, there are less detrimental techniques that can be used, such as temporarily lowering the forgetting factor. The controller in Figure 3g demonstrates the efficacy of this latter technique. One could aid the process further by adding a small perturbation to the input signal, as well as by including an automated, continuous adaptation of the forgetting factor as described in (Goodwin and Sin, 1984,

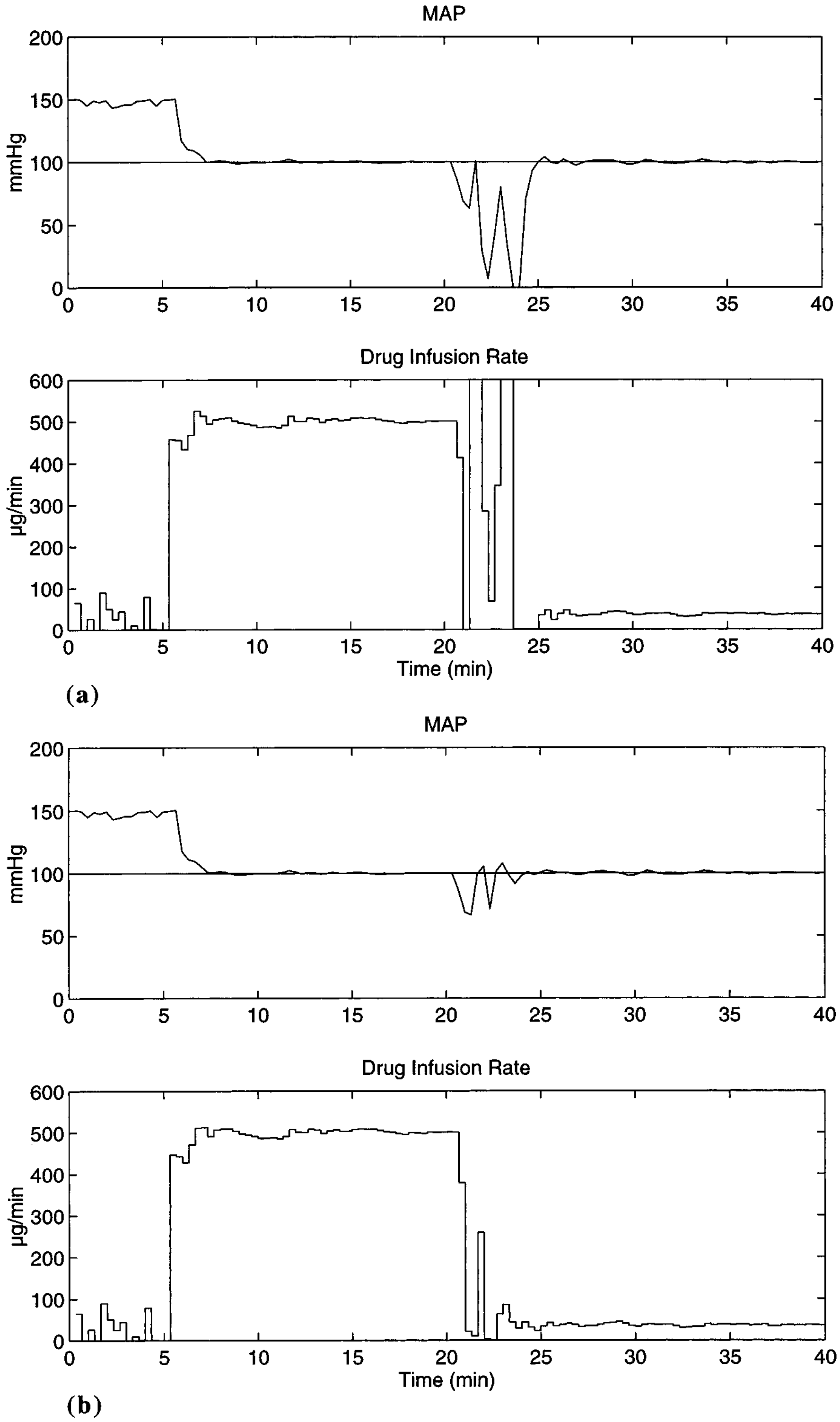

Figure 4. Second-order ARMAX model simulations: (a) unconstrained case; (b) constrained case.

p 227). Indeed, the adaptive forgetting factor proved to be the most important adjuvant in the blood pressure application in Figure 4.

