Divide columns $2n + 1, 2n + 2 \cdots 3n$ by 3!. Continue in this fashion through to columns $(N - 1)n + 1, (N - 1)n + 2, \cdots Nn$ divided by $N!$. ii) Multiply rows $2r + 1, 2r + 2, \cdots, 3r$ by 2!. Multiply rows $3r + 1, 3r + 2, \cdots, 4r$ by 3!. Continue in this fashion through to rows $Nr + 1, Nr + 2, \cdots, (N + 1)r (= r^2 + 2r)$ multiplied by $N!$. The matrix obtained is W_N^1.

Now let us perform the next sequence of elementary row and columns operations on W_N^1. Multiply rows $r + 1, r + 2, \cdots, 2r$ by (-1). Multiply columns $n + 1, n + 2, \cdots, 2n$ by (-1). Multiply rows $3r + 1, 3r + 2, \cdots, 4r$ by (-1). Multiply columns $3n + 1, 3n + 2, \cdots 3n$ by (-1). Continue in this fashion through to rows $Nr + 1, Nr + 2, \cdots (N + 1)r$ (or, if N is even, through to rows $(N - 1)n + 1, (N - 1)r + 2, \cdots Nn$) multiplied by (-1). The matrix obtained is W_c. ■

Example:

1) The system

$$\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \dot{x}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \text{ is c.c..}$$

First we see that $sE - A = \left[\begin{smallmatrix} -1 & s \\ 0 & -1 \end{smallmatrix}\right]$ and thus (2.1b) is satisfied. Also, we have $C = [1\ 0]$, $CE = [0\ 1]$ and $CA = [1\ 0]$. Hence, the generalized controllability matrix W_c,

$$W_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

is surjective and the system is completely controllable.

Assume now that it is desired to transfer the system from $x(0-) = \left[\begin{smallmatrix} 0 \\ -1 \end{smallmatrix}\right]$ to $x(1) = \left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right]$. Then, using (3.14) and (2.4) one finds that $u^*(t) = 1 - 3t^2 + 2t^3$ transfers the system along

$$x^*(t) = \begin{bmatrix} 6t - 6t^2 \\ -1 + 3t^2 - 2t^3 \end{bmatrix}.$$

If one increases the degree of the polynomials, one can perform the transfer along a "smoother trajectory."

IV. Conclusions

This note has established that c-controllability of singular systems is associated with a set of linear algebraic equations from which one can explicitly obtain the polynomial control function $u(t)$ which transfers the system under consideration from its initial position to the desired target. The results which have been obtained in this note provide a direct approach to the solution of optimal control problems by a state parameterization approach.

Acknowledgment

The author is grateful to anonymous reviewers for helpful comments.

References

[1] D. G. Luenberger, "Dynamic systems in descriptor form," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 312–321, 1977.
[2] D. G. Luenberger, "Time invariant descriptor systems," *Automatica*, vol. 14, pp. 473–480, 1978.
[3] A. Dervisoğlu and C. A. Desoer, "Degenerate networks and minimal differential equations," *IEEE Trans. Circ. Syst.*, vol. CAS-22, pp. 769–775, 1975.
[4] G. C. Verghese, P. V. Dooren, and T. Kailath, "Properties of the system matrix of a generalized state-space system," *Int. J. Contr.*, vol. 30, pp. 235–243, 1979.
[5] G. C. Verghese, B. C. Levy, and T. Kailath, "A generalized state-space for singular systems," *IEEE Trans. Automat. Contr.*, vol. AC-26, pp. 811–831, 1981.
[6] J. D. Aplevich, "Time-domain input-output representation of linear systems," *Automatica*, vol. 17, pp. 509–522, 1981.
[7] P. Bernhard, "On singular implicit linear dynamical systems," *SIAM J. Contr.*, vol. 20, pp. 612–633, 1982.
[8] R. F. Sincovec, A. M. Erisman, E. L. Yip, and M. A. Epton, "Analysis of descriptor systems using numerical algorithms," *IEEE Trans. Automat. Contr.*, vol. AC-26, pp. 139–147, 1981.
[9] E. L. Yip and R. F. Sincovec, "Solvability, controllability, and observability of continuous descriptor systems," *IEEE Trans. Automat. Contr.*, vol. AC-26, pp. 702–707, 1981.
[10] D. Cobb, "Controllability, observability and duality in singular systems," *IEEE Trans. Automat. Contr.*, vol. AC-29, pp. 1076–1082, 1984.
[11] F. L. Lewis, "Fundamental, reachability, and observability matrices for discrete descriptor systems," *IEEE Trans. Automat. Contr.*, vol. AC-30, pp. 502–505, 1985.
[12] A. Ailon, L. Barrachart, J. Grimm, and G. Langholz, "On polynomial controllability with polynomial state of linear constant system," *IEEE Trans. Automat. Contr.*, vol. AC-31, pp. 155–156, 1986.

