JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 16, No. 4, July–August 1993

Constrained Control Allocation

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

This paper addresses the problem of the allocation of several airplane flight controls to the generation of specified body-axis moments. The number of controls is greater than the number of moments being controlled, and the ranges of the controls are constrained to certain limits. They are assumed to be individually linear in their effect throughout their ranges of motion and independent of one another in their effects. The geometries of the subset of the constrained controls and of its image in moment space are examined. A direct method of allocating these several controls is presented that guarantees the maximum possible moment can be generated within the constraints of the controls. It is shown that no single generalized inverse can yield these maximum moments everywhere without violating some control constraint. A method is presented for the determination of a generalized inverse that satisfies given specifications which are arbitrary but restricted in number. We then pose and solve a minimization problem that yields the generalized inverse that best approximates the exact solutions. The results are illustrated at each step by an example problem involving three controls and two moments.

Introduction

CLASSICALLY, airplane flight controls are designed with the idea of a single controller for each rotational degree of freedom. That is, elevators usually control pitching moments, rudders control yawing moments, and ailerons control rolling moments. With three independent controls and three moments to be generated, solutions to the control allocation problem are unique. Usually, the longitudinal control is uncoupled from the lateral-directional control, which means that pitching moments may be generated independently of roll-yaw effects. The coupling of aileron and rudder in roll and yaw gives rise to a common control allocation problem, namely, the generation of a rolling moment without yaw, which is customarily solved by use of a mechanical aileron-rudder interconnect (ARI), or the rolling surface-rudder interconnect (RSRI) in the F-18.[1]

Many modern aircraft have more than three independent moment generators. These additional moment generators may arise from the freeing of opposing control surfaces to operate independently of one another. For example, the left and right horizontal tails may operate independently, each generating rolling, pitching, and yawing moments. They also arise from the addition of nontraditional forms of controllers, such as thrust vectoring, canard control, elevons, etc. We consider each such controller individually; for example, the left and right trailing edge flap are separate controls. In advanced, high performance tactical aircraft [like the high angle-of-attack research vehicle (HARV)], one has potentially 13 or more independent moment controllers: horizontal tail, aileron, leading-edge flap, trailing-edge flap, and rudder, each left and right; and three thrust-vectoring moment generators. Add spoilers to this on the leading-edge extensions, vertical lift and side-force generators, and the number of controls nears 20.

These controls are all constrained to certain limits, determined by the physical geometry of the control actuators, or in some cases by aerodynamic considerations. If the control constraints are not considered, the mathematical solutions to the allocation of these controls are, in general, infinite. Quite often, the minimum-norm solution (sometimes employing weighting based on the limits of the controllers) is used.[2,3] The minimum-norm solution yields minimum control energy, which is useful in itself, but it does not yield maximum performance (maximum attainable moment, where the moment is expressed as a vector). It is the purpose of this paper to develop the tools and insights necessary to allocate these many controllers in a manner that always guarantees the maximum control-generated moments within the control constraints.

We will view the controls strictly as moment generators, which loosely translates to angular-acceleration generators. Thus, we are thinking of maneuvering tasks, as opposed to flight-path control. The usual methods of solving automatic control problems, such as output feedback, do not explicitly specify instantaneous forces and moments for the controls to generate. On the other hand, the increasingly popular method of dynamic inversion[2,3] (as well as model-following control, of which dynamic inversion is a special case) does specify instantaneous forces and moments for the controls to generate, after "subtracting out" the other aerodynamic and inertial forces. Therefore, the methods in this paper are offered as a replacement for that last step in dynamic inversion, wherein one "solves" for the control vector.

The results in this paper may also be extended to classical problem solutions by replacing the many controls of the original problem with just three: one each for rolling, pitching, and yawing moment. The resultant control law may then be interpreted as commands for each of those three moments, which provide the input for the control allocation scheme. For stabilization problems this touch is usually unnecessary, since small control deflections are normally required for small disturbances, and saturation is not an issue. For large disturbances, however, maximum control effort may be required. Moreover, for stabilization during maneuvering flight, use of the proper control allocation scheme will result in greater margins of control power available for stabilization.

It is important to keep in mind that the control allocation schemes discussed in this paper do not constitute a control law in themselves. Questions of robustness, handling qualities, and so on, belong in the control law. Given that the control law has determined what moments it wants the controls to deliver, we seek here to determine how best to apportion the existing controls to satisfy that requirement.

This paper is organized as follows: A practical sample problem is introduced. It will be referred to throughout to illustrate the points being made. The section following the sample describes the mathematics of the problem in geometrical terms and establishes the basis for subsequent discussion. The subsequent section describes in general terms what is required to solve the problem, and in specific terms how to solve the problem for low-order (two-moment) problems. The next

Received Jan. 25, 1992; revision received June 15, 1992; accepted for publication July 1, 1992.

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

section discusses the role of generalized inverses in the solution to the problem and presents methods for choosing a generalized inverse that satisfies certain requirements. Finally, other methods of control allocation currently in use are examined and compared.

Sample Problem

For the purposes of illustration, we consider a two-dimensional space whose coordinates are the body-axis rolling and yawing moment coefficients C_l and C_n. Associated with these moments we assume three constrained controllers: conventional aileron, operating differentially; horizontal tail, also operating differentially; and a single conventional rudder. This problem is low order but not trivial and demonstrates all of the important features of the more general problem while allowing two- and three-dimensional figures for illustration.

The control derivatives are taken from the NASA Dryden "Controls Design Challenge" model,[4] evaluated at $M = 0.5$, $h = 10{,}000$ ft. Note that the derivatives are expressed as per degree.

$$B = \begin{bmatrix} C_{l_{\delta a}} & C_{l_{\delta HT}} & C_{l_{\delta r}} \\ C_{n_{\delta a}} & C_{n_{\delta HT}} & C_{n_{\delta r}} \end{bmatrix} = \begin{bmatrix} 7.35\times10^{-4} & 7.55\times10^{-4} & -1.35\times10^{-4} \\ 8.56\times10^{-5} & 5.13\times10^{-4} & -1.37\times10^{-3} \end{bmatrix} \tag{1}$$

$$u = \begin{Bmatrix} \delta \text{ aileron} \\ \delta \text{ diff. } HT \\ \delta \text{ rudder} \end{Bmatrix} = \begin{Bmatrix} \delta a \\ \delta HT \\ \delta r \end{Bmatrix} \tag{2}$$

$$m = \begin{Bmatrix} C_l \\ C_n \end{Bmatrix} = Bu \tag{3}$$

The control limits are $\delta a = \pm 20$ deg, $\delta HT = \pm 20$ deg, and $\delta r = \pm 30$ deg. These are the same as for the Dryden model trimmed at the indicated flight condition. As a practical matter, the differential tail limits will vary as the datum, or longitudinal trim position of the symmetric horizontal tail, changes, but for this example that is not important. Deflections are positive for left aileron and left differential tail trailing-edge down (TED), right surface trailing-edge up (TEU). The rudder is positive trailing-edge left (TEL).

