

An Analog “Neural Net” Based Suboptimal Controller for Constrained Discrete-time Linear Systems

MARIO SZNAIER† and MARK J. DAMBORG‡

Key Words—Analog computer control; discrete-time systems; feedback control; multivariable control systems; neural nets; on-line operations; stability; suboptimal control.

Abstract—A large class of problems frequently encountered in practice involves the control of linear time invariant systems with states and controls restricted to closed convex regions of their respective spaces. In spite of the significance of this problem, to date it has not been solved satisfactorily except in some restricted cases. In this paper we propose a suboptimal feedback control algorithm based upon on-line optimization during the sampling interval. Theoretical results are presented showing that our approach yields asymptotically stable systems. Finally an implementation of the control algorithm using an analog circuit is discussed. This implementation provides an alternative to the use of digital computers in the feedback loop that offers advantages in terms of cost and reliability. We believe that it may prove to be specially valuable when the time available for computations is limited. A 5th-order model of a F-100 jet engine is used as an example application of the controller.

1. Introduction

A LARGE CLASS of problems frequently encountered in practice involves the control of linear, time-invariant, systems with states and controls restricted to closed convex regions of the respective spaces. For example, the constraints may represent physical limitations of the system or they may originate in the process of modeling. The latter occurs when a complex system is represented by a linear model obtained through a linearization around a nominal trajectory. In this case the states are constrained to remain in a neighborhood of the nominal trajectory where the representation is sufficiently accurate.

In spite of the significance of this problem, it has not been solved satisfactorily. There have been several recent attempts to design linear and nonlinear feedback controllers for constrained systems. However, most of the design procedures available are severely restricted in their domain of application, as discussed in Sznaier and Damborg (1990). Another approach casts the problem as an optimization problem and then uses the vast machinery available for optimization to solve it. This approach is appealing because it guarantees an acceptable system response in the sense of some performance index. However, in most cases the control law generated is an open-loop control law that must be recalculated entirely, with considerable computational effort,

if the system is disturbed. Recently, we proposed a suboptimal controller (Sznaier and Damborg, 1987, 1989, 1990), based upon on-line optimization, that solves this problem and we have shown that the resulting closed-loop system is asymptotically stable even in the face of computing time restrictions. At this stage, we are conducting tests of such controllers in different systems. However, the stability results hinge on the availability of a certain minimum time for computation, and this minimum depends on the dimension of the system.

In this paper we propose a feedback controller, based upon on-line optimization, for the suboptimal minimum-time control of a class of systems. The proposed controller can be implemented using an analog (“neural”) circuit to carry-out the on-line minimization. This circuit has the potential to perform the minimization very fast, thus offering an interesting alternative, that provides advantages in terms of cost and reliability, to a digital computer-based implementation.

2. Statement of the problem

We will consider linear, time-invariant, controllable discrete-time systems modeled by the difference equation:

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k, \quad k = 0, 1, \dots \quad (1)$$

with initial condition \mathbf{x}_0 , and the constraints

$$\mathbf{u}_k \in \Omega \subset R^m, \quad \mathbf{x}_k \in \mathcal{G} \subset R^n \quad (2)$$

where Ω and \mathcal{G} are compact convex polyhedrons containing the origin in their interior and defined by a set of inequalities of the form:

$$\mathcal{G} = \{x : |Gx| \leq \gamma\} \\ \Omega = \{u : Wu \leq \omega\}$$

where $\gamma \in R^p$, $\gamma_i > 0$, G is a $p \times n$ matrix such that $\text{rank}(G) = n$, W is a $q \times m$ matrix, $\omega \in R^q$, $_-$ denotes a vector quantity and where the $| \cdot |$ and the inequalities should be interpreted on a component by component sense. An additional hypothesis on the region \mathcal{G} , a constraint qualification hypothesis, will be introduced in the next section. The objective is to find a sequence of admissible controls, $\mathbf{u}_k[\mathbf{x}_k]$, that brings the system to the origin in minimum time. The notation $\mathbf{u}_k[\mathbf{x}_k]$ emphasizes the fact that a closed-loop solution is desired. We will call such a sequence a “global optimum”. This problem will be denoted as problem (P) and throughout this paper we will assume that (P) is feasible for any initial condition in \mathcal{G} . (In Section 3 we will show how this assumption can be checked).