Computation of Minimum-Time Feedback Control Laws for Discrete-Time Systems with State-Control Constraints

S. S. KEERTHI AND E. G. GILBERT

Abstract—**The problem of finding a feedback law that drives the state of a linear discrete-time system to the origin in minimum-time subject to state-control constraints is considered. Algorithms are given to obtain facial descriptions of the M-step admissible sets. These descriptions are then used to characterize the complete class of minimum-time feedback laws. Moreover, the characterization leads to a conceptually simple on-line implementation. The main ideas are illustrated with two simple examples.**

I. Introduction

The problem of finding a feedback law that drives the state of a linear discrete-time system to the origin in minimum time subject to magnitude constraints on the control is a classic one, first formulated by Kalman [8] and then analyzed in detail by Desoer and Wing [2]–[4]. Their description of the finite-step admissible set is by its vertices. Their main result is a switching surface in R^n that separates the region of positive and negative control. The switching surface is easily developed for second-order systems; but for higher order systems its characterization is very complicated and difficult to implement. Since [2]–[4], relatively little work has been done. Lin [10] has suggested an alternative method which uses a representation of admissible sets by their faces; and, Gutman and Cwikel [5]–[7] extend the vertex representation to problems in which a compact polyhedral state-space constraint set is also included. However, the algorithms in [10] and [7] involve rather difficult computations; also, the determination of a minimum-time feedback law is not simple.

This note offers a new approach to the problem in which a facial description of the M-step admissible sets is also used. However, the computations are more systematic and straightforward and problems with joint state-control constraints can be treated. Furthermore, in our approach, the complete class of minimum-time feedback laws is characterized. The description of the admissible set and the feedback law is entirely different from the ones of Desoer and Wing and, while more complicated, is more likely to yield feasible on-line implementations. The main limitation is the dimensionality of the facial representation of the M-step admissible sets, which can become very large for high-order systems.

We conclude this section with some notations and definitions. Let N denote the set of positive integers; for $I \subset N$ let card I = cardinality of I. For $z \in R^p$ and $P \in R^{q \times p}$: z^i denotes the ith component of z; P^i is the ith row of P; and P' denotes the transpose of P. $I_q \in R^{q \times q}$ is the identity matrix. A set $Z \subset R^p$ is *polyhedral convex* if $\exists q \in N$, $P \in R^{q \times p}$ and $\gamma \in R^q$ such that $Z = \{z \in R^p : Pz + \gamma \leq 0\}$. The notation $a := b$ means that the value of a is replaced by that of b. $\emptyset$ denotes the empty set.

Manuscript received July 15, 1986; revised November 24, 1986 and January 5, 1987.
The authors are with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109-2140.
IEEE Log Number 8613534.

0018-9286/87/0500-0432$01.00 © 1987 IEEE

II. THE MINIMUM-TIME PROBLEM

Consider the linear system with constraints

$$x_{k+1}=Ax_k+Bu_k,\ (x_k, u_k) \in Z,\ 0 \le k \le M-1, \tag{2.1}$$

$$x_0=a,\ x_M=0 \tag{2.2}$$

where $x_k \in R^n$, $u_k \in R^m$, $A \in R^{n\times n}$, $B \in R^{n\times m}$,

$$Z=\{(x, u) : Ex+Fu+\lambda \le 0\} \tag{2.3}$$

$E \in R^{l\times n}$, $F \in R^{l\times m}$ and $\lambda \in R^l$. For meaningfulness, we need the following.

Assumption A: $(0, 0) \in Z$, i.e., $\max_{1\le i\le l} \lambda^i \le 0$.

Assumption A will be assumed throughout this note without further mention. For $M > 0$, the set of admissible intial states is

$$X(M)=\{a : \exists\ \{(x_k, u_k)\}_{k=0}^{M-1} \text{ that satisfies (2.1) and (2.2)}\}. \tag{2.4}$$

Also, let $X(0) = \{0\}$. Clearly, $0 \in X(M)$ and $X(M) \subset X(M+1)$, $M \ge 0$. It is easy to see that the maximal admissible set is given by $\bar{X} = \cup_{M\ge 0} X(M)$. While $X(M)$ is closed for each $M \ge 0$, it is easy to give examples where $\bar{X}$ is open [7]. The minimum-time problem can now be precisely formulated.

