Minimax Input Shaper Design Using Linear Programming

Tarunraj Singh
Department of Mechanical and Aerospace Engineering,
SUNY at Buffalo,
Buffalo, NY 14260

The focus of this paper is on the design of robust input shapers where the maximum value of the cost function over the domain of uncertainty is minimized. This nonlinear programming problem is reformulated as a linear programming problem by approximating a n-dimensional hypersphere with multiple hyperplanes (as in a geodesic dome). A recursive technique to approximate a hypersphere to any level of accuracy is developed using barycentric coordinates. The proposed technique is illustrated on the spring-mass-dashpot and the benchmark floating oscillator problem undergoing a rest-to-rest maneuver. It is shown that the results of the linear programming problem are nearly identical to that of the nonlinear programming problem. [DOI: 10.1115/1.2963039]

1 Introduction

Attenuating vibrations induced by maneuvering flexible structures by judiciously designing the reference profile is a technique that has been exploited by numerous researchers [1–4]. Motivated by a simple wave cancellation concept for the elimination of the oscillatory motion of underdamped systems, Smith [1] proposed a technique refered to as *Posicast control*, which decomposes a step input into a staircase form. This technique required exact knowledge of the damping and natural frequency of the plant to be able to eliminate residual vibrations, which make it sensitive to errors in estimated damping and frequency of the system.

Singer and Seering [2] enhanced the performance of the Posicast controller by formulating a problem that requires determination of a series of impulses that force the response of the system to the impulse sequence and the derivative of the response of the system with respect to damping or frequency to be zero at the time corresponding to the application of the final impulse. They called the resulting controller an input shaper. Singh and Vadali [5] illustrated that a time-delay filter designed to locate a pair of zeros of the time-delay filter transfer function at the nominal location of the uncertain poles of the system resulted in a response that was identical to the input shaper. The input shaper/time-delay filter provides robustness of the filter in the vicinity of the nominal model of the system.

Given knowledge of the domain of uncertainty, different techniques are necessary to design controllers that exploit knowledge of the region of uncertainty. The multihump input shapers by Singhose et al. [6], and the minimax filters by Singh [7] are techniques that have been proposed to design robust filters. These controllers are designed by solving a nonlinear optimization problem, which is computationally expensive and requires numerous initial guesses for the gradient based optimizer to provide resonable confidence that the solution is globally optimal. There is therefore a desire to formulate the optimization problem as a convex problem to permit easy determination of a globally optimal solution [8].

Linear programming (LP) and semidefinite programming (SDP), which permit solving problems where the cost function is linear and the constraints are liner matrix inequalities (LMIs), are techniques that use efficient algorithms to permit solving large scale problems. This paper describes a technique to rewrite the minimax robust control design problem as a linear programming problem. Section 2 presents a standard technique [9] to convert a minimax problem to a linear programming problem. This is followed by the development of a technique to approximate a hypersphere with hyperplanes. The hypersphere is the bound for the residual energy of the dynamic system over the entire domain of uncertainty. Section 3 illustrates the proposed technique on two benchmark problems and compares the results of the LP formulation to the results of the nonlinear programming (NLP) problem.

2 Minimax Control

The design of input shaped controllers, which are insensitive to modeling uncertainties, has been posed in the framework of a minimax problem by Singh [7]. The residual energy of the system over the domain of uncertainty is calculated and the maximum magnitude is minimized. Since the residual energy is a quadratic function, it is not compatible with the constraints of a LP problem.

Consider the minimax problem,

$$\text{minimize} \max_{i=1,2,\ldots,M} c_i^T z + d_i \tag{1a}$$

$$\text{subject to } A_{\text{eq}} z = b_{\text{eq}} \tag{1b}$$

$$Az \le b \tag{1c}$$

where z is a vector to be solved for and M is the number of linear functions over which the maximum is evaluated. Defining a variable f, which is equal to the maximum of $c_i^T z + d_i$ for all i, the minimax problem can be stated as

$$\text{minimize} \quad f \tag{2a}$$

$$\text{subject to } A_{\text{eq}} z = b_{\text{eq}} \tag{2b}$$

$$Az \le b \tag{2c}$$

$$c_i^T z + d_i \le f, \quad \forall\, i = 1,2,\ldots,M \tag{2d}$$

This corresponds to a LP problem where the variables f and z are solved for. We propose to use the form of Eqs. (2*a*)–(2*d*) to solve the problem of design of minimax input shapers.

A linear model of a structure with n degrees of freedom can be represented as

$$M\ddot{x} + C\dot{x} + Kx = Du \tag{3}$$

where M, C, and K are the mass, damping, and stiffness matrices and D is the control influence matrix, which is a function of the location of the actuators. The residual energy of this system is the sum of the kinetic and the potential energy at the end of the maneuver (t_f) as follows:

Contributed by the Dynamic Systems, Measurement, and Control Division of ASME for publication in the JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT, AND CONTROL. Manuscript received May 4, 2007; final manuscript received May 13, 2008; published online August 4, 2008. Assoc. Editor: Ranjan Mukherjee.

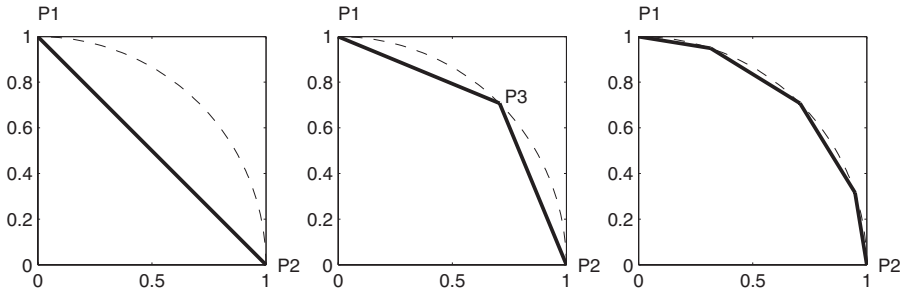

Fig. 1 Approximation of circle with lines

$$E(t_f) = \frac{1}{2}\dot{x}^T M\dot{x} + \frac{1}{2}x^T Kx = \frac{1}{2}(\sqrt{M}\dot{x})^T(\sqrt{M}\dot{x}) + \frac{1}{2}(\sqrt{K}x)^T\sqrt{K}x \quad (4)$$

which is a common metric to use in the design of robust controllers [7]. For systems with rigid body modes, a pseudopotential energy term has to be added to the kinetic and potential energy terms to guarantee that $E(t_f)$ is positive definite. The resulting cost function is

$$E(t_f) = \frac{1}{2}\dot{x}^T M\dot{x} + \frac{1}{2}x^T Kx + \frac{1}{2}k'(y_r - y_f)^2 \quad (5)$$

where y_r refers to the rigid body mode and y_f refers to the corresponding desired final displacement. k' is the stiffness of a pseudospring whose inclusion results in a positive definite cost function.

The residual energy corresponds to the l_2 norm and does not satisfy the constraints for a LP problem formulation. In the following development, the l_2 metrics will be approximated to formulate a LP problem for the design of desensitized input profiles.