3. Definitions and theoretical results

In this section we introduce the definitions and theoretical results required to support our controller. First we will introduce a norm in the set \mathcal{G} and show that there exists a control sequence such that this norm defines a Lyapunov

* Received 2 December 1989; revised 29 July 1990; revised 28 April 1991, received in final form 1 May 1991. The original version of this paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form by Associate Editor T. Başar under the direction of Editor A. P. Sage.

† Electrical Engineering Department, University of Central Florida, Orlando, FL 32816-0450, U.S.A. Author to whom all correspondence should be addressed.

‡ Electrical Engineering Dept., FT-10 University of Washington, Seattle, WA 98195, U.S.A.

function for the system. Then, we will show that a linear programming problem can be solved using a penalty function method with proper choice of the penalties. This result will be used in the next section to find the control sequence mentioned above.

Lemma 1. Let $v(x)$ be defined as:

$$v(x) = \max_{0 \leq i \leq p} \left\{ \frac{|Gx|_i}{\gamma_i} \right\}. \quad (3)$$

Then $v(\cdot)$ is a norm in \mathcal{G} . We will denote this norm as $\|\cdot\|_v$.

The proof of the lemma follows from the definitions of \mathcal{G} and $v(\cdot)$. It also follows that $\|x\|_v \leq 1$ for all $x \in \mathcal{G}$ and $\|x\|_v = 1$ for x in the boundary of \mathcal{G} , $\partial\mathcal{G}$.

Constraint qualification hypothesis. Throughout this paper we will consider systems such that:

$$\min_{u \in \Omega} \{ \|Ax + Bu\|_v \} < 1 \quad \forall x \in \partial\mathcal{G}. \quad (4)$$

Condition (4) implies that for any initial condition on the boundary of the admissible region there exists a control that brings the system to its interior. Since the problem was assumed to be feasible in \mathcal{G} , the only effect of the additional constraints is to rule out the possibility of the system staying on the boundary for consecutive sampling instants. For the class of systems that we are considering in this paper, condition (4) can be reduced to a system of linear inequalities and checked using linear programming. Note that the convexity of Ω implies that satisfaction of (4) is a sufficient condition for feasibility of (P) in \mathcal{G} .

Theorem 1. Consider problem (P) with the additional constraint (4). Then, there exist a control sequence $u = \{u_0, \dots\}$, $u_k \in \Omega$ such that

$$\|x_{k+1}\|_v < \|x_k\|_v, \quad k = 0, 1, \dots, \forall x_0 \in \mathcal{G}. \quad (5)$$

Proof. The proof follows by noting that \mathcal{G} satisfies the constraint qualification conditions [equations (6) and (7)] in Sznajder and Damborg (1990) and therefore the theorem reduces to a special case of Theorem 2 therein.

Corollary 1. Consider a point x in \mathcal{G} , $x \neq 0$ and let:

$$\rho(x) = \min_{u \in \Omega} \left\{ \frac{\|Ax + Bu\|_v}{\|x\|_v} \right\}. \quad (6)$$

Then $\rho(x) < 1$.

The following theorem shows that a linear programming problem can be solved using an exact penalty function method. This is the basic result exploited in the implementation of our algorithm using an analog circuit. The theorem is originally due to Pyne (1956), although in his derivation it was assumed that at most only one constraint was violated at any given time. In the following proof this limitation is removed.