Minimum-Time Problem: For $x \in \bar{X}$, find the minimum time

$$\bar{M}(x)=\min \{M : x \in X(M)\}. \tag{2.5}$$

Also, find a feedback law, $\mu:\bar{X} \to R^m$ which satisfies $\mu(0) = 0$ and

$$(x, \mu(x)) \in Z,\ Ax+B\mu(x) \in X(\bar{M}(x)-1),\ x \in \bar{X},\ x\ne 0. \blacksquare \tag{2.6}$$

The feedback law μ is, in general, nonunique. For problems with control constraints, this has been noted by Desoer and Wing [2]. For $M \ge 1$ let

$$W(x; M)=\{u : (x, u) \in Z,\ Ax+Bu \in X(M-1)\},\ x \in X(M). \tag{2.7}$$

Clearly μ is a minimum-time feedback law if and only if $\mu(0) = 0$ and

$$\mu(x) \in W(x; \bar{M}(x)), \qquad x \in \bar{X},\ x\ne 0. \tag{2.8}$$

Thus, by obtaining a representation for $W(\cdot; M)$ we can characterize the class of all minimum-time feedback laws. By (2.7), this depends on the finding of a representation for $X(M)$. An algorithm for obtaining a representation of $X(M)$ is discussed in Section IV. The ideas given there are based on results in Section III. Section V describes the implementation of μ; examples are given in Section VI.

III. THE PROJECTION ALGORITHM

We now present an algorithm for projecting a polyhedral convex set on to a subspace. It is a slight modification of a method of solving linear inequalities, first developed by Fourier in 1826 and which later came to be known as the Fourier–Motzkin elimination method [12].

Problem 3.1: Consider the set

$$X=\{x \in R^n : \exists u \in R^m \ni Gx+Hu+\psi \le 0\} \tag{3.1}$$

where $G \in R^{s\times n}$, $H \in R^{s\times m}$, and $\psi \in R^s$. Find: $r \in N$, $P \in R^{r\times n}$, and $\gamma \in R^r$ such that

$$X=\{x \in R^n : Px+\gamma \le 0\}. \qquad \blacksquare \tag{3.2}$$

We begin by considering the case $m = 1$. The algorithm for this case will lead to a solution for the case $m > 1$. For the case $m = 1$, note that $H \in R^{s\times 1}$.

Algorithm 3.1. Solution of Problem 3.1 for $m = 1$:

Step 1: Identify the following subsets of $\{1, \cdots, s\}$: $I^0 = \{i:H^i = 0\}$, $I^+ = \{i:H^i > 0\}$ and $I^- = \{i:H^i < 0\}$. Let $s^0 = \text{card}\, I^0$, $s^+ = \text{card}\, I^+$ and $s^- = \text{card}\, I^-$. Go to Step 2.

Step 2: Let $C = [G\ \psi] \in R^{s\times(n+1)}$. Form the matrix $[P\ \gamma] \in R^{r\times(n+1)}$, $r = s^0 + s^+s^-$, whose rows are the elements of the sets: $\{C^i: i \in I^0\}$ and $\{(H^iC^j - H^jC^i),\ i \in I^+,\ j \in I^-\}$. Stop. ■

Proof: If $s^- = 0$ or $s^- = 0$ or $s^+ = s^- = 0$ it is obvious that the rows of $[P\ \gamma]$ are the elements of $\{C^i: i \in I^0\}$. Hence, assume hereafter that $s^+ > 0$ and $s^- > 0$. Define: $\mathcal{L} = \{z \in R^{n+1}: \exists u \in R \ni Cz + Hu \le 0\}$, so that, $X = \{x: z = (x, \alpha),\ \alpha = 1,\ z \in \mathcal{L}\}$. Let: $T \in R^{s\times s}$ be a diagonal matrix whose ith diagonal element is $(|H^i|)^{-1}$ if $i \in I^+ \cup I^-$, and 1 if $i \in I^0$; and $S = TC$. Clearly then, all the elements of the vector (TH) are either 1, -1, or 0. Also,

$$\mathcal{L}=\{z : \exists u \in R \ni Sz+THu \le 0\}. \tag{3.3}$$

Define: $a(z) = \max_{i\in I^+} S^iz$; and $b(z) = \max_{i\in I^-} S^iz$. This, together with the structure of (TH) allows (3.3) to be rewritten as

$$\mathcal{L}=\{z : S^iz \le 0,\ i \in I^0,\ \exists u \in R \ni a(z) \le -u,\ b(z) \le u\}. \tag{3.4}$$

