

NASA
Reference
Publication
1252

March 1991

AMES

IN-05

1723

8134

State Estimation Applications in Aircraft Flight-Data Analysis

A User's Manual for SMACK

Ralph E. Bach, Jr.

ANALYTIC: A USER'S MANUAL FOR SMACK (NASA)

CSCL 010

41/05

Unclass
0001723

N

[The left side of the page is heavily obscured by dense, illegible black noise and artifacts, likely from a scanning error or a corrupted document. No text is legible in this region.]

[The right side of the page is mostly blank white space, with a few scattered, faint black specks and a vertical line of noise along the far right edge, possibly representing the gutter of a scanned book or a scanning artifact.]

**NASA
Reference
Publication
1252**

1991

State Estimation Applications in Aircraft Flight-Data Analysis

A User's Manual for SMACK

Ralph E. Bach, Jr.
*Ames Research Center
Moffett Field, California*



National Aeronautics and
Space Administration
Office of Management
Scientific and Technical
Information Division

SUMMARY

This report traces the evolution in the use of state estimation for the analysis of aircraft flight data. A unifying mathematical framework for state estimation is reviewed, and several examples are presented that illustrate a general approach for checking instrument accuracy and data consistency, and for estimating variables that are difficult to measure. Recent applications associated with research-aircraft flight tests and airline turbulence upsets are described. A computer program for aircraft state estimation is discussed in some detail. This document is intended to serve as a user's manual for the program, called SMACK (SMoothing for AirCRAFT Kinematics). The diversity of the applications described in the report emphasizes the potential advantages in using SMACK for flight-data analysis.

1. INTRODUCTION

Accurate determination of aircraft motions from noisy or incomplete measurements is an important problem in the analysis of flight-test experiments. The measurements often may contain significant errors which must be identified before the data are used in any performance or stability-and-control calculations. Furthermore, direct measurements of certain important dynamic variables may be unreliable or impractical to perform. A similar problem occurs in the analysis of aircraft accidents, where the actual motions may have to be determined from a very limited data set. These problems are being solved by the analytical method known as state estimation. This report presents an algorithmic approach for aircraft state estimation, demonstrates its application for solving several example problems, and describes the computer program used to obtain the solutions.

The first application of state estimation to postflight data analysis can probably be attributed to the pioneering work of Otto Gerlach in the 1960s at the Delft Technological University, The Netherlands. This early contribution (refs. 1 and 2), called "flightpath reconstruction," was primarily concerned with the accurate determination of angle of attack, pitch angle, and vehicle velocity during dynamic maneuvers. These "states" were obtained by integrating functions of measurements from the pitch-rate gyroscope and normal and longitudinal accelerometers. Initial conditions and bias terms were determined from airspeed and altitude measurements at steady-state end points of the maneuver. The resulting "smoothed" time-histories were then used as a basis for subsequent parameter identification studies.

Application of the state-estimation method to aircraft problems is possible because the forces and resulting motions of an aircraft along a flightpath are related by well-known equations of motion. The equations may be used to produce estimates of force and motion variables that are compared with corresponding measurement time-histories in an iterative procedure until a suitable "match" is obtained. As Gerlach has pointed out, the technique of state estimation provides a check on instrument accuracy and data consistency as well as estimates of unmeasured or poorly measured variables.

These items have been the primary objectives in most of the studies that followed the initial work of Gerlach. His students later improved and formalized the techniques that Gerlach had developed (refs. 3 and 4). In this country, early advocates of the use of state estimation for flightpath reconstruction were Wingrove (refs. 5 and 6) at NASA Ames, Eulrich and Weingarten (ref. 7) at Calspan, and Molusis (ref. 8) at Sikorsky Aircraft. Over the past few years, the work in this field has been evolving consistently toward the development of more sophisticated algorithms, the use of more complete kinematic models, and the treatment of more difficult applications.

Most recently-developed algorithms (refs. 9-18) utilize a version of an extended Kalman filter. Although good results have been reported in offline flight-data-processing applications, such algorithms are not optimum in their use of future as well as past data in the measurement record. The algorithm advocated here for aircraft state-estimation is based on a variational solution of a nonlinear, fixed-interval smoothing problem. It is iterative in nature, providing improved state estimates until a minimum squared-error measure is achieved. Linearization is about a nominal trajectory and convergence is quadratic. It is based on the "successive sweep" algorithm of McReynolds and Bryson (ref. 19), originally devised to solve an optimal control problem. This algorithm, however, had not been applied to determine aircraft motions along a flight trajectory until the development of the state estimation program described in this report. This program, SMOOTHING for AIRCRAFT Kinematics, called SMACK, has been implemented at NASA Ames for use in flight-test analyses and to assist the National Transportation Safety Board in its investigation of aircraft accidents.

The main purpose of this report is to provide the information necessary for effective use of the SMACK program. The text portion, indicated by chapter headings, includes discussions of the underlying mathematics, typical applications, and coding rules for problem setup. A set of appendices supplies the detailed information needed for installation and testing of the program. The main text is organized as follows. Chapter 2 outlines two mathematically equivalent algorithms for solution of the smoothing problem, which have been published previously (ref. 20). The first algorithm, which consists of a forward covariance filter and backward smoother, facilitates comparison of the variational and extended Kalman filter methods. The second algorithm, which consists of a backward information filter and forward smoother, is shown to have certain computational advantages. It is this second algorithm that forms the basis of SMACK. Chapter 3 then describes the six-degree-of-freedom aircraft state and measurement models used in the program. The application examples presented in chapter 4 should help the analyst recognize the potential advantages in using state estimation. Chapter 5 covers the preparation of a coding list for problem solution by SMACK. Sample lists are given for the application examples presented in chapter 4. The use of SMACK as part of an overall flight-test methodology is illustrated in chapter 6 and its application in the analysis of windshear accidents is the subject of chapter 7.

For installation and testing of the SMACK program, the user should refer to the appendices: appendix A discusses subroutine hierarchy and shows key block diagrams, which should help the user to understand program concept and flow. Appendix B covers some important aspects of program implementation, such as COMMON structure and memory requirements. Two test problems, using simulated flight data (internally generated) are presented in Appendices C and D: the first represents a typical flight-test analysis; the second, a typical accident analysis. These problems illustrate the essential features of a solution by SMACK, and provide reference output listings for testing a new installation of the program. A final appendix, appendix E, describes the preparation of a user subroutine for processing a set of flight data with SMACK for analysis in the manner discussed in this report.

The author thanks students Hieu Truong, Hien Tran, Omid and Vafa Kordestani, and Duc Tran of San Jose State University for their able assistance in the development and maintenance of the SMACK program.

2. STATE-ESTIMATION ALGORITHMS

In this report, the term "state estimation" refers to offline processing of a set of independent measurements from a given physical system. Each record in the set consists of a time history that covers the same fixed interval, but does not necessarily share a common sampling rate with other records in the set. If the relationships among the dynamic variables of the system are well understood mathematically, all the measurement records may be processed together in an "optimal" way. In this case, the objective is to determine a set of initial conditions and forcing functions that will cause the output of a mathematical model to "match" the measurement time histories, usually in a least squared-error sense. The task of postflight state estimation is known as a fixed-interval smoothing problem. With its solution, the analyst can determine that the measurements are consistent, and in addition, can obtain estimates of unmeasured variables, as well as instrument bias errors and scale factors.

The state-estimation process solves a state model

$$x(i + 1) = f[x(i),w(i)] ; \quad x(0) = x_0 \quad (2.1)$$

such that $y(i + 1)$ in the measurement model

$$z(i + 1) = y(i + 1) + v(i + 1) ; \quad y(i + 1) = h[x(i + 1)] \quad (2.2)$$

acceptably matches the data record over a time interval ($i = 0, 1, \dots, N - 1$), usually in a least-squared error or minimum variance sense. In equation (2.1), $x(i)$ is an NX -element state vector and $w(i)$ is an NW -element forcing-function vector. In equation (2.2), $z(i + 1)$ and $v(i + 1)$ are NV -element vectors representing the measurements and corresponding (random) measurement errors; $y(i + 1)$ is the system output vector. Note that the output vector is generally a nonlinear function of the state variables.

For aircraft applications, the state and measurement models together represent a finite-difference approximation for the six-degree-of-freedom dynamics of a rigid body. As described in the next chapter, the models are used to generate time histories which are likely to be found in a flight-test measurement set. These include onboard variables such as Euler angles, angular rates, and linear accelerations, as well as tracking variables such as slant range, bearing, and elevation. Any bias errors or scale factors associated with the state or measurement models are appended to the state vector and treated as constant but unknown parameters.

The solution of the fixed-interval smoothing problem consists of determining the initial condition x_0 and forcing function $\{w(i)\}$ that, subject to the dynamic constraint of equation (2.1), minimize the following squared-error performance measure:

$$J = (x_0 - \bar{x}_0)^t P_0^{-1} (x_0 - \bar{x}_0) / 2 + \sum_{i=0}^{N-1} \{ [z(i+1) - y(i+1)]^t R^{-1} [z(i+1) - y(i+1)] + w^t(i) Q^{-1} w(i) \} / 2 \quad (2.3)$$

In equation (2.3), \bar{x}_0 is an a priori estimate of x_0 , and P_0 , Q , and R are weighting matrices. Sage and Melsa (ref. 21) discuss a maximum-likelihood interpretation of the performance measure in which P_0 is the error-covariance matrix for the a priori estimate, and Q and R are error-covariance matrices for the input and output sequences, respectively (assumed to be stationary). Note that the first term of equation (2.3) serves as a "penalty" function and tends to bias the estimate of x_0 toward its a priori value.

The fixed-interval smoothing problem is solved using a method of successive approximations based on expansion of the performance measure (eq. (2.3)) to the second order and of the dynamic constraint (eq. (2.1)) to the first order. Suppose we choose x_0 , $\{w(i)\}$ and obtain a nominal trajectory by solving equation (2.1). It is unlikely that our solution minimizes J , but we shall try to determine a neighboring solution that yields a smaller value. To do this, we first express a variation in the performance measure in terms of variations δx_0 and $\{\delta w(i)\}$:

$$\begin{aligned} \delta J = & (x_0 - \bar{x}_0)^t P_0^{-1} \delta x_0 + (1/2) \delta x_0^t P_0^{-1} \delta x_0 + \sum_{i=0}^{N-1} \{ w^t(i) Q^{-1} \delta w(i) + (1/2) \delta w^t(i) Q^{-1} \\ & \times \delta w(i) - [z(i+1) - y(i+1)]^t R^{-1} h_x \delta x(i+1) + (1/2) \delta x^t(i+1) h_x^t R^{-1} h_x \delta x \\ & \times (i+1) \} \end{aligned} \quad (2.4)$$

Next, we assume that deviation from the nominal trajectory will be governed by

$$\delta x(i+1) = f_x \delta x(i) + f_w \delta w(i) \quad , \quad \delta x(0) = \delta x_0 \quad (2.5)$$

In equations (2.4) and (2.5), the Jacobian matrices (partial derivatives) are defined as

$$f_x = \partial f[x(i), w(i)] / \partial x(i) \quad ; \quad f_w = \partial f[x(i), w(i)] / \partial w(i) \quad ; \quad h_x = \partial h[x(i+1)] / \partial x(i+1) \quad (2.6)$$

and are to be evaluated along the nominal trajectory.

Our objective now is to specify δx_0 and $\{\delta w(i)\}$ such that δJ has the most negative value possible, subject to the dynamic constraint of equation (2.5). We solve this "accessory minimization" problem in the usual way (ref. 22) by adjoining the constraint to equation (2.4) using a Lagrange multiplier. Hence,

$$\delta \bar{J} = \delta J + \sum_{i=0}^{N-1} \lambda^t(i+1) [f_x^t \delta x(i) + f_w^t \delta w(i) - \delta x(i+1)] \quad (2.7)$$

The necessary conditions for minimizing J lead to a linear, two-point boundary-value problem (LTPBVP) given by equation (2.5) and

$$\begin{aligned} \lambda(i) &= f_x^t \{ \lambda(i+1) + h_x^t R^{-1} h_x \delta x(i+1) - h_x^t R^{-1} [z(i+1) - y(i+1)] \} \\ \lambda(0) &= -P_0^{-1} [x_0 - \bar{x}_0] + \delta x_0 ; \quad \lambda(N) = 0 \end{aligned} \quad (2.8)$$

$$\delta w(i) = -w(i) - Q f_w^t f_x^{-t} \lambda(i) \quad (2.9)$$

This LTPBVP has an exact solution (ref. 22). Hence, it is possible to determine δx_0 and $\{\delta w(i)\}$, recompute the nominal trajectory with

$$x_0 \leftarrow x_0 + \delta x_0 ; \quad w(i) \leftarrow w(i) + \delta w(i)$$

and evaluate the performance measure, iterating until J is minimized. The change in J that should be realized at any iteration is found by substituting equations (2.8) and (2.9) into equation (2.7).

$$\delta J = -\delta x_0^t P_0^{-1} \delta x_0 / 2 - \sum_{i=0}^{N-1} [\delta w^t(i) Q^{-1} \delta w(i) + \delta x^t(i+1) h_x^t R^{-1} h_x \delta x(i+1)] / 2 \quad (2.10)$$

Two equivalent sweep solutions of the LTPBVP are given here. The first is derived by introducing a vector $\delta \hat{x}(i)$ and matrix $P(i)$ and letting

$$\delta x(i) = \delta \hat{x}(i) - P(i) \lambda(i) \quad (2.11)$$

Notice that the boundary conditions of equation (2.8) require that

$$\delta \hat{x}(0) = \bar{x}_0 - x_0 ; \quad P(0) = P_0 ; \quad \delta x(N) = \delta \hat{x}(N)$$

Straightforward algebraic manipulation yields the algorithm outlined in table 2.1(a), which is essentially the procedure proposed by Cox in 1965 (ref. 23). It consists of a forward covariance filter and a backward smoother, a form that invites comparison with the extended Kalman filter often employed for nonlinear state and parameter estimation (ref. 24). We observe that, for a class of systems with a state model that is linear in its forcing function, as

$$f[x(i), w(i)] = g[x(i)] + Gw(i)$$

TABLE 2.1.- EQUIVALENT ALGORITHMS FOR SOLUTION OF A NONLINEAR SMOOTHING PROBLEM

With x_0 and $\{w(i)\}$ obtained from the preceding iteration (or an initial guess), compute a nominal trajectory using equations (2.1) and (2.2) and evaluate the performance measure using equation (2.3). Now perform (a) or (b) as follows:

(a) Forward filter/backward smoother	(b) Backward filter/forward smoother
Forward filter with a time update	Backward filter with a measurement update
$\delta \bar{x}(i) = f_x \delta \hat{x}(i-1) - f_w w(i-1)$ $M(i) = f_x P(i-1) f_x^t + f_w Q f_w^t$ $\delta \hat{x}(0) = \bar{x}_0 - x_0 ; \quad P(0) = P_0$	$\beta(i) = \alpha(i) - h_x^t R^{-1} [z(i) - y(i)]$ $S(i) = M(i) + h_x^t R^{-1} h_x$ $\alpha(N) = 0 ; \quad M(N) = 0$
and a measurement update	and a time update
$\delta \hat{x}(i) = \delta \bar{x}(i) + K(i) e(i)$ $P(i) = [I - K(i) h_x] M(i)$	$\alpha(i-1) = f_x^t [\beta(i) - S(i) f_w d(i)]$ $M(i-1) = f_x^t [I - f_w L(i)]^t S(i) f_x$
where	where
$e(i) = [z(i) - y(i)] - h_x \delta \bar{x}(i)$ $K(i) = M(i) h_x^t \bar{R}^{-1}$ $\bar{R} = R + h_x M(i) h_x^t$	$d(i) = \bar{Q} [f_w^t \beta(i) + Q^{-1} w(i-1)]$ $L(i) = \bar{Q} f_w^t S(i)$ $\bar{Q} = [Q^{-1} + f_w^t S(i) f_w]^{-1}$
Backward smoother	Forward smoother
$\beta(i) = [I - K(i) h_x]^t [\lambda(i) - h_x^t R^{-1} e(i)]$ $\lambda(i-1) = f_x^t \beta(i) ; \quad \lambda(N) = 0$	$\delta x(i+1) = f_x \delta x(i) + f_w \delta w(i)$ $\delta x(0) = \delta x_0$
and	where
$\delta w(i-1) = -w(i-1) - Q f_w^t \beta(i)$ $\delta x_0 = \bar{x}_0 - P_0 \lambda(0)$	$\delta x_0 = -[P_0^{-1} + M(0)]^{-1}$ $\quad \times [\alpha(0) + P_0^{-1} (x_0 - \bar{x}_0)]$ $\delta w(i) = -d(i+1) - L(i+1) f_x \delta x(i)$

Update x_0 and $\{w(i)\}$. Loop and iterate until δx_0 and $\{\delta w(i)\}$ are "sufficiently" small and the performance measure is minimized.

the forward covariance filter of table 2.1(a) is identical to an extended Kalman filter linearized about a (prior) nominal solution. The usual linearization, however, is about a current solution. In at least one case (ref. 25), the extended Kalman filter has been coupled with a backward smoother. Such a procedure requires no starting solution but does not iterate to minimize a performance measure, and so provides only an approximate solution of the nonlinear smoothing problem.

A second, more useful sweep solution of the LTPBVP is obtained by introducing a vector $\alpha(i)$ and matrix $M(i)$ and letting

$$\lambda(i) = \alpha(i) + M(i)\delta x(i) \quad (2.12)$$

In this case the boundary conditions of equation (2.8) require that $\alpha(N) = 0$, $M(N) = 0$, and

$$\delta x_0 = -[P_0^{-1} + M(0)]^{-1}[P_0^{-1}(x_0 - \bar{x}_0) + \alpha(0)] \quad (2.13)$$

The resulting algorithm is outlined in table 2.1(b) and consists of a backward information filter and a forward smoother (for offline application). It can be shown that this algorithm is equivalent to the "modified" Newton-Raphson method if there are no unknown forcing functions (ref. 26). Notice that the sequences $\{d(i)\}$, $\{L(i)\}$, computed during the filter pass, are utilized during the smoothing pass. Here the (temporary) storage requirement depends on the dimensions of $w(i)$ and $x(i)$ and, of course, on the length of the data record. This formulation, which has been implemented with SMACK, has the following advantages over the algorithm of table 2.1(a):

1. The a priori covariance is easily specified by setting $P_0^{-1} = 0$ in equation (2.13), which is often a good choice in practice. This is equivalent to "no confidence" in an a priori estimate.

2. Constant elements (bias-error and scale-factor parameters) of the state vector are naturally decoupled from the dynamic states. This feature reduces the computational burden.

In applying either algorithm of table 2.1, the analyst must be careful in choosing starting values for x_0 and $\{w(i)\}$, and in selecting the weighting matrices P_0 , Q , and R . The convergence properties of the algorithm are influenced directly by the nominal solution generated by the initial choice of x_0 and any unmeasured forcing functions. Suitable starting values can be obtained by solving the finite-difference approximation of the state model for the forcing-function sequences by using filtered versions of the measurement records to construct the state estimates. On the other hand, the nature of the solution, once convergence is obtained, depends to a considerable degree on the choice of the weighting matrices P_0 , Q , and R . The effect of P_0 is to bias the estimate x_0 toward the a priori value \bar{x}_0 ; it can usually be ignored when applying the algorithm of table 2.1(b). Reasonable values for the elements of Q and R may be determined as follows: filter each measurement record until the residual sequence appears sufficiently

"white," and use its variance as the appropriate diagonal element of R ; then construct an estimate of each forcing function, and use the mean-square value of the starting sequence $\{w(i)\}$ as the appropriate diagonal element of Q .

It should be observed in passing that, heuristically, smoothing is a process of zero phase-shift filtering in which bandwidth increases as the scale of Q increases and the scale of R decreases. One expects that forcing-function and residual variances will agree with the corresponding elements of Q and R used in obtaining the solution. That solution, however, is not unique. Scaling the elements of Q up and the elements of R down by the same factor (bandwidth increase) will result in a solution having closer fits to the data, but with "noisier" forcing-function estimates. This situation emphasizes the need for the analyst to carefully consider the engineering aspects of the problem.

Lest the potential SMACK user despair over possible pitfalls to be encountered in choosing starting values and selecting weighting matrices, he or she should be assured that the program has been designed to require little user intervention. A subroutine included with SMACK provides the set of initial conditions and forcing functions needed to generate a starting trajectory. This routine also calculates sets of diagonal element values for the Q and R weighting matrices, using the procedure suggested earlier. An outline and block diagram for the starting subroutine are given in appendix A.

The Linear Case

Finally, for the sake of completeness, a formulation of the algorithms of table 2.1 for a linear system is considered that is useful in offline digital filtering applications. For the linear case, the state and measurement models of equations (2.1) and (2.2) become

$$x(i) = Fx(i - 1) + Gw(i - 1) , \quad x(0) = x_0 , \quad z(i) = Hx(i) + v(i) \quad (2.14)$$

where F is an $NX \times NX$ matrix, G an $NX \times NW$ matrix, and H is an $NV \times NX$ matrix. Using the notation expressed in equation (2.6), we obtain for the Jacobians

$$f_x = F ; \quad f_w = G ; \quad h_x = H \quad (2.15)$$

Now consider a simple change of variable

$$\delta x(i) = x(i) - x_n(i) \quad (2.16)$$

where $x_n(i)$ is any (nominal) solution of equation (2.14) and $x(i)$ is to be the solution of the linear fixed-interval smoothing problem. If equations (2.15) and (2.16) are used in the forward-filter, backward-smoother algorithm of table 2.1, along with

$$\delta \bar{x}(i) = \bar{x}(i) - x_n(i) ; \quad \delta \hat{x}(i) = \hat{x}(i) - x_n(i)$$

the algorithm of table 2.2(a) results. In similar fashion, if equations (2.15) and (2.16) are used in the backward-filter, forward-smoother algorithm of table 2.1, along with

$$\alpha(i) + \alpha(i) + M(i)x_n(i) ; \quad \beta(i) + \beta(i) + S(i)x_n(i)$$

the algorithm of table 2.2(b) results.

TABLE 2.2.- EQUIVALENT ALGORITHMS FOR SOLUTION OF A LINEAR SMOOTHING PROBLEM

(a) Forward filter/backward smoother	(b) Backward filter/forward smoother
Forward filter with a time update	Backward filter with a measurement update
$\bar{x}(i) = F\hat{x}(i-1)$	$\beta(i) = \alpha(i) - H^t R^{-1} z(i)$
$M(i) = FP(i-1)F^t + GQG^t$	$S(i) = M(i) + H^t R^{-1} H$
$\hat{x}(0) = \bar{x}_0 ; \quad P(0) = P_0$	$\alpha(N) = 0 ; \quad M(N) = 0$
and a measurement update	and a time update
$\hat{x}(i) = \bar{x}(i) + K(i)e(i)$	$\alpha(i-1) = F^t [\beta(i) - S(i)Gd(i)]$
$P(i) = [I - K(i)H]M(i)$	$M(i-1) = F^t [I - GL(i)]^t S(i)F$
where	where
$e(i) = z(i) - H\bar{x}(i)$	$d(i) = \bar{Q}G^t \beta(i)$
$K(i) = M(i)H^t \bar{R}^{-1}$	$L(i) = \bar{Q}G^t S(i)$
$\bar{R} = R + HM(i)H^t$	$\bar{Q} = [Q^{-1} + G^t S(i)G]^{-1}$
Backward smoother	Forward smoother
$\beta(i) = [I - K(i)H]^t [\lambda(i) - H^t R^{-1} e(i)]$	$x(i+1) = Fx(i) + Gw(i)$
$\lambda(i-1) = F^t \beta(i) ; \quad \lambda(N) = 0$	$x(0) = x_0$
and	where
$w(i-1) = QG^t \beta(i)$	$x_0 = -[P_0^{-1} + M(0)]^{-1} [\alpha(0) - P_0^{-1} \bar{x}_0]$
$x_0 = \bar{x}_0 - P_0 \lambda(0)$	$w(i) = -d(i+1) - L(i+1)Fx(i)$

Each of these algorithms converges in one step: no starting solution is needed. However, the considerations concerning the choice of weighting matrices Q and R , and the a priori estimate \bar{x}_0, P_0 are the same as for the nonlinear case. Hence, the backward-filter, forward-smoother algorithm also has an advantage in solving any linear fixed-interval smoothing problem. That algorithm has been used in the realization of a low-pass filter employed in the SMACK starting procedure. The filter is described in appendix E.

3. STATE AND MEASUREMENT MODELS

In this chapter the mathematical models utilized by SMACK to obtain flight trajectories are defined and discussed. Aircraft motions are assumed to be governed by a six-degree-of-freedom kinematic model, referred to a flat, nonrotating Earth. The usual choice of state variables leads to a formulation in which both state and measurement models are nonlinear (ref. 25). The solution algorithm outlined in the previous chapter must then evaluate both the state and measurement model Jacobian matrices (f_x , f_w , and h_x in eq. (2.6)) along the trajectory. However, if the state variables are properly chosen it is possible to obtain a state model that is linear. All nonlinearities then appear in the measurement model. The advantage of using such a formulation with SMACK is computational: the Jacobian matrix for the state model is constant along any trajectory, a feature that significantly improves the efficiency of the solution algorithm (ref. 27).

For realization of a linear state model, we start with the vehicle attitude, defined by the Euler angles (ϕ, θ, ψ) , and the vehicle position, defined by the Earth-surface coordinates (x, y, h) as state variables. Other state variables consist of time derivatives $(\dot{\phi}, \dot{\theta}, \dot{\psi})$, $(\ddot{\phi}, \ddot{\theta}, \ddot{\psi})$, and $(\dot{x}, \dot{y}, \dot{h})$, $(\ddot{x}, \ddot{y}, \ddot{h})$. When motion of the air mass must be considered, the state model is augmented with wind-velocity states (w_x, w_y, w_h) . The simplicity of this linear state model is clearly evident in the block diagram shown in figure 3.1. There we see that the state model consists of integrator "bundles," a structure known as a Brunovsky canonical form (ref. 28). Note that the forcing functions for this system are (d_x, d_y, d_h) , (d_ℓ, d_m, d_n) and (g_x, g_y, g_h) . In some situations it may be preferable to use a simpler model with forcing functions $(\ddot{x}, \ddot{y}, \ddot{h})$ instead of (d_x, d_y, d_h) , and/or $(\dot{\phi}, \dot{\theta}, \dot{\psi})$ instead of (d_ℓ, d_m, d_n) .

The measurements available in an aircraft state-estimation problem often include tracking data, vehicle attitude, velocities, and accelerations. All nonlinearities associated with aircraft kinematics appear in the measurement model shown on the right side of figure 3.1. For example, the blocks labeled with an \mathcal{L} represent the transformation from Earth-surface axes to vehicle-fixed body axes. Here the body velocities with respect to the air mass (u, v, w) and the body accelerations (a_x, a_y, a_z) are calculated from

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = L \begin{bmatrix} \dot{x} - w_x \\ \dot{y} - w_y \\ \dot{h} - w_h \end{bmatrix} ; \quad \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = L \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{h} + g \end{bmatrix} \quad (3.1)$$

respectively, where the transformation is defined by the direction-cosine matrix

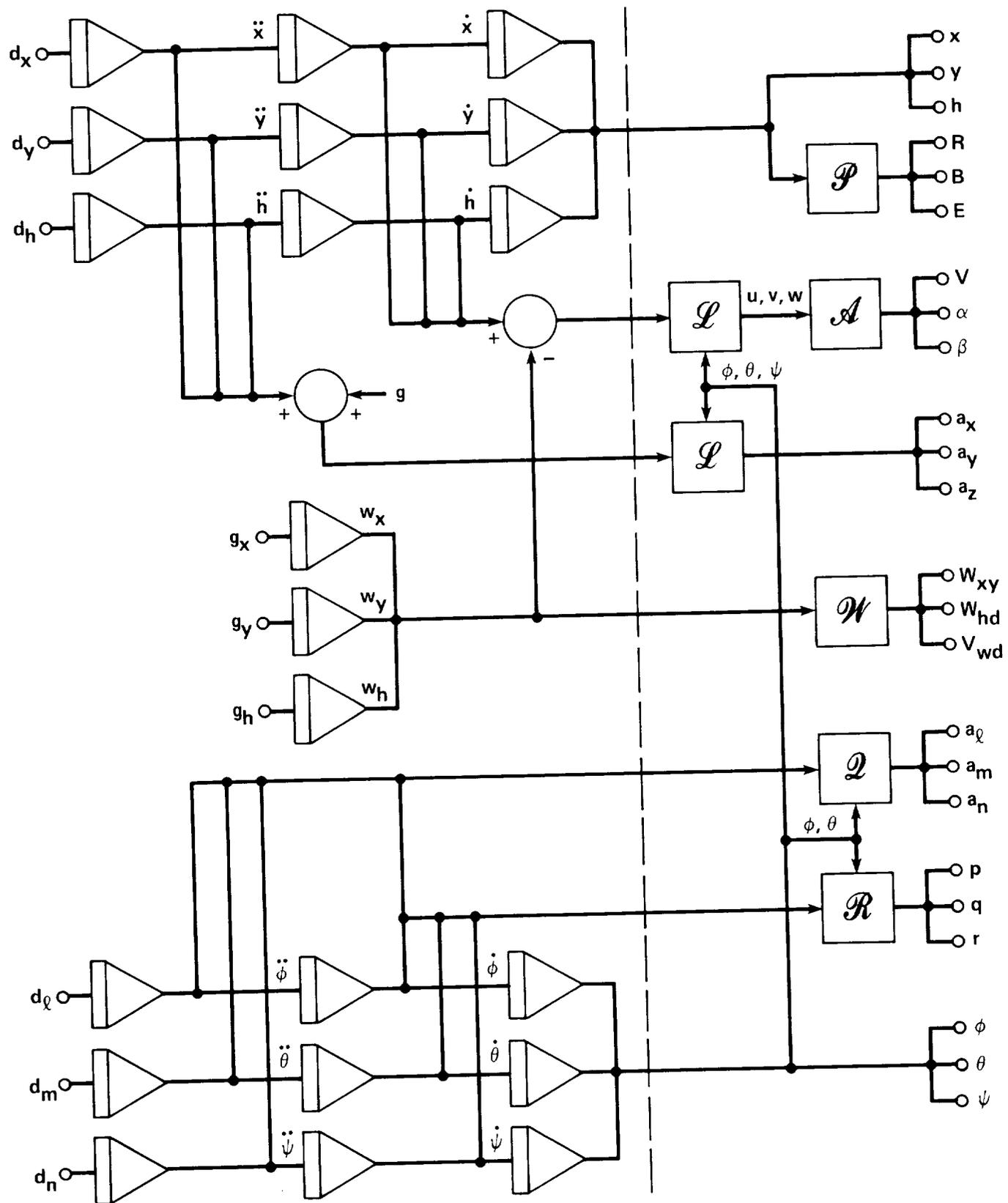


Figure 3.1.- State and measurement models used by SMACK.

$$L = \begin{bmatrix} \cos\theta \cos\psi & \cos\theta \sin\psi & \sin\theta \\ \sin\phi \sin\theta \cos\psi & \sin\phi \sin\theta \sin\psi & -\sin\phi \cos\theta \\ -\cos\phi \sin\psi & +\cos\phi \cos\psi & \\ \cos\phi \sin\theta \cos\psi & \cos\phi \sin\theta \sin\psi & -\cos\phi \cos\theta \\ +\sin\phi \sin\psi & -\sin\phi \cos\psi & \end{bmatrix} \quad (3.2)$$

Radar measurement variables R (slant range), B (bearing angle), and E (elevation angle), which are given by

$$R = [(x - x_r)^2 + (y - y_r)^2 + (h - h_r)^2]^{1/2}$$

$$B = \tan^{-1}[(y - y_r)/(x - x_r)] ; \quad E = \sin^{-1}[(h - h_r)/R] \quad (3.3)$$

where (x_r, y_r, h_r) is the tracking antenna location, are calculated in the block labeled with a \mathcal{P} in figure 3.1. Other inertial data might be supplied by an onboard inertial navigation system (INS). Although not shown in figure 3.1, the measurement model does include the INS velocity variables V_g (groundspeed) and ϕ_g (groundtrack), which are

$$V_g = (\dot{x}^2 + \dot{y}^2)^{1/2} ; \quad \phi_g = \tan^{-1}(\dot{y}/\dot{x}) \quad (3.4)$$

In order to fit air-data records of true airspeed and the flow angles, the measurement model can provide estimates of the aerodynamic variables V (airspeed), α (angle of attack), and β (sideslip angle), from the relations (ref. 29)

$$V = (u^2 + v^2 + w^2)^{1/2} ; \quad \alpha = \tan^{-1}(w/u) ; \quad \beta = \tan^{-1}(v/u) \quad (3.5)$$

These are calculated in the block labeled \mathcal{A} . Notice that the variable β models the vane flow angle, which is not quite the same as the usual sideslip angle (ref. 30). When air-data measurements are included in the SMACK estimation procedure, the winds along the flightpath can also be estimated. The wind variables in the measurement model, W_{xy} and W_{hd} (horizontal magnitude and heading), and V_{wd} (vertical magnitude) are calculated in the block labelled \mathcal{W} from the relations

$$W_{xy} = (w_x^2 + w_y^2)^{1/2} ; \quad W_{hd} = \tan^{-1}(-w_y/-w_x) ; \quad V_{wd} = w_h \quad (3.6)$$

Blocks labelled \mathcal{R} and \mathcal{Q} represent the nonlinear relations that express the body angular velocities (p, q, r) and angular accelerations (a_l, a_m, a_n) in terms of the state variables. The angular velocities are calculated from (ref. 30)

$$p = \dot{\phi} - \dot{\psi} \sin\theta , \quad q = \dot{\theta} \cos\phi + \dot{\psi} \sin\phi \cos\theta , \quad r = -\dot{\theta} \sin\phi + \dot{\psi} \cos\phi \cos\theta \quad (3.7)$$

and the angular accelerations are calculated from

$$\begin{aligned}
a_{\ell} &= \ddot{\phi} - \ddot{\psi} \sin\theta - \dot{\theta}\dot{\psi} \cos\theta \\
a_m &= \ddot{\psi} \sin\phi \cos\theta + \ddot{\theta} \cos\phi - \dot{\theta}\dot{\psi} \sin\phi \sin\theta + \dot{\phi}r \\
a_n &= \ddot{\psi} \cos\phi \cos\theta - \ddot{\theta} \sin\phi - \dot{\theta}\dot{\psi} \cos\phi \sin\theta - \dot{\phi}q
\end{aligned} \tag{3.8}$$

If required, position corrections for location of air-data or accelerometer instruments can be made in the SMACK measurement model. Body-velocity corrections for the air-data system are given by (ref. 31)

$$\Delta u = qz_i - ry_i, \quad \Delta v = rx_i - pz_i, \quad \Delta w = py_i - qx_i \tag{3.9}$$

where (x_i, y_i, z_i) are the body coordinates of the instrument position with respect to the aircraft center of gravity. Airspeed (pitot-static) and aerodynamic angle (vane) corrections are treated separately. Corrections to the body accelerations are (ref. 31)

$$\begin{aligned}
\Delta a_x &= (pq - a_n)y_i + (pr + a_m)z_i - (q^2 + r^2)x_i \\
\Delta a_y &= (pq + a_n)x_i + (qr - a_{\ell})z_i - (p^2 + r^2)y_i \\
\Delta a_z &= (pr - a_m)x_i + (qr + a_{\ell})y_i - (p^2 + q^2)z_i
\end{aligned} \tag{3.10}$$

Note that equations (3.9) and (3.10) refer the corrected variables to the instrument location. Furthermore, equation (3.10) utilizes angular acceleration estimates (a_{ℓ}, a_m, a_n) , which from equation (3.8) are seen to be functions of (ϕ, θ, ψ) . The algorithm of chapter 2 as implemented in SMACK requires all estimates formed in the measurement model to be functions of state variables. In this case, then, the user must specify the forcing functions to be (d_{ℓ}, d_m, d_n) .

A general rule in the application of SMACK is that if there are any elements of sets (a_x, a_y, a_z) or (a_{ℓ}, a_m, a_n) to be estimated, then (d_x, d_y, d_h) or (d_{ℓ}, d_m, d_n) must be specified as forcing functions. Otherwise, the forcing functions may be chosen from the sets (ϕ, θ, ψ) or (x, y, h) . Care should be taken not to mix elements of (d_{ℓ}, d_m, d_n) with elements of (ϕ, θ, ψ) as forcing functions, or, similarly, elements of (d_x, d_y, d_h) with elements of (x, y, h) . It is true, of course, that the more integrators there are between input and output of the state model, the more "smooth" the output will be.

In certain situations, such as performing a preliminary data-consistency check, a user may wish to employ measurements of (a_x, a_y, a_z) and/or (p, q, r) to generate forcing functions. For example, accelerometer measurements can be used to produce the earth-frame accelerations required for the SMACK state model by solving equation (3.1) to obtain

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{h} \end{bmatrix} = L^t \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (3.11)$$

In similar fashion, it can be seen that rate-gyro measurements will provide the required Euler-angle derivatives by solving equation (3.7) to obtain

$$\dot{\psi} = (q \sin\phi + r \cos\phi)/\cos\theta, \quad \dot{\theta} = q \cos\phi - r \sin\phi, \quad \dot{\phi} = p + \dot{\psi} \sin\theta \quad (3.12)$$

Clearly, if either equation (3.11) or (3.12) is employed, the state model will no longer be linear. The tradeoff is, of course, that time-histories for $(\ddot{x}, \ddot{y}, \ddot{h})$ and/or $(\dot{\phi}, \dot{\theta}, \dot{\psi})$ need not be estimated. A separate consideration here is that the use of noise-contaminated measurements to generate forcing functions is likely to bias the solution in an unpredictable way.

4. APPLICATION EXAMPLES

State estimation as a means of checking instrument accuracy and data consistency is now used by many flight-test groups (refs. 9-18). Once a consistent, smoothed set of time histories is obtained from the data, other analyses, such as identification of stability and control derivatives, are readily performed. In fact, relatively simple routines may be used for identification tasks, allowing the analyst freedom to develop a proper aerodynamic model. Since the data-consistency application is extensively treated in chapter 6, and a flight-test example of a complete SMACK solution is given in appendix C, it will not be discussed here. Instead, some of the more recent applications of aircraft state estimation in obtaining estimates of unmeasured or poorly measured variables will be addressed.

In this chapter four examples, based on recently-reported applications of aircraft state estimation, and one example of an application not previously reported, are discussed (ref. 32). The applications, quite diverse in terms of the available measurements and desired estimates, illustrate the wide range of problems that can be treated in a unified way by using SMACK. Data for each example were taken from a simulated trajectory consisting of a rising, coordinated, 180° turn in the presence of wind. The trajectory is generated by a SMACK subroutine for user testing of a problem coding list. Small amounts of random noise, usually 1% or less, were added to each measured variable, and all measurements were recorded once per second. A summary of the available measurements and variables to be estimated for each example is given in table 4.1.

Example 1

For aircraft accident analysis, state estimation can be effectively used to combine data from several sources (e.g., radar site and flight recorder) to determine motions along a trajectory (refs. 33 and 34). In addition, the winds along a flight trajectory can often be estimated. Wind estimation has been used in the analysis of recent airline turbulence upsets, and is the subject of the first example. This application is covered in some detail in chapter 7. Parks et al. (ref. 35) describe the estimation of winds by using data from a DC-10 encounter with severe high-altitude turbulence. The wind estimates from that analysis led Parks to hypothesize the presence of a classical "cat's-eyes" vortex phenomenon in the jet-stream shear layer at the time of the encounter.

Data from a digital flight recorder like the one carried by a DC-10 includes accelerations, Euler angles, altitude, and airspeed, sampled at intervals of 0.25-4.0 sec. Sufficient additional information is available to approximate the records of angles of attack and sideslip. The addition of ground-based air traffic control (ATC) radar provides a number of measurements approaching that available from flight test. To obtain the desired wind estimates, Parks first transformed the accelerations into an Earth frame, then integrated them to obtain aircraft velocity with respect to the Earth. A consistent set of initial conditions and accelerometer

TABLE 4.1.- LIST OF VARIABLES FOR STATE-ESTIMATION EXAMPLES

Variable	Example				
	1	2	3	4	5
Linear Acc. (a_x, a_y, a_z)	Measured	Measured		Measured	Measured
Angular Vel. (p, q, r)		Measured			Measured
Position (R, B, h)	Measured		Measured	Measured	Measured
Winds (W_{xy}, W_{hd}, V_{wd})	Estimated	Measured	Measured	Measured	Measured
Angles (ϕ, θ, ψ)	Measured	Estimated	Measured	Measured	Estimated
Air Data (V, α, β)	Measured	Measured	Estimated	Estimated	Estimated

bias corrections was obtained by matching calculated-position time histories with radar and barometric altitude records. The wind components were then found as the difference between the aircraft velocities with respect to Earth and air mass, in the Earth frame.

The first example illustrates a wind-estimation application and uses the measurement set available to Parks et al. in their analysis of the DC-10 turbulence upset (see table 4.1). In the analysis of this problem by SMACK, all elements of the forcing-function vector (ϕ, θ, ψ), (g_x, g_y, g_h), and (d_x, d_y, d_h) are estimated. All of the measurement time histories (a_x, a_y, a_z), (V, α, β), and (R, B, h) are fitted in the least-squared-error procedure. The resulting wind estimates are shown in figure 4.1, along with the "true" winds for comparison. The close agreement of the horizontal wind records indicated there and in table 4.2 is probably better than could be expected in practice, since ATC enroute radar data are recorded only about once every 10 sec.

