Table 1 Comparison of accuracy and computational effort for simulation with different time steps and orders of polynomials interpolation

Time step	Zero-order simulation		First-order simulation		Second-order simulation	
T, s	Error	Mflops	Error	Mflops	Error	Mflops
0.020	116.445	2.735	25.643	2.872	15.669	3.013
0.015	88.972	3.641	15.721	3.820	8.743	4.005
0.010	60.151	5.465	8.140	5.729	4.823	6.001
0.005	30.352	10.925	3.027	11.443	2.337	11.977

expected. Note that first-order simulations result in better accuracy than zero-order simulations at smaller time steps. Although the reduction in computational error from zero-order simulations to first-order simulations is significant, the improvement in accuracy from first-order simulations to second-order simulations is smaller. Moreover, a comparison of the flop counts shows that there is a small computational overhead in using higher order simulations. However, the computational effort for a higher order simulation is less than that for simulation with the same order of polynomial interpolation with a smaller time step.

This study shows that significant increase in solution accuracy can be achieved by using first- or second-order interpolations between the input samples as opposed to zero-order hold discretizations. In general, the benefits achieved by polynomial interpolation depend on the time step size, the dynamic model being simulated, and the frequency content of the input. Further, the advantages seem to taper off for higher order polynomial interpolations; therefore, it is advisable to use smaller time steps rather than higher order polynomial interpolations beyond second-order interpolation for better solution accuracy.

Conclusions

This Note presents a computationally efficient approach for simulation of linear time-invariant systems, with polynomial interpolation between input samples. Explicit expressions have been developed for such simulations, and efficient techniques for computation of the matrix integrals required in these expressions are presented. Numerical simulations were presented for comparing accuracy and computational effort involved with various time step sizes and for different orders of polynomial interpolation. It is concluded that polynomial interpolation between input samples leads to significant improvement in solution accuracy over the usual zero-order hold discretization for LTI simulations.

Acknowledgment

This research was supported by the Spacecraft Controls Branch, Guidance and Control Division, NASA Langley Research Center, Hampton, VA 23681, under contract NAS1-19341.

References

[1] *MatrixX Users Manual*, Integrated Systems, Santa Clara, CA, 1990.

[2] Grace, A., Laub, A. J., Little, J. N., and Thompson, C. M., *Control System Toolbox, for Use with Matlab*, The MathWorks, Natick, MA, 1992.

[3] Van Loan, C. F., "Computing Integrals Involving the Matrix Exponential," *IEEE Transactions on Automatic Control*, Vol. AC-23, No. 3, 1978, pp. 395–404.

[4] Golub, G. H., and Van Loan, C. F., *Matrix Computations*, Johns Hopkins Univ. Press, Baltimore, MD, 1989.

[5] Moler, C., and Van Loan, C. F., "Nineteen Dubious Ways to Compute the Exponential of a Matrix," *SIAM Review*, Vol. 20, No. 4, 1978, pp. 801–836.

[6] Armstrong, E. S., and Caglayan, A. K., "An Algorithm for the Weighting Matrices in the Sampled-Data Optimal Regulator Problem," NASA TN D-8372, 1976.

[7] Horta, L. G., Reaves, M., Elliott, K. B., Belvin, W. K., Sulla, J. L., Gregory, J., Ugoletti, R., and Sandridge, C. A., "Langley's CSI Evolutionary Model: Phase II," NASA TM 109059, 1993.

Attainable Moments for the Constrained Control Allocation Problem

Wayne C. Durham*
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061

Introduction

MODERN tactical aircraft are being designed with many more than the classical three sets of control effectors (ailerons, elevator, and rudder). The next generation of highly maneuverable airplanes are projected to have as many as 20 primary flight control effectors. These controls will all be constrained to certain limits, determined by the physical geometry of the control actuators or in some cases by aerodynamic considerations. The effective allocation, or blending, of these controls to achieve specific objectives is the control allocation problem.

