Development of a Versatile Rotation Transformation Algorithm for Automatic Model Attitude Positioning

E. R. Heim* and R. W. Hobbs†
Calspan Corporation/AEDC Operations, Arnold Air Force Base, Tennessee

About 1970, some of the facilities at the Arnold Engineering Development Center began to position wind tunnel models automatically using computer control. The equations for determining model position were expressed explicitly for each of the numerous support systems in use, and programmed as individual sets of algebraic equations. Any nonstandard support system required modification of a standard control algorithm, minor compromises to the standard definitions of the model attitudes, or sometimes both at a considerable expenditure of effort. A recently developed generalized algorithm has replaced several of the original support-system-dedicated algorithms. The applicability of the new algorithm to any one-, two-, or three-degree-of-freedom support system has substantially reduced the effort required to tailor the model control and data reduction codes for each test entry. In addition to its primary advantage of versatility, the algorithm is also more precise in determining the model attitude. The primary focus of this paper is presentation of the details essential to development of a control algorithm, which is applicable to support systems with up to three degrees of freedom. A list of advantages offered by the algorithm is also included.

Nomenclature

$[A]$, $[E]$	= general matrices
a	= element of $[A]$
$[B]$	= 3 × 3 rotation matrix of all support system angles between $\measuredangle 1$ and $\measuredangle 2$
$[C]$	= 3 × 3 rotation matrix of all support system angles between $\measuredangle 2$ and $\measuredangle 3$
e	= element of $[E]$
$[F]$	= 3 × 3 rotation matrix of specified model attitude angles
$[I]$	= 3 × 3 rotation matrix of $\measuredangle 1$ functions
Ik	= row and/or column subscript of $[I]$
i	= row location subscript
$[J]$	= 3 × 3 rotation matrix of $\measuredangle 2$ functions
Jk	= row and/or column subscript of $[J]$
j	= column location subscript
$[K]$	= 3 × 3 rotation matrix of $\measuredangle 3$ functions
Kk	= row and/or column subscript of $[K]$
k	= subscript, $k = 1 - 3$
M	= matrix dimension, $M = 3$
N	= integer-dependent variable
T	= transpose of the matrix
$\measuredangle 1$, $\measuredangle 2$, $\measuredangle 3$	= support mechanism angles 1, 2, and 3, respectively
$[\alpha]$	= model angle-of-attack rotation matrix
$[\alpha s]$	= support system pitch rotation matrix
$[\Phi]$	= model bank angle rotation matrix
$[\Phi s]$	= support system roll rotation matrix
$[\Psi]$	= model yaw angle rotation matrix
$[\Psi s]$	= support system yaw rotation matrix

Introduction

FOR 16 years, the flight dynamics testing effort at Arnold Engineering Development Center (AEDC) has made use of automatic model attitude positioning systems (AMAPS). One of the first algorithms was developed for a two-degree-of-freedom mechanical system employing a pitch sector and a roll head, as illustrated in Fig. 1. The assumption of a straight sting/balance/model arrangement allowed development of a simple iterative AMAPS algorithm that was convergent even for bent-sting arrangements. The associated model attitude equations used in the data reduction were inflexible in that model support system misalignment angles and mechanical system control angles were specified in number, order, and identity. As a consequence, any variation (other than in the value of the prescribed rotation angles) in the model support system required rather involved test-peculiar changes to be made in the data reduction program on a case-by-case basis. To concisely restate the original capability for automatic model attitude control: The AMAPS algorithm and the associated model attitude equations applied specifically to a two-degree-of-freedom pitch-roll mechanical control sequence in which support system angles were fixed in every respect except value. Each different support system, for instance a pitch-yaw system as represented by the captive trajectory support (CTS) system in Fig. 2, required a different positioning algorithm and a different set of model attitude equations.

As test requirements grew more complex, external store and control surface loads data requirements mandated calculation of control surface and store attitude angles relative to the gravity vector and sometimes to the freestream velocity vector as well. These requirements demanded generation of test-peculiar equations for essentially every test involving auxiliary balances. At the same time, escalating manhour, material, and power costs directed attention toward changes that could improve the cost effectiveness of testing. By the early 1980's, model attitude accuracy requirements were sufficiently strin-

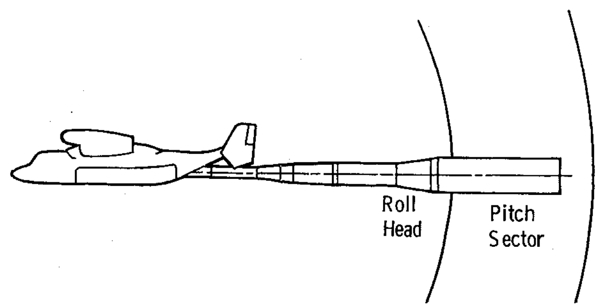

Fig. 1 Pitch-roll support system.

