JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 69, No. 3, JUNE 1991

Hyperplane Method for Reachable State Estimation for Linear Time-Invariant Systems[1]

T. J. GRAETTINGER[2] AND B. H. KROGH[3]

Communicated by M. Simaan

Abstract. A numerical algorithm is presented for generating inner and outer approximations for the set of reachable states for linear time-invariant systems. The algorithm is based on analytical results characterizing the solutions to a class of optimization problems which determine supporting hyperplanes for the reachable set. Explicit bounds on the truncation error for the finite-time case yield a set of so-called ϵ-supporting hyperplanes which can be generated to approximate the infinite-time reachable set within an arbitrary degree of accuracy. At the same time, an inner approximation is generated as the convex hull of points on the boundary of the finite-time reachable set. Numerical results are presented to illustrate the hyperplane method. The concluding section discusses directions for future work and applications of the method to problems in trajectory planning in servo systems.

Key Words. Reachable sets, simplicial approximation, linear systems, optimal control.

1. Introduction

A general nth order continuous-time dynamic system S can be described by the set of differential equations

$$\dot{x}(t) = f(x(t), u(t)), \qquad x(0) = x_0,$$

[1] This research was supported in part by Digital Equipment Corporation through the American Electronics Association Fellowship Loan Program and by the National Science Foundation under Grant No. ECS-84-04607.

[2] Graduate Student, Department of Electrical Engineering and Computer Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania.

[3] Associate Professor, Department of Electrical Engineering and Computer Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania.

where $x(t) \in E^n$ is the state vector, $u(t) \in E^m$ is the input vector, and $f: E^n \times E^m \to E^n$ is a continuously differentiable function that determines the evolution of the system state. The initial state x_0 is contained in an initial condition set $X_0 \subseteq E^n$. In the sequel, we adopt the convention that $x(t)$ denotes the state vector at a particular time t, and x denotes the state for all time $t \geq 0$. Further, given an input signal u, we let x_u be the corresponding state trajectory.

The input signal u is constrained pointwise in time such that $u(t) \in \Omega$, where $\Omega \subseteq E^m$. A class of input signals is then defined as

$$U_\Omega = \{u \in C_0 \mid u(t) \in \Omega, \forall t \geq 0\},$$

where C_0 is the set of all piecewise continuous functions. With these designations, we can define the reachable set at time T as

$$\Gamma_T(X_0, U_\Omega) = \{\zeta \in E^n \mid \exists u \in U_\Omega \text{ such that } \zeta = x_u(t), \text{for some } t \in [0, T]\}.$$

Hence, the reachable set $\Gamma_T(X_0, U_\Omega)$ is the subset of the n-dimensional state space that is reachable in time T. In the sequel, we omit the arguments X_0 and U_Ω when they are fixed, and denote the reachable set at time T simply as Γ_T.

The capability of the system S to be driven to a region of the state space, subject to the input bounds Ω, is characterized by the reachable set. For example, consider an optimal control problem for which the objective is to drive the system S from an initial state $x_0 \in X_0$ to a final state x_f in a target set $X_f \subseteq E^n$ in minimum time. The minimum time required to reach the set X_f is the first time T for which $\Gamma_T \cap X_f \neq \emptyset$. Conversely, suppose that a subset $A \subseteq E^n$ of the state space is to be avoided. If $\Gamma_T \cap A = \emptyset$, then any input $u \in U_\Omega$ guarantees avoidance of the subset A during the time interval $[0, T]$. The subset A can represent state operating constraints that must be satisfied for acceptable performance of the system S. To determine whether a given set of control bounds ensures satisfaction of such operating constraints, the reachable set for system S must be computed or approximated.

Recent analytical work on the theory of reachable sets concerns the determination of conditions under which a target set is reachable by a nonlinear control system (Ref. 1). The set is reachable or not reachable if the value of a particular convex optimization problem is zero or infinite, respectively. Locally Lipschitz functions are shown to permit arbitrarily accurate estimates of the reachable set for a nonlinear system in reference (Ref. 2). Under relatively weak assumptions, such functions allow an exact description of the reachable set. However, selection of appropriate, locally Lipschitz functions for the description has not been addressed.

Several approaches hve been taken to provide a useful description of the reachable set Γ_∞. Analytic techniques produce state trajectories on or asymptotically approaching the boundary of the reachable set. These methods are based on the reachability maximum principle and so-called abnormal control laws, which generate the appropriate state trajectories (Refs. 3–4). However, these approaches are limited to first- and second-order systems.

Other methods yield conservative outer approximations to the reachable set. The box method of Gayek (Ref. 5) decomposes (via the eigenvector modal matrix) an n-dimensional, stable, linear, time-invariant (LTI) system into one- and two-dimensional subsystems. The analytic results, described above, are applied to determine rectangular (boxlike) bounds in the one- and two-dimensional state spaces of the subsystems. These bounds are mapped linearly (via the modal matrix) into the original, n-dimensional state space, forming a parallelepiped that contains the reachable set Γ_∞, the reachable set as $T \to \infty$.

Grantham has proposed a technique of reachable set estimation applicable to both linear and nonlinear systems (Ref. 6). A Lyapunov-like boundary surface is defined, and a semi-infinite optimization problem is solved to determine the unknown parameters of the boundary surface. The region enclosed by this boundary surface contains the reachable set Γ_∞. A volume criterion has been applied by Summers to determine reachable set approximations via this approach (Refs. 7–8). No guidelines exist, however, for selecting an appropriate Lyapunov-like boundary surface; hence, the method tends to be very conservative.

In this paper, we describe a hyperplane method to generate reachable set estimates for a class of linear, time-invariant systems. The input u is a piecewise-continuous vector function of time, and it is constrained pointwise such that $u(t) \in \Omega$, where

$$\Omega = \{u(t) \in E^m \,|\, |u_j(t)| \leq \bar{u}_j, j = 1, \ldots, m\}.$$

The hyperplane method is based on a finite-time, optimal control problem, formulated as a search in a given direction of the state space. The solution to the problem yields a point on the boundary $\partial\Gamma_T$ and a supporting hyperplane to the reachable set Γ_T. By searching a number of directions in the state space, an outer approximation to the reachable set Γ_T is generated as the intersection of the reachable half-spaces determined by the supporting hyperplanes. This approach is illustrated in Fig. 1, where the reachable set Γ_T for a system S is shown as the shaded region. Several supporting hyperplanes forming an outer approximation are illustrated by the solid lines. Dashed lines depict the convex hull of the boundary points which comprises an inner approximation to the reachable set.

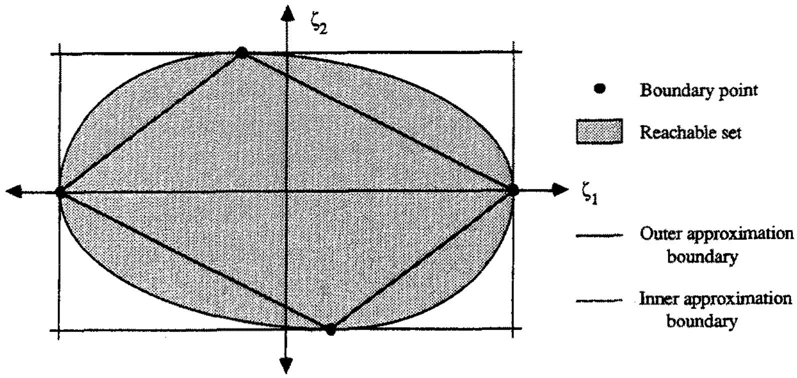

Fig. 1. Inner and outer approximations to the reachable set for a system S.

If the system S is stable, the finite-time results can be extended to approximate the infinite-time horizon case by computing a truncation error tolerance ϵ. An ϵ-supporting hyperplane is then determined which is parallel to an actual supporting hyperplane and within a distance ϵ of the boundary of the reachable set Γ_∞. The ϵ-supporting hyperplanes give an outer approximation to the reachable set Γ_∞. Thus, the outer approximation has a simple representation as a list of linear inequalities. An inner approximation to the reachable set Γ_∞ is obtained simultaneously as the convex hull of the points on the boundary of the reachable set. By selectively choosing directions for refinement, the accuracy of the outer approximation can be significantly improved with only a small number of ϵ-supporting hyperplanes. We employ a heuristic approach to determine search directions in the state space which best improve the inner and outer approximations to the reachable set Γ_∞.

To summarize, the hyperplane method of reachable state estimation that is described in the sequel has the following features. First, the hyperplane method is flexible since arbitrary directions in the state space can be searched, in contrast to the box method which obtains bounds only in n fixed directions. Second, both finite and infinite-time problems can be solved. Finally, an arbitrarily good approximation to the reachable set is obtained by searching sufficiently many directions in the state space and computing the corresponding supporting hyperplanes or ϵ-supporting hyperplanes.

Details of the hyperplane method are developed in the following sections. In Section 2, the optimal control problem for computing supporting hyperplanes and ϵ-supporting hyperplanes is formulated and solved. Section 3 presents a complete description of the hyperplane method as an algorithm. The generation of initial inner and outer approximations is specified. The selection of search directions for refinement is discussed, followed by the procedure for updating the inner and outer approximations. The criteria for terminating the algorithm is then described. A flowchart outlines the steps in the computations. To illustrate the algorithm, several examples are presented in Section 4. Inner and outer approximations are computed, and the results are plotted. The concluding section summarizes the contributions to this work and discusses directions for future research.

2. Computing ϵ-Supporting Hyperplanes

In this section, we develop the notions of supporting hyperplanes and ϵ-supporting hyperplanes for the reachable set Γ_T. We consider an LTI system S described by the state model

$$\dot{x}(t) = Ax(t) + Bu(t), \qquad x(0) = 0, \tag{1}$$

where the state vector is $x(t) \in E^n$, the input vector is $u(t) \in E^m$, and A, B are matrices of appropriate dimension. The n eigenvectors of A are assumed to be linearly independent; hence, A is diagonalizable. Thus, the modal matrix M, composed of the eigenvectors of A, is invertible. The input signal u is a piecewise-continuous vector function of time, constrained pointwise such that $u(t) \in \Omega$, where

$$\Omega = \{u(t) \in E^m \,|\, |u_j(t)| \leq \bar{u}_j, j = 1, \ldots, m\}. \tag{2}$$

Under these assumptions, the forward solution to the state equations is unique for any $u \in U_\Omega$.