The geometry of the constrained control allocation problem was developed in Ref. 1. In Ref. 2 we described a means of determining the subset of attainable moments, yielding a description of the boundary that contained the necessary information for the determination of controls in the allocation problem. The method presented in Ref. 2, although offering the advantage of generality, was admittedly complicated and difficult to implement. For the control allocation problems of particular interest, such generality is not required, and a simpler method of determining the attainable moment subset is available. That method is the subject of this Note.

The problem statement and nomenclature used in this note may be found in Refs. 1 and 2.

Method

Subset of Constrained Controls

First, consider the subset of constrained controls, Ω. Its m-dimensional bounding surface, $\partial(\Omega)$, is made up of rectangular two-dimensional surfaces, each of which corresponds to a given pair of controls varying within their constraints, whereas all other controls are at one or the other of their constraining values. Since there are $m - 2$ other such controls, and each of these other controls may assume one of two values, there are thus 2^{m-2} surfaces, or facets, corresponding to a given pair of controls. These 2^{m-2} surfaces are all parallel in the m-dimensional control space.

For example, if there are three controls ($m = 3$), say simple aileron, rudder, and horizontal tail, the constrained control subset is a simple box in three-dimensional space. There are two ($2^{m-2} = 2$) facets of the box for each combination of the three controls taken two at a time.

If we use the horizontal tail differentially, then with the left and right horizontal tails considered as separate controls, there are four controls ($m = 4$). In this case the subset of constrained controls is a four-dimensional hypercube that is bounded by eight three-dimensional cubes made up of 24 facets. With some imagination, one may sketch two three-dimensional boxes next to each other and connect all the corresponding vertices. This is the two-dimensional projection of the four-dimensional hypercube, in which one may visualize parallel facets occurring in groups of four.

This example could be continued by adding more controls, but our ability to visualize these higher dimensional spaces becomes strained. Suffice to say that, in considering the facets generated by considering the controls in pairs, there are 2^{m-2} such facets, all of which are parallel. Collectively, all of the facets in Ω comprise $\partial(\Omega)$.

Received July 8, 1993; revision received Dec. 14, 1993; accepted for publication March 7, 1994.

*Assistant Professor, Department of Aerospace and Ocean Engineering. Member AIAA.

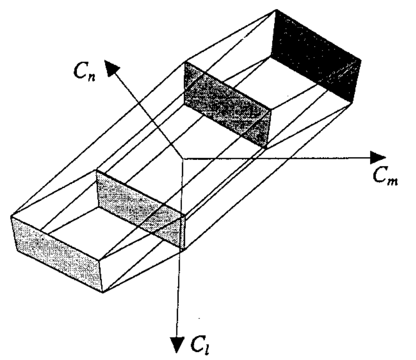

Fig. 1 Subset of attainable moments.

Subset of Attainable Moments

The set of all moments is given by $\boldsymbol{m} = B\boldsymbol{u}$. The subset of attainable moments, $\boldsymbol{\Phi}$, is B times all the constrained controls, or $\boldsymbol{\Phi} = \{\boldsymbol{m} \mid \boldsymbol{m} = B\boldsymbol{u}, \boldsymbol{u} \in \boldsymbol{\Omega}\}$. Because B is a linear operator, the boundary of the attainable moments subset, $\partial(\boldsymbol{\Phi})$, is a subset of the image of the boundary of the constrained controls subset, $\partial(\boldsymbol{\Omega})$, or $\partial(\boldsymbol{\Phi}) \subset B\boldsymbol{u}^*$. Since $\partial(\boldsymbol{\Omega})$ is made up of all of the facets of $\boldsymbol{\Omega}$, then $\partial(\boldsymbol{\Phi})$ is made up of the images of some subset of the facets of $\boldsymbol{\Omega}$. In other words, if we consider all of the images of the facets of $\boldsymbol{\Omega}$ and can then decide which of these are on the boundary of $\boldsymbol{\Phi}$, then we will have completely determined the boundary $\partial(\boldsymbol{\Phi})$.