By examining the set $\mathcal{E} = \{(a, b): \exists u \in R \ni a \le -u,\ b \le u\} \subset R^2$, it is easy to see that $\mathcal{E} = \{(a, b): a + b \le 0\}$. Using this in (3.4) we get

$$\begin{aligned}\mathcal{L}&=\{z : S^iz \le 0,\ i \in I^0,\ a(z)+b(z) \le 0\}\\ &=\{z : S^iz \le 0,\ i \in I^0,\ (S^i+S^j)z \le 0,\ i \in I^+,\ j \in I^-\}.\end{aligned} \tag{3.5}$$

It is then easy to verify Step 2 of the algorithm. ■

For $m = 1$, Problem 3.1 consists of eliminating the real variable, u, from the representation for X in (3.1) to obtain (3.2). When $m > 1$, the algorithm for $m = 1$ can be repeated m times, eliminating one component of the vector u at a time. To make this precise, define:

$$X^m=\{(x, u) \in R^{n+m} : Gx+Hu+\psi \le 0\}, \tag{3.6}$$

and for $j = m-1, \cdots, 0$, let $x_j = (x, u^1, \cdots, u^j) \in R^{n+j}$ and

$$X^j=\{x_j \in R^{n+j} : \exists u^{j+1} \ni R \quad (x_j, u^{j+1}) \in X^{j+1}\}. \tag{3.7}$$

Clearly, $X^0 = X$, where X is the set in (3.1). To obtain (3.2), proceed by the following steps.

Algorithm 3.2 Solution of Problem 3.1 for $m > 1$:

Step 1: Let: $H_m = m$th column of H; G_m be given by $[G_m\ H_m] = [G\ H]$; and $\psi_m = \psi$. Then by (3.6) and (3.7), we get, for $j = m - 1$,

$$X^j=\{x_j \in R^{n+j} : \exists u^{j-1} \in R \quad \ni G_{j-1}x_j+H_{j-1}u^{j+1}+\psi_{j-1} \le 0\}. \tag{3.8}$$

Let $j = m - 1$ and go to Step 2.

Step 2: Given (3.8), eliminate u^{j-1} using the algorithm for $m = 1$ to obtain P_j and γ_j such that

$$X^j=\{x_j \in R^{n-j} : P_jx_j+\gamma_j \le 0\}. \tag{3.9}$$

Go to Step 3.

Step 3: If $j = 0$ then let: $P = P_0$, $\gamma = \gamma_0$ and stop. Otherwise go to Step 4.

Step 4: Let H_j be a vector and G_j be a matrix such that $[G_j\ H_j] = P_j$; also let $\psi_j = \gamma_j$. Then by (3.9) and (3.7), (3.8) holds for $(j - 1)$. Let $j := j - 1$ and go to Step 2. ■

Remark 3.1: It usually turns out that the matrix $[P\ \gamma]$ given by Algorithms 3.1 and 3.2 contains many redundant rows that are not needed to represent the set X. A row may be tested for redundancy by solving a linear programming problem in which: a) the tested row determines the cost, b) the remaining rows form the constraints [11]. See [1] for an efficient procedure.

IV. REPRESENTATIONS FOR $X(M)$ AND $W(x; M)$

We now describe the representations for $X(M)$ and $W(x; M)$ and describe an algorithm for finding them.

Theorem 4.1: i) For $M \ge 1$: $\exists r_M \in N$, $P_M \in R^{r_M\times n}$ and $\gamma_M \in R^{r_M}$ such that

$$X(M)=\{x \in R^n : P_Mx+\gamma_M \le 0\}. \tag{4.1}$$

ii) For $M \ge 1$: $\exists s_M \in N$, $G_M \in R^{s_M\times n}$, $H_M \in R^{s_M\times m}$ and $\psi_M \in R^{s_M}$ such that

$$W(x; M)=\{u : H_Mu \le -G_Mx-\psi_M\},\ x \in X(M). \tag{4.2}$$

iii) The data in i) and ii) are obtained by the following steps.

Step 1: Let: $r_0 = 2n$, $P_0 = [I_n \; -I_n]'$, $\gamma_0 = 0$ and define $X(0)$ by (4.1). Set $M = 0$.