We state the following well-known properties of the reachable set Γ_T without proof (Refs. 9–10).

Symmetry Property. If the control bounds Ω are symmetric with respect to the origin, the reachable set Γ_T is symmetric with respect to the origin.

Convexity Property. If the control bounds Ω are convex, the reachable set Γ_T is convex.

The control bounds (2) that we have specified are both symmetric and convex. We make a further normality assumption, namely, we assume

$$\operatorname{rank}[b_j | Ab_j | \cdots | A^{n-1} b_j] = n, \qquad \text{for all } j = 1, \ldots, m,$$

where b_j denotes the jth column of B. Under this assumption, the reachable set Γ_T is guaranteed to be a full n-dimensional body. Furthermore, this assumption guarantees that the time-optimal trajectories used in the theory of this section will not have singular arcs; i.e., the controls of interest will be bang-bang (Ref. 11).

A hyperplane in n-dimensional state space can be expressed in terms of a normal vector $c \in E^n$ and a scalar constant $v \in E^1$ as

$$c'\zeta = v, \tag{3}$$

where the dummy variable ζ is used to denote an arbitrary point in the state space. Without loss of generality, we assume $\|c\| = 1$. A supporting hyperplane to the convex set Γ_T in the direction c is tangent to Γ_T at a point ζ_c^* and satisfies (3), where $v = c'\zeta_c^*$, as shown in Fig. 2. An ϵ-supporting hyperplane in the direction c is given by

$$c'\zeta = v + \epsilon.$$

Hence, the ϵ-supporting hyperplane is parallel to the supporting hyperplane (since the normal vectors are identical).

Figure 2 depicts an ϵ-supporting hyperplane to the reachable set Γ_T at a distance ϵ from the supporting hyperplane. If the reachable set boundary $P\Gamma_T$ is not smooth, there are an infinite number of supporting hyperplanes at each boundary point where the surface is not smooth. This does not introduce difficulty, however, since the hyperplane direction is selected *a priori*. One boundary point may have an infinite number of supporting hyperplanes, but a supporting hyperplane in a specified direction corresponds to a unique boundary point.

The supporting hyperplane to Γ_T in the direction c can be determined from the solution of the following optimization problem:

$$(\mathrm{P(c)}) \quad \max_{\zeta} \quad c'\zeta,$$

$$\text{s.t.} \quad \zeta \in \Gamma_T.$$

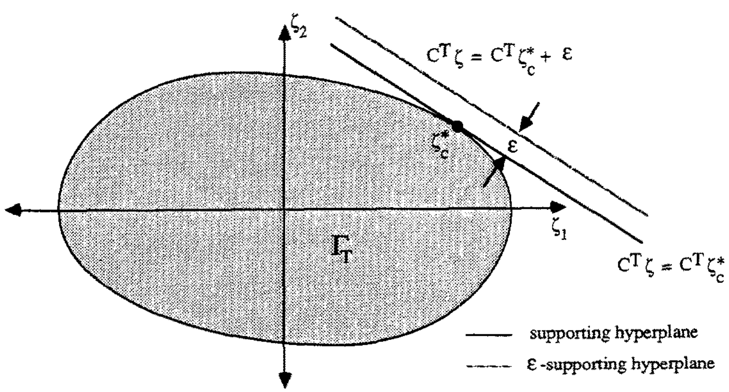

Fig. 2. Supporting hyperplane and ϵ-supporting hyperplane for the reachable set.

The objective function value $v = c'\zeta_c^*$ at the optimal solution ζ_c^* defines the supporting hyperplane. To compute a supporting hyperplane when the reachable set Γ_T is unknown, we reformulate the problem as an equivalent optimal control problem:

$$\text{(OCP(c))} \quad \max_{t\in[0,T]} \quad c'x(t),$$
$$\text{s.t.} \quad \hat{x}(t) = Ax(t) + Bu(t),\ x(0) = 0,$$
$$u(t) \in \Omega.$$

To pose Problem (OCP(c)) above as a standard, linear optimal control problem, we present the following result.

Lemma 2.1. Given times T_1, T_2, such that $0 < T_1 < T_2$, if a control signal u_1 on $[0, T_1]$ reaches the state $\zeta^* = x_{u_1}(T_1)$ at time T_1, then there exists a control signal u_2 on $[0, T_2]$ that reaches $\zeta^* = x_{u_2}(T_2)$ at time $T_2 \geq T_1$.

Proof. The final state $x_{u_1}(T_1)$ corresponding to the control signal $u_1(t)$ on $[0, T_1]$ is given as

$$x_{u_1}(T_1) = \int_0^{T_1} \exp[A(T_1 - \tau)]Bu_1(\tau)\, d\tau = \zeta^*.$$

Since the system is time-invariant, choosing the control input $u_2(t)$ to be

$$u_2(t) = \begin{cases} 0, & 0 \leq t \leq T_1 - T_2, \\ u_1(t - (T_2 - T_1)), & T_2 - T_1 \leq t \leq T_2, \end{cases}$$

drives the system to the same final state at T_2, which proves the lemma. □

The result of Lemma 2.1 implies that the optimal solution to Problem (OCP(c)) can always be made to occur at the final time T. Thus, (OCP(c)) can be expressed as a fixed, finite final time optimal control problem. We define this problem to be the supporting hyperplane problem:

$$\text{(SHP(c))} \quad \max \quad c'x(T),$$
$$\text{s.t.} \quad \dot{x}(t) = Ax(t) + Bu(t),\ x(0) = 0,$$
$$u(t) \in \Omega.$$

The solution to (SHP(c)) is presented in the following supporting hyperplane theorem.

Theorem 2.1. The final state $x^*_{u^*}(T)$ maximizes the performance index of (SHP(c)) if and only if the control signal u^* (expressed componentwise) satisfies

$$u_j^*(t) = \begin{cases} \bar{u}_j, & c' \exp[A(T-t)]b_j > 0, \\ -\bar{u}_j, & c' \exp[A(T-t)]b_j < 0, \\ -\bar{u}_j \leq u \leq \bar{u}_j, & c' \exp[A(T-t)]b_j = 0, \end{cases}$$

where the vector $b_j \in E^n$ is the jth column of the matrix B.

Proof. The Hamiltonian for Problem (SHP(c)) is

$$H = p^T(t)[Ax(t) + Bu(t)],$$

giving the necessary conditions

$$\dot{x}^*(t) = \partial H(x^*(t), u^*(t), p^*(t))/\partial p, \tag{4}$$

$$\dot{p}^*(t) = -\partial H(x^*(t), u^*(t), p^*(t))/\partial x, \tag{5}$$

$$H(x^*(t), u^*(t), p^*(t)) \geq H(x^*(t), u(t), p^*(t)),$$
$$\forall u \text{ such that } |u_j(t)| \leq \bar{u}_j, j = 1, \ldots, m, \tag{6}$$

with the boundary conditions

$$x^*(0) = 0, \tag{7}$$

$$p^*(T) = c. \tag{8}$$

The necessary conditions (4) and (5), coupled with the boundary conditions (7) and (8), respectively, give the state and costate equations which must be satisfied by an optimal solution. The costate equations have the solution

$$p^*(t) = \exp[A'(T-t)]c. \tag{9}$$

The necessary condition (6) requires $u^*(t)$ to satisfy

$$u_j^*(t) = \begin{cases} \bar{u}_j, & c' \exp[A(T-E)]b_j > 0, \\ -\bar{u}_j, & c' \exp[A(T-t)]b_j < 0, \\ -\bar{u}_j \leq u \leq \bar{u}_j, & c' \exp[A(T-t)]b_j = 0. \end{cases}$$

The assumption of normality eliminates the possibility of singular control (Ref. 12). Applying the optimal control input $u^*(t)$ gives

$$c'x^*_{u^*}(T) = \int_0^T c' \exp[A(T-\tau)]b_1 u_1^*(\tau)\, d\tau + \cdots$$
$$+ \int_0^T c' \exp[A(T-\tau)]b_m u_m^*(\tau)\, d\tau,$$

which is equivalent to

$$c'x_{u^*}^*(T)=\bar{u}_1\int_0^T |c'\exp[A(T-\tau)]b_1|\,d\tau+\cdots$$
$$+\bar{u}_m\int_0^T |c'\exp[A(T-\tau)]b_m|\,d\tau,$$

or

$$v(T)=c^Tx_{u^*}^*(T)=\sum_{j=1}^{\infty}\left\{\bar{u}_j\int_0^T |c'\exp[A(T-\tau)]b_j|\,d\tau\right\}.$$

This completes the proof. □

From the solution to (SHP(c)), we can construct a supporting hyperplane to the reachable set Γ_T. The final state $x_{u^*}^*(T)$ is well defined by the control signal u^*, and it is on the boundary of $\partial\Gamma_T$. The supporting hyperplane in the direction c is tangent to $\partial\Gamma_T$ at the point $x_{u^*}^*(T)$. It is expressed as $c'\xi=v$, where $v=c'x_{u^*}^*(T)$. For the control signal u^*, the final state $x_{u^*}^*(T)$ is given as

$$x_{u^*}^*(T)=\int_0^T c'\exp[A(T-\tau)]Bu^*(\tau)\,d\tau.$$

This expression permits the supporting hyperplane constant v to be determined as

$$v=c'x_{u^*}^*(\tau)=\int_0^T c'\exp[A(T-\tau)]Bu^*(\tau)\,d\tau,$$

which, upon substitution of the control law u^*, yields

$$v=\sum_{i=1}^{m}\left\{\bar{u}_i\int_0^T |c'\exp[A(T-\tau)]b_i|\,d\tau\right\}, \tag{10}$$

where b_i is the ith column of the matrix B. We remark that, in this form (10), the supporting hyperplane in the direction c can be determined by a scalar integration, without computing the entire state trajectory.