Theorem 2. Consider the following optimization problems:

$$\min_{x \in \mathcal{G}} \{ L(x) = c'x \} \quad (7)$$

where \mathcal{G} is a convex polyhedron defined by

$$\mathcal{G} = \{x: Gx \leq \gamma\}$$

and where $\gamma \in R^p$, G is an $p \times n$ matrix and $'$ denotes transpose.

$$\min_x \{ H(x) = c'x + \delta'(Gx - \gamma) \} \quad (8)$$

where the components δ_i of δ are defined by:

$$\begin{aligned} \delta_i &= K\theta[(Gx - \gamma)_i], \quad K > 0 \\ \theta(x) &= \begin{cases} 0, & \text{if } x \leq 0; \\ 1, & \text{otherwise.} \end{cases} \end{aligned} \quad (9)$$

Then, there exist K_o such that for $K \geq K_o$, (7) and (8) have the same solution. Note that the penalties (δ_i) in (8) are essentially the Lagrange multipliers for the Kuhn and Tucker conditions (Luenberger, 1969).

Proof. Consider a point x^o outside the region \mathcal{G} . It follows that at least one of the components of δ is strictly positive. Let $j(x^o) = \{i_1, i_2, \dots, i_m\}$ be the set of indices such that $\delta_{i_k}|_{x^o} > 0$ and denote the j th row of G as N_j . (Note that N_j is the outward pointing normal to the hyperplane that bounds the j th constraint.) Then $H(x)$ in (8) can be rewritten as

$$H(x) = c'x + \sum_{j=1}^m \delta_j(N_j'x - \gamma_j). \quad (10)$$

Thus we have

$$\begin{aligned} \nabla H &= c + \sum_{j=0}^p \delta_j N_j \\ &= c + \sum_{j \in I} KN_j. \end{aligned} \quad (11)$$

From the convexity of \mathcal{G} it follows that if $\delta_{i_1}, \delta_{i_2}, \dots, \delta_{i_m}$ are nonzero simultaneously then $z = N_{i_1} + N_{i_2} + \dots + N_{i_m} \neq 0$. Hence from (11) it follows that:

$$\begin{aligned} -\nabla H'z &= -c'z - \sum_{j \in I} KN_j'z \\ &= -c'z - Kz'z. \end{aligned} \quad (12)$$

Therefore, if K is selected such that:

$$K \geq K_o = \max_{i \in I} \left\{ \frac{c'z}{\|z\|_2^2} \right\} \quad (13)$$

where $I = \{j\}$ (i.e. the set of indices corresponding to all possible combinations of constraints that are violated simultaneously), then we have that, for every point $x^o \notin \mathcal{G}$ there exist a vector z^o such that $-\nabla H'z^o < 0$. It follows that H can have a minimum only in the region \mathcal{G} (where $L(x) = H(x)$). \square

Note that the maximization in (13) is well defined since I is a finite set. Note also that condition (13) reduces to the condition obtained by Pyne (1956), for the special case when it is assumed that at most only one constraint is violated at a given time.

4. Proposed control algorithm

It is well known that problem (P) can be solved for formulating a sequence of linear programs. This approach was used by Gutman (1986) to design a feedback controller for a reservoir based upon the use of on-line optimization. However, this approach presents the difficulty that asymptotic stability of the closed-loop system is guaranteed only if there is enough computation time available to expand the sequence of linear programs until a complete solution to the problem is obtained. *Ad hoc* techniques (Gutman, 1986) have been proposed to obtain a control law when there is not enough time available to compute a complete solution, but they can not guarantee asymptotic stability. Further, it is not guaranteed that the control law generated by these *ad hoc* techniques is a "sensible" control strategy, i.e. one that will steer the system in a convenient direction. In this paper we will concentrate in the stability aspects of the problem presenting a controller that is guaranteed to yield an asymptotically stable system while complying with all the restrictions. As a result, our controller is only suboptimal. However, we expect this controller to perform well compared with the true minimum time solution. Section 4 provides an example where this expectation is met.