To illustrate this idea, consider Fig. 1, the subset of attainable moments that results from four controls with deflection limits of ± 1 unit, and the B matrix:

$$B = \begin{bmatrix} 0.2 & -0.2 & 0.8 & 0.1 \\ -0.5 & -0.5 & -1.0 & 0.0 \\ -0.3 & 0.3 & -0.4 & -0.5 \end{bmatrix} \tag{1}$$

Although it is not important to this example, it may help the discussion if we identify the rows of B with rolling, pitching, and yawing moment coefficients and the columns with left-horizontal tail, right-horizontal tail, ailerons, and rudder. (Neither this B matrix nor the control limits are representative of any real airplane; they were chosen arbitrarily for purposes of illustration.)

To the point of this discussion, all of the mutually parallel facets of the four-dimensional subset of constrained controls are mutually parallel faces of Fig. 1, since we have performed a linear transformation. Considering any four such facets, it is clear that, unless they project to the same plane in Fig. 1, only two of the four can be on the boundary of the figure shown. None of the faces in question will be coplanar if we require that every 3×3 submatrix of B is of rank 3. Four mutually parallel faces have been shaded in Fig. 1 to illustrate this. In Fig. 1, we can easily visualize that the lower left and upper right shaded faces will be on the boundary and that the other two will lie in the interior of the shape. In general, for a given pair of controls, there will be 2^{m-2} such parallel faces, and only two will be on the boundary.

We seek a simple rule for deciding which two are on the boundary. Consider a line through the origin that is perpendicular to the 2^{m-2} faces under consideration. The two faces that are on the boundary are therefore those for which the magnitude of the moment vector in the direction of this line are a minimum and a maximum. The problem therefore reduces to a linear minimization/maximization problem with hard constraints on the variables. The method of solution is as follows: We will first perform a pure rotation transformation of the coordinates in moment space to align one axis with the perpendicular to the faces in question and then determine from the transformed B matrix what controls yield the maximum and the minimum along this axis.

Solution

We seek first to align one of the three axes in moment space with the perpendicular to the faces under consideration. For sake of discussion let us choose to examine the faces corresponding to the left and right horizontal tails and to rotate the C_l axis to the direction perpendicular to these faces. Consider some 3×3 transformation matrix T that rotates the coordinates in moment space as desired. Since we are concerned only with the C_l axis, we only need the first row of T. Call this first row (as a row vector) $\boldsymbol{t} = [t_1\ t_2\ t_3]$, and partition B as column vectors, $\boldsymbol{v}_i$: $B = [\boldsymbol{v}_1 : \boldsymbol{v}_2 : \boldsymbol{v}_3 : \boldsymbol{v}_4]$, where $\boldsymbol{v}_i = \{v_{i1}, v_{i2}, v_{i3}\}^T$.

The first row of TB is then $[\boldsymbol{tv}_1 : \boldsymbol{tv}_2 : \boldsymbol{tv}_3 : \boldsymbol{tv}_4]$. In order for this transformation to align the C_l axis with the perpendicular to the faces formed by the first two controls, we must have $\boldsymbol{tv}_1 = 0$, $\boldsymbol{tv}_2 = 0$. This gives us two equations for the three unknowns in $\boldsymbol{t}$, so we arbitrarily assign a value to one of the three unknowns, say t_3, and calculate the other two according to

$$\begin{Bmatrix} t_1 \\ t_2 \end{Bmatrix} = -\begin{bmatrix} v_{11} v_{12} \\ v_{21} v_{22} \end{bmatrix}^{-1} t_3 \begin{Bmatrix} v_{13} \\ v_{23} \end{Bmatrix} \tag{2}$$

In our example, this results in

$$\begin{Bmatrix} t_1 \\ t_2 \end{Bmatrix} = -\begin{bmatrix} 0.2 & -0.5 \\ -0.2 & -0.5 \end{bmatrix}^{-1} t_3 \begin{Bmatrix} -0.3 \\ 0.3 \end{Bmatrix} = \begin{Bmatrix} 1.5 \\ 0.0 \end{Bmatrix} t_3$$

If we wished, we could pick t_3 to make $|\boldsymbol{t}| = 1$, but that is not needed. We note here that, if the matrix inverse does not exist, this just means that $t_3 = 0$, and we proceed accordingly. We select $t_3 = 1$, $\boldsymbol{t} = [1.5\ 0.0\ 1.0]$, and $\boldsymbol{t}B = [0.0\ 0.0\ 0.8\ -0.35]$.