The foregoing analysis was performed for the case where a final time T_1 was finite. We now seek to determine an ϵ-supporting hyperplane for a final time $T_2>T_1$, parallel and within a perpendicular distance ϵ to the actual supporting hyperplane to the reachable set at time T_2. In this way, the results can be extended to the case where T_2 is infinite, and an estimate of the ultimate reachable set Γ_∞ is obtained. We first note that the optimal values of Problem (SHP(c)) are monotonically increasing with respect to the final time T.

Lemma 2.2. Given a direction c, let $v(T)$ be the maximal value of the objective function in (SHP(c)). Then, $v(T)$ is monotonically increasing with respect to the final time T. That is,

$$v(T_2) > v(T_1), \qquad T_2 > T_1.$$

Proof. Given the times $T_2 > T_1 < 0$, we have from the expression (10),

$$v(T_1) = \sum_{i=1}^{m} \left\{ \bar{u}_i \int_0^{T_1} |c' \exp[A(T_1 - \tau)] b_i| \, d\tau \right\}, \tag{11}$$

$$v(T_2) = \sum_{i=1}^{m} \left\{ \bar{u}_i \int_0^{T_2} |c' \exp[A(T_2 - \tau)] b_i| \, d\tau \right\}. \tag{12}$$

By splitting the integration at time $T_2 - T_1$, we can write (12) as

$$v(T_2) = \sum_{i=1}^{m} \left\{ \bar{u}_i \left[\int_0^{T_2 - T_1} |c' \exp[A(T_2 - \tau)] b_i| \, d\tau + \int_{T_2 - T_1}^{T_2} |c' \exp[A(T_2 - \tau)] b_i| \, d\tau \right] \right\}.$$

Under a change of variable, $\beta = \tau - (T_2 - T_1)$, this becomes

$$v(T_2) = \sum_{i=1}^{m} \left\{ \bar{u}_i \left[\int_0^{T_2 - T_1} |c' \exp[A(T_2 - \tau)] b_i| \, d\tau + \int_0^{T_1} |c' \exp[A(T_1 - \beta)] b_i| \, d\beta \right] \right\}.$$

We recognize the second term in the sum to be exactly the expression for $v(T_1)$. Hence,

$$v(T_2) = v(T_1) + \sum_{i=1}^{m} \left\{ \bar{u}_i \int_0^{T_2 - T_1} |c' \exp[A(T_2 - \tau)] b_i| \, d\tau \right\}. \tag{13}$$

Since

$$|c' \exp[A(T_2 - T_1)] b_i| > 0$$

on any interval of finite length (Ref. 13), Eq. (13) implies

$$v(T_2) > v(T_1), \qquad T_2 > T_1,$$

which completes the proof. □

The proof of the monotonicity property suggests an approach for determining ϵ-supporting hyperplanes. Defining a truncation error

$$\epsilon(T_1, T_2) = v(T_2) - v(T_1),$$

the result (13) gives

$$\epsilon(T_1, T_2) = \sum_{i=1}^{m} \left\{ \bar{u}_i \int_0^{T_2 - T_1} |c' \exp[A(T_2 - \tau)] b_i| \, d\tau \right\}.$$

A bound on this truncation error can be determined as a function of the final time T_1 only, as stated in the theorem below.

Theorem 2.2. If the system in Problem (SHP(c)) is stable, and if the matrix A has n linearly independent eigenvectors which form a modal matrix M, then the truncation error can be bounded as

$$\epsilon(T_1, T_2) \le \sum_{i=1}^{m} \left\{ \bar{u}_i \left[\sum_{k=1}^{n} \|d_{k,i}\| \exp[\mathrm{Re}\{\lambda_k\} T_1] / |\mathrm{Re}\{\lambda_k\}| \right] \right\},$$

where λ_k, $k = 1, \ldots, n$, are the eigenvalues of matrix A. The (in general) complex coefficients $d_{k,i}$ are defined as

$$d_{k,i} = (c' m_k)(\hat{m}_k^T b_i),$$

with m_k denoting the kth column of the modal matrix M and $\hat{m}_k'$ being the kth row of M^{-1}.

Proof. The truncation error is written originally as

$$\epsilon(T_1, T_2) = \sum_{i=1}^{m} \left\{ \bar{u}_i \int_0^{T_2 - T_1} |c' \exp[A(T_2 - \tau)] b_i| \, d\tau \right\}. \tag{14}$$

Writing $\exp[A(T_2 - \tau)]$ as

$$\exp[A(T_2 - \tau)] = M \exp[\Lambda(T_2 - \tau)] M^{-1},$$

where M is the modal matrix of the n linearly independent eigenvectors, and Λ is a diagonal matrix of eigenvalues, we have

$$c' \exp[A(T_2 - \tau)] b_i = \sum_{k=1}^{n} d_{k,i} \exp[\lambda_k (T_2 - \tau)], \tag{15}$$

where λ_k is the kth diagonal element (eigenvalue) of Λ and

$$d_{k,i} = (c' m_k)(\hat{m}_k' b_i),$$

with m_k denoting the kth column of M and $\hat{m}_k'$ being the kth row of M^{-1}.

Partitioning the terms in (15) yields

$$\begin{aligned} &\sum_{k=1}^{n} d_{k,i} \exp[\lambda_k (T_2 - \tau)] \\ &= \sum_{kcc=1}^{ncc} (d_{kcc,i} \exp[\lambda_{kcc}(T_2 - \tau)] + d_{kcc,i}^{*} \exp[\lambda_{kcc}^{*}(T_2 - \tau)]) \\ &+ \sum_{kr=1}^{nr} (d_{kr,i} \exp[\lambda_{kr}(T_2 - \tau)]), \end{aligned} \tag{16}$$

where ncc and nr are the numbers of pairs of complex conjugate and real eigenvalues, respectively. The first sum in (16) is composed of the complex conjugate eigenvalues λ_{kcc} and λ^*_{kcc}, with the corresponding complex conjugate coefficients $d_{kcc,i}$ and $d^*_{kcc,i}$, respectively. The real eigenvalues λ_{kr} in the second sum have real coefficients $d_{kr,i}$. The total number of eigenvalues is $n=2ncc+nr$.

Writing the complex coefficients in polar form [e.g., $d_{kcc,i}=\delta_{kcc,i}\exp(j\theta_{kcc,i})$] and the complex eigenvalues in rectangular form [e.g., $\lambda_{kcc}=\mu_{kcc}+j\omega_{kcc}$], Eq. (16) becomes, upon simplification,

$$\begin{aligned}&\sum_{k=1}^{n} d_{k,i}\exp[\lambda_k(T_2-\tau)]\\&=2\sum_{kcc=1}^{ncc}\bigg\{\delta_{kcc,i}\exp[\mu_{kcc}(T_2-\tau)]\cos(\omega_{kcc}(T_2-\tau)+\theta_{kcc,i})\\&\qquad\qquad+\sum_{kr=1}^{nr}\{d_{kr,i}\}\exp[\lambda_{kr}(T_2-\tau)]\bigg\}.\end{aligned}$$

Substituting this result into expression (14) gives the truncation error as

$$\begin{aligned}&\epsilon(T_1,T_2)\\&\le\sum_{i=1}^{m}\bigg\{\bar{u}_i\bigg[\int_0^{T_2-T_1}\bigg|2\sum_{kcc=1}^{ncc}\delta_{kcc,i}\exp[\mu_{kcc}(T_2-\tau)]\cos(\omega_{kcc}(T_2-\tau)+\theta_{kcc,i})\\&\qquad\qquad+\sum_{kr=1}^{nr}d_{kr,i}\exp[\lambda_{kr}(T_2-\tau)]\bigg|\,d\tau\bigg]\bigg\}.\end{aligned}$$

An upper bound on the truncation error can be obtained by taking the absolute value of each sum in the integrand, yielding

$$\begin{aligned}&\epsilon(T_1,T_2)\\&\le\sum_{i=1}^{m}\bigg\{\bar{u}_i\bigg[\int_0^{T_2-T_1}2\sum_{kcc=1}^{ncc}|\delta_{kcc,i}\exp[\mu_{kcc}(T_2-\tau)]|\,|\cos(\omega_{kcc}(T_2-\tau)+\theta_{kcc,i})|\\&\qquad\qquad+\sum_{kr=1}^{nr}|d_{kr,i}\exp[\lambda_{kr}(T_2-\tau)]|\,d\tau\bigg]\bigg\}.\end{aligned}$$

Since

$$|\cos(\omega_{kcc}(T_2-\tau)+\theta_{kcc,i})|\le 1,$$

we have

$$\begin{aligned}&\epsilon(T_1,T_2)\\&\le\sum_{i=1}^{m}\bigg\{\bar{u}_i\bigg[\int_0^{T_2-T_1}2\sum_{kcc=1}^{ncc}|\delta_{kcc,i}\exp[\mu_{kcc}(T_2-\tau)]|\\&\qquad\qquad+\sum_{kr=1}^{nr}|d_{kr,i}\exp[\lambda_{kr}(T_2-\tau)]|\,d\tau\bigg]\bigg\}.\end{aligned}$$

Performing the indicated integration, we obtain

$$\epsilon(T_1, T_2) \leq \sum_{i=1}^{m} \left\{ \bar{u}_i \left[2 \sum_{kcc=1}^{ncc} |\delta_{kcc,i}|(-\exp[\mu_{kcc}T_1] + \exp[\mu_{kcc}(T_2)])/\mu_{kcc} + \sum_{kr=1}^{nr} |d_{kr,i}|(-\exp[\lambda_{kr}T_1] + \exp[\lambda_{kr}T_2])/\lambda_{kr} \right] \right\}. \quad (17)$$

Under the assumption of stability, $\mu_{k_{cc}} < 0$, $\lambda_{k_r} < 0$, and

$$\lim_{T_2 \to \infty} \exp[\mu_{kcc}T_2] = \lim_{T_2 \to \infty} \exp[\lambda_{kr}T_2] = 0,$$

for all k_{cc}, k_r, respectively. Thus, the worst case is obtained as $T_2 \to \infty$ for which

$$\epsilon(T_1, T_2) = \epsilon(T_1) \leq \sum_{i=1}^{m} \left\{ \bar{u}_i \left[2 \sum_{kcc=1}^{ncc} |\delta_{kcc,i}/\mu_{kcc}| \exp[\mu_{kcc}T_1] + \sum_{kr=1}^{nr} |d_{kr,i}/\lambda_{kr}| \exp[\lambda_{kr}T_1] \right] \right\}.$$

We recall that

$$\mu_{kcc} = \mathrm{Re}\{\lambda_{kcc}\} = \mathrm{Re}\{\lambda^*_{kcc}\},$$

$$\delta_{kcc,i} = \|d_{kcc,i}\| = \|d^*_{kcc,i}\|,$$

where $\|\cdot\|$ denotes the complex magnitude. With this, we have the final result

$$\epsilon(T_1, T_2) \leq \sum_{i=1}^{m} \left(\bar{u}_i \left[\sum_{k=1}^{n} \|d_{k,i}\| \exp[\mathrm{Re}\{\lambda_k\}T_1]/|\mathrm{Re}\{\lambda_k\}| \right] \right),$$

which completes the proof. □

Obtaining a bound on the truncation error that is independent of the final time permits the finite time results to be extended to the infinite time case. The ϵ-supporting hyperplane can then be determined in a straightforward way as stated in the following ϵ-supporting hyperplane theorem.