2.1 l_2 Optimal Control. In this section, an approach to use linear programming to solve the minimax problem that minimizes the residual energy of the system, which is a l_2 norm, is presented. This involves approximating the n-dimensional hypersphere with $(n-1)$-dimensional hyperplanes (simplex). A simplex is the convex hull of $n+1$ points in n-dimensional space where the points do not all lie in some $(n-1)$-dimensional subspace. A n-simplex is a n-dimensional analog of a triangle. For the single dimension, the simplex is a line, for two dimensions, it is a triangle; for the third dimension, it is a tetrahederon; for the fourth dimension, it is a pentatope; and so on. This section will first provide the motivation for approximating hyperspheres with hyperplanes. It is patent that the greater the number of hyperplanes, the better the approximation of the hypersphere. Thus, it is desirous to develop a recursive scheme, which can approximate the hypersphere with any desired level of accuracy. To provide the tools necessary for the development of the recursive algorithm, the concept of barycentric coordinates is presented next.

Barycentric coordinates, also referred to as homogeneous coordinates or trilinear coordinates, are a means of defining a local coordinate system with a given $(n+1)$ set of points in n-dimensional space. For a set of points $\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \ldots, \mathcal{P}_{n+1}$, the combination

$$w_1\mathcal{P}_1 + w_2\mathcal{P}_2 + w_3\mathcal{P}_3 + \cdots + w_{n+1}\mathcal{P}_{n+1} \quad (6)$$

where

$$w_1 + w_2 + w_3 + \cdots + w_{n+1} = 1 \quad (7)$$

is referred to as an affine combination. The coordinates $[w_1\ w_2\ w_3 \cdots w_{n+1}]$ are called the barycentric coordinates of the point given by Eq. (6). If all the w_i are required to be positive, the barycentric coordinates represent all the points that lie in the convex hull of the points $\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \ldots, \mathcal{P}_{n+1}$.

Figure 1 illustrates the approximation of a unit circle with lines and Fig. 2 a unit sphere with multiple planes. The sequence of approximations of the circle and sphere with increasing number of lines and planes is designated as Levels 0, 1, 2, etc. It is clear that by Level 2, the approximation of the l_2 norm by planes is quite good for the 3D case.

Figure 3 illustrates the barycentric concept. The points $\mathcal{Q}$ and $\mathcal{R}$ on the line connecting the points $\mathcal{P}_1$ and $\mathcal{P}_2$ are given by the barycentric coordinates [0.5, 0.5] and [0.25, 0.75], respectively. Similarly, the barycentric coordinates for the points $\mathcal{S}$ and $\mathcal{T}$ on the triangle formed by the points $\mathcal{P}_1$, $\mathcal{P}_2$, and $\mathcal{P}_3$ are given by the 3-tuples [1/3 1/3 1/3] and [1/2 1/2 0], respectively.

Parametrizing specific points in a n-simplex using barycentric coordinates provides a simple mechanism to recursively divide the n-simplex into smaller segments. For example, if we start with two points specifying a line,

$$\mathcal{P}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{and } \mathcal{P}_2 = \begin{bmatrix} 3 \\ 5 \end{bmatrix} \quad (8)$$

and if we are interested in dividing the line into four segments, we first determine the point with a barycentric coordinate of [0.5 0.5], which corresponds to the point

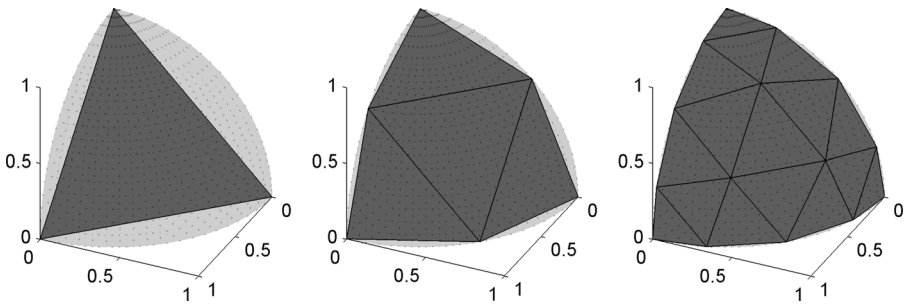

Fig. 2 Approximation of sphere with planes

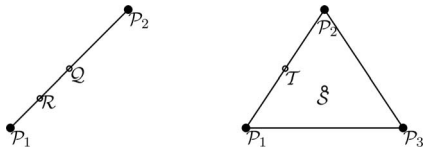

Fig. 3 Barycentric examples

$$\mathcal{P}_3 = 0.5\mathcal{P}_1 + 0.5\mathcal{P}_2 = 0.5\begin{bmatrix}1\\1\end{bmatrix} + 0.5\begin{bmatrix}3\\5\end{bmatrix} = \begin{bmatrix}2\\3\end{bmatrix} \tag{9}$$

The next set of points is determined by recursively calculating the points corresponding to the barycentric coordinate of [0.5 0.5] of the points $\mathcal{P}_1$ and $\mathcal{P}_3$ and $\mathcal{P}_2$ and $\mathcal{P}_3$, which results in the points

$$\mathcal{P}_4 = 0.5\mathcal{P}_1 + 0.5\mathcal{P}_3 = 0.5\begin{bmatrix}1\\1\end{bmatrix} + 0.5\begin{bmatrix}2\\3\end{bmatrix} = \begin{bmatrix}1.5\\2\end{bmatrix} \tag{10}$$

and

$$\mathcal{P}_5 = 0.5\mathcal{P}_3 + 0.5\mathcal{P}_2 = 0.5\begin{bmatrix}2\\3\end{bmatrix} + 0.5\begin{bmatrix}3\\5\end{bmatrix} = \begin{bmatrix}2.5\\4\end{bmatrix} \tag{11}$$

The approach that we will follow to generate hyperplanes to approximate the hypersphere involves determining the hyperplane that corresponds to the polytope that corresponds to the constant l_1 norm. For example, the constant $l_1 = 1$, polytopes for the two-dimensional space are given by the lines

$$x_1 + x_2 = 1 \tag{12}$$

$$-x_1 - x_2 = 1 \tag{13}$$

$$-x_1 + x_2 = 1 \tag{14}$$

$$x_1 - x_2 = 1 \tag{15}$$

which can be parametrized in terms of lines connecting the points

$$\mathcal{P}_1 = \begin{bmatrix}1\\0\end{bmatrix}, \quad \mathcal{P}_2 = \begin{bmatrix}0\\1\end{bmatrix} \quad \text{for the line} + x_1 + x_2 = 1$$

$$\mathcal{P}_4 = \begin{bmatrix}0\\-1\end{bmatrix}, \quad \mathcal{P}_3 = \begin{bmatrix}-1\\0\end{bmatrix} \quad \text{for the line} - x_1 - x_2 = 1$$

$$\mathcal{P}_3 = \begin{bmatrix}-1\\0\end{bmatrix}, \quad \mathcal{P}_2 = \begin{bmatrix}0\\1\end{bmatrix} \quad \text{for the line} - x_1 + x_2 = 1$$

$$\mathcal{P}_1 = \begin{bmatrix}1\\0\end{bmatrix}, \quad \mathcal{P}_4 = \begin{bmatrix}0\\-1\end{bmatrix} \quad \text{for the line} + x_1 - x_2 = 1$$