Example 2

Other applications of state estimation that are becoming increasingly important are associated with the testing of high-performance aircraft. In large angle-of-attack maneuvers and spin tests, for example, measurements of Euler angles, air-speed, and aerodynamic angles (e.g., angles of attack and sideslip) may contain significant errors. In a recent paper, Taylor (ref. 36) discussed the estimation of Euler-angle time histories and air-data instrument bias errors and scale factors for a spinning airplane. The measurement set for this application consisted of accelerometer, rate-gyro, and air-data measurements. The winds were assumed to be known. With the measured accelerations and angular velocities as forcing functions, Taylor "fitted" the air-variable measurement records, using a squared-error

ESTIMATION OF WINDS

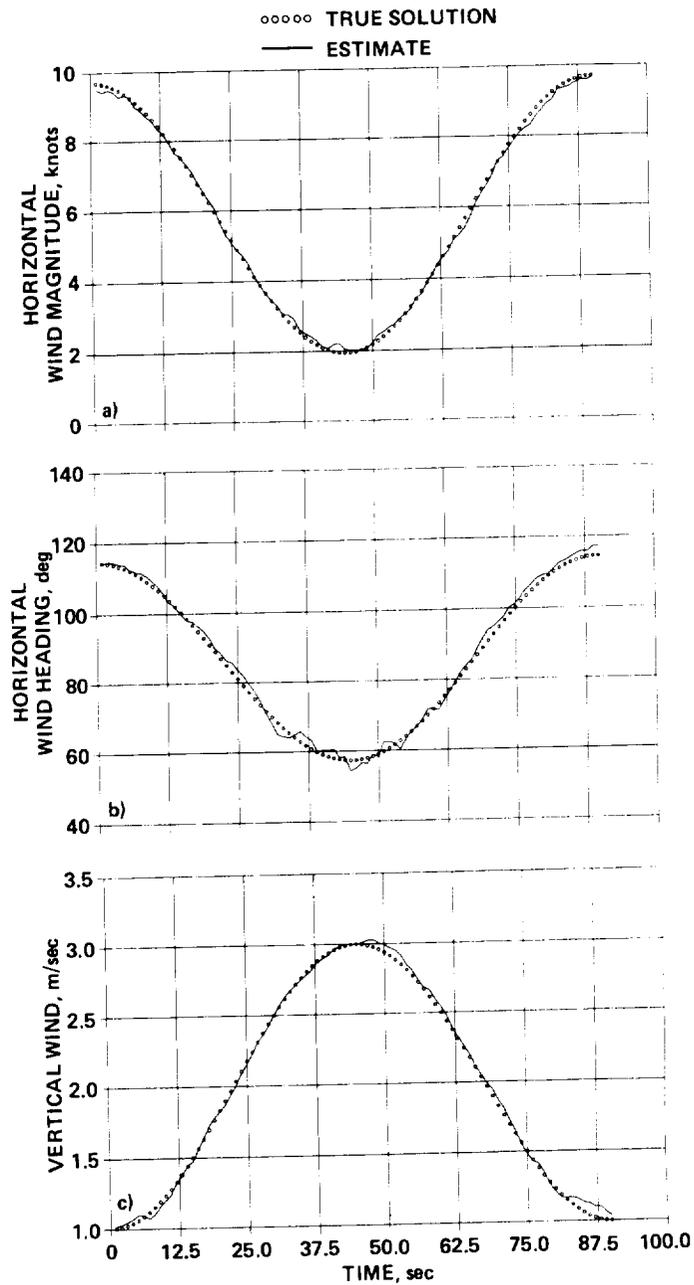


Figure 4.1.- Winds for example 1. (a) Horiz. magnitude, (b) horiz. heading, (c) vertical wind.

criterion and a Newton-Raphson algorithm to determine the desired estimates of biases, scale factors, and Euler-angle time histories. To avoid possible singularities in angle calculations, Taylor utilized the differential equations relating the angular velocities and direction cosines.

TABLE 4.2.- RESULTS FOR STATE-ESTIMATION EXAMPLES

Variable	1		2		3		4		5	
	Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.
Winds										
W_{xy} , kt	0.03	0.12								
W_{hd} , deg	-0.84	1.45								
V_{wd} , m/s	-0.01	0.03								
Angles										
ϕ , deg			-0.05	0.10					0.00	0.07
θ , deg			0.03	0.04					-0.04	0.09
ψ , deg			-0.75	0.88					0.24	0.29
Air Data										
V , kt					0.08	0.44	0.02	0.05	-0.05	0.16
α , deg					0.02	0.11	0.01	0.05	0.00	0.07
β , deg					0.04	0.16	-0.02	0.06	-0.22	0.28

Note: Mean and S.D. (standard deviation) refer to error between true and estimated time histories.

Example 2 illustrates the application of state estimation for determining Euler angles using the measurement set of Taylor as summarized in table 4.1. In the analysis by SMACK, the inertial wind components (w_x , w_y , w_h) were obtained from the measured winds and used in equation (3.1). Estimated elements of the forcing function were (ϕ , θ , ψ) and (d_x , d_y , d_h). The measurements (a_x , a_y , a_z), (p , q , r), and (V , α , β) were fitted, with bias-error and scale-factor estimates obtained for the air-data records. The Euler-angle estimates are shown in figure 4.2, plotted with the corresponding true values. Estimation errors are given in table 4.2. It should be noted that the pitch-angle excursion is not large along the simulated trajectory. For extreme maneuvers, in which the pitch angle may approach 90° , it is not possible to avoid singularities using the linear (coordinate-transformed) state model used in the SMACK program.

Examples 3 and 4

For some large angle-of-attack maneuvers, merely estimating bias errors and scale factors for the air data may not be sufficient. In a paper describing the identification of indicial functions, Gupta and Iliff (ref. 37) found it necessary to obtain estimates for air-variable time histories for the high angle-of-attack flight-test regime. The data used in the solution of this problem consisted of onboard measurements of Euler angles, as well as radar tracking data (slant range, bearing, and elevation angles). Winds were estimated during low angle-of-attack portions of the test when air data were usable. The winds were assumed to remain constant during those test segments when the air variables were to be estimated. The estimates were obtained by "smoothing" the radar data for the Earth-frame

ESTIMATION OF EULER ANGLES

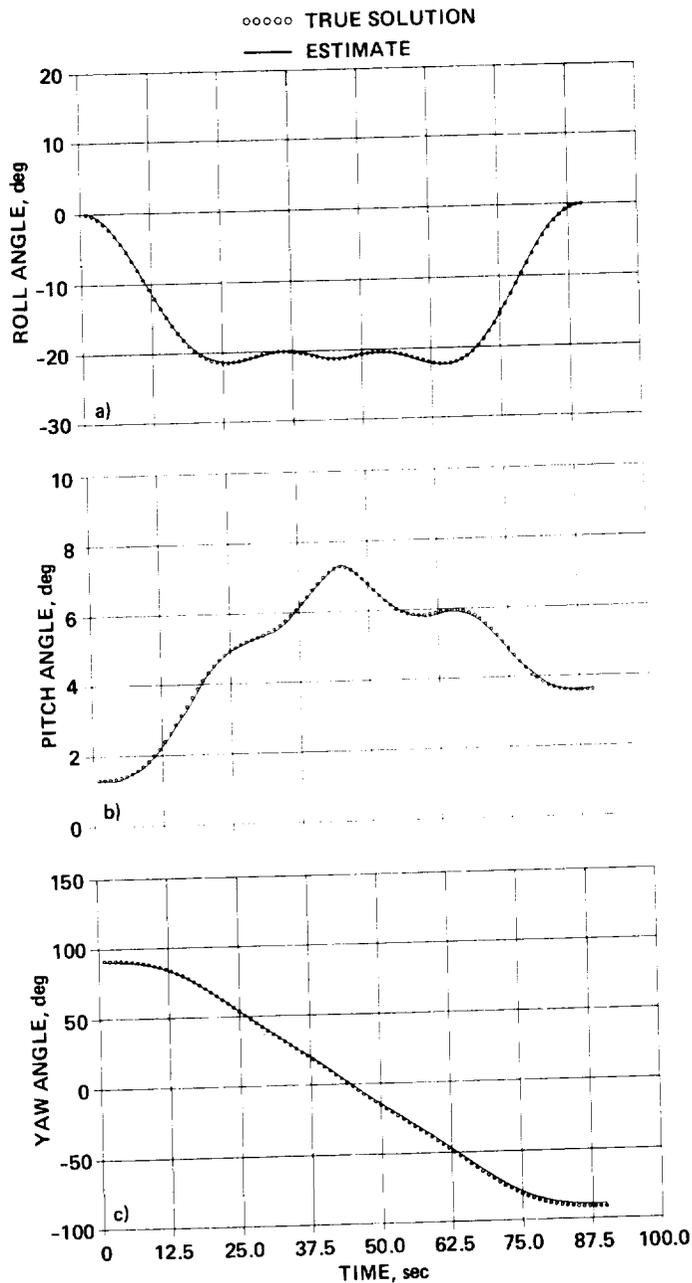


Figure 4.2.- Angles for example 2. (a) Roll angle, (b) pitch angle, (c) yaw angle.

components of aircraft velocity, subtracting the winds, and then transforming to the aircraft body-frame system to calculate the desired estimates of airspeed, angle of attack, and angle of sideslip.

The measurement set employed by Gupta and Iliff for the estimation of air-data variables, shown in table 4.1, is the basis for example 3. In the solution of this

example, the wind components are again assumed to be known. Here the estimated part of the forcing-function vector consisted of elements (ϕ, θ, ψ) and (x, y, h) , and the measurements fitted were (ϕ, θ, ψ) and (R, B, h) . The results of the solution for the air variables (V, α, β) are shown in figure 4.3 and table 4.2. In an application such as this, the radar data-sample rate may not be high enough to provide sufficient air-variable estimates (in the Gupta application, the sample rate was 1 Hz). It may be both necessary and convenient to augment the measurement set with onboard accelerometer data. Example 4 illustrates this case by including (a_x, a_y, a_z) in the measurement set to be fitted. The results are shown in figure 4.4 and table 4.2, where a comparison can be made with the results of the preceding example.

Example 5

The Taylor application requires air-data measurements, whereas the Gupta-Iliff application requires Euler-angle measurements. It would be useful in some extreme flight-test situations to be able to estimate both sets of variables. That this can be accomplished by state estimation is illustrated by a final example (example 5). As indicated in table 4.1, this procedure utilizes radar position data (including altitude), and measurements of the "strap-down" variables (linear accelerations and angular velocities). Results of a simulation experiment as obtained by SMACK are shown in figure 4.5, where good correspondence between estimated and true time-histories can be observed. A comparison of the estimation accuracy obtained here with the results of the three previous experiments can be seen in table 4.2.

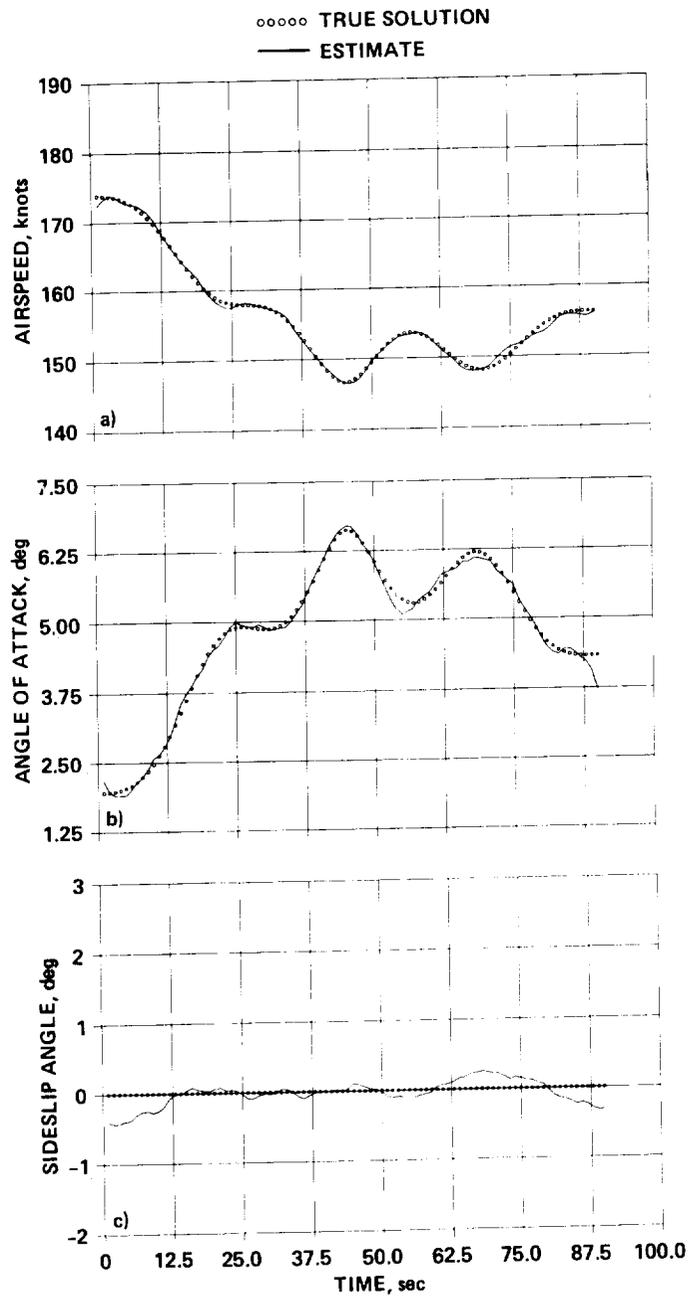


Figure 4.3.- Air data for example 3. (a) True airspeed, (b) angle of attack, (c) sideslip angle.

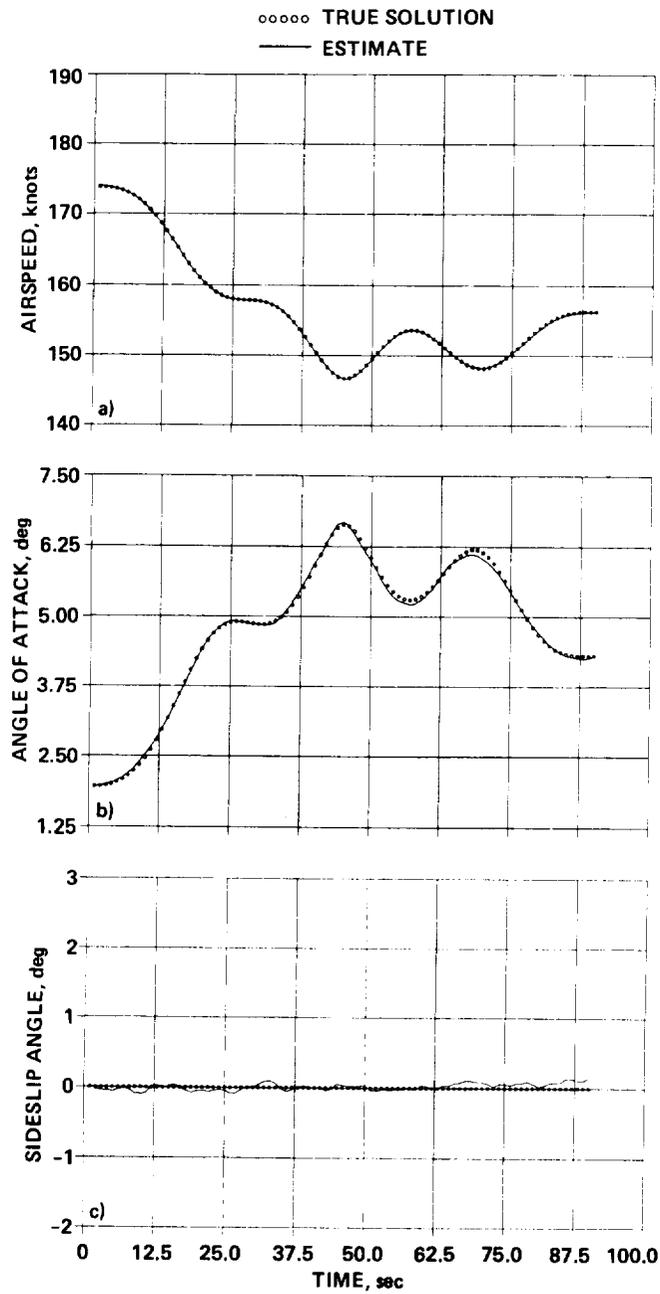


Figure 4.4.- Air data for example 4. (a) True airspeed, (b) angle of attack, (c) sideslip angle.

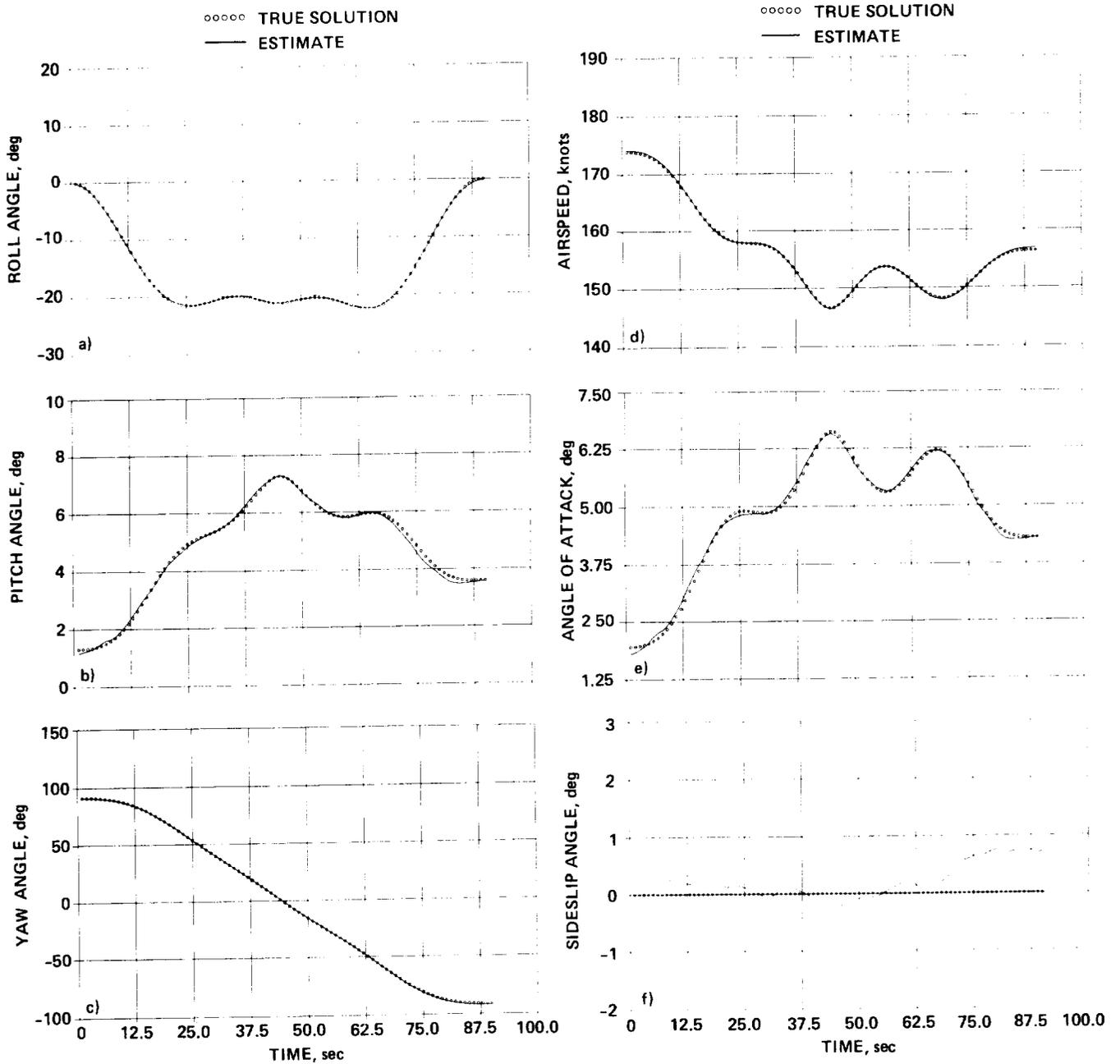


Figure 4.5.- Euler angles and air data variables for example 5. (a) Roll angle, (b) pitch angle, (c) yaw angle, (d) true airspeed, (e) angle of attack, (f) sideslip angle.

5. CODING A PROBLEM FOR SMACK

The algorithm and rigid-body model utilized by SMACK, and several state estimation applications have been discussed in previous chapters. In this chapter, the coding procedure for analyzing flight-test and accident data will be presented. All coding is prepared in 80-column statement lines. The FORTRAN 77 conventions for integer constants (I5 format), decimal constants (F10.0 format), and Hollerith characters (A3 format) are used. With the exception of the first, all statement lines in the coding list are similarly formatted in columns (col.), as follows:

Col. 1 - 3	Variable name or function descriptor
Col. 6 - 10	I, a right-justified integer constant
Col. 11 - 15	J, a right-justified integer constant
Col. 16 - 20	K, a right-justified integer constant
Col. 21 - 25	L, a right-justified integer constant
Col. 31 - 40	VAL1, a decimal constant
Col. 41 - 50	VAL2, a decimal constant
Col. 51 - 60	VAL3, a decimal constant
Col. 61 - 70	VAL4, a decimal constant
Col. 71 - 80,	VAL5, a decimal constant

The first statement line of the SMACK coding list must be a problem description, such as

```
CODING LIST FOR AN A/C STATE ESTIMATION PROBLEM
```

a message that may contain 48 characters. The next line in the list must be a solution description, which should be coded as

```
123 10 15 20 25
```

```
MKS I J K L
```

or

```
ENG I J K L
```

where MKS and ENG define the system of units used to display problem variables. Integers I, J, K, and L are interpreted as follows:

I number of iterations to obtain a final solution
J number of iterations of a starting solution
K=1 output format for accident analysis
L=1 aircraft simulation (rising, 180° turn in wind)

Convergence of the SMACK algorithm is usually accomplished within ten iterations ($I < 10$). The starting set of initial conditions and forcing functions determined by the program directly influences convergence properties. Occasionally it may be helpful to iterate the starting solution, which is done by using state measurements (when available) to evaluate the Jacobian matrices. The number of iterations desired is specified by the value of J . A special output format for displaying the results of an accident analysis is chosen by setting $K=1$. A test problem included in appendix D illustrates this option. The last parameter ($L=1$) is used to initiate an analysis of a simulated maneuver, which is useful for testing a given coding list. All of the examples shown in the previous chapter were prepared by using the simulated maneuver.

Other statements in the coding list may be placed in any order, except for the END statement, which must be last. It appears as

```
123 10 15
```

```
END I J
```

where, for

```
I=-1    page plot of starting solution
=1      page plot of final solution
=-2,-3  x-y plot of starting solution
=2, 3   x-y plot of final solution
J=1     analysis of coding list printed
```

Plots include all output variables mentioned in the coding list. In the page plot, the time variable runs lengthwise on a printer page, and may continue for several pages. The page-plot routine exists as a SMACK subroutine; the x-y plot routines require IMSL and DISSPLA libraries. The IMSL plots ($I=-2,2$) are produced on the system line printer, whereas DISSPLA plots ($I=-3,3$) are produced on a plotting device. Specification of a coding list analysis ($J=1$) is useful for detecting coding errors and as an aid in learning how the program works.

The coding list must have one entry for each quantity considered as an "output" variable in the solution. Outputs include measured variables as well as variables to be estimated. Specification of an output variable appears as

```
123 10 15 20 25 31-40 41-50 51-60 61-70 71-80
VAR I J K L VAL1 VAL2 VAL3 VAL4 VAL5
```

where

VAR variable name, chosen from list of table 5.1

I=1 variable has been measured

J=1 estimate variable time history

K=1 estimate instrument bias error

L=1 estimate instrument scale factor

TABLE 5.1.- LIST OF OUTPUT VARIABLES

Row	Symbol	Description	Internal units ^a	External units	
				MKS	ENG
1	PHI	Roll angle	r	d	d
2	THT	Pitch angle	r	d	d
3	PSI	Yaw angle	r	d	d
4	X	Position (north)	m	nm	nm
5	Y	Position (east)	m	nm	nm
6	H	Altitude (ASL)	m	m	f
7	RNG	Slant range	m	nm	nm
8	BRG	Bearing angle	r	d	d
9	ELV	Elevation angle	r	d	d
10	WXY	Horizontal wind speed	m/s	kt	kt
11	WHD	Horizontal wind heading	r	d	d
12	VWD	Vertical wind speed	m/s	m/s	f/s
13	VT	True airspeed	m/s	kt	kt
14	AV	Angle of attack	r	d	d
15	BV	Sideslip angle	r	d	d
16	AX	Body specific force	m/s ²	g	g
17	AY	Body specific force	m/s ²	g	g
18	AZ	Body specific force	m/s ²	g	g
19	P	Roll rate	r/s	d/s	d/s
20	Q	Pitch rate	r/s	d/s	d/s
21	R	Yaw rate	r/s	d/s	d/s
22	AL	Roll acceleration	r/s ²	d/s ²	d/s ²
23	AM	Pitch acceleration	r/s ²	d/s ²	d/s ²
24	AN	Yaw acceleration	r/s ²	d/s ²	d/s ²
25	RNA	Slant range (aux)	m	nm	nm
26	BRA	Bearing angle (aux)	r	d	d
27	ELA	Elevation angle (aux)	r	d	d
28	HDG	Heading angle	r	d	d
29	VGR	Groundspeed	m/s	kt	kt
30	TRK	Groundtrack	r	d	d

^aFlight data must be converted to internal units in subroutine DATA (see Appendix E).

VAL1 one-sigma value of measurement noise
 VAL2 a priori estimate of instrument bias error
 VAL3 one-sigma value for a priori bias estimate
 VAL4 a priori estimate of instrument scale factor
 VAL5 one-sigma value for an a priori scale-factor estimate

The units for VAL1, VAL2, and VAL3 are designated by the choice of MKS or ENG (see table 5.1). The default value for either VAL3 or VAL5 is infinity (indicating no confidence in the a priori estimate). Note that for external data records, setting VAL1>0 takes precedence over a program-determined noise RMS value. For simulated (internally generated) data, setting VAL1>0 specifies the amount of noise to be added to the record, as well as the weight to be used in the solution. For VAL1=0, no noise is added and a weight of unity is used in the solution.

Note that the preceding statement-line format should also be used when it is desired to use measured linear accelerations and/or angular velocities as forcing functions. This may be useful when performing an initial data-consistency check. Each measured forcing function should be specified by setting I=1, J=0 in the coding list. Bias errors and scale factors may also be specified. However, it is usually desirable to estimate all forcing-function time histories. An entry in the coding list for each forcing function to be estimated should appear as

```
123 10 15 20      31-40 41-50 51-60
VAR 2  J  K      VAL1  VAL2  VAL3
```

where

VAR variable name, chosen from list of table 5.2
 J=1 estimate forcing-function time history
 K=1 estimate forcing-function mean value
 VAL1 RMS value of forcing function
 VAL2 a priori estimate of forcing-function mean value
 VAL3 one-sigma value for a priori mean value estimate

The units for VAL1, VAL2, and VAL3 are designated by the choice of MKS or ENG. The default value for VAL3 is infinity. Note that if VAL1>0, that value takes precedence over a program-determined RMS value for either external or simulated data records. When all the elements of either (AL, AM, AN) or (AX, AY, AZ) are specified

TABLE 5.2.- LIST OF FORCING-FUNCTION VARIABLES

Row	Symbol	Description	Internal units	External units MKS	ENG
1	DL	PH2 time-derivative	r/s ³	d/s ³	d/s ³
2	DM	TH2 time-derivative	r/s ³	d/s ³	d/s ³
3	DN	PS2 time-derivative	r/s ³	d/s ³	d/s ³
4	DX	X2 time-derivative	m/s ³	m/s ³	f/s ³
5	DY	Y2 time-derivative	m/s ³	m/s ³	f/s ³
6	DH	H2 time-derivative	m/s ³	m/s ³	f/s ³
7	GX	WX time-derivative	m/s ²	m/s ²	f/s ²
8	GY	WY time-derivative	m/s ²	m/s ²	f/s ²
9	GH	WH time-derivative	m/s ²	m/s ²	f/s ²
10	PH2	PH1 time-derivative	r/s ²	d/s ²	d/s ²
11	TH2	TH1 time-derivative	r/s ²	d/s ²	d/s ²
12	PS2	PS1 time-derivative	r/s ²	d/s ²	d/s ²
13	X2	X1 time-derivative	m/s ²	m/s ²	f/s ²
14	Y2	Y1 time-derivative	m/s ²	m/s ²	f/s ²
15	H2	H1 time-derivative	m/s ²	m/s ²	f/s ²

with I=1, J=1 in the coding list, forcing functions (DL, DM, DN) or (DX, DY, DH) are selected by the program. Hence, those forcing functions should be included in the list only to override a program-determined RMS weight, or to specify estimation of a mean value. Their inclusion in the coding list, however, is a useful reminder of the excitations chosen for the state model shown in figure 3.1. Care should be taken not to mix elements of (DL, DM, DN) with elements of (PH2, TH2, PS2), or elements of (DX, DY, DH) with elements of (X2, Y2, H2).

Inclusion of a particular state variable in the coding list is necessary only to specify an a priori initial condition of that state. The statement line should appear as

```
123 10          41-50 51-60
VAR 3          VAL2  VAL3
```

where

VAR variable name, chosen from list of table 5.3
 VAL2 an a priori estimate of the initial condition
 VAL3 one-sigma value for the a priori state estimate

TABLE 5.3.- LIST OF STATE VARIABLES

Row	Symbol	Description	Internal units	External units MKS	ENG
1	PH2	PH1 time-derivative	r/s ²	d/s ²	d/s ²
2	TH2	TH1 time-derivative	r/s ²	d/s ²	d/s ²
3	PS2	PS1 time-derivative	r/s ²	d/s ²	d/s ²
4	X2	X1 time-derivative	m/s ²	m/s ²	f/s ²
5	Y2	Y1 time-derivative	m/s ²	m/s ²	f/s ²
6	H2	H1 time-derivative	m/s ²	m/s ²	f/s ²
7	WX	Wind speed (north)	m/s	m/s	f/s
8	WY	Wind speed (east)	m/s	m/s	f/s
9	WH	Wind speed (up)	m/s	m/s	f/s
10	PH1	PHI time derivative	r/s	d/s	d/s
11	TH1	THT time-derivative	r/s	d/s	d/s
12	PS1	PSI time-derivative	r/s	d/s	d/s
13	X1	X time-derivative	m/s	m/s	f/s
14	Y1	Y time-derivative	m/s	m/s	f/s
15	H1	H time-derivative	m/s	m/s	f/s
16	PHI	Roll angle	r	d	d
17	THT	Pitch angle	r	d	d
18	PSI	Yaw angle	r	d	d
19	X	Position (north)	m	nm	nm
20	Y	Position (east)	m	nm	nm
21	H	Altitude (ASL)	m	m	f

It should be emphasized that specifying an a priori estimate with its corresponding one-sigma value for any variable will bias the solution towards that estimate. A priori estimates will seldom be necessary for convergence of the algorithm. Again, the units for VAL2 and VAL3 are designated by the choice of MKS or ENG.

Each coding list must contain a description of the data record and the way in which it is to be processed. This description includes the number of data points, the sampling interval, and the integration time step. The code should appear as

```
123 10 15 20 25 31-40 41-50
```

```
REC I J K L VAL1 VAL2
```

where

I starting point of record

J ending point of record

K number of integration steps per sampling interval

L integration steps per output point (plotting only)

VAL1 data sampling interval in seconds

VAL2 filter cutoff frequency in Hertz

For measurement sets with multiple-rate data, the starting and ending points should correspond to the record with the highest data rate. The time step used for integration is chosen to be an integral submultiple of each of the sampling intervals (see appendix E). The sampling interval VAL1 should be that of the record with the highest data rate. Note that each measurement record is low-pass filtered in the starting routine in order to obtain a measure of the residual covariance. The cutoff frequency is VAL2, which should be adjusted so that residuals of the records with the highest data rate are as "white" as possible. The cutoff frequencies for other records in the measurement set are program-scaled by sample-rate ratios. The default value for VAL2 is 0.1/VAL1 Hz.

It may sometimes be useful to independently specify xy-plot scales or a filter cutoff frequency for a record. This can be done by including in the coding list the statement line

```

123 10                    31-40 41-50 51-60 61-70 71-80
VAR 4                    VAL1  VAL2  VAL3  VAL4  VAL5

```

where

VAR variable name, chosen from list of table 5.1

VAL1 filter cutoff frequency in Hertz

VAL2 x-axis minimum value

VAL3 x-axis maximum value

VAL4 y-axis minimum value

VAL5 y-axis maximum value

The units for VAL2 through VAL5 are designated by the choice of MKS or ENG. If all four values are zero, no changes in program-determined plot scales will be made. If VAL1 is zero, the filter cutoff frequency for the record will be that specified by the REC statement line. Note that the filter residuals may be examined for whiteness by obtaining plots of the starting solution (I=-1 in the END statement).

The SMACK program can compensate for instrument offset from the aircraft c.g. One correction can be made for each of three instrument sets: accelerometer, pressure ports, and alpha-beta vanes. These statement lines should appear as

```
123          41-50  51-60  61-70
```

```
ACC          VAL2   VAL3   VAL4
```

for the accelerometer package,

```
P-S          VAL2   VAL3   VAL4
```

for the pitot-static system, and

```
VNE          VAL2   VAL3   VAL4
```

for the vane measurement system. In each statement, VAL2, VAL3, and VAL4 represent the (x, y, z) location of the instrument set with respect to the c.g., in meters (MKS) or in feet (ENG).

For a problem that includes radar tracking data, it is possible to specify the site location with respect to a desired origin using the statement line

```
123          41-50  51-60  61-70
```

```
RAD          VAL2   VAL3   VAL4
```

where VAL2, VAL3, and VAL4 represent the (x, y, h) location of the tracking antenna. Here VAL2 and VAL3 are in nautical miles, and VAL4 is in meters (MKS) or feet (ENG). When an auxiliary site has provided tracking data, its location can similarly be represented with

```
RDA          VAL2   VAL3   VAL4
```

For accident analysis, when the only data available are the radar track (including altitude), winds, air temperature and aircraft performance data, none of the aircraft trajectory variables (e.g., attitude or velocity) need be specified in the coding list. The trajectory variables will be determined following the radar solution by specifying K=1 in the solution description. The first problem included in appendix D illustrates this application.

Preparation of the coding list will here be illustrated by returning to the application examples of the previous chapter. A list for each example is found in figures 5.1 through 5.5. In each case, the solution description specifies MKS units, eight iterations and the SMACK aircraft simulation to provide the data records. The REC statement specifies that each record will have 90 points, the integration step will be the same as the sampling interval, every point will be plotted, and the sampling interval is one second. To help interpret the VAR

EXAMPLE 1: ESTIMATION OF WINDS

MKS	8			1	
REC	1	90	1	1	1.0
AX	1	1	1		0.001
AY	1	1	1		0.001
AZ	1	1	1		0.001
RNG	1	1			0.001
BRG	1	1			0.05
H	1	1			0.5
WXY		1			
WHD		1			
VWD		1			
GX	2	1			
GY	2	1			
GH	2	1			
PHI	1	1			0.05
THT	1	1			0.05
PSI	1	1			0.05
PH2	2	1			
TH2	2	1			
PS2	2	1			
VT	1	1			0.1
AV	1	1			0.05
BV	1	1			0.05
END	2	1			

Figure 5.1.- Coding list for application example 1.

EXAMPLE 2: ESTIMATION OF EULER ANGLES

MKS	8			1	
REC	1	90	1	1	1.0
AX	1	1			0.001
AY	1	1			0.001
AZ	1	1			0.001
P	1	1	1		0.05
Q	1	1	1		0.05
R	1	1	1		0.05
WXY	1				
WHD	1				
VWD	1				
PHI		1			
PHI	3				0.
THT		1			5.
THT	3				1.31
PSI		1			5.
PSI	3				90.
PH2	2	1			1.
TH2	2	1			
PS2	2	1			
VT	1	1	1	1	0.1
AV	1	1	1	1	0.05
BV	1	1	1		0.05
END	2	1			

Figure 5.2.- Coding list for application example 2.

EXAMPLE 3: ESTIMATION OF AIR VARIABLES

MKS	8			1	
REC	1	90	1	1	1.0
RNG	1	1			0.001
BRG	1	1			0.05
H	1	1			0.5
X2	2	1	1		
Y2	2	1	1		
H2	2	1	1		
WXY	1				0.1
WHD	1				0.05
VWD	1				0.1
PHI	1	1			0.05
THT	1	1			0.05
PSI	1	1			0.05
PH2	2	1			
TH2	2	1			
PS2	2	1			
VT		1			
AV		1			
BV		1			
END	2	1			

Figure 5.3.- Coding list for application example 3.

EXAMPLE 4: ESTIMATION OF AIR VARIABLES

MKS	8			1	
REC	1	90	1	1	1.0
AX	1	1			0.001
AY	1	1			0.001
AZ	1	1			0.001
RNG	1	1			0.001
BRG	1	1			0.05
H	1	1			0.5
WXY	1				0.1
WHD	1				0.05
VWD	1				0.1
PHI	1	1			0.05
THT	1	1			0.05
PSI	1	1			0.05
PH2	2	1			
TH2	2	1			
PS2	2	1			
VT		1			
AV		1			
BV		1			
END	2	1			

Figure 5.4.- Coding list for application example 4.

EXAMPLE 5: ESTIMATION OF ANGLES, AIR VARIABLES

MKS	8			1		
REC	1	90	1	1	1.0	
AX	1	1			0.001	
AY	1	1			0.001	
AZ	1	1			0.001	
P	1	1	1		0.05	
Q	1	1	1		0.05	
R	1	1	1		0.05	
RNG	1	1			0.001	
BRG	1	1			0.05	
H	1	1			0.5	
WXY	1					
WHD	1					
VWD	1					
PHI		1				
PHI	3					0.
THT		1				2.0
THT	3				1.313	2.0
PSI		1				
PSI	3				91.35	1.0
PH2	2	1				
TH2	2	1				
PS2	2	1				
VT		1				
BV		1				
AV		1				
END	2	1				

Figure 5.5.- Coding list for application example 5.

statement lines, refer again to table 4.1, which lists the measured and/or estimated variables for each example. Notice that in examples 2 and 5, a priori estimates of the Euler angles were specified. For example 3, forcing functions (X2, Y2, H2) were specified, but in example 4, accelerometer measurements (AX, AY, AZ) were available, and corresponding forcing functions (DX, DY, DH) were used by the program. In example 4, the user does not need to specify the forcing-function set. For examples 2 through 5, the winds (WXY, WHD, VWD) were specified as measured, but not estimated. In these examples, states (WX, WY, WH) are computed from the measurements and used with (X1, Y1, H1) in the estimation of air velocities. Finally, note that the coding list for each example concludes with an END statement specifying that x-y plots of the output variables and a diagnostic analysis of the problem are to be printed.

6. A FLIGHT-TEST METHODOLOGY

This chapter describes a flight-test methodology for acquiring a data base to identify a full-envelope aerodynamic model of a V/STOL Research Aircraft (VSRA). The model will serve to update and improve an existing VSRA simulation, in order to aid the design of guidance, control, and display systems for the aircraft. A key element in the methodology is the application of SMACK for the processing of each test maneuver before its entry to the data base. It should be helpful to the reader to see how the state-estimation method may be used in a flight-test setting. For a more complete discussion see references 38 and 39.

The NASA VSRA is a YAV-8B aircraft, a prototype of the subsonic, vectored-thrust "Harrier" fighter aircraft; its engine nozzles can be rotated from zero degrees for forward flight to somewhat greater than 90° for hover and vertical flight. A reaction-control system (RCS), in which compressor air is piped to the extremities of the aircraft, provides attitude control in hover and low-speed flight. The VSRA aerodynamic model must represent the three body forces and three moments over a flight envelope that includes hover, transition to forward flight and back to hover, and STOL operation and normal cruise.

The resulting model, strongly nonlinear with respect to aircraft variables such as angles of attack and sideslip, Mach number, nozzle angle, and power setting, can be conveniently expressed with functions that are linear in the parameters to be identified (refs. 40-42). A linear least-squares (regression) method (refs. 43-45) is well-suited to identify a highly nonlinear model that is linearly parameterized. Because regression methods are computationally simple, careful attention can be given to the structuring of an accurate and physically meaningful model. Good results with regression methods, however, are highly dependent on the quality of the flight data. Therefore, state-estimation methods are often used before modelling to correct the data records for bias and scale-factor errors and to provide estimates of unmeasured or poorly measured variables.

The methodology for acquiring a data base matched to a least-squares (regression) identification task is outlined in the flow diagram shown in figure 6.1. The important aspects of the preflight planning, flight testing, and postflight processing phases necessary to acquire the data base are covered in this chapter. The role of state estimation in the processing is emphasized. The actual modelling of VSRA aerodynamics is beyond the scope of this discussion.

Maneuver Design

The data base required for least-squares aerodynamic model identification can be obtained quite efficiently. Because the model is nonlinear, it is not necessary (or useful) to maintain trim during a maneuver. In addition, because a regression procedure will be used to identify the model, large amounts of data may be batch-processed. Accordingly, each flight-test maneuver has been designed to yield large changes in aircraft variables while covering a (nearly) closed course within five

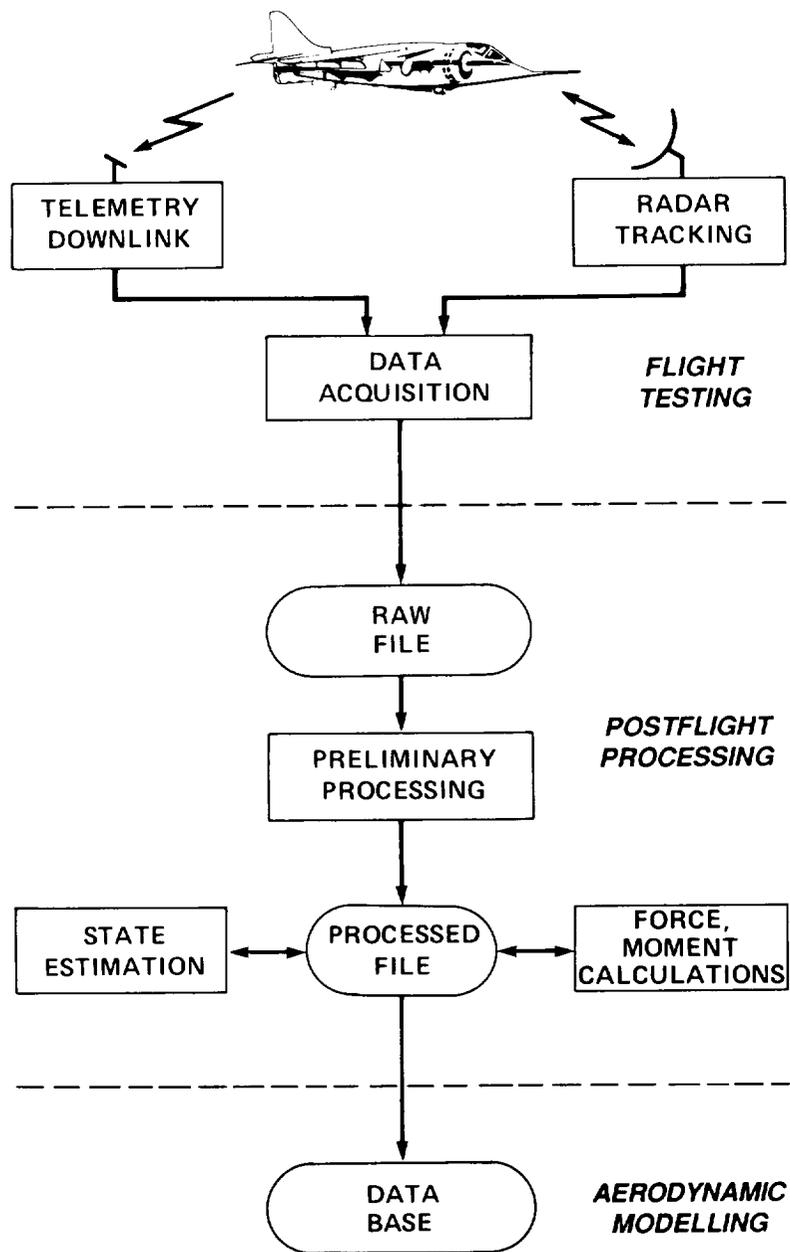


Figure 6.1.- Flow diagram for VSRA flight-test methodology.

minutes under continuous radar tracking. The raw data base consists of as many longitudinal, lateral, and transitional (to and from hover) maneuvers as are necessary to cover the flight envelope. After processing, model sections may be identified using long (15-30 min) records, each consisting of concatenated segments from several maneuvers.