To determine what values of aileron and rudder yield the minimum and maximum values of moment along this new direction, we simply note the signs of the nonzero terms in $\boldsymbol{t}B$. The third entry (corresponding to ailerons) is positive, and the fourth (corresponding to rudder) is negative. Therefore, the maximum occurs when the ailerons are at their greatest positive value $(+1)$ and the rudder at its greatest negative value (-1). The minimum, of course, is just the opposite. Based on this, we begin our construction of the boundary of the attainable moment subset by selecting the two facets we have identified.

The coordinates of the four vertices of each facet are obtained by using the values of the third and fourth controls obtained above and the four combinations of minimum and maximum constraints for the first two controls. That is, for the facet that lies at the maximum, the ailerons are $+1$ and the rudder is -1, and the four combinations are $\{+1, +1, +1, -1\}$, $\{+1, -1, +1, -1\}$, $\{-1, +1, +1, -1\}$, and $\{-1, -1, +1, -1\}$. The coordinates in moment space are then B times each of these, or $\{0.7, -2.0, 0.1\}$, $\{1.1, -1.0, -0.5\}$, $\{0.3, -1.0, 0.7\}$, and $\{0.7, 0.0, -0.1\}$. This operation is repeated for the opposing facet, eliminating the two interior facets in Fig. 1.

We continue this process, taking the controls two at a time, and finally arrive at the completed attainable moment subset. Figure 1 in Ref. 2 is typical of the results obtained using this method.

In describing the method to be used to determine the facets, we assumed that none of the parallel facets in the constrained control subset projected to the same plane in the attainable moment subset. Mathematically, this assumption is invalid if there are any 3×3 submatrices of B that are less than full rank. Physically, this condition will exist if the action of any single control may be duplicated, without consideration of the constraints, by the combination of one or two others.

If any of the 3×3 submatrices of B are less than full rank, one should add physically insignificant deltas to the appropriate entries to remove the singularities. Although the resulting submatrices will be poorly conditioned, the operations described above may be completed. The effect of modifying the entries is to provide a tie breaker in the event of nonunique solutions on the boundary of the attainable moment subset.

Conclusion

The algorithm presented in this Note is computationally simple, requiring only the inversion of a 2×2 matrix and subsequent multiplication of an m-dimensional control vector (where m is the number of controls) by the control effectiveness matrix to calculate the coordinates of each vertex in moment space. With the condition that there be no completely redundant controls, the algorithm directly identifies each of the $m(m-1)$ facets of the attainable moment subset without searching or iteration. Since there are $2^{(m-2)}m!/[(2!(m-2)!]$ faces in the subset that are potentially on the boundary, the computational savings of this algorithm over a brute-force search can be impressive. Further computational savings can be achieved by noting that the vertices of a particular facet are always shared by other facets, and their coordinates need not be recalculated for each new facet.

For example, with 20 controls, there are almost 50 million facets in the subset of constrained controls, and the image of each of these is potentially a facet in the subset of attainable moments. Since facets are generated in pairs, the algorithm will identify the 380 facets out of the 50 million candidates in 190 passes. Whereas each facet has four vertices, there are not $4 \times 380 = 1520$ separate vertices to be calculated, but only $m(m-1)+2 = 382$ vertices.

The ability to calculate attainable moment subsets rapidly and reliably has two important benefits. First, one need not precalculate the subsets for use in allocating the controls in flight. Instead, aerodynamic tables of control effectiveness may be interpolated on-the-fly and new moment subsets generated for any flight condition encountered. Second, the attainable moment subset may be recalculated immediately following the identification of a control effector failure. The failed control effector is simply removed from the problem, both in the B matrix and in the control vector. If the control law used in the flight control system deals with generalized controls (three orthogonal moment generators) and leaves the actual allocation of the surfaces to the allocator, then the control law proper will be oblivious to the failure of a control effector, so long as the remaining effectors are capable of generating the moments implied by the generalized controls.