Probably because of the apparent difficulty of coding an optimal projector, the simplified orthogonal projector is used more commonly. For example, many of the robust stability schemes that depend on parameter bounding do so using an orthogonal projector, as in Ydstie (1989), Naik et al. (1992), and Praly and Kumar (1989). Yet clearly, as demonstrated by Figure 3e, the orthogonal projector does not necessarily lead to improved control, although it did reduce the magnitude of the transient excursions. We speculate that these findings also apply to variants of the orthogonal projector, such as the σ-projector, so that the extra effort required to code for the optimal projector is worthwhile. Furthermore, many reliable quadratic programming packages are available at low cost, substantially reducing the effort needed to construct an optimal projector.

A more serious concern, however, is that the development and conversion of *a priori* knowledge into suitable linear constraints may be problematic for high-order and nonlinear systems. For example, the Jury stability criteria become nonlinear above second order. For these types of problems, a nonlinear (and possibly nonconvex) program solver would be required. As such, there would be no guarantee that a unique or global solution could be found in a reasonable amount of time. On the other hand, these solvers would allow the inclusion of even more *a priori* information, such as the nonconvex constraints arising from control and trajectory admissibility conditions, nonminimum phase plant zeros, and generalized minimum and maximum plant settling times. A treatment of one such optimizer may be found in Luyben and Floudas (1994), where it is used to impose output controllability on several process designs.

Of course, the constraints developed here could still be applied to high-order MIMO systems which can be decoupled, so that at least low-order, local information can be imposed. Or, using a different approach, it may be possible to reduce the required model order by using only control-relevant information, as in Rivera et al. (1987). Also, many classical control designs consider only the two most dominant poles regardless of the actual "best" order of the system. Hence, for many high-order control applications, the constraints in Table 1 may still be practical and directly useful.

Soft, moving, or adaptive constraints can be used to impose important and useful information about the rate at which a plant is expected to change or drift over time. Several of the constraints in Table 1 lend themselves directly to this type of application. Examples include simple bounds around a parameter, a gain, or a pole radius that changes as a function of the recent mean of the input and output signals. This approach further allows the possibility of using the constraints themselves to detect system parameter jumps. For example, a sudden increased cost associated with a soft constraint could be used to trigger re-identification (guideline 6) or at least to alert a supervisory system of a potential change in the plant. If the parameter jumps are exclusive, then two sets of constraints could be used to define each feasible space. One or the other should then have a high cost associated with its imposition.

7. Conclusion

In summary, an optimal constrained identifier has been derived as a semirecursive quadratic program, making it attractive for real-time applications. A reasonably fast, novel quadratic program solver has been included for this purpose in the Appendix. Linear constraints from commonly available information have been developed (Table 1), and application guidelines have been demonstrated and listed in Table 4. Of the experiments shown here, most surprising was the poor performance exhibited by the orthogonal projector, suggesting that it should not be used in low-tolerance applications. Additionally, these experiments clearly showed that *a priori* information reduced the need for system excitation, which is consistent with results from other related areas. Constraint set completeness remains a potentially thorny issue, though it possibly may be solved using stability analysis, crossover frequency analysis, and other control-relevant information. Finally, it was also shown that optimal, parameter-constrained identification can lead to dramatically improved controller performance when combined with selective re-identification. The potential benefit of parameter constraints for adaptive control has thus been shown to be significant, though additional work is needed to extend this idea to high-order and nonlinear systems, as well as to generate formal mathematical proofs of the guidelines.

Acknowledgment

Parts of this work were supported by NSF Grant BCS-8908713, Eli Lilly and Company, and The Whitaker Foundation. The authors thank Drs. G. C. Giakos and T. T. Hartley for their helpful comments on the manuscript.