One set of maneuvers was designed to excite large changes in longitudinal-model variables from several nominal trim points. In each of these maneuvers, the nozzle angle, flap deflection, and power are held constant while the stabilator is varied to obtain changes in angle of attack and pitch rate. The maneuver includes "stick pulses," sinusoidal "stick pumping," an "alpha ladder," and a "wind-up" turn (ref. 39). Near the end of the maneuver, power is added to return to the nominal trim point. Note that a considerable variation in Mach number may be experienced during the maneuver. Another set of maneuvers was designed to excite large changes in lateral-model variables (angle of sideslip, yaw rate, and roll rate). Most of the maneuvers were performed without "stability augmentation" to ensure a full range of aircraft response activity. All V/STOL procedures were performed in and out of ground effect.

One characteristic that sets the VSRA apart from conventional aircraft is that it exhibits significant thrust-induced aerodynamic effects when the nozzles are not in the full-aft position. These are largest during transition from hover to forward flight (and back to hover) and during periods of low-speed flight. Standard V/STOL procedures were used to provide data for identification of thrust-induced aerodynamics. One of these procedures, a short-takeoff and slow-landing maneuver, is outlined on the flight-test card shown in figure 6.2. In this maneuver, the ground roll begins with nozzles at 10° . At V_r (indicated air speed) the nozzles are rotated to an angle θ_r (in the example for this chapter, $V_r = 50$ kt and $\theta_r = 55^\circ$). Shortly after liftoff, the nozzles are rotated to the full-aft position. For the slow-landing portion, nozzles are rotated to 40° just before the final turn, and during the final approach are further rotated to 60° .

Data Acquisition

The VSRA measurement system is equipped with a 10-bit digital data acquisition and telemetry (TM) system. A pulse-code modulation format is used to encode 156 mainframe channels sampled at 120 Hz and 160 subframe channels sampled at 30 Hz. Before encoding, each analog channel is passed through a third-order Butterworth anti-aliasing filter with its cutoff frequency set at one-fifth of the channel sampling rate. After encoding, all flight data are transmitted to a ground station where they are recorded. A partial list of onboard measurements, those necessary for aerodynamic model identification, is given in table 6.1.

Flight tests of the VSRA were performed at the NASA test facility located at Crows Landing, California. The facility control room, which has a clear view of the runway and hover pad, is equipped with five eight-channel strip-chart recorders and

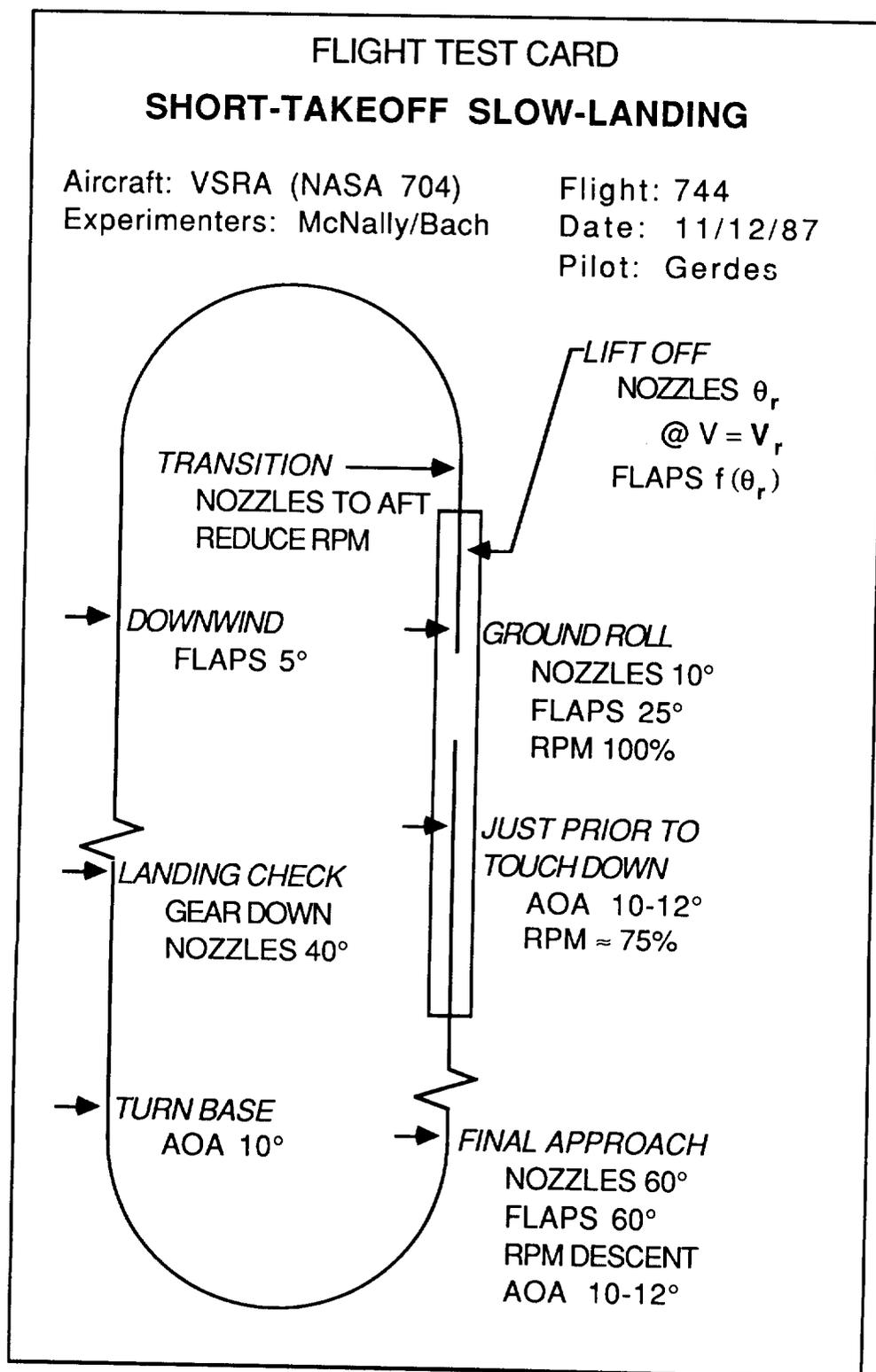


Figure 6.2.- Plan view of example VSRA flight-test maneuver (exact reproduction of flight-test card used by pilot).

TABLE 6.1.- VARIABLE LIST FOR AERODYNAMIC MODEL DATA BASE

Channel	Measured	Estimated
Euler angles	Onboard	SMACK
Angular rates	Onboard	SMACK
Angular accelerations		SMACK
Linear accelerations	Onboard	SMACK
Inertial positions	Radar	SMACK
Inertial velocities		SMACK
Air-flow angles	Onboard	SMACK
Static pressure	Onboard	
Total pressure	Onboard	
Total temperature	Onboard	
True airspeed		SMACK ^a
Flightpath winds		SMACK
Flap setting	Onboard	
Aileron deflections	Onboard	
Stabilator deflection	Onboard	
Rudder deflection	Onboard	
Engine nozzle angle	Onboard	
Engine fan speed	Onboard	
Compressor pressure	Onboard	
Fuel and water weights	Onboard	
RCS roll-valve positions	Onboard	
RCS pitch-valve positions	Onboard	
RCS yaw-valve position	Onboard	
Engine and RCS body forces		ENCAL
Engine and RCS moments		ENCAL
Gross weight and inertias		ENCAL

^aSMACK utilizes a "measurement" of true airspeed, which is computed from the ratio of total and static pressures, and the total temperature (see chapter 3 and ref. 29).

three color monitors for real-time display of the TM data. Two on-site radar systems are available to provide continuous tracking of the test aircraft position. (A laser tracking system is used for all hover maneuvers.) During flight test, TM data from the VSRA onboard system are downlinked, merged at the facility with range, bearing, and elevation data from the tracking systems, and then recorded.

Preliminary Processing

Following real-time acquisition of data during flight test, the records from each maneuver are converted to engineering units and made available to researchers in a raw flight-data file. The first program in the postflight processing system reads the raw file and creates a "processed" file of selected channels. The processing begins by removing wild points from the records. Several options are available, but one effective (but time-consuming) method is to pass each record through a "moving window". Points that fall outside the window are considered wild, and are tagged but not removed. When all wild points in a record have been tagged, the record is passed through a low-pass digital filter (see appendix E) to obtain an interpolated time history free of wild points. After interpolation, the data rate can be reduced to a submultiple of the mainframe sampling frequency. The filter cutoff frequency is set at one-half the final data rate desired. The final rate was chosen to be 20 Hz for all VSRA maneuvers.

Each channel processed from a maneuver raw-data file is stored in a processed flight-data file set up for that maneuver. The analyst now may use a program to interactively select processed data channels for plotting in either x-y or strip-chart format. An x-y cross plot, for example, might display Mach number plotted against angle of attack. Such plots offer a convenient way to evaluate how well the flight envelope has been covered during a maneuver. It is unlikely that a single maneuver will provide enough variation in aircraft variables to identify all model terms: the analyst may also use this program to create a "map" file, which will contain addresses of time segments selected from several processed maneuvers. This file can later be used to concatenate the selected segments to create a long record suitable for model identification.

State Estimation

The next step in the processing of each maneuver is to apply SMACK to check data consistency and derive unmeasured variables from the measurement set given in table 6.1. The relatively long (3-5 min) maneuvers with large dynamic variations are well-suited to state-estimation analysis. The closed course yields good tracking accuracy and facilitates estimation of winds along the flightpath. The aircraft dynamic response to the control inputs is analyzed (along with the radar track) by the SMACK procedure, which determines integrator initial conditions, selected instrument bias errors and scale factors, and forcing-function time histories that provide the "best fits" to the measurement records. The body angular accelerations, true airspeed, and flightpath winds are also estimated as part of the solution.

The user must write and link subroutine DATA with the SMACK program. This subroutine accesses the data records and performs the chores required to prepare the problem for solution. In addition to setting up the arrays according to the list in table 5.1, for example, there may be intervals during which some instruments are known to be "saturated" or else sections of data may be unusable for some other reason. The data in such intervals must be "blanked" so that they will not

influence the estimates. These chores are relatively easy to accomplish; the procedures are described in appendix E.

A coding list for the analysis of a typical VSRA flight-test maneuver is shown in figure 6.3. The REC statement indicates a data record of 6000 points and a sampling interval of 0.05 sec (these may be changed by the user in subroutine DATA). A filter cutoff frequency of 6 Hz has been specified for determination of performance-index weights. Notice that data from both tracking systems are being used; the RAD and RDA statements specify their positions with respect to a runway origin. Notice also that position corrections for accelerometer and air-data instrument locations on the aircraft are specified by the ACC, P-S, and VNE statements.

Force and Moment Calculations

The aerodynamic forces and moments acting on the VSRA during flight are determined as the difference between the total forces and moments and the engine forces and moments. Here the term "engine" includes the reaction control system as well as the main nozzles. The engine forces and moments are calculated (offline) by a program called ENCAL (ENgine CALculations). This program uses a nominal propulsion model of the VSRA Pegasus engine (YF402-RR-404) (ref. 46). Fan dynamics are not included in this version, since fan speed is measured in flight. It should be noted that the propulsion model provides only thrust forces and moments. Any thrust-induced aerodynamic effects are to be included in the VSRA aerodynamic model.

Inputs to the ENCAL routine include all the air-data, reaction-control, engine, and weight measurements listed in table 6.1. Outputs to the processed flight-data file are the three body-axis components of engine force and moment. The ENCAL routine also calculates aircraft weight and inertias, and the variation in center-of-gravity location. These variables are added to the processed-data file. Note that the aerodynamic model to be identified from flight data can only be as accurate as the engine model. A fully-instrumented Pegasus engine has recently been installed on the VSRA, and the engine model will be validated after the next set of flight tests.

Total VSRA force and moment time-histories are obtained from the SMACK-derived estimates of accelerations and angular rates, and from ENCAL-derived estimates of weight and inertias. The body-axis forces are given by

$$F_x = ma_x \quad ; \quad F_y = ma_y \quad ; \quad F_z = ma_z \quad (6.1)$$

where m is vehicle mass. The moments are calculated from

$$\begin{aligned} T_l &= I_{xx}a_l - I_{zx}(a_n + pq) - (I_{yy} - I_{zz})qr \\ T_m &= I_{yy}a_m - I_{zx}(r^2 - p^2) - (I_{zz} - I_{xx})pr \\ T_n &= I_{zz}a_n - I_{zx}(a_l - qr) - (I_{xx} - I_{yy})pq \end{aligned} \quad (6.2)$$

(a) VSRA FLIGHT-TEST ANALYSIS (INPUTS ESTIMATED)

ENG	10	0	0	0				
REC	1	6000	1	1	0.05	6.		
PHI	1	1	0	1	0.2	0.8		
THT	1	1	0	0	0.2	-0.6		
PSI	1	1	0	0	0.5	-2.3		
P	1	1	1	0	0.1			
Q	1	1	1	0	0.1			
R	1	1	1	0	0.1			
AL	0	1	0	0				
AM	0	1	0	0				
AN	0	1	0	0				
DL	2	1	0	0				
DM	2	1	0	0				
DN	2	1	0	0				
RNG	1	1	0	0	0.005			
BRG	1	1	0	0	0.05			
ELV	1	1	0	0	0.05			
RAD	0	0	0	0	0.			
RNA	1	1	0	0	0.005	.28337019	.19029757	-1.59
BRA	1	1	0	0	0.05			
ELA	1	1	0	0	0.05			
RDA	0	0	0	0	0.			
AX	1	1	1	0	0.002	.33024389	.14118231	-1.58
AY	1	1	1	0	0.002			
AZ	1	1	1	0	0.005			
ACC	0	0	0	0	0.			
DX	2	1	0	0		0.58	0.	2.58
DY	2	1	0	0				
DH	2	1	0	0				
END	3	1	0	0				

(b)

VT	1	1	0	0	0.1			
P-S	0	0	0	0	0.	25.	0.	1.670
AV	1	1	1	0	0.1	0.	0.	1.136
BV	1	1	1	0	0.1	0.	0.	1.333
VNE	0	0	0	0	0.	25.	0.	1.670
WXY	0	1	0	0				
WHD	0	1	0	0				
VWD	0	1	0	0				
GX	2	1	0	0				
GY	2	1	0	0				
GH	2	1	0	0				

Figure 6.3.- Coding list for VSRA data-consistency analysis. (a) Inertial solution only, (b) additional statements for full solution.

where I_{xx} , I_{yy} , I_{zz} , I_{zx} are vehicle moments of inertia.

Example Maneuver

The short-takeoff and slow-landing maneuver described earlier illustrates the type of information that is stored in the VSRA data base. The maneuver contains abrupt changes in nozzle and flap angles. The aircraft transitions to normal flight after takeoff, performs a "go-around," and then transitions back to a STOL configuration for a slow landing. The raw data file includes all the onboard inertial and air data, and radar tracking measurements as indicated in table 6.1. The variations in nozzle angle, flap setting, power, and control-surface positions required to perform the maneuver are shown in figure 6.4. Only the left aileron is shown: both ailerons are set to 15° down (drooped) during takeoff and landing. Note how these time histories correlate with the activity requested of the pilot on the flight-test card of figure 6.2.

Results of the SMACK analysis required for calculating forces and moments are shown in figure 6.5. During a preliminary solution, a large error was noticed in the fit of longitudinal acceleration (AX) during the takeoff portion of the maneuver. The accelerometer had saturated at 0.6 g's, and its output in that interval had to be "blanked". Fortunately, the good tracking data provided the redundancy necessary to yield the estimate during the blanked interval. The other fits to the measurement time histories were quite good. Although there are no measurements of angular accelerations in the processed data file, the measurements of angular rates are of sufficient quality to ensure confidence in the acceleration estimates.

It should be noted that the large activity in the angular accelerations of figure 6.5(b) is related to the reduced damping of the aircraft without stability augmentation. In effect, the pilot must provide the control inputs to stabilize the aircraft. The control-surface motions in these test data are well-correlated with the angular accelerations. A similar maneuver flown with stability augmentation shows significantly smaller excursions. The larger aircraft response activity obtained without augmentation will, of course, enhance the "identifiability" of the aerodynamic model.

As a final step in maneuver processing, the aerodynamic forces and moments are calculated as the difference of total and engine forces and moments as outlined in the previous section. These are the time histories that must be adequately represented by the VSRA aerodynamic model. Results of the ENCAL calculations for the maneuver are shown in figure 6.6, with the corresponding aerodynamic variables shown in figure 6.7. Notice the tradeoff between engine and aerodynamic vertical forces during the STOL portions of the maneuver.

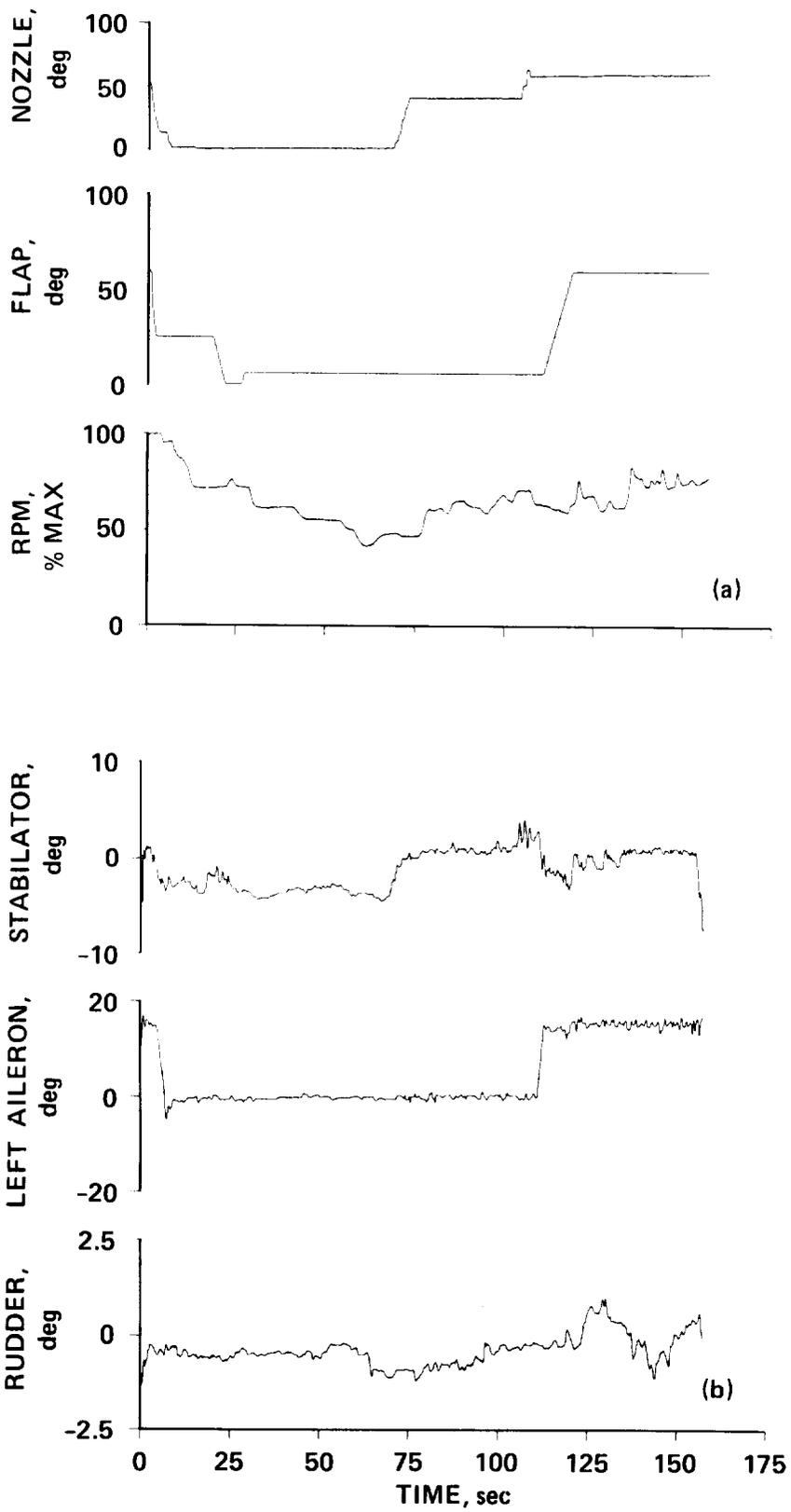


Figure 6.4.- Control variations, VSRA flight-test maneuver. (a) Nozzle, flap, power, (b) stabilator, aileron, rudder.

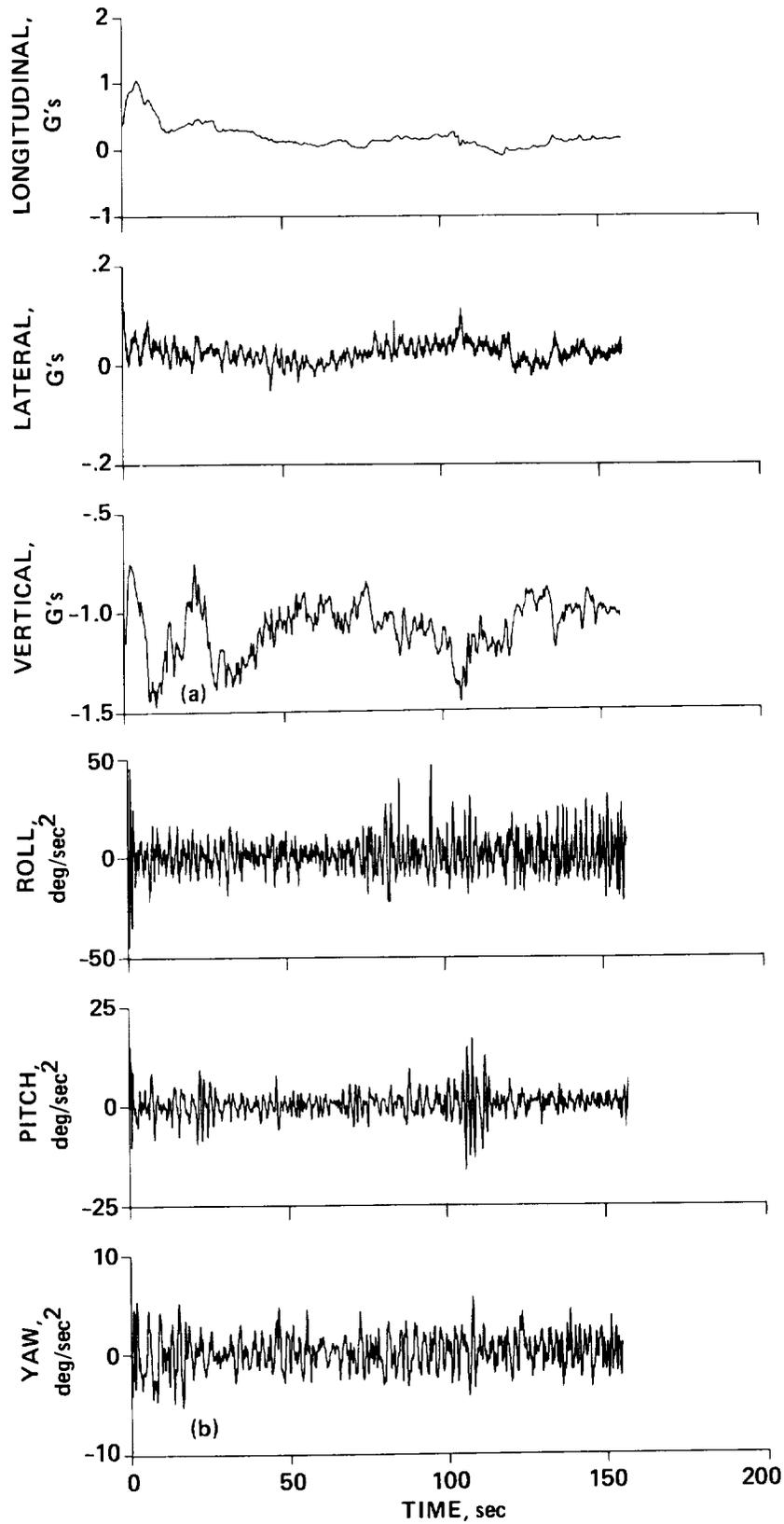


Figure 6.5.- SMACK estimates, VSRA flight-test maneuver. (a) Linear accelerations, (b) angular accelerations.

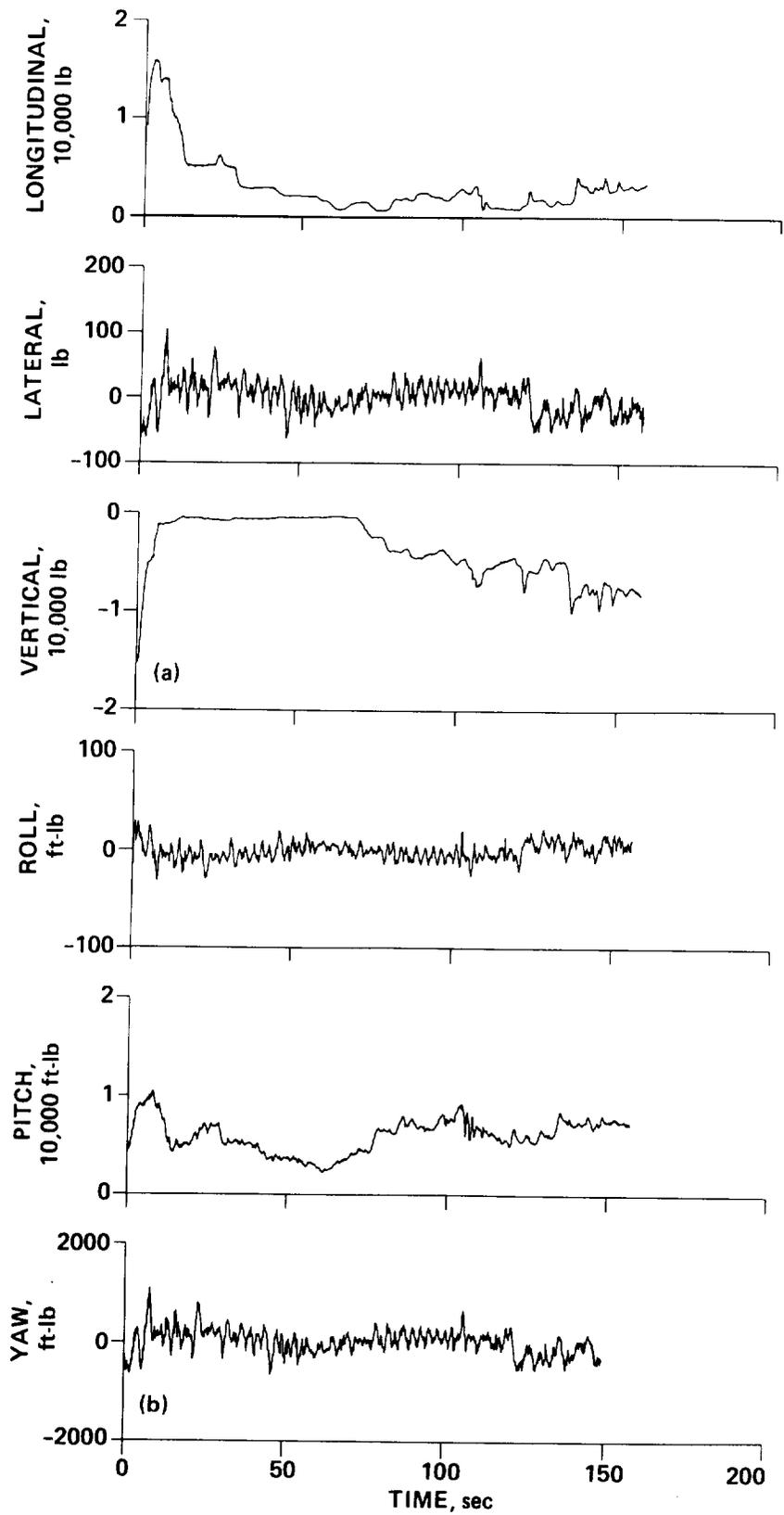


Figure 6.6.- ENCAL estimates, VSRA flight-test maneuver. (a) Engine forces, (b) engine moments.

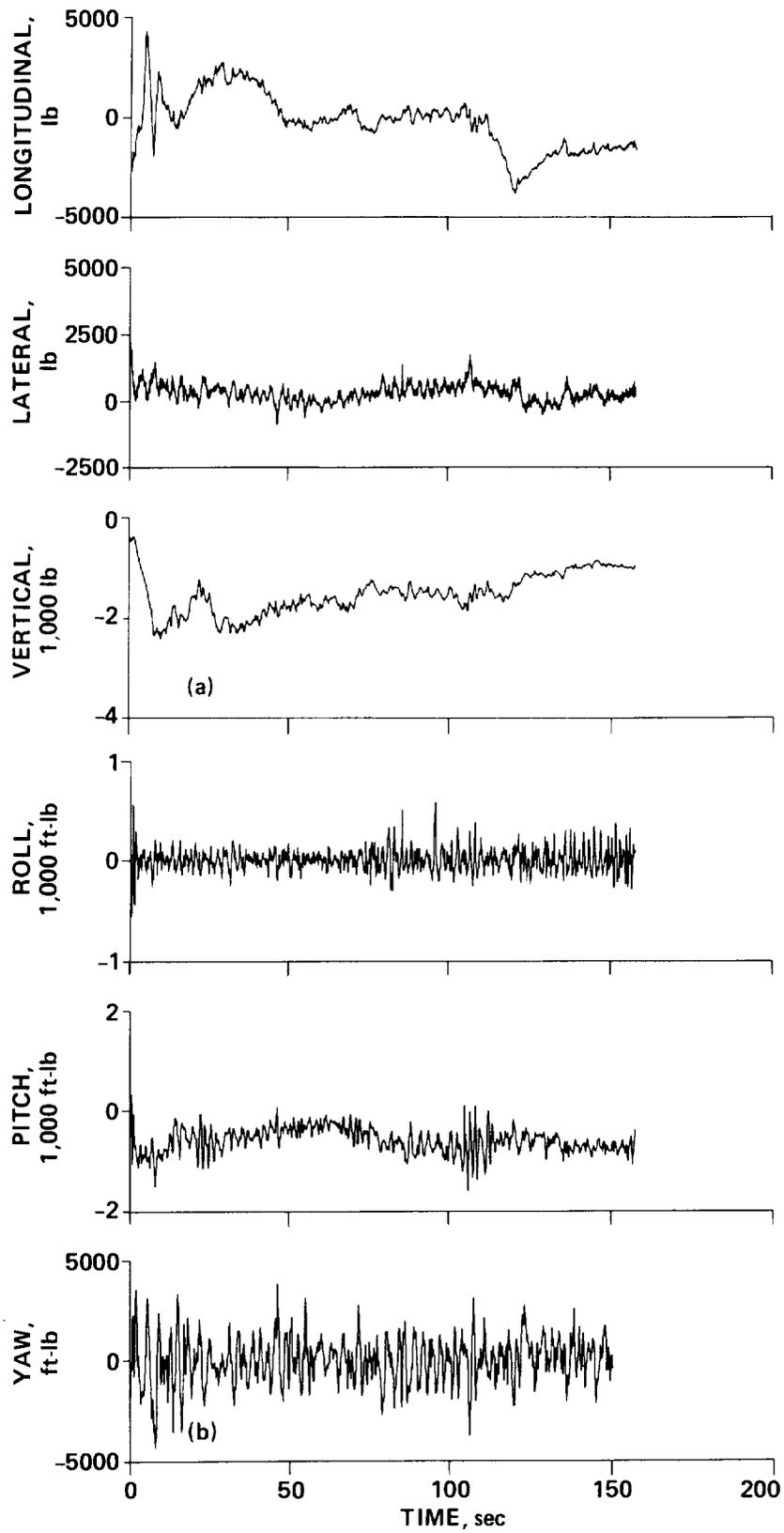


Figure 6.7.- Aerodynamic estimates, VSRA flight-test maneuver. (a) Body forces, (b) body moments.

7. THE ANALYSIS OF WINDSHEAR

This chapter expands on the application of state estimation for the analysis of windshear, which was the subject of the first example of chapter 4. Encounters with severe turbulence represent a continuing safety problem that must be better understood. Atmospheric disturbances that affect airline operations can be studied by analyzing flight data recorded during typical encounters. In the past, such investigations were usually hampered by the lack of good data, but more recent turbulence incidents have involved airliners equipped with digital flight-data recorders (DFDRs). DFDR records, together with ground-based air-traffic control (ATC) radar records, comprise a number of measurements approaching that available from flight test. Hence, it is feasible to determine aircraft performance in turbulence, and to characterize the turbulence environment.

In assisting the National Transportation Safety Board (NTSB) in its investigations of accidents involving aircraft not equipped with digital flight recorders, Ames Research Center developed methods to determine aircraft motions along a flight-path from the limited data available following an accident (refs. 33, 34, and 47). The aircraft motions can now be determined with the SMACK state-estimation method (see appendix D). In studies of turbulence encounters involving DFDR-equipped airliners, SMACK has been applied to determine winds along the flightpath. A partial list of airline turbulence incidents that Ames has analyzed with the NTSB is given in table 7.1. The first seven were encounters with severe clear-air turbulence (CAT) at cruise altitudes. The analyses of the data from these incidents indicate that the aircraft encountered vortex arrays, caused by the breakdown of windshear layers over thunderstorms or mountain ranges (refs. 35, 48, and 49). A typical CAT encounter is illustrated in figure 7.1(a).

A more hazardous type of atmospheric disturbance is a "downburst," which is a strong, concentrated downflow that induces a high-velocity outflow, with imbedded vortices, near the ground. A typical downburst encounter is shown in figure 7.1(b). The last incident listed in table 7.1 resulted in the loss of Delta Airlines Flight 191 at Dallas/Ft. Worth Airport (DFW) on August 2, 1985. In that case, an L-1011 on final approach flew into a downburst and first contacted the ground about one mile short of the runway. Members of the NTSB Performance Group from Ames Research Center investigating the accident analyzed the available flight records in an effort to characterize the downburst phenomenon. Some of the results presented in the NTSB Accident Report (ref. 50) are included in this chapter (see also refs. 51 and 52).

This chapter describes the analysis of airline flight data to determine performance and winds, and illustrates the methodology with the Flight 191 downburst encounter. It is organized as follows: the next section describes the process of merging and synchronizing the flight data; subsequent sections outline the lift-drag performance calculations and wind estimation procedures used in analyzing flightpath turbulence. The results of the DFW downburst accident investigation are presented in the last section.

TABLE 7.1.- AIRLINE TURBULENCE ENCOUNTERS REPORTED TO THE NTSB AND INVESTIGATED AT AMES RESEARCH CENTER

Case	Aircraft	Location	Date
1	DC-10	Hannibal, MO	4/81
2	DC-10	Morton, WY	7/82
3	DC-10	Near Bermuda	10/83
4	L-1011	Offshore SC	11/83
5	DC-10	Calgary, AL	11/75
6	B-747	Over Greenland	1/85
7	B-747	Over Greenland	2/85
8	B-747SP	Offshore CA	2/85
9	L-1011	Dallas/Ft.Worth	8/85

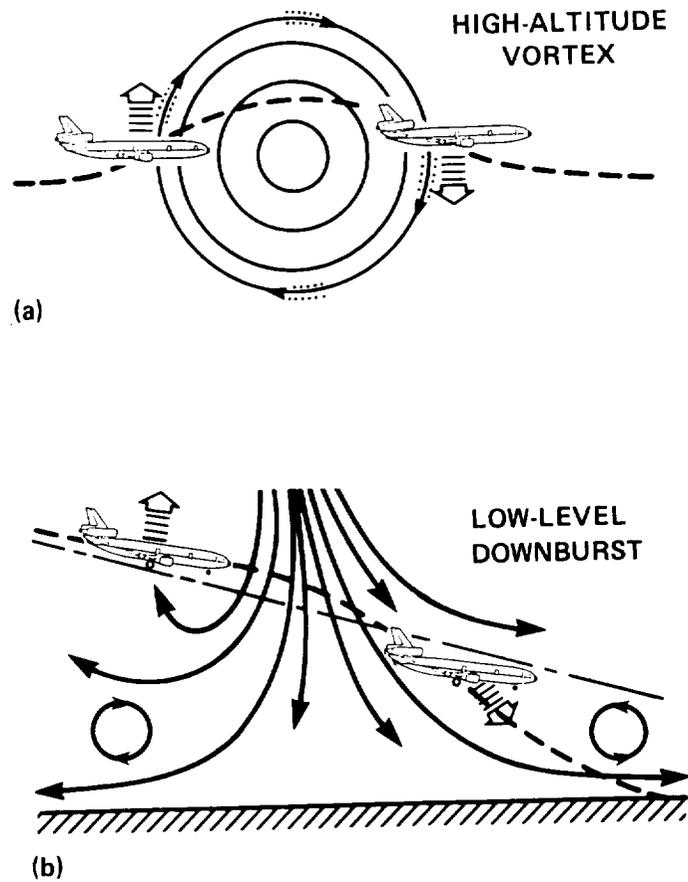


Figure 7.1.- Windshear disturbances. (a) High-altitude vortex, (b) low-level downburst.

Flight Data Processing

The procedure used to determine winds along the flightpath is the same whether the turbulence encountered is associated with vortices at cruise altitude or with a downburst near the ground. In this procedure, which is outlined in figure 7.2, data from the DFDR and ATC radar records are synchronized and merged and the air data are corrected. Performance calculations necessary to synthesize unmeasured time histories of angles of attack and sideslip are made. When a complete data set is in place, the SMACK state-estimation algorithm is employed to estimate the winds. This section describes the processing required to prepare the data set for performance calculations and wind estimation.

Data from a DFDR usually include measurements of accelerations, Euler angles, pressure altitude, airspeed, and other variables, typically sampled at intervals of 0.25-4.0 sec. The important recorder parameters for an L-1011 DFDR data system are shown in table 7.2. The "frame" duration for the system is four seconds; there are

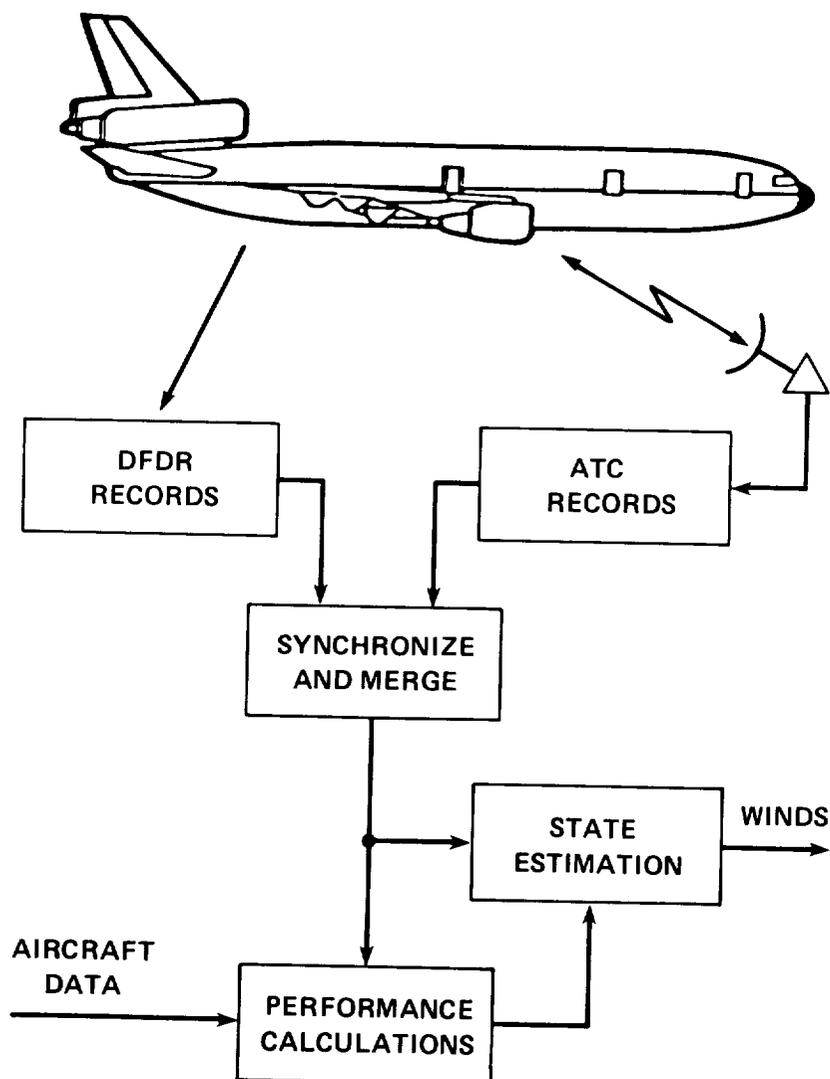


Figure 7.2.- Estimation of flightpath winds from flight records.

TABLE 7.2.- PARAMETERS IN THE L-1011 DFDR SYSTEM

Record	Rate, Hz	Skew	Subframe
Vertical acceleration	4	12	1-4
Lateral acceleration	4	14	1-4
Longitudinal acceleration	4	1	1-4
Roll angle	1	16	1-4
Pitch angle	1	50	1-4
Heading angle	1	2	1-4
Indicated airspeed	1	18	1-4
Angle of attack (l vane)	2	10	1-4
Angle of attack (r vane)	2	24	1-4
Pressure altitude	1	4	1-4
Air temperature	1/2	54	2,4
Stabilator deflection	1	39	1-4
Rudder deflection	2	26	1-4
Thrust (engine 1)	1/4	32	1
Thrust (engine 2)	1/4	32	2
Thrust (engine 3)	1/4	32	3

four "subframes" in each frame, and each subframe has 64 sampling "slots." The column headed by "Rate" defines the basic sampling rate for each parameter; the one headed by "Skew" defines the delay (in 64ths of a second) from the start of a subframe until the parameter is first sampled. The last column specifies the subframe(s) in which the sample appears. For example, a parameter sampled at a rate of 4 Hz with a skew of 14 would occupy slots 14, 30, 46, and 62 in each subframe. However, a parameter sampled at a rate of 0.25 Hz with a skew of 32 would occupy slot 32 in only one subframe of each frame.

The first step in processing the DFDR data is to interpolate each measured parameter at the highest sampling rate (usually 4 Hz) before performing air-data corrections and other calculations. The interpolation is accomplished with a digital filtering algorithm (see appendix E) operating at a rate of 64 Hz, in order to properly accommodate parameter skews. The filter also provides Euler-angle time-derivative estimates for use in computing body angular rates needed for estimating angles of attack and sideslip (or for correcting vane angles). After filtering, each parameter is down-sampled from 64 Hz to the appropriate rate (4 Hz) and the aforementioned calculations are performed.