Consider the line connecting $\mathcal{P}_1$ and $\mathcal{P}_2$ (Eq. (12)), which approximates the arc of an unit radius circle in the first quadrant (Fig. 1). The barycentric coordinates [0.5 0.5] of the points $\mathcal{P}_1$ and $\mathcal{P}_2$ correspond to the point

$$\mathcal{P}_5 = 0.5\mathcal{P}_1 + 0.5\mathcal{P}_2 = \begin{bmatrix}0.5\\0.5\end{bmatrix} \tag{16}$$

Normalizing this point such that its Euclidean norm is unity results in the point

$$\mathcal{P}_5 = \begin{bmatrix}\frac{1}{\sqrt{2}}\\\frac{1}{\sqrt{2}}\end{bmatrix} \tag{17}$$

The line that connects $\mathcal{P}_1$ and $\mathcal{P}_5$, and $\mathcal{P}_5$ and $\mathcal{P}_2$ are given by the equations

$$0.9239x_1 + 0.3827x_2 = 0.9239 \tag{18}$$

$$0.3827x_1 + 0.9239x_2 = 0.9239 \tag{19}$$

Similarly the lines defined by Eqs. (13)–(15) can be divided to result in a total of eight lines to approximate the circle.

Similarly for the three-dimensional space, the planes can be parametrized in terms of the points

$$\mathcal{P}_1 = \begin{bmatrix}1\\0\\0\end{bmatrix}, \quad \mathcal{P}_2 = \begin{bmatrix}0\\1\\0\end{bmatrix}, \quad \mathcal{P}_3 = \begin{bmatrix}0\\0\\1\end{bmatrix} \quad \text{for } x_1 + x_2 + x_3 = 1$$

$$\mathcal{P}_1 = \begin{bmatrix}1\\0\\0\end{bmatrix}, \quad \mathcal{P}_2 = \begin{bmatrix}0\\1\\0\end{bmatrix}, \quad \mathcal{P}_6 = \begin{bmatrix}0\\0\\-1\end{bmatrix} \quad \text{for } x_1 + x_2 - x_3 = 1$$

$$\mathcal{P}_1 = \begin{bmatrix}1\\0\\0\end{bmatrix}, \quad \mathcal{P}_5 = \begin{bmatrix}0\\-1\\0\end{bmatrix}, \quad \mathcal{P}_3 = \begin{bmatrix}0\\0\\1\end{bmatrix} \quad \text{for } x_1 - x_2 + x_3 = 1$$

$$\mathcal{P}_1 = \begin{bmatrix}1\\0\\0\end{bmatrix}, \quad \mathcal{P}_5 = \begin{bmatrix}0\\-1\\0\end{bmatrix}, \quad \mathcal{P}_6 = \begin{bmatrix}0\\0\\-1\end{bmatrix} \quad \text{for } x_1 - x_2 - x_3 = 1$$

$$\mathcal{P}_4 = \begin{bmatrix}-1\\0\\0\end{bmatrix}, \quad \mathcal{P}_2 = \begin{bmatrix}0\\1\\0\end{bmatrix}, \quad \mathcal{P}_3 = \begin{bmatrix}0\\0\\1\end{bmatrix} \quad \text{for } -x_1 + x_2 + x_3 = 1$$

$$\mathcal{P}_4 = \begin{bmatrix}-1\\0\\0\end{bmatrix}, \quad \mathcal{P}_2 = \begin{bmatrix}0\\1\\0\end{bmatrix}, \quad \mathcal{P}_6 = \begin{bmatrix}0\\0\\-1\end{bmatrix} \quad \text{for } -x_1 + x_2 - x_3 = 1$$

$$\mathcal{P}_4 = \begin{bmatrix}-1\\0\\0\end{bmatrix}, \quad \mathcal{P}_5 = \begin{bmatrix}0\\-1\\0\end{bmatrix}, \quad \mathcal{P}_3 = \begin{bmatrix}0\\0\\1\end{bmatrix} \quad \text{for } -x_1 - x_2 + x_3 = 1$$

$$\mathcal{P}_4 = \begin{bmatrix}-1\\0\\0\end{bmatrix}, \quad \mathcal{P}_5 = \begin{bmatrix}0\\-1\\0\end{bmatrix}, \quad \mathcal{P}_6 = \begin{bmatrix}0\\0\\-1\end{bmatrix} \quad \text{for } -x_1 - x_2 - x_3 = 1$$

These points that define patches are used to divide each patch into smaller patches using the barycentric coordinate approach. These result in points that lie on the plane defined by the basic n-simplex. After the division of the n-simplex, all the points are normalized to lie on a unit radius hypersphere. The patches after normalization define new planes, which are inscribed within the unit radius hypersphere. These planes now define linear constraint equations, which permit us to approximate the spherical constraint with a set of linear constraints. To illustrate this process, consider the plane defined by the points $\mathcal{P}_1$, $\mathcal{P}_2$, and $\mathcal{P}_3$ in three-dimensional space. All possible combinations ${}^3C_2$ of the points $\mathcal{P}_1$, $\mathcal{P}_2$, and $\mathcal{P}_3$ are determined, which are $\{\mathcal{P}_1\mathcal{P}_2\}$, $\{\mathcal{P}_2\mathcal{P}_3\}$, and $\{\mathcal{P}_1\mathcal{P}_3\}$. The points that correspond to the barycentric coordinates [0.5 0.5] of the pairs of points are determined and are shown in Fig. 4(*a*). Normalizing the points $\mathcal{P}_4$, $\mathcal{P}_5$, and $\mathcal{P}_6$ results in points that lie on the sphere as shown in Fig. 4(*b*). The six points define four planes that are defined by the equations