Appendix: An Enhanced CLP Algorithm

The CLP Tableau: Definitions and Structure

Given a real n-vector $\mathbf{y}$ and a real positive definite or positive semidefinite (but not necessarily symmetric) $n \times n$ matrix $\mathbf{Q}$, find vectors $\mathbf{x}$ and $\mathbf{z}$ satisfying the conditions

$$\mathbf{x} = \mathbf{Q}\mathbf{z} + \mathbf{y}$$

$$\mathbf{x} \geq 0, \mathbf{z} \geq 0$$

$$\mathbf{x}^{\mathrm{T}}\mathbf{z} = 0 \tag{29}$$

The complement of x_i is z_i and vice versa. The goal of CLP is to permute the equations $\mathbf{x} = \mathbf{Q}\mathbf{z} + \mathbf{y}$ using simplex pivots until the complementary and nonnegativity conditions are met. *Basic* and *nonbasic* variables are defined as for the simplex method: left-hand variables are basic and right-hand variables are nonbasic. For an arbitrary tableau, let $\bar{\mathbf{x}}$ and $\bar{\mathbf{z}}$ define the basic and nonbasic variables, respectively. For each pivot, we define the *entering variable* as the variable that becomes basic and the *exiting* (or *blocking*) *variable* as the variable that becomes nonbasic. When the complementary conditions are satisfied ($z_i x_i = 0$ for all i), the tableau is *complementary*, and the vectors $\bar{\mathbf{x}}$ and $\bar{\mathbf{z}}$ form a *complementary solution*. If the basic variables are additionally nonnegative, then the complementary solution is optimal. If one of the $x_i z_i \neq 0$, then the tableau is *almost complementary*, and the vectors $\bar{\mathbf{x}}$ and $\bar{\mathbf{z}}$ form an *almost complementary solution*.

During the tableau permutations, nonnegative variables are not allowed to become negative. This restriction may cause the formation of an almost complementary tableau. A *major cycle* is the sequence of pivots needed to restore (or maintain) the complementary condition. Each major cycle begins with the selection of a *target* variable from the basic variables and ends when this variable becomes nonbasic. Within a major cycle, the entering variable is not necessarily the complement of the target variable, and the exiting variable is not necessarily the target variable.

Given (29), a complementary *starting tableau* can be initialized with $\bar{\mathbf{x}} = \mathbf{x}$ (=$\mathbf{y}$) and $\mathbf{z} = \mathbf{z}$ (=0):

	z_1	...	z_n	
x_1	Q_{11}	...	Q_{1n}	y_1
⋮	⋮	⋮	⋮	⋮
x_n	Q_{1n}	...	Q_{nn}	y_n
F	$-y_1$	...	$-y_n$	0

where the bottom row entries are the relative cost coefficients for a quadratic program of the form (16) (without equality constraints). For a complementary solution, the value of the quadratic projection cost function (19) is $^1/_2F$ (where F is the lower right entry in the tableau). For an almost complementary solution, the value of the cost function is $^1/_2(F + \bar{\mathbf{x}}^T\bar{\mathbf{z}})$ (Timmons, 1992).

The Enhanced CLP Algorithm

Step 0. *Initialize tableau.*

Step 1. *Start major cycle.*

$\mathbf{S}$ = the set $\{x_i \mid x_i < 0, i = 1, ..., n\}$ /*$\mathbf{S}$ is the set of potential target variables*/

Step 2. *Find target variable.*

If ($\bar{\mathbf{x}} \geq 0$) then
  Return with solution /*Solution is optimal*/
Else if ($\mathbf{S} = \varnothing$) then
  Return with error /*No solution*/
Else
  Target = $\bar{x}_i$, where $\bar{x}_i \in \mathbf{S}$ and $\bar{y}_i^2/\bar{Q}_{ii}$ is maximum
Endif