From Theorem 1 and its corollary, it follows that for any point $x \in \mathcal{G}$ there exists a control $u(x) \in \Omega$ such that the ratio ρ of the norm of the state resulting from applying the control $u(x)$ at x to the norm of x is strictly smaller than 1. Consider now the control law $u_k^*(x_k)$ obtained by minimizing the value of ρ at each stage, i.e.

$$u_k^* = \operatorname{argmin}_{u \in \Omega} \{ \rho(x_k) \} \quad (14)$$

where

$$\rho(x) = \frac{\|Ax + Bu\|_v}{\|x\|_v} = \frac{\|x_{k+1}\|_v}{\|x_k\|_v} \quad (15)$$

We will call the sequence μ_k^* a "local optimum" and use it as our control law. Hence we have the following algorithm.

Control algorithm. Let k be the current time instant, x_k the current state of the system and u_k^* the control law computed during the last sampling interval. Then

Repeat until the origin is reached:

Begin:

(1) use u_k^* as control for the next interval

(2) compute $\bar{x} = Ax_k + Bu_k^*$

(3) find u_{k+1}^* by solving (14)

End.

Note that when using u^* , $\|x\|_v$ becomes a Lyapunov function for the system (since from Corollary 1, $\rho(x) < 1$, (15) implies that $\|x_{k+1}\|_v < \|x_k\|_v$) and hence asymptotic stability is guaranteed. With this choice of Lyapunov function, it is apparent that our algorithm is related to the well known technique of maximizing the rate of decrease of a Lyapunov function [see for instance Kalman and Bertram (1960)]. However, our choice of $v(x)$ guarantees that the algorithm is applicable to the entire domain of definition of the problem (rather than to a subset as is the case when using quadratic Lyapunov functions). Note also that the "local optimum" strategy is shortsighted in the sense that it minimizes the norm of the target state in one step as opposed to the "global optimum" sequence that minimizes the transit time to the origin. Clearly the two strategies do not coincide in general, although it is reasonable to expect similar behavior in the region far from the origin.

5. Implementation of the proposed control algorithm using a neural net

From (14) we have that, given the present state of the system x_k , u_k^* and ρ_k can be computed by solving the following optimization problem

$$\min_{u \in \Omega} \rho \quad (16)$$

subject to

$$\|Ax_k + Bu\|_v = \rho \|x_k\|_v$$

or equivalently, by using the definition of $v(x)$

$$\min \mu \quad (17)$$

subject to

$$\begin{aligned} -\mu\gamma + GBu &< -GAx_k \\ -\mu\gamma - GBu &< GAx_k \end{aligned} \quad (18)$$

where

$$\mu = \rho \|x_k\|_v.$$

Note that equations (17) and (18) define a linear programming problem that must be solved at each sampling interval. Rather than solving this problem using an on-board digital computer, we will follow an approach similar to those in Pyne (1956), Tank and Hopfield (1986) and Kennedy and Chua (1988), and use an analog circuit.

Consider a Hopfield continuous neural net (Hopfield and Tank, 1985). Each "neuron" is an analog element with its dynamics given by:

$$\frac{dy_i}{dt} = \frac{1}{c_i} \left(\sum_{j=1}^N T_{ij} x_j + I_i - \frac{y_i}{R_i} \right) \quad (19)$$

where g_i is a continuous, monotonically increasing function as illustrated in Fig. 1a. Hopfield and Tank (1986) showed that a problem of type (8) can be approximately solved employing a network with two types of amplifiers ("graded-response neurons"): (i) linear amplifiers, each one representing a variable; and (ii) nonlinear amplifiers, each one representing one of the Lagrange multipliers δ_i in (8).