Presented as Paper 88-0056 at the AIAA 26th Aerospace Sciences Meeting, Reno, NV, Jan. 11-14, 1988; received April 16, 1988; revision received Oct. 4, 1988. This paper is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

*Principal Engineer.
†Senior Engineer.

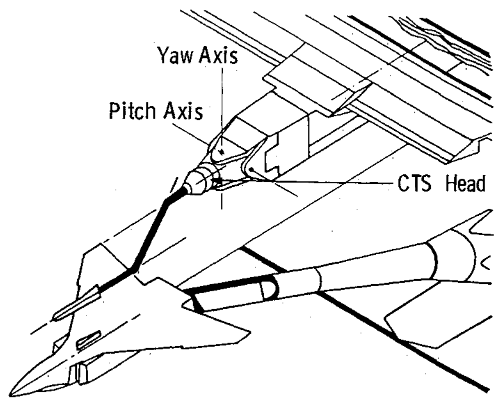

Fig. 2 Pitch and pitch-yaw support systems.

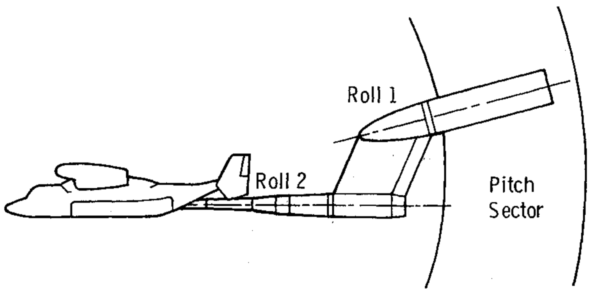

Fig. 3 Pitch-roll-roll support system.

gent so that flow angularities of a few tenths of a degree were no longer regarded as negligible, and test articles themselves began to tax the capabilities of existing support hardware. For instance, the data accuracy requirements and model support hardware capability for a particular test entry dictated that the test be conducted only at wings-level pitch and yaw attitudes. The installation used was a pitch-roll-roll system with a prebend between the two roll mechanisms, as shown in Fig. 3. The support system, known as the High Angle Automated Sting (HAAS) support system, was designed and built to provide the required wings-level pitch and/or yaw-attitude positioning capability, thus essentially eliminating any effects caused by spatial variations in flow angularity. It was the HAAS system that added the final incentive to develop a generalized three-degree-of-freedom automatic model attitude positioning algorithm.

The bases of the AMAPS II algorithm are closed form solutions to a set of three-degree-of-freedom equations. Although the solutions are closed form, the algorithm itself is still iterative in the sense that the support system deflections under load are dependent on the model attitude at the time the loads are determined.

The AMAPS II control algorithm is primarily a generalization of the original model attitude calculation equations. The code allows the computer to create the proper terms for each individual rotation matrix, construct the matrix equations, and reduce the equations using a matrix multiplication subroutine. Fixed-form solutions for the mechanism angles, which establish the required model attitude, are the most unique features of the algorithm. The solutions are versatile in that they apply to any model support system and accept requested model attitude angles in any of the axis systems in use, such as stability axes, missile or nonrolling body axes, aeroballistic axes, and a modified aeroballistic axis system called aero-axes, which combines useful features of both missile and aeroballistic axes.

Complete details of the algorithm are beyond the scope of this paper. Instead, the aspects essential to development of the algorithm are presented.

Basic Considerations

Since aerodynamic coefficients depend on two independent model attitude angles, wind tunnel model support systems usually require only two degrees of freedom in model positioning. The application of a three-degree-of-freedom system of equations to obtain solutions to one- or two-degree-of-freedom problems appears to complicate the task needlessly; however, the added complication is easily addressed in one- and two-degree-of-freedom cases, and the additional degree of freedom allows tailoring the math model to fit any one-, two-, or three-degree-of-freedom support system. The third degree of freedom simply incorporates the fact that, in addition to its aerodynamic attitude, the model has some spatial attitude with respect to a gravity axis system.