Corollary 2.1. If $c'\zeta = v$ is a supporting hyperplane in the direction c to the reachable set Γ_{T_1} at final time T_1, then $c'\zeta = v + \epsilon(T_1, T_2)$ is an ϵ-supporting hyperplane to the reachable set Γ_{T_2} at final time $T_2 > T_1$, including $T_2 \to \infty$.

Proof. This follows directly from the definition of the truncation error $\epsilon(T_1, T_2)$ in Theorem 2.2. □

Another interesting outcome of this error analysis is that an estimate of Γ_∞ can be obtained without integration of any kind. By evaluating the truncation error for $T_1 = 0$, where $v = c^T x(0) = 0$, the ϵ-hyperplane boundary in the direction c is given by

$$c'\zeta = \sum_{i=1}^{m} \left\{ \bar{u}_i \sum_{k=1}^{n} \|d_{k,i}\| / |\mathrm{Re}\{\lambda_k\}| \right\}.$$

This conservative estimate is improved, of course, by solving (SHP(c)) for some final time T_1 and adding the corresponding truncation error term.

Certain applications require computation of a set of ϵ-supporting hyperplanes that are guaranteed to be within a single, given bound $\bar{\epsilon}$ of the actual supporting hyperplanes. It is useful to determine a fixed final time T_1 which ensures that $\epsilon(T_1, T_2) \leq \bar{\epsilon}$ for any such direction in the state space. Toward this end, we present the following result.

Theorem 2.3. Given an arbitrary vector $c \in E^n$ and a truncation error bound $\bar{\epsilon} > 0$, the truncation error is $\epsilon(T_1, T_2) \leq \bar{\epsilon}$ if

$$T_1 \geq |\mathrm{Re}\{\lambda^*\}|^{-1} \log[(\mathrm{cond}(M) S_{u,b})/(\bar{\epsilon}|\mathrm{Re}\{\lambda^*\}|)],$$

where $\mathrm{cond}(M)$ is the condition number of the modal matrix M, S_{ub} is given by

$$S_{ub} = \sum_{i=1}^{m} \bar{u}_i \|b_i\|,$$

and the eigenvalue λ^* satisfies

$$\mathrm{Re}\{\lambda^*\} \geq \lambda_k, \qquad k = 1, \ldots, n.$$

Proof. The expression for $\epsilon(T_1, T_2)$ is

$$\epsilon(T_1, T_2) = \sum_{i=1}^{m} \bar{u}_i \int_0^{T_2 - T_1} |c' \exp[A(T_2 - \tau)] b_i| \, d\tau,$$

where $|c' \exp[A(T_2 - \tau)] b_i|$ can be bounded as

$$\begin{aligned} &|c'M \exp[\Lambda(T_2 - \tau)] M^{-1} b_i| \\ &\leq \|c\| \, \|M\| \, \|\exp[\Lambda(T_2 - \tau)]\| \, \|M^{-1}\| \, \|b_i\|. \end{aligned}$$

The norm $\|\cdot\|$ of a matrix F is denoted by its maximum singular value $\sigma_{\max}$, where

$$\sigma_{\max}(F) = \max_{\|x\|=1} \|Fx\|.$$

Thus, with $\|c\|$ normalized to 1 always,

$$|c'M \exp[\Lambda(T_2-\tau)]M^{-1}b_i|$$
$$\leq \sigma_{\max}(M)\sigma_{\max}(\exp[\Lambda(T_2-\tau)])\sigma_{\max}(M^{-1})\|b_i\|.$$

The singular values $\sigma_{\max}(M^{-1})$ and $\sigma_{\min}(M)$ (the minimum singular values of M) are related such that

$$\sigma_{\max}(M^{-1})=1/\sigma_{\min}(M).$$

Hence, with the condition number of the matrix M given as

$$\mathrm{cond}(M)=\sigma_{\max}(M)/\sigma_{\min}(M),$$

we have

$$|c'M \exp[\Lambda(T_2-\tau)]M^{-1}b_i|\leq \mathrm{cond}(M)\sigma_{\max}(\exp[\Lambda(T_2-\tau)])\|b_i\|.$$

The maximum singular value of $\exp\{\Lambda(T_2-\tau)\}$ is given by

$$\sigma_{\max}(\exp[\Lambda(T_2-\tau)])=\exp[\mathrm{Re}\{\lambda^*\}(T_2-\tau)],$$

where the eigenvalue λ^* satisfies

$$\mathrm{Re}\{\lambda^*\}\geq \lambda_k, \qquad k=1,\ldots,n.$$

Thus,

$$\int_0^{T_2-T_1} |c' \exp[A(T_2-\tau)]b_i|\, d\tau$$
$$\leq \mathrm{cond}(M)\|b_i\|(\exp[\mathrm{Re}\{\lambda^*\}T_2]-\exp[\mathrm{Re}\{\lambda^*\}T_1])/\mathrm{Re}\{\lambda^*\}.$$

For a stable system, $\mathrm{Re}\{\lambda^*\}<0$ and

$$\lim_{T_2\to\infty} \exp[\mathrm{Re}\{\lambda^*\}T_2]=0,$$

which implies that

$$\int_0^{T_2-T_1} |c' \exp[A(T_2-\tau)]b_i|\, d\tau$$
$$\leq \mathrm{cond}(M)\|b_i\|(\exp[\mathrm{Re}\{\lambda^*\}T_1])/|\mathrm{Re}\{\lambda^*\}|.$$

Finally, we define the sum

$$S_{ub}=\sum_{i=1}^{\infty} \bar{u}_i\|b_i\|.$$

To bound $\epsilon(T_1)$ by a specified value $\bar{\epsilon}$, we require

$$|\mathrm{Re}\{\lambda^*\}|^{-1}\, \mathrm{cond}(M)S_{ub} \exp[\mathrm{Re}\{\lambda^*\}T_1]\leq \bar{\epsilon}.$$

Taking logarithms and simplifying, the integration time required to satisfy the error bound $\bar{\epsilon}$ is

$$T_1 \geq |\mathrm{Re}\{\lambda^*\}|^{-1} \log[(\mathrm{cond}(M) S_{ub})/(\bar{\epsilon}|\mathrm{Re}\{\lambda^*\}|)],$$

completing the proof. □

3. Hyperplane Method for Reachable State Estimation

Throughout the following development, the symmetry of the reachable set will be exploited, and we will work with the half-body only. In the previous section, a boundary point $x^*(T)$ and an ϵ-supporting hyperplane,

$$c'\zeta = c'x^*(T) + \epsilon,$$

were determined from the solution to the optimal control problem (SHP(c)) and from the truncation error bound for a given direction c in the state space. By solving (SHP(c)) for the unique directions c_i, $i = 1, \ldots, n_{bp}$, a set of boundary points is obtained as

$$B = \{x_i^*(T), \ldots, x_{n_{bp}}^*(T)\}.$$

To form an outer approximation to the reachable set Γ_∞, we form the intersection of the halfspaces defined by the corresponding ϵ-supporting hyperplanes to yield

$$\bar{\Gamma} = \{\zeta \in E^n \mid c_i'\zeta \leq c_i'x_i^*(T) + \epsilon, i = 1, \ldots, n_{bp}\}.$$

The set of boundary points can be employed to give an inner approximation to the reachable set Γ_∞. Specifically, the convex hull $\underline{\Gamma}$ of the points in B, denoted by

$$\underline{\Gamma} = \mathrm{conv}\, H\{B\},$$

is contained in Γ_∞ since Γ_∞ is convex and since any point reachable at time T is also reachable at any time $T_1 > T$, by the result of Lemma 2.1.

The objective of the hyperplane method is to determine both inner and outer approximations to the reachable set Γ_∞. To obtain such approximations, it is necessary to select directions for search in the state space. We employ a heuristic approach that has proven successful in design centering problems for VLSI fabrication (Refs. 14–15). First, a set of search directions is chosen until a closed outer approximation is found. Then, to improve the approximation, the largest simplex (face) of the inner approximation $\underline{\Gamma}$ is chosen to be broken, since it is assumed to be the worst approximation to the reachable set boundary. The outward normal to the simplex is selected as the direction for refinement. Solving the optimal control problem

(SHP(c)) yields a new boundary point and an ϵ-supporting hyperplane. This procedure is illustrated in Fig. 3, where simplex s_1 is the largest face of the current inner approximation to the reachable set. A search is performed in the direction of the outward normal η_1. The boundary point $x^*(T)$ determines a new ϵ-supporting hyperplane (solid line) for the outer approximation, parallel to the broken face of the inner approximation. The new inner approximation (enclosed by the dashed lines) incorporates the new boundary point x^* in the convex hull.