Step 3. *Find entering variable.* /**i.e.*, find the pivot column*/

If (tableau is complementary) then
  Entering variable = complement of target
Else
  Entering variable = complement of previous blocking variable
Endif

Step 4. *Find exiting variable.* /**i.e.*, find the pivot row*/

/*A block occurs when, upon increasing the entering variable, (a) the target variable increases to zero or (b) another basic variable decreases to zero*/
If (blocked) then
  blocking variable = first variable to block
    /*Ties go to the target variable, if involved. Otherwise, resolve degeneracies using Charnes's method (*e.g.*, see Luenberger, 1984)*/
Else if (tableau is complementary) then
  Remove the target from $\mathbf{S}$ /*No block, so reject target*/
  Goto to Step 2
Else
  Return with error /*No solution*/
Endif

Step 5. *Pivot.*

Exchange the entering variable for the blocking variable

Step 6. *Loop back.*

If (tableau is complementary) then
  Goto Step 1 /*The major cycle ends*/
Else
  Goto Step 3 /*The major cycle continues*/
Endif

Our modifications to the original algorithm are (a) the least distance target selection rule in Step 2 and (b) the target rejection rule in Step 4. See Timmons (1992) for details.

Equality Constraints

For numerical robustness, any equality constraints that are present should be included in the tableau (as in eq 16) (see Golub and Saunders, 1970). Their Lagrange multipliers and constraint errors should be made basic and nonbasic, respectively (thus driving the constraint errors to zero), and then removed from the list of possible pivots.

Literature Cited

Åström, K. J.; Wittenmark, B. On self-tuning regulators. *Automatica* **1973**, *9*, 185−199.

Bai, E.-W.; Sastry, S. S. Discrete Time Adaptive Control Utilizing Prior Information. *IEEE Trans. Autom. Control* **1986**, *31*, 779−782.

Chia, T. L.; Chow, P.-C.; Chizeck, H. J. Recursive Parameter Identification of Constrained Systems: An Application to Electrically Stimulated Muscle. *IEEE Trans. Biomed. Eng.* **1991**, *38*, 429−442.

Clarke, D. W. Self-tuning Control of Nonminimum-phase Systems. *Automatica* **1984**, *20*, 501−517.

Clary, J. P.; Franklin, G. F. Self-tuning Control with *A Priori* Plant Knowledge. Proceedings of the 23rd IEEE Conference on Decision and Control, Las Vegas, NV, 1984; IEEE: New York, 1984; pp 369−374.

Dantzig, G. B.; Cottle, R. W. Positive (Semi-) Definite Programming. *Nonlinear Programming*; Abadie, J., Ed.; North-Holland Publishing Co.: Amsterdam (John Wiley and Sons, Inc.: New York), 1967; pp 55−73.

Dasgupta, S.; Anderson, B. D. O.; Kaye, R. J. Output Error Identification of Partially Known Systems. *Int. J. Control* **1986**, *43*, 177−191.

Egardt, B. Stability Analysis of Discrete-Time Adaptive Control Schemes. *IEEE Trans. Autom. Control* **1980**, *25*, 710−716.

Elliott, H. Hybrid Adaptive Control of Continuous Time Systems. *IEEE Trans. Autom. Control* **1982**, *27*, 419−426.

Eskinat, E.; Johnson, S. H.; Luyben, W. L. Use of Auxiliary Information in System Identification. *Ind. Eng. Chem. Res.* **1993**, *32*, 1981−1992.

Fletcher, R. *Practical Methods of Optimization*, 2nd ed. John Wiley & Sons: New York, 1987.

Franklin, G. F.; Powell, J. D. *Digital Control of Dynamic Systems*. Addison-Wesley: Reading, MA, 1980.

Giri, F.; M'Saad, M.; Dugard, L.; Dion, J.-M. Robust Adaptive Regulation with Minimal Prior Knowledge. *IEEE Trans. Autom. Control* **1992**, *37*, 305−315.

Golub, G. H.; Saunders, M. A. Linear Least Squares and Quadratic Programming. *Integer and Nonlinear Programming*; Abadie, J., Ed.; North-Holland Publishing Co: Amsterdam (American Elsevier Publishing Co.: New York), 1970; Vol. 10, pp 229−256.