The order of rotation in a given reference axis system allows any model attitude angle to be changed without affecting any prior rotation angle. Thus, the stability axis system reference angle rotation order of yaw before pitch in two degrees of freedom becomes a roll-yaw-pitch order in three degrees of freedom. Likewise, the aero-axis and aeroballistic system rotation sequence of pitch before roll becomes a roll-pitch-roll sequence.

Three-degree-of-freedom model attitude angles allow the set of equations for a one- or two-degree-of-freedom support system to be manipulated into a three-degree-of-freedom form by "transposing" the unspecified model attitude angles. Consider a two-degree-of-freedom pitch-roll support system in positioning an aircraft model. The matrix equation is[1]:

$$[\Phi s]\ [\alpha s] = [\alpha]\ [\Psi]\ [\Phi]$$

Since only two angles can be set, only two angles can be requested (specified). The third (unspecified) model attitude angle is determined by the two specified angles and the two-degree-of-freedom support system. The three-degree-of-freedom form of the equation then is:

$$[\Phi s]\ [\alpha s]\ [\Phi]^{-1} = [\alpha]\ [\Psi]\ [\Phi]\ [\Phi]^{-1} = [\alpha]\ [\Psi]$$

A single degree-of-freedom support system equation is similarly manipulated:

$$[\alpha s] = [\alpha]\ [\Psi]\ [\Phi]$$

$$[\alpha s]\ [\Phi]^{-1}\ [\Psi]^{-1} = [\alpha]\ [\Psi]\ [\Phi]\ [\Phi]^{-1}[\Psi]^{-1} = [\alpha]$$

The required general equation needed to allow for misalignments between support system mechanisms as well as to make the equation applicable to other support system and attitude angle matrix arrays is:

$$[\measuredangle 3]\ [C]\ [\measuredangle 2]\ [B]\ [\measuredangle 1] = [F] \tag{1}$$

In Eq. (1), C and B are matrices that result when all of the rotation angle matrices between two variable angles in a support system have been properly expanded. Since any one-, two-, or three-degree-of-freedom system can be manipulated to fit the form of the above general equation, the equations as derived subsequently are applicable to any system.

Algorithm Development

Development of the basic AMAPS algorithm rests on the following facts:

1) Three degrees of freedom are necessary and sufficient to describe any spatial angular orientation.

2) Any system with lesser or greater degrees of freedom must be made to simulate a three-degree-of-freedom system through matrix manipulations involving the inverses of any unspecified model attitudes and/or by restricting the additional degrees of freedom.

3) $a_{ij} = e_{ij}$.

4) $[A]^{-1} = [A]^T = [-A]$ for orthogonal vectors.

5) In the expansion of any three-degree-of-freedom series

of orthogonal matrices, there will be one element involving only one unknown and, in that element's row (and column), two other elements that are functions of that unknown and one of the other two unknowns.

The last fact indicates the most direct solution technique for the system of equations. The location (row and column) of the simplest element in the expanded (3 × 3) matrix identifies the positions of all elements required to completely solve the system of equations. Once the simplest element is equated with its counterpart element from the right-hand known element matrix and the equation solved for the first unknown, similar equations involving another element from the same row (and column) will yield the other two unknown angles. The only prerequisite for developing an efficient code is a scheme for generalizing the element identifiers so that the resulting expressions for each individual element in the expanded 3 × 3 matrix are independent of the type and order of the support system angles.

The insight needed to develop the subscripting rationale may be attained by considering the types of matrices and pattern of the elements within the types of matrices involved:

$$\text{Yaw, type 3, } a_{33} = 1 \qquad \begin{bmatrix} \cos & \sin & 0 \\ -\sin & \cos & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\text{Roll, type 1, } a_{11} = 1 \qquad \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos & \sin \\ 0 & -\sin & \cos \end{bmatrix}$$

$$\text{Pitch, type 2, } a_{22} = 1 \qquad \begin{bmatrix} \cos & 0 & -\sin \\ 0 & 1 & 0 \\ \sin & 0 & \cos \end{bmatrix}$$

(Note: Arguments purposely omitted.)