In the remainder of this section, we present the computational steps required to implement the hyperplane method. A flowchart of the algorithm is described, noting the initialization, refinement, and stopping procedures. Also detailed are the necessary steps to perform the direction selection and the inner and outer approximation set updates.

The flowchart presented in Fig. 4 depicts the iterations of the hyperplane method algorithm. The system matrices A and B are assumed to be known. A vector $\bar{u}$ of input bounds defining Ω is given, along with the permissible truncation error ϵ used to form the ϵ-supporting hyperplanes. From these values, the initialization procedure generates an initial list of boundary points B^0, an initial inner approximation $\underline{\Gamma}^0$ to the halfbody Γ_∞, and an initial outer approximation $\bar{\Gamma}^0$. These results can be accomplished in several ways. In our implementation of the algorithm, the initialization procedure uses the unit direction vector c_i corresponding to the ith coordinate direction. The boundary point x_i^* is then determined from the solution[4] to (SHP(c_i)). This boundary point x_i^* and its symmetric counterpart $-x_i^*$ are entered in

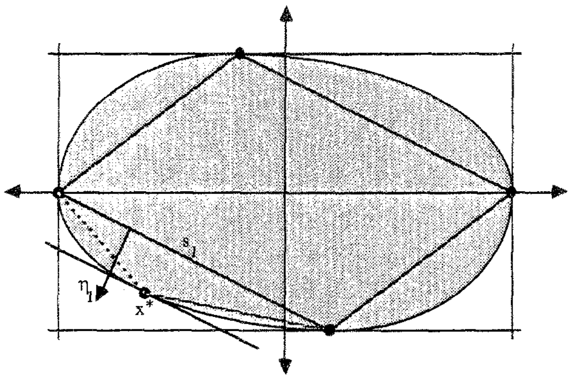

Fig. 3. Inflation of the inner approximation and cutting of the outer approximation to the reachable set (single step).

[4] The final time dependence of the boundary points is omitted in the sequel for notational convenience.

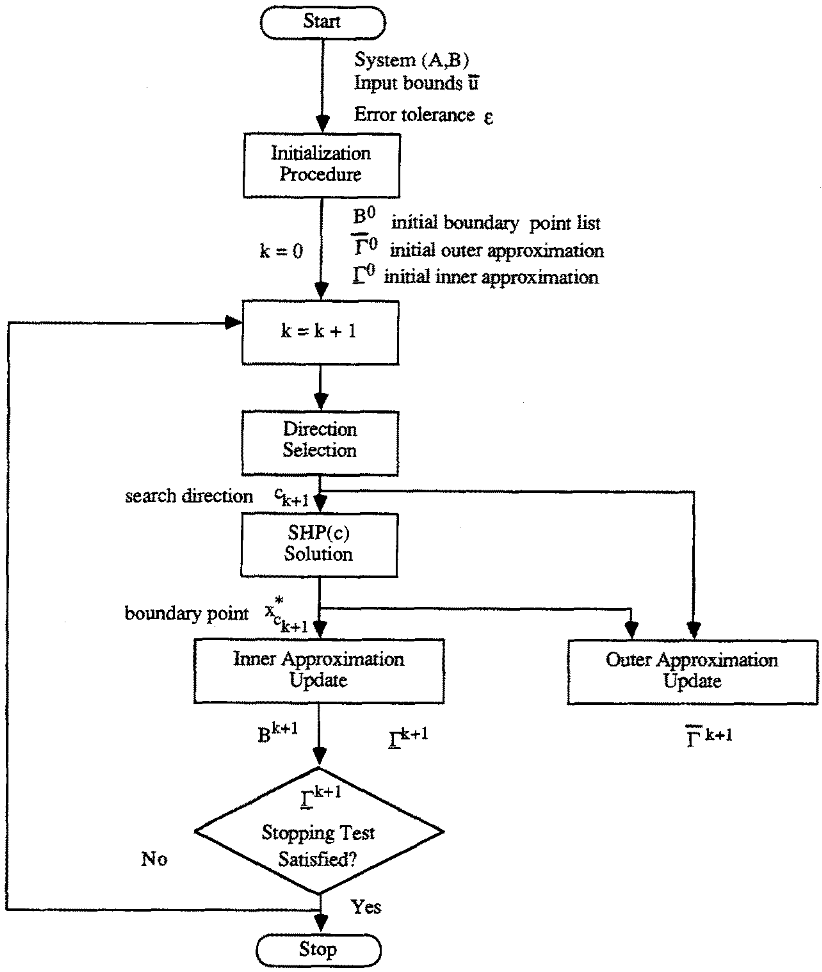

Fig. 4. Flowchart of the hyperplane method algorithm.

the list B^0 as x_i^* and x_{i+n}^*, respectively. An ϵ-supporting hyperplane $c_i'\zeta = c_i'x_i^* + \epsilon$ is formed, and the outer approximation $\bar{\Gamma}^0$ is updated by appending the appropriate halfspace, represented by the inequality

$$c_i'\zeta \geq c_i'x_i^* + \epsilon.$$

This process is repeated n times, i.e., once for each of the n coordinate directions. As a result, $2n$ boundary points are contained in the completed initial boundary point set B^0, and n inequalities form the initial outer approximation $\bar{\Gamma}^0$.

The initial inner approximation (convex hull) $\underline{\Gamma}^0$ is formed only at the conclusion of the n coordinate direction searches. Each simplex s_j, $j=1,\ldots,h_s$, of the inner approximation is represented by a vector of n integers corresponding to the indices of the n boundary points that form its vertices (where n is the dimension of the state space). For example, in Fig. 5, simplex s_1 is defined by boundary points x_1^* and x_2^*, and simplex s_2 is defined by boundary points x_2^* and x_3^*. Hence, these simplices are represented by integer vectors as

$$s_1=[1,2],$$
$$s_2=[2,3].$$

The ith element of s_j is denoted by $s_j(i)$.

For the initialization procedure, the simplices are formed straightforwardly using a table of binary combinations as illustrated in Table 1 for the case $n=4$. In this way, 2^{n-1} simplex vectors are formed to describe the initial inner approximation $\underline{\Gamma}(0)$.

To complete the initialization procedure and to prepare for subsequent iterations, it is necessary to determine an analytic expression for each simplex of the inner approximation. The equation of the simplex s_j can be written as

$$\eta_j'\zeta=\kappa_j,$$

where $\eta_i \in E^n$ is the vector (outward) normal to simplex s_j and $\kappa_j \in E^1$ is a (positive) constant. From the n boundary points $x_i^* \in B^0$, $i \in s_j$ (which compose simplex s_j), the normal vector η_j can be computed by first forming $n-1$ difference vectors $d_i \in E^n$ as

$$d_i=(x^*_{s_j(i+1)}-x^*_{s_j(i)}), \qquad i=1,\ldots,n-1.$$

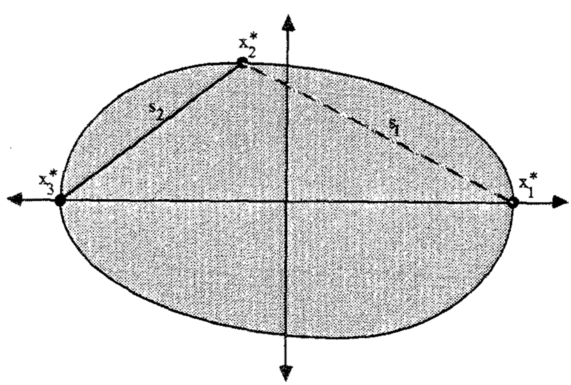

Fig. 5. Simplices forming the convex hull for the inner approximation.

Table 1. Tableau for forming the initial inner approximation from $2n$ initial boundary points.

Boundary points				Simplex
x_1^*	x_2^*	x_3^*	x_4^*	s_1
x_5^*	x_2^*	x_3^*	x_4^*	s_2
x_1^*	x_6^*	x_3^*	x_4^*	s_3
x_5^*	x_6^*	x_3^*	x_4^*	s_4
x_1^*	x_2^*	x_7^*	x_4^*	s_5
x_5^*	x_2^*	x_7^*	x_4^*	s_6
x_1^*	x_6^*	x_7^*	x_4^*	s_7
x_5^*	x_6^*	x_7^*	x_4^*	s_8

Calculating the cross product of the $n-1$ difference vectors d_i produces a vector $\hat{\eta}_j$ that is perpendicular to the difference vectors, and hence normal to the simplex. In Fig. 6, an example in 3-space is illustrated. Simplex s_1 is defined by boundary points x_1^*, x_2^*, x_3^*. Forming the difference vectors d_1, d_2 and determining the cross product yields the normal vector η_1, perpendicular to the face of s_1.

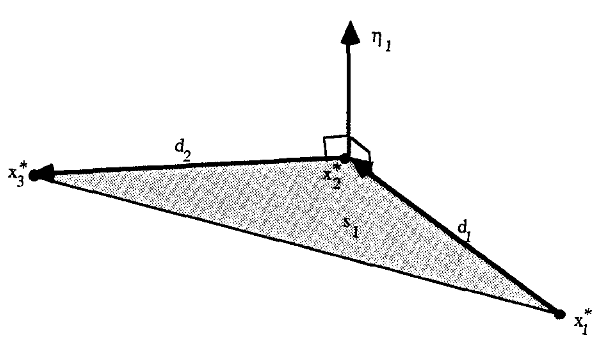

Fig. 6. Cross product vectors normal to the simplex faces.

Taking the inner product of $\hat{\eta}_j$ with any boundary point $x_i^* \in B^0$, $i \in s_j$, gives (for example)

$$\kappa_j = \hat{\eta}_j x^*_{s_j(1)}.$$

If $\kappa_j > 0$, then $\hat{\eta}_j$ is an outward normal, $\eta_j = \hat{\eta}_j$, and the constant κ_j is unchanged. Otherwise, we obtain the outward normal as $\eta_j = -\hat{\eta}_j$ and the constant κ_j is negated, i.e., $\kappa_j = -\kappa_j$. The main reason for determining the simplex normal via the cross product calculation is that the magnitude of $\hat{\eta}_j$ for simplex s_j is the size (volume) of the simplex. Determining the vector normals η and the constants κ for each of the 2^{n-1} initial simplices completes the initialization procedure.