$$0.5774x_1 + 0.5774x_2 + 0.5774x_3 = 0.8165 \tag{20}$$

$$0.8629x_1 + 0.3574x_2 + 0.3574x_3 = 0.8629 \tag{21}$$

$$0.3574x_1 + 0.8629x_2 + 0.3574x_3 = 0.8629 \tag{22}$$

$$0.3574x_1 + 0.3574x_2 + 0.8629x_3 = 0.8629 \tag{23}$$

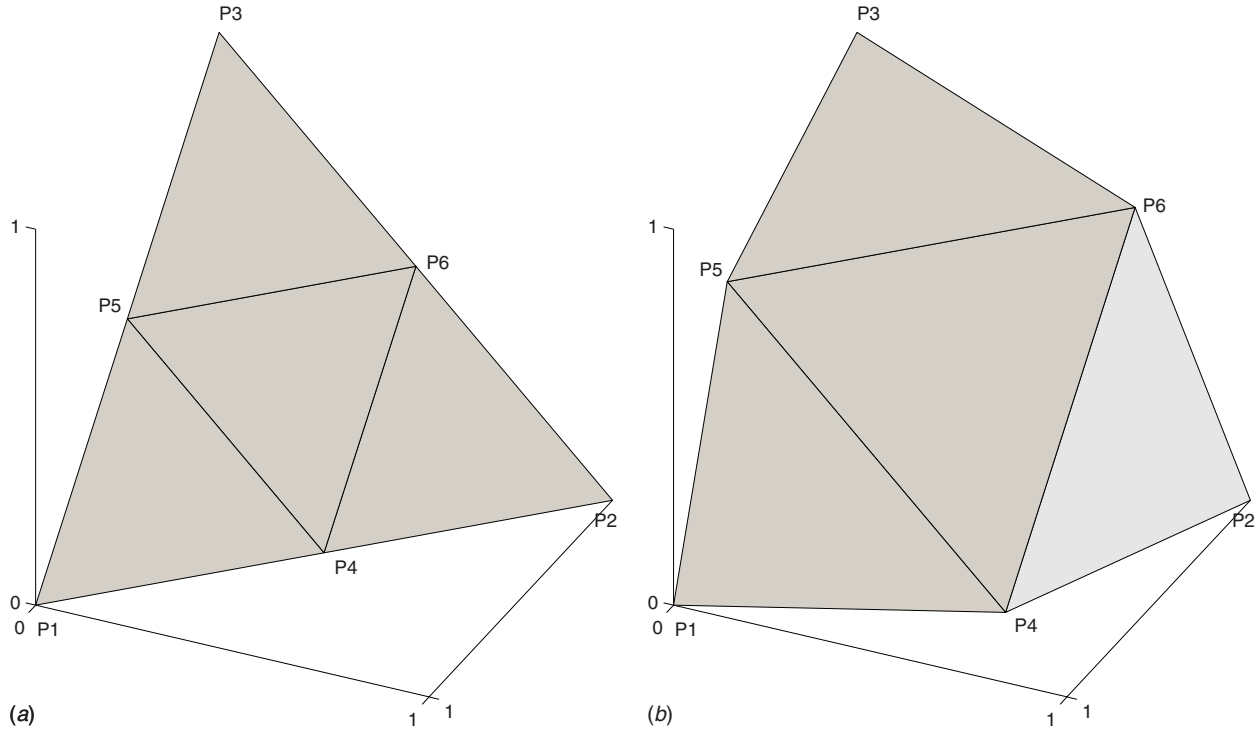

Fig. 4 Division of 2-simplex

Equations (20)–(23) represent the planes that approximate the hypersphere at Level 1. Level 0 refers to the minimum number of planes that approximate the hypersphere.

The number of patches that approximate the n-dimensional hyperplane is given by the equation

$$1 + \sum_{i=1}^{n-2} {}^{n}C_i \tag{24}$$

for each of the l_1 plane, and 2^n number of the l_1 planes create a polytope, which approximate the hypersphere. For example in 3-dimension ($n=3$), the number of patches to approximate the l_1 plane is 4 and there are a total of $2^3=8$ number of these l_1 planes as shown in Fig. 4(a). The first level ($L=1$) of recursion generates a total of 32 planes, which approximate the sphere and the second level ($L=2$) of recursion results in $2^n(4^L)=128$ planes.

To determine the largest error in using hyperplanes to approximate hyperspheres, we need to calculate the shortest distance of every hyperplane from the origin and the difference between this distance and unity is the error.

The equation of a hyperplane in n-dimension is given by the equation

$$\sum_{i=1}^{n} a_i x_i = b \tag{25}$$

and the normal to the plane is given by the vector

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{n-1} \\ a_n \end{bmatrix} \tag{26}$$

The equation of the normal to the plane passing through the origin is given by the equation

$$\mathbf{x} = \alpha \mathbf{a} \tag{27}$$

where α is a scalar. To determine the point of intersection of the normal (27) with the hyperplane (25), substitute Eq. (27) into Eq. (25) resulting in the equation

$$\sum_{i=1}^{n} a_i x_i = b \tag{28}$$

$$\alpha \sum_{i=1}^{n} a_i^2 = b \Rightarrow \alpha = \frac{b}{\sum_{i=1}^{n} a_i^2} \tag{29}$$

Having determined the coefficients of the hyperplanes to approximate the n-dimensional hypersphere, the maximum error in approximating the hypersphere with a set of planes is studied as a function of the number of hyperplanes.

The point of intersection of a line normal to plane passing through the origin and the plane is

$$\frac{b}{\sum_{i=1}^{n} a_i^2} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{n-1} \\ a_n \end{bmatrix} \tag{30}$$

The shortest distance of the plane from the origin in now given by the equation

$$\frac{b}{\sum_{i=1}^{n} a_i^2} \sqrt{\sum_{i=1}^{n} a_i^2} = \frac{b}{\sqrt{\sum_{i=1}^{n} a_i^2}} \tag{31}$$

When approximating a circle (sphere) with lines (planes), Figs. 5(a) and 5(b) illustrate the reduction in the error as a function of number of lines (planes) used to approximate the circle and sphere, respectively. As can be expected a greater number of hy-

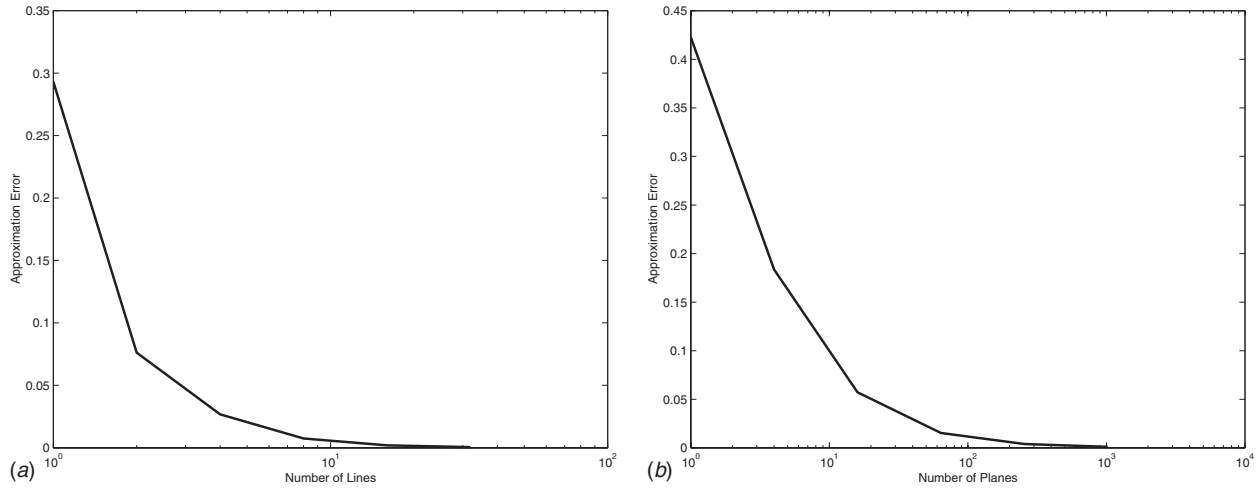

Fig. 5 Error in approximating hypersphere with hyperplanes

perplanes are necessary to achieve the same error in approximation as the dimension of the hypersphere increases.