The topology of the network is illustrated in Fig. 1b. The

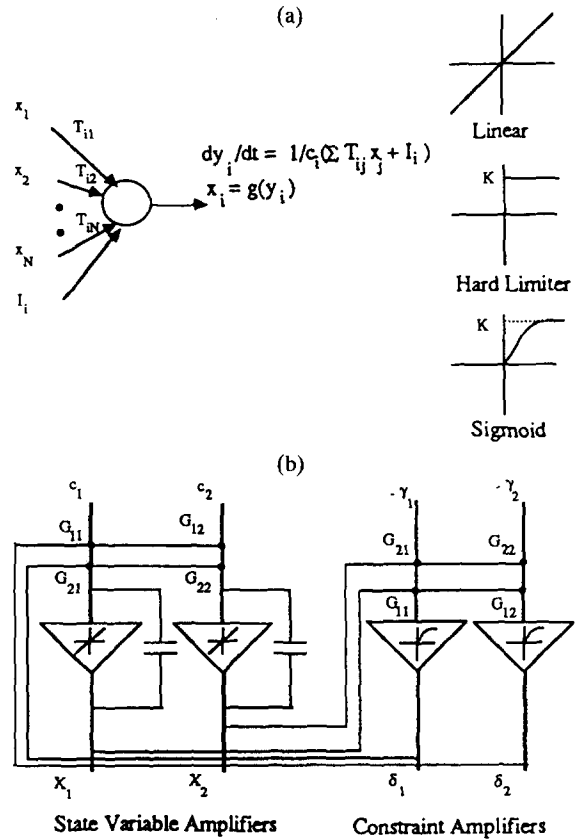


FIG. 1(a). Different types of "Neurons", (b) topology of the network to solve a LP problem.

amplifier representing the i th variable has a bias current input $I_i = c_i$ and is connected to each of the amplifiers representing the constraints, with interconnecting weight to the j th constraint given by $T_{ij} = G_{ij}$. The amplifier representing the j th constraint has a bias current input $I_j = \gamma_j$ and is connected to each of the amplifiers representing the variables, with interconnecting weight to the i th variable given by $T_{ji} = G_{ji}$. Note that when the neurons representing the variables are implemented using operational amplifiers with very large input impedance the last term in equation (19) vanishes and this network coincides with the dynamical, canonical, nonlinear programming circuit of Kennedy and Chua (1988). Hence, provided that the constraint amplifiers are sufficiently fast, the network converges without oscillations to a minimum of the total co-content function of the network, given by

$$\bar{G}(x) = c'x + \sum_{j=1}^p \int_0^{N_j x - \gamma_j} g(z) dz. \quad (20)$$

Comparing equations (20) and (8) it follows that by selecting $g(\cdot)$ as any continuous function sufficiently close to the transfer function of a "hard limiter" (such as a sigmoid), $G(x) \sim H(x)$ to any desired accuracy. From the result of Theorem 2 we have that, with proper choice of the weights, the minimum of H coincides with the minimum of the original LP problem (7). Therefore a circuit of this form can be used during the sampling interval to find the "local optimum" control sequence u_k^* , by solving the problem defined by (17) and (18). By identifying the corresponding terms in (7), (8), (17) and (18) we have

$$\begin{aligned} H(\mu, u) &= \mu + \delta' [(-\mu\gamma + GBu + GAx_k)', \\ &\quad (-\mu\gamma - GBu - GAx_k)']. \end{aligned} \quad (21)$$

Hence

$$\begin{aligned}
 \zeta^T &= (1, 0 \dots 0) \\
 \delta_i &= K\theta(-\mu\gamma + GBu + GAx_k), \\
 \delta_{i+p} &= K\theta(-\mu\gamma - GBu - GAx_k), \quad i = 1, p \\
 I_1 &= 1, I_i = 0; \quad i = 2, m + 1 \\
 T_{ii} &= -\gamma_i T_{1,i+p} = -\gamma_i; \quad i = 1, p \\
 T_{i+1,j} &= (GB)_{ij}, T_{i+1,j+p} = -(GB)_{ij}; \quad i = 1, m; \quad j = 1, p.
 \end{aligned}
 \tag{22}$$

A difficulty with the proposed circuit arises from the potentially damaging effects of nonlinear phenomena in the amplifiers, including saturation, that we have neglected so far. However, note that although the op-amps representing the variables have been assumed linear, our results hold as long as the feedback configuration used functions as an integrator, which depends (as long as there is no saturation) only on the assumptions of very large gain and input impedance. Furthermore, saturation has the effect of limiting the variables to the closed hypercube \mathcal{H} given by

$$\mathcal{H} = \{x : v_{\min} \leq x_i \leq v_{\max}\}.$$

Therefore, by scaling the problem so that $\mathcal{G} \subseteq \mathcal{H}$ saturation effects can be avoided. Note also that in this case condition (13) guarantees that for any initial condition in \mathcal{H} the net converges to the desired solution.