The second step is to correctly merge the DFDR data with the ATC radar data. Although each data source is time-tagged, there may be an absolute timing error of several seconds on either (or both) of the sources. However, there is usually included with the radar track an independent (transponded) record of the aircraft pressure altitude which can be compared with the DFDR altitude record for time-synchronization of the sources. Since the encoding altimeter for the transponder registers in increments of 100 ft, a fairly large change in altitude is necessary

for synchronizing the sparsely-sampled radar track with the one-Hz DFDR altimeter measurement record.

The last step in data processing prior to performance calculations or wind estimation is to make the usual air-data computations (ref. 29). These include calculation of Mach number, dynamic pressure, true airspeed, and correction of the vane angle measurement for upwash and pitch rate to obtain the angle of attack (when the vane angle is included with the DFDR records). It should be noted that the angle-of-attack time history is essential in determining vertical wind in an investigation of a severe turbulence encounter.

Performance Calculations

The time histories of force coefficients derived from flight data can be quite useful in accident investigations. The lift coefficient can be employed to estimate angle of attack (α) when that record is not among DFDR measurements (ref. 53); a similar procedure is generally used to reconstruct the sideslip angle (β) from a time history of the side-force coefficient. Both lift and drag are used in studies of possible performance degradation which might be caused by heavy rain or ice. This section reviews the use of performance calculations in analyzing turbulence encounters.

Aircraft force coefficients can be expressed in two ways. In the first set, the lift, drag, and side-force coefficients are given in terms of measurements of body-axis accelerations (a_x, a_y, a_z) and thrust components (T_x, T_y, T_z) by

$$C_L = [(ma_x - T_x)\sin\alpha - (ma_z - T_z)\cos\alpha]/QS \quad (7.1)$$

$$C_D = -[F \cos\beta + (ma_y - T_y)\sin\beta]/QS \quad (7.2)$$

$$C_C = [F \sin\beta - (ma_y - T_y)\cos\beta]/QS \quad (7.3)$$

where

$$F = (ma_x - T_x)\cos\alpha + (ma_z - T_z)\sin\alpha$$

where m is aircraft mass, Q is dynamic pressure, and S is wing area. The thrust is determined from tabular data that relate actual thrust to the particular engine parameter recorded. A second set of expressions for the force coefficients is obtained by specifying the corresponding aerodynamic models of the form

$$C_L = C_L(\alpha, M) + \sum_i \ell_i \quad (7.4)$$

$$C_D = C_D(\alpha, M) + \sum_i d_i \quad (7.5)$$

$$C_C = C_C(\beta, M) + \sum_i c_i \quad (7.6)$$

where M is the Mach number. Terms forming the sums in equations (7.4) through (7.6) represent the contributions of angular rates, flaps, spoilers, control surfaces, landing gear, and ground effects. It should be emphasized that, in general, the coefficient models represent all that is known about the aerodynamic properties of the aircraft from theoretical predictions, wind-tunnel experiments, and flight testing.

To estimate an angle-of-attack time history, the lift-coefficient expressions of equations (7.1) and (7.4) are equated, giving a nonlinear algebraic equation to be solved for angle of attack at each time point. An iterative procedure like the Newton-Raphson method works very well for this problem. A similar technique is used with side-force coefficient expressions for estimating the sideslip angle. The coefficient method yields good, "wide-band" estimates of both angles of attack and sideslip (ref. 53). In the analysis of the Flight 191 data described later in the chapter, however, measured flow angles (left and right alpha vanes) were used to derive angle of attack, whereas the coefficient method was used to estimate the angle of sideslip (beta-vane measurements are not included with DFDR records).

Wind Estimation

As illustrated in chapter 4, the SMACK procedure can be used with data from several sources (e.g., flight recorder and ATC radar) to determine the wind pattern along the flightpath of an aircraft. This technique has been useful in the analysis of recent airliner encounters with severe turbulence. To solve the aircraft flightpath wind problem discussed in the next section, one procedure would use SMACK to determine the integrator initial conditions, accelerometer biases, and forcing-function time histories (d_x, d_y, d_h), $(\dot{\phi}, \dot{\theta}, \dot{\psi})$, and (g_x, g_y, g_h) that provide the "best fits" to the measurement records (x, y, h) , (ϕ, θ, ψ) , (V, α, β) , and (a_x, a_y, a_z) (i.e., those that minimize eq. (2.3)). The wind estimates (w_{xy}, w_{hd}, v_{wd}) along the flightpath are obtained as part of the SMACK solution.

A second procedure would fit only the inertial data using SMACK and then calculate the wind components separately from

$$w_x = \dot{x} - V \cos\theta_w \cos\psi_w \quad (7.7)$$

$$w_y = \dot{y} - V \cos\theta_w \sin\psi_w \quad (7.8)$$

$$w_h = \dot{h} - V \sin\theta_w \quad (7.9)$$

where the wind-axis Euler angles (θ_w, ψ_w) are given by

$$\theta_w = \sin^{-1}(\cos\alpha \cos\beta \sin\theta - C \cos\theta) \quad (7.10)$$

$$\psi_w = \psi + \tan^{-1}[(\sin\beta \cos\phi - \sin\alpha \cos\theta \sin\phi)/D] \quad (7.11)$$

where

$$C = \sin\alpha \cos\beta \cos\phi + \sin\beta \sin\phi ; \quad D = \cos\alpha \cos\beta \cos\theta + C \sin\theta$$

Because the air-data variables (V, α , β) derived from the Flight 191 DFDR were relatively "smooth," the second procedure was used to obtain the wind estimates presented in the next section. The coding list for this application is shown in figure 7.3.

The Flight 191 Accident

The Delta Airlines Flight 191 windshear accident occurred during an attempted landing at the Dallas/Ft. Worth Airport. A 14-min portion of the approach path of the L-1011 aircraft as measured by ATC radar is shown in figure 7.4(a). The figure shows the aircraft approaching from the northeast and turning south onto the final glidepath to the runway. The point of initial contact with the ground (shown as a solid circle in fig. 7.4) is at a location 361 ft to the east and 6343 ft to the north of the runway origin. Figure 7.4(b) shows for the same time interval, two curves of pressure altitude that represent the "best fit" of the DFDR record to the transponder record. The time shift required to synchronize the two data sources was one second.

FLIGHT 191 (DFW) INERTIAL SOLUTION						
ENG	6	0	0	0		
REC	1	1200	1	1	0.25	1.5
X	1	1	0	0	0.1	
Y	1	1	0	0	0.1	
H	1	1	0	0	10.	
AX	1	1	1	0	0.005	
AY	1	1	1	0	0.005	
AZ	1	1	1	0	0.009	
DX	2	1	0	0	0.9	
DY	2	1	0	0	1.4	
DH	2	1	0	0	3.4	
PHI	1	1	0	0	0.25	
THI	1	1	0	0	0.25	
PSI	1	1	1	0	0.25	
PH2	2	1	0	0	1.4	
TH2	2	1	0	0	0.7	
PS2	2	1	0	0	0.25	
END	3	0	0	0		

Figure 7.3.- Coding list for the SMACK state-estimation procedure.

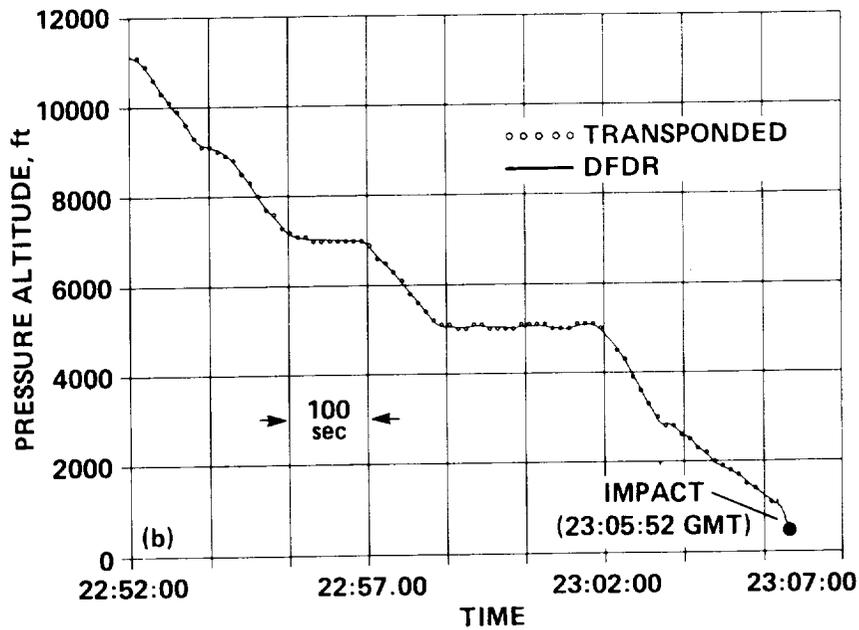
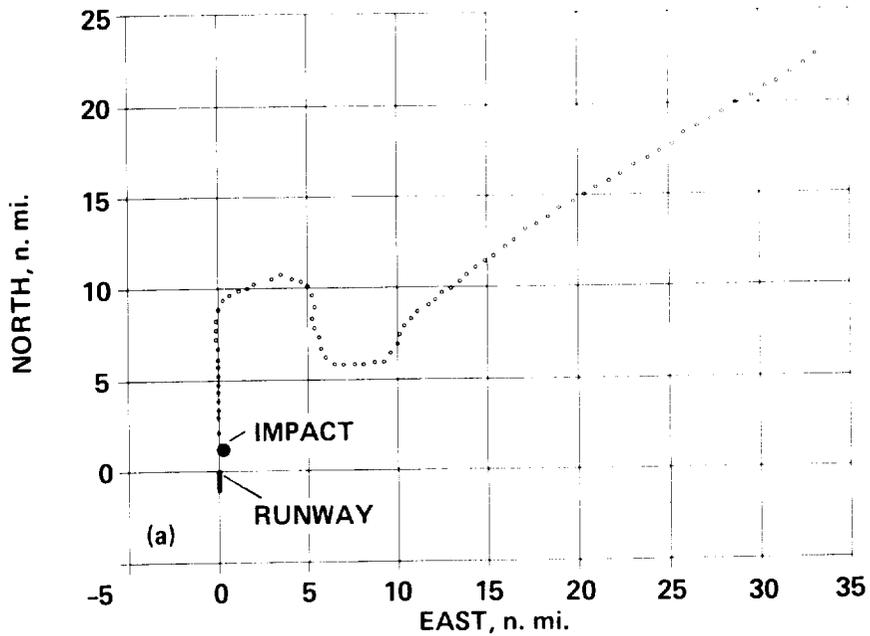


Figure 7.4.- Flight 191 approach. (a) ATC radar groundtrack, (b) transponed and DFDR altitude records compared for time synchronization.

Following data synchronization, and calculation of true airspeed, angle of attack (from the vanes), and sideslip angle (from the performance equations), the state-estimation program SMACK was applied to determine the winds. The kinematic equations were integrated over a five-minute period that starts before the turn onto final approach and ends with the initial ground contact. The fits to the position data for this period are shown in figures 7.5(a) and (b). In these figures the

small circles represent the measured values and the dashed lines represent the time histories generated by SMACK. When the fact that the least count of the ATC tracking data is on the order of 1/8 nautical mile is taken into consideration, there is good agreement between the estimated path and the radar groundtrack in figure 7.5(a). The inertial altitude estimate is compared with the DFDR barometric altitude record in figure 7.5(b). During most of this five-minute interval, there is good agreement between the estimated inertial altitude and the measured barometric altitude. However, during the final portion (in the downburst) there is some discrepancy, apparently due to local pressure variations caused by the atmospheric disturbance.

Because the aircraft was in the downburst for less than a minute before its initial contact with the ground, the rest of the analysis described in this section will cover only the final 60 seconds of flight. Figure 7.6 shows time histories of the three body-axis accelerations, while figure 7.7 shows time histories of the three body-axis Euler angles. Note that the plots of figures 7.6 and 7.7 include the SMACK-derived "best fits" to the DFDR data records. Figure 7.8 shows time histories of the aerodynamic variables (true airspeed, angles of attack and sideslip). The angle of attack was computed from the average value of right and left vanes after correction for upwash and pitch rate. Since vane-rate limiting (at about 19°/sec) occurred during the last 20 sec, the rapid excursions in angle of attack shown in figure 7.8(b) are probably attenuated. As mentioned earlier, the angle of sideslip was computed from the measured side force using predicted aerodynamics and including terms for rudder deflection and yaw rate.

Figure 7.9(a) presents a time history of the aircraft heading angle shown with the groundtrack angle. The value observed for the groundtrack angle at the final time is 174° from true north. This estimate of groundtrack angle at the final time is in agreement with the orientation of the landing gear marks found in the field where the aircraft first contacted the ground. Figure 7.9(b) presents a time history of the true airspeed together with the estimated groundspeed. The groundspeed is seen to be increasing beyond 210 knots at the point of initial contact. Figure 7.9 shows that during the final few seconds there appears to be a tailwind of about 60 ft/sec (35 knots).

The general pattern of the winds can be deduced from figure 7.10, which shows the three components of the wind vector. The horizontal components are shown in figure 7.10(a); the vertical component is shown in figure 7.10(b). Because the vertical wind estimate depends on the angle of attack, the vane rate-limiting mentioned earlier will also attenuate the vertical wind excursions. The results shown in figure 7.10 indicate that the aircraft encountered a strong downflow for a time period of 20 sec followed by a rapid change in vertical wind direction, followed by further changes about 5 sec apart. During the period of major downflow, the aircraft experienced vertical winds on the order of -10 to -40 ft/sec. When the aircraft entered the downflow, the headwind increased from about 20 ft/sec to more than 50 ft/sec. Then, during a period of 26 sec, there was a change to a tailwind of more than 50 ft/sec.

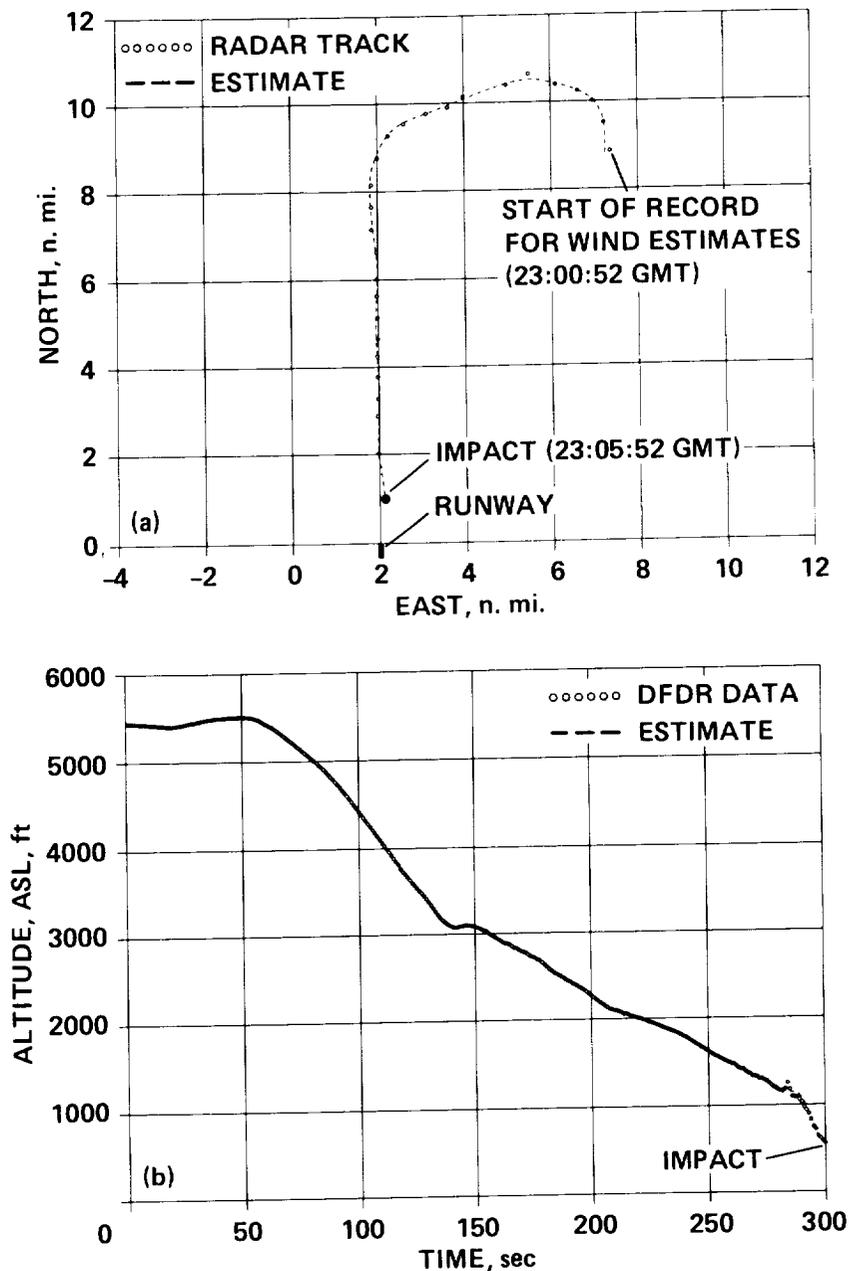


Figure 7.5.- Position solutions (5 min). (a) Groundtrack, (b) pressure altitude.

Figure 7.11 shows winds along the flightpath from different perspectives that clearly indicate the pattern of winds in the downburst. The diagram in figure 7.11(a) shows the flightpath viewed from above with the wind arrows computed from the horizontal components w_x and w_y . These results show the changes in the magnitude and direction of the horizontal wind as the aircraft proceeds through the downburst. As shown by the rotation of the horizontal wind vector, the source of the downflow appears to be located west of the flightpath. The diagram in

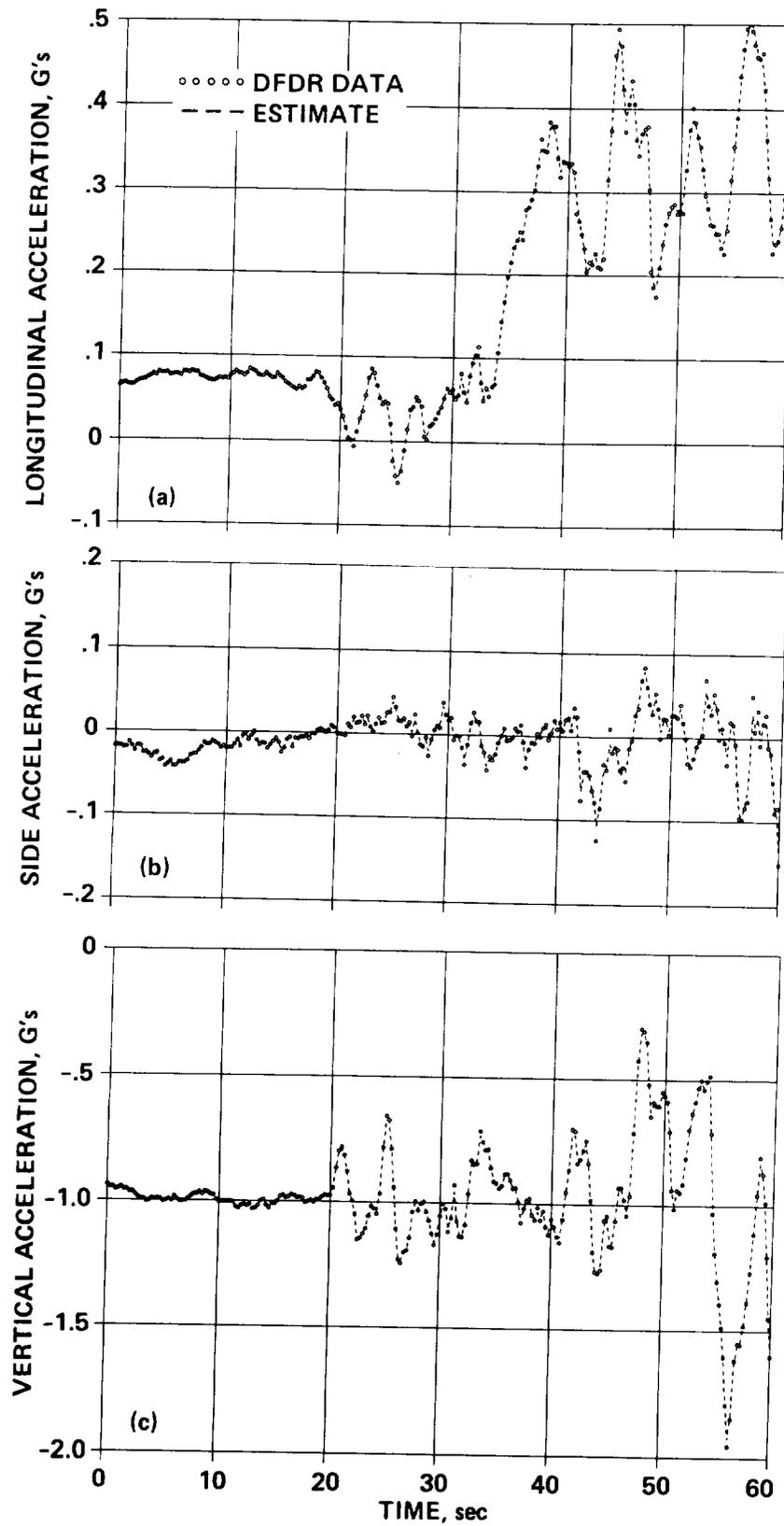


Figure 7.6.- Accelerations (last 60 sec). (a) X-body axis, (b) y-body axis
(c) z-body axis.

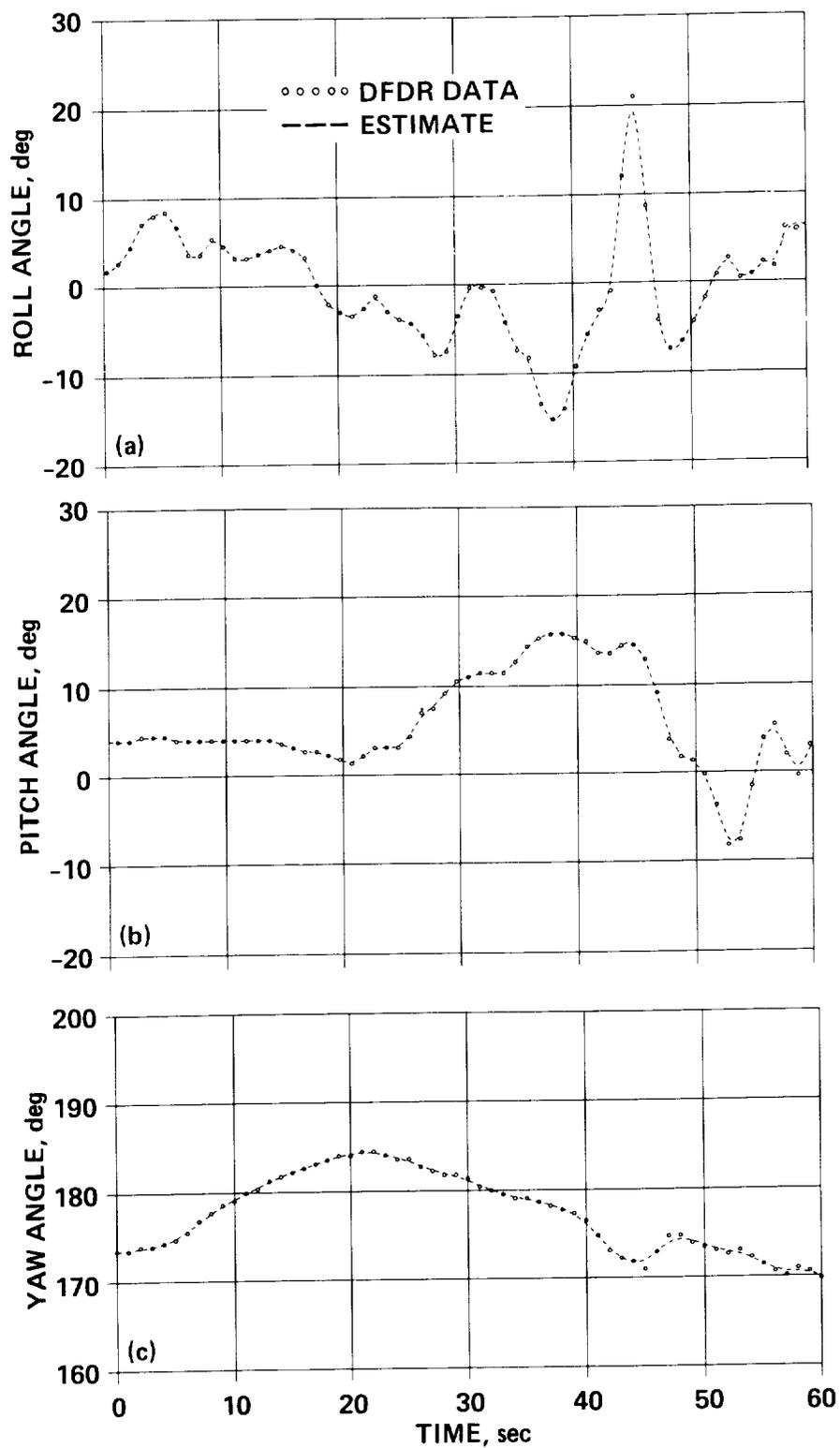


Figure 7.7.- Euler angles (last 60 sec). (a) Roll, (b) pitch, (c) yaw.

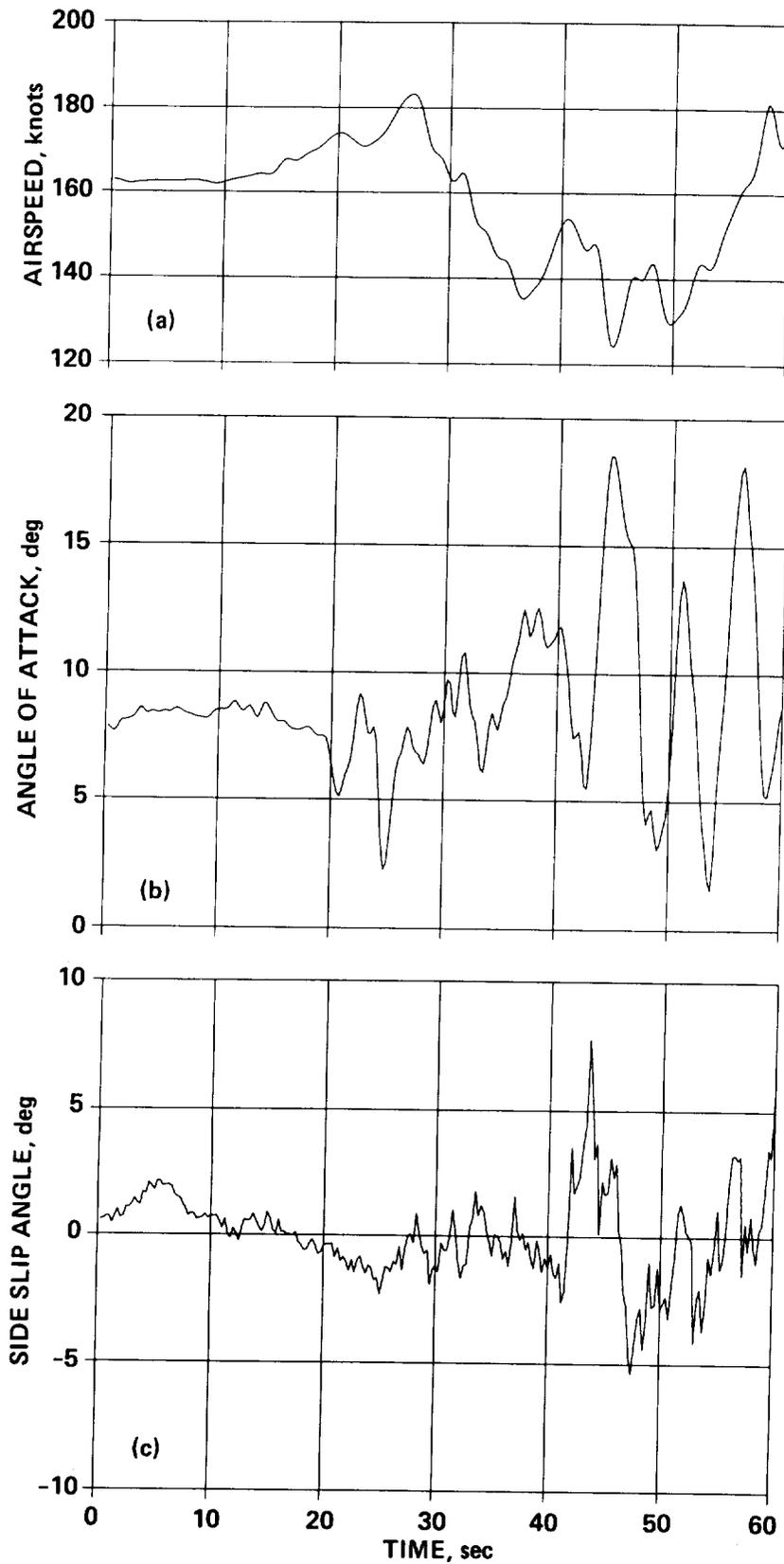


Figure 7.8 Air-data records (last 60 sec). (a) True airspeed, (b) angle of attack, (c) sideslip angle.

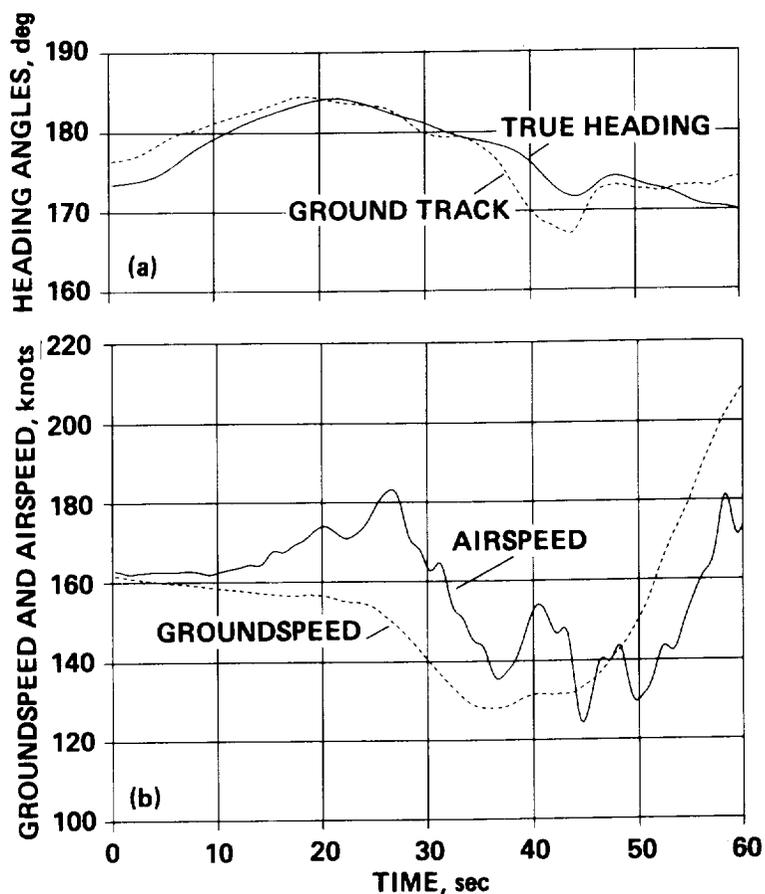


Figure 7.9.- Velocity estimates (last 60 sec). (a) True heading and groundtrack angles, (b) true airspeed and groundspeed.

figure 7.11(b) shows the flightpath viewed from the west with the wind arrows computed from the w_x and w_h components. Following the downflow portion, the outflow near the ground is evident along with changes in the vertical wind. The winds before and after the downflow indicate the presence of vortex rings (ref. 54). According to the vortex-ring model, when a ring impacts the surface, its circulation is spun up, providing a mechanism for the changes in vertical wind that are observed near the ground. In particular, the rapid changes just after the downflow are typical of a series of strong vortices.

The analysis of this accident, which is one of the first involving an aircraft with an onboard digital recording system, provides a detailed look at the pattern of low-level windshear in a downburst environment. The winds derived from the Delta 191 records provide important new information to augment ongoing experiments and theoretical research on the downburst phenomenon. Furthermore, the estimated wind time-histories provide a new set of data that represents a severe downburst for use in simulation and pilot training.

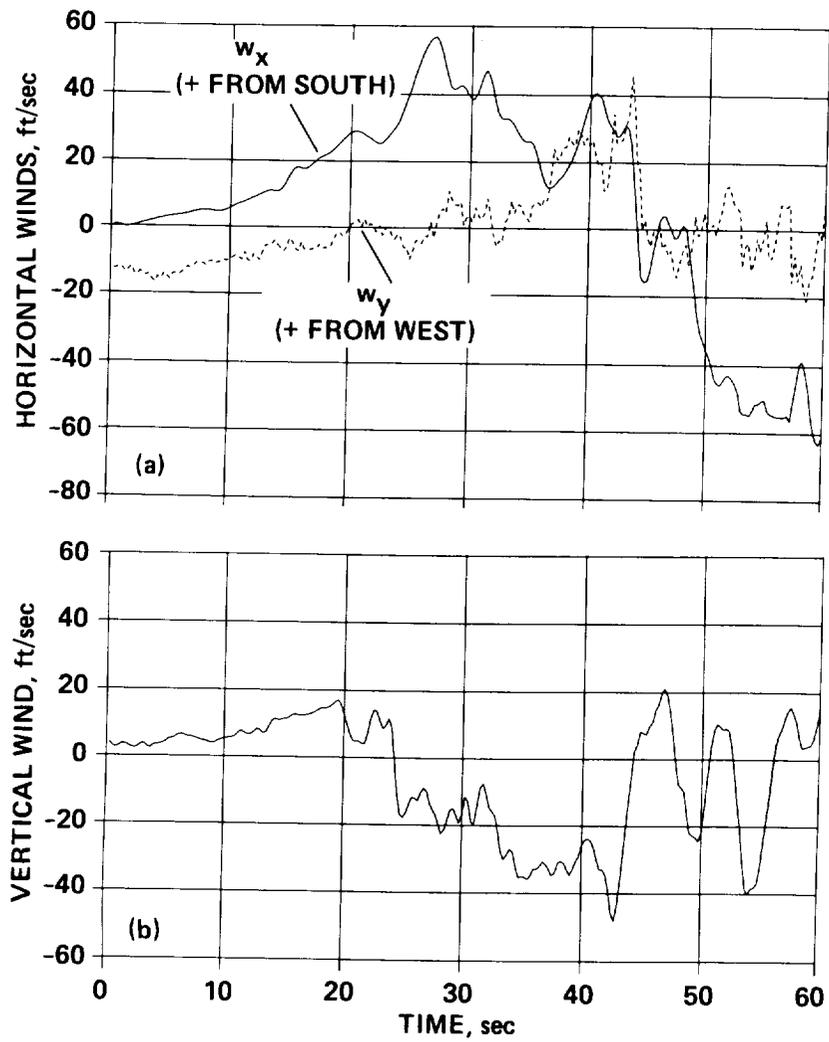


Figure 7.10.- Flightpath wind-component estimates (last 60 sec). (a) Southerly and westerly, (b) vertical.

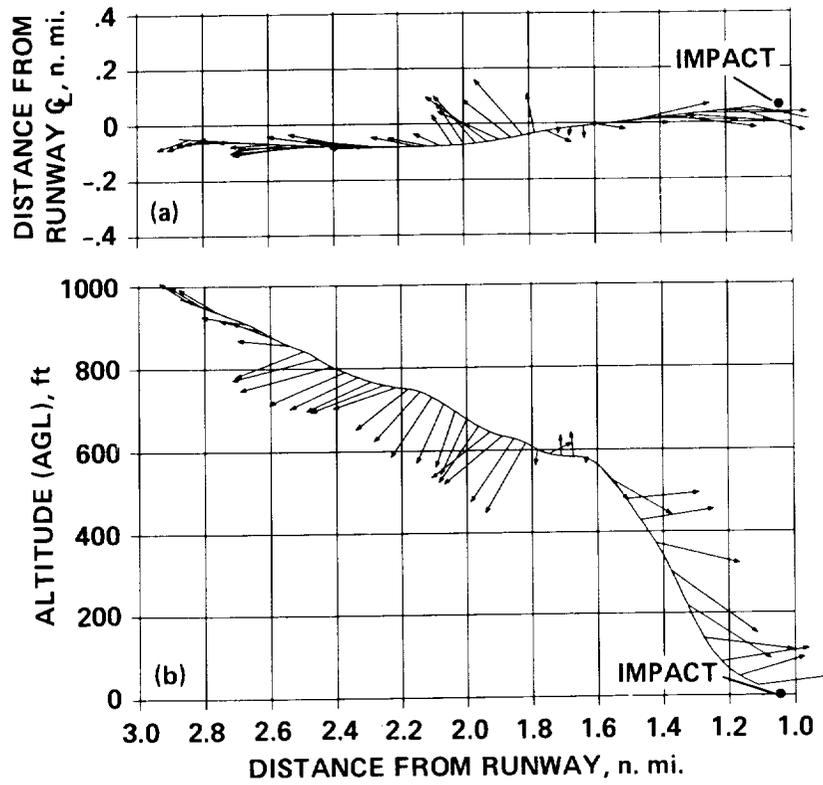


Figure 7.11- Flightpath wind vectors (last 44 sec). (a) Seen from above,
 (b) seen from the west.

8. CONCLUDING REMARKS

In this report the evolution of state-estimation technology for use in the analysis of aircraft flight data has been traced. The underlying mathematics have been reviewed, and a general-purpose aircraft state-estimation program called SMACK has been described. Three recent applications involving the estimation of winds, Euler angles, and air variables have been discussed, and several examples based on the applications have been presented. The examples demonstrate that a general-purpose state estimation program can be used to solve the applications discussed. In one additional example of an application not previously reported, it was shown that inertial position measurements and onboard strap-down measurements can be combined using a state-estimation procedure to provide estimates of Euler angles and air-data variables.

The coding procedure for solving flight-data consistency problems with SMACK has been introduced, and two "real" flight-data applications have been discussed. In the first, a flight-test methodology for identification of an aerodynamic model that includes the use of state estimation was described. In the second, the application of state estimation in the analysis of a windshear accident was presented. It is hoped that the text portion of this report has helped to make the flight-data analyst aware of the potential advantages of using state estimation in solving a variety of problems, and that it and the appendices that follow will serve adequately as a User's Manual for the SMACK computer program.

REFERENCES

1. Gerlach, O. H.: Determination of Performance, Stability and Control Characteristics from Measurements in Nonsteady Maneuvers. Stability and Control, AGARD CP-17, Pt. 1, Sept. 1966, pp. 499-523.
2. Gerlach, O. H.: Determination of Performance and Stability Parameters from Nonsteady Flight Test Maneuvers. National Business Aircraft Meeting of the Society of Automotive Engineers, Wichita, KS, Mar. 1970.
3. Hosman, R. J. A. W.: A Method to Derive Angle of Pitch, Flightpath Angle, and Angle of Attack from Measurements in Nonsteady Flight. Rep. VTH-156, Delft Technological U., The Netherlands, 1971.
4. Mulder, J. A.: Estimation of the Aircraft State in Nonsteady Flight. AGARD CP-172, Chap. 19, Nov. 1974.
5. Wingrove, R. C.: Quasilinearization Technique for Estimating Aircraft States from Flight Data. AIAA Paper 72-965, Palo Alto, CA., 1972; also J. Aircraft, vol. 10, no. 5, 1973, pp. 303-307.
6. Wingrove, R. C.: Parameter Estimation of Powered-Lift STOL Aircraft Characteristics Including Turbulence and Ground Effects. AGARD CP-172, Chap. 28, Nov. 1974.
7. Eulrich, B. J.; and Weingarten, N. D.: Identification and Correlation of the F-4E Stall/Poststall Aerodynamic Stability and Control Characteristics from Existing Test Data. Rep. AK-5126-F-1, Calspan Corp., Buffalo, NY, Nov. 1973; AFFDL-TR-73-125.
8. Molusis, J. A.: Analytical Study to Define a Helicopter Stability Derivative Extraction Method. Vol. 1, Final Report, Contract NAS1-11613, Sikorsky Aircraft, Stratford, CT, 1973.
9. Hall, W. E, Jr.; Gupta, N. K.; and Tyler, J. S.: Model Structure Determination and Parameter Identification for Nonlinear Aerodynamic Flight Regimes, in Methods for Aircraft State and Parameter Identification. AGARD CP-172, Paper 21, 1974.
10. Galbrith, T. J.; Petersen, T. J.; and Roth, E. C.: A Computer System for Identifying Aircraft Characteristics from Flight Test Data. The Boeing Co., Seattle, WA, 1976.
11. McBrinn, D. E.; and Brassell, B. B.: Aerodynamic Parameter Identification for the A-7 Airplane at High Angles of Attack. Proceedings of 3rd Atmospheric Flight Mechanics Conference, Arlington, TX, 1976.

12. Jonkers, H. L.: Application of the Kalman Filter to Flightpath Reconstruction from Flight Test Data Including Estimation of Instrumental Bias Error Corrections. Ph.D. Thesis, Delft Tech. Univ., The Netherlands, 1976.
13. Klein, V.; and Schiess, J. R.: Compatibility Check of Measured Aircraft Responses Using Kinematic Equation and Extended Kalman Filter. NASA TM-D-8514, 1977.
14. Stalford, H. L.; and Ramachandran, S.: Application of the Estimation-Before-Modeling (EBM) System Identification Method to the High Angle of Attack/Sideslip Flight of T-2C Jet Trainer Aircraft. Vol. II, NADC-76097-30, June 1978.
15. Keskar, D. A.; and Klein, V.: Determination of Instrumentation Errors from Measured Data Using Maximum Likelihood Method. AIAA Paper 80-1602, Danvers, MA, 1980.
16. Fratter, C.: Determination of Aerodynamic Coefficients for the Avionics Research Aircraft Using the Estimation-Before-Modeling Technique. M.S.E. Thesis, 1591-T, Princeton Univ., Princeton, NJ, Oct. 1982.
17. Sri-Jayantha, M.; and Stengel, R. F.: Data Acquisition System and Methodology for High Angle of Attack Parameter Estimation. SAE Paper No. 830719, Apr. 1983.
18. Breeman, J. H.; Erkelens, L. J. J.; and Nieuwpoort, A. M. H.: Determination of Performance and Stability Characteristics from Dynamic Manoeuvres with a Transport Aircraft Using Parameter Identification Techniques. AGARD CP-373, Ch. 4, 1984.
19. McReynolds, S. R.; and Bryson, A. E.: A Successive Sweep Method for Solving Optimal Programming Problems. Proceedings of the Joint Automatic Control Conference, Aug. 1965, pp. 551-555.
20. Bach, R. E., Jr.: A Variational Technique for Smoothing Flight-Test and Accident Data. AIAA Paper 80-1601, Danvers, MA, 1982; also J. Aircraft, vol. 19, no. 7, 1982, pp. 546-552.
21. Sage, A. P.; and Melsa, J.L.: System Identification. Chs. 3 and 4, Academic Press, NY, 1971.
22. Bryson, A. E.; and Ho, Y.C.: Applied Optimal Control. Ch. 2, Blaisdell, Waltham, MA, 1969.
23. Cox, H.: Recursive Nonlinear Filtering. Proceedings of the National Electronics Conference, vol. 21, 1965.