Clearly, the three rotation matrices have the same elements (disregarding the arguments). Only the location of a given element within each matrix differs. Furthermore, the position of each specific element moves down one row and right one column as the matrix "type" changes successively from 1 to 2, 2 and 3, or 3 to 1. Because of these relationships among the matrix types, the following generalization of subscripts can be made:

$J1$ = "Type" Identifies the element "1" row and column

$J2$ = ? Identifies the subsequent ("type" + 1) row and column

$J3$ = ?? Identifies the subsequent ("type" + 2) row and column

If the rotation matrix being considered is a yaw (type 3) matrix, for example, then $J1 = 3$, $J2 = 1$, and $J3 = 3$. General expressions for $J2$ and $J3$ are required, however, so that the correct sequential values result for $J2$ and $J3$ for each matrix type. The general expressions needed for $J2$ and $J3$ are provided in the AMAPS II algorithm using the MOD (N,M) function: MOD $(N,M) = N - [\mathrm{INT}(N/M)^*M]$.

This function returns the value of N for $N = 1$ or 2, or 0 for $N = 3$. The required expressions for $J2$ and $J3$ are:

$$J2 = \mathrm{MOD}\ (\text{type},3) + 1$$

$$J3 = \mathrm{MOD}\ (\text{type} + 1,3) + 1$$

These expressions give the desired results as shown in Table 1.

Every particular element in a given matrix can now be denoted by a unique generalized subscript, which, as Table 2 shows, correctly specifies both the form (or value) of the element and its location in each matrix type; e.g., $J1$, $J1$ identifies the value 1 located in the proper position on the diagonal of each matrix type.

Table 1 Generalized subscript development

Row/column	$J1$	$J2$	$J3$
Equation	Type	MOD(type,3) + 1	MOD(type + 1,3) + 1
Roll matrix	1	2	3
Pitch matrix	2	3	1
Yaw matrix	3	1	2

$\mathrm{MOD}(N,M) = N - \mathrm{INT}(N/M)^*M$

Table 2 Generalized subscript element values and locations

General subscript	Element value/form	Element location Type 1	Type 2	Type 3
$J1,J1$	1	1,1	2,2	3,3
$J1,J2$	0	1,2	2,3	3,1
$J1,J3$	0	1,3	2,1	3,2
$J2,J1$	0	2,1	3,2	1,3
$J2,J2$	cos	2,2	3,3	1,1
$J2,J3$	sin	2,3	3,1	1,2
$J3,J1$	0	3,1	1,2	2,3
$J3,J2$	−sin	3,2	1,3	2,1
$J3,J3$	cos	3,3	1,1	2,2

The aforementioned subscripting technique must be applied to each of the matrices in Eq. (1). This is done by letting J denote the ∡ 2 matrix, I denote the ∡ 1 matrix, and K denote the ∡ 3 matrix [see Eq. (2)]. The intervening matrix (B and C) element subscripts are then determined by appropriately applying the preceding matrix row subscripts and the succeeding matrix column subscripts as shown in Eq. (2). Finally, the known model attitude matrix (F) element subscripts are determined by applying the K matrix row subscripts and the I matrix column subscripts appropriately.

$$\begin{matrix} K1 & K2 & K3 \end{matrix}$$
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\measuredangle 3 & \sin\measuredangle 3 \\ 0 & -\sin\measuredangle 3 & \cos\measuredangle 3 \end{bmatrix} \begin{bmatrix} C_{K1,J2} & C_{K1,J3} & C_{K1,J1} \\ C_{K2,J2} & C_{K2,J3} & C_{K2,J1} \\ C_{K3,J2} & C_{K3,J3} & C_{K3,J1} \end{bmatrix}$$

$$\begin{matrix} J2 & J3 & J1 \end{matrix}$$
$$\begin{bmatrix} \cos\measuredangle 2 & \sin\measuredangle 2 & 0 \\ -\sin\measuredangle 2 & \cos\measuredangle 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} B_{J2,I3} & B_{J2,I1} & B_{J2,I2} \\ B_{J3,I3} & B_{J3,I1} & B_{J3,I2} \\ B_{J1,I3} & B_{J1,I1} & B_{J1,I2} \end{bmatrix}$$

$$\begin{matrix} I3 & I1 & I2 \end{matrix}$$
$$\begin{bmatrix} \cos\measuredangle 1 & 0 & -\sin\measuredangle 1 \\ 0 & 1 & 0 \\ \sin\measuredangle 1 & 0 & \cos\measuredangle 1 \end{bmatrix} = \begin{bmatrix} F_{K1,I3} & F_{K1,I1} & F_{K1,I2} \\ F_{K2,I3} & F_{K2,I1} & F_{K2,I2} \\ F_{K3,I3} & F_{K3,I1} & F_{K3,I2} \end{bmatrix} \qquad (2)$$