Geometry of Constrained Control Allocation

Problem Statement

We consider an m-dimensional control space $u \in R^m$. The controls are constrained to minimum and maximum values, defined by the subset Ω

$$\Omega = \{u \in R^m \mid u_{i\,\min} \le u_i \le u_{i\,\max}\} \subset R^m \tag{4}$$

The subset of controls that lie on the boundary of Ω, $\partial(\Omega)$, are denoted by u^*.

$$u^* \in \partial(\Omega) \tag{5}$$

These controls generate moments through a mapping B onto n-dimensional moment space through a linear matrix multiplication of u,

$$B : R^m \rightarrow R^n \tag{6}$$

$$Bu = m \tag{7}$$

where $m > n$. B is the control effectiveness matrix with respect to the moments. The interpretation of our requirement that the controls be independent is that every $n \times n$ partition of B be nonsingular.

Denote by Φ the image of Ω in R^n, $\Phi \subset R^n$. The subset Φ, therefore, represents all of the moments that are attainable within the constraints of the controls. Moments which lie on the boundary of Φ, $\partial(\Phi)$, are denoted by an asterisk

$$m^* \in \partial(\Phi) \tag{8}$$

A unit vector in the direction m will be denoted by $\hat{m}$,

$$\hat{m} = \frac{m}{|m|} \tag{9}$$

The control allocation problem is defined as follows: given B, Ω, and some desired moment m_d, determine the controls $u \in \Omega$ that generate that moment for the largest possible magnitude of m in the direction $\hat{m}_d$. That is, we desire a rule for allocating the controls that generates the maximum moment in a given ratio (direction) without exceeding the constraints on the controls.

In terms of our sample problem, Ω is defined by the limits given, yielding Fig. 1. All of the admissible controls u lie within or on the figure, and those that lie on its surface are denoted u^*. The moments that are attainable from the constrained controls in Fig. 1 are shown in Fig. 2.

Nomenclature

We will adopt the following nomenclature for referring to $\partial(\Omega)$.

Vertices are the points generated by placing all m of the controls at one or the other of their constraining values. Vertices will be numbered according to a binary representation that reflects which controls are minimum and which are maximum, such that a "0" in the ith significant figure of the binary number indicates the ith control is a minimum, and a "1" indicates it is at a maximum. Thus, in Fig. 1, the vertex generated by $\delta a = -20$ deg, $\delta HT = +20$ deg, and $\delta r = +30$ deg is binary 011, or decimal 3, and will be referred to as vertex 3. As a vector from the origin, vertices of Ω will be denoted by u_i^*, where the single subscript i refers to the number of the vertex.

Edges are lines that connect vertices and that lie on $\partial(\Omega)$. They are generated by varying only one of the m controls whereas the remaining $m-1$ are at the constraining values associated with the two connected vertices. Edges will be distinguished by ij, where i is the from vertex and j is the to vertex. As vectors, the edges are denoted by u_{ij}^*, and it is clear that $u_{ij}^* = u_j^* - u_i^*$ and $u_{ij}^* = -u_{ji}^*$. Note that two vertices are connected by an edge if and only if their binary representations differ in only one bit.

Facets are plane surface on $\partial(\Omega)$ that contain two adjacent edges. Since the two edges are adjacent, they have a common

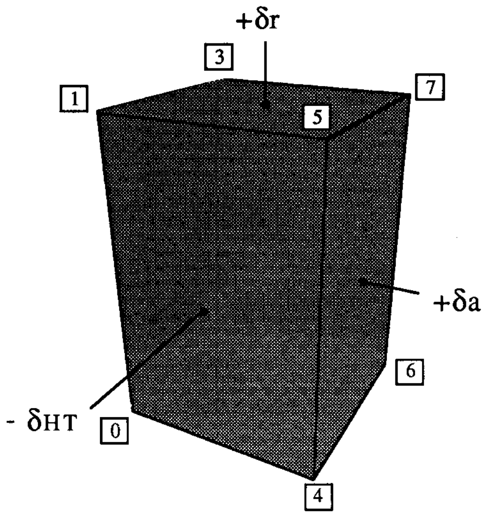

Fig. 1 Constrained control subset Ω.

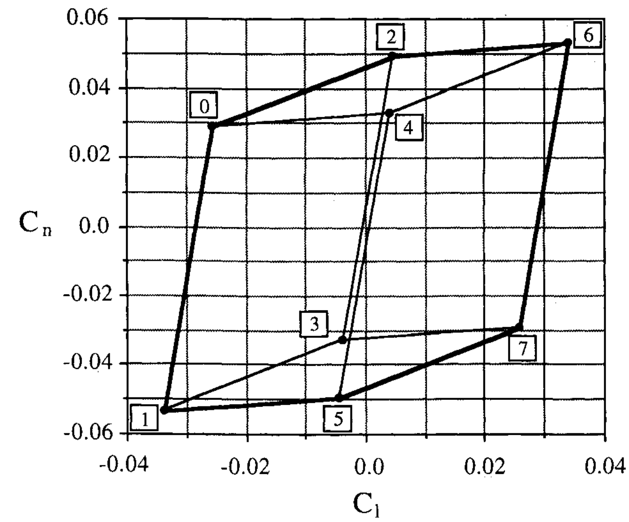

Fig. 2 Attainable moment subset Φ.

from vertex. As vectors, these edges are u_{ij}^* and u_{ik}^*, meaning that three of the vertices of a facet are u_i^*, u_j^*, and u_k^*. All vectors from u_i^* that lie in the plane of a facet are linear combinations of u_{ij}^* and u_{ik}^*, say $au_{ij}^* + bu_{ik}^*$. In order that these points remain on $\partial(\Omega)$ we must have $0 \le a \le 1$ and $0 \le b \le 1$. In particular, if $a = b = 1$, then $u_{ij}^* + u_{ik}^*$ lies on the facet. But to move along u_{ij}^* is equivalent to varying a single control from one extreme to the other, and similarly for u_{ik}^* (varying a different control). Therefore, this new point is itself a vertex, differing from u_i^* in two of its controls (those that generated u_{ij}^* and u_{ik}^*). Thus, facets are rectangles generated by varying exactly two of the controls, and these two controls are those that are allowed to vary along its edges. In Fig. 1, each of the six faces of the "box" are facets.

Since the controls lie in R^n, we could continue defining geometrical aspects of $\partial(\Omega)$. But the projection is into R^3 (or R^2 for our example). Vertices, edges, and facets are all we will need in R^3, and these will be the images of vertices, edges, and facets from R^n.

We modify our nomenclature somewhat in referring to Φ. The projections of vertices from $\partial(\Omega)$ are called *nodes*, unless they lie on the boundary $\partial(\Phi)$, in which case they are called vertices of $\partial(\Phi)$. Vertices and nodes of Φ are numbered to correspond with the vertices of Ω of which they are images. In Fig. 2, u_3^* and u_4^* of $\partial(\Omega)$ map to nodes m_3 and m_4 in Φ, and the remaining vertices map to the six vertices (m_0^*, m_1^*, etc.) of $\partial(\Phi)$.

Edges of $\partial(\Omega)$ project to either *connections* in Φ (interior) or to edges of $\partial(\Phi)$. In Fig. 2, there are six edges and six connections. The vector notation for edges of $\partial(\Phi)$ is analogous to that used for controls.

Facets of $\partial(\Omega)$ project to either *faces* in Φ (interior) or to facets of $\partial(\Phi)$. In our example problem, Φ is two dimensional, so there are no faces or facets. For the somewhat more general problem of three moments, there will be both faces and facets in the attainable moment space.

Direct Solution of the Problem: Approach

The methodology we will adopt is as follows: given B, Ω 1) determine Φ, then 2) find its boundary $\partial(\Phi)$, and, finally, 3) determine $u \in \Omega$ that maps to the points within or on that boundary.