3 Examples

3.1 Spring-Mass-Dashpot. To illustrate the proposed approach, consider the system

$$\ddot{x} + 0.2\dot{x} + kx = ku \tag{32}$$

where the spring stiffness k is uncertain and lies in the range

$$0.7 \leq k \leq 1.3 \tag{33}$$

The optimization problem is to design a control profile, which moves the system from rest to a final position at rest while minimizing the maximum magnitude of the residual states over the range of values of stiffness (k) defining the uncertain band. The boundary conditions are

$$\begin{bmatrix} x(0) \\ \dot{x}(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} x(t_f) \\ \dot{x}(t_f) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{34}$$

The cost function is

$$\min(f = \max_i k^i (x^i(t_f) - 1)^2 + (\dot{x}^i(t_f))^2) \tag{35}$$

which corresponds to the residual energy for a system whose boundary conditions correspond to a final displacement of unity and a final velocity of zero. The first approximate solution corresponds to the minimization of the l_2 norm of the residual states using eight lines to approximate the circle. The resulting linear programming problem is

$$\text{minimize } f \tag{36a}$$

$$\text{subject to} -0.9239f < +0.9239\dot{x}(t_f) + 0.3827\sqrt{k^i}x(t_f) < 0.9239f \tag{36b}$$

$$-0.9239f < +0.3827\dot{x}(t_f) + 0.9239\sqrt{k^i}x(t_f) < 0.9239f \tag{36c}$$

$$-0.9239f < -0.9239\dot{x}(t_f) + 0.3827\sqrt{k^i}x(t_f) < 0.9239f \tag{36d}$$

$$-0.9239f < -0.3827\dot{x}(t_f) + 0.9239\sqrt{k^i}x(t_f) < 0.9239f \tag{36e}$$

$$0 \leq u(m) \leq 1, \quad \forall\, m = 1,2,\ldots,501 \tag{36f}$$

$$u(m) - u(m+1) \leq 0, \quad \forall\, m = 1,2,\ldots,500 \tag{36g}$$

Figure 6(a) illustrates the variations of the square root of the residual energy as a function of the spring stiffness. The graph

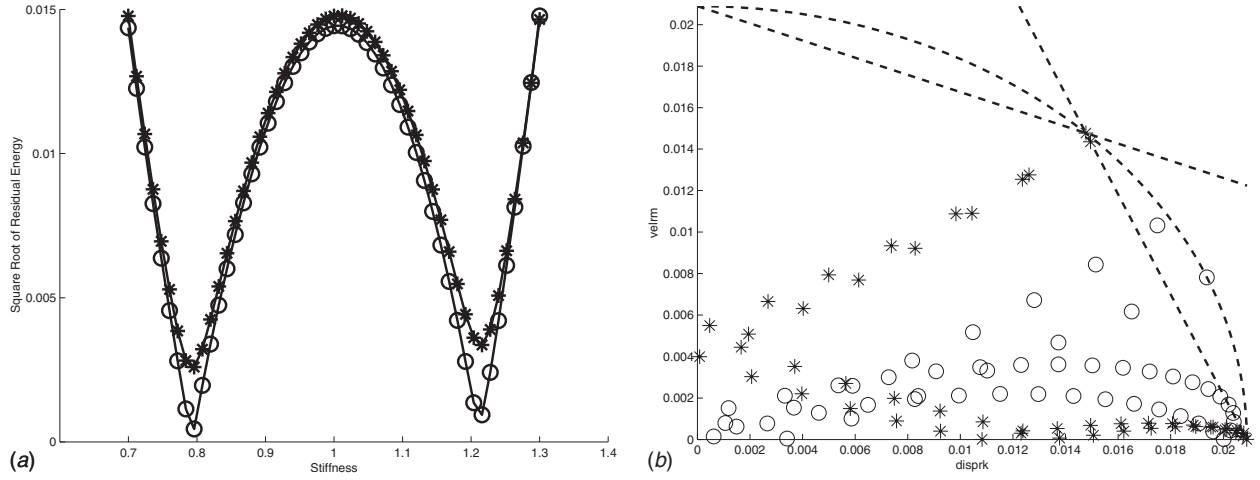

Fig. 6 Energy and norm plots

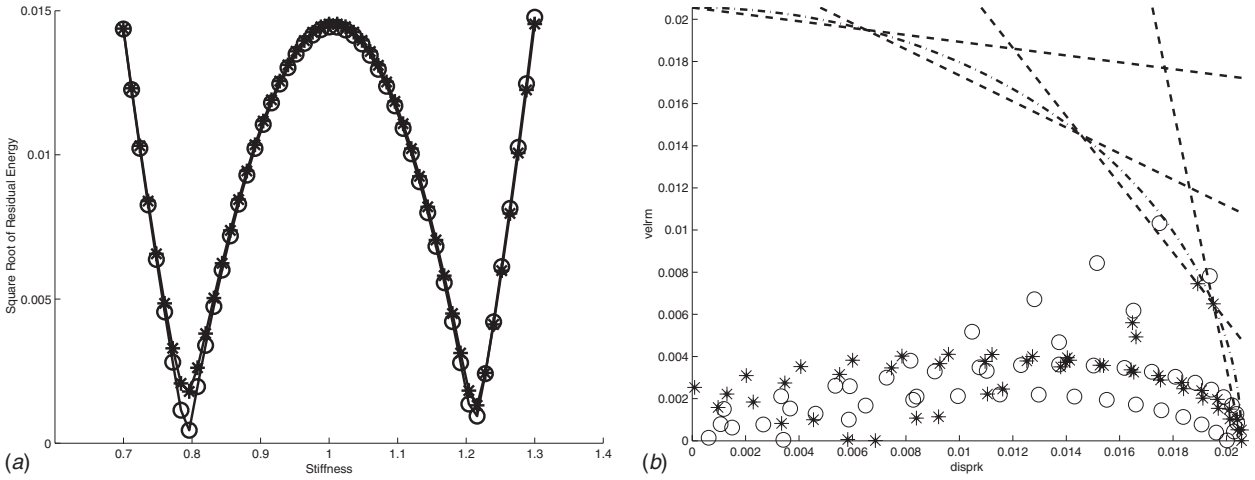

Fig. 7 Energy and norm plots

with the ∗ symbols corresponds to the solution determined by the linear programming problem where eight lines are used to approximate the circle, which contains the residual states, and the graph with the ○ symbols corresponds to the solution derived from a NLP problem. It can be seen that the difference between the solutions is minimal. Figure 6(*b*) illustrates the absolute magnitude of residual displacement plotted versus the absolute magnitude of the velocity. The ∗ and the ○ symbols correspond to the residual states for 51 models, which span the range of uncertain stiffness for the LP and NLP solutions. The dashed lines represent the lines that are used to approximate the circle. It can be seen that all the ∗ symbols lie within the polygon bounded by the dashed lines and the abscissa and the ordinate and the ○ symbols lie in the region bounded by the arc of a circle and the x and y axes.

Figure 7 illustrates the sensitivity curves and the norm of the residual states when two levels of recursion are used to generate 16 lines to approximate a circle. It can be seen that compared to Fig. 6 the solution is closer to that of the NLP solution, but the incremental improvement might not warrant solving a larger size LP problem.

Figure 8(*a*) and 8(*b*) illustrate the minimax control profile generated using 8 and 16 lines to approximate a circle, respectively. It can be noted that the solution with increasing number of lines to approximate the circle tends toward the two switch minimax solution generated by the NLP solver, which is given by the equation

$$u(t) = 0.3452 + 0.4730\mathcal{H}(t - 3.1703) + 0.1818\mathcal{H}(t - 6.3405) \tag{37}$$

which is the solution of the problem

$$\min J = \min_{T_i, A_i} \max_{0.7 \leq k \leq 1.3} \left(\frac{1}{2}\dot{x}^2 + \frac{1}{2}k(x-1)^2 \right) \tag{38a}$$

$$\text{subject to } \ddot{x} + 0.2\dot{x} + kx = ku \tag{38b}$$

$$x(0) = \dot{x}(0) = 0 \quad \text{and } x(t_f) = 1, \dot{x}(t_f) = 0 \tag{38c}$$

$$u(t) = A_0 + A_1\mathcal{H}(t - T_1) + A_2\mathcal{H}(t - T_2) \tag{38d}$$

$$A_0 + A_1 + A_2 = 1 \tag{38e}$$

and is illustrated by the dashed line in Fig. 8.