The pitch-yaw-roll support system rotation sequence math-modeled in Eq. (2) has no particular significance since the matrix element referencing scheme insures that the same general expressions for the elements in the matrix will result regardless of the constituent matrix types and/or the order of the matrices in the math model of the actual support system. Only the positions of identical elements will differ in the expansions for various systems. The expansion process for the simplest element is shown below along with the resulting general expression required to solve for the unknown angle.

$$\begin{bmatrix} C_{K1,J2} & C_{K1,J3} & C_{K1,J1} \\ \text{------} & \text{------} & \text{------} \\ \text{------} & \text{------} & \text{------} \end{bmatrix} \begin{bmatrix} \cos\measuredangle 2 & \sin\measuredangle 2 & 0 \\ -\sin\measuredangle 2 & \cos\measuredangle 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \text{------} & B_{J2,I1} & \text{------} \\ \text{------} & B_{J3,I1} & \text{------} \\ \text{------} & B_{J1,I1} & \text{------} \end{bmatrix} = \begin{bmatrix} \text{------} & F_{K1,I1} & \text{------} \\ \text{------} & \text{------} & \text{------} \\ \text{------} & \text{------} & \text{------} \end{bmatrix}$$

$$(C_{K1,J2}\, B_{J2,I1} + C_{K1,J3}\, B_{J3,I1}) \cos\measuredangle 2 +$$

$$(C_{K1,J2}\, B_{J3,I1} - C_{K1,J3}\, B_{J2,I1}) \sin\measuredangle 2$$

$$+ C_{K1,J1}\, B_{J1,I1} = F_{K1,I1}$$

$$(X1/X4) \cos\measuredangle 2 + (X2/X4) \sin\measuredangle 2 = 1$$

where

$$X1 = C_{K1,J2}\, B_{J2,I1} + C_{K1,J3}\, B_{J3,I1}$$

$$X2 = C_{K1,J2}\, B_{J3,I1} - C_{K1,J3}\, B_{J2,I1}$$

$$X3 = C_{K1,J1}\, B_{J1,I1}$$

$$X4 = F_{K1,I1} - X3$$

$$\measuredangle 2 = \tan^{-1}(X2/X1) - \cos^{-1}[\pm X4/(X1^2 + X2^2)^{1/2}] \tag{3}$$

Equation (3) was obtained by expanding the matrices using row $K1$ and column $I1$, which contained only 1 and 0 value elements. To obtain a solution for $\measuredangle 1$, row $K1$ and column $I3$ are selected with the following result:

$$X7 \cos\measuredangle 1 + X8 \sin\measuredangle 1 = F_{K1,I3} \tag{4}$$

where

$$X7 = X5\, B_{J3,I3} + X6\, B_{J2,I3} + C_{K1,J1}\, B_{J1,I3}$$

$$X8 = X5\, B_{J3,I2} + X6\, B_{J2,I2} + C_{K1,J1}\, B_{J1,I2}$$

$$X5 = C_{K1,J2} \sin\measuredangle 2 + C_{K1,J3} \cos\measuredangle 2$$

$$X6 = C_{K1,J2} \cos\measuredangle 2 + C_{K1,J3}\, (-\sin\measuredangle 2)$$

Equation (4) may be sufficient to solve for $\measuredangle 1$ by itself; however, a parallel expression can be obtained using column $I2$ (instead of $I3$) for the expansion and the two equations (4) and (5) can then be solved simultaneously:

$$X8 \cos\measuredangle 1 - X7 \sin\measuredangle 1 = F_{K1,I2} \tag{5}$$

$$\measuredangle 1 = \tan^{-1}[(X8\, F_{K1,I3} - X7\, F_{K1,I2})/$$

$$(X8\, F_{K1,I2} + X7\, F_{K1,I3})] \tag{6}$$

Two solutions for $\measuredangle 1$ are obtained by using both values obtained for $\measuredangle 2$.