Determination of Φ

This step is straightforward

$$\Phi = \{m \in R^n | Bu = m, u \in \Omega\} \subset R^n \tag{10}$$

That is, the set of all attainable moments is B times all the attainable controls. For the linear problem we have posed, it is sufficient to map $\partial(\Omega)$ onto moment space, then to rely on the fact that we have performed a linear transformation of a compact set of points and so must have generated a compact set of points (see Fig. 2).

Determination of the Boundary ∂(Φ)

The boundary of Φ is not so easily found. The boundary $\partial(\Phi)$ is obviously obtained from $\partial(\Omega)$, but it is not simply the image of $\partial(\Omega)$ since some points of $\partial(\Omega)$ are mapped into the interior of Φ. In Fig. 2, the vertices and edges determine the boundary, whereas the nodes and connections are interior. It is not generally known a priori which parts of $\partial(\Omega)$ map to $\partial(\Phi)$, and $\partial(\Phi)$ must be determined by some other means.

The boundary we seek in this problem is called the convex hull. The convex hull of a set of points may be determined by any of several methods.[5] These methods begin with an arbitrary set of points and extensively use trigonometric relationships to find the convex hull of the set of points. Current research promises more efficient means of determining the boundary for the many-control, three-moment problem. These methods exploit the structure of the subsets and rely on the a priori knowledge of the connectedness of the nodes. For the purposes of this paper, we will assume that the boundary has been found. In our example (Fig. 2), the subset of attainable moments is a plane figure whose boundary is easily determined by inspection.

Determination of u

Given that we have found $\partial(\Phi)$, we postulate an arbitrary demand for some desired combination of moments for the controls to generate. This is a vector in moment space. We need to know if this vector is within the attainable moment subset. To do this, we find the intersection of the half-line in the direction of the desired moment with $\partial(\Phi)$, and compare the length of the desired vector with the distance to this intersection. If the desired moment is on the boundary, we find the controls that generate the point of intersection. If the desired moment is within the boundary, we will scale down the controls that generate the point of intersection. If the desired moment is outside the boundary, then no combination of controls can generate it, and we take the controls that generate the intersection as being the "best" we can do.

The required steps are as follows: given $\partial(\Phi)$ and a desired moment m_d,

1) Determine which edge or facet of $\partial(\Phi)$ m_d points to. It is on this edge or facet that the maximum attainable moment in the desired direction lies. For the sample (two-moment) problem, we simply note the angles subtended by each edge with respect to some reference (say, $C_n = 0$), find the angle of m_d, and then determine the edge within whose range of angles the desired moment lies. For the three-moment problem, determination of the facet to which the desired moment points is decidedly nontrivial. The problem is complicated by the fact that the facets appear as parallelograms on the surface of a three-dimensional figure resembling a mad cubist's design of a geodesic dome. Preliminary research into the determination of the intersection for the three-moment problem has produced a fast but highly inelegant algorithm for determining the correct facet. These results will be reported separately.

2) Find the intersection of am_d, $a > 0$, with the edge or facet determined in step 1. This is the maximum attainable moment in the desired direction. If $a \ge |m_d|$, the desired moment is attainable, otherwise not. This is a straightforward linear algebra problem, that of finding the intersection of two lines, or of a line and a plane.

3) Given the intersection, calculate the controls that generate that point. The intersection will be the vector sum of an adjoining vertex and some positive fractional part of either the vector that describes the edge (two-moment problem) or the two vectors that define the facet (three-moment problem). The controls that generate that point are determined by adding the corresponding control vectors from Ω in the same ratio.

4) If $a = |m_d|$ (from step 2), step 3 yields the desired solution. If $a > |m_d|$, the controls obtained from step 3 may be scaled down by the factor $K = |m_d|/a$ to obtain a solution. If $a < |m_d|$ no solution is possible, but the results from step 3 may be used as being the best possible solution. It is only best in the sense that the control generated moments are in the same direction as desired, and says nothing about the effect on the resulting angular accelerations.

Example Problem

In Fig. 2, the boundary is easily seen to be the edges from vertices 0-2-6-7-5-1-0. All of the points (combinations of moments) on and within this boundary are attainable by some combination of controls; no points outside the boundary can be obtained without violating some control constraint. The edges that bound the plane figure, therefore, represent the maximum moments available in any particular direction in moment space, using any allowable combination of controls.

To illustrate the procedure, we take for our desired moment $C_l = 4.0 \times 10^{-2}$, $C_n = 0$. Thus,

$$m_d = |m_d|\hat{m}_d = 4.0 \times 10^{-4}\begin{Bmatrix}1\\0\end{Bmatrix} \tag{11}$$

The relevant geometry of this problem is

$$m_7^* = B\begin{Bmatrix}20\text{ deg}\\20\text{ deg}\\30\text{ deg}\end{Bmatrix} = \begin{Bmatrix}2.5750\times10^{-2}\\-2.9128\times10^{-2}\end{Bmatrix} \tag{12}$$

$$m_6^* = B\begin{Bmatrix}20\text{ deg}\\20\text{ deg}\\-30\text{ deg}\end{Bmatrix} = \begin{Bmatrix}3.3850\times10^{-2}\\5.3072\times10^{-2}\end{Bmatrix} \tag{13}$$

$$m_{76}^* = m_6^* - m_7^* = \begin{Bmatrix}8.1000\times10^{-3}\\8.2195\times10^{-2}\end{Bmatrix} \tag{14}$$

Figure 3 shows the geometry of this problem.

As for step 1 of our procedure, we note the angle of m_7^* with the positive C_n axis is -48.5 deg, that of m_6^* is $+57.5$ deg, and that edge m_{76}^* subtends these angles. Our desired moment has angle zero (between -48.5 and $+57.5$ deg), so m_{76}^* is the correct edge.

At the intersection, we have

$$a\hat{m} = m_7^* + bm_{76}^* \tag{15}$$

$$[\hat{m}_d : -m_{76}^*]\begin{Bmatrix}a\\b\end{Bmatrix} = m_7^* \tag{16}$$

The scalar a is related to K according to $K = |m_d|/a$. The matrix $[\hat{m}_d : m_{76}^*]$ must be nonsingular or the desired moment is parallel to the edge, and we have picked the wrong edge in step 1. Therefore, we solve for a and b,

$$\begin{Bmatrix}a\\b\end{Bmatrix} = \begin{Bmatrix}2.862\times10^{-2}\\3.544\times10^{-1}\end{Bmatrix} \tag{17}$$

The scalar b must lie between 0 and 1 or again we have picked the wrong edge. We note that for this solution $K = 1.4$, and the moment is not attainable (as was obvious from Fig. 3). Step 3 is then

$$u^* = u_7^* + 0.354u_{76}^* = \begin{Bmatrix}20\text{ deg}\\20\text{ deg}\\30\text{ deg}\end{Bmatrix} + 0.354\begin{Bmatrix}0\text{ deg}\\0\text{ deg}\\-60\text{ deg}\end{Bmatrix} = \begin{Bmatrix}20\text{ deg}\\20\text{ deg}\\8.76\text{ deg}\end{Bmatrix} \tag{18}$$

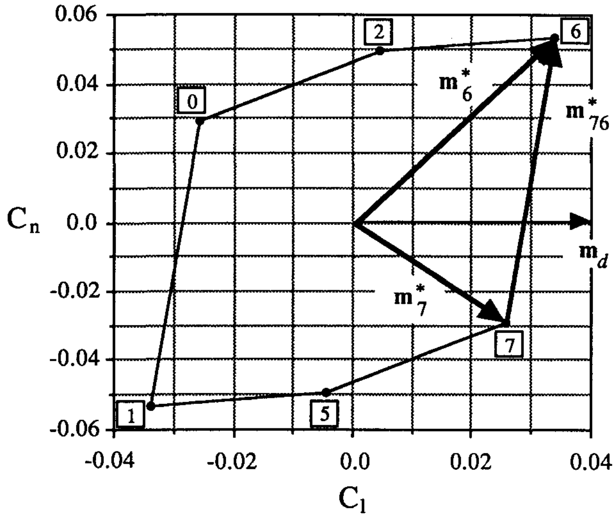

Fig. 3 Example problem.