Goodwin, G. C.; Payne, R. L. *Dynamic System Identification: Experiment Design and Data Analysis*; Academic Press: New York, 1977.

Goodwin, G. C.; Sin, K. W. *Adaptive Filtering, Prediction, and Control*; Prentice-Hall: Englewood Cliffs, NJ, 1984.

Kosut, R. L. On The Use of The Method of Averaging For The Stability Analysis of Adaptive Linear Control Systems. Proceedings of the 26th IEEE Conference on Decision and Control, Los Angeles, CA, 1987; IEEE: New York, 1987; pp 366−371.

Kuo, B. C. *Analysis and Synthesis of Sampled-Data Control Systems*; Prentice-Hall: Englewood Cliffs, NJ, 1963.

Kwong, G. K.; Kwok, K. E.; Finegan, B. A.; Shah, S. L. Clinical Evaluation of Long Range Adaptive Control for Mean Arterial Blood Pressure Regulation. *Proc. Am. Control Conf.* **1995**, *1*, 786−790.

Li, Z. Discrete-Time Adaptive Control of Systems Consisting of Known and Unknown Subsystems. *IEEE Trans. Autom. Control* **1989**, *34*, 375−379.

Lozano-Leal, R.; Collado, J. Adaptive Control for Systems With Bounded Disturbances. *IEEE Trans. Autom. Control* **1989**, *34*, 225−228.

Luenberger, D. G. *Linear and Nonlinear Programming*, 2nd ed.; Addison-Wesley: Reading, MA, 1984.

Luyben, M. L.; Floudas, C. A. Analyzing the interaction of design and control-1. A multiobjective framework and application to binary distillation synthesis. *Comput. Chem. Eng.* **1994**, *18*, 933−969.

Naik, S. M.; Kumar, P. R.; Ydstie, B. E. Robust Continuous-Time Adaptive Control by Parameter Projection. *IEEE Trans. Autom. Control* **1992**, *37*, 182−197.

Ortega, R.; Tang, Y. Robustness of Adaptive Controllers—A Survey. *Automatica* **1989**, *25*, 651−677.

Praly, L.; Lin, S.-F.; Kumar, P. R. A Robust Adaptive Minimum Variance Controller. *Siam J. Control Optim.* **1989**, *27*, 235−266.

Rivera, D. L.; Morari, M. Control Relevant Model Reduction Problems for SISO H2, H Infinity, and μ Controller Synthesis. *Int. J. Control* **1987**, *46*, 505−527.

Timmons, W. D. *Constrained Identification for Adaptive Control: Application to Biomedical Systems*, Ph.D. Dissertation, Department of Biomedical Engineering, Case Western Reserve University, Cleveland, OH, May, 1992.

Timmons, W. D.; Katona, P. G.; Chizeck, H. J. Adaptive Control is Enhanced by Background Estimation. *IEEE Trans. Biomed. Eng.* **1991**, *38*, 273−279.

Voss, G. I.; Chizeck, H. J.; Katona, P. G. Regarding Self-tuning Controllers for Nonminimum Phase Plants. *Automatica* **1987**, *23*, 405−408.

Voss, G. I.; Chizeck, H. J.; Katona, P. G. Self-Tuning Controller for Drug Delivery Systems. *IEEE Trans. Biomed. Eng.* **1988**, *47*, 1507−1520.

Ydstie, B. E. Stability of Discrete Model Reference Control-Revisited. *Syst. Control Lett.* **1989**, *13*, 429−438.

Zheng, L. Discrete-Time Adaptive-Control of Systems Consisting of Known and Unknown Subsystems. *IEEE Trans. Autom. Control* **1989**, *34*, 375−379.

Received for review October 15, 1996
Revised manuscript received June 12, 1997
Accepted June 16, 1997[⊗]

IE9606597

[⊗] Abstract published in *Advance ACS Abstracts,* August 15, 1997.