Another problem arises from the difficulty in estimating the convergence time of the net. An order of magnitude estimate of the speed of convergence had been proposed by Wolfe in a discussion appearing with Pyne (1956), by assuming that the system moves along the line determined by the intersection of $n - 1$ constraints and averaging this value over all possible directions in n -space. In this case the velocity at which the state moves is given by

$$v = \frac{v_u}{n} \tag{23}$$

where v_u is the unconstrained velocity. In the next section we will make use of this formula to estimate the convergence time of the network in a particular application.

6. An example

As an example application of our controller, we will consider the problem of minimum time control of an F-100 jet engine. The system at intermediate power, sea level static and Power Lever Angle (PLA) = 83° can be represented by a model with five states and five controls (DeHoff *et al.*, 1977). However, two of these controls are sufficient to control the system. Therefore, for simplicity, we will use only two controls. Hence, after discretizing and normalizing, we have:

$$\begin{aligned}
 A &= \begin{pmatrix} 0.8907 & 0.0474 & -0.0980 & 0.2616 & 0.0689 \\ 0.0237 & 0.9022 & -0.0202 & 0.1057 & 0.0311 \\ 0.0233 & -0.0149 & 0.8167 & 0.2255 & 0.0295 \\ 0.0 & 0.0 & 0.0 & 0.7788 & 0.0 \\ -0.0979 & 0.3532 & 0.3662 & 0.6489 & 0.0295 \end{pmatrix} \\
 \gamma &= \begin{pmatrix} 50.0 \\ 64.0 \\ 20.0 \\ 5.0 \\ 18.1 \end{pmatrix} \\
 B &= \begin{pmatrix} 0.0213 & -0.3704 \\ 0.0731 & -0.1973 \\ -0.0357 & -0.5538 \\ 0.2212 & 0.0 \\ 0.0527 & -3.9068 \end{pmatrix} \\
 G = I; \quad \Omega &= \{u \in R^2 : |u_1| \leq 31.0; |u_2| \leq 200.0\}.
 \end{aligned}
 \tag{24}$$

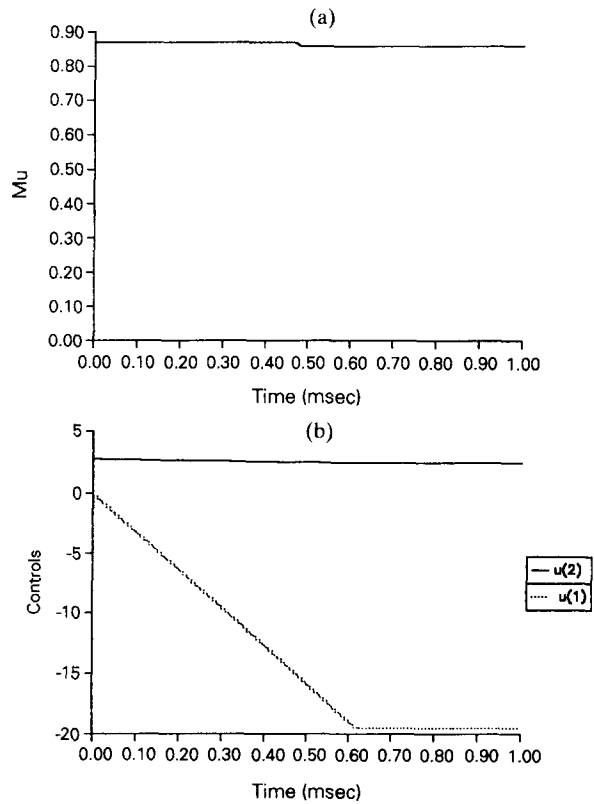


FIG. 2. Computing the controls using a neural net. (a) Time evolution of μ , (b) time evolution of the controls.