This control, an aileron deflection of 20 deg, differential horizontal tail 20 deg, and rudder 8.76 deg, generates the maximum amount of rolling moment available with zero yawing moment. Since $K = 1.4$, according to step 4 we may either accept this value or redefine our requirement.

Had we taken for our desired moment $C_l = 1.0 \times 10^{-2}$, $C_n = 0$, we would have had $K = 0.35$. In that case, the desired combination of controls would have been

$$u = 0.35\begin{Bmatrix}20\text{ deg}\\20\text{ deg}\\8.76\text{ deg}\end{Bmatrix} = \begin{Bmatrix}7\text{ deg}\\7\text{ deg}\\3.07\text{ deg}\end{Bmatrix} \tag{19}$$

We have effectively designed a RSRI in which the aileron and differential horizontal tail are ganged in a 1:1 ratio for roll generation. The ratio 20:20:8.76 is the RSRI gearing for this flight condition, and controls in this combination generate rolling moments with no yaw.

Role of Generalized Inverses

Implementations of control laws using direct solutions to the allocation problem are complicated by the requirement to determine the correct edge or facet for solution. This determination is fast and easy for the two-moment problem, and is certainly amenable to real-time implementation. Much more research is required to determine practical methods for three-moment and higher order problems. For the time being, a simpler solution for higher order problems is preferable.

By simpler solutions we are thinking of multiplying the vector of desired moments by a single matrix and getting the right controls. In the parlance of linear algebra, this matrix is a generalized inverse.[6,7] If the controls are unconstrained there are an infinite number of generalized inverses that solve the problem. The most commonly used of these is the minimum-norm solution, which is also called the pseudoinverse. The use of the pseudoinverse, with its simple closed-form expression in terms of the control effectiveness matrix, so dominates the literature that it is difficult to find counter examples.

Several methods have been addressed that "normalize" the control effectiveness matrix prior to applying the pseudoinverse calculations.[2] In its simplest form, this normalization attempts to make the matrix representative of controls that have limits of plus and minus one unit, where the units are consistent with the controllers' effects on angular accelerations. Although such methods may improve the solution (in terms of more nearly attaining greater control moments within the constraints on the controls), it will be shown that no method can attain the maximum available moment everywhere without exceeding some control constraints.

We will present two methods of determining the best generalized inverse for a specific problem. The first is referred to as tailoring of the generalized inverse to satisfy requirements for maximum moment generation in a limited number of specified directions in moment space. The second method finds the generalized inverse that minimizes the difference between the attainable moment subset and that achievable by any generalized inverse. With respect to generalized inverses, we have three questions.

1) Does a generalized inverse solution to the control allocation problem exist that yields solutions everywhere in Φ or on $\partial(\Phi)$ without violating constraints?

2) If not, may we select a generalized inverse solution that satisfies specific requirements?

3) Is there a best generalized inverse, one that yields solutions that approximate $\partial(\Phi)$ more closely than any other?

To answer the questions we have posed, we will first examine how generalized inverse solutions "fit" inside the boundary. The following theorems help define the problem. The theorems used in this paper are consequences of the linearity of the problem, and their proofs are omitted. Given the control allocation problem with $\boldsymbol{Bu} = \boldsymbol{m}$, $\boldsymbol{u} \in \Omega$, denote by P any generalized inverse of B that satisfies $BP = I_n$ and which yields controls corresponding to moments according to $\boldsymbol{u} = \boldsymbol{Pm}$.

Theorems

Theorem 1. Denote by Ψ the subset of $\partial(\Omega)$ that maps to $\partial(\Phi)$, and by $\boldsymbol{u}'$ the vectors in Ψ. Then $\text{span}(\Psi) = R^m$.

P must satisfy $\boldsymbol{u}' = \boldsymbol{PBu}' \forall \boldsymbol{u}'$, or $[PB - I]\boldsymbol{u}' = \boldsymbol{0}\ \forall \boldsymbol{u}'$. That is, the constrained control vectors that map to $\partial(\Phi)$ must lie in $\aleph[PB - I]$, the null space of $[PB - I]$

$$\boldsymbol{u}' = \{\boldsymbol{u} \in \Omega \cap \aleph\,[PB - I] \mid B\boldsymbol{u} \in \Phi\}$$

This null space will play an important role in our subsequent analysis.

Theorem 2. The equation $\boldsymbol{u} = \boldsymbol{Pm}$ can be satisfied exactly at no more than $(m - n)\cdot n$ arbitrary values of m.

The significance of Theorem 2 is that we can "anchor" the generalized inverse at no more than $(m - n)\cdot n$ arbitrary points on the boundary. All of the other points on the boundary that are satisfied by the generalized inverse do so by virtue of the next theorem.

Theorem 3. If a generalized inverse satisfies

$$P\boldsymbol{m}_1^* = \boldsymbol{u}_1^* \qquad P\boldsymbol{m}_2^* = \boldsymbol{u}_2^* \tag{20}$$

and if

$$\exists \boldsymbol{m}_3^* = B(a\boldsymbol{u}_1^* + b\boldsymbol{u}_2^*) \tag{21}$$

then

$$P\boldsymbol{m}_3^* = a\boldsymbol{u}_1^* + b\boldsymbol{u}_2^* \tag{22}$$

Theorem 3 has application as follows: If the generalized inverse is selected to satisfy $(m - n)\cdot n$ arbitrary points on the boundary, and if there is another point on the boundary achieved by some linear combination of the controls at the selected points (e.g., the linear combinations defining the null space of $[PB - I]$), then the generalized inverse satisfies the other point as well (i.e., lies in the null space as well).

As further consequences of Theorem 3, if a generalized inverse satisfies any two points on a given edge, then it satisfies all points on that edge. If the problem is symmetric (if the control limits are symmetric about zero deflection) then we may conclude that if a generalized inverse satisfies a certain point on the boundary, then points of control (and moment) symmetry are also satisfied by the generalized inverse.