The elements required to obtain the expression for $\measuredangle 3$ are developed by expanding by the second and third rows ($K2$ and $K3$) of the $\measuredangle 3$ matrix in combination with the middle column ($I1$) of the $\measuredangle 1$ matrix. The solution is then developed parallel with the $\measuredangle 1$ solution development.

$$X9 \cos\measuredangle 3 + X10 \sin\measuredangle 3 = F_{K2,I1}$$

$$X10 \cos\measuredangle 3 - X9 \sin\measuredangle 3 = F_{K3,I1}$$

where

$$X9 = C_{K2,J2}\, (B_{J2,I1} \cos\measuredangle 2 + B_{J3,I1} \sin\measuredangle 2)$$

$$+ C_{K2,J3}\, [B_{J2,I1}(-\sin\measuredangle 2) + B_{J3,I1} \cos\measuredangle 2]$$

$$+ C_{K2,J1}\, B_{J1,I1}$$

$$X10 = C_{K3,J2}\, (B_{J2,I1} \cos\measuredangle 2 + B_{J3,I1} \sin\measuredangle 2)$$

$$+ C_{K3,J3}\, [B_{J2,I1}(-\sin\measuredangle 2) + B_{J3,I1} \cos\measuredangle 2]$$

$$+ C_{K3,J1}\, B_{J1,I1}$$

$$\measuredangle 3 = \tan^{-1}[(X10\, F_{K2,I1} - X9\, F_{K3,I1})/$$

$$(X9\, F_{K2,I1} + X10\, F_{K3,I1})] \tag{7}$$

From input arrays, which include the number of degrees of freedom of the support system, the requested model attitude angles and the type, order and arguments of all known, unknown, specified, and unspecified angle matrices, the individual matrices can be created and manipulated into the proper form according to the rules of matrix algebra. The C, B, and F matrices can then be obtained using a matrix multiplication routine as necessary. All three angles (the unknown mechanism angles and the negatives of any unspecified model attitude angles) are then obtained directly from Eqs. (3), (6), and (7).

Concluding Remarks

Since implementation of AMAPS II in the 16-ft transonic tunnel (Tunnel 16T) in early 1985, AMAPS II has also been implemented in Tunnels 16S and 4T as the facility standard. As a result, several very unusual three-degree-of-freedom support systems have been controlled without problems, and at least twice early evaluation of a proposed support system using the AMAPS II algorithm has exposed angle limitations in the support system. The following list is a comprehensive summary of the advantages offered by AMAPS II:

1) It is applicable to one-, two-, and three- (or higher order systems when reducible to three) degree-of-freedom systems.

2) Code modifications are not required for any new model support mechanism.

3) It can accommodate any changes in a support system through input changes.

4) It will accept requested angles in any axis system.

5) It can be used to position any model or model component (external store, control surface, etc.) with respect to the freestream or gravity vector.

6) With three-degree-of-freedom support systems, the model or control surface can be positioned as desired with respect to both the gravity and freestream vectors simultaneously (e.g., constant angle-of-attack yaw polars with wings level).

7) It provides noniterative solutions that account for every support system angle that can be quantified.

8) It can be used to evaluate existing or proposed support system mechanical limits with respect to model attitude requirements and vice versa.

9) It offers two solutions ("upright" and "inverted") for stability, aeroballistic, and aeroaxis model attitude angle inputs to two- and three-degree-of-freedom systems.

10) It is potentially faster than simple, strictly iterative algorithms.

11) Part of the algorithm can be used in the data reduction program to calculate the actual positions of external stores, control surfaces, or other model components relative to the freestream velocity vector and/or the gravity vector.

It cannot be said that AMAPS II is itself faster than the simpler iterative algorithms of the past; however, it is certainly as efficient, and it is considerably more accurate in its solutions. The versatility of the code (more specifically, the attendant reduction in requirements for test-peculiar code generation and checkout) and its inherent accuracy in solutions are the chief advantages of AMAPS II. Although the potential for both major (equation) and minor errors appears to be increased as a result of the algorithm's versatility, standard checkout procedures are sufficient to recognize the presence of an error and subsequently to identify it. It must be noted, however, that application of the algorithm to new support systems requires a thorough understanding of the support system, matrix math modeling, and model attitude angles and their associated axis systems.

Acknowledgments

The research reported herein was performed by the Arnold Engineering Development Center (AEDC), Air Force Systems Command. Work and analysis for this research were done by personnel of Calspan Corporation/AEDC Operations, operating contractor for the AEDC aerospace flight dynamics facilities. Further reproduction is authorized to satisfy needs of the U. S. Government.

Reference

[1]Trimmer, L. L. and Clark, E. L., "Transformation of Axes Systems by Matrix Methods and Application to Wind Tunnel Data Reduction," Arnold Engineering Development Center, AEDC-TDR-63-224 (AD-422287), Oct. 1963.