The sampling time for this system is 25 msec. The network was assumed to be constructed of operational amplifiers acting as switches, adders and integrators, and simulated using a digital computer. In order to be consistent with a physical implementation, the following values were chosen for the components: x_i (voltages) ~ volts, I_i (bias currents) ~ mA, $1/T_{ij}$ ~ K Ω , C_i (capacitance for the integrators) ~ 0.1 to 10 nF, $K = 2$ volts [from (13)].

With these values, (23) yield an estimate of 0.1 ~ 1 msec for the neural net to converge to the next control sequence. This is consistent with the observed convergence time for a typical iteration, as shown in Fig. 2, and well below the maximum time of 25 msec available for computations.

Figure 3 shows the states of the system, for the initial condition:

$$x_0 = (50.0, 0.0, 0.0, 0.0, 0.0)'$$

when using the strategy u^* versus the true minimum. Note that the "local optimum" strategy takes on the order of one and a half times longer to drive the states to zero than the true minimum controller. As we mentioned before, this is due to the fact that the "local minimum" strategy is suboptimal, minimizing the norm of the next state of the system rather than the total transit time to the goal. Note also that the conservative nature inherent to the Lyapunov function based design is apparent in the fact that the states are precluded from riding the boundaries of the admissible region, as is the case with the true optimal controller.

7. Conclusions

Most realistic control problems involve some types of constraints. However, up to date there are no feedback controllers that allow dealing with this class of problems except in restricted cases. In this paper we propose a suboptimal minimum time controller, based upon on-line optimization, for systems with linear state and control inequality constraints. In the first portion of the paper we present theoretical results showing that our control algorithm yields asymptotically stable closed-loop systems. In the