24. Gelb, A.: Applied Optimal Estimation. Ch. 6, M.I.T. Press, Cambridge, MA, 1974.
25. Hansen, R. S.: DEKFIS User's Guide -- Discrete Extended Kalman Filter/Smoother Program for Aircraft and Rotorcraft Data Consistency. NASA CR-159081, 1979.
26. Bach, R. E., Jr.: Variational Algorithms for Nonlinear Smoothing Applications. NASA TM-73211, 1977.
27. Bach, R. E., Jr.: A Mathematical Model for Efficient Estimation of Aircraft Motions. Sixth IFAC Symposium on Identification and System Parameter Estimation, Washington, DC, June 1982.
28. Brunovsky, P.: A Classification of Linear Controllable Systems. Kibernetika (Praka), vol. 6, 1970, pp. 1973-1987.
29. Dunlap, E. W.; and Porter, M. B.: Theory of the Measurement and Standardization of In-Flight Performance of Aircraft. AFFTC TD-71-1, Apr. 1971.
30. Etkin, B.: Dynamics of Atmospheric Flight. John Wiley and Sons, Inc., New York, 1972.
31. Gainer, T. G.; and Hoffman, S.: Summary of Transformation Equations and Equations of Motion Used in Free-Flight and Wind-Tunnel Data Reduction and Analysis. NASA SP-3070, 1972.
32. Bach, R. E., Jr.; and Wingrove, R.C.: Applications of State Estimation in Aircraft Flight-Data Analysis. AIAA J. Aircraft, vol. 22, no. 7, 1985, pp. 547-554.
33. Wingrove, R. C.; and Bach, R.E., Jr.: Analysis of General Aviation Accidents Using ATC Radar Records. AIAA Paper 82-1310, 1982; also J. Aircraft, vol. 20, no. 10, 1983, pp. 872-876.
34. Wingrove, R. C.; Bach, R. E., Jr.; and Parks, E. K.: Aircraft Motion Analysis Using Limited Flight and Radar Data. Tenth Annual Symposium of the Society of Flight Test Engineers, Las Vegas, NV, Sept. 1979; also SFTE Journal, vol. 2, May 1980, pp. 2-18.
35. Parks, E. K.; Bach, R. E., Jr.; and Wingrove, R.C.: Analysis of the Nature and Cause of Turbulence Upset Using Airline Flight Records. Symposium of the Society of Flight Test Engineers, New York, NY, Sept. 1982.
36. Taylor, L. W., Jr.: Applications of Parameter Estimation in the Study of Spinning Airplanes. AIAA Paper 82-1309, 1982.
37. Gupta, N. K.; and Iliff, K. W.: Identification of Aerodynamic Indicial Functions Using Flight Data. AIAA Paper 82-1375, 1982.

38. McNally, B. D.: Full-Envelope Aerodynamic Modelling of the Harrier Aircraft. NASA TM-88376, Oct. 1986.
39. McNally, B. D.; and Bach, R. E., Jr.: Flight Testing a V/STOL Aircraft to Identify a Full-Envelope Aerodynamic Model. NASA TM-100996, May 1988.
40. De Boor, C.: A Practical Guide to Splines. Applied Mathematical Sciences, vol. 27, Springer-Verlag, Inc., New York, 1978.
41. Anderson, L. C.: AV-8B System Identification Results: Lateral-Directional Model, Ground Effects, and High Angle-of-Attack Model. Systems Control Technology, Inc., June 1986.
42. Anderson, L. C.; and Hansen, R. S.: Review of AV-8B Aerodynamic Model Identification. SCT Report (ARC PO A46963C), Oct. 1987.
43. Draper, N. R.; and Smith, H.: Applied Regression Analysis. John Wiley and Sons, Inc., New York, 1981, ch. 6.
44. Klein, V.; and Batterson, J. G.: Determination of Airplane Model Structure from Flight Data Using Splines and Stepwise Regression. NASA TP-2126, Mar. 1983.
45. Klein, V.; and Batterson, J. G.: Aerodynamic Parameters Estimated from Flight and Wind-Tunnel Data. AIAA J. Aircraft, vol. 23, no. 4, 1986, pp. 306-312.
46. YAV-8B Simulation and Modelling, Volume 1: Aircraft Description and Program Summary. NASA CR-170397, Mar. 1983.
47. Bach, R. E., Jr.; and Wingrove, R. C.: Equations for Determining Aircraft Motions from Accident Data. NASA TM-78609, 1980.
48. Parks, E. K.; Wingrove, R. C.; Bach, R. E., Jr.; and Mehta, R. S.: Identification of Vortex-Induced Clear-Air Turbulence Using Airline Flight Records. AIAA Paper 84-0270, Reno, NV, Jan. 1984; also J. Aircraft, vol. 22, no. 2, Feb. 1985, pp. 124-129.
49. Mehta, R. S.: Modelling Clear-Air Turbulence with Vortices using Parameter-Identification Techniques. AIAA Paper 84-2083, Seattle, WA, Aug. 1984; also Journal of Guidance, Control, and Dynamics, vol. 10, no. 1, Jan.-Feb. 1987, pp. 27-31.
50. National Transportation Safety Board. Delta Airlines, Inc., Lockheed L-1011-385-1, N726DA, Dallas/Ft.Worth International Airport, Texas, Aug. 2, 1985. NTSB/AAR-86/05, Washington, DC, 1986.
51. Bray, R. S.: Aircraft Performance in Downburst Windshear. SAE SP-681-Windshear, 1986, pp. 13-37.

52. Bach, R. E., Jr.; and Wingrove, R. C.: The Analysis of Airline Flight Records for Winds and Performance with Application to the Delta 191 Accident. AIAA Paper 86-2227-CP, Williamsburg, VA, Aug. 1986; also J. Aircraft, vol. 26, no. 2, Feb. 1989, pp. 103-109.
53. Bach, R. E., Jr.; and Parks, E. K.: Angle-of-Attack Estimation for Analysis of Windshear Encounters. Sixteenth Annual Symposium of The Society of Flight Test Engineers, Seattle, WA, Aug. 1985; also J. Aircraft, vol. 24, no. 11, Nov. 1987, pp. 789-792.
54. Schultz, T. A.: A Multiple Vortex-Ring Model of the DFW Microburst. AIAA Paper 88-0685, Reno, NV, Jan. 1988.

APPENDIX A

PROGRAM DESCRIPTION

This appendix presents an overview of SMACK which is intended to aid the user in understanding program structure and flow. The program is written entirely in FORTRAN 77 and consists of approximately 8000 lines of code (5600 statements). The main program and two of the most important subroutines will be described in some detail here. However, every routine used by SMACK will at the least be mentioned, and its place in the calling hierarchy will be described. For ease of reference, the calling hierarchy is shown in figure A1. Further information on program implementation will be found in appendix B; output listings for test problems are included in appendices C and D, and instructions for preparing the user subroutine DATA are presented in appendix E.

The Main Program (SMACK)

A block diagram of the main program is shown in figure A2. It calls several subroutines that initialize arrays, read and analyze the problem coding list, acquire a data record, provide a starting trajectory, solve the problem, and print and plot the solution. The subroutines called directly by SMACK are described as follows:

INIT initializes all arrays in the COMMON blocks. Maximum dimensions and values for constants are set in a BLOCK DATA routine.

READ accepts and analyzes the problem coding list, sets solution, estimation, and measurement flags, and determines entries in the Jacobian matrices. The "bookkeeping" chores performed here allow efficient manipulation of sparse matrices later in the program.

ARIN displays the contents of state-variable, forcing-function, and output-variable integer arrays, including the Jacobian matrix structure. The information may be helpful for understanding how SMACK works.

MODL creates an aircraft flight trajectory consisting of a rising, coordinated 180° turn in wind, adds noise (if specified) to the measurement records, and stores these records, along with the "true" time histories of all output variables for later comparison with estimates. The trajectory is used for program testing.

DATA accesses an external flight-data record for analysis by SMACK. The preparation of this subroutine, which must be supplied by the user, is discussed in appendix E.

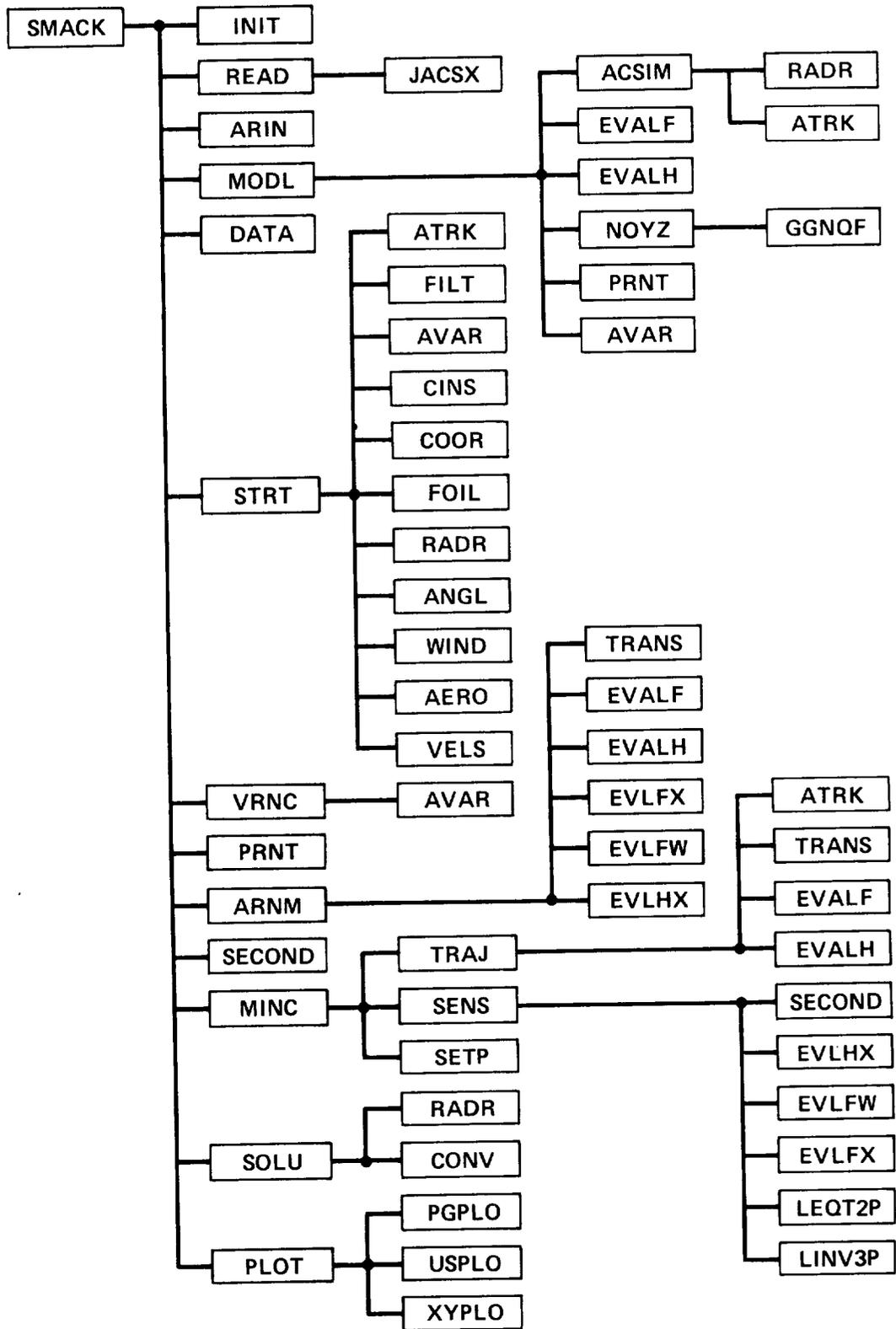


Figure A1.- SMACK subroutine calling hierarchy.

COMMENTS

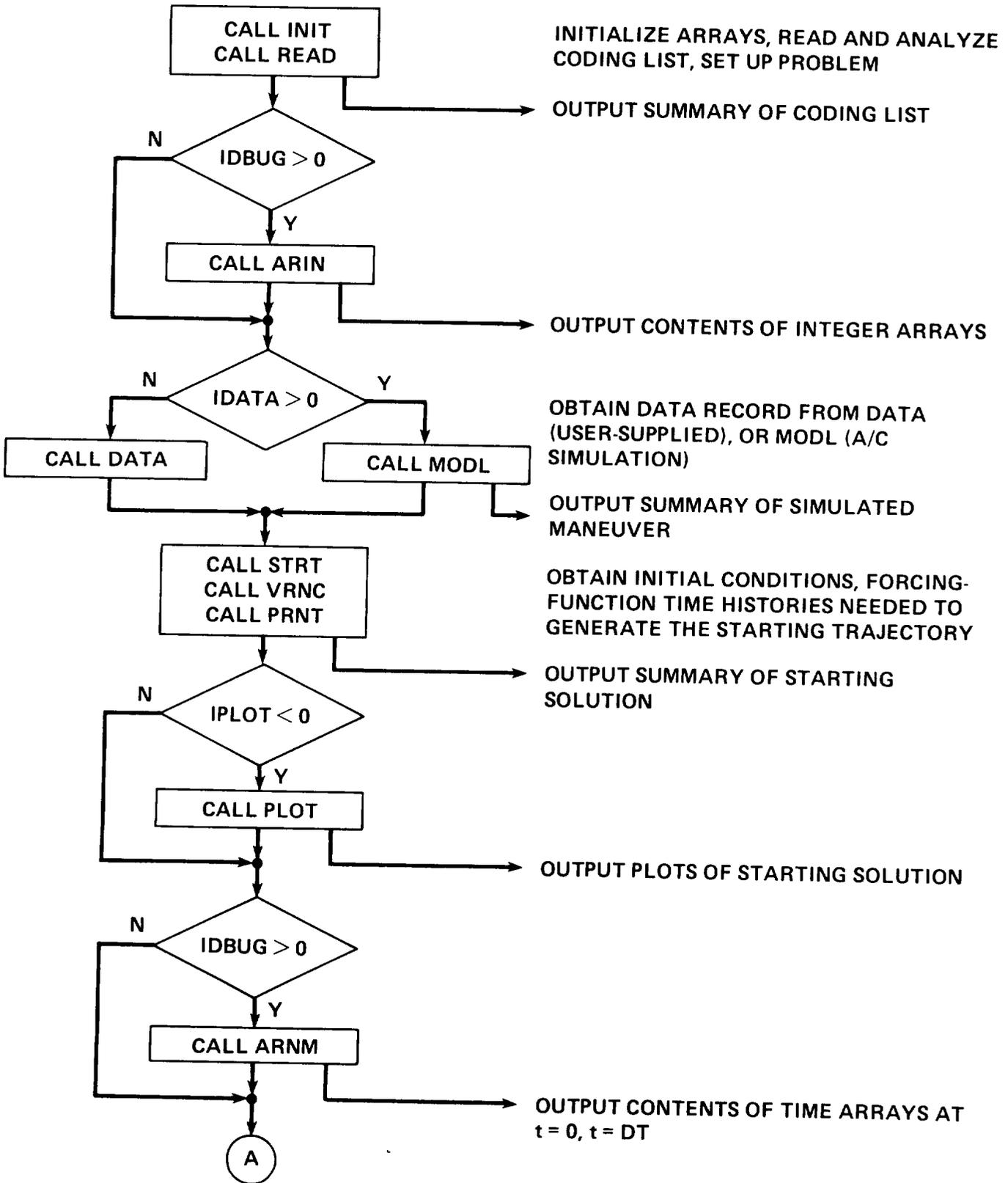


Figure A2.- Flow diagram of SMACK (main program).

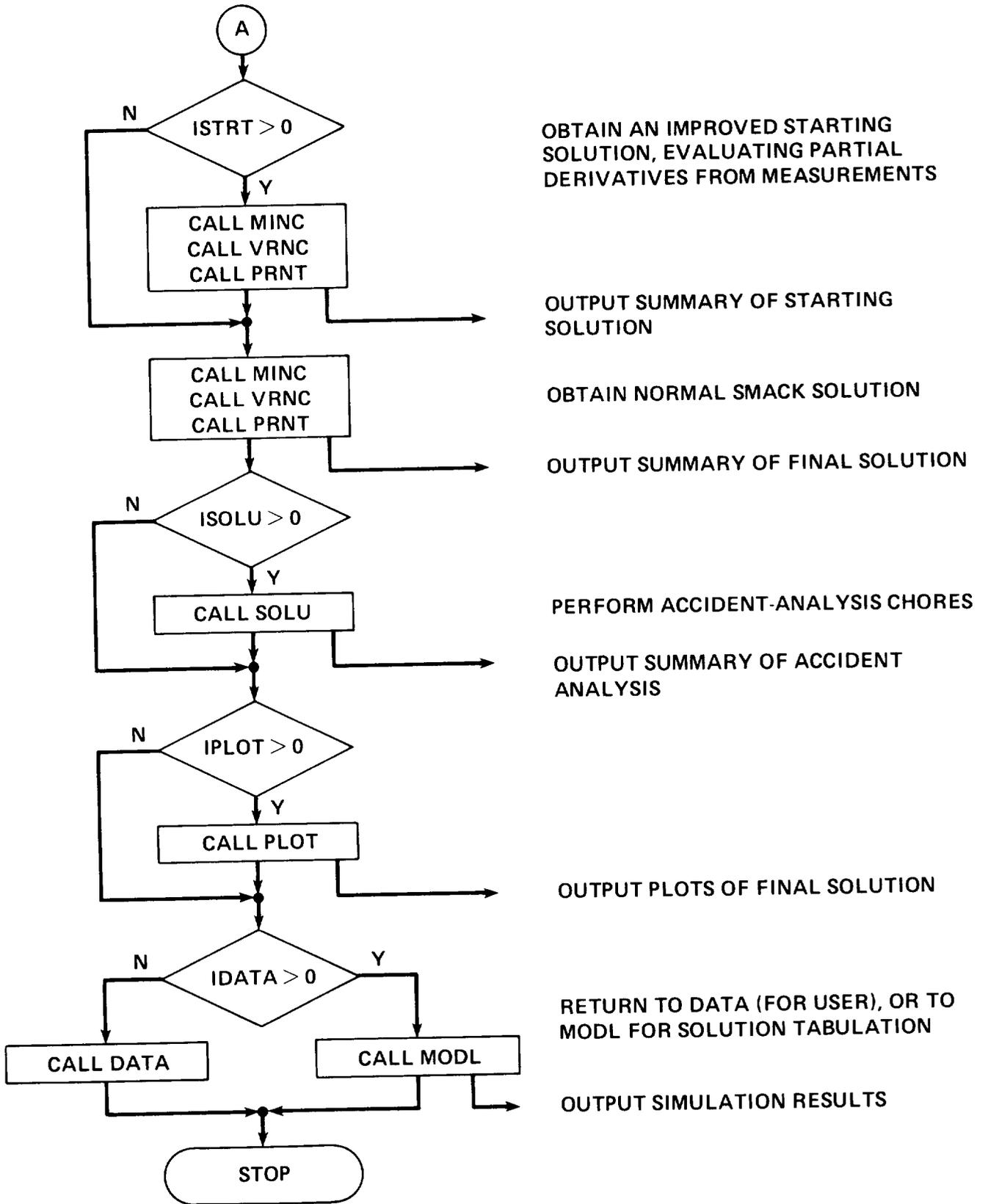


Figure A2.- Concluded.

STRT derives the initial conditions and forcing-function time histories necessary to create the starting nominal trajectory required by the SMACK algorithm. It also determines the diagonal elements of the weighting matrices used in the performance measure.

VRNC calculates and displays mean and rms values of measurement-residual and forcing-function time histories after a solution has been obtained.

PRNT displays values of initial conditions, bias errors and scale factors after a solution has been obtained. This routine is also called by MODL.

ARNM displays the contents of state-variable, forcing-function, and output-variable real arrays, as well as the Jacobian matrices at the first time point calculated. This information may help in problem diagnosis.

SECOND (a Cray utility) determines the CPU time for the computationally-intensive parts of the program. It is included with the VAX versions, to work with the LIB\$INIT TIMER VAX utility.

MINC performs the minimization of the cost function (performance measure), utilizing the backward-filter, forward-smoother algorithm discussed in chapter 2, and outlined in table 2.1.

SOLU completes accident analyses by computing Euler angles from position data (when appropriate) and prints a summary of results, which includes groundspeed, groundtrack, airspeed, flightpath angle, angle of attack, magnetic heading, lift and drag forces.

PLOT generates printer or DISSPLA plots of output-variable time histories.

The several places at which output is produced during problem solution are indicated in the block diagram of figure A2. Notice that results will be printed for the starting solution, for an improved starting solution (if selected), and for the final solution. Plots can be made of either the starting solution or the final solution. The solution option flags ISTRT, ISOLU, IDATA, IPLOT, and IDBUG are specified in the problem coding list covered in chapter 5. The first three of these are set by the solution description statement (ENG or MKS), with

ISTRT = J, ISOLU = K, IDATA = L,

while flags IPLOT and IDBUG are set by the END statement, with

IPLOT = I, IDBUG = J.

The Starting Routine (STRT)

The SMACK program requires a starting trajectory "close enough" to the optimum path to provide reasonable assurance that the solution algorithm will converge. The initial conditions and forcing functions necessary to create a starting trajectory are derived by subroutine STRT. A block diagram for carrying out these important calculations is shown in figure A3. The procedure begins with the filtering of each measurement record, which provides estimates of residual variances for use in the measurement-error weighting matrix R in equation (2.3). Algebraic methods are then used to determine state-variable estimates from the available measurements, from which are obtained the required initial conditions and forcing functions and corresponding variances for elements of the weighting matrix Q in equation (2.3). Subroutines called directly by STRT are the following:

ATRK eliminates all 360° jumps in heading records, thus creating continuous-angle time histories. This routine is also called by TRAJ and ACSIM.

FILT provides digital filtering of a record with estimates of the first and second time derivatives. The filter frequency characteristic is that of a fourth-order, zero phase-shift, low-pass filter (see appendix E).

AVAR computes the sample mean and variance for a time history. This routine is also called by MODL and VRNC.

COOR computes position coordinates from range, bearing, and elevation data and the location of the tracking site.

CINS computes vehicle inertial velocity components from groundspeed and groundtrack data.

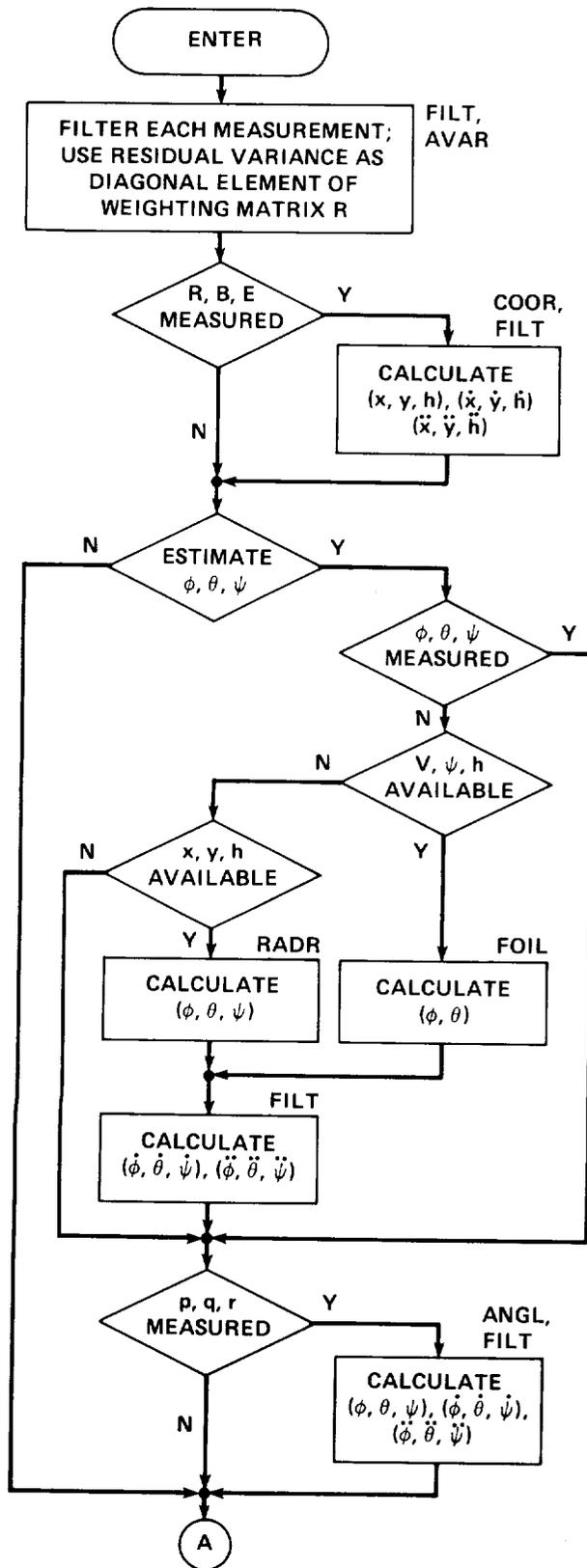
FOIL computes roll and pitch angles from airspeed, heading, altitude, winds, and aircraft performance data (ref. 47).

RADR computes roll, pitch, and yaw angles from radar position, winds, and aircraft performance data (ref. 47). This routine is also called by ACSIM and SOLU.

ANGL computes roll, pitch, and yaw angle time derivatives from rate-gyro data and integrates them to obtain the Euler-angle time histories.

WIND computes wind components in the Earth frame from wind magnitude and direction data.

AERO computes components of vehicle velocity in the Earth frame, with respect to the air mass, from air and Euler-angle data.



COMMENTS

THE FILTERING ROUTINE FITS THE RECORD AND ESTIMATES THE FIRST TWO DERIVATIVES (SEE APPENDIX E)

COORDINATES GIVEN BY

$$x = x_r + R \cos B \cos E$$

$$y = y_r + R \sin B \cos E$$

$$h = h_r + R \cos B \sin E$$

EULER-ANGLE ESTIMATES OBTAINED FROM AIRSPEED, HEADING, ALTITUDE, OR, FROM POSITION DATA [47]

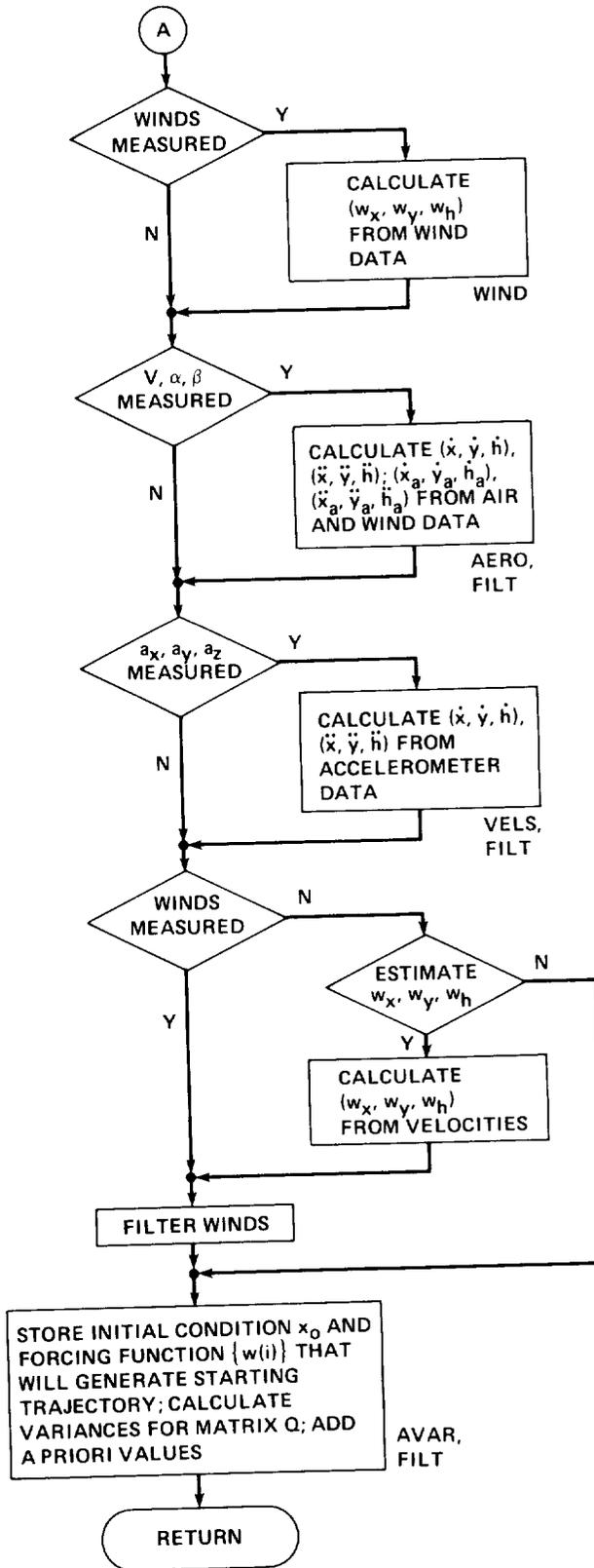
ANGLES DETERMINED BY INTEGRATING ANGULAR RATES FROM

$$\dot{\psi} = (q \sin \phi + r \cos \phi) / \cos \theta$$

$$\dot{\theta} = q \cos \phi - r \sin \phi$$

$$\dot{\phi} = p + \dot{\psi} \sin \theta$$

Figure A3.- Flow diagram for subroutine STRT.



WIND COMPONENTS ARE

$$w_x = -W_{xy} \cos W_{hd}$$

$$w_y = -W_{xy} \sin W_{hd}$$

$$w_h = V_{wd}$$

VELOCITY COMPONENTS ARE

$$u = V \cos \alpha \cos \beta$$

$$v = V \sin \beta$$

$$w = V \sin \alpha \cos \beta$$

$$\begin{bmatrix} \dot{x}_a \\ \dot{y}_a \\ \dot{h}_a \end{bmatrix} = L^t \begin{bmatrix} u \\ v \\ w \end{bmatrix}; \quad \begin{aligned} \dot{x} &= \dot{x}_a + w_x \\ \dot{y} &= \dot{y}_a + w_y \\ \dot{h} &= \dot{h}_a + w_h \end{aligned}$$

VELOCITIES DETERMINED BY INTEGRATING INERTIAL ACCELERATIONS FROM

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{h} \end{bmatrix} = L^t \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$$

Figure A3.- Concluded.

VELS computes components of acceleration in the Earth frame from angle and accelerometer data and integrates them to obtain inertial-velocity time histories.

The Minimizing Routine (MINC)

The subroutine that directs the procedure of minimizing the performance measure is MINC. The function of this subroutine is to carry out the steps of the SMACK algorithm outlined in table 2.1. Its block diagram is shown in figure A4, where it can be seen that when the performance measure (cost) is reduced (or first computed), changes in initial conditions and forcing-function time histories are calculated, and the iteration counter is tested. If the iteration count is less than NIT, an update occurs. If the cost is not reduced by this update, the change is halved and the update is recalculated. If this process succeeds in reducing cost, another iteration is performed; if not, it terminates with a message "NO IMPROVEMENT ON THIS ITERATION". The subroutines called by MINC are described as follows:

TRAJ solves the differential equations of the state model and evaluates the measurement model (see fig. 3.1), and the performance measure. Note that this is the first step of the SMACK algorithm outlined in table 2.1.

SENS solves the backward-information filter, calculates the parameter changes, and solves the forward smoother to determine the forcing-function changes, which are the steps of table 2.1(b). When the solution is complete (IFIN=1), it calculates estimates of parameter standard deviations from the diagonal elements of the information-matrix inverse.

SETP performs the parameter and forcing-function update (the last step outlined in table 2.1).

Other Subroutines

For the rest of the discussion concerning program structure, it will be convenient to refer to the subroutine calling hierarchy shown in figure A1. The subroutines yet to be mentioned are first called by READ, MODL, ARNM, SOLU, and PLOT and will be considered in that order. They are as follows:

JACSX sets up the three Jacobian (partial derivative) arrays FX, FW, and HX according to the problem to be solved.

ACSIM computes the required initial conditions and forcing-function time histories for an aircraft maneuver consisting of a rising, coordinated 180° turn in a wind environment.

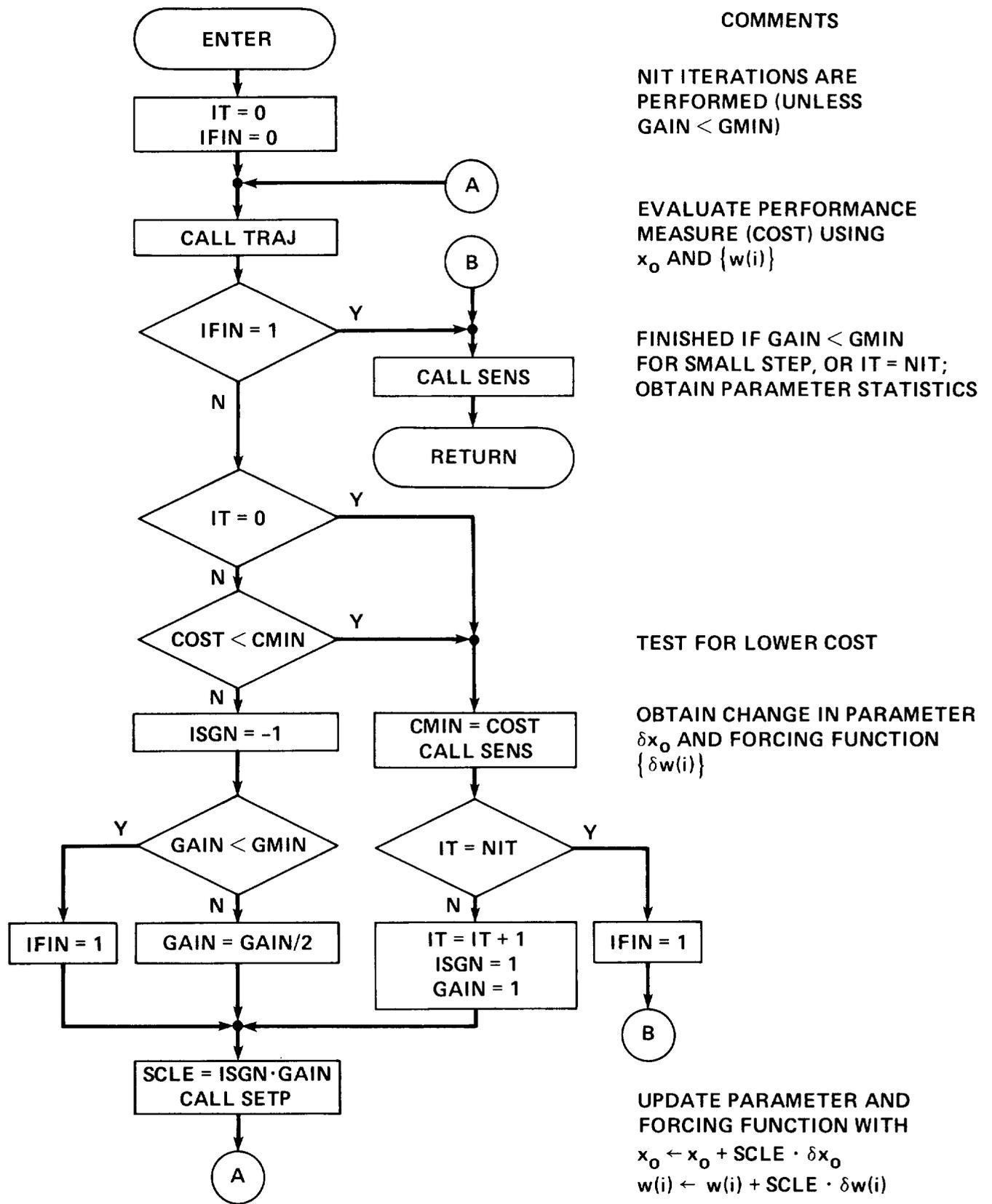


Figure A4.- Flow diagram of subroutine MINC.

EVALF evaluates the state-variable time derivatives (see the state model of eq. (2.1) and fig. 3.1). This routine is called by MODL, ARNM, and TRAJ.

EVALH evaluates the output variables (see the measurement model of eq. (2.2) and fig. 3.1). This routine is called by MODL, ARNM, and TRAJ.

NOYZ creates a sequence of Gaussian random numbers with user-specified variance. The sequence is used in MODL to contaminate simulated measurement records.

GGNQF (An IMSL routine) chooses a number at random from a "Gaussian hat" -- a pseudo-random normal (0, 1) deviate.

TRANS transforms user-specified measurements for use as forcing functions, under an option covered in chapter 5 that is useful in some data-consistency experiments: rate-gyro measurements are transformed to Euler-angle rates (eq. (3.12)); accelerometer measurements are transformed to accelerations in the Earth frame (eq. (3.11)); wind measurements are transformed to components in the Earth frame. This routine is called by ARNM and TRAJ.

EVLFX evaluates active elements of the Jacobian array FX (partial-derivative matrix defined in eq. (2.6)). This routine is called by ARNM and SENS.

EVLFW evaluates active elements of the Jacobian array FW (partial-derivative matrix defined in eq. (2.6)). This routine is called by ARNM and SENS.

EVLHX evaluates active elements of the Jacobian array HX (partial-derivative matrix defined in eq. (2.6).) This routine is called by ARNM and SENS.

LEQT2P (an IMSL routine) computes the solution of the linear equations implied by equation (2.13) to obtain a parameter change DELP. The information matrix EM is stored in symmetric-storage mode. This routine is called each iteration by SENS, when IFIN=0.

LINV3P (an IMSL routine) computes the inverse of the information matrix to obtain estimates of parameter standard deviations. This routine is called by SENS when a solution is complete (IFIN=1).

CONV transforms accelerations from body to wind axes to make calculations of lift and drag forces; computes groundspeed, groundtrack, flightpath

angle, indicated airspeed, and magnetic heading (all variables of interest for accident analysis).

PGPLO produces a printer plot with the time axis running the length of a page. A total of 300 points per variable may be displayed.

USPLO (an IMSL routine) produces an x-y printer plot, one plot per page.

XYPLO produces x-y plot files with DISSPLA for output by an external device (installation-dependent).

APPENDIX B

PROGRAM IMPLEMENTATION

This appendix considers some aspects important to program implementation, such as the computer systems utilized, the structure of COMMON and the way in which calculations involving the Jacobian matrices are performed. Should the user desire to make modifications to the program, an example showing how to add a new output variable is included. Finally, some options are suggested for providing the large temporary storage required in subroutine SENS, since such storage may be machine-dependent.

Computer Systems

SMACK has been implemented at Ames Research Center on the Cray X-MP, Y-MP, and VAX 11/785 and 8650 computers. Running times on the VAX 8650 (double-precision) version have been observed to be about an order of magnitude greater than the Cray for the same problems. The user should note that the Cray word is 64 bits (8 bytes), while the single-precision VAX word is 32 bits (4 bytes). Differences between the Cray and VAX versions are few, and are indicated in the code with the comments "CRAY SPECIFIC" or "VAX SPECIFIC" and "VAX DP". The double-precision VAX version was created by adding to the beginning of each routine the statement

```
IMPLICIT DOUBLE PRECISION (A-H,O-Z) !VAX DP
```

and accessing the double-precision IMSL library.

The program has been designed to be portable. For example, all Hollerith variables are stored four characters per word, a feature compatible with most machines. The SMACK program layout is shown in figure B1, and consists of the COMMON groups, the main program, and the subroutines described in appendix A. Those routines that make use of the IMSL or DISSPLA libraries are indicated. The terms *COMDECK and *DECK are directives to the UPDATE utility, the Cray program "librarian". The directives are considered as comments in the VAX versions of SMACK. It should be noted that the VAX double-precision version has been submitted to COSMIC. Computer Software Management and Information Center, Suite 112, Barrow Hall, The University of Georgia, Athens, GA 30601 (404-542-3265).

COMMON Structure

The COMMON area used by SMACK is divided into three groups, COMTME, COMVAR, and COMNME. The first group, COMTME, shown in table B1, includes all the time-history arrays for the program, and requires 1,350,000 words of memory to store 6000-point records. In block /STM/, array W will contain forcing-function time histories, from

FILE STRUCTURE

COMMON DECKS: *COMDECK COMTME
 *COMDECK COMVAR
 *COMDECK COMMME

MAIN PROGRAM:

 *DECK SMACK: SMACK

SUBROUTINES:

 *DECK SETUP: INIT, READ, JAC SX
 *DECK START: STR T, AERO, ANGL, ATRK, CINS,
 COOR, FOIL, RAD R, VEL S, WIND
 *DECK OPTIM: MINC, SENS, SETP, TRAJ
 *DECK EVLMD: EVALF, EVALH, TRANS
 *DECK EVLJC: EVLFW, EVLFX, EVLHX
 *DECK ACMOD: MODL, ACSIM, NOYZ
 *DECK DEBUG: ARIN, ARNM
 *DECK MISCL: AVAR, CONV, FILT, SECOND (1)
 *DECK PRNTS: PRNT, SOLU, VRNC
 *DECK PLOTS: PLOT, PGPLO, XYPLO (2)

EXTERNALS:

 IMSL LIB (3): GGNQF, LEQT2P, LINV3P, USPLO
 USER CREATED: DATA

PROGRAM NOTES

- (1) SR SECOND A CRAY UTILITY (SUPPLIED FOR VAX VERSION TO BE USED WITH VMS LIB\$INIT_TIMER UTILITY).
- (2) SR XYPLO UTILIZES THE DISSPLA LIBRARY FROM ISSCO, 10505 SORRENTO VALLEY ROAD, SAN DIEGO, CA 92121.
- (3) IMSL INC., 7500 BELLAIRE BLVD., HOUSTON, TX 77036.

Figure B1.- SMACK file structure and program notes.

TABLE B1.- *COMDECK COMTME

```

COMMON/STM/X(21,6000),W(9,6000),DMW(54000)
DIMENSION DW(9,6000)
EQUIVALENCE (DW,DMW)
DIMENSION XA(6000,3),YA(6000,3)
EQUIVALENCE (XA,DMW(1)),(YA,DMW(18001))
COMMON/MSM/Z(30,6000),V(30,6000),KZN(30,6000),
*      H(30,6000),D(30,6000), S(30,6000)
DIMENSION Y(30,6000)
EQUIVALENCE (Y,V)
COMMON/AUX/XN(6000),YN(6000),ZN(6000),FN(6000),
*      KFN(6000),NFN(6000)

```

the list of table 5.2, while X will contain state-variable time histories from the list of table 5.3. The array DW contains the forcing-function updates, while XA and YA are used for plotting. In block /MSM/, arrays Z and KZN are used for storage of measurement time histories and H for output-variable time histories, in the order specified by table 5.1 (see also appendix C). The array V contains residuals for performance measure evaluation, and arrays D and S are used in STRT to store derivative estimates, and again later to store other time histories needed in the SMACK solution. The use of array Y is discussed in appendix E. Included in block /AUX/ are six 6000-word scratch arrays.