Null Space

The pseudoinverse is one of a family of generalized inverses and is the one which yields minimum control energy. It is easily evaluated as $P_{\text{min norm}} = B^T[BB^T]^{-1}$ (Ref. 8). The family of which this is a special case may be expressed as $P = W[BW]^{-1}$, where W must be selected such that the inverse exists. It appears from this formulation that the $m\cdot n$ terms in W are parameters available for selecting a different P. However, the terms in W are not all independent. We will use a different characterization of P that has the right number of parameters.[7]

We proceed as follows: partition B (rearranging if necessary) as

$$B = [B_1 : B_2], \quad B_1 \in R^{n\times n}, \quad |B_1| \neq 0, \quad B_2 \in R^{n\times(m-n)} \tag{23}$$

Partition P conformably,

$$P = \begin{bmatrix} P_1 \\ \cdots \\ P_2 \end{bmatrix} \tag{24}$$

Then,

$$BP = I \Leftrightarrow B_1P_1 + B_2P_2 = I \Leftrightarrow P_1 = B_1^{-1} - B_1^{-1}B_2P_2 \tag{25}$$

For any choice of P_2, P is completely determined.[7] With the partitioning given,

$$PB - I = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}[B_1\ B_2] - I$$

$$= \begin{bmatrix} \{-B_1^{-1}B_2[P_2B_1]\} & \{-B_1^{-1}B_2[P_2B_2 - I]\} \\ \{P_2B_1\} & \{P_2B_2 - I\} \end{bmatrix} \tag{26}$$

Therefore, if

$$\left[\{P_2B_1\}\ \{P_2B_2 - I\}\right]\boldsymbol{u} = \boldsymbol{0} \tag{27}$$

then

$$\left[\{-B_1^{-1}B_2[P_2B_1]\}\{-B_1^{-1}B_2[P_2B_2 - I]\}\right]\boldsymbol{u}$$

$$= -B_1^{-1}B_2\left[\{P_2B_1\}\{P_2B_2 - I\}\right]\boldsymbol{u}$$

$$= \boldsymbol{0} \tag{28}$$

For a given P_2 and B, Eq. (27) completely defines the null space $\aleph[PB - I]$. We are now ready to answer the questions posed earlier.

Existence of an Exact Generalized Inverse Solution

Question 1 asks if any P exists that satisfies all points in Φ, but especially in $\partial(\Phi)$ without violating the control constraints. That is,

$$\exists ? P \mid \boldsymbol{u} = P\boldsymbol{m}^*, \qquad \boldsymbol{u} \in \Omega, \qquad \forall \boldsymbol{m}^* \tag{29}$$

In Theorem 1 we identified such controls as $\boldsymbol{u}'$. By Theorem 1 the controls $\boldsymbol{u}'$ span R^m, so they cannot all lie in $\aleph[PB - I]$. Therefore, there is *no* generalized inverse that yields solutions that are everywhere on the boundary.

Tailoring the Generalized Inverse

There are $(m - n)\cdot n$ arbitrary selections available for specifying the fit of the generalized inverse, and various choices are available for assigning them. Selection of any two points on an edge nets the whole edge, three points on a facet nets the whole facet, and so on. So, for example, a generalized inverse may be made to fit the boundary in Fig. 2 along the whole of edge (0–2) [and of edge (5–7) by symmetry]. Although this exacts a large penalty in the directions of vertices 1 and 6, it does yield maximum yawing moment with no rolling moment, which will lower the minimum controllable airspeed in conditions of large yawing moments.

The method we will use to specify the points at which the generalized inverse exactly matches the boundary of attainable moments is based on Eq. (25), which effectively demonstrates that a choice of P_2 completely specifies P. At a particular point on the boundary of Φ, both $\boldsymbol{m}^*$ and $\boldsymbol{u}^*$ are known. Therefore, we wish to solve for P in the equation $P\boldsymbol{m}^* = \boldsymbol{u}^*$.

Partition $\boldsymbol{u}^*$ conformably with P

$$\boldsymbol{u}^* = \begin{Bmatrix} \boldsymbol{u}_1^* \\ \boldsymbol{u}_2^* \end{Bmatrix}, \qquad \boldsymbol{u}_1^* \in R^{n \times 1}, \qquad \boldsymbol{u}_2^* \in R^{(m-n) \times 1} \tag{30}$$

The order of the problem is reduced, and we need only to solve for P_2 in $P_2\boldsymbol{m}^* = \boldsymbol{u}_2^*$. In our sample problem, $(m-n)\cdot n = 2$. For purposes of illustration, we will determine a generalized inverse that fits *exactly* at two points in our sample problem: maximum roll with no yaw and maximum yaw with no roll. We denote the two points at which the generalized inverse is to fit as $\boldsymbol{M}_1^*$ and $\boldsymbol{M}_2^*$. The corresponding controls are denoted $\boldsymbol{U}_1^*$ and $\boldsymbol{U}_2^*$, partitioned as

$$\boldsymbol{U}_1^* = [\boldsymbol{U}_{1,1}^*\, \boldsymbol{U}_{1,2}^*]^T \qquad \text{and} \qquad \boldsymbol{U}_2^* = [\boldsymbol{U}_{2,1}^*\, \boldsymbol{U}_{2,2}^*]^T$$

The controls and moments at these points are determined using the direct method just described.

$$\boldsymbol{U}_1^* = \begin{Bmatrix} 20 \text{ deg} \\ 20 \text{ deg} \\ 8.76 \text{ deg} \end{Bmatrix}, \qquad \boldsymbol{U}_{1,2}^* = \{8.76 \text{ deg}\} \tag{31}$$

$$\boldsymbol{M}_1^* = \begin{Bmatrix} 0.0286 \\ 0 \end{Bmatrix} \tag{32}$$

$$\boldsymbol{U}_2^* = \begin{Bmatrix} -20 \text{ deg} \\ 14.11 \text{ deg} \\ -30 \text{ deg} \end{Bmatrix}, \qquad \boldsymbol{U}_{2,2}^* = \{-30 \text{ deg}\} \tag{33}$$

$$\boldsymbol{M}_2^* = \begin{Bmatrix} 0 \\ 0.0465 \end{Bmatrix} \tag{34}$$

P_2 is a row vector, so we may write

$$P_2\boldsymbol{M}_1^* = \boldsymbol{M}_1^{*T} P_2^T = \boldsymbol{U}_{1,2}^{*\,T}$$

$$P_2\boldsymbol{M}_2^* = \boldsymbol{M}_2^{*T} P_2^T = \boldsymbol{U}_{2,2}^{*\,T}$$

$$\begin{bmatrix} \boldsymbol{M}_1^{*T} \\ \boldsymbol{M}_2^{*T} \end{bmatrix} P_2^T = \begin{Bmatrix} \boldsymbol{U}_{1,2}^* \\ \boldsymbol{U}_{2,2}^* \end{Bmatrix}^T$$

$$P_2^T = \begin{bmatrix} \boldsymbol{M}_1^{*T} \\ \boldsymbol{M}_2^{*T} \end{bmatrix}^{-1} \begin{Bmatrix} \boldsymbol{U}_{1,2}^* \\ \boldsymbol{U}_{2,2}^* \end{Bmatrix}^T \tag{35}$$

We use Eq. (35) to determine P_2

$$P_2^T = \begin{bmatrix} 0.0286 & 0 \\ 0 & 0.0465 \end{bmatrix}^{-1} \begin{Bmatrix} 8.76 \\ -30 \end{Bmatrix} = \begin{bmatrix} 306.1 \\ -645.2 \end{bmatrix}$$

We may now use Eq. (25) to get the rest of P:

$$P_1 = B_1^{-1} - B_1^{-1} B_2 P_2$$

where

$$B_1 = \begin{bmatrix} 7.35 \times 10^{-4} & 7.55 \times 10^{-4} \\ 8.56 \times 10^{-5} & 5.13 \times 10^{-4} \end{bmatrix}, \quad B_2 = \begin{bmatrix} -1.35 \times 10^{-4} \\ -1.37 \times 10^{-3} \end{bmatrix}$$