Step 2: Define

$$G_{M+1}=\begin{pmatrix} E \\ P_M A \end{pmatrix}, \; H_{M+1}=\begin{pmatrix} F \\ P_M B \end{pmatrix} \text{ and } \psi_{M-1}=\begin{pmatrix} \lambda \\ \gamma_M \end{pmatrix}. \quad (4.3)$$

Step 3: Let

$$\hat{Z}=\{(x, u) : G_{M+1}x+H_{M+1}u+\psi_{M+1}\le 0\}. \quad (4.4)$$

Then

$$X(M+1)=\{x : (x, u) \in \hat{Z}\}. \quad (4.5)$$

Apply Algorithm 3.1 or 3.2 to (4.5) to obtain P_{M+1} and γ_{M+1}.

Step 4: Use the idea in Remark 3.1 to remove redundant rows of $[P_{M+1} \; \gamma_{M+1}]$ which are not needed to represent $X(M + 1)$.

Step 5: Set $M := M + 1$ and return to Step 2. ■

Proof: Note that

$$\hat{Z}=\{(x, u) : (x, u) \in Z, Ax+Bu \in X(M)\}. \quad (4.6)$$

Then it is clear that the steps recursively define $X(M)$, $M \ge 1$. The result (4.2) follows from (2.7), (4.6), (4.3), and (4.4). ■

Remark 4.1: The representation (4.1) is a facial description of the polyhedral convex set, $X(M)$; the rows of P_M denote the normals of the hyperplanes that contain the faces of $X(M)$. Our algorithm is entirely "facial." The algorithm of Gutman and Cwikel [7] is more involved because it goes back and forth between facial and vertex representations. Also, [7] does not give a method for developing a minimum time feedback law.

Remark 4.2: for the scalar control case ($m = 1$), (4.2) can be simplified. When $m = 1$, note that $H_M \in R^{s_M \times 1}$. Let $I^+ = \{i: H_M^i > 0\}$ and $I^- = \{i: H_M^i < 0\}$. Define the functions $\eta_M^-: X(M) \to R$ and $\eta_M^-: X(M) \to R$ as follows:

$$\eta_M^+(x)=\begin{cases} \min_{i\in I^+} (H_M^i)^{-1}(-G_M^i x-\psi_M^i) & \text{if } I^+ \ne \emptyset, \\ \infty & \text{if } I^+ = \emptyset; \end{cases} \quad (4.7)$$

$$\eta_M^-(x)=\begin{cases} \max_{i\in I^-} (H_M^i)^{-1}(-G_M^i x-\psi_M^i) & \text{if } I^- \ne \emptyset, \\ -\infty & \text{if } I^- = \emptyset. \end{cases} \quad (4.8)$$

Then

$$W(x; M)=\{u : \eta_M^-(x)\le u\le \eta_M^+(x)\}, \qquad x \in X(M). \quad (4.9)$$

Remark 4.3: The main problem with our algorithm is the growth of r_M. Because of the combinatoric calculations in Step 2 of Algorithm 3.1 (case $m = 1$), it is possible that r_M grows horrendously with M. For example, if r_M is large and the worst case situation occurs ($s^+ = s^- \cong s/2$ in Algorithm 3.1), then $r_{M+1} \cong (r_M/2)^2$. For problems with compactness constraints on the state, we have generally observed that such a growth of r_M does not occur; see, for instance the two examples in Section VI.

While $X(M)$, $M > 0$, can be described using Theorem 4.1, it is usually difficult to find a representation for the maximal admissible set, $\bar{X}$. If $\bar{X} = X(\hat{M})$ for some $\hat{M}$, then $\bar{X}$ is polyhedral convex and its description becomes easy. The following theorem gives conditions under which this may occur.

Theorem 4.2: i) $\exists \hat{M}$ such that $\bar{X} = X(\hat{M})$ if and only if $X(\hat{M} + 1) = X(\hat{M})$; ii) if $\{x:(x, u) \in Z\}$ is compact, $\exists \hat{M}$ such that $\bar{X} = X(\hat{M})$ if and only if $\bar{X}$ is polyhedral convex. ■

Proof: If $\bar{X} = X(\hat{M})$ then $X(\hat{M} + 1) = X(\hat{M})$ since $X(\hat{M}) \subset X(\hat{M} + 1) \subset \bar{X}$. Now suppose $X(\hat{M} + 1) = X(\hat{M})$. Let $M = \hat{M}$. Then from (4.3) and (4.4) with $M := M + 1$ it follows that $X(M + 2) = X(M + 1)$. Using this result inductively for $M \ge \hat{M}$ completes the proof of part i).