The second group, COMVAR, is shown in table B2. It defines the rest of COMMON memory, and requires about 5,760 words. The equivalencing for variable array ED indicated by table B2 is illustrated in figure B2. The array ED has component arrays WD, XD, and HD which are used to hold single time-point values of forcing function W, state X, and output H arrays, respectively. The overlap among the component arrays occurs because six variables (PH2, TH2, PS2) and (X2, Y2, H2) can be either forcing functions or state variables, while another six variables (PHI, THT, PSI) and (X, Y, H) are both state variables and output variables. A similar equivalencing exists for the name array NME and its components (NMW, NMX, NMH), the units array UNE and (UNW, UNX, UNH), the scaling array SCE and (SCW, SCX, SCH), and the index array KE and (KW, KX, KH).

The equivalencing scheme employed for the parameter vector XP in table B2 is also shown in figure B2. Array XP has component arrays X0, WB, VB, and SF which represent the initial conditions (table 5.3), forcing-function means (table 5.2), bias errors, and scale factors (table 5.1), respectively. Although the length of the parameter vector XP is 96, the maximum number of parameters that can be identified is 42. The parameter addresses are stored in array IXP. Notice that there is no overlap among the parameter components. A similar equivalencing exists for the "statistics" vector SXP with components (SX0, SWB, SVB, SSF), and the parameter index vector KXP and its components (KX0, KWB, KVB, KSF).

The third COMMON group COMNME is shown in table B3. This group consists entirely of EQUIVALENCE statements which make possible the use of variable names in

TABLE B2.- *COMDECK COMVAR

COMMON/MOD/ALFO,CLAO,WGLD,VARM
COMMON/DEF/NTITL(12),SKL(8),UNM(12),NIT
COMMON/TIM/DT,NPTS,NINT,NSMP,NSKP,MPTS,FC1
COMMON/SKL/CMK,CMF,CRD,CNL,CMM,CMG
COMMON/CON/RGAS,TSSL,RHOO,PI,C(30)
COMMON/DEC/IH1(6),IH2(6),ISO(6),IS1(6),IS2(6),IIO(6),II1(6)
COMMON/FFN/IW(15),NW,IWU(15),NWU,NUT,WO(9),DWO(9),NWT,
* KIN(15),IIN(15),NIN,WU(15),WP(15)
COMMON/STV/IX(21),NX,NXT,KSX(21),ISX(21),NSX,
* KF(21),IF(21),NF,FD(21),
* KZX(21),IZX(21),NZX,IXZ(21),NXZ
COMMON/OUT/KZ(30),IZ(30),NZ,IH(30),NH,NHT,NST,
* KV(30),IV(30),NV,VD(30),DD(30),SD(30),
* IZH(30),NZH,IHZ(30),NHZ,IZS(15),NZS,
* HO(30),DO(30),SO(30),PLS(30,5),FC(30),NZPT(30)
COMMON/JAC/LFX(72),MFX(72),KFX(72),FX(72),NFX,NFXT,
* LFW(15),MFW(15),KFW(15),FW(15),NFW,NFWT,
* LHX(196),MHX(196),KHX(196),HX(196),NHX,NHXT,
* LHTH(196),MHTH(1089),KPHX(1089),
* MPFW(42,15)MPFX(42,72)
COMMON/VNC/WS(15),WMN(15),WSD(15),QD(15),QI(15),
* VS(30),VMN(30),VSD(30),RD(30),RI(30)
COMMON/PAR/KXP(96),IXP(42),NXP,NXPT,XP(96),
* SXP(96),XPB(96),POI(96),DELP(42),NPAR,
* IWB(15),NWB,NWBT,IVB(30),NVB,NVBT,ISF(30),NSF,NSFT
COMMON/FLG/KANG,KXYH,KRAD,KWND,KAIR,KSPF,KPQR,KMOM,KPOS,KVEL,
* IANG,IXYH,IRAD,IWND,IAIR,ISPF,IPQR,IMOM,IPOS,IVEL,
* LWND,LAIR,LSPF,LPQR,LMOM,IAC,IPS,IVN,IRE,IWD,
* ISOLU,IDATA,IDBUG,ISTRRT,IPLLOT,INOYZ
COMMON/MST/KE(54),IE(54),NET,NME(54),ED(54),
* SCE(54),ISCE(54),UNE(54),IUNE(54)
DIMENSION WD(15),KW(15),SCW(15),UNW(15),NMW(15),
* WB(15),KWB(15),SWB(15),
* XD(21),KX(21),SCX(21),UNX(21),NMX(21),
* XO(21),KXO(21),SXO(21),
* HD(30),KH(30),SCH(30),UNH(30),NMH(30),
* VB(30),KVB(30),SVB(30),SF(30),KSF(30),SSF(30)
EQUIVALENCE (WD,ED(1)),(XD,ED(10)),(HD,ED(25)),
* (KW,KE(1)),(KX,KE(10)),(KH,KE(25)),
* (SCW,SCE(1)),(SCX,SCE(10)),(SCH,SCE(25)),
* (UNW,UNE(1)),(UNX,UNE(10)),(UNH,UNE(25)),
* (NMW,NME(1)),(NMX,NME(10)),(NMH,NME(25))
EQUIVALENCE (XO,XP(1)),(WB,XP(22)),(VB,XP(37)),
* (SF,XP(67)),
* (KXO,KXP(1)),(KWB,KXP(22)),(KVB,KXP(37)),
* (KSF,KXP(67)),
* (SXO,SXP(1)),(SWB,SXP(22)),(SVB,SXP(37)),
* (SSF,SXP(67))

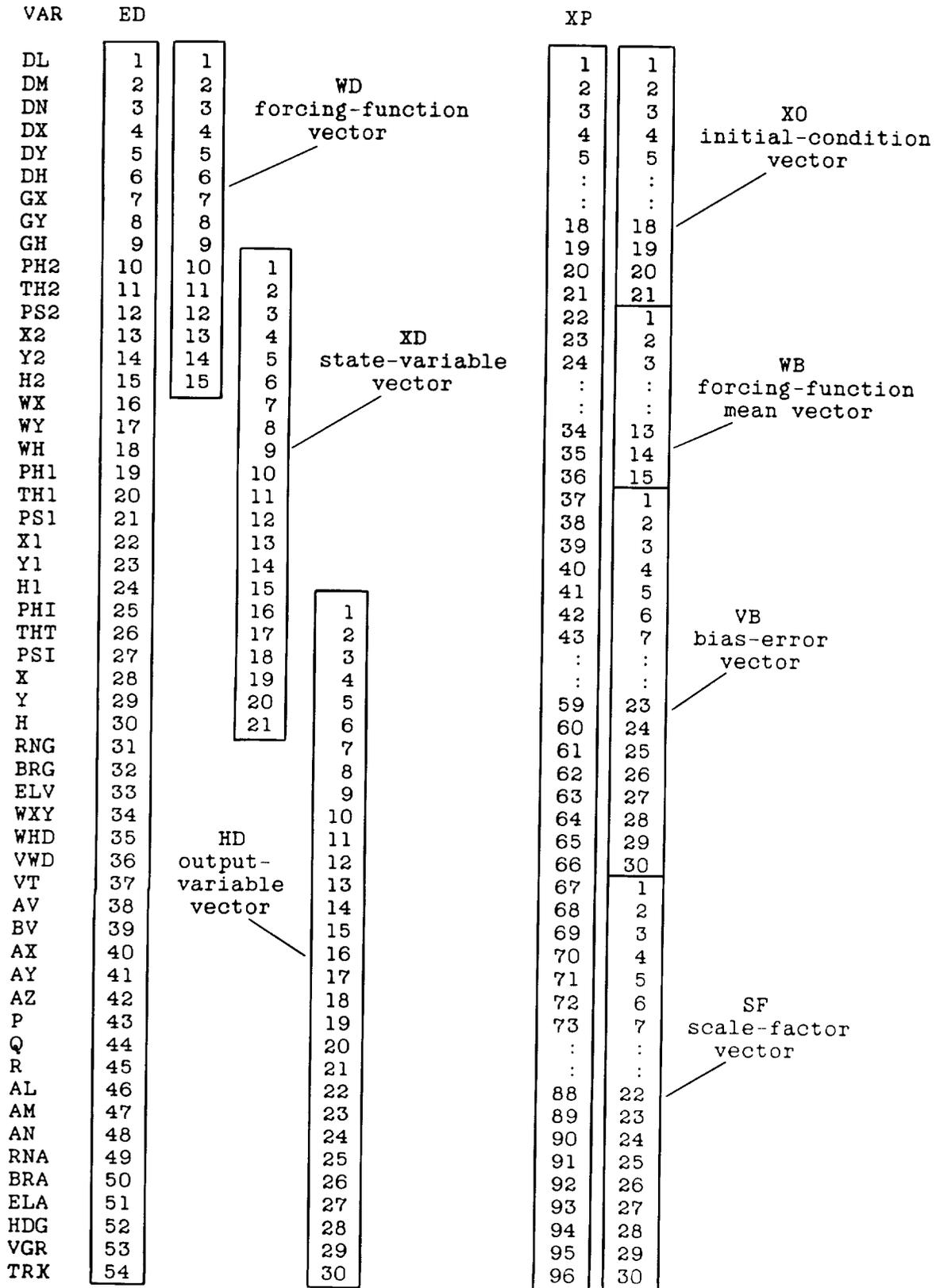


TABLE B3.- *COMDECK COMNME

EQUIVALENCE	(DL ,ED(1)),(DM ,ED(2)),(DN ,ED(3)),
*	(DX ,ED(4)),(DY ,ED(5)),(DH ,ED(6)),
*	(GX ,ED(7)),(GY ,ED(8)),(GH ,ED(9)),
*	(PH2,ED(10)),(TH2,ED(11)),(PS2,ED(12)),
*	(X2 ,ED(13)),(Y2 ,ED(14)),(H2 ,ED(15)),
*	(WX ,ED(16)),(WY ,ED(17)),(WH ,ED(18)),
*	(PH1,ED(19)),(TH1,ED(20)),(PS1,ED(21)),
*	(X1 ,ED(22)),(Y1 ,ED(23)),(H1 ,ED(24)),
*	(PHI,ED(25)),(THT,ED(26)),(PSI,ED(27)),
*	(X ,ED(28)),(Y ,ED(29)),(H ,ED(30)),
*	(RNG,ED(31)),(BRG,ED(32)),(ELV,ED(33)),
*	(WXY,ED(34)),(WHD,ED(35)),(VWD,ED(36)),
*	(VT ,ED(37)),(AV ,ED(38)),(BV ,ED(39)),
*	(AX ,ED(40)),(AY ,ED(41)),(AZ ,ED(42)),
*	(P ,ED(43)),(Q ,ED(44)),(R ,ED(45)),
*	(AL ,ED(46)),(AM ,ED(47)),(AN ,ED(48)),
*	(RNA,ED(49)),(BRA,ED(50)),(ELA,ED(51)),
*	(HDG,ED(52)),(VGR,ED(53)),(TRK,ED(54))
EQUIVALENCE	(PH2D,FD(1)),(TH2D,FD(2)),(PS2D,FD(3)),
*	(X2DT,FD(4)),(Y2DT,FD(5)),(H2DT,FD(6)),
*	(WXDT,FD(7)),(WYDT,FD(8)),(WHDT,FD(9)),
*	(PH1D,FD(10)),(TH1D,FD(11)),(PS1D,FD(12)),
*	(X1DT,FD(13)),(Y1DT,FD(14)),(H1DT,FD(15)),
*	(PHID,FD(16)),(THTD,FD(17)),(PSID,FD(18)),
*	(XDOT,FD(19)),(YDOT,FD(20)),(HDOT,FD(21))
EQUIVALENCE	(UA ,SD(1)),(VA ,SD(2)),(WA ,SD(3)),
*	(AXB ,SD(4)),(AYB ,SD(5)),(AZB ,SD(6)),
*	(UAPS,SD(7)),(VAPS,SD(8)),(WAPS,SD(9)),
*	(UAVN,SD(10)),(VAVN,SD(11)),(WAVN,SD(12))
EQUIVALENCE	(X1A,DD(10)),(Y1A,DD(11)),(H1A,DD(12)),
*	(VT1,DD(13)),(AV1,DD(14)),(BV1,DD(15))
EQUIVALENCE	(XAC,C(1)),(YAC,C(2)),(ZAC,C(3)),
*	(XPS,C(4)),(YPS,C(5)),(ZPS,C(6)),
*	(XVN,C(7)),(YVN,C(8)),(ZVN,C(9)),
*	(RX ,C(10)),(RY ,C(11)),(RH ,C(12)),
*	(RXA,C(13)),(RYA,C(14)),(RHA,C(15))

subroutines that carry out specific model-related calculations (such as in EVALF and EVALH). Note that the array SD holds single time-point values of time histories stored in the S array. An alphabetical listing of all SMACK routines summarizing their COMMON requirements is given in table B4. Notice that each routine that requires COMTME or COMNME also requires COMVAR, but that no routine uses both COMTME and COMNME.

TABLE B4.- SUBROUTINE COMMON REQUIREMENTS

ROUTINE	*DECK	*COMDECK REQUIREMENTS		
		COMTME	COMVAR	COMNME
ACSIM	ACMOD	X	X	
AERO	START		X	X
ANGL	START		X	X
ARIN	DEBUG		X	
ARNM	DEBUG	X	X	
ATRK	START			
AVAR	MISCL			
CINS	START		X	X
CONV	MISCL		X	X
COOR	START		X	X
EVALF	EVLMD		X	X
EVALH	EVLMD		X	X
EVLFW	EVLJC		X	X
EVLFX	EVLJC		X	X
EVLHX	EVLJC		X	X
FILT	MISCL			
FOIL	START		X	X
INIT	SETUP	X	X	
JAC SX	SETUP			
MINC	OPTIM			
MODL	ACMOD	X	X	
NOYZ	ACMOD			
PGPLO	PLOTS			
PLOT	PLOTS	X	X	
PRNT	PRNTS		X	
RADR	START		X	X
READ	SETUP	X	X	
SECOND	MISCL			
SENS	OPTIM	X	X	
SETP	OPTIM	X	X	
SOLU	PRNTS	X	X	
STRT	START	X	X	
TRAJ	OPTIM	X	X	
TRANS	EVLMD		X	X
VELS	START		X	X
VRNC	PRNTS	X	X	
WIND	START		X	X
XPLO	PLOTS			

Jacobian Calculations

Calculations involving the Jacobian matrices defined in chapter 2 are a large part of the computational burden for the SMACK algorithm. The Jacobians are sparse, and special techniques are employed in SMACK to make their use as efficient as possible. If a user should wish to make a change in the state or output model, he will need to know how to modify the corresponding Jacobian arrays. This section

gives the details necessary to understand the Jacobian structure. All calculations for the backward filter and forward smoother are carried out in subroutine SENS; the Jacobian arrays FX, FW, and HX are evaluated in EVLFX, EVLFW, and EVLHX, respectively, which are called by SENS.

The Jacobians and their companion arrays are found in COMMON block /JAC/ (table B2). A description of Jacobian sets (FX, KFX, LFX, MFX), (FW, KFW, LFW, MFW), and (HX, KHx, LHX, MHX) will be illustrated by considering the first set only: the other two sets are similarly structured. The key for the row-column location for any element of FX is given in table B5. The corresponding row index is stored in LFX, the column index in MFX. For example, the fifth element of FX, which corresponds to the fifth partial-derivative expression to be evaluated in EVLFX is, in fact, the partial derivative of Y1DT with respect to Y2. This is seen to be the fourteenth row and fifth column of the Jacobian matrix FX. Hence,

$$LFX(5)=14; \quad MFX(5)=5$$

TABLE B5.- KEY FOR LOCATION OF FX ENTRIES

	P H 2	T H 2	P S 2	X 2	Y 2	H 2	P H 1	T H 1	P S 1	X 1	Y 1	H 1	P H I	T H I	S I	W B 1	W to B 15	V B 16	V B 17	V B 18	V B 19	V B 20	V B 21	S F 16	S F 17	S F 18	S F 19	S F 20	S F 21				
PH2D																13															1		
TH2D																14																2	
PS2D																15																3	
X2DT																16																4	
Y2DT																17																5	
H2DT																18																6	
WXDT																19																7	
WYDT																20																8	
WHDT																21																9	
PH1D	1															22																10	
TH1D		2														23																	11
PS1D			3													24																	12
X1DT				4									46	49	52	25	28	31	34						37	40	43					13	
Y1DT					5								47	50	53	26	29	32	35						38	41	44					14	
H1DT						6							48	51		27	30	33	36						39	42	45					15	
PHID							7						68	71								54	55	58				61	62	65		16	
THID								8					69									56	59					63	66			17	
PSID									9				70	72								57	60					64	67			18	
XDOT										10																						19	
YDOT											11																						20
HDOT												12																					21
	1	2	3	4	5	6	10	11	12	13	14	15	16	17	18	22 - 36	52	53	54	55	56	57	82	83	84	85	86	87					

These entries are verified by inspection of the DATA statements for arrays LFX and MFX found in subroutine EVLFX and repeated here:

```
DATA(LFX(I),I=1,72)/10,11,12,13,14,15,16,17,18,19,20,21, 1, 2,
*      3, 4, 5, 6, 7, 8, 9,10,11,12,13,14,15,13,
*      14,15,13,14,15,13,14,15,13,14,15,13,14,15,
*      13,14,15,13,14,15,13,14,15,13,14,16,16,17,
*      18,16,17,18,16,16,17,18,16,17,18,16,17,18,
*      16,18/
DATA(MFX(I),I=1,72)/ 1, 2, 3, 4, 5, 6,10,11,12,13,14,15,22,23,
*      24,25,26,27,28,29,30,31,32,33,34,35,36,52,
*      52,52,53,53,53,54,54,54,82,82,82,83,83,83,
*      84,84,84,16,16,16,17,17,17,18,18,55,56,56,
*      56,57,57,57,85,86,86,86,87,87,87,16,16,16,
*      17,17/
```

Tables B6 and B7 give the corresponding structural details for the FW and HX Jacobians. The DATA statements from subroutine EVLFW are

```
DATA(LFW(I),I=1,15)/ 1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12,13,14,15/
DATA(MFW(I),I=1,15)/ 1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12,13,14,15/
```

and the DATA statements from subroutine EVLHX are

```
DATA(LHX(I),I=1,136)/ 1, 2, 3, 4, 5, 6, 7, 7, 7, 8, 8, 9, 9, 9,
*      10,10,11,11,12,13,13,13,14,14,14,15,15,15,
*      13,13,13,14,14,14,15,15,15,13,13,13,14,14,
*      14,15,15,15,13,13,13,14,14,14,15,15,15, 0,
*      16,16,17,17,17,18,18,18,16,16,16,17,17,17,
*      18,18,18,16,16,16,17,17,17,18,18,18,16,16,
*      16,17,17,17,18,18,18,19,19,20,20,21,21,19,
*      20,20,21,21,22,22,23,23,24,24,22,22,23,23,
*      23,24,24,24,22,23,23,24,24,25,25,26,26,
*      27,27,27,28,28,28,29,29,30,31/
DATA(MHX(I),I=1,136)/16,17,18,19,20,21,19,20,21,19,20,19,20,21,
*      7, 8, 7, 8, 9, 7, 8, 9, 7, 8, 9, 7, 8, 9,
*      10,11,12,10,11,12,10,11,12,13,14,15,13,14,
*      15,13,14,15,16,17,18,16,17,18,16,17,18, 0,
*      2, 3, 1, 2, 3, 1, 2, 3,10,11,12,10,11,12,
*      10,11,12, 4, 5, 6, 4, 5, 6, 4, 5, 6,16,17,
*      18,16,17,18,16,17,18,10,12,11,12,11,12,17,
*      16,17,16,17, 1, 3, 2, 3, 2, 3,11,12,10,11,
*      12,10,11,12,17,16,17,16,17,19,20,21,19,20,
*      19,20,21,16,17,18,13,14,13,14/
```

Note that location 56 is not used. Data for the VB and SF locations (137-196) of HX are provided in subroutine INIT.

TABLE B6.- KEY FOR LOCATION OF FW ENTRIES

	D L	D M	D N	D X	D Y	D H	G X	G Y	G H	P H	T H	P S	X 2	Y 2	H 2	
PH2T	1															1
TH2D		2														2
PS2D			3													3
X2DT				4												4
Y2DT					5											5
H2DT						6										6
WXDT							7									7
WYDT								8								8
WHDT									9							9
PH1D										10						10
TH1D											11					11
PS1D												12				12
X1DT													13			13
Y1DT														14		14
H1DT															15	15
PHID																
THTD																
PSID																
XDOT																
YDOT																
HDOT																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	

It should be emphasized that the array structures shown in tables B5-B7 are master lists: any particular problem will use subsets of the partial-derivative expressions in subroutines EVLFX, EVLFW, and EVLHX. The selection of those to be included is done in subroutine READ, following the coding list analysis. The index arrays KFX, KFW, and KHx are used for this purpose: the location of a partial derivative to be evaluated is set equal to the array location of FX, FW or HX that it will fill or else the location is set to zero. For example, the fifth partial derivative expression in EVLFX is set up as

```
K=KFX(5)
IF(K.GT.0) FX(K)=1
```

In this way the Jacobian subset in use completely fills the first NFX locations of FX. Corresponding elements of LFX and MFX are modified accordingly. Examples of the Jacobian setup are shown with the output for the test problem given in appendix C.

TABLE B7.- KEY FOR LOCATION OF HX ENTRIES

	P H 2	T H 2	P S 2	X 2	Y 2	H 2	W X	W Y	W H	P H 1	T H 1	P H 1	X 1	Y 1	H 1	P H I	T H I	P H I	X H	Y H	H H	V B 1	V B 30	S F 1	S F 30
PHI																1						137		167	1
THT																2						138		168	2
PSI																3						139		169	3
X																						140		170	4
Y																						141		171	5
H																						142		172	6
RNG																						143		173	7
BRG																						144		174	8
ELV																						145		175	9
WXY							15	16														146		176	10
WHD							17	18														147		177	11
VWD								19														148		178	12
VT							20	21	22	29	30	31	38	39	40	47	48	49				149		179	13
AV							23	24	25	32	33	34	41	42	43	50	51	52				150		180	14
BV							26	27	28	35	36	37	44	45	46	53	54	55				151		181	15
AX		57	58	74	75	76				65	66	67				83	84	85				152		182	16
AY		59	60	61	77	78	79			68	69	70				86	87	88				153		183	17
AZ		62	63	64	80	81	82			71	72	73				89	90	91				154		184	18
P										92	93					98						155		185	19
Q										94	95					99	100					156		186	20
R										96	97					101	102					157		187	21
AL	103	104								109	110					117						158		188	22
AM		105	106							111	112	113				118	119					159		189	23
AN		107	108							114	115	116				120	121					160		190	24
RNA																221	231	24				161		191	25
BRA																25	126					162		192	26
ELA																27	128	129				163		193	27
HDG													130	131	132							164		194	28
VGR													133	134								165		195	29
TRK													135	136								166		196	30
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	37 - 66		67 - 96	

Output Modification

The mathematical representation for an aircraft utilized in SMACK may not be sufficient for every application. Since the measurement model is the most likely area to require modification, an example is considered here that illustrates the addition of a new measurement to the set discussed in chapter 3 and listed in table 5.1. The new measurement is to be the heading from a directional gyro. This instrument is a two-degree-of-freedom gyro, and has its outer gimbal axis parallel to the body vertical axis, with its spin axis aligned with the magnetic North Pole. The heading, as measured by the outer gimbal angle, is a function of the aircraft attitude (roll, pitch, and yaw). The output relation is summarized in table B8, which also includes expressions for partial derivatives of heading with respect to the Euler angles.

TABLE B8.- TWO-DEGREE-OF-FREEDOM DIRECTIONAL GYRO

Output Relation

$$\psi_g = \tan^{-1}[(\cos\phi \sin\psi_m - \sin\phi \sin\theta \cos\psi_m / \cos\theta \cos\psi_m)]$$

where

$$\psi_m = \psi + \Delta\psi$$

and (ϕ, θ, ψ) are the Euler angles, ψ_m is the magnetic heading, and $\Delta\psi$ is the magnetic variation. The user must specify the magnetic variation by setting parameter VARM in subroutine DATA (see appendix E).

Partial Derivatives

$$(\partial\psi_g/\partial\phi) = -\cos\theta \cos\psi_m (\cos\phi \sin\theta \cos\psi_m + \sin\phi \sin\psi_m) / D$$

$$(\partial\psi_g/\partial\theta) = \cos\psi_m (\cos\phi \sin\theta \sin\psi_m - \sin\phi \cos\psi_m) / D$$

$$(\partial\psi_g/\partial\psi) = \cos\phi \cos\theta / D$$

where

$$D = (\cos\phi \sin\psi_m - \sin\phi \sin\theta \cos\psi_m)^2 + (\cos\theta \cos\psi_m)^2$$

The changes made to the program were relatively simple: only COMMON group COMNME and subroutines READ, INIT, STRT, EVALH, and EVLHX required modification. The name HDG was chosen to designate the new variable; it now occupies the formerly unused 28th location of the output vector (element 52 of the variable arrays). This choice required that HDG be equivalenced to ED(52) in COMNME (see table B3), and that the DATA statement for the NME array in subroutine READ include

```
NME(52)=3HHDG
```

and the conversion factor (0.01745) and unit name (DEG) be assigned in the DATA statements for the ISCE and IUNE arrays (in BLOCK DATA, following INIT) by setting

```
ISCE(52)=3 ; IUNE(52)=7
```

Note that these integers point to locations of arrays SKL and UNM (in /DEF/) defined in BLOCK DATA as

```
SKL(3)=0.01745 ; UNT(7)=4HDEG
```

Now, to accommodate the requirements for a starting solution should there be no direct measurement of yaw angle (PSI), there has been inserted in subroutine READ the following code:

```
C      SET UP OUTPUT ARRAY FOR HEADING ESTIMATE

      IF(KH(28).EQ.1) KH(3)=1
```

and in subroutine STRT the code:

```
      IF(KZ(3).NE.1.AND.KZ(28).NE.0) THEN

C      PUT HDG ESTIMATES INTO PSI CHANNELS FOR STARTING SOLUTION

      DO 80 N=1,NPTS
      H(3,N)=H(28,N)
      D(3,N)=D(28,N)
80    CONTINUE
      KZ(3)=-1
      END IF
```

Next, the following statements defining the output relation defined in table B8 were added to subroutine EVALH:

```
      IF(KH(28).GT.0) THEN

C      PERFORM TWO-DOF GYRO CALCULATIONS

      PSM=PSI+VARM
      SNPSM=SIN(PSM)
      CSPSM=COS(PSM)
      A1=CSPHI*SNPSM-SNPHI*SNTHT*CSPSM
      A2=CSTHT*CSPSM
      HDG=ATAN2(A1,A2)

      END IF
```

Finally, the three new elements of the output Jacobian (HX, KHX, LHX, MHX) were assigned to locations 130-132 (see table B7) and provision for evaluation of the partial derivatives from table B8 is accomplished by inserting into subroutine EVLHX the following statements:

```
IF(KH(28).GT.0) THEN
```

```
PSM=PSI+VARM  
SNPSM=SIN(PSM)  
CSPSM=COS(PSM)  
A1=CSPHI*SNPSM-SPHST*CSPSM  
A2=CSTHT*CSPSM  
DSQ=A1**2+A2**2  
CHDG=SF(28)/DSQ
```

```
C PARTIALS OF HDG WRT PHI, THT, PSI
```

```
K=KHX(130)  
IF(K.GT.0) HX(K)=CHDG*(-A2)*(CPHST*CSPSI+SNPHI*SNPSI)  
K=KHX(131)  
IF(K.GT.0) HX(K)=CHDG*CSPSI*(CPHST*SNPSI-SNPHI*CSPSI)  
K=KHX(132)  
IF(K.GT.0) HX(K)=CHDG*CPHCT
```

```
END IF
```

Use of the heading measurement in an accident situation is illustrated by the second problem of appendix D. There, records of airspeed, altitude, heading, and vertical acceleration recovered from an aircraft flight data recorder are combined with radar tracking data to estimate forces and motions along the accident trajectory.

Temporary Storage

When the SMACK algorithm of table 2.1(b) estimates forcing-function time histories, the arrays $d(i)$ and $L(i)$, calculated during the backward-filter pass, must be stored for use during the forward-smoother pass. These steps of the algorithm are performed in subroutine SENS, where $d(i)$, an $NW*1$ vector, and $L(i)$, an $NW*NXP$ matrix are combined in a vector QFWTBP of length $NW*(NXP+1)$. Here i refers to the time point, NW is the number of forcing functions ($NW \leq 9$), and NXP is the number of parameters ($NXP \leq 42$). Hence, a maximum of 387 words must be stored at each of a possible 6000 time points, requiring a total of 2,322,000 words of temporary storage.

The method of acquiring large amounts of high-speed scratch storage is usually machine dependent. Table B9 illustrates three ways of realizing and accessing the temporary storage needed in SENS. The first method shown is probably limited to systems that have virtual memory. It has been used in the VAX versions of SMACK. The second and third methods are suitable for use with either VAX or Cray versions. The storage required is provided on the Cray system by high-speed, solid-state memory. The third method shown in table B9 is the most efficient, and is now used in both Cray and VAX versions of SMACK. The statements that open the scratch

TABLE B9.- REALIZATION OF TEMPORARY STORAGE

Method 1	
	DIMENSION SCRATCH(387,6000)
	DIMENSION QFWTBP(387)
	:
	:
	DO 100 L=1,NWRD
	SCRATCH(L,N)=QFWTBP(L)
100	CONTINUE
	:
	:
	DO 200 L=1,NWRD
	QFWTBP(L)=SCRATCH(L,N)
200	CONTINUE
	:
	:
Method 2	
	DIMENSION QFWTBP(387)
	:
	:
	WRITE(3) (QFWTBP(L),L=1,NWRD)
	:
	:
	BACKSPACE(3)
	READ(3) (QFWTBP(L),L=1,NWRD)
	BACKSPACE(3)
	:
	:
Method 3	
	DIMENSION QFWTBP(387)
	:
	:
	WRITE(3,REC=N) (QFWTBP(L),L=1,NWRD)
	:
	:
	READ(3,REC=N) (QFWTBP(L),L=1,NWRD)
	:
	:

file are found in the main program:

```
C    OPEN SCRATCH FILE USED IN SR SENS (BF-FS)
```

```
    NBYT=8    !CRAY/VAX DP SPECIFIC
```

```
    LREC=NWT*(NXP+1)*NBYT
```

```
    OPEN(UNIT=3,STATUS='SCRATCH',ACCESS='DIRECT',RECL=LREC)
```

where LREC is the record length in bytes, and NBYT is the number of bytes per word. Note that the number of bytes per word is machine dependent, i.e., for the VAX single-precision version, NBYT=4, while for the Cray and VAX double-precision versions, NBYT=8.

APPENDIX C

FLIGHT-TEST PROBLEM

A test problem is presented here to further illustrate the important features of a solution by SMACK, and to provide a reference output listing for checking a new installation of the program. The internal aircraft-trajectory simulation (a rising, coordinated 180° turn in winds) written into subroutine MODL is utilized to provide the data record. The problem solved is typical of a data-consistency analysis following a flight-test experiment (as described in chapter 6). Radar tracking of aircraft position is included in the measurements, thus making it possible to estimate winds along the flightpath. The description of the solution and its output will follow the steps outlined in the block diagram shown in figure A2. The examples of SMACK output given here and in appendix D were run on the Cray system.

A coding list for the test problem is shown in figure C1. In addition to the radar tracking data (RNG, BRG, ELV), there are measurement records available from onboard instruments including accelerometers (AX, AY, AZ), attitude and rate gyros (PHI, THT, PSI) and (P, Q, R), altimeter (H), and aerodynamic angle vanes (AV, BV). The record of dynamic pressure is assumed to have been converted to true airspeed (VT). For this problem, all the data records are to be fit (I=1, J=1 on the VAR statement), except for the rate-gyro measurements, which are to be used as forcing functions (I=1, J=0 on the VAR statement). A number of bias errors and scale factors are to be determined, along with time histories of forcing functions (GX, GY, GH) and (DX, DY, DH). Note that the second group need not be mentioned in the coding list because accelerometer measurements (AX, AY, AZ) are included in the data set. Note also that the low-pass filter cutoff frequency for roll angle (PHI) has been specified to be 0.15 Hz. The nominal value is 0.10 Hz (as calculated from the sampling interval specified on the REC statement). The nominal value will be used to filter all other records. Choice of a value different than nominal is made to obtain a suitably "white" residual for construction of a starting solution. The MKS and END statements indicate that the solution option flags will be set as

```
ISTRT=0, ISOLU=0, IDATA=1, IPLOT=0, IDBUG=1
```

A coding list summary is printed by subroutine READ, as shown in figure C1(b). When READ has finished setting up all the integer arrays, control is transferred to subroutine ARIN, which prints them. The first group of these, the state-variable, forcing-function, and output-variable integer arrays, are shown in figure C2. An interpretation of the output-variable arrays will be given here, because these are directly determined by coding-list entries. The first column of figure C2(c) refers to the order of output variables listed in table 5.1. The second and third columns are the arrays that indicate which variables are to be estimated. The entry in the KH array is one if J=1 in the corresponding VAR statement. The total number of output variables to be estimated is seen to be NH=18. The IH array contains a (compressed) list of addresses for all variables to be

(a) FLIGHT-TEST ANALYSIS

MKS	6			1	
REC	1	90	1	1	1.0
PHI	1	1			0.05
PHI	4				0.15
THT	1	1			0.05
PSI	1	1			0.05
P	1		1	1	0.001
Q	1		1	1	0.001
R	1		1	1	0.001
RNG	1	1			0.005
BRG	1	1			0.05
ELV	1	1			0.05
H	1	1			0.1
VT	1	1			0.1
AV	1	1			0.05
BV	1	1			0.05
AX	1	1	1		0.01
AY	1	1	1		0.01
AZ	1	1	1		0.01
WXY		1			
WHD		1			
VWD		1			
GX	2	1			
GY	2	1			
GH	2	1			
END		1			

(b) SUMMARY OF CODING LIST:

NF - 15 (DIFFERENTIAL EQUATIONS IN STATE MODEL)
NX - 15 (ESTIMATED VARIABLES IN STATE MODEL)
NW - 6 (ESTIMATED FORCING FUNCTIONS)
NH - 18 (ESTIMATED VARIABLES IN OUTPUT MODEL)
NZ - 16 (AVAILABLE MEASUREMENTS)
NV - 13 (OUTPUT ESTIMATES FIT TO MEASUREMENTS)
NZH- 3 (OUTPUTS MEASURED, NOT ESTIMATED)
NHZ- 21 (NH + NZH)
NZX- 3 (STATES MEASURED, NOT ESTIMATED)
NXZ- 18 (NX + NZX)
NXO- 15 (STATE-VARIABLE INITIAL CONDITIONS)
NWB- 0 (FORCING-FUNCTION MEAN VALUES)
NVB- 6 (INSTRUMENT BIAS ERRORS)
NSF- 3 (INSTRUMENT SCALE FACTORS)
NT = 24 (NXO + NWB + NVB + NSF)

Figure C1.- Flight-test problem. (a) Coding list, (b) summary.

estimated. The next two columns, arrays KZ and IZ, similarly identify those variables specified as measured in the coding list (an entry in array KZ is one if I=1 on the VAR statement). The sixth column, array IV, identifies those variables that are both measured and estimated. Hence, for this problem, NV=13 estimates

(a) STATE VARIABLE ARRAYS:

I	KX	IX	KSX	ISX	KXO	KZX	IZX	IXZ
1	0	4	0	4	0	0	10	4
2	0	5	0	5	0	0	11	5
3	0	6	0	6	0	0	12	6
4	1	7	1	7	1	0	0	7
5	1	8	1	8	1	0	0	8
6	1	9	1	9	1	0	0	9
7	1	13	1	10	1	0	0	13
8	1	14	1	11	1	0	0	14
9	1	15	1	12	1	0	0	15
10	0	16	1	13	0	1	0	16
11	0	17	1	14	0	1	0	17
12	0	18	1	15	0	1	0	18
13	1	19	1	16	1	0	0	19
14	1	20	1	17	1	0	0	20
15	1	21	1	18	1	0	0	21
15	1	21	1	18	1	0	0	21
16	1	0	1	19	1	0	0	10
17	1	0	1	20	1	0	0	11
18	1	0	1	21	1	0	0	12
19	1	0	1	0	1	0	0	0
20	1	0	1	0	1	0	0	0
21	1	0	1	0	1	0	0	0

(b) FORCING FUNCTION ARRAYS:

I	KW	IW	IWU	KWB	IWB	KIN	IIN
1	0	4	4	0	0	0	4
2	0	5	5	0	0	0	5
3	0	6	6	0	0	0	6
4	1	7	7	0	0	1	7
5	1	8	8	0	0	1	8
6	1	9	9	0	0	1	9
7	1	0	0	0	0	1	10
8	1	0	0	0	0	1	11
9	1	0	0	0	0	1	12
10	0	0	0	0	0	1	0
11	0	0	0	0	0	1	0
12	0	0	0	0	0	1	0
13	1	0	0	0	0	0	0
14	1	0	0	0	0	0	0
15	1	0	0	0	0	0	0

Figure C2.- Contents of integer arrays for flight-test problem. (a) State variables, (b) forcing functions.

will be "fit" to the measurement records during the solution. The next column (array IZH) lists those variables measured, but not to be estimated (NZH=3) and the following column (array IHZ) lists all output variables mentioned in the coding list

(c) OUTPUT VARIABLE ARRAYS:

I	KH	IH	KZ	IZ	IV	IZH	IHZ	KVB	IVB	KSF	ISF
1	1	1	1	1	1	19	1	0	16	0	19
2	1	2	1	2	2	20	2	0	17	0	20
3	1	3	1	3	3	21	3	0	18	0	21
4	1	4	0	6	6	0	4	0	19	0	0
5	1	5	0	7	7	0	5	0	20	0	0
6	1	6	1	8	8	0	6	0	21	0	0
7	1	7	1	9	9	0	7	0	0	0	0
8	1	8	1	13	13	0	8	0	0	0	0
9	1	9	1	14	14	0	9	0	0	0	0
10	1	10	0	15	15	0	10	0	0	0	0
11	1	11	0	16	16	0	11	0	0	0	0
12	1	12	0	17	17	0	12	0	0	0	0
13	1	13	1	18	18	0	13	0	0	0	0
14	1	14	1	19	0	0	14	0	0	0	0
15	1	15	1	20	0	0	15	0	0	0	0
16	1	16	1	21	0	0	16	1	0	0	0
17	1	17	1	0	0	0	17	1	0	0	0
18	1	18	1	0	0	0	18	1	0	0	0
19	0	0	1	0	0	0	19	1	0	1	0
20	0	0	1	0	0	0	20	1	0	1	0
21	0	0	1	0	0	0	21	1	0	1	0
22	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0

Figure C2.- Concluded. (c) Output variables.

(NHZ=21). Information about the bias-error and scale-factor parameters to be identified (K=1 and/or L=1 on VAR statement) is stored in the arrays shown in the last four columns. Note that NVB=6 and NSF=3 for this problem.

The next section of output displayed by subroutine ARIN is shown in figure C3. There the contents of the arrays defining the structure of the Jacobians FX, FW, and HX are listed. The method for Jacobian representation was discussed earlier in appendix B: the Jacobian matrices are usually quite sparse, and it is convenient to store the nonzero elements of each in a linear array. Here the method is illustrated by considering the representation of HX, the output Jacobian. Hence, although HX in figure C3(c) is referred to as an NV*NXP matrix, it is actually represented in the program by an NHX-element vector. Addresses of the solution-active elements of HX are stored in the linear array KHx, as shown in figure C3(c), where consecutive locations are listed from left to right. Corresponding row and column addresses for each of the NHX active elements of HX are stored in the linear arrays LHX and MHX, also listed from left to right. This information is summarized in the NV*NXP array that represents the actual row-column locations of consecutive

(a) JACOBIAN FX(NF*NT): NF= 15 , NT= 24 , NFX= 25 , NFXT= 72

ARRAY KFX (SPECIFIES ACTIVE ELEMENTS)

```

0 0 0 1 2 3 0 0 0 4 5 6 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 7 8 9 10 11 12 13 14 15 16
17 18 19 20 21 22 23 24 25

```

ARRAY LFX (ROW INDEX)

```

7 8 9 13 14 15 10 10 11 12 10 11 12 10 10 11 12 10 11 12 10
11 12 10 12

```

ARRAY MFX (COL INDEX)

```

1 2 3 7 8 9 19 20 20 20 21 21 21 22 23 23 23 24 24 24 10
10 10 11 11

```

ARRAY FX(L,M) = FX(KFX(K)), KFX(K) > 0

	X	Y	H	W	W	W	X	Y	H	P	T	P	X	Y	H	V	V	V	V	V	V	S	S	S		
	2	2	2	X	Y	H	1	1	1	H	H	S				B	B	B	B	B	B	F	F	F		
										I	T	I				16	17	18	19	20	21	19	20	21		
X2DT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
Y2DT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
H2DT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
WXDT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
WYDT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
WHDT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
X1DT	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
Y1DT	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
H1DT	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
PHID	0	0	0	0	0	0	0	0	0	21	24	0	0	0	0	0	0	0	0	0	0	0	0	0		
THTD	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	0	0	0	0	0	7	8	11	14	15	18
PSID	0	0	0	0	0	0	0	0	0	23	25	0	0	0	0	0	0	0	0	0	9	12	0	16	19	
XDOT	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	10	13	0	17	20	
YDOT	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
HDOT	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Figure C3.- Contents of Jacobian arrays for flight-test problem.
(a) State model FX.

entries of HX. Note that the NXP parameters defining the columns comprise 15 state initial conditions, 6 bias errors, and 3 scale factors. The NV rows of the array correspond to the output variables that are to be fit to corresponding measurement records. Note also that because the rate-gyro measurements are used as forcing functions in this problem, the partial derivatives of P, Q, and R with respect to their respective bias errors and scale factors are associated with the state model, not the output model.

(b) JACOBIAN FW(NF*NW): NF= 15 , NW= 6 , NFW= 6 , NFWT= 15

ARRAY KFW (SPECIFIES ACTIVE ELEMENTS)

0 0 0 1 2 3 4 5 6 0 0 0 0 0 0

ARRAY LFW (ROW INDEX)

1 2 3 4 5 6

ARRAY MFW (COL INDEX)

1 2 3 4 5 6

ARRAY FW(L,M) = FW(KFW(K)), KFW(K) > 0

	D	D	D	G	G	G
	X	Y	H	X	Y	H

X2DT	1	0	0	0	0	0
Y2DT	0	2	0	0	0	0
H2DT	0	0	3	0	0	0
WXDT	0	0	0	4	0	0
WYDT	0	0	0	0	5	0
WHDT	0	0	0	0	0	6
X1DT	0	0	0	0	0	0
Y1DT	0	0	0	0	0	0
H1DT	0	0	0	0	0	0
PHID	0	0	0	0	0	0
THTD	0	0	0	0	0	0
PSID	0	0	0	0	0	0
XDOT	0	0	0	0	0	0
YDOT	0	0	0	0	0	0
HDOT	0	0	0	0	0	0

Figure C3.- Continued. (b) State model FW.