This operation yields

$$P = \begin{bmatrix} 696.4 & -423.5 \\ 701.3 & 296.9 \\ 306.1 & -645.2 \end{bmatrix} \tag{36}$$

There will be a third intersection on edge $\boldsymbol{m}_{15}^*$ (as well as $\boldsymbol{m}_{26}^*$ by symmetry), and the controls at this intersection will lie in $\aleph\,[PB - I]$ using the just-determined matrix P. We use Eq. (27) on $\boldsymbol{u}_{15}^*$

$$[0.170 \quad -0.100 \quad -0.157] \begin{Bmatrix} u_1 \\ -20 \text{ deg} \\ 30 \text{ deg} \end{Bmatrix} = \boldsymbol{0} \tag{37}$$

This is solved to yield $u_1 = 15.9$ deg. The corresponding moment at this third intersection is found to be

$$\boldsymbol{m} = \begin{Bmatrix} -7.46 \times 10^{-3} \\ -5.0 \times 10^{-2} \end{Bmatrix} \tag{38}$$

Figure 4 shows the moment space covered by the generalized inverse we have determined [Eq. (36)]. The lightly shaded area is attainable with our generalized inverse, and the darkly shaded area is not. Clearly, a lot of capability has been given up to satisfy this specific requirement.

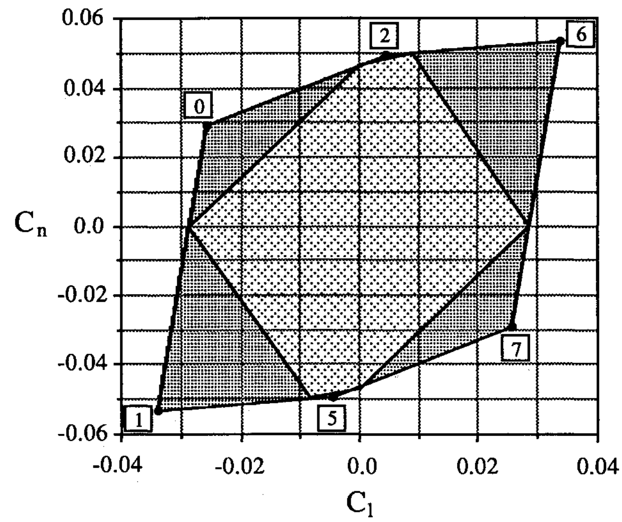

Fig. 4 Solution to the two-point fit problem.

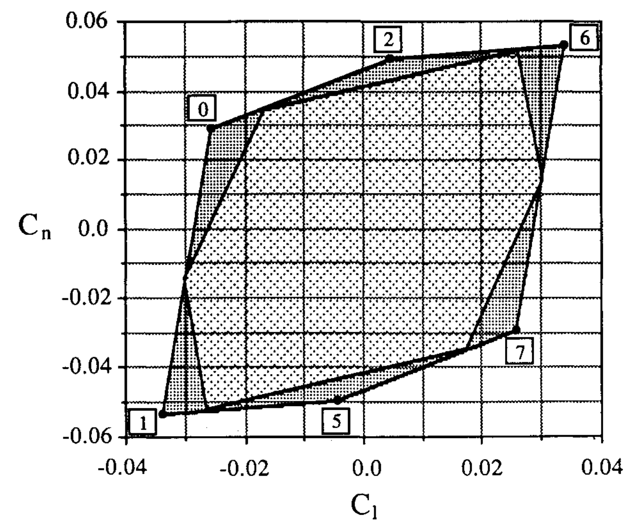

Fig. 5 Minimum-norm solution.

Determination of the Best Generalized Inverse

To find the generalized inverse that most closely approximates the boundary within the constraints, we will formulate a minimization problem whose solution determines the best generalized inverse.

We will define the best generalized inverse as that which maximizes the area or volume of attainable moment space without violating any control constraints. This is clearly the same as minimizing the difference between the area or volume of the boundary and that attainable by the generalized inverse within the constraints. The geometry of the problem is such that at each vertex a triangular or pyramidal area is described by parts of the edges of the boundary and the lines of single saturated control associated with the generalized inverse solution. Figure 5, representing the minimum-norm solution, illustrates this. The lightly shaded area indicates those moments obtained by the minimum-norm solution within the constraints of the controls. The heavily shaded areas are to be minimized by the choice of some different generalized inverse.

To minimize the sum of all such areas or volumes, we are permitted to hypothesize $(m-n)\cdot n$ arbitrary points of intersection of the generalized inverse solution (Theorem 2). Based on this selection, other points of intersection will result (Theorem 3).

For the two-moment problem we use the fact that the area of the parallelogram is equal to the magnitude of the cross product of two adjacent vectors defining it, and the area we wish to minimize is half a parallelogram. We will use this for our sample problem, but note in passing that the triple scalar product does the same thing for the pyramidal shapes in three-moment problems.

Consider an arbitrary vertex $\boldsymbol{m}_i^*$. Hypothesize generalized inverse intersections at each of the edges from that vertex. Denote by $\boldsymbol{m}_j'$ and $\boldsymbol{m}_k'$ the vectors from the origin to each of the intersections. The area we wish to minimize is given by one-half the magnitude of the cross product of the vectors $\{\boldsymbol{m}_i^*-\boldsymbol{m}_j'\}$ and $\{\boldsymbol{m}_i^*-\boldsymbol{m}_k'\}$. The magnitude of this cross product is the product of the magnitudes of the vectors and the sine of the angle between them. Since the angle between these vectors is fixed at any vertex, we conclude that it is sufficient to minimize the product of the lengths of the vectors. This is to be done at each vertex and summed over all of the vertices. For our sample problem, the geometry associated with this discussion is shown in Fig. 6. The area of the cross product is hatched vertically, and the area to be minimized (one-half the cross product area) is cross hatched. It is understood that a similar area is associated with each of the six vertices.

The parameters available for minimizing this sum of products are the elements of P_2, and the constraint is that the controls associated with the generalized inverse solution lie in $\aleph[PB-I]$. Rather than adjoin the constraints, we will incorporate them into the function to be minimized.

In our example problem, we begin with any vertex, say $\boldsymbol{m}_0^*$. On its adjacent edges, hypothesize intersections $\boldsymbol{m}_1'$ on the edge to $\boldsymbol{m}_1^*$ and $\boldsymbol{m}_2'$ on the edge to $\boldsymbol{m}_2^*$. Along each edge, two of the controls are saturated; associated with $\boldsymbol{m}_1'$ we have

$$\boldsymbol{u}_1' = \begin{Bmatrix} u_{1\,\min} \\ u_{2\,\min} \\ u_3 \end{Bmatrix} \tag{39}$$

Associated with $\boldsymbol{m}_2'$,

$$\boldsymbol{u}_2' = \begin{Bmatrix} u_{1\,\min} \\ u_2 \\ u_{3\,\min} \end{Bmatrix} \tag{40}$$

The only other edge (except by symmetry) at which there can be an intersection is that from $\boldsymbol{m}_2^*$ to $\boldsymbol{m}_6^*$. Denote the intersection along this edge as $\boldsymbol{m}_3'$, associated with which we have

$$\boldsymbol{u}_3' = \begin{Bmatrix} u_1 \\ u_{2\,\max} \\ u_{3\,\min} \end{Bmatrix} \tag{41}$$