Now consider part ii). If $\exists \hat{M}$ such that $\bar{X} = X(\hat{M})$, then clearly $\bar{X}$ is polyhedral convex. Suppose $\bar{X}$ is polyhedral convex. Since $\bar{X} \subset \{x:(x, u) \in Z\}$, $\bar{X}$ is a compact polyhedral convex set; so it can be represented as the convex hull of its vertices, $\bar{x}_i$, $i = 1, \cdots, t$. For $1 \le i \le t$, let $\bar{M}_i = \min\{M: \bar{x}_i \in X(M)\}$, the minimum time from $\bar{x}_i$. Since $\bar{x}_i \in \bar{X}$, $\bar{M}_i$ is finite. Let $\hat{M} = \max\{\bar{M}_i: 1 \le i \le t\}$. It is then easy to verify that $\bar{X} = X(\hat{M})$. ■

V. Implementation of Minimum-Time Feedback Laws

In general the representations for $X(M)$ and $W(x; M)$ are computed off-line. Then (4.1) and (4.2) are used to give a simple on-line procedure for implementing a minimum-time feedback law, $\mu(\cdot)$. Given $x \in \bar{X}$, $\mu(x)$ is computed in two steps: 1) find the minimum time, $\bar{M}(x)$; 2) compute $\mu(x) \in W(x; \bar{M}(x))$. The first step can be achieved by the following algorithm.

Algorithm 5.1: Finding $\bar{M}(x)$ given $x \in \bar{X}$.

Step 1: If $x = 0$, let $\bar{M}(x) = 0$ and stop. Otherwise go to Step 2.

Step 2: Compute $\bar{M}(x) = \min\{M: M \ge 1, P_M x + \gamma_M \le 0\}$ as follows.

Step 2a: Let $M = 1$ and go to Step 2b.

Step 2b: Compute $\beta_M^j = P_M^j x + \gamma_M^j$. If $\beta_M^j > 0$ for any $j \in \{1, \cdots, r_M\}$, go to Step 2d. Otherwise go to Step 2c.

Step 2c: Let $\bar{M}(x) = M$ and stop.

Step 2d: Let $M := M + 1$ and go to Step 2b. ■

Once the minimum time $\bar{M}(x)$ is computed, the second step in the feedback law implementation involves the determination of $\mu(x) \in R^m$: $\mu(x) \in W(x; \bar{M}(x))$. This can be done using (4.2) as follows. Let $M = \bar{M}(x)$ and compute $\delta = -G_M x - \psi_M$. Then obtain $\mu(x)$ to be any feasible point in the set $W(x; M) = \{u: H_M u \le \delta\}$. This can be done efficiently using special procedures in linear programming [11]. Since m, the dimension of u, is usually small, the above computation is not difficult. In particular, when $m = 1$, $W(x; M)$ simplifies to (4.9) and finding a feasible point in that set becomes very easy.

Both Algorithm 5.1 and the determination of $\mu(x)$ permit efficient use of computer pipelining and parallelism. Because of this and the simplicity of the required computations, rapid on-line implementations may be possible even though r_M and the resulting data storage requirements may be large.

VI. Numerical Examples

The first example that we consider is a double integrator system, described by the following state equations: $\dot{x}^1(t) = x^2(t)$, $\dot{x}^2(t) = u(t)$, $t \ge 0$. Using a zero-order hold with $T = 1$ gives a system of the form (2.1) where $x_k = (x^1(kT)x^2(kT))'$,

$$A=\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \text{ and } B=\begin{pmatrix} 0.5 \\ 1 \end{pmatrix}.$$

Let $Z = \{(x, u): |x^1| \le 25, |x^2| \le 5, |u| \le 1\}$. Let

$$\hat{E}=\begin{pmatrix} 0.04 & 0 \\ 0 & 0.2 \\ 0 & 0 \end{pmatrix} \text{ and } \hat{F}=\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Then it is easy to see that Z is given by (2.3) if we let: $l = 6$; $E = (\hat{E}' \; -\hat{E}')'$; $F = (\hat{F}' \; -\hat{F}')'$; and $\lambda^i = -1$, $1 \le i \le 6$.