The output summarizing the trajectory for the simulated aircraft maneuver provided by subroutine MODL for the measurement record is shown in figure C4. The initial-condition set for the active elements of the state model is shown first, followed by bias-error and scale-factor values. Next are listed the statistics of random noise added to the output variables to simulate measurement errors. Each value in the STDEV column (specified by VAL1 on the VAR statement) is used to form the corresponding diagonal element of the measurement-error covariance matrix. Note that each measurement is biased by the value shown in the column headed MEAN. The mean and rms values of each unmeasured forcing function are also computed and listed. The rms value for each of these variables is used to form the corresponding diagonal element of the unknown forcing-function covariance matrix. The final data-record information gives the number of points per record, the length of the record,

(c) JACOBIAN HX(NV*NT): NV= 13 , NT= 24 , NHX= 60 , NHXT=192

ARRAY KHx (SPECIFIES ACTIVE ELEMENTS)

1	2	3	0	0	4	5	6	7	8	9	10	11	12	0	0	0	0	0	13	14
15	16	17	18	19	20	21	0	0	0	0	0	0	0	0	0	22	23	24	25	26
27	28	29	30	31	32	33	34	35	36	37	38	39	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	40	41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
58	59	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0																		

ARRAY LHx (ROW INDEX)

1	2	3	4	5	5	5	6	6	7	7	7	8	8	8	9	9	9	10	10	10
8	8	8	9	9	9	10	10	10	8	8	8	9	9	9	10	10	10	11	11	11
12	12	12	13	13	13	11	11	11	12	12	12	13	13	13	11	12	13			

ARRAY MHx (COL INDEX)

10	11	12	15	13	14	15	13	14	13	14	15	4	5	6	4	5	6	4	5	6
7	8	9	7	8	9	7	8	9	10	11	12	10	11	12	10	11	12	1	2	3
1	2	3	1	2	3	10	11	12	10	11	12	10	11	12	16	17	18			

ARRAY HX(L,M) - HX(KHX(K)), KHX(K) > 0

	X	Y	H	W	W	W	X	Y	H	P	T	P	X	Y	H	V	V	V	V	V	V	S	S	S
	2	2	2	X	Y	H	1	1	1	H	H	S				B	B	B	B	B	B	F	F	F
																16	17	18	19	20	21	19	20	21
PHI	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
THT	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	
PSI	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	
H	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	
RNG	0	0	0	0	0	0	0	0	0	0	0	0	5	6	7	0	0	0	0	0	0	0	0	
BRG	0	0	0	0	0	0	0	0	0	0	0	0	8	9	0	0	0	0	0	0	0	0	0	
ELV	0	0	0	0	0	0	0	0	0	0	0	0	10	11	12	0	0	0	0	0	0	0	0	
VT	0	0	0	13	14	15	22	23	24	31	32	33	0	0	0	0	0	0	0	0	0	0	0	
AV	0	0	0	16	17	18	25	26	27	34	35	36	0	0	0	0	0	0	0	0	0	0	0	
BV	0	0	0	19	20	21	28	29	30	37	38	39	0	0	0	0	0	0	0	0	0	0	0	
AX	40	41	42	0	0	0	0	0	0	49	50	51	0	0	0	58	0	0	0	0	0	0	0	
AY	43	44	45	0	0	0	0	0	0	52	53	54	0	0	0	0	59	0	0	0	0	0	0	
AZ	46	47	48	0	0	0	0	0	0	55	56	57	0	0	0	0	0	60	0	0	0	0	0	

Figure C3.- Concluded. (c) Measurement model HX.

and the nominal bandwidth for the low-pass filter used to process the measurements in the STRT subroutine.

DATA FROM AN INTERNAL SIMULATION

THE TRAJECTORY IS AN A/C MANEUVER CONSISTING
OF A RISING COORDINATED 180-DEG TURN IN WIND

INITIAL CONDITIONS:

VARIABLE	VALUE	STDEV
X2, M/S ²	0.5410E-02	+,- 0.0000E+00
Y2, M/S ²	-0.1534E-02	+,- 0.0000E+00
H2, M/S ²	0.3206E-03	+,- 0.0000E+00
WX, M/S	0.2081E+01	+,- 0.0000E+00
WY, M/S	-0.4546E+01	+,- 0.0000E+00
WH, M/S	0.1000E+01	+,- 0.0000E+00
X1, M/S	0.0000E+00	+,- 0.0000E+00
Y1, M/S	0.8500E+02	+,- 0.0000E+00
H1, M/S	0.0000E+00	+,- 0.0000E+00
PHI, DEG	0.0000E+00	+,- 0.0000E+00
THT, DEG	0.1422E+01	+,- 0.0000E+00
PSI, DEG	0.9135E+02	+,- 0.0000E+00
X, NM	-0.9112E+00	+,- 0.0000E+00
Y, NM	-0.5400E+00	+,- 0.0000E+00
H, M	0.1000E+04	+,- 0.0000E+00
PH1, D/S	-0.9415E-01	+,- 0.0000E+00
TH1, D/S	0.2079E-03	+,- 0.0000E+00
PS1, D/S	-0.1163E-01	+,- 0.0000E+00

BIAS ERRORS:

VARIABLE	VALUE	STDEV
AX, G'S	0.0000E+00	+,- 0.0000E+00
AY, G'S	0.0000E+00	+,- 0.0000E+00
AZ, G'S	0.0000E+00	+,- 0.0000E+00
P, D/S	0.0000E+00	+,- 0.0000E+00
Q, D/S	0.0000E+00	+,- 0.0000E+00
R, D/S	0.0000E+00	+,- 0.0000E+00

SCALE FACTORS:

VARIABLE	VALUE	STDEV
P, D/S	0.1000E+01	+,- 0.0000E+00
Q, D/S	0.1000E+01	+,- 0.0000E+00
Q, D/S	0.1000E+01	+,- 0.0000E+00
R, D/S	0.1000E+01	+,- 0.0000E+00

Figure C4.- Output summarizing simulated flight-test maneuver.

After acquiring the data record, SMACK passes program control to subroutine STRT, which determines a set of initial conditions and forcing-function time histories necessary to generate a starting solution for the SMACK algorithm. The block

MEASUREMENT NOISE:

VARIABLE	MEAN	STDEV
PHI, DEG	-0.3289E-02	+,- 0.5000E-01
THT, DEG	-0.2518E-02	+,- 0.5000E-01
PSI, DEG	-0.1502E-02	+,- 0.5000E-01
H, M	0.1099E-01	+,- 0.1000E+00
RNG, NM	-0.1634E-03	+,- 0.5000E-02
BRG, DEG	-0.1458E-02	+,- 0.5000E-01
ELV, DEG	0.6007E-03	+,- 0.5000E-01
VT, KT	0.4505E-02	+,- 0.1000E+00
AV, DEG	-0.1008E-01	+,- 0.5000E-01
BV, DEG	0.5341E-04	+,- 0.5000E-01
AX, G'S	-0.1700E-03	+,- 0.1000E-01
AY, G'S	0.6972E-03	+,- 0.1000E-01
AZ, G'S	0.4576E-03	+,- 0.1000E-01
P, D/S	0.1719E-03	+,- 0.1000E-02
Q, D/S	0.8182E-04	+,- 0.1000E-02
R, D/S	-0.1006E-03	+,- 0.1000E-02

FORCING FUNCTIONS (UNMEASURED):

VARIABLE	MEAN	STDEV
DX, M/S3	-0.1202E-03	+,- 0.1643E+00
DY, M/S3	-0.5255E-11	+,- 0.9321E-01
DH, M/S3	-0.7124E-05	+,- 0.9736E-02
GX, M/S2	-0.1579E-15	+,- 0.8128E-01
GY, M/S2	-0.2368E-15	+,- 0.9604E-01
GH, M/S2	0.7895E-16	+,- 0.4936E-01

DATA RECORD INFORMATION:

NPTS- 90 (LENGTH OF RECORD, POINTS)
 RECL- 0.9000E+02 (LENGTH OF RECORD, SECONDS)
 FREQ- 0.1000E+00 (LP FILTER BANDWIDTH, HZ)

Figure C4.- Concluded.

diagram describing this procedure was shown in figure A3. Generally, each measurement record is filtered with the cutoff frequency set to make the residual appear "white." The residuals are used to determine the diagonal elements of R, the measurement-error weighting matrix. For a solution using simulated data, however, the weights used are the actual measurement-noise covariance values. The filtered records are then used to estimate a set of initial conditions for the trajectory, and to construct a set of forcing-function time histories. The variance of each is used as the appropriate diagonal element of weighting matrix Q, unless the data has been simulated, in which case the "true" value is used. Results of starting solution calculations are shown in figure C5.

STARTING SOLUTION TABULATION

RESPONSE ERRORS AND WEIGHTS:

VARIABLE	MEAN	STDEV	(R**0.5)	FILTER
PHI, DEG	0.6384E-13 +,-	0.4408E-01	0.5000E-01	0.1500E+00
THT, DEG	0.1598E-13 +,-	0.4442E-01	0.5000E-01	0.1000E+00
PSI, DEG	-0.1949E-12 +,-	0.5192E-01	0.5000E-01	0.1000E+00
H, M	-0.4689E-11 +,-	0.8529E-01	0.1000E+00	0.1000E+00
RNG, NM	0.5893E-14 +,-	0.4173E-02	0.5000E-02	0.1000E+00
BRG, DEG	0.2172E-11 +,-	0.5154E-01	0.5000E-01	0.1000E+00
ELV, DEG	-0.1697E-13 +,-	0.4268E-01	0.5000E-01	0.1000E+00
VT, KT	0.9823E-13 +,-	0.1070E+00	0.1000E+00	0.1000E+00
AV, DEG	-0.4468E-13 +,-	0.4081E-01	0.5000E-01	0.1000E+00
BV, DEG	-0.7069E-16 +,-	0.4509E-01	0.5000E-01	0.1000E+00
AX, G'S	0.1540E-15 +,-	0.8203E-02	0.1000E-01	0.1000E+00
AY, G'S	0.0000E+00 +,-	0.8688E-02	0.1000E-01	0.1000E+00
AZ, G'S	0.6440E-14 +,-	0.8911E-02	0.1000E-01	0.1000E+00

FORCING FUNCTIONS AND WEIGHTS:

VARIABLE	MEAN	STDEV	(Q**0.5)
DX, M/S3	0.9814E-03 +,-	0.1639E+00	0.1643E+00
DY, M/S3	-0.3048E-03 +,-	0.9270E-01	0.9321E-01
DH, M/S3	-0.3726E-03 +,-	0.1599E-01	0.9736E-02
GX, M/S2	-0.5356E-01 +,-	0.8816E-01	0.8128E-01
GY, M/S2	-0.1677E-01 +,-	0.1050E+00	0.9604E-01
GH, M/S2	-0.1078E-02 +,-	0.5965E-01	0.4936E-01
PH1, D/S	-0.8461E-02 +,-	0.7375E+00	0.0000E+00
TH1, D/S	0.2532E-01 +,-	0.1357E+00	0.0000E+00
PS1, D/S	-0.2033E+01 +,-	0.9920E+00	0.0000E+00

Figure C5.- Tabulation of starting solution for flight-test problem.

Following the starting solution, the program calls subroutine ARNM, which loads initial state-variable and forcing-function values into variable arrays XD, WD, displays them, and calls EVALF, which evaluates the state derivative FD, and then calls EVLFX and EVLFW, which evaluate the state-model Jacobians FX and FW, and displays the contents of these arrays. The purpose of this output is for debugging and/or general information only. The contents of arrays XD, WD, FD, FX, and FW are shown in figure C6(a), where the element order is from left to right across the page. The variables are referenced in tables 5.2, 5.3, and B4. Units are the internal (mks-radian) units used by SMACK. The first integration step is then performed, and the results used to evaluate the output model and output Jacobian by calling EVALH and EVLHX. The output of arrays HD, SD, and HX is shown in figure C6(b). These variables are referenced in tables 5.1, B3, and B4.

The problem is now ready for solution by the algorithm of table 2.1(b). Subroutine MINC is called to perform this task (see fig. A.4), and its output is shown in figure C7. Convergence properties are indicated by the behavior of COST (the

INITIAL CONDITIONS:

VARIABLE	VALUE	STDEV
X2 , M/S2	-0.1205E+00	+,- 0.0000E+00
Y2 , M/S2	-0.8895E-01	+,- 0.0000E+00
H2 , M/S2	0.2402E-01	+,- 0.0000E+00
WX , M/S	0.5003E+00	+,- 0.0000E+00
WY , M/S	0.1078E+01	+,- 0.0000E+00
WH , M/S	0.8971E+00	+,- 0.0000E+00
X1 , M/S	-0.2106E+01	+,- 0.0000E+00
Y1 , M/S	0.9074E+02	+,- 0.0000E+00
H1 , M/S	0.3723E-01	+,- 0.0000E+00
PHI, DEG	0.5849E+00	+,- 0.0000E+00
THT, DEG	0.1443E+01	+,- 0.0000E+00
PSI, DEG	0.9175E+02	+,- 0.0000E+00
X , NM	-0.9120E+00	+,- 0.0000E+00
Y , NM	-0.5467E+00	+,- 0.0000E+00
H , M	0.9999E+03	+,- 0.0000E+00
PH1, D/S	-0.3503E+00	+,- 0.0000E+00
TH1, D/S	-0.5845E-02	+,- 0.0000E+00
PS1, D/S	-0.3616E-01	+,- 0.0000E+00

BIAS ERRORS:

VARIABLE	VALUE	STDEV
AX , G'S	0.0000E+00	+,- 0.0000E+00
AY , G'S	0.0000E+00	+,- 0.0000E+00
AZ , G'S	0.0000E+00	+,- 0.0000E+00
P , D/S	0.0000E+00	+,- 0.0000E+00
Q , D/S	0.0000E+00	+,- 0.0000E+00
R , D/S	0.0000E+00	+,- 0.0000E+00

SCALE FACTORS:

VARIABLE	VALUE	STDEV
P , D/S	0.1000E+01	+,- 0.0000E+00
Q , D/S	0.1000E+01	+,- 0.0000E+00
R , D/S	0.1000E+01	+,- 0.0000E+00

Figure C5.- Concluded.

value of the performance measure), and GRAD (its gradient), printed at each iteration. The other parameters displayed, GAIN and DETM, are, respectively, the multiplier of the step and the determinant of the information matrix. Following the solution by MINC, control is returned to SMACK, which tabulates the solution as shown in figure C8. Listed there are the statistics (mean and rms values) of the residuals (response, or "fit," errors) and forcing-function time histories. A "good" solution is judged not only by robust convergence, but also by how closely rms values in the STDEV columns match corresponding values in the columns headed

(a) DISPLAY OF STATE MODEL ARRAYS AT T=0:

ARRAY: XD

0.0000E+00	0.0000E+00	0.0000E+00	-0.1205E+00	-0.8895E-01	0.2402E-01
0.5003E+00	0.1078E+01	0.8971E+00	-0.2333E-02	-0.4187E-03	0.1182E-02
-0.2106E+01	0.9074E+02	0.3723E-01	0.1021E-01	0.2518E-01	0.1601E+01
-0.1689E+04	-0.1012E+04	0.9999E+03			

ARRAY: WD

0.0000E+00	0.0000E+00	0.0000E+00	0.2169E-01	-0.5074E-02	-0.1343E-02
-0.1469E+00	-0.5226E-01	0.6128E-01	0.0000E+00	0.0000E+00	0.0000E+00
-0.1205E+00	-0.8895E-01	0.2402E-01			

ARRAY: FD

0.0000E+00	0.0000E+00	0.0000E+00	0.2169E-01	-0.5074E-02	-0.1343E-02
-0.1469E+00	-0.5226E-01	0.6128E-01	0.0000E+00	0.0000E+00	0.0000E+00
-0.1205E+00	-0.8895E-01	0.2402E-01	-0.2333E-02	-0.4187E-03	0.1182E-02
-0.2106E+01	0.9074E+02	0.3723E-01			

ARRAY: FX

0.1000E+01	0.1000E+01	0.1000E+01	0.1000E+01	0.1000E+01	0.1000E+01
-0.1000E+01	-0.2571E-03	-0.9999E+00	-0.1021E-01	-0.2519E-01	0.1021E-01
-0.1000E+01	0.2362E-02	0.1045E-06	0.4066E-03	0.4151E-05	-0.2987E-04
0.1210E-04	-0.1186E-02	-0.1055E-04	-0.1182E-02	-0.4188E-03	0.1182E-02
0.2977E-04					

ARRAY: FW

0.1000E+01	0.1000E+01	0.1000E+01	0.1000E+01	0.1000E+01	0.1000E+01
------------	------------	------------	------------	------------	------------

Figure C6.- Display of arrays for flight-test problem. (a) State model arrays at $t = 0$.

($R^{*0.5}$) and ($Q^{*0.5}$). For this problem, the rms values match closely. Notice that the mean of the response error for the accelerometer fits is quite small, which should be the case when instrument bias errors are identified. Notice also that state-variables (PH1, TH1, PS1) are included in the forcing-function list. These were computed from the measured (P, Q, R) (adjusted for bias-error and scale-factor parameters). Finally, the parameters identified in the solution are listed. Their standard deviations are found from the diagonal elements of the information matrix inverse, which are estimates of the Cramer-Rao lower bound. The last step in this solution is to return control to subroutine MODL, where a simulation summary is computed and printed, as shown in figure C9. A comparison of response error and measurement noise can be made for each variable fit to a measurement record. The error between the "true" solution and the estimated solution is given for each variable not measured.

(b) DISPLAY OF OUTPUT MODEL ARRAYS AT T=DT:

ARRAY: HD

0.7874E-02	0.2476E-01	0.1602E+01	-0.1691E+04	-0.9218E+03	0.9999E+03
0.2170E+04	-0.2643E+01	0.4789E+00	0.1085E+01	-0.1903E+01	0.9584E+00
0.8967E+02	0.3479E-01	-0.2306E-02	0.1525E+00	0.2428E-01	-0.9829E+01
-0.4904E-02	0.1831E-04	-0.8545E-03	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00

ARRAY: SD

0.8962E+02	-0.2066E+00	0.3119E+01	0.8962E+02	-0.2066E+00	0.3119E+01
0.1525E+00	0.2428E-01	-0.9829E+01	0.8962E+02	-0.2066E+00	0.3119E+01
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00

ARRAY: HX

0.1000E+01	0.1000E+01	0.1000E+01	0.1000E+01	-0.7793E+00	-0.4248E+00
0.4608E+00	0.2485E-03	-0.4559E-03	0.1864E-03	0.1016E-03	0.4090E-03
0.2877E-01	-0.9995E+00	0.1001E-01	-0.9122E-04	0.1090E-03	0.1115E-01
0.1115E-01	0.3219E-03	0.8720E-04	-0.2877E-01	0.9995E+00	-0.1001E-01
0.9122E-04	-0.1090E-03	-0.1115E-01	-0.1115E-01	-0.3219E-03	-0.8720E-04
0.0000E+00	0.1421E-13	0.0000E+00	0.2303E-02	0.1000E+01	0.7895E-02
0.3480E-01	0.7794E-02	-0.1001E+01	-0.3134E-01	0.9992E+00	0.2476E-01
-0.9995E+00	-0.3116E-01	-0.7872E-02	0.7094E-02	0.2499E-01	-0.9997E+00
0.0000E+00	0.9829E+01	0.1016E+00	-0.9829E+01	0.1201E-02	0.9090E-01
-0.2428E-01	0.1525E+00	0.1802E-02	0.1000E+01	0.1000E+01	0.1000E+01

Figure C6.- Concluded. (b) Measurement model arrays at t = DT.

One final comment should be made here: the fact that rate-gyro measurements have been chosen to generate forcing functions in this problem, instead of estimating (PH2, TH2, PS2) or (DL, DM, DN) and fitting estimates of (P, Q, R) to the rate-gyro records, probably is not important because the noise level on those records is quite small. The results would differ, however, if the measurements of (P, Q, R) were relatively noisy. The user can readily explore this topic by replacing the VAR statements for P, Q, and R with

P	1	1	1	1	0.001
Q	1	1	1	1	0.001
R	1	1	1	1	0.001
PH2	2	1			
TH2	2	1			
PS2	2	1			

and obtain another solution, and then repeat the pair of experiments with a higher rate-gyro noise level. A similar argument would hold for using accelerometer measurements (AX, AY, AZ) to generate forcing functions (X2, Y2, H2). In general, however, it is preferable to fit all measurements, and estimate all forcing functions.

PROBLEM SOLUTION:

COST=	0.7578E+05	GAIN=	0.0000E+00
GRAD=	0.5463E+08	DETM=	0.4058E-11

*** IT= 1 ***

COST=	0.6617E+03	GAIN=	0.1000E+01
GRAD=	0.2696E+06	DETM=	0.4058E-11

*** IT= 2 ***

COST=	0.1637E+02	GAIN=	0.1000E+01
GRAD=	0.1423E+05	DETM=	0.4056E-11

*** IT= 3 ***

COST=	0.1612E+02	GAIN=	0.1000E+01
GRAD=	0.5953E+02	DETM=	0.4057E-11

*** IT= 4 ***

COST=	0.1612E+02	GAIN=	0.1000E+01
GRAD=	0.1163E+00	DETM=	0.4057E-11

*** IT= 5 ***

COST=	0.1612E+02	GAIN=	0.1000E+01
GRAD=	0.5823E-04	DETM=	0.4057E-11

Figure C7.- Iterative solution of flight-test problem.

SOLUTION TABULATION: CPUTIME= 1.2201 SEC

RESPONSE ERRORS AND WEIGHTS:

VARIABLE	MEAN	STDEV	(R**0.5)	FILTER
PHI, DEG	-0.1118E-03 +,-	0.4982E-01	0.5000E-01	0.1500E+00
THT, DEG	0.3307E-03 +,-	0.4845E-01	0.5000E-01	0.1000E+00
PSI, DEG	-0.8621E-03 +,-	0.4942E-01	0.5000E-01	0.1000E+00
H , M	-0.1540E-03 +,-	0.8965E-01	0.1000E+00	0.1000E+00
RNG, NM	-0.2051E-03 +,-	0.4948E-02	0.5000E-02	0.1000E+00
BRG, DEG	0.8718E-03 +,-	0.4913E-01	0.5000E-01	0.1000E+00
ELV, DEG	0.1558E-02 +,-	0.5097E-01	0.5000E-01	0.1000E+00
VT , KT	0.1254E-02 +,-	0.3687E-01	0.1000E+00	0.1000E+00
AV , DEG	-0.5596E-03 +,-	0.3336E-01	0.5000E-01	0.1000E+00
BV , DEG	-0.1288E-02 +,-	0.2725E-01	0.5000E-01	0.1000E+00
AX , G'S	-0.7338E-14 +,-	0.7128E-02	0.1000E-01	0.1000E+00
AY , G'S	0.6943E-15 +,-	0.7489E-02	0.1000E-01	0.1000E+00
AZ , G'S	-0.1932E-14 +,-	0.9844E-02	0.1000E-01	0.1000E+00

FORCING FUNCTIONS AND WEIGHTS:

VARIABLE	MEAN	STDEV	(Q**0.5)
DX , M/S3	0.5323E-03 +,-	0.1735E+00	0.1643E+00
DY , M/S3	-0.1282E-03 +,-	0.9976E-01	0.9321E-01
DH , M/S3	-0.3163E-03 +,-	0.9856E-02	0.9736E-02
GX , M/S2	0.1089E-02 +,-	0.7811E-01	0.8128E-01
GY , M/S2	-0.3480E-02 +,-	0.1017E+00	0.9604E-01
GH , M/S2	0.1208E-03 +,-	0.5334E-01	0.4936E-01
PH1, D/S	-0.1099E-03 +,-	0.7382E+00	0.0000E+00
TH1, D/S	0.2510E-01 +,-	0.1311E+00	0.0000E+00
PS1, D/S	-0.2031E+01 +,-	0.9960E+00	0.0000E+00

Figure C8.- Solution tabulation for flight-test problem.

INITIAL CONDITIONS:

VARIABLE	VALUE	STDEV
X2 , M/S2	0.8228E-02 +,-	0.1789E+00
Y2 , M/S2	-0.6534E-01 +,-	0.1118E+00
H2 , M/S2	0.1010E-01 +,-	0.1851E-01
WX , M/S	0.1684E+01 +,-	0.2110E+00
WY , M/S	-0.4291E+01 +,-	0.2040E+00
WH , M/S	0.9498E+00 +,-	0.7920E-01
X1 , M/S	-0.3882E+00 +,-	0.2849E+00
Y1 , M/S	0.8535E+02 +,-	0.2345E+00
H1 , M/S	-0.5937E-02 +,-	0.5164E-01
PHI, DEG	0.2150E-01 +,-	0.1561E-01
THT, DEG	0.1468E+01 +,-	0.1443E-01
PSI, DEG	0.9130E+02 +,-	0.1908E-01
X , NM	-0.9101E+00 +,-	0.8360E-03
Y , NM	-0.5407E+00 +,-	0.6059E-03
H , M	0.1000E+04 +,-	0.1168E+00
PH1, D/S	-0.1335E+00 +,-	0.0000E+00
TH1, D/S	-0.2478E-01 +,-	0.0000E+00
PS1, D/S	0.6582E-01 +,-	0.0000E+00

BIAS ERRORS:

VARIABLE	VALUE	STDEV
AX , G'S	0.1281E-03 +,-	0.1111E-02
AY , G'S	0.1310E-02 +,-	0.1118E-02
AZ , G'S	0.3850E-03 +,-	0.1059E-02
P , D/S	-0.1228E-03 +,-	0.3238E-03
Q , D/S	0.1412E-02 +,-	0.1332E-02
R , D/S	0.2100E-02 +,-	0.2120E-02

SCALE FACTORS:

VARIABLE	VALUE	STDEV
P , D/S	0.1000E+01 +,-	0.7339E-03
Q , D/S	0.9984E+00 +,-	0.1526E-02
R , D/S	0.1001E+01 +,-	0.9246E-03

Figure C8.- Concluded.

SIMULATION RESULTS:

VARIABLE	ESTIMATION ERROR		MEASUREMENT NOISE	
	MEAN	STDEV	MEAN	STDEV
PHI, DEG	-0.1118E-03	0.4982E-01	-0.3289E-02	0.5000E-01
THT, DEG	0.3307E-03	0.4845E-01	-0.2518E-02	0.5000E-01
PSI, DEG	-0.8621E-03	0.4942E-01	-0.1502E-02	0.5000E-01
X, NM	0.1260E-03	0.4016E-03	0.0000E+00	0.0000E+00
Y, NM	-0.2051E-04	0.3354E-03	0.0000E+00	0.0000E+00
H, M	-0.1540E-03	0.8965E-01	0.1099E-01	0.1000E+00
RNG, NM	-0.2051E-03	0.4948E-02	-0.1634E-03	0.5000E-02
BRG, DEG	0.8718E-03	0.4913E-01	-0.1458E-02	0.5000E-01
ELV, DEG	0.1558E-02	0.5097E-01	0.6007E-03	0.5000E-01
WXY, KT	0.7110E-01	0.2348E+00	0.0000E+00	0.0000E+00
WHD, DEG	0.5131E-01	0.2687E+01	0.0000E+00	0.0000E+00
VWD, M/S	0.9985E-02	0.3355E-01	0.0000E+00	0.0000E+00
VT, KT	0.1254E-02	0.3687E-01	0.4505E-02	0.1000E+00
AV, DEG	-0.5596E-03	0.3336E-01	-0.1008E-01	0.5000E-01
BV, DEG	-0.1288E-02	0.2725E-01	0.5341E-04	0.5000E-01
AX, G'S	-0.7338E-14	0.7128E-02	-0.1700E-03	0.1000E-01
AY, G'S	0.6943E-15	0.7489E-02	0.6972E-03	0.1000E-01
AZ, G'S	-0.1932E-14	0.9844E-02	0.4576E-03	0.1000E-01

Figure C9.- Summary of flight-test problem results.

APPENDIX D

ACCIDENT PROBLEMS

This appendix describes how SMACK can be applied for accident analysis, and illustrates the special output format useful for display of the results. In appendix A (see fig. A2), it was indicated that this option will be chosen when the flag ISOLU=1 (obtained by setting K=1 in the solution description). The purpose of this option, which is produced by subroutine SOLU, is to provide printed time histories of trajectory information, including groundtrack, lift and drag, aircraft attitude, indicated airspeed, and angle of attack. Two accident problems will be given here, each using the simulated trajectory described in chapter 4 and appendix C to provide measurement records for SMACK analysis.

Problem 1

For this first problem, the trajectory data are obtained from two separate tracking stations. There are no records of onboard measurements available, except for barometric altitude (from the aircraft transponder), a record that is usually included with ATC tracking data. Other information available to the analyst consists of air temperature and winds (from balloon measurements made as close as possible to the time and location of the accident), magnetic variation, and aircraft performance parameters. The latter include zero-lift angle of attack, wing loading, and the derivative of the normal-force coefficient, which are necessary for estimation of angle of attack. These parameters are normally defined when the user prepares the DATA subroutine, as described in appendix E. For the SMACK internal simulation, the performance parameters correspond to those of a large commercial airliner.

A coding list for this problem is shown in figure D1(a). Notice that the first directive specifies the accident solution format (K=1 in the MKS statement). The locations of the tracking stations with respect to a user-selected origin are input next, followed by the record information. Notice also that no attitude variables have been included in the coding list. Since there are no onboard measurements available (except altitude), the solution will be more efficient if only the point-mass part of the rigid-body model is excited. The summary of the coding list shown in figure D1(b) illustrates the form of the state and measurement models for this problem. In this special case, the Euler angles are determined following the fits to the radar data, by an algebraic method (ref. 47). This method, coded into subroutine RADR, assumes zero side force and sideslip angle, and uses the performance-parameter derived estimate of angle of attack and the smoothed accident trajectory to calculate the aircraft attitude along the flightpath. The results of the angle calculations are included with the accident output format shown in figure D1(c).

(a) ACCIDENT ANALYSIS-MULTIPLE RADAR TRACK

MKS	5		1	1				
REC	1	90	1	1	1.0			
RAD						0.	0.	100.
RNG	1	1			0.005			
BRG	1	1			0.001			
BRG	4				0.4			
H	1	1			0.2			
H	4				0.075			
WXY	1							
WHD	1							
VWD	1							
RDA						1.	1.	200.
RNA	1	1			0.005			
BRA	1	1			0.005			
BRA	4				0.2			
ELA	1	1			0.005			
ELA	4				0.2			
DX	2	1						
DY	2	1						
DH	2	1						
END		1						

(b) SUMMARY OF CODING LIST:

NF - 9 (DIFFERENTIAL EQUATIONS IN STATE MODEL)
NX - 9 (ESTIMATED VARIABLES IN STATE MODEL)
NW - 3 (ESTIMATED FORCING FUNCTIONS)
NH - 8 (ESTIMATED VARIABLES IN OUTPUT MODEL)
NZ - 9 (AVAILABLE MEASUREMENTS)
NV - 6 (OUTPUT ESTIMATES FIT TO MEASUREMENTS)
NZH- 3 (OUTPUTS MEASURED, NOT ESTIMATED)
NHZ- 11 (NH + NZH)
NZX- 3 (STATES MEASURED, NOT ESTIMATED)
NXZ- 12 (NX + NZX)
NXO- 9 (STATE-VARIABLE INITIAL CONDITIONS)
NWB- 0 (FORCING-FUNCTION MEAN VALUES)
NVB- 0 (INSTRUMENT BIAS ERRORS)
NSF- 0 (INSTRUMENT SCALE FACTORS)
NT - 9 (NXO + NWB + NVB + NSF)

Figure D1.- Accident problem 1 (multiple radar track). (a) Coding list, (b) list summary.

Problem 2

A second problem illustrates an accident solution for a measurement set that includes not only tracking data (from a single station) and winds, but also onboard-instrument data records typically recovered from a metal-foil crash recorder. Such records contain time histories of airspeed, pressure altitude, magnetic heading and vertical acceleration. For this problem, it is assumed that the heading measurement comes from a two-degree-of-freedom directional gyro of the type described in appendix B. A coding list for the measurement set is shown in figure D2(a).

(c) ACCIDENT ANALYSIS RESULTS FOR PROBLEM 1

POINT NO	TIME		ALTITUDE FT	GROUND SPEED KNOTS	TRACK ANGLE DEG	VERT VEL FPS	FLIGHT PATH DEG	FORCES			ANGLES		AIRSPEED			ALPHA DEG
	MIN	SEC						LIFT G'S	T-D G'S	ROLL DEG	PITCH DEG	HEADING DEG	MAG	TRUE KNOTS	IND KNOTS	
1	0	1.00	3280	164.9	90.13	0.02	0.00	1.00	-0.01	-0.56	1.47	91.43	173.8	165.8	2.11	
2	0	2.00	3280	165.1	90.07	0.07	0.01	1.00	-0.01	-0.99	1.46	91.34	173.9	165.9	2.10	
3	0	3.00	3280	165.1	89.95	0.12	0.02	1.00	-0.02	-1.49	1.47	91.19	173.9	165.8	2.10	
4	0	4.00	3281	165.0	89.78	0.18	0.04	1.00	-0.02	-1.67	1.49	90.99	173.8	165.7	2.12	
5	0	5.00	3281	164.9	89.58	0.24	0.05	1.00	-0.03	-2.46	1.51	90.73	173.6	165.6	2.14	
6	0	6.00	3281	164.6	89.30	0.32	0.07	1.00	-0.03	-3.50	1.56	90.37	173.3	165.2	2.19	
7	0	7.00	3281	164.1	88.89	0.42	0.09	1.01	-0.04	-4.78	1.63	89.87	172.8	164.7	2.27	
8	0	8.00	3282	163.6	88.32	0.54	0.11	1.01	-0.04	-6.06	1.72	89.21	172.2	164.2	2.37	
9	0	9.00	3282	163.2	87.60	0.70	0.15	1.01	-0.04	-7.45	1.82	88.38	171.6	163.6	2.47	
10	0	10.00	3283	162.6	86.72	0.92	0.19	1.02	-0.05	-8.95	1.95	87.38	170.9	162.9	2.60	
11	0	11.00	3284	162.0	85.64	1.18	0.25	1.03	-0.06	-10.53	2.11	86.18	170.1	162.1	2.76	
12	0	12.00	3285	161.1	84.36	1.51	0.32	1.03	-0.07	-11.86	2.31	84.77	169.0	161.1	2.95	
13	0	13.00	3287	160.1	82.91	1.89	0.40	1.04	-0.06	-12.85	2.52	83.20	167.8	160.0	3.14	
14	0	14.00	3288	159.3	81.32	2.32	0.50	1.04	-0.07	-13.58	2.73	81.51	166.8	159.0	3.31	
15	0	15.00	3291	158.3	79.62	2.81	0.60	1.05	-0.07	-14.72	2.98	79.68	165.6	157.9	3.53	
16	0	16.00	3294	157.3	77.77	3.33	0.72	1.06	-0.06	-16.29	3.26	77.64	164.4	156.7	3.78	
17	0	17.00	3297	156.5	75.69	3.89	0.84	1.07	-0.06	-17.57	3.52	75.40	163.3	155.7	4.01	
18	0	18.00	3301	155.7	73.43	4.48	0.98	1.08	-0.05	-18.82	3.79	72.96	162.3	154.7	4.24	
19	0	19.00	3305	155.1	70.98	5.10	1.12	1.08	-0.05	-19.53	4.02	70.40	161.4	153.8	4.42	
20	0	20.00	3310	154.5	68.42	5.73	1.26	1.09	-0.04	-20.19	4.26	67.73	160.5	153.0	4.60	
21	0	21.00	3316	154.0	65.77	6.37	1.40	1.09	-0.03	-20.78	4.48	64.96	159.7	152.2	4.76	
22	0	22.00	3322	153.7	63.02	7.01	1.55	1.09	-0.02	-20.87	4.65	62.17	159.2	151.7	4.85	
23	0	23.00	3329	153.6	60.27	7.65	1.69	1.09	-0.02	-21.01	4.80	59.36	158.8	151.3	4.92	
24	0	24.00	3337	153.5	57.49	8.29	1.83	1.09	-0.01	-21.22	4.95	56.53	158.4	150.9	4.99	
25	0	25.00	3345	153.5	54.68	8.91	1.97	1.09	0.00	-21.61	5.08	53.65	158.2	150.7	5.05	
26	0	26.00	3354	153.8	51.83	9.51	2.10	1.09	0.01	-21.64	5.16	50.78	158.2	150.6	5.06	
27	0	27.00	3364	154.1	48.98	10.09	2.22	1.09	0.00	-21.47	5.23	47.93	158.2	150.6	5.04	
28	0	28.00	3374	154.3	46.15	10.65	2.34	1.09	0.00	-21.20	5.30	45.11	158.1	150.5	5.02	
29	0	29.00	3385	154.4	43.38	11.19	2.46	1.09	0.01	-21.11	5.37	42.32	158.0	150.4	5.02	
30	0	30.00	3396	154.6	40.62	11.70	2.57	1.08	0.00	-20.75	5.42	39.58	158.0	150.3	4.99	
31	0	31.00	3407	154.7	37.91	12.18	2.67	1.08	-0.01	-20.44	5.49	36.88	157.8	150.2	4.98	
32	0	32.00	3420	154.5	35.25	12.62	2.77	1.08	-0.01	-20.34	5.58	34.19	157.5	149.8	5.02	
33	0	33.00	3432	154.3	32.61	13.02	2.86	1.08	-0.02	-20.04	5.67	31.54	157.0	149.4	5.05	
34	0	34.00	3445	153.9	30.00	13.36	2.95	1.07	-0.02	-20.08	5.79	28.88	156.4	148.7	5.14	
35	0	35.00	3459	153.4	27.38	13.65	3.02	1.07	-0.02	-20.14	5.92	26.21	155.7	148.1	5.24	
36	0	36.00	3472	152.8	24.75	13.89	3.08	1.07	-0.04	-20.11	6.04	23.53	155.0	147.3	5.34	
37	0	37.00	3486	152.0	22.11	14.09	3.14	1.07	-0.04	-20.22	6.21	20.82	154.0	146.4	5.50	
38	0	38.00	3500	151.0	19.45	14.24	3.20	1.07	-0.05	-20.13	6.39	18.09	152.9	145.3	5.67	
39	0	39.00	3515	150.0	16.78	14.36	3.25	1.07	-0.04	-20.09	6.58	15.36	151.7	144.1	5.86	
40	0	40.00	3529	149.0	14.10	14.44	3.29	1.07	-0.05	-20.22	6.76	12.60	150.6	143.0	6.05	
41	0	41.00	3543	148.0	11.39	14.50	3.32	1.07	-0.04	-20.57	6.96	9.78	149.5	142.0	6.27	
42	0	42.00	3558	147.1	8.61	14.54	3.35	1.07	-0.02	-20.97	7.15	6.89	148.5	141.0	6.49	
43	0	43.00	3572	146.5	5.77	14.57	3.37	1.07	-0.02	-21.12	7.28	3.99	147.8	140.3	6.62	
44	0	44.00	3587	146.1	2.90	14.60	3.39	1.07	-0.01	-21.20	7.38	1.09	147.3	139.8	6.73	
45	0	45.00	3602	145.8	0.02	14.61	3.40	1.07	0.01	-21.33	7.45	358.18	146.8	139.3	6.81	
46	0	46.00	3616	145.9	357.12	14.61	3.40	1.07	0.03	-21.32	7.44	355.32	146.9	139.4	6.79	
47	0	47.00	3631	146.5	354.24	14.60	3.38	1.07	0.05	-21.14	7.34	352.52	147.4	139.8	6.69	
48	0	48.00	3645	147.2	351.40	14.57	3.36	1.07	0.05	-20.98	7.21	349.79	148.1	140.4	6.54	
49	0	49.00	3660	148.1	348.61	14.52	3.33	1.07	0.05	-20.77	7.07	347.11	148.8	141.1	6.38	
50	0	50.00	3674	148.9	345.86	14.45	3.29	1.06	0.06	-20.60	6.92	344.47	149.6	141.8	6.22	
51	0	51.00	3689	150.0	343.16	14.35	3.25	1.06	0.06	-20.45	6.73	341.90	150.6	142.7	6.03	
52	0	52.00	3703	151.1	340.50	14.23	3.19	1.06	0.05	-20.39	6.56	339.36	151.5	143.6	5.86	
53	0	53.00	3718	151.9	337.87	14.06	3.14	1.06	0.05	-20.31	6.42	336.83	152.2	144.2	5.72	
54	0	54.00	3732	152.7	335.26	13.86	3.08	1.06	0.04	-20.37	6.29	334.32	152.9	144.8	5.60	
55	0	55.00	3745	153.4	332.66	13.61	3.01	1.06	0.03	-20.43	6.18	331.80	153.4	145.3	5.52	
56	0	56.00	3759	154.0	330.06	13.32	2.93	1.06	0.02	-20.81	6.10	329.24	153.8	145.6	5.48	
57	0	57.00	3772	154.3	327.41	12.99	2.86	1.06	0.01	-21.06	6.05	326.63	153.8	145.6	5.48	
58	0	58.00	3785	154.5	324.74	12.61	2.77	1.06	0.01	-21.15	6.01	324.01	153.8	145.5	5.49	
59	0	59.00	3798	154.7	322.06	12.18	2.67	1.06	-0.01	-21.46	5.99	321.33	153.6	145.3	5.53	
60	1	0.00	3810	154.5	319.33	11.72	2.57	1.06	-0.01	-21.44	6.00	318.63	153.1	144.8	5.60	
61	1	1.00	3822	154.4	316.61	11.22	2.47	1.06	-0.02	-21.69	6.01	315.89	152.6	144.3	5.70	
62	1	2.00	3833	154.2	313.85	10.70	2.35	1.06	-0.02	-21.92	6.04	313.09	152.0	143.7	5.81	
63	1	3.00	3844	154.1	311.06	10.14	2.23	1.06	-0.03	-22.55	6.09	310.19	151.4	143.1	5.97	
64	1	4.00	3854	153.7	308.17	9.55	2.11	1.06	-0.02	-22.27	6.12	307.28	150.6	142.3	6.07	
65	1	5.00	3863	153.6	305.33	8.94	1.98	1.06	-0.02	-22.03	6.12	304.40	150.0	141.7	6.15	
66	1	6.00	3872	153.5	302.52	8.31	1.84	1.05	-0.01	-21.79	6.11	301.53	149.4	141.2	6.23	
67	1	7.00	3881	153.6	299.75	7.66	1.69	1.06	-0.01	-21.92	6.09	298.64	148.9	140.7	6.32	
68	1	8.00	3888	153.7	296.96	7.01	1.55	1.05	0.00	-21.48	6.04	295.79	148.5	140.3	6.35	
69	1	9.00	3895	154.0	294.24	6.35	1.40	1.05	0.01	-21.01	5.96	293.00	148.4	140.2	6.35	
70	1	10.00	3902	154.5	291.60	5.71	1.25	1.05	0.01	-20.60	5.85	290.27	148.4	140.2	6.32	
71	1	11.00	3907	155.0	289.02	5.07	1.11	1.04	0.02	-19.52	5.71	287.67	148.5	140.3	6.22	
72	1	12.00	3912	155.7	286.59	4.45	0.97	1.03	0.03	-17.98	5.52	285.28	148.8	140.5	6.06	
73	1	13.00	3917	156.6	284.38	3.86	0.84	1.03	0.03	-16.95	5.33	283.04	149.3	141.0	5.91	
74	1	14.00	3921	157.4	282.31	3.31	0.71	1.02	0.04	-15.69	5.14	280.95	149.8	141.5	5.76	
75	1	15.00	3924	158.3	280.42	2.79	0.60	1.02	0.04	-14.44	4.94	279.03	150.5	142.1	5.60	
76	1	16.00	3927	159.3	278.68	2.32	0.50	1.02	0.04	-13.49	4.75	277.24	151.3	142.8	5.44	
77	1	17.00	3929	160.2	277.07	1.91	0.40	1.01	0.04	-12.19	4.57	275.61	152.0	143.6	5.28	
78	1	18.00	3931	161.1	275.62	1.54	0.33	1.01	0.03	-10.70	4.40	274.15	152.8	144.3	5.11	
79	1	19.00	3933	161.8	274.36	1.23	0.26	1.00	0.03	-9.08	4.26	272.90	153.4	144.9	4.98	
80	1	20.00	3934	162.6	273.30	0.97	0.20	1.00</								

(a) ACCIDENT ANALYSIS-RADAR+AIRSPEED, HEADING

MKS	5		1	1				
REC	1	90	1	1	1.0			
RAD						0.	0.	100.
RNG	1	1			0.005			
BRG	1	1			0.001			
BRG	4				0.4			
H	1	1			0.2			
H	4				0.075			
WXY	1							
WHD	1							
VWD	1							
HDG	1	1			0.05			
VT	1	1			0.1			
AX		1						
AY	1	1			0.01			
AZ	1	1	1		0.01			
AV	1	1			0.02			
BV	1	1			0.02			
PHI		1						
THT		1						
PH2	2	1						
TH2	2	1						
PS2	2	1						
DX	2	1						
DY	2	1						
DH	2	1						
END		2						

(b) SUMMARY OF CODING LIST:

NF - 15 (DIFFERENTIAL EQUATIONS IN STATE MODEL)
NX - 15 (ESTIMATED VARIABLES IN STATE MODEL)
NW - 6 (ESTIMATED FORCING FUNCTIONS)
NH - 15 (ESTIMATED VARIABLES IN OUTPUT MODEL)
NZ - 12 (AVAILABLE MEASUREMENTS)
NV - 9 (OUTPUT ESTIMATES FIT TO MEASUREMENTS)
NZH- 3 (OUTPUTS MEASURED, NOT ESTIMATED)
NHZ- 18 (NH + NZH)
NZX- 3 (STATES MEASURED, NOT ESTIMATED)
NXZ- 18 (NX + NZX)
NXO- 15 (STATE-VARIABLE INITIAL CONDITIONS)
NWB- 0 (FORCING-FUNCTION MEAN VALUES)
NVB- 1 (INSTRUMENT BIAS ERRORS)
NSF- 0 (INSTRUMENT SCALE FACTORS)
NT - 16 (NXO + NWB + NVB + NSF)

Figure D2.- Accident problem 2 (radar, airspeed, heading). (a) Coding list, (b) list summary.