To study the effect of the maneuver time on the maximum residual energy over the domain of uncertain variables, a series of linear programming based minimax optimization problems is solved for specified maneuver times. Figure 9 illustrates the variation of the maximum residual energy versus maneuver time. It is clear that the residual energy of the worst plant decreases rapidly as the maneuver time increases. Increasing the maneuver time past 9 s results in a marginal improvement in the performance. Such curves can be used to trade off maneuver time versus robustness.

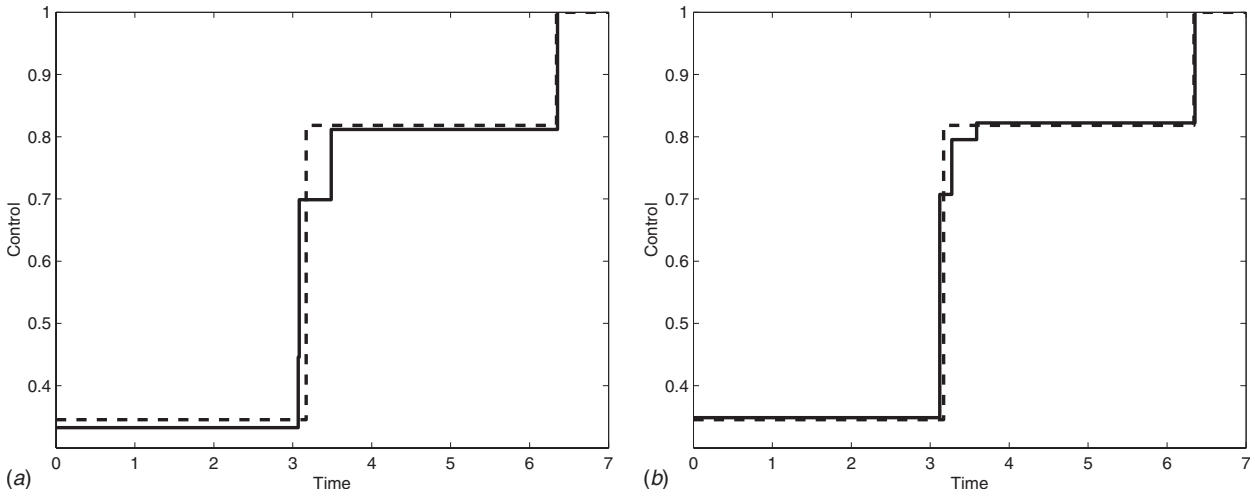

Fig. 8 Input shaped control profile

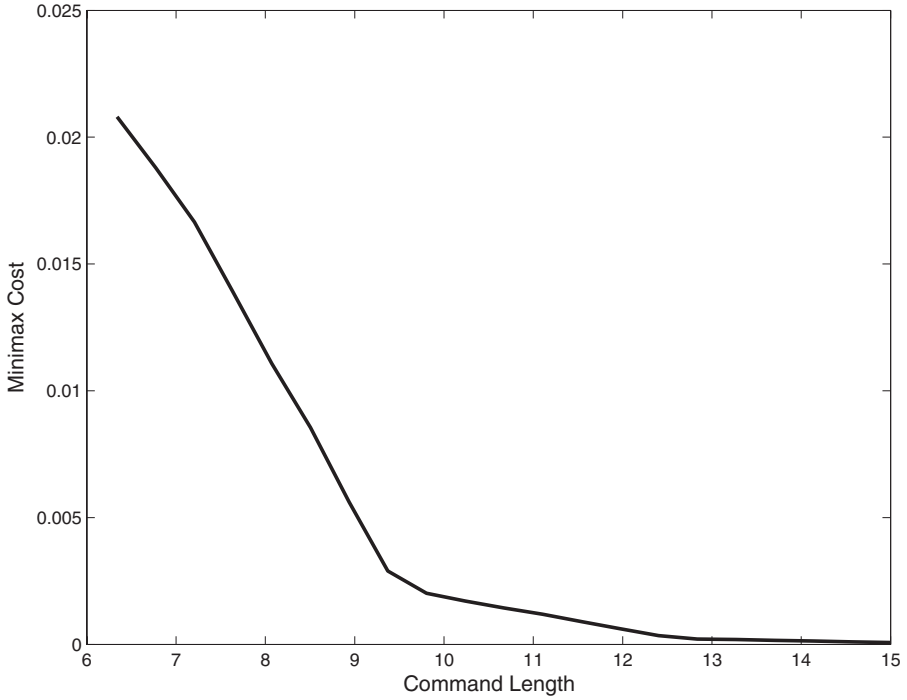

Fig. 9 Minimax cost sensitivity

3.2 Benchmark Floating Oscillator. To illustrate the l_2 robust optimization technique on a system with a rigid body mode, consider the the benchmark floating oscillator system (Fig. 10)

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{Bmatrix} + \begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} u \tag{39}$$

where the spring stiffness k is uncertain and lies in the range

$$0.7 \leqslant k \leqslant 1.3 \tag{40}$$

The optimization problem is to design a control profile, which moves the system from rest to a final position of rest while minimizing the maximum magnitude of the residual states over the range of uncertain k's. The boundary conditions are

$$\begin{bmatrix} x_1(0) \\ x_2(0) \\ \dot{x}_1(0) \\ \dot{x}_2(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} x_1(t_f) \\ x_2(t_f) \\ \dot{x}_1(t_f) \\ \dot{x}_2(t_f) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \tag{41}$$

Since the stiffness matrix is singular, the system states cannot be weighted by the square root of the stiffness matrices. To account for this a pseudospring of stiffness 1 is attached to the first mass so as to have zero potential energy about the final position. The resulting cost function is

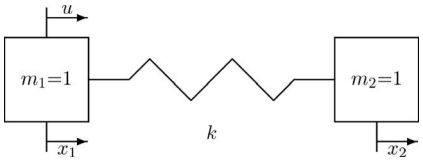

Fig. 10 Floating oscillator

$$\min \left\{ f = \max_i \begin{pmatrix} (x_1^i(t_f) - 1) \\ (x_2^i(t_f) - 1) \end{pmatrix}^T \begin{bmatrix} k+1 & -k \\ -k & k \end{bmatrix} \begin{pmatrix} (x_1^i(t_f) - 1) \\ (x_2^i(t_f) - 1) \end{pmatrix} \begin{pmatrix} \dot{x}_1^i(t_f) \\ \dot{x}_2^i(t_f) \end{pmatrix}^T + \begin{pmatrix} \dot{x}_1^i(t_f) \\ \dot{x}_2^i(t_f) \end{pmatrix} \right\} \tag{42}$$