We now describe the refinement procedure which is outlined in the flowchart of Fig. 4. At the conclusion of step k, a list of boundary points B^k, an inner approximation $\underline{\Gamma}^k$, and an outer approximation $\bar{\Gamma}^k$ are available. The number of boundary points contained in B^k is denoted by n_{bp}, and the number of inequalities (halfspaces) forming the outer approximation $\bar{\Gamma}^k$ is given by n_h. The counter k is incremented to $k+1$, and the refinement procedure begins with a direction selection step. The largest ~~simplex~~ of the inner approximation set (convex hull) is presumed to be the worst approximation to the reachable set boundary. The simplex s_{j^*} having the greatest normal vector magnitude is thus selected to be broken, and the normal vector η_{j^*} provides the direction for refinement of the inner and outer approximations via the solution of (SHP(c)).

Solving the optimal control problem (SHP(c)) for the normalized direction $c_{k+1} = \eta_{j^*}/\|\eta_{j^*}\|$ yields a new boundary point $x^*_{c_{k+1}}$. The number of boundary points is incremented by one (i.e., $n_{bp} = n_{bp} + 1$), and the new point is included in the boundary point list B^{k+1}. A new ϵ-supporting hyperplane is given by

$$c'_{k+1}\zeta = c'_{k+1}x^*_{c_{k+1}} + \epsilon.$$

The corresponding halfspace is represented by the inequality

$$c'_{k+1}\zeta \le c'_{k+1}x^*_{c_{k+1}} + \epsilon. \tag{18}$$

The number of halfspaces is incremented by one (i.e., $n_h = n_h + 1$), and constraint (18) is appended to the current list $\bar{\Gamma}^k$ to form the updated outer approximation $\bar{\Gamma}^{k+1}$.

To update the inner approximation set $\underline{\Gamma}^k$, n new simplices are created. They are formed as all combinations of the n integers specifying the boundary points of the largest simplex s_{j^*}, taken with the integer n_{bp} representing the new boundary point. The list of simplex vectors becomes

$$s_j, \qquad j = 1, \ldots, n_s, n_s + 1, \ldots, n_s + n - 1,$$

where the simplex s_{j^*} is replaced with a new simplex, and the other $n-1$ new simplices are appended to the list.

The new simplex vectors are formed by n single replacements of the integer elements of s_{j^*} with the integer n_{bp} corresponding to the new boundary point. That is,

$$s_{ns+k} = s_{j^*}, \qquad k = 1, \ldots, n-1,$$

with

$$s_{ns+k}(k) = n_{bp},$$

and the simplex s_{j^*} is replaced by an identical integer vector with the single modification

$$s_{j^*}(n) = n_{bp}.$$

For each of the n newly created simplices, an outward normal vector η_j is computed via the difference vector/cross product procedure described above. The constant κ_j is also computed to ensure that η_j is indeed an outward normal. This completes the update procedure for the inner approximation set.

As an illustration, consider the list of $n_s = 2$ simplices

$$s_1 = [1, 2],$$

$$s_2 = [2, 3],$$

whre s_1 is to be broken. A new boundary point x_4^* is determined, and n_{bp} is incremented such that $n_{bp} = 4$. Simplex s_3 is created as

$$s_3 = [4, 2]$$

by copying s_1 and replacing the first element with $n_{bp} = 4$. The broken plane s_1 is modified such that the element $n = 2$ is replaced by $n_{bp} = 4$ to give

$$s_1 = [1, 4].$$

The number of simplices is incremented by $n - 1 = 1$ such that $n_s = 3$, the normal vectors η and the constants κ are computed for the newly created simplices, and the update of the inner approximation is finished.

We note that each step in the hyperplane method yields one new hyperplane for the outer approximation, one new boundary point, and new simplices for the inner approximation. Hence, the number of simplices can become very large for large n or for a large number of iterative refinements. This problem is always present in numerical methods when a nonlinear boundary is approximated with linear elements, for example, when finite element meshes are used to fill curved boundaries.

At the conclusion of the refinement procedure, a stopping criterion is evaluated. We use the following criterion: if all simplices forming the inner approximation are less than a user-specified size; i.e., if the normal vector magnitudes are less than a user-specified size, then stop. Otherwise, continue the refinement and select a new direction for search. This approach permits the reachable set to be approximated with a relatively uniform granularity with both inner and outer approximations. Other approaches for terminating the algorithm are discussed in the concluding section.

4. Examples

In this section, the hyperplane method is applied to two LTI example systems. One is a second-order, two-input system, and the other is a third-order, single-input system. The examples illustrate the algorithmic steps in the hyperplane method. The initialization procedure generates the initial inner and outer approximations. Refinement of the reachable set estimates continues via the direction selection and approximation set updates described in the previous section. Iterations are terminated when the stopping criteria is satisfied. Once again, since the reachable set is symmetric, we work only with the half-body.

Example 4.1. A linear, time-invariant system S having real eigenvalues $[\lambda_1, \lambda_2] = [-2.0, -1.0]$ is described by the state equations

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix},$$

where $x(t) \in E^2$ is the state vector and $u(t) \in E^2$ is the input vector. The input is bounded such that

$$|u_l(t)| \leq 1.0, \qquad l = 1, 2.$$

For this problem, an estimate of Γ_∞ is sought. A truncation error less than 0.001 is required, fixing the final time at

$$T = 9.45 \text{ sec},$$

by the bound determined in Section 2.

Having specified the system and the problem parameters, the initial inner and outer approximations are determined as follows. A search is performed in the coordinate direction $c_1' = [1, 0]$ to yield the boundary point

$$x_{c_2}^* = [1.999, -0.999]',$$

its symmetric counterpart

$$-x^*_{c_1} = [-1.999, 0.999]',$$

and the ϵ-supporting hyperplane

$$c_1'\zeta = 2.000$$

from the solution of (SHP(c_1)). Selecting the other positive coordinate direction $c_2' = [0, 1]$ and solving (SHP(c_2)) gives the boundary point

$$x^*_{c_2} = [-1.749, 1.499]',$$

its symmetric counterpart

$$-x^*_{c_2} = [1.749, -1.499]',$$

and the ϵ-supporting hyperplane

$$c_2'\zeta = 1.507.$$

The initial outer approximation to Γ_∞ is determined as the intersection of the halfspaces defined by the ϵ-supporting hyperplanes. Specifically, $\bar{\Gamma}^0$ is the set of $\zeta \in E^2$ is defined in Table 2 and displayed graphically in Fig. 7 as the region enclosed by the solid lines.

Table 3 lists the initial set B^0 of boundary points x^*. The inner approximation $\underline{\Gamma}^0$ is formed as the symmetric convex hull of the boundary points. The simplices s_1 and s_2 are thus specified by the integer vectors which contain the indices of the boundary points that are vertices. The normal vectors for each simplex and their corresponding magnitudes (simplex sizes) are given in Table 4. This halfbody inner approximation is enclosed by the dashed lines shown in Fig. 7.

To refine the inner/outer reachable set estimates, a direction for computing a new supporting hyperplane is selected as the largest simplex normal vector, which yields $c_3' = [0.5546, 0.8321]$ upon normalization of $[2.498, 3.748]$. Hence, simplex s_1 will be broken and replaced by $n = 2$ new planes. Solving (SHP(c_3)) gives the boundary point

$$x^*_{c_3} = [-0.1645, 0.8593]'$$

and the ϵ-supporting hyperplane

$$c_3\zeta = 0.6257.$$

Table 2. Outer approximation $\bar{\Gamma}^0$.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \end{bmatrix} \le \begin{bmatrix} 2.000 \\ 1.507 \end{bmatrix}$$

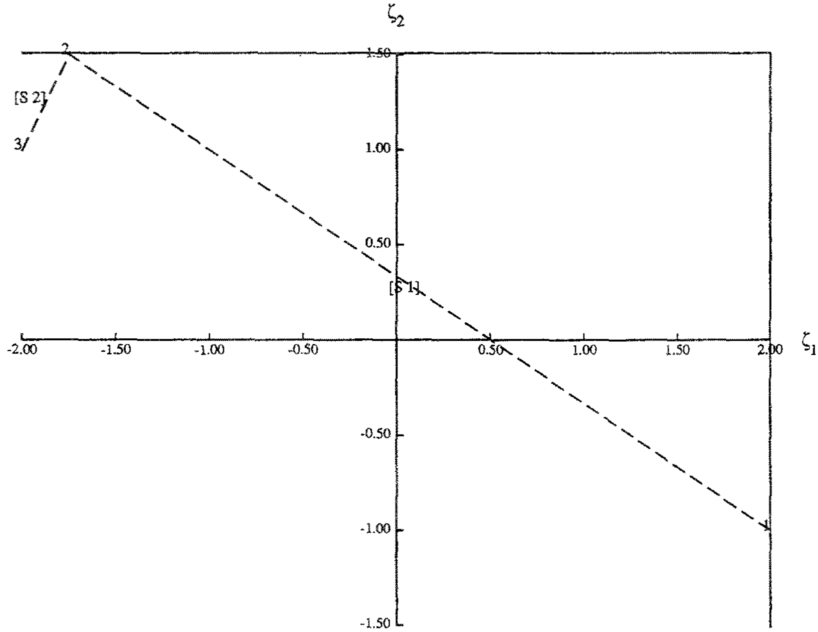

Fig. 7. Initial inner (dotted) and outer (solid) approximations to the reachable set halfbody.

Table 3. Boundary point list B^0.

Boundary point	Coordinates $[\zeta_1, \zeta_2]$
1	$[1.999, -0.999]$
2	$[-1.749, 1.499]$
3	$[-1.999, 0.999]$
4	$[1.749, -1.499]$

Table 4. Inner approximation simplices $\underline{\Gamma}^0$.

Simplex	Integer vector	Normal vector	Normal vector magnitude
s_1	[1, 2]	$[2.498, 3.748]'$	4.504*
s_2	[3, 2]	$[-0.500, 0.250]'$	0.559

* Largest magnitude indicates simplex to be broken.