The determination of the attainable moment subset only solves half of the control allocation problem. The other half, the actual allocation of controls based on the moment subset description, is not a difficult task. Following this discussion, we conclude that real-time implementation of the algorithm presented in this Note, as part of a multicontrol effector control system, is both practical and desirable.

Acknowledgment

A major portion of this work was conducted under NASA Research Cooperative Agreement NCC1-158, supervised by John V. Foster of the NASA Langley Research Center.

References

[1]Durham, W. C., "Constrained Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, 1993, pp. 717–725.

[2]Durham, W. C., "Constrained Control Allocation: Three Moment Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 2, 1994, pp. 330–336.

Analysis of Nonlinear Equations by Robust Stability Theory

Howard Musoff*
Charles Stark Draper Laboratory, Inc.,
Cambridge, Massachusetts 02139

Introduction

A PROPOSITION and its proof relating the determination of asymptotic stability of nonlinear differential equations to robust stability theory for structured uncertainties are introduced. An example illustrating the resulting technique is presented.

Received April 3, 1993; revision received Aug. 26, 1993; accepted for publication Jan. 6, 1994.

*Principal Member, Technical Staff. Member AIAA.

Proposition

The nonlinear state equation is written

$$\dot{x} = A(x)x \tag{1}$$

where the elements of $A(x)$ range according to all possible values of the components of x.

If required, the elements of $A(x)$ can always be bounded with finite values by a suitable change in the time scale given by

$$\frac{d\tau}{dt} = g(x) \tag{2}$$

where t is the original time scale, τ is the transformed scale, and

$$g(x) \geq 1 \tag{3}$$

The resulting system of equations

$$\frac{dx}{d\tau} = \frac{A(x)}{g(x)}x \tag{4}$$

has bounded coefficients of x and is asymptotically stable at the equilibrium point of $\mathbf{0}$ if Eq. (1) is and vice versa.

The elements of the bounded matrix can be considered to belong to interval sets. Thus for a bounded $B(x)$

$$B_{ij} \in [\min B_{ij}, \max B_{ij}] \tag{5}$$

where the brackets denote an interval set.

The proposition for determining asymptotic stability of the original nonlinear equations is that robust stability theory (i.e., using the classical Routh-Hurwitz array with interval arithmetic[1]) for structured uncertainties can also be applied to those equations (derived from the nonlinear equations) that contain interval sets of the type (5).

Proof of Proposition

If the original differential equation (1) is asymptotically stable and has a suitable autonomous Lyapunov positive-definite function $v(x)$ in the domain $0 < x < x_m$, then

$$\frac{\partial v}{\partial x}\frac{dx}{dt} = \frac{\partial v}{\partial x}A(x)x \tag{6}$$

$$\frac{\partial v}{\partial x}A(x)x = w(x) \tag{7}$$

$$w(x) < 0 \tag{8}$$

and $w(x)$ is negative definite according to Lyapunov's direct method.[2]

After the suitable time scale change given by Eq. (4) we have

$$\frac{\partial v}{\partial x}\frac{dx}{d\tau} = \frac{\partial v}{\partial x}\frac{A(x)}{g(x)}x \tag{9}$$

$$\frac{\partial v}{\partial x}\frac{A(x)}{g(x)}x = \frac{w(x)}{g(x)} < 0 \tag{10}$$

because of Eq. (3).

Thus the equations with the new time scale τ are asymptotically stable if the original equations are and vice versa.

Therefore, we assume in the domain

$$0 < x < x_m \tag{11}$$

that suitable time scale changes have been made to yield

$$\frac{dx}{\partial \tau} = B(x)x \tag{12}$$

where the elements of $B(x)$ are bounded by finite values. That is,

$$B(x) \in [B_l, B_u] \tag{13}$$