The LP based approach to solving a minimax control problem for a system with a rigid body mode is illustrated here. The two-mass spring system with an uncertain stiffness coefficient is considered. The system is described by a fourth order model and hence a four-dimensional hypersphere needs to be approximated by hyperplanes. As was described earlier, planes that correspond to the l_1 norm of unity are determined, which are subsequently divided to generated a series of planes that are circumscribed by the hypersphere. For illustrative purposes consider the plane defined by the four points

$$\mathcal{P}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathcal{P}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathcal{P}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathcal{P}_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{43}$$

which are defined by the equation

$$x_1 + x_2 + x_3 + x_4 = 1 \tag{44}$$

The four points $\{\mathcal{P}_1\mathcal{P}_2\mathcal{P}_3\mathcal{P}_4\}$ can be used to represent the barycentric coordinates of a tetrahedron. The point that corresponds to the barycentric coordinate of $\left[\frac{1}{3}\ \frac{1}{3}\ \frac{1}{3}\right]$ are determined for all ${}^4C_3$ combinations of the points $\{\mathcal{P}_1\mathcal{P}_2\mathcal{P}_3\mathcal{P}_4\}$, which result in the points

$$\mathcal{P}_5 = \tfrac{1}{3}\mathcal{P}_1 + \tfrac{1}{3}\mathcal{P}_2 + \tfrac{1}{3}\mathcal{P}_3 = \left[\tfrac{1}{3} \quad \tfrac{1}{3} \quad \tfrac{1}{3} \quad 0\right]^T$$

$$\mathcal{P}_6 = \tfrac{1}{3}\mathcal{P}_1 + \tfrac{1}{3}\mathcal{P}_2 + \tfrac{1}{3}\mathcal{P}_4 = \left[\tfrac{1}{3} \quad \tfrac{1}{3} \quad 0 \quad \tfrac{1}{3}\right]^T$$

$$\mathcal{P}_7 = \tfrac{1}{3}\mathcal{P}_1 + \tfrac{1}{3}\mathcal{P}_3 + \tfrac{1}{3}\mathcal{P}_4 = \left[\tfrac{1}{3} \quad 0 \quad \tfrac{1}{3} \quad \tfrac{1}{3}\right]^T$$

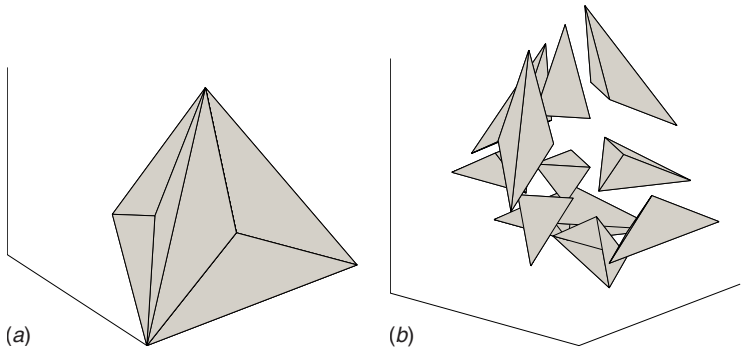

Fig. 11 Recursive division

$$\mathcal{P}_8 = \tfrac{1}{3}\mathcal{P}_2 + \tfrac{1}{3}\mathcal{P}_3 + \tfrac{1}{3}\mathcal{P}_4 = \begin{bmatrix} 0 & \tfrac{1}{3} & \tfrac{1}{3} & \tfrac{1}{3} \end{bmatrix}^T$$

The tetrahedron defined by the points $\{\mathcal{P}_1\mathcal{P}_2\mathcal{P}_3\mathcal{P}_4\}$ can be subdivided into 11 tetrahedra, which are given by the combinations

$$\{\mathcal{P}_5\mathcal{P}_6\mathcal{P}_7\mathcal{P}_8\}, \quad \{\mathcal{P}_1\mathcal{P}_5\mathcal{P}_6\mathcal{P}_7\}, \quad \{\mathcal{P}_2\mathcal{P}_5\mathcal{P}_6\mathcal{P}_8\}$$

$$\{\mathcal{P}_3\mathcal{P}_5\mathcal{P}_7\mathcal{P}_8\}, \quad \{\mathcal{P}_4\mathcal{P}_6\mathcal{P}_7\mathcal{P}_8\}, \quad \{\mathcal{P}_1\mathcal{P}_2\mathcal{P}_5\mathcal{P}_6\}$$

$$\{\mathcal{P}_1\mathcal{P}_3\mathcal{P}_5\mathcal{P}_7\}, \quad \{\mathcal{P}_1\mathcal{P}_4\mathcal{P}_6\mathcal{P}_7\}, \quad \{\mathcal{P}_2\mathcal{P}_3\mathcal{P}_5\mathcal{P}_8\}$$

$$\{\mathcal{P}_2\mathcal{P}_4\mathcal{P}_6\mathcal{P}_8\}, \quad \{\mathcal{P}_3\mathcal{P}_4\mathcal{P}_7\mathcal{P}_8\}$$

Figure 4(*a*) illustrated the division of a 2-simplex (triangle) into smaller segments. For the current four-dimensional problem the 3-simplex (tetrahedron) illustrated in Fig. 11(*a*) is subdivided into 11 smaller tetrahedra as shown in Fig. 11(*a*), which are also shown in an exploded view in Fig. 11(*b*). Each of the vertices of the tetrahedra is normalized so as to lie on the unit hypersphere. These 11 normalized tetrahedra represent 11 planes which approximate part of the hypersphere. The equations for these 11 planes are

$$0.5000x_1 + 0.5000x_2 + 0.5000x_3 + 0.5000x_4 = 0.8660$$

$$0.6830x_1 + 0.6830x_2 - 0.1830x_3 - 0.1830x_4 = 0.6830$$

$$0.6830x_1 - 0.1830x_2 + 0.6830x_3 - 0.1830x_4 = 0.6830$$

$$0.6830x_1 - 0.1830x_2 - 0.1830x_3 + 0.6830x_4 = 0.6830$$

$$-0.1830x_1 + 0.6830x_2 - 0.1830x_3 + 0.6830x_4 = 0.6830$$

$$-0.1830x_1 + 0.6830x_2 + 0.6830x_3 - 0.1830x_4 = 0.6830$$

$$-0.1830x_1 - 0.1830x_2 + 0.6830x_3 + 0.6830x_4 = 0.6830$$

$$0.8446x_1 + 0.3091x_2 + 0.3091x_3 + 0.3091x_4 = 0.8446$$

$$0.3091x_1 + 0.8446x_2 + 0.3091x_3 + 0.3091x_4 = 0.8446$$

$$0.3091x_1 + 0.3091x_2 + 0.8446x_3 + 0.3091x_4 = 0.8446$$

$$0.3091x_1 + 0.3091x_2 + 0.3091x_3 + 0.8446x_4 = 0.8446$$

where x_1–x_4 refer to the four states of the system. The 16 $l_1 = 1$ planes are divided into 11 planes resulting in a total of 176 planes to approximate the four-dimensional hypersphere. Each of these planes can be recursively divided into 11 planes each for better approximation of the hypersphere. In this example, we will stay with the 176 planes to approximate the hypersphere. Using 21 points to discretize the uncertain spring stiffness space results in a LP problem with a total of 3696 inequality constraints. Dividing the time interval into 301 time instants results in a total of 302 variables to be optimized for the magnitude of the control at 301 time instants, and the variable that approximates the radius of the hypersphere, which bounds the residual energy of the 21 uncertain models.

Figure 12(*a*) illustrates the variations of the residual energy of the system as a function of varying spring stiffness for solutions generated by the LP problem (∗) and the NLP (○). It can be seen that the sensitivity plots are nearly identical. Figure 12(*b*) plots the absolute magnitude of the displacement error versus the velocity

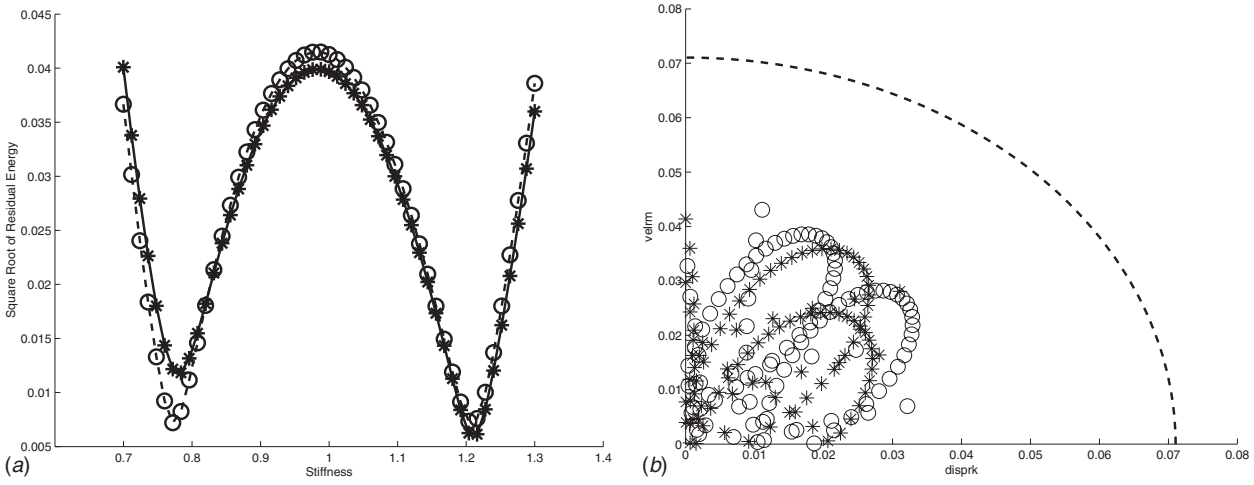

Fig. 12 l_2 Minimax control

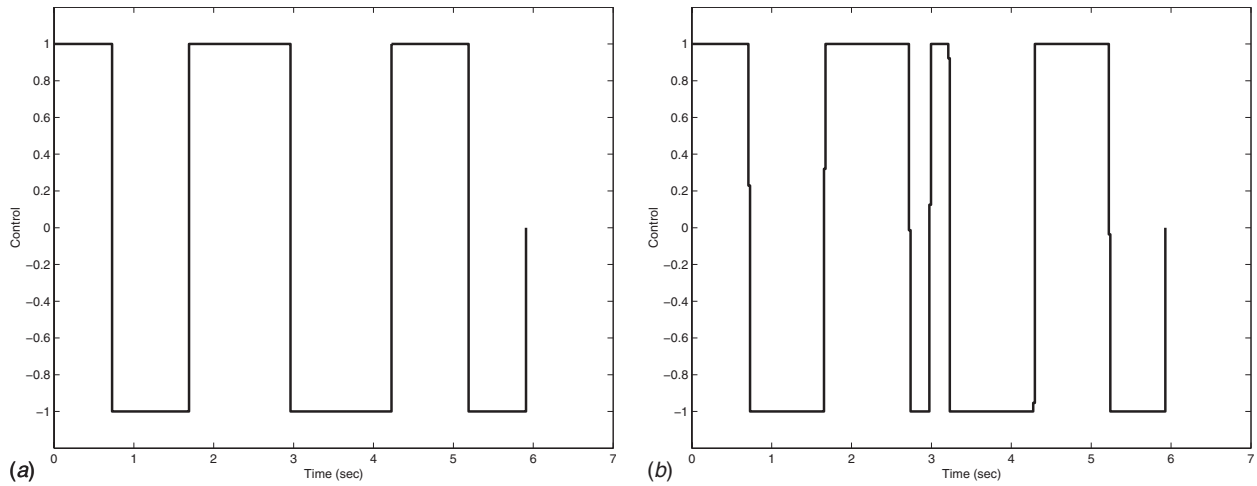

Fig. 13 l_2 Minimax control

error of each of the modes. The asterisks and circles correspond to the errors resulting from the control profiles generated by the LP and NLP problems, respectively. Figures 13(*a*) and 13(*b*) illustrate the bang-bang l_2 minimax control profiles generated by the NLP and the LP problems, respectively.

4 Conclusions

A simple technique to approximate n-dimensional hyperspheres with numerous n-dimensional hyperplanes has been proposed to convert a nonlinear programming problem to a linear programming problem. Powerful solvers for LP guarantee convergence to the global optimum. The barycentric coordinates have been proposed for the development of a recursive scheme to approximate hyperspheres to any desired level of accuracy. The proposed technique has been illustrated on two benchmark problems and the resulting controllers are shown to approximate the solution of the NLP problem very closely. The proposed technique inscribes the hyperplanes in the hypersphere, which results in solutions that are conservative compared to the NLP. The proposed technique can easily be extended to the case where the hypersphere is inscribed in the polytope generated by the hyperplanes.

Acknowledgment

The author would like to thank Professor William Menasco for helpful discussions about homogeneous coordinates.

References

[1] Smith, O. J. M., 1957, "Posicast Control of Damped Oscillatory Systems," Proc. IRE, **45**(9), pp. 1249–1255.
[2] Singer, N. C., and Seering, W. P., 1990, "Preshaping Command Inputs to Reduce System Vibrations," ASME J. Dyn. Syst., Meas., Control, **112**, pp. 76–82.
[3] Swigert, C. J., 1980, "Shaped Torque Techniques," J. Guid. Control, **3**, pp. 460–467.
[4] Singh, T., and Vadali, S. R., 1993, "Robust Time-Delay Control," ASME J. Dyn. Syst., Meas., Control, **112**(2A), pp. 303–306.
[5] Singh, T., and Vadali, S. R., 1995, "Robust Time-Delay Control of Multimode Systems," Int. J. Control, **62**(6), pp. 1319–1339.
[6] Singhose, W., Derezinski, S., and Singer, N., 1996, "Extra-Insensitive Input Shapers for Controlling Flexible Spacecraft," J. Guid. Control Dyn., **19**, pp. 385–91.
[7] Singh, T., 2002, "Minimax Design of Robust Controllers for Flexible Systems," J. Guid. Control Dyn., **25**(5), pp. 868–875.
[8] Singh, T., and Vadali, S. R., 1994, "Robust Time-Optimal Control: A Frequency Domain Approach," J. Guid. Control Dyn., **17**(2), pp. 346–353.
[9] Vanderbei, R. J., 2001, *Linear Programming: Foundations and Extensions*, 2nd ed., Springer, New York.