For this example, it turns out from numerical calculations based on the algorithms presented above that the conditions of Theorem 4.2 are satisfied, and $\bar{X} = X(15)$. The admissible set $\bar{X}$, together with $X(M)$ for $M = 3, 7, 11$, and 15 are shown in Fig. 1. An interesting observation can be made from these sets. The number of faces in $X(M)$, r_M, does not necessarily increase with M; for instance, $r_7 = 20$ and $r_{15} = 14$. A minimum time trajectory from the initial condition, $x_0 = (25 \; 0)'$, developed using the expressions in (4.7)–(4.9) is also indicated on the phase plane in Fig. 1.

The above example is also considered in [7]. There, the admissible sets, $X(M)$, $M \ge 0$, are described in terms of their vertices. The computations are more complex and a minimum-time feedback law is not given.

As a second example consider the triple integrator problem, described by the following equations: $\dot{x}^1(t) = x^2(t)$, $\dot{x}^2(t) = x^3(t)$ and $\dot{x}^3(t) = u(t)$,

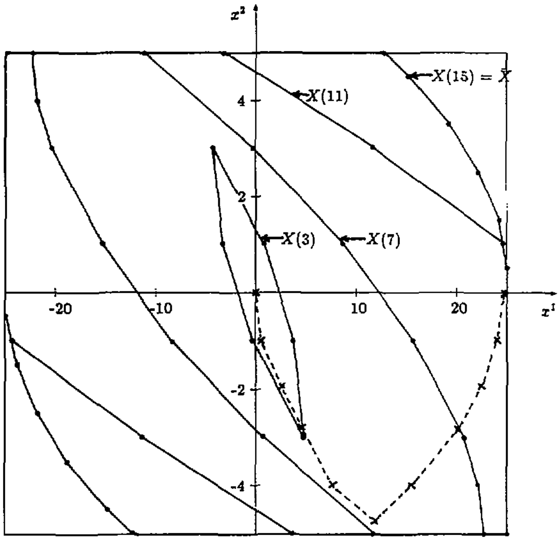

Fig. 1. Phase plane for the double integrator example.

TABLE I
COMPLEXITY OF $X(M)$ FOR TRIPLE INTEGRATOR WITH $T = 1$

M	2	3	4	5	6	7
r_{M+1} before Step 4	24	48	246	1128	3546	8382
r_{M+1} after Step 4	12	30	66	118	186	212

$t \geq 0$. Consider a zero-order hold discretization with period, $T = 2$, and let $x_k = (x^1(kT)x^2(kT)x^3(kT))'$. Let $Z = \{(x, u):|x^1| \leq 25, |x^2| \leq 5, |x^3| \leq 1, |u| \leq 1\}$. Calculations were done as for the double integrator to obtain representations for $X(M)$, $M \geq 0$. The convergence conditions of Theorem 4.2 are satisfied for this problem too, and $\hat{X} = X(9)$. The row dimension of P_9 is $r_9 = 26$. In both the examples, r_M, the row dimension of P_M does not grow rapidly with M. This is mainly due to the presence of compactness constraints on the state.

The above examples may suggest that the development of P_M is easy whenever $\{x:(x, u) \in Z\}$ is compact. This is not always true. For example, consider the triple integrator problem with $T = 1$. The computations are extensive because r_{M+1} before Step 4 of Theorem 4.1 is large. Moreover, numerical errors become critical in Step 4 because many redundant rows of $[P_{M+1}\gamma_{M-1}]$ are nearly active. To assure that active rows were not erroneously eliminated in our computations, an error tolerance was used in implementary Step 4. Table I shows the results for $M \leq 7$. Recently, R. J. Caron has implemented Step 4 using an improved version of the algorithm described in [1]. The computations are much quicker and permit a higher standard of accuracy in eliminating redundant rows. For example, r_8 has been reduced from 212 to 56, indicating that our computed $[P_8\ \gamma_8]$ has many redundant rows which were missed because of the error tolerance.

References

[1] R. J. Caron, J. F. McDonald, and C. M. Ponic, "Classification of linear constraints as redundant or necessary," Dep. Math., Univ. Windsor, Windsor, Ont., Canada, Winsor Math. Rep. WMR 85-09, Nov. 1985.

[2] C. A. Desoer and J. Wing, "An optimal strategy for a saturating sampled-data system," *IRE Trans. Automat. Contr.*, vol. AC-6, pp. 5–15, Feb. 1961.

[3] ——, "A minimal time discrete system," *IRE Trans. Automat. Control*, vol. AC-6, pp. 111-125, May 1961.