For a given choice of P_2, the "free" controls at each intersection (u_1, u_2, and u_3) are determined by the requirement that the controls at the point selected on each of these three edges lie in $\aleph[PB-I]$. Then each intersection is calculated from $\boldsymbol{m}_i' = B\boldsymbol{u}_i'$. We may then formulate a cost function to be minimized that is a function of P_2 as follows:

$$\min_{P_2} J$$

where

$$J = |\boldsymbol{m}_0^*-\boldsymbol{m}_1'|\cdot|\boldsymbol{m}_0^*-\boldsymbol{m}_2'| + |\boldsymbol{m}_1^*-\boldsymbol{m}_1'|\cdot|\boldsymbol{m}_1^*+\boldsymbol{m}_3'| + |\boldsymbol{m}_2^*-\boldsymbol{m}_2'|\cdot|\boldsymbol{m}_2^*-\boldsymbol{m}_3'| \tag{42}$$

$$\boldsymbol{m}_0^* = B\begin{Bmatrix} u_{1\,\min} \\ u_{2\,\min} \\ u_{3\,\min} \end{Bmatrix},\quad \boldsymbol{m}_1^* = B\begin{Bmatrix} u_{1\,\min} \\ u_{2\,\min} \\ u_{3\,\max} \end{Bmatrix},\quad \boldsymbol{m}_2^* = B\begin{Bmatrix} u_{1\,\min} \\ u_{2\,\max} \\ u_{3\,\min} \end{Bmatrix} \tag{43}$$

$$\boldsymbol{m}_1' = B\begin{Bmatrix} u_{1\,\min} \\ u_{2\,\min} \\ u_3 \end{Bmatrix},\quad \boldsymbol{m}_2' = B\begin{Bmatrix} u_{1\,\min} \\ u_2 \\ u_{3\,\min} \end{Bmatrix},\quad \boldsymbol{m}_3' = B\begin{Bmatrix} u_1 \\ u_{2\,\max} \\ u_{3\,\min} \end{Bmatrix} \tag{44}$$

$$\big[\{P_2B_1\}\{P_2B_2-I\}\big]\boldsymbol{m}_i' = \boldsymbol{0}, \qquad i = 1, 2, 3 \tag{45}$$

Equation (42) defines the cost function as the product of the lengths of the vectors along adjacent edges from three of the six vertices, the other three being symmetric to these. Equation (43) defines the three vertices appearing in the cost function. Equations (44) and (45) specify the free controls on each of the three edges such that they lie in the null space. Each of these is solved for the appropriate free control on its respective edge. The components of P_2 appearing in these equations are the parameters to be varied in the minimization.

The preceding formulation was applied to the example problem. Minimization was performed using the IMSL subroutine DUMINF.[9] The initial guess for P_2 was the P_2 of the minimum-norm solution, namely, [287.3 −723.0]. The algorithm did converge, but convergence was accompanied by a warning that the solution may be an approximate local minimum. The reason for this may be seen in the results, which were

$$P = \begin{bmatrix} 699 & -68.9 \\ 699 & -68.9 \\ 305 & -760 \end{bmatrix} \tag{46}$$

$$u_1 = +20.0\text{ deg},\quad u_2 = -20.0\text{ deg},\quad u_3 = +27.3\text{ deg} \tag{47}$$

$$\boldsymbol{m}_1' = \begin{Bmatrix} -0.0335 \\ -0.0494 \end{Bmatrix},\quad \boldsymbol{m}_2' = \begin{Bmatrix} -0.0258 \\ -0.0291 \end{Bmatrix},\quad \boldsymbol{m}_3' = \begin{Bmatrix} 0.0339 \\ 0.0531 \end{Bmatrix} \tag{48}$$

When overlaid on Φ, Fig. 7 results. The minimization process appears to have "anchored" the generalized inverse at vertices $\boldsymbol{m}_0^*$ and $\boldsymbol{m}_1^*$ (and at 6 and 7 by symmetry). The value of u_3 (27.3 deg) places $\boldsymbol{m}_2'$ on the edge $\boldsymbol{m}_{01}^*$, but note that the same figure is achieved for *any* value of u_3 that places $\boldsymbol{m}_2'$ on that edge. It is believed that there is thus a certain "flatness" to the cost function in the vicinity of the minimum, and the complaints from DUMINF are to be expected.

Comparison with Other Methods

Two other methods of control allocation are considered. These are referred to as "pseudocontrols" and "daisy chaining."

Pseudocontrols

In the original formulation of the "pseudocontrols" method,[10,11] one postulates pseudocontrols as being pure mode controllers (Dutch roll, roll and spiral modes, for example). This is accomplished by aligning the effects of the pseudocontrols with the eigenvectors associated with the modes. These pseudocontrols are dereferenced to the actual aerodynamic and thrust-vectoring controls through an allocation scheme based on relative effectiveness. In recent applications of this method,[12] the pseudocontrols are related to stability axis moment generators rather than mode controllers. Following some preprocessing (including the removal of inertial coupling terms), the pseudocontrols are taken to be the required control moment coefficients. Thus, the method of pseudocontrols may be identified as a form of dynamic inversion, in which only the roll and yaw axes are considered.

The subsequent allocation (distribution) of the physical controls according the required pseudocontrols is accomplished through multiplication by a single control mixing matrix. Since the pseudocontrols are required moments, the control mixing matrix is a generalized inverse, and the remarks made earlier regarding generalized inverse apply to the pseudocontrol method as well. Maximum attainable moments in arbitrary directions are not generated by the pseudocontrol method. The optimality of this method[12] lies in the fact that one can command pure rolling moments or pure yawing moments.

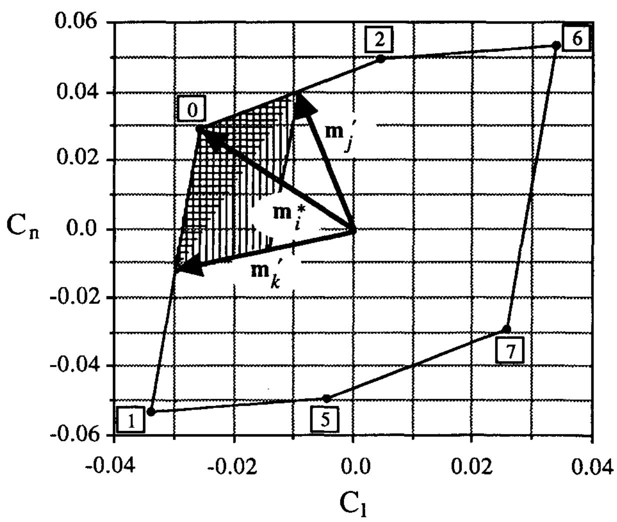

Fig. 6 Area to be minimized (similar for each vertex).

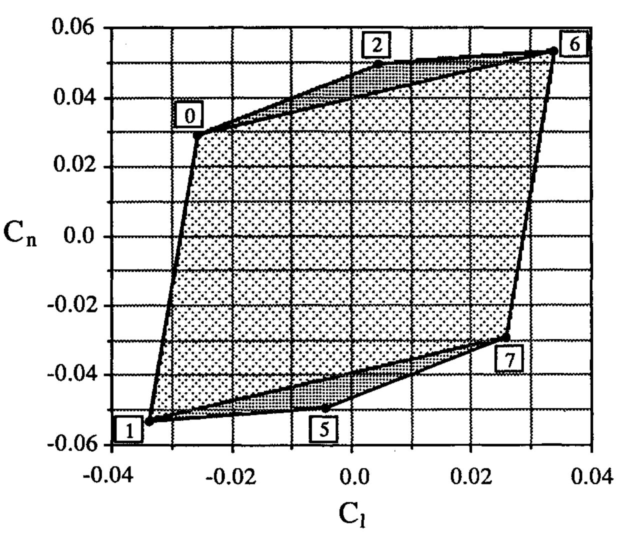

Fig. 7 "Best" generalized inverse.