Table 5. Outer approximation $\bar{\Gamma}^1$.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0.5546 & 0.8321 \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \end{bmatrix} \leq \begin{bmatrix} 2.000 \\ 1.507 \\ 0.6257 \end{bmatrix}$$

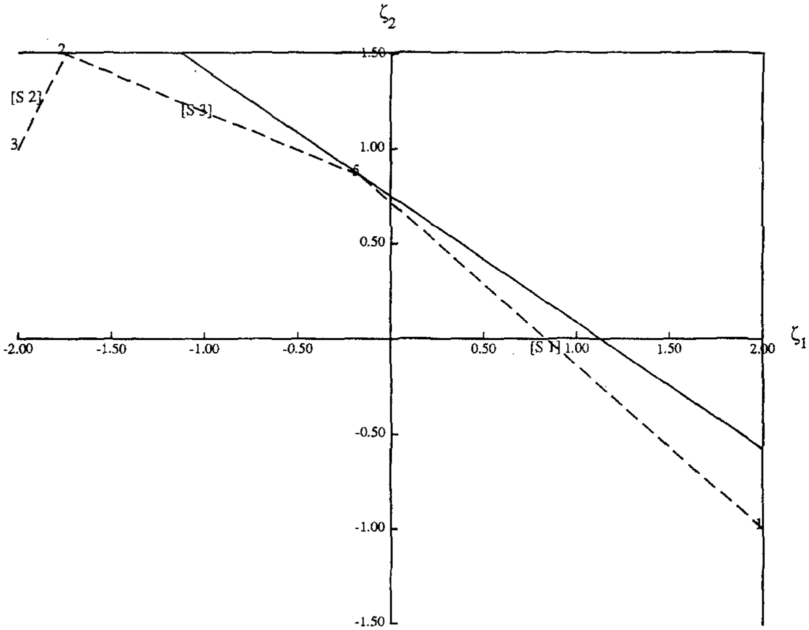

Fig. 8. Inner (dotted) and outer (solid) approximations after one iteration.

Table 6. Boundary point list B^1.

Boundary point	Coordinates $[\zeta_1, \zeta_2]$
1	$[1.999, -0.999]$
2	$[-1.749, 1.499]$
3	$[-1.999, 0.999]$
4	$[1.749, -1.499]$
5	$[-0.1645, 0.8593]$

Table 7. Inner approximation simplices $\underline{\Gamma}^1$.

Simplex	Integer vector	Normal vector	Normal vector magnitude
s_1	[1, 5]	[1.858, 2.163]	2.852*
s_2	[3, 2]	[−0.500, 0.250]	0.559
s_3	[5, 2]	[0.640, 1.584]	1.709

* Largest magnitude indicates simplex to be broken.

The outer approximation $\bar{\Gamma}^0$ is updated by appending the constraint $c_3'\zeta \le 0.6257$ to form $\bar{\Gamma}^1$. The complete representation is given in Table 5 and is plotted in Fig. 8. Solid lines enclose the outer approximation.

To form the updated inner approximation $\underline{\Gamma}^1$, the boundary point $[-0.1645, 0.8593]$ is appended to B^0 to form B^1, given in Table 6. Simplex s_1 is broken into two new simplices, described by the integer index vectors [1, 5] and [5, 2]. These are the $n=2$ combinations of the new boundary point (index number 5) and the $n=2$ boundary points (index numbers 1 and 2) that comprised simplex s_1 prior to the refinement step. Table 7 presents the new inner approximation $\underline{\Gamma}^1$, which is also illustrated in Fig. 8 as the region enclosed by the dashed lines.

Iterations continue by selecting a refinement direction corresponding to the largest simplex normal. In this case, normalization of [1.858, 2.163] yields $c_4 = [0.6516, 0.7586]'$ as a search direction. The optimal control solution to (SHP(c_4)) gives the boundary point $x^*_{c_4} = [0.8380, 0.0851]'$ and the ϵ-supporting hyperplane $c_4'\zeta = 0.613$.

The updated outer approximation $\bar{\Gamma}^2$ is represented analytically in Table 8. A new boundary point list B^2 and inner approximation $\underline{\Gamma}^2$ are given in Tables 9 and 10, respectively. At this stage in the refinement, the reachable set estimates are shown graphically in Fig. 9. Solid lines enclose the outer approximation, and dashed lines enclose the inner approximation. The exact reachable set boundary shown by the dotted line in Fig. 9 illustrates the accuracy of the approximations generated via the hyperplane method.

Table 8. Outer approximation $\bar{\Gamma}^2$.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0.5546 & 0.8321 \\ 0.6516 & 0.7586 \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \end{bmatrix} \le \begin{bmatrix} 2.000 \\ 1.507 \\ 0.626 \\ 0.613 \end{bmatrix}$$

Table 9. Boundary point list B^2.

Boundary point	Coordinates $[\zeta_1, \zeta_2]$
1	$[1.999, -0.999]$
2	$[-1.749, 1.499]$
3	$[-1.999, 0.999]$
4	$[1.749, -1.499]$
5	$[-0.165, 0.859]$
6	$[0.838, 0.085]$

Table 10. Inner approximation simplices $\underline{\Gamma}^2$.

Simplex	Integer vector	Normal vector	Normal vector magnitude
s_1	$[1, 6]$	$[1.084, 1.161]$	1.588
s_2	$[3, 2]$	$[-0.500, 0.250]$	0.599
s_3	$[5, 2]$	$[0.640, 1.584]$	1.709
s_4	$[6, 5]$	$[0.774, 1.003]$	1.267

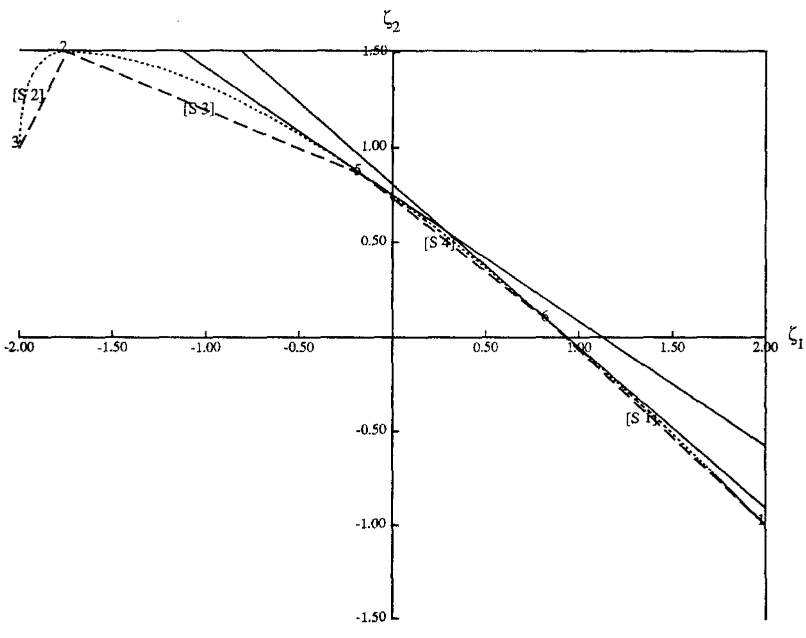

Fig. 9. Inner (dotted) and outer (solid) approximations after two iterations.

As can be seen from the sequence of inner and outer approximations in Figs. 7–9, rapid progress has been made in just two steps of the algorithm. Further improvements can be achieved by selecting new directions c_k and by solving (SHP(c_k)) to determine new boundary points and new ϵ-supporting hyperplanes. This example graphically illustrates the mechanics of the hyperplane method iterations and demonstrates its utility for simultaneously estimating a reachable set from within and from without. The reader is referred back to Section 3 for the complete details of each step in the procedure.

Example 4.2. Consider the third-order, single-input LTI system defined by the state equations

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -100 & -80 & -17 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t),$$

where $x(t) \in E^3$ is the state vector and the scalar input $u(t)$ is bounded such that

$$|u(t)| \leq 1.0.$$

The hyperplane method can be applied for this system to estimate Γ_∞ with an absolute ϵ error tolerance of 0.0001. Preliminary searches in the three coordinate directions yield

$$|\xi_1| \leq 1.010(10)^{-2}, \qquad |\zeta_2| \leq 2.075(10)^{-2}, \qquad |\zeta_3| \leq 1.065(10)^{-1}. \tag{19}$$

Since these bounds are relatively close (in magnitude) to the ϵ tolerance, improved numerical conditioning can be achieved by scaling the state equations appropriately. By defining a diagonal scaling matrix $D \in E^{n \times n}$ and a new state vector $x_s \in E^n$ such that

$$x(t) = Dx_s(t), \tag{20}$$

the state equations can be written as

$$\dot{x}_s(t) = D^{-1}ADx_s(t) + D^{-1}u(t).$$

Inner and outer approximations to the reachable set in the scaled state space can be computed via the hyperplane method. The results are then mapped linearly into inner and outer approximations in the original state space using the transformation (20).

For the current example, the individual coordinate bounds in (19) are used as the diagonal elements in the scaling matrix

$$D = \mathrm{diag}[1.010(10)^{-2}, 2.075(10)^{-2}, 1.065(10)^{-1}].$$

Table 11. Outer approximation for the scaled stated space.

$$\begin{bmatrix} 1.000 & 0.000 & 0.000 \\ 0.000 & 1.000 & 0.000 \\ 0.000 & 0.000 & 1.000 \\ 0.406 & 0.593 & 0.695 \\ -0.557 & -0.817 & 0.149 \\ 0.452 & 0.661 & 0.600 \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \end{bmatrix} \leq \begin{bmatrix} 0.990 \\ 0.984 \\ 1.000 \\ 0.519 \\ 0.749 \\ 0.472 \end{bmatrix}$$

The resulting scaled state equations are

$$\begin{bmatrix} \dot{x}_{s,1}(t) \\ \dot{x}_{s,2}(t) \\ \dot{x}_{s,3}(t) \end{bmatrix} = \begin{bmatrix} 0 & 2.0545 & 0 \\ 0 & 0 & 5.1322 \\ -9.4841 & -15.5877 & -17.000 \end{bmatrix} \begin{bmatrix} x_{s,1}(t) \\ x_{s,2}(t) \\ x_{s,3}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 9.3891 \end{bmatrix} u(t).$$

In the scaled state space, inner and outer approximations to the reachable set are computed using the hyperplane method. The results are presented in Tables 11–13. We note that the choice of the scaling matrix D derived from the preliminary coordinate searches has a normalizing effect. That is, the coordinate constraints in the scaled space nearly define the unit cube, as evidenced by the first three inequalities in Table 11.