Because there are now body-axis measurements available, the complete rigid-body aircraft model can be specified in the coding list. Note, however, that the data

(c) ACCIDENT ANALYSIS RESULTS FOR PROBLEM 2

POINT NO	MIN		ALTITUDE FT	GROUND SPEED KNOTS	TRACK ANGLE DEG	VERT VEL FPS	FLIGHT PATH DEG	FORCES		ROLL DEG	ANGLES		AIRSPEED		ALPHA DEG
	MIN	SEC						LIFT G'S	T-D G'S		PITCH DEG	HEADING DEG	TRUE KNOTS	IND KNOTS	
1	0	1.00	3280.	165.2	89.94	0.00	0.00	1.00	-0.01	0.63	1.45	91.31	174.1	166.0	2.09
2	0	2.00	3280.	165.2	90.01	0.06	0.01	1.00	-0.01	-0.11	1.44	91.31	174.1	166.0	2.08
3	0	3.00	3280.	165.2	89.95	0.11	0.02	1.00	-0.02	-0.07	1.45	91.19	174.0	165.9	2.08
4	0	4.00	3281.	165.1	89.80	0.15	0.03	1.00	-0.02	-1.67	1.47	91.02	173.9	165.8	2.11
5	0	5.00	3281.	164.9	89.59	0.21	0.04	1.00	-0.02	-2.52	1.52	90.76	173.7	165.6	2.16
6	0	6.00	3281.	164.7	89.24	0.28	0.06	1.00	-0.03	-3.45	1.57	90.33	173.4	165.3	2.22
7	0	7.00	3281.	164.3	88.86	0.39	0.08	1.01	-0.04	-4.47	1.63	89.82	172.9	164.8	2.28
8	0	8.00	3282.	163.7	88.31	0.54	0.11	1.01	-0.04	-5.57	1.70	89.19	172.2	164.2	2.36
9	0	9.00	3282.	163.3	87.54	0.73	0.15	1.01	-0.05	-6.76	1.81	88.35	171.7	163.7	2.45
10	0	10.00	3283.	162.7	86.65	0.95	0.20	1.02	-0.05	-8.03	1.95	87.34	170.9	163.0	2.59
11	0	11.00	3284.	162.0	85.59	1.23	0.26	1.03	-0.06	-9.35	2.11	86.19	170.1	162.1	2.74
12	0	12.00	3285.	161.1	84.35	1.56	0.33	1.03	-0.07	-10.71	2.29	84.85	169.0	161.1	2.90
13	0	13.00	3287.	160.1	82.93	1.93	0.41	1.04	-0.06	-12.07	2.50	83.28	167.9	160.1	3.09
14	0	14.00	3289.	159.3	81.36	2.36	0.50	1.04	-0.07	-13.40	2.74	81.58	166.8	159.1	3.30
15	0	15.00	3291.	158.3	79.62	2.83	0.61	1.05	-0.07	-14.69	3.00	79.66	165.6	157.9	3.55
16	0	16.00	3294.	157.3	77.66	3.36	0.72	1.06	-0.06	-15.91	3.25	77.58	164.4	156.7	3.76
17	0	17.00	3297.	156.5	75.60	3.92	0.85	1.07	-0.06	-17.04	3.51	75.36	163.3	155.7	3.97
18	0	18.00	3301.	155.7	73.34	4.50	0.98	1.07	-0.05	-18.08	3.76	72.96	162.3	154.7	4.18
19	0	19.00	3306.	155.0	70.95	5.10	1.12	1.08	-0.05	-19.01	4.01	70.43	161.3	153.8	4.38
20	0	20.00	3311.	154.5	68.41	5.71	1.26	1.09	-0.04	-19.80	4.24	67.79	160.5	153.0	4.56
21	0	21.00	3316.	154.0	65.76	6.33	1.40	1.09	-0.03	-20.42	4.46	65.03	159.7	152.2	4.71
22	0	22.00	3323.	153.7	63.06	6.96	1.54	1.09	-0.02	-20.86	4.65	62.22	159.2	151.7	4.85
23	0	23.00	3330.	153.6	60.30	7.59	1.68	1.09	-0.01	-21.14	4.82	59.37	158.8	151.3	4.95
24	0	24.00	3337.	153.5	57.48	8.23	1.82	1.10	-0.01	-21.26	4.96	56.52	158.5	151.0	5.01
25	0	25.00	3345.	153.6	54.62	8.85	1.96	1.10	0.00	-21.29	5.07	53.62	158.2	150.7	5.04
26	0	26.00	3354.	153.7	51.76	9.46	2.09	1.09	0.01	-21.25	5.16	50.72	158.1	150.6	5.06
27	0	27.00	3364.	154.0	48.90	10.05	2.21	1.09	0.00	-21.18	5.23	47.87	158.1	150.6	5.04
28	0	28.00	3374.	154.2	46.12	10.62	2.34	1.09	0.01	-21.09	5.30	45.11	158.1	150.5	5.02
29	0	29.00	3384.	154.4	43.36	11.17	2.45	1.09	0.01	-20.99	5.37	42.32	158.0	150.4	5.02
30	0	30.00	3396.	154.6	40.58	11.69	2.57	1.08	0.00	-20.88	5.43	39.55	157.9	150.3	5.00
31	0	31.00	3407.	154.6	37.88	12.18	2.67	1.08	0.00	-20.75	5.50	36.81	157.7	150.1	5.01
32	0	32.00	3419.	154.5	35.21	12.64	2.77	1.08	-0.01	-20.61	5.57	34.13	157.4	149.8	5.02
33	0	33.00	3432.	154.3	32.61	13.04	2.87	1.07	-0.02	-20.48	5.66	31.50	157.0	149.4	5.06
34	0	34.00	3445.	153.8	30.02	13.39	2.95	1.07	-0.02	-20.38	5.78	28.88	156.3	148.7	5.13
35	0	35.00	3459.	153.4	27.43	13.69	3.03	1.07	-0.02	-20.30	5.91	26.23	155.7	148.1	5.23
36	0	36.00	3472.	152.8	24.82	13.93	3.09	1.07	-0.04	-20.28	6.06	23.62	154.9	147.3	5.34
37	0	37.00	3486.	151.9	22.21	14.13	3.16	1.07	-0.04	-20.29	6.21	20.91	153.9	146.3	5.50
38	0	38.00	3500.	151.1	19.55	14.29	3.21	1.07	-0.05	-20.35	6.38	18.18	152.9	145.3	5.66
39	0	39.00	3515.	150.0	16.88	14.39	3.25	1.07	-0.05	-20.48	6.58	15.42	151.7	144.2	5.86
40	0	40.00	3529.	149.0	14.18	14.46	3.29	1.07	-0.04	-20.64	6.78	12.61	150.6	143.0	6.09
41	0	41.00	3543.	148.0	11.40	14.50	3.32	1.07	-0.04	-20.81	6.97	9.75	149.5	142.0	6.29
42	0	42.00	3558.	147.2	8.58	14.53	3.35	1.07	-0.03	-20.97	7.13	6.87	148.5	141.0	6.47
43	0	43.00	3572.	146.5	5.73	14.54	3.37	1.07	-0.02	-21.12	7.27	3.96	147.8	140.3	6.62
44	0	44.00	3587.	146.1	2.90	14.54	3.38	1.07	-0.01	-21.22	7.37	1.09	147.2	139.7	6.73
45	0	45.00	3602.	145.8	0.01	14.53	3.38	1.07	0.01	-21.23	7.42	358.21	146.9	139.4	6.78
46	0	46.00	3616.	146.0	357.13	14.54	3.38	1.07	0.03	-21.14	7.40	355.34	147.0	139.4	6.77
47	0	47.00	3631.	146.5	354.24	14.54	3.37	1.07	0.04	-20.98	7.33	352.55	147.4	139.8	6.67
48	0	48.00	3645.	147.2	351.40	14.53	3.35	1.07	0.05	-20.79	7.22	349.80	148.0	140.4	6.55
49	0	49.00	3660.	148.1	348.65	14.51	3.32	1.07	0.05	-20.59	7.08	347.14	148.9	141.2	6.39
50	0	50.00	3674.	149.0	345.90	14.47	3.29	1.06	0.06	-20.42	6.92	344.50	149.6	141.8	6.23
51	0	51.00	3689.	150.0	343.21	14.39	3.25	1.06	0.06	-20.28	6.76	341.97	150.6	142.7	6.04
52	0	52.00	3703.	151.1	340.52	14.28	3.21	1.06	0.05	-20.18	6.60	339.40	151.5	143.6	5.88
53	0	53.00	3717.	151.9	337.87	14.12	3.15	1.06	0.05	-20.14	6.45	336.83	152.3	144.2	5.74
54	0	54.00	3731.	152.8	335.23	13.92	3.09	1.06	0.04	-20.15	6.32	334.31	152.9	144.8	5.61
55	0	55.00	3745.	153.5	332.64	13.67	3.02	1.06	0.03	-20.21	6.20	331.77	153.4	145.3	5.53
56	0	56.00	3759.	154.0	330.02	13.38	2.95	1.06	0.02	-20.32	6.11	329.27	153.8	145.6	5.46
57	0	57.00	3772.	154.3	327.39	13.04	2.87	1.06	0.01	-20.49	6.05	326.67	153.9	145.6	5.44
58	0	58.00	3785.	154.6	324.75	12.66	2.78	1.06	0.00	-20.74	6.00	324.04	153.8	145.6	5.46
59	0	59.00	3798.	154.6	322.05	12.22	2.68	1.06	-0.01	-21.04	5.98	321.37	153.6	145.3	5.50
60	1	0.00	3810.	154.5	319.34	11.73	2.58	1.06	-0.01	-21.41	5.98	318.66	153.1	144.8	5.58
61	1	1.00	3822.	154.4	316.62	11.21	2.46	1.06	-0.02	-21.76	6.00	315.91	152.6	144.4	5.68
62	1	2.00	3833.	154.3	313.88	10.65	2.34	1.06	-0.02	-22.05	6.03	313.12	152.0	144.1	5.81
63	1	3.00	3844.	154.0	311.05	10.07	2.22	1.06	-0.02	-22.26	6.06	310.23	151.3	143.1	5.94
64	1	4.00	3854.	153.8	308.22	9.47	2.09	1.06	-0.02	-22.39	6.10	307.29	150.6	142.4	6.09
65	1	5.00	3863.	153.6	305.36	8.85	1.96	1.06	-0.01	-22.41	6.11	304.39	150.0	141.8	6.19
66	1	6.00	3872.	153.6	302.50	8.23	1.82	1.06	-0.02	-22.33	6.11	301.47	149.5	141.2	6.26
67	1	7.00	3881.	153.6	299.69	7.60	1.68	1.05	-0.01	-22.11	6.07	298.57	148.9	140.7	6.32
68	1	8.00	3888.	153.7	296.94	6.97	1.54	1.05	0.00	-21.73	6.03	295.71	148.6	140.4	6.37
69	1	9.00	3895.	154.0	294.19	6.35	1.40	1.05	0.01	-21.19	5.95	292.90	148.4	140.2	6.36
70	1	10.00	3901.	154.5	291.53	5.72	1.26	1.04	0.01	-20.52	5.86	290.20	148.4	140.2	6.32
71	1	11.00	3907.	155.0	289.01	5.11	1.12	1.04	0.02	-19.71	5.72	287.62	148.5	140.3	6.24
72	1	12.00	3912.	155.7	286.64	4.52	0.99	1.03	0.03	-18.77	5.55	285.21	148.8	140.5	6.12
73	1	13.00	3917.	156.6	284.40	3.95	0.86	1.03	0.03	-17.70	5.36	282.95	149.3	141.0	5.96
74	1	14.00	3921.	157.4	282.26	3.40	0.73	1.02	0.04	-16.55	5.16	280.80	149.9	141.5	5.79
75	1	15.00	3924.	158.3	280.35	2.88	0.62	1.02	0.03	-15.32	4.96	278.87	150.6	142.2	5.62
76	1	16.00	3927.	159.2	278.56	2.40	0.51	1.01	0.04	-14.02	4.78	277.10	151.2	142.8	5.45
77	1	17.00	3929.	160.2	276.97	1.96	0.41	1.01	0.04	-12.66	4.60	275.47	152.0	143.5	5.30
78	1	18.00	3931.	161.1	275.58	1.56	0.33	1.01	0.03	-11.27	4.42	274.07	152.8	144.2	5.14
79	1	19.00	3933.	161.8	274.40	1.23	0.26	1.00	0.03	-9.86	4.26	272.86	153.4	144.8	4.99
80	1	20.00	3934.	162.6	273.38	0.95	0.20	1							

set is still not sufficient to allow the removal of the assumption of zero side force and sideslip angle. For the simulated trajectory, which consists of a fully-coordinated turn, the assumption is justified. The analyst puts these constraints on the problem by declaring measurements of AY and BV as indicated in the coding list, and placing zeros in the corresponding time-history arrays (see appendix E). The coding-list summary given in figure D2(b) shows that the proper attitude and position variables are excited in the state and measurement models.

The output format for this problem is shown in figure D2(c). One might expect better estimates of the roll and pitch angle time histories in this problem compared to the previous case because of the additional measurements. The improvement actually realized, however, depends on the quality of the additional measurements, as well as on the weights assigned to the "pseudo" measurements (AY and BV) used to match the assumptions made in solving the first problem. A comparison of the two solutions can be made by observing the simulation results for the two cases shown in figure D3. The largest improvement is seen to be in the estimate of roll angle, PHI. This is illustrated by the roll-angle plots of the estimates compared with the "true" time histories shown in figure D4.

(a)	ESTIMATION ERROR		MEASUREMENT NOISE	
	VARIABLE	MEAN	STDEV	MEAN
X , NM	-0.3736E-04	+,- 0.1092E-03	0.0000E+00	+,- 0.0000E+00
Y , NM	-0.2122E-04	+,- 0.4867E-04	0.0000E+00	+,- 0.0000E+00
H , M	0.2749E-02	+,- 0.1952E+00	-0.1316E-01	+,- 0.2000E+00
RNG, NM	-0.2279E-03	+,- 0.5026E-02	-0.2518E-03	+,- 0.5000E-02
BRG, DEG	-0.1131E-04	+,- 0.4740E-03	-0.3003E-04	+,- 0.1000E-02
RNA, NM	0.5844E-03	+,- 0.4995E-02	0.5494E-03	+,- 0.5000E-02
BRA, DEG	0.5934E-04	+,- 0.4028E-02	-0.1634E-03	+,- 0.5000E-02
ELA, DEG	-0.1537E-03	+,- 0.4759E-02	-0.1458E-03	+,- 0.5000E-02
PHI, DEG	0.3442E-01	+,- 0.4432E+00	0.0000E+00	+,- 0.0000E+00
THT, DEG	-0.7822E-03	+,- 0.1983E-01	0.0000E+00	+,- 0.0000E+00
PSI, DEG	-0.6208E-02	+,- 0.6060E-01	0.0000E+00	+,- 0.0000E+00
VT , KT	0.9709E-02	+,- 0.6211E-01	0.0000E+00	+,- 0.0000E+00
AV , DEG	0.1251E-02	+,- 0.2247E-01	0.0000E+00	+,- 0.0000E+00
AX , G'S	-0.8864E-04	+,- 0.3371E-02	0.0000E+00	+,- 0.0000E+00
AZ , G'S	-0.5054E-04	+,- 0.1294E-02	0.0000E+00	+,- 0.0000E+00

(b)	ESTIMATION ERROR		MEASUREMENT NOISE	
	VARIABLE	MEAN	STDEV	MEAN
PHI, DEG	-0.1833E-01	+,- 0.2263E+00	0.0000E+00	+,- 0.0000E+00
THT, DEG	0.1263E-02	+,- 0.1888E-01	0.0000E+00	+,- 0.0000E+00
PSI, DEG	-0.2547E-03	+,- 0.2949E-01	0.0000E+00	+,- 0.0000E+00
X , NM	-0.4934E-05	+,- 0.8413E-04	0.0000E+00	+,- 0.0000E+00
Y , NM	-0.2701E-04	+,- 0.7930E-04	0.0000E+00	+,- 0.0000E+00
H , M	0.1463E-03	+,- 0.1915E+00	-0.1316E-01	+,- 0.2000E+00
RNG, NM	-0.3046E-03	+,- 0.5002E-02	-0.2518E-03	+,- 0.5000E-02
BRG, DEG	0.2111E-05	+,- 0.6532E-03	-0.3003E-04	+,- 0.1000E-02
VT , KT	0.4474E-03	+,- 0.8654E-01	0.1099E-01	+,- 0.1000E+00
AV , DEG	-0.1903E-03	+,- 0.1193E-01	-0.6538E-03	+,- 0.2000E-01
BV , DEG	-0.7746E-03	+,- 0.7100E-02	-0.5832E-03	+,- 0.2000E-01
AX , G'S	0.1017E-03	+,- 0.2771E-02	0.0000E+00	+,- 0.0000E+00
AY , G'S	0.2260E-03	+,- 0.7848E-02	0.1201E-03	+,- 0.1000E-01
AZ , G'S	-0.2969E-13	+,- 0.9923E-02	0.4505E-03	+,- 0.1000E-01
HDG, DEG	-0.4722E-02	+,- 0.3295E-01	-0.1008E-01	+,- 0.5000E-01

Figure D3.- Summaries of accident solution results (a) problem 1,
b) problem 2.

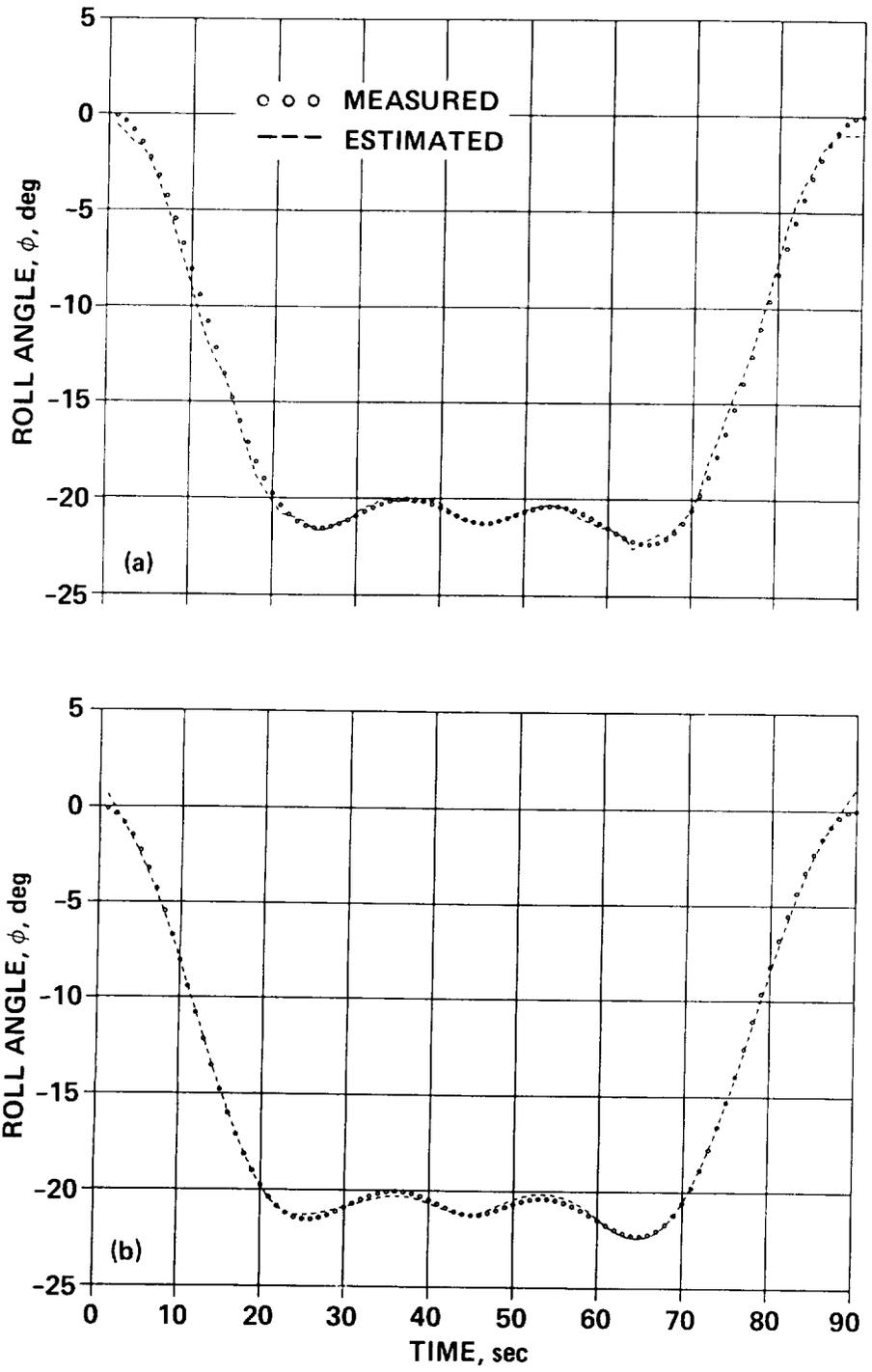


Figure D4.- Roll-angle estimates from accident problems. (a) Problem 1, (b) problem 2.

APPENDIX E

USER SUBROUTINE

This appendix describes the subroutine DATA that the user must prepare to provide the link between a source of flight data and the SMACK state estimation program. Some chores the user may wish to perform in DATA include wild-point windowing, amplitude scaling, sample compression, and air-data calculations. The user may prefer, of course, to do these chores separately and provide a file of time histories prepared especially for easy processing by subroutine DATA. Compensation for cg offset of accelerometer and air-data measurement systems is optional in DATA, since position errors may be specified by including ACC, P-S, and VNE statements in the coding list. The first few lines of code for subroutine DATA should appear as follows:

```
SUBROUTINE DATA (IFIN)
```

```
COMMON/MSM/Z(30,6000),Y(30,6000),KZN(30,6000),  
*      H(30,6000),D(30,6000), S(30,6000)  
COMMON/STM/X(21,6000),W( 9,6000),DMW(54000)  
COMMON/AUX/XN(6000),YN(6000),ZN(6000),FN(6000),  
*      KFN(6000),NFN(6000)
```

```
COMMON/SKL/CMK,CMF,CRD,CNL,CMM,CMG  
COMMON/CON/RGAS,TSSL,RHOO,PI,C(30)  
COMMON/TIM/DT,NPTS,NINT,NSMP,NSKP,MPTS,FC1  
COMMON/MOD/ALFO,CNAO,WGLD,VARM
```

These arrays and parameters are defined as:

```
Z(30,6000)  Flight data array to be filled  
KZN(30,6000) Companion array to be filled  
Y(30,6000)  Output estimates returned with solution  
H(30,6000)  Output estimates returned with solution  
D(30,6000)  Auxiliary array  
S(30,6000)  Auxiliary array  
  
X(21,6000)  State-variable estimates returned with solution  
W( 9,6000)  Forcing-function estimates returned with solution  
DMW(54000)  Auxiliary array  
  
XN(6000)   Auxiliary array  
YN(6000)   Auxiliary array  
ZN(6000)   Auxiliary array  
FN(6000)   Auxiliary array
```

KFN(6000) Auxiliary array
NFN(6000) Auxiliary array

CMK Conversion to meter/sec from knots (0.5144)
CMF Conversion to meters from feet (0.3048)
CRD Conversion to radians from degrees (0.01745)
CNL Conversion to newtons from pounds (4.4482)
CMM Conversion to meters from nautical miles (1852.)
CMG Conversion to meter/s² from g units (9.807)
DT Time step for integrating equations of motion, seconds
NPTS Number of time steps per record (maximum is MPTS)
NINT Number of integration steps per datum point
NSMP Number of datum points per record
NSKP Plotting interval for output
MPTS Maximum number of time steps per record (6000)
FC1 Low-pass-filter cutoff frequency

ALFO Zero-lift angle of attack, radians
CNAO Normal force derivative, /radian
WGLD Wing loading, newton/m²
VARM Magnetic variation, radians

RGAS Gas constant (286.924 m²/s²/deg K)
TSSL Sea-level temperature (288.15 deg K)
RHO0 Air density at sea level (1.225 kg/m³)
PI Value of Pi (3.14159265 radians)
C(30) Array of constants set by coding list

Subroutine DATA is called by SMACK twice, once to fill the appropriate rows of the Z and KZN arrays with measurement records, and again after the solution has been completed. Entry to DATA is determined by the value of IFIN (0 or 1). Each entry must be accompanied by a separate RETURN statement.

Both the Z and KZN arrays are initialized to zero before SMACK calls DATA the first time (IFIN = 0). When a "good" datum point is entered in a row of Z, a 1 should be entered in the corresponding location of KZN. In this way channels with wild points and/or nonuniform sampling rates can be handled in a mathematically correct way (no weight for a bad or missing point). Rows of the flight-data arrays Z and KZN are defined in table 5.1. Note that internally, all rotational and linear motion variables are in radian and mks units. The conversion constants CMK,---,CMG and the other constants in block /CON/ are provided for user convenience. The auxiliary time-history arrays may be used as temporary storage in performing necessary data-processing chores.

As indicated in chapter 5 which covers the preparation of the coding list, DT is the time step for integration of the state equations and NPTS is the total number of points in each state (and output) estimate, not including the initial condition. Although these parameters (and NINT, NSMP, and NSKP in block /TIM/) are

normally set by the REC statement in the coding list, they may be reset in subroutine DATA. The user should be aware that each data record (row of Z, KZN) must have samples that occur at integer multiples of DT. Nonuniform sample rates are permissible as long as every interval is divisible by DT. It should be emphasized that for each "good" datum point entered in a row of Z, a 1 should be entered in the corresponding location of KZN.

The aircraft-dependent performance parameters ALFO, CNAO, and WGLD in block /MOD/ need to be specified in DATA only when an estimate of angle of attack is needed but no measurement is available, as in an accident analysis from radar track and wind data. For example, for small values the estimate can be approximated by

$$AV = ALFO - WGLD * AZ / (CUE * CNAO) \quad (E1)$$

where AV is angle of attack, AZ is vertical acceleration, and CUE is dynamic pressure. This estimate is provided by SMACK for the radar solution illustrated by the first problem of appendix D; in other situations the user should compute the pseudo measurement of angle of attack within DATA, when necessary.

Note that IFIN is zero when DATA is first called. When the solution has been completed, IFIN is set to one and DATA is again called. At this time the user may wish to perform other operations with the data or the output estimates, which are returned in the arrays H and Y. The ith row of the output estimate Y is calculated in SMACK as

$$Y(I,N) = SF(I) * H(I,N) + VB(I), \quad N=1, \dots, NPTS \quad (E2)$$

where SF(I) is the scale factor, VB(I) is the bias error and H(I,N) is the output variable, with rows in the order specified by table 5.1. The estimates of aerodynamic variables (VT, AV, BV) and specific forces (AX, AY, AZ) stored in array H may include position corrections. The eg-referenced time histories of true airspeed, angle of attack, and sideslip angle may be calculated from the air velocities (UA, VA, WA) stored in rows 1-3 of array S. The true specific forces (AXB, AYB, AZB) are stored in rows 4-6 of array S.

If it is necessary to compress, filter, or interpolate any records in DATA, the user may call subroutine FILT. As indicated in appendix A, this subroutine is used in the SMACK starting procedure to provide filtering of each measurement time history. The amplitude-frequency characteristic is that of a fourth-order low-pass filter with zero phase shift. FILT is based on the backward-filter, forward-smoother algorithm used by SMACK, adapted for linear state and measurement models, which is discussed in chapter 2 and outlined in table 2.2(b). The filter model and its calling statement are described in the following section.

A Digital Filter

The low-pass digital filtering subroutine FILT is called several times during the SMACK starting procedure. Since the user may have occasion to call FILT from

the DATA subroutine, its characteristics and calling statement are discussed here. The filter design is based on the solution properties of a fixed-interval smoothing problem for a linear, single-input, single-output (SISO), nth-order continuous system. It is well known that the solution of this problem yields a filter transfer function with $2*n$ poles equally distributed on a circle centered at the origin of the s-plane. The resulting frequency characteristic, therefore, is equivalent to two nth-order Butterworth filters in cascade, each with equal bandwidth and equal but opposite phase shift at all frequencies. Hence, the filter response is $3*n$ db down at the cutoff frequency, rolls off at $12*n$ db/octave, and exhibits no phase shift.

The linear continuous problem and its solution are illustrated in figure E1 for a second order system ($n=2$). Notice that the forcing function is the second derivative of the filter output. A finite-difference approximation for this case is used with the linear algorithm of table 2.2(b) as the basis for subroutine FILT. It has a state model with

$$F = \begin{bmatrix} 1 & h & h^2/2 \\ 0 & 1 & h \\ 0 & 0 & 1 \end{bmatrix}; \quad G = \begin{bmatrix} h^2/2 \\ h \\ 0 \end{bmatrix} \quad (E3)$$

and a measurement model with

$$H = (1 \ 0 \ 0) \quad (E4)$$

For an SISO system, the weighting matrices Q and R are scalars: the filter realization determines the ratio to be

$$Q/R = (2\pi f_c h)^4 \quad (E5)$$

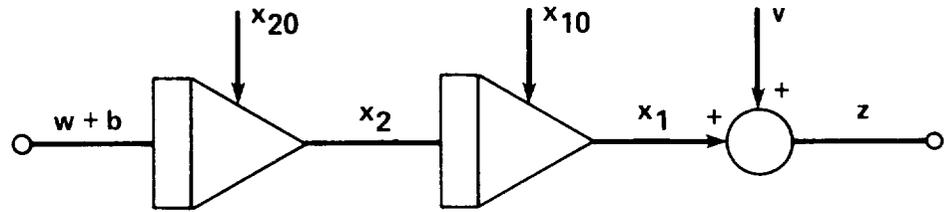
where h is the integration time step (in seconds), and f_c is the filter cutoff frequency (in Hz). The resulting response will be 6 db down at f_c , and the attenuation rate will be 24 db/octave. The extra state in both the continuous and discrete state models is added to accommodate an average value of the forcing function. It does not affect the second-order nature of the response. The calling statement for FILT is

```
CALL FILT(FN, KFN, XN, YN, WN, XO, YO, WO, FC, H, NPTS)
```

where

- FN Data record to be filtered (destroyed on return)
- KFN Companion array: each "good" point of FN should have a corresponding entry of KFN set to one; otherwise the entry should be set to zero

(a) FIXED-INTERVAL SMOOTHING PROBLEM:



STATE MODEL: $\dot{x}_1 = x_2$, $x_1(0) = x_{10}$
 $\dot{x}_2 = w + b$, $x_2(0) = x_{20}$
 $\dot{b} = 0$, $b(0) = b$

MEASUREMENT MODEL: $z = x_1 + v$, FOR $(0, T)$

CHOOSE x_{10} , x_{20} , b AND FORCING-FUNCTION $w(t)$ TO MINIMIZE:

$$J = (1/2) \int_0^T [w^2/q + (z - x_1)^2/r] dt$$

(b) SOLUTION PROPERTIES:

TRANSFER FUNCTION:

$$H(s) = X_1(s)/Z(s) = \omega_c^4 / (s^4 + \omega_c^4)$$

WHERE $\omega_c = 2\pi f_c = (q/r)^{1/4}$

FREQUENCY RESPONSE:

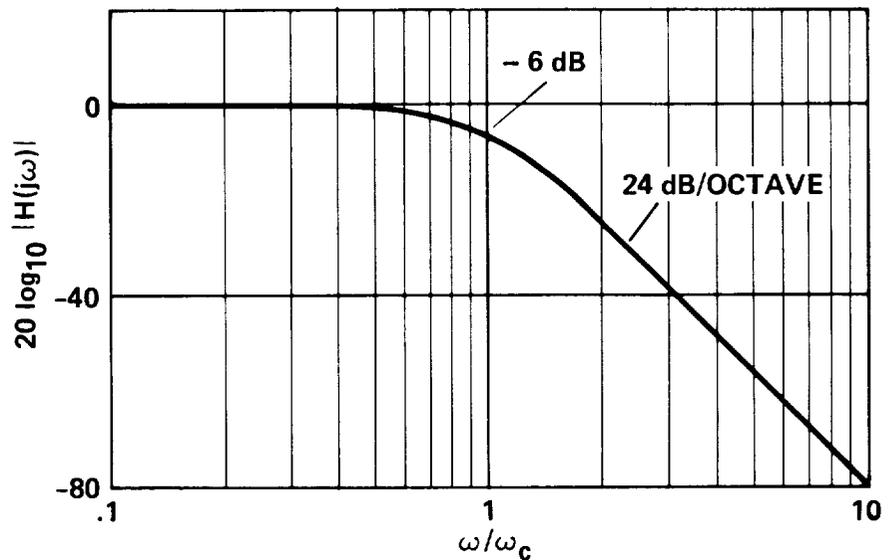
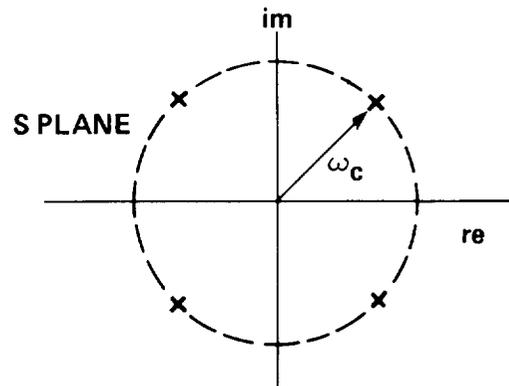
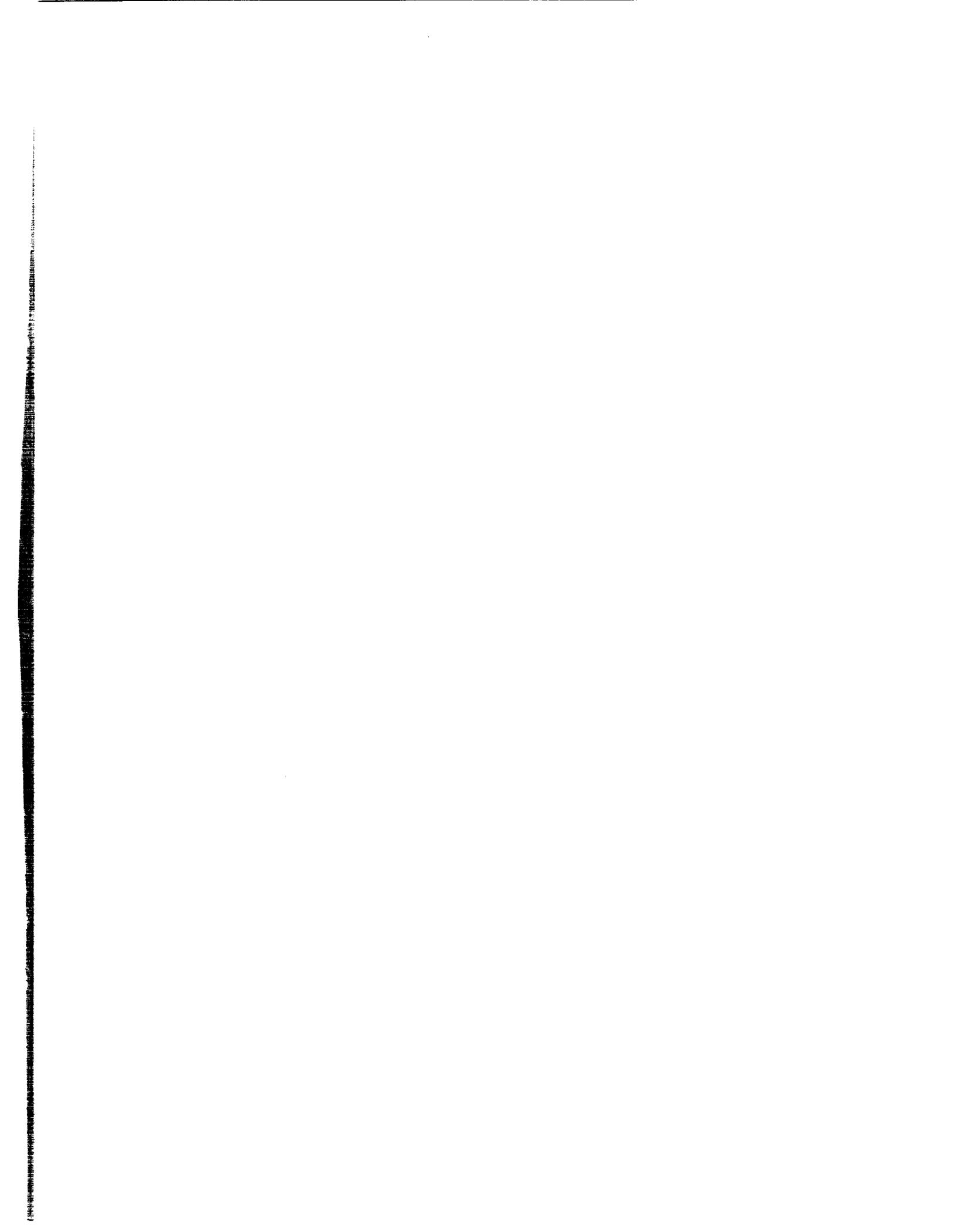


Figure E1.- Basis for digital-filtering subroutine FILT. (a) Fixed-interval smoothing problem, (b) fourth-order frequency characteristic.

XN Filtered estimate of FN (array of MPTS)
YN First derivative of XN (array of MPTS)
WN Second derivative of XN (array of MPTS)
XO Initial condition of XN
YO Initial condition of YN
WO Initial condition of WN
FC Low-pass cutoff frequency, in Hz
H Integration step, in seconds
NPTS Number of integration steps ($NPTS \leq MPTS$)

When FILT is used as an interpolating filter, the cutoff frequency FC should be set less than or equal to one-half the sampling frequency (H is the data-sample interval). If data compression is required after filtering, the final sampling rate should be greater than or equal to twice FC. To obtain good results with FILT, the original record length NPTS should be greater than 50 points.

1. Report No. NASA RP-1252		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle State Estimation Applications in Aircraft Flight-Data Analysis <i>A User's Manual for SMACK</i>				5. Report Date March 1991	
				6. Performing Organization Code	
7. Author(s) Ralph E. Bach, Jr.				8. Performing Organization Report No. A-88203	
				10. Work Unit No. 505-66-41	
9. Performing Organization Name and Address Ames Research Center Moffett Field, CA 94035-1000				11. Contract or Grant No.	
				13. Type of Report and Period Covered Reference Publication	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001				14. Sponsoring Agency Code	
15. Supplementary Notes Point of Contact: Ralph E. Bach, Jr., Ames Research Center, MS 210-9 Moffett Field, CA 94035-1000 (415) 604-5429 or FTS 464-5429					
16. Abstract <p>This report traces the evolution in the use of state estimation for the analysis of aircraft flight data. A unifying mathematical framework for state estimation is reviewed, and several examples are presented that illustrate a general approach for checking instrument accuracy and data consistency, and for estimating variables that are difficult to measure. Recent applications associated with research-aircraft flight tests and airline turbulence upsets are described. A computer program for aircraft state estimation is discussed in some detail. This document is intended to serve as a user's manual for the program, called SMACK (SMoothing for AirCRAFT Kinematics). The diversity of the applications described in the report emphasizes the potential advantages in using SMACK for flight-data analysis.</p>					
17. Key Words (Suggested by Author(s)) Flight-data analysis State estimation Smoothing			18. Distribution Statement Unclassified-Unlimited Subject Category - 05		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 132	22. Price A07



National Aeronautics and
Space Administration
Code NTT-4

Washington, D.C.
20546-0001

Official Business
Penalty for Private Use, \$300

BULK RATE
POSTAGE & FEES PAID
NASA
Permit No. G-27

NASA

POSTMASTER: If Undeliverable (Section 158
Postal Manual) Do Not Return