Daisy Chaining

The method of "daisy chaining" involves the separation of the available controls into two or more groups. Each group is complete in the sense that arbitrary combinations of required moments may be generated by each group. In responding to demands for control generated moments, a particular group is applied while the others are held constant. As any of the controls in the group being applied reaches saturation, that group is held in its last position and another group is brought online to continue generating required moments.

The method has been applied to a dynamic inversion HARV control allocation problem.[13] Two groups of controls were selected: three aerodynamic controls $\boldsymbol{u}_1$ and three thrust-vectoring controls $\boldsymbol{u}_2$. With three moments to be generated, the control effectiveness matrix is partitioned into two 3×3 matrices, B_1 and B_2

$$Bu = [B_1 \quad B_2]\begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = B_1 u_1 + B_2 u_2 \tag{49}$$

For a given moment to be generated, the aerodynamic controls are first used to the point of saturation, then thrust-vectoring controls are brought to bear. That is, while none of the aerodynamic controls are saturated,

$$u_1 = B_1^{-1} m_d \qquad u_2 = \mathbf{0} \tag{50}$$

When any of the aerodynamic controls are saturated,

$$u_1 = u_{1(\text{sat})} \qquad u_2 = B_2^{-1}\{m_d - B_1 u_{1\,(\text{sat})}\} \tag{51}$$

Since in this application B_1 and B_2 are square and assumed invertible, their inverses are unique, and the solutions obtained are unique for the particular control groupings chosen. Other combinations of controls are possible, of course, so the solutions obtained from daisy chaining are not unique in the same sense that those on the boundary of the attainable moment set are. In general, daisy chaining will yield maximum attainable moments in only a very limited number of directions in moment space.

With respect to generating maximum attainable moments, daisy chaining may be compared to the use of generalized inverses. Limiting solutions afforded by generalized inverses are generally characterized by a single controller reaching saturation, whereas daisy chaining guarantees that more than one (one each from the control groupings) will be saturated. However, this does not necessarily mean that daisy chaining yields limiting solutions that are more nearly on the boundary of the attainable moment set than generalized inverses. The superiority of one method over the other in this regard is determined by many factors and will vary according to the direction in moment space being considered.

A disadvantage of daisy chaining arises in consideration of control deflection *rates*. Cooperative control efforts are those in which all available controls are simultaneously varied to meet a time-varying demand. The total rate of change of the moment produced is a linear combination of the individual control rates. For a given rate of change of the required moment, in magnitude and/or direction, a cooperative effort among all available controls will require lower individual control rates than will a noncooperative effort. Daisy chaining is a noncooperative allocation scheme and potentially will command unattainable deflection rates that would not be commanded by cooperative control allocation methods.

Conclusions

The geometry of constrained control problems is straightforward, requiring only basic linear algebra concepts and a means to determine the bounding surface of the attainable moment space. Determination of this bounding surface is the most difficult part of the process. There are existing methods for determining the boundary. Other methods that exploit the structure of the constrained control allocation problem, and that apply to many controls and three moments, are the subject of ongoing research.

The unraveling of the problem to determine the controls that generate a particular moment on the surface of the attainable moment subset is also easy, once the applicable part of the bounding surface is identified. The method of finding the applicable edge described herein, that of using the angles subtended by bounding edges, works well for two-moment problems. For three-moment problems, more ingenuity is required. A simple dynamic inversion control law using three-moment control allocation with 11 controls is currently being tested as part of research in this area.

The direct determination of constrained controls is computationally more complicated than other control allocation methods. It offers advantages that may outweigh this additional complexity in certain applications: First, the direct method is guaranteed to yield the maximum attainable moments. The amount of extra capability that it offers is application specific, and no generalizations are possible. Second, the direct method is a fully cooperative allocation scheme and does not suffer from unnecessary rate limiting.

This paper has presented several tools for the design of flight control systems. First, one may determine and graphically evaluate the maximum capabilities of a particular control configuration by plotting the attainable moment subset. Such an evaluation may be used to determine whether additional control capabilities are required, or whether existing controls offer little additional capability and may be eliminated. It is also useful for direct comparison of the maximum capabilities of two different control configurations. Second, simple allocation schemes using generalized inverses may be directly designed using either a tailored inverse or the "best" inverse. Finally, given a particular control configuration, one may evaluate many potential control allocation schemes with respect to the maximum capabilities. This will enable the designer to decide whether it is necessary to use a complex alllocation scheme, or if a simpler scheme will suffice.

Acknowledgment

A major portion of this work was conducted under NASA Research Cooperative Agreement NCC1-158.

References

[1]"F/A-18A Flight Control System Design Report," McDonnell Aircraft Corp., MDC A7813, March 15, 1983.

[2]Snell, S. A., Enns, D. F., and Garrard, W. L., "Nonlinear Inversion Flight Control for a Supermaneuverable Aircraft," *Proceedings of the AIAA Guidance, Navigation, and Control Conference* (Portland, OR), AIAA, Washington, DC, 1990, pp. 808–817 (AIAA Paper 90-3406).

[3]"Design Methods for Integrated Control Systems," Air Force Wright Aeronautical Lab., AFWAL-TR-88-2061, June 1988.

[4]Brumbaugh, R., "A High Performance Aircraft Model," *Proceedings of the AIAA Guidance, Navigation, and Control Conference* (New Orleans, LA), AIAA, Washington, DC, 1991 (AIAA Paper 91-2631).

[5]Sedgewick, R., *Algorithms,* 2nd ed., Addison-Wesley, Reading, MA, 1988.

[6]Boullion, T. L., and Odell, L., *Generalized Inverse Matrices,* Wiley, New York, 1971.

[7]Ficken, F. A., *Linear Transformation and Matrices,* Prentice-Hall, Englewood Cliffs, NJ, 1967.

[8]Brogan, W. L., *Modern Control Theory,* 2nd ed., Prentice-Hall, Englewood Cliffs, NJ, 1982.

[9]IMSL, *IMSL Math/Library User's Manual,* Ver. 1.1, Jan. 1989.

[10]Lallman, F. J., "Relative Control Effectiveness Technique With Application to Airplane Control Coordination," NASA TP 2416, April 1985.

[11]Lallman, F. J., "Preliminary Design Study of a Lateral-Directional Control System Using Thrust Vectoring," NASA TM 86425, Nov. 1985.

[12]Davidson, J. B., et al., "Development of a Control Law Design Process Utilizing Advanced Synthesis Methods With Application to the NASA F-18 HARV," High-Angle-of-Attack Projects and Technology Conf., NASA Dryden Flight Research Facility, April 21–23, 1992.

[13]Bugajski, D., Enns, D., and Hendrick, R., "Nonlinear Control Law Design for High Angle-of-Attack," High-Angle-of-Attack Projects and Technology Conf., NASA Dryden Flight Research Facility, April 21–23, 1992.