Table 12. Boundary point list for the scaled state space.

Boundary point	Coordinates $[\zeta_1, \zeta_2, \zeta_3]$
1	[0.9900, 0.0000, −0.0018]
2	[−0.4509, 0.9835, 0.0001]
3	[−0.1114, −0.4178, 0.9975]
4	[−0.9900, −0.0000, 0.0018]
5	[0.4509, −0.9835, −0.0001]
6	[0.1114, 0.4178, −0.9975]
7	[−0.0028, 0.0057, 0.7395]
8	[−0.1297, −0.7173, 0.6046]
9	[0.1497, 0.2382, 0.4076]

Table 13. Inner approximation simplices for the scaled state space.

Simplex	Integer vector	Normal vector	Simplex RHS
s_1	[1, 2, 9]	[0.402, 0.588, 0.483]	0.3973
s_2	[4, 2, 3]	[−0.979, 0.538, 1.089]	0.9708
s_3	[1, 5, 3]	[0.982, −0.537, 0.858]	0.9708
s_4	[4, 5, 8]	[−0.594, −0.870, −0.187]	0.5879
s_5	[7, 2, 3]	[−0.061, 0.196, 0.296]	0.2202
s_6	[1, 7, 3]	[0.315, 0.176, 0.421]	0.3115
s_7	[8, 5, 3]	[0.076, −0.239, 0.179]	0.2698
s_8	[4, 8, 3]	[−0.462, −0.327, 0.271]	0.4582
s_9	[9, 2, 7]	[0.153, 0.261, 0.253]	0.1884
s_{10}	[1, 9, 7]	[0.174, 0.216, 0.232]	0.1720

The inner and outer approximations in the scaled state space are mapped via the scaling matrix D into the original state space. Hyperplane constraints of the form

$$c'x_s \leq c_0$$

become

$$c'D^{-1}x \leq c_0.$$

Boundary points x_s^* are mapped to boundary points x^* in the original state space as

$$x^* = Dx_s^*.$$

The results of these transformations for the present example are summarized in Tables 14–16 which present the outer approximation, boundary point list, and inner approximation, respectively.

Table 14. Outer approximation in the original state space.

$$\begin{bmatrix} 98.998 & 0.000 & 0.000 \\ 0.000 & 48.187 & 0.000 \\ 0.000 & 0.000 & 93.891 \\ 40.193 & 28.575 & 6.5254 \\ -55.142 & -39.369 & 1.399 \\ 44.747 & 31.852 & 5.6334 \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \end{bmatrix} \leq \begin{bmatrix} 0.990 \\ 0.984 \\ 1.000 \\ 0.519 \\ 0.749 \\ 0.742 \end{bmatrix}$$

Table 15. Boundary point list in the original state space.

Boundary Point	Coordinates $[\zeta_1, \zeta_2, \zeta_3] \times 10^3$
1	[10.000, 0.0000, −0.1917]
2	[−4.5546, 20.410, 0.0106]
3	[−1.1253, −8.6704, 106.24]
4	[−10.000, 0.0000, 0.1917]
5	[4.5546, −20.410, −0.0106]
6	[1.1253, 8.6704, −106.24]
7	[−0.0283, 0.01183, 78.762]
8	[−1.3101, −14.886, 64.394]
9	[1.5121, 4.9432, 43.412]

5. Conclusions

We conclude with some remarks regarding properties of the inner and outer reachable set estimates generated via the hyperplane method. First, the inner approximation is conservative with respect to system capability. If the objective of a reachable set analysis is to assess the extent of the system maneuvering capability at time T (e.g., for a minimum-time, optimal control problem), the inner approximation should be used. On the other hand, to guarantee avoidance of a specified set of states (e.g., to satisfy a set of operating constraints), the outer approximation ensures conservative results. The hyperplane method yields reachable set estimates for either end purpose, within arbitrarily specified accuracy.

Table 16. Inner approximation simplices in the original state space.

Simplex	Integer vector	Normal vector	Simplex RHS
s_1	[1, 2, 9]	[39.797, 28.334, 4.5349]	0.3973
s_2	[4, 2, 3]	[−96.919, 25.925, 10.225]	0.9708
s_3	[1, 5, 3]	[97.216, −26.876, 8.0558]	0.9708
s_4	[4, 5, 8]	[−58.805, −41.923, −1.7558]	0.5879
s_5	[7, 2, 3]	[−6.0389, 9.4446, 2.7792]	0.2202
s_6	[1, 7, 3]	[31.184, 8.4809, 3.9528]	0.3115
s_7	[8, 5, 3]	[7.5239, −11.517, 1.6806]	0.2698
s_8	[4, 8, 3]	[−45.737, −15.757, 2.5444]	0.4582
s_9	[9, 2, 7]	[15.147, 12.577, 2.3754]	0.1884
s_{10}	[1, 9, 7]	[17.226, 10.408, 2.1783]	0.1720

The finite precision of the numerical computations affects the accuracy of the reachable set estimates, particularly when the problem is poorly scaled. A problem is poorly scaled when the reachable set for a system is relatively elongated in one or more dimensions, and is relatively small in other dimensions. From our numerical experience, this difficulty can arise quite often. We have found it useful to do preliminary searches in the coordinate directions. These searches yield values for a normalization matrix that can be used to scale the state equations. By computing the reachable set in the normalized space, the numerical accuracy is enhanced. The inner and outer approximations in the normalized space can then be mapped easily into inner and outer approximations for the original space via the normalization matrix.

In our implementation of the algorithm, we used a heuristic direction selection in which the normal to the largest face of the inner approximation was chosen for improvement. Other heuristic approaches might prove to be more successful, and this remains a topic for future research. Finally, the stopping criteria was based on the size of the planes forming the inner approximation. Another approach would be to compute the volumes of the inner and outer approximations, and stop when the difference in volume is below a prescribed tolerance. Computation of the volume of convex sets is a topic of current research and has not been implemented in this work.

We are currently applying the hyperplane method to the problem of trajectory planning for servosystems. In this application, the objective is to determine a set of constraints in the parameter space used to generate reference trajectories for which it can be guaranteed that the system operating constraints are not violated and the tracking error does not exceed a specified tolerance. Outer approximations for the reachable set generated by the hyperplane method are being used in an embedded algorithm for generating bounds on the reference trajectory and its derivatives (Ref. 15). Another potential application of the hyperplane method is avoidance control (Refs. 16–17).

References

1. VINTER, R., *A Characterization of the Reachable Set for Nonlinear Control Systems*, SIAM Journal on Control and Optimization, Vol. 18, No. 6, pp. 599–610, 1980.
2. KHRUSTALEV, M. M., *Exact Description of Reachable Sets and Global Optimality Conditions for Dynamic Systems*, Automation and Remote Control, Vol. 49, No. 5, pp. 597–605, 1988 (in Russian).

3. GAYEK, J. E., and VINCENT, T. L., *On the Asymptotic Stability of Boundary Trajectories*, International Journal of Control, Vol. 41, No. 4, pp. 1077-1086, 1985.
4. GAYEK, J. E., and VINCENT, T. L., *On the Intersection of Controllable and Reachable Sets*, Journal of Optimization Theory and Applications, Vol. 50, No. 2, pp. 267-278, 1980.
5. GAYEK, J. E., and VINCENT, T. L., *Approximating Reachable Sets for a Class of Linear Control Systems*, International Journal of Control, Vol. 43, No. 2, pp. 441-453, 1986.
6. GRANTHAM, W. J., *Estimating Reachable Sets*, ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 103, No. 2, pp. 420-422, 1980.
7. SUMMERS, D., *Lyapunov Approximation of Reachable Sets for Uncertain Linear Systems*, International Journal of Control, Vol. 41, No. 5, pp. 1235-1243, 1985.
8. SUMMERS, D., *Lyapunov Estimation of Reachable Sets*, Abstracts of Papers to the American Mathematics Society, Vol. 10, p. 132, 1989.
9. HERMES, H., and LASALLE, J. P., *Functional Analysis and Time-Optimal Control*, Academic Press, New York, New York, 1969.
10. SNOW, D. R., *Reachable Regions and Optimal Controls*, Advances in Control Systems, Edited by C. T. Leondes, Academic Press, New York, New York, Vol. 5, pp. 133-196, 1967.
11. ATHANS, M. A., and FALB, P. L., *Optimal Control*, McGraw-Hill, New York, New York, pp. 553-554, 1966.
12. LEITMANN, G., *The Calculus of Variations and Optimal Control*, Plenum, New York, New York, 1981.
13. DIRECTOR, S., and HATCHEL, G., *The Simplicial Approximation Approach to Design Centering*, IEEE Transactions on Circuits and Systems, Vol. CAS-24, No. 7, pp. 363-372, 1977.
14. MOZUMDER, P. K., *Statistical Quality Control for VLSI Fabrication Processes*, MS Thesis, Carnegie Mellon University, 1986.
15. GRAETTINGER, T. J., and KROGH, B. H., *Reference Signal Derivative Constraints for Guaranteed Tracking Performance*, Proceedings of the 1989 American Control Conference, Pittsburgh, Pennsylvania, pp. 2702-2707, 1989.
16. AGGARWAL, R., and LEITMANN, G., *Avoidance Control*, ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 99, No. 2, pp. 152-154, 1977.
17. STICHT, D., VINCENT, T., and SCHULTZ, D., *Sufficiency Theorems for Target Capture*, Journal of Optimization Theory and Applications, Vol. 17, No. 5, pp. 523-543, 1975.