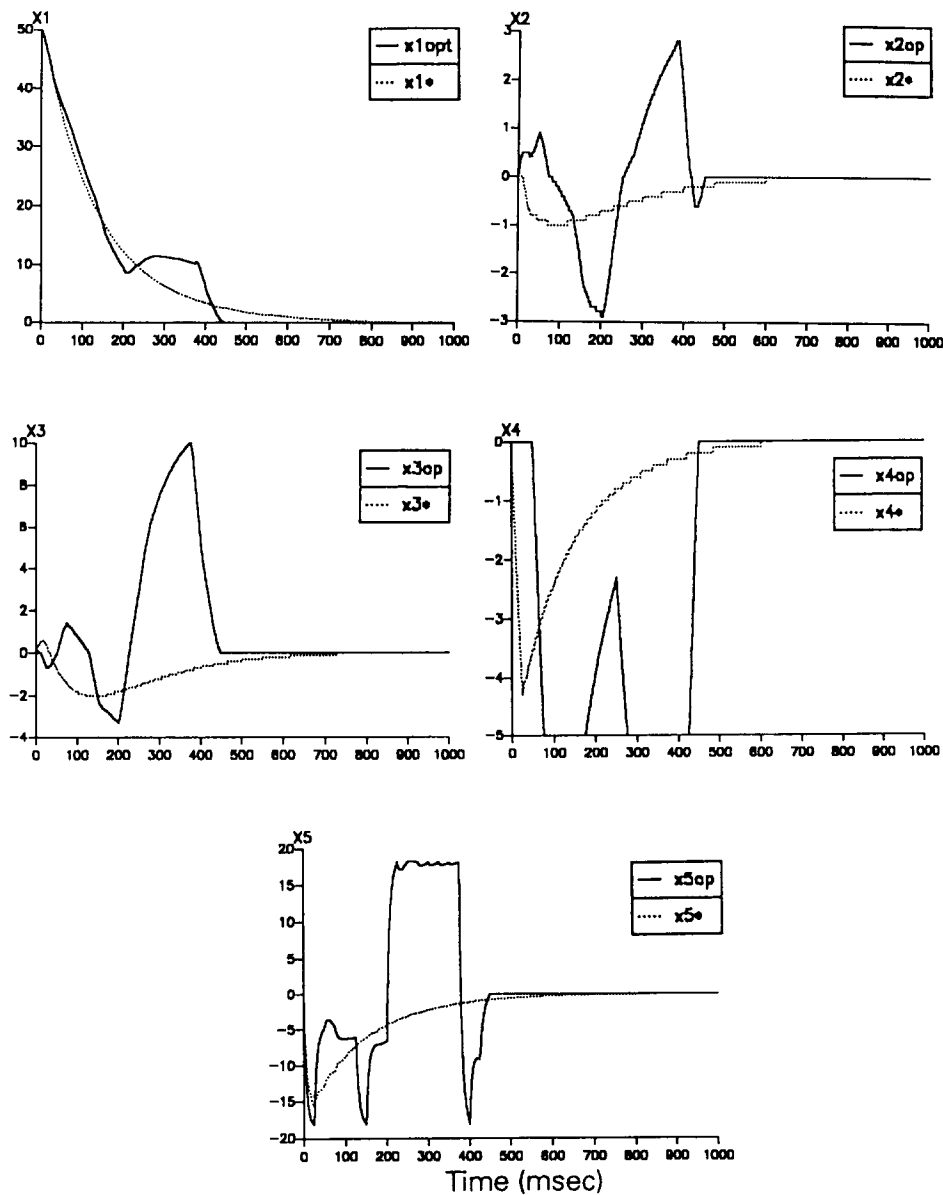


FIG. 3. Suboptimal vs optimal strategies.

second portion of the paper, we show that our control algorithm can be implemented using an analog neural net.

Although the proposed controller is overly conservative due to the incorporation of the stability results, experiments show that it performs reasonably well when compared with the true optimal controller. We believe that our controller may provide significant advantages over the controllers available at the present time for the control of constrained systems. In particular, the analog circuit implementation has the potential to carry-out the required minimization very fast, thus providing an alternative to the use of a digital computer in the feedback loop. This alternative offers advantages in terms of cost and reliability and may prove to be specially valuable when the time available for computations is limited.

Acknowledgements—We would like to thank the reviewers for a thorough discussion and many helpful suggestions for improving the original manuscript.

References

DeHoff, R. L. (1977). *F-100 Multivariable Control Synthesis Program*, Vol 2, Appendices A–K. Systems Control Inc.,

AFAPL-TR-77-35.

Gutman, P. O. (1986). A linear programming regulator applied to hydroelectric reservoir level control. *Automatica*, **22**, 533–541.

Hopfield, J. J. and D. W. Tank (1985). Neural computation of decisions on optimization problems. *Biol. Cybern.*, **52**, 141–152.

Kalman, R. E. and J. E. Bertram (1960). Control system analysis and design via the second method of Lyapunov. Part II, Discrete time systems. *ASME J. Basic Engng*, pp. 394–400.

Kennedy, M. P. and L. O. Chua (1988). Neural networks for nonlinear programming. *IEEE Trans. Circ. Syst.*, **CAS-35**, 554–562.

Luenberger, D. G. (1969). *Optimization by Vector Space Methods*. Wiley, New York.

Pyne, I. B. (1956). Linear programming on an electronic analogue computer. *Trans. AIEE, Part I (Comm. Elect.)*, **75**, 139–143.

Sznaier, M. and M. J. Damborg (1987). Suboptimal control of linear systems with state and control inequality constraints. *Proc. 26th IEEE Conf. Decision Control*, Los Angeles, CA, pp. 761–762.

Sznaier, M. and M. J. Damborg (1989). Control of constrained discrete time linear systems using quantized controls. *Automatica*, **25**, 623–628.

Sznaier, M. and M. J. Damborg (1990). Heuristically enhanced feedback control of constrained discrete time

linear systems. *Automatica*, **26**, 521–532.

Tank, D. W. and J. J. Hopfield (1986). Simple neural optimization networks: an A/D converter, signal decision circuit, and a linear programming circuit. *IEEE Trans. Circ. Syst.*, **CAS-33**, 533–541.