[4] ——, "Minimal time regulator problem for linear sampled-data systems (General theory)," *J. Franklin Inst.*, vol. 271, pp. 208–228, Sept. 1961.

[5] P. O. Gutman and M. Cwikel, "Admissible sets and feedback control for discrete-time linear dynamical systems with bounded controls and states," *IEEE Trans. Automat. Contr.*, vol. AC-31, pp. 373–376, Apr. 1986; part 2 in *IEEE Trans. Automat. Contr.*, vol. AC-31, pp. 457–459, May 1986.

[6] P. O. Gutman and M. Cwikel, "Convergence of an algorithm to find maximal state constraint sets for discrete-time linear dynamical systems with bounded controls and states," *IEEE Trans. Automat. Contr.*, vol. AC-31, pp. 457–459, May 1986.

[7] ——, "An algorithm to find maximal state constraint sets for discrete-time linear dynamical systems with bounded controls and states," *IEEE Trans. Automat. Contr.*, vol. AC-32, pp. 251-254, Mar. 1987.

[8] R. E. Kalman, "Optimal nonlinear control of saturating systems by intermittent action," 1957 IRE Wescon Convention Record, pt. 4, pp. 130–135.

[9] S. S. Keerthi, "Optimal feedback control of discrete-time systems with state-control constraints and general cost functions," Ph.D. dissertation, Univ. Michigan, Ann Arbor, 1986.

[10] J. N. Lin, "Determination of reachable set for a linear discrete system," *IEEE Trans. Automat. Control*, vol. AC-15, pp. 339-342, June 1970.

[11] C. B. Milham, "Fast feasibility methods for linear programming," *OPSEARCH*, vol. 13, nos. 3-4, pp. 198-204, 1976.

[12] K. G. Murty, *Linear Programming*. New York: Wiley, 1983.

[13] J. Telgen, "Minimal representation of convex polyhedral sets," *J. Opt. Theory Appl.*, vol. 38, no. 1, pp. 1-24, Sept. 1982.

Variable Structure Controller Design for Spacecraft Nutation Damping

HEBERTT SIRA-RAMIREZ AND THOMAS A. W. DWYER, III

Abstract—**Variable structure systems theory is used to design an automatic controller for active nutation damping in momentum biased stabilized spacecraft. Robust feedback stabilization of roll and yaw angular dynamics is achieved with prescribed qualitative characteristics which are totally independent of the spacecraft defining parameters.**

I. Introduction

The use of variable structure systems theory (hereafter called VSS) is receiving increased attention in the aerospace field as referenced in [1], [2]. The technique, extensively developed in the Soviet Union and Eastern Europe for a number of years [3], permits the use of a lower order system model for generating control commands, and is robust with respect to external disturbances as well as vehicle configuration and mass properties: indeed, the latter typically are needed only for estimates of the required level of control effort for the attainment of a desired "sliding motion" trajectory. The required accompanying switching logic, used for overshoot correction, is based only on the designer-selected sliding motion, as well as on invariant kinematic equations.

Recent results on the use of VSS for spacecraft slewing maneuvers are found in [1], [2], where global nonlinear methods are employed and in [4] where passive damping mechanisms during slewing are taken into account.

In this note, multivariable but linear VSS theory is used for active nutation damping. Nutation is defined as the rotational periodic motion exhibited by spacecraft when control or environmental disturbance torques perturb its stable spin-free equilibrium position (see also Wertz *et al.* [5]).

Nutation damping is accomplished in a passive manner by energy dissipation mechanisms such as: fluidic friction, used in connection with one or two degree of freedom penduli, generation of eddy currents arising from relative motion between a conducting plate and a magnet, or free rolling ball-in-the tube viscous friction dampers. On the other hand, active nutation dampers involve the use of feedback control in order to exercise

Manuscript received August 11, 1986; revised January 9, 1987. This work was supported by NASA under Grant NAG-1-436, by the National Science Foundation under Grant ECS-8516445, and by the Joint Services Electronics Program under Contract N00014-84-C-0149.

H. Sira-Ramirez is with the Department of Aeronautical and Astronautical Engineering and the Coordinated Science Laboratory, University of Illinois, Urbana, IL 61801, on leave from the Control Systems Department, Universidad de Los Andes, Merida, Venezuela.

T. A. W. Dwyer, III is with the Space Systems Laboratory, Department of Aeronautical and Astronautical Engineering, University of Illinois, Urbana, IL 61801.

IEEE Log